
ELSEVIER
Foundations of ARTIFICIAL INTELLIGENCE
SERIES EDITORS: J. HENDLER, H. KITANO, B. NEBEL
1
Handbook of Temporal Reasoning in Artificial Intelligence
EDITORS
M. FISHER
D. GABBAY
L. VILA

FOUNDATIONS OF
ARTIFICIAL INTELLIGENCE
VOLUME 1

# Foundations of Artificial Intelligence

VOLUME 1

*Series Editors*

J. Hendler
H. Kitano
B. Nebel

ELSEVIER
AMSTERDAM–BOSTON–HEIDELBERG–LONDON–NEW YORK–OXFORD
PARIS–SAN DIEGO–SAN FRANCISCO–SINGAPORE–SYDNEY–TOKYO

# Handbook of Temporal Reasoning in Artificial Intelligence

Edited by

M. Fisher
Department of Computer Science
University of Liverpool
Liverpool, UK

D. Gabbay
Department of Computer Science
King's College London
London, UK

L. Vila
Department of Software
Technical University of Catalonia
Barcelona, Catalonia, Spain

2005
ELSEVIER
AMSTERDAM–BOSTON–HEIDELBERG–LONDON–NEW YORK–OXFORD
PARIS–SAN DIEGO–SAN FRANCISCO–SINGAPORE–SYDNEY–TOKYO

**ELSEVIER B.V.**
**Radarweg 29**
**P.O. Box 211, 1000 AE Amsterdam**
**The Netherlands**

ELSEVIER Inc.
525 B Street, Suite 1900
San Diego, CA 92101-4495
USA

ELSEVIER Ltd
The Boulevard, Langford Lane
Kidlington, Oxford OX5 1GB
UK

ELSEVIER Ltd
84 Theobalds Road
London WC1X 8RR
UK



First edition 2005

Library of Congress Cataloging in Publication Data
A catalog record is available from the Library of Congress.

British Library Cataloguing in Publication Data
A catalogue record is available from the British Library.

ISBN: 0-444-51493-7
ISSN (Series): 1574-6526

♾ The paper used in this publication meets the requirements of ANSI/NISO Z39.48-1992 (Permanence of Paper).
Printed in The Netherlands.

# Contents

# Preface

This collection represents the primary reference work for researchers and students working in the area of Temporal Reasoning in Artificial Intelligence. As can be seen from the content, temporal reasoning has a vital role to play in many areas of Artificial Intelligence. Yet, until now, there has been no single volume collecting together the breadth of work in this area. This collection brings together the leading researchers in a range of relevant areas and provides an coherent description of the variety of activity concerning temporal reasoning within the field of Artificial Intelligence.

To give readers an indication of what is to come, we provide an initial, simple example. By examining what options are available for modelling time in such an example, we can get a picture of the variety of topics related to temporal reasoning in Artificial Intelligence. Since many of these topics are covered within chapters in this Handbook, this also serves to give an introduction to the subsequent chapters.

Consider the labelled graph represented in Figure 1:

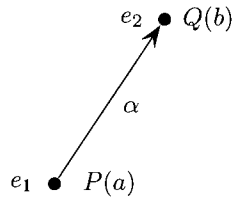



Figure 1: Simple graph structure.

This is a simple directed graph with nodes $e_1$ and $e_2$. The edge is labelled by $\alpha$ and $P(a)$ and $Q(b)$ represent some contents associated with the nodes. We can think of $P$ and $Q$ as predicates and $a$ and $b$ as individual elements. This can represent many things. The two nodes might represent physical positions, with the '$\alpha$' representing movement. Alternatively, $e_1$ and $e_2$ might represent alternate views of a systems, or mental states of an agent, or relationships. Thus this simple graph might characterise a wide range of situations. In general such a situation is a small part of a bigger structure described by a bigger graph. However, we have simply identified some typical components.

Now add a temporal dimension to this, i.e., assume our graph varies over time. One can think of the graph as, for example, representing web pages, agent actions, or database updates. Thus, the notion of change over time is natural. The arrow simply represents an accessibility relation with a parameter $\alpha$ and with $P(a)$ and $Q(b)$ relating to node contents. As time proceeds, the contents may change, the accessibility relation may change; in fact, everything may change.

Now, if we are to model the dynamic evolution of our graph structure, then there are a number of questions that must be answered. Answers to these will define our formal model and, as we will see below, the possible options available relate closely to the chapters within this collection.

**Question 1:** *What properties of time do we need for our application?*
Formally, we might use $(T, <, 0)$ to represent our flow of time, where $T$ represents the temporal structure, $0 \in T$ represents 'now' and '$<$' characterises the earlier–later relation within $T$. The choice of $T$ thus significantly affects the formalisation and properties of the theory, and some options are as follows.

*Option 1.1* Take the flow of time as the Natural Numbers.

*Option 1.2* Take the flow of time as points and intervals and various relations between them, including varying additional granularity, and the question of whether time is discrete or continuous continues.

*Option 1.3* Any other kind of reasonable structure.

**Question 2:** *How do we connect the model of time with the changes in our graph structure?*

*Option 2.1* For each time $t$ (point, interval, etc.) let $G_t$ be a graph for that time.

*Option 2.2* Make each component in a general graph $G$ time dependent. So, in Figure 2, we parameterise the elements in Figure 1 with a temporal component:

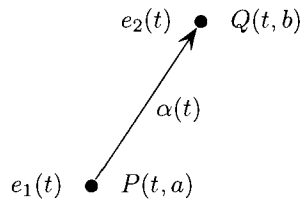



Figure 2: Simple graph with a temporal parameter.

*Option 2.3* We ask how is time created? Is it given to us as in Question 1 or is it generated by our actions on the graph? Is the future open or do we create it by executing actions? If we opt for the action view we need a language of actions on the graphs so that we can generate the temporal movements. This approach will immediately connect time with the graph.

**Question 3:** *How do we talk about the temporal graph?*

Typically, we need a mixed language to talk about the change. There are, again, several options:

*Option 3.1* Use Option *2.1* and devise a special language to talk about time and special connectives to connect the graph with time. In technical terms we are forming the product $T \times G$.

*Option 3.2* Use Option *2.2* and use classical logic or logic programming (or any other language tailored for our application) to represent change directly.

*Option 3.3* Use Option *2.3* and use a state langauge to talk about the graph, an action language to generate time and a metalevel connecting language. Of course, all this can sometimes be embedded in the same language (e.g. Prolog).

**Question 4:** *Can we identify sublanguages that can do all relevant tasks, while ensuring both naturalness of expression and computational tractability?*

We can again consider a number of possibilities.

*Option 4.1* Computational fragments of classical logic: logic programming, description logics, etc.

*Option 4.2* Modal/dynamic logics.

*Option 4.3* Temporal logics.

**Question 5:** *What are the various problems arising from the choice of representation?*

*Option 5.1* Theorem proving requirements.

*Option 5.2* Nonmonotonic problem of representation (persistence, frame problem, etc).

*Option 5.3* Updates, deletions, change, etc.

*Option 5.4* Planning problems.

**Question 6:** *What are the synchronisation aspects?*
Consider Figure 1 again. We may wish to traverse the graph from $e_1$ to $e_2$. If the graph changes during traversal, we may need to synchronise in order to ensure the graph we have is up-to-date. And there are yet more questions concerning this.

- How long does it take to move from $e_1$ to $e_2$?
- How long does it take to execute an action?
- How long does it take to update the graph?

Note that synchronisation questions are only now being studied logically!

**Question 7:** *What metalevel questions are relevant?*
Suppose we wish to describe how the graph changes. We may receive instructions describing it, or its updates, or its states, in some specific language. How do we implement that? How do we synchronise the time involved in the input and the time in the graph?

**Question 8:** *Are there additional features?*
These can be imposed on top of the previous questions.

*Option 8.1* Probabilistic features. To deal with change computationally we need to have some knowledge of how things change. We can either supply algorithms or supply probabilites controlling the change.

*Option 8.2* Fuzzy features. We can make everything fuzzy.

*Option 8.3* Partiality: lack of complete infomation; partial models, mechanisms to overcome our lack of information, etc.

*Option 8.4* Inconsistency and its problems.

**Question 9:** *Can we identify special features relevant to major application areas?*

*Option 9.1* Databases.

*Option 9.2* Law and legal domains.

*Option 9.3* Medicine.

*Option 9.4* Dynamics, space and time.

*Option 9.6* Natural language analysis.

*Option 9.7* Agent-based systems.

Thus, we can see that, even for such a simple initial scenario, there is a very wide range of questions and applications. Looking at the above classifications and at the detailed table of contents, it is not difficult to see the role each chapter plays in this Handbook. Note that each chapter has to address several of these options together in an integrated way with emphasis on the main subject matter of the chapter.

We hope that you, the reader, will find this Handbook both interesting and useful, as the chapters have been contributed by many of the world's leading researchers in temporal reasoning. We would like to thank all these authors for their patience, and for their excellent expositions. We also thank them, together with a number of other experts, for helping us review versions of chapters in this Handbook. Finally, we would like to thank those at Elsevier Publishers who have worked so hard to ensure that this Handbook came into existence.

**Michael Fisher, Dov Gabbay** and **Lluis Vila** [Liverpool, London and Barcelona 2004]

# Contributors

**Alessandro Artale** email: artale@inf.unibz.it

Faculty of Computer Science, Free University of Bozen-Bolzano I-39100 Bozen-Bolzano BZ, Italy

**Chitta Baral** email: chitta@asu.edu

Department of Computer Science and Engineering, Arizona State University, Tempe, Arizona 85287, USA

– The authors would like to acknowledge the help of Marc Denecker in explaining abductive logic programming system.

**Howard Barringer** email: howard@cs.man.ac.uk

Department of Computer Science, University of Manchester, Manchester M13, UK

**Jan Chomicki** email: chomicki@cse.buffalo.edu

Department of Computer Science and Engineering, University at Buffalo, New York 14260-2000, USA

**Dan Clancy** email: clancy@ptolemy.arc.nasa.gov

NASA Ames Research Center, California 94035, USA

**Marc Denecker** email: marcd@cs.kuleuven.ac.be

Department of Computer Science, Katholieke Universiteit Leuven, B-3001 Heverlee, Belgium

**Clare Dixon** email: C.Dixon@csc.liv.ac.uk

Department of Computer Science, University of Liverpool, Liverpool L69, UK

**Thomas Drakengren** email: thomas.drakengren@cainetech.com

Caine Technologies, c/o Drakengren, Varpmossevagen 55A, SE-436 39, Askim, Sweden

– The authors wish to thank the anonymous reviewer and Charlotte Drakengren for their useful comments.

**Jérôme Euzenat** email: Jerome.Euzenat@inrialpes.fr

INRIA Rhône-Alpes, Montbonnot Saint Martin, 38334 Saint-Ismier, France

**Michael Fisher** email: M.Fisher@csc.liv.ac.uk

Department of Computer Science, University of Liverpool, Liverpool L69, UK

**Maria Fox** email: Maria.Fox@cis.strath.ac.uk

Department of Computer and Information Sciences, University of Strathclyde, Glasgow G1 1XQ, UK

**Enrico Franconi** email: franconi@inf.unibz.it

Faculty of Computer Science, Free University of Bozen-Bolzano I-39100 Bozen-Bolzano BZ, Italy

**Dov Gabbay** email: dg@dcs.kcl.ac.uk

Department of Computer Science, King's College, London WC2R, UK

**Antony Galton** email: A.P.Galton@exeter.ac.uk

School of Engineering, Computer Science and Mathematics, University of Exeter, Exeter EX4, UK

**Michael Gelfond** email: mgelfond@cs.ttu.edu

Department of Computer Science, Texas Tech University, Lubbock, Texas 79409, USA

**Alfonso Gerevini** email: gerevini@ing.unibs.it

Dipartimento di Elettronica per l'Automazione, Università di Brescia, 25123 Brescia, Italy

– Figure 8.15 was kindly provided by Bernhard Nebel. The description of the time-graph approach is based on material contained in some papers the author wrote with Len Schubert. The author would like to thank the anonymous reviewer and Alessandro Saetti for useful comments on a preliminary version on this chapter.

**Steve Hanks** email: hanks@u.washington.edu

Department of Computing and Software Systems, University of Washington Tacoma, Washington, USA

– This work was supported, in part, by ARPA / Rome Labs Grant F30602–95–1–0024 and in part by NSF grant IRI–9523649.

**Peter Jonsson** email: petej@ida.liu.se

Department of Computer and Information Science, Linköping University, SE-581 83 Linköping, Sweden

**Elpida Keravnou** email: elpida@ucy.ac.cy

Department of Computer Science, University of Cyprus, CY-1678 Nicosia, Cyprus

**Manolis Koubarakis** email: manolis@intelligence.tuc.gr

Department of Electronic and Computer Engineering, Technical University of Crete, Chania, Crete, Greece

– This work was partially supported by the CHOROCHRONOS project funded by the EU's 4th Framework Programme (1996-2000) and by a grant from the Greek General Secretariat for Research and Technology (1998-1999). The author would also like to thank Spiros Skiadopoulos, Peter Jeavons and David Cohen for their collaboration on various topics related to this work.

**Benjamin Kuipers** email: kuipers@cs.utexas.edu

Computer Science Department, University of Texas at Austin, Texas, USA

– This work took place in the Intelligent Robotics Lab at the Artificial Intelligence Laboratory, The University of Texas at Austin. Research of the Intelligent Robotics lab is supported in part by NSF grants IRI-9504138 and CDA 9617327, and by funding from Tivoli Corporation.

**Derek Long** email: Derek.Long@cis.strath.ac.uk

Department of Computer and Information Sciences, University of Strathclyde, Glasgow G1, UK

**David Madigan** email: madigan@stat.rutgers.edu

Department of Statistics, Rutgers University, New Jersey, USA

– This work was supported, in part, by NSF grant DMS–97–04–573.

**Alice ter Meulen** email: atm@let.rug.nl

Center for Language and Cognition, University of Groningen, The Netherlands

**Angelo Montanari** email: montana@dimi.uniud.it

Dipartimento di Matematica e Informatica, Università di Udine, Udine, Italy

**Han Reichgelt** email: han@georgiasouthern.edu

Department of Information Technology, Georgia Southern University, Statesboro, Georgia, USA

**Mark Reynolds** email: mark@csse.uwa.edu.au

School of Computer Science and Software Engineering, The University of Western Australia, Australia

**Yuval Shahar** email: yshahar@bgumail.bgu.ac.il

Department of Information Systems Engineering, Ben-Gurion University of the Negev, Israel

**David Toman** email: david@uwaterloo.ca

School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada.

– This chapter is an extended and updated version of [Chomicki and Toman, 1998]. The authors gratefully acknowledge the United States National Science Foundation (grants IIS-9110581 and IIS-9632870) and the Natural Sciences and Engineering Research Council of Canada for their support of this research.

**Kristof Van Belleghem** email: kristof@cs.kuleuven.ac.be

Department of Computer Science, Katholieke Universiteit Leuven, B-3001 Heverlee, Belgium.
(Currently at: PharmaDM, Kapeldreef 60, B-3001 Leuven, Belgium)

**Lluis Vila** email: vila@lsi.upc.es

Department of Software, Technical University of Catalonia, Barcelona, Spain

– The author's contributions to this collection were supported by the Spanish CICYT under grant TIC2002-04470-C03-01 and under "Web-I(2)" grant TIC2003-08763-C02-01.

**Michael Wooldridge** email: M.Wooldridge@csc.liv.ac.uk

Department of Computer Science, University of Liverpool, Liverpool L69, UK

**Hajime Yoshino**

Meiji Gakuin University, Tokyo, Japan

# Foundations

This Page Intentionally Left Blank

Handbook of Temporal Reasoning in Artificial Intelligence
Edited by M. Fisher, D. Gabbay and L. Vila


# Chapter 1

# Formal Theories of Time and Temporal Incidence

## Lluis Vila

The design of intelligent agents acting in a changing environment must be based on some form of temporal reasoning system. Such a system should, in turn, be founded on a formal theory of time. Theories of time are based on some primitive time units (instants, intervals, etc. ) and determine both the expressiveness of the language and the completeness of the reasoning system.

The time theory of a temporal reasoning system is closely connected with the so-called theory of *temporal incidence*, meaning the set of domain-independent properties for the truth-value of temporal propositions throughout time. Classically, for a given domain, we distinguish between two classes of temporal propositions: changing domain properties (or *fluents*) and events whose occurrence may cause change on fluents.

Formal theories of time and temporal incidence involve some controversial issues such as (i) the expression of *instantaneous events* and *fluents* that hold instantaneously, (ii) the *dividing instant problem* and (iii) the formalization of the properties for *non-instantaneous holding of fluents*.

This chapter surveys the most relevant theories of time proposed in Artificial Intelligence according to various representational issues including the ones above. Also, the chapter presents a brief overview of *temporal incidence theories* and proposes a theory of temporal incidence defined upon a theory of instants and periods whose key insight is the distinction between continuous and discrete fluents.

## 1.1 Introduction

An intelligent agent interacting in a changing environment must be able to reason about these changes as well as the events and actions causing them, the effects it may have in the rest of the environment and the time when all these things happen or cease happening. Therefore, the design of intelligent agents acting in a changing environment must be based, among other components, on some form of temporal reasoning system. If we want this system to be well-founded and its properties formally studied it must be based upon a formal theory of time. Time theories are based in some time primitive unit (instants, intervals, etc. ) and determine both the expressiveness of the language as well as the completeness of the reasoning system.

As a matter of fact, time has been recognized as a fundamental notion in reasoning about changing domains and many frameworks for reasoning about change and action are built upon a temporal representation [McDermott, 1982; Allen, 1984; Kowalski and Sergot, 1986; Dean and McDermott, 1987; Williams, 1986; Shoham, 1987; Kuipers, 1988; Forbus, 1989; Galton, 1990; Schwalb *et al.*, 1994; Pinto, 1994; Miller and Shanahan, 1994; Koubarakis, 1994a; Iwasaki *et al.*, 1995; Fusaoka, 1996; Bacchus and Kabanza, 1996; Vila and Reichgelt, 1996]. In these frameworks, the domain at hand is formalized by expressing how propositions are true or false throughout time. Commonly there is a distinction between propositions describing the state of the world (*fluents*) and those representing occurrences that happen in the world and may change its state (*events*). Examples of fluents are "the light is on", "the ball is moving at speed $v$", and "the battery charge is increasing", and examples of events are "turn the light off", "kick the ball" and "the controller sent a signal to the relay".

A temporal representation has two basic components: (i) a theory of time, and (ii) a theory of temporal incidence. The *theory of time* defines the structure of the primitive time units (e.g. *transitivity* of the ordering relation over instants). The *theory of temporal incidence* defines the domain-independent properties for the truth-value of fluents and events throughout time (e.g. if a fluent is true during a period it must be true during the instants within that period).

Time and temporal incidence are issues that traditionally attracted the interest of areas such as philosophy [Whitehead, 1919; Russell, 1956; Hamblin, 1972; Kamp, 1979; Newton-Smith, 1980], *physics* and *linguistics* [Kenny, 1963; Vendler, 1967; Davidson, 1967; Jackendoff, 1976; Mourelatos, 1978; Dowty, 1979; Allen, 1984; Bach, 1986]. But, why are time and temporal incidence theories important for *automated temporal reasoning* ? They have impact on major properties of the system for answering queries with some temporal component such as "Was the light open when the controller sent the signal to the relay ? "At what times has been the light on and the door open simultaneously ?". Consider, for instance, *Constraint Logic Programming* [Jaffar and Maher, 1994] with temporal constraints [Hrycej, 1993; Brzoska, 1993; Früehwirth, 1996; Schwalb *et al.*, 1996]: the theory of time characterizes the *constraint domain* which determines the properties for the constraint solving procedure; the theory of temporal incidence has impact on the completeness of the overall proof procedure.

From an efficiency point of view, these theories enable inferring implicit, redundant or inconsistent temporal information that can be used to expedite the query answering search. For example, consider two tasks competing on a single resource. The theory of temporal incidence allows us to infer that the time periods during which these tasks utilize the common resource do not overlap. In turn it enables some temporal constraint propagation.

A lot of research in *Artificial Intelligence* has focussed on formalizing time and temporal incidence [van Benthem, 1983; Allen and Hayes, 1985; Ladkin, 1987; Shoham, 1987; Tsang, 1987a; Allen and Hayes, 1989; Galton, 1990; Lin, 1991; Vila, 1994]; however it turns out not to be a simple task. First because a theory of time must naturally reflect commonsense intuitions about time, and sometime these intuitions have to do with instantaneous phenomena while others more naturally concern durative phenomena. Second because they must be adequate to describe events that happen and change the values of fluents without contradicting those intuitions. For instance, determining the value of a fluent at the time it changes has been a controversial issue (the so-called *dividing instant problem*). Moreover, we may face different types of change: discrete, continuous, etc. Real world domains usually involve parameters whose change is modelled as continuous and others whose change

is viewed as discrete. A system with both types of change is called a *hybrid system*, and its model a *hybrid model* [Iwasaki *et al.*, 1995]. For example, consider an electro-mechanical battery charger: re-charging a battery can be viewed as a continuous change whereas closing a relay would be better regarded as discrete*. When both types of change happen concurrently, describing what is true and false becomes more difficult.

This chapter presents these challenges more precisely, presents the most relevant theories of time proposed in Artificial Intelligence and discusses the successes and failures with respect to these representational issues. Also, the chapter presents a brief overview of temporal incidence theories and proposes a theory of *temporal incidence* defined upon a theory of instants and periods whose key insight is the distinction between continuous and discrete fluents.

## 1.2 Requirements and Problems

In this section we identify some problematic issues that must be addressed when formalizing time and temporal incidence:

**Instantaneous Events** A dynamic system often involves events that are naturally modelled as instantaneous. Some prototypical examples are "turn off the light", "shoot the gun", "start moving", "sign a contract". Representing such events can cause some problems, especially when *sequences* of them occur in presence of continuous change (Section 1.11 discusses this in detail).

**Fluents that Hold at an Instant** We often talk about the value of fluents at certain instants (e.g. "the temperature of patient X at 9:00 was high", "Was the light red when the car hit John?"). Also, modelling *continuous change* requires having fluents that may hold at isolated instants. A simple, representative example is the parameter *speed* ($v$) of a ball tossed upwards in what we call the *Tossed Ball Scenario* (TBS) (see Figure 1.1). The ball moves up during

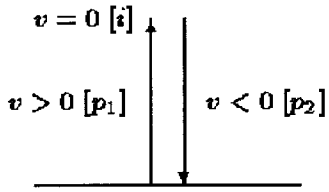



Figure 1.1: The Tossed Ball Scenario (TBS).

$p_1$ and down during $p_2$. By continuity of $v$, there must be a time piece "in between" $p_1$ and $p_2$ where the $v = 0$. Since the ball cannot be stopped in the air for a while, such a time piece can only be durationless. However, being able to talk about the truth value of fluents at durationless times may lead to the problem described next.

*Hybrid models are important because many daily used electro-mechanical devices are suitably modelled as such.

**The Dividing Instant Problem (DIP)** Let us assume that time is composed of both instants and periods, and we need to determine the truth-value of a fluent $f$ (e.g. "the light is on") at an instant $i$, given that $f$ is true on a period $p_1$ ending at $i$ and is false at a period $p_2$ beginning at $i$ (see Figure 1.2) [Hamblin, 1972; van Benthem, 1983; Allen, 1984; Galton, 1990]. The problem is a matter of logical consistency: if periods are closed then $f$

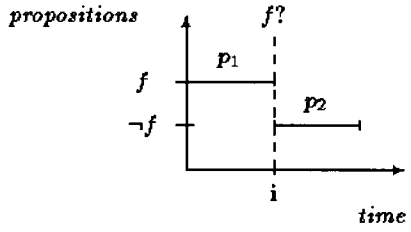



Figure 1.2: The Dividing Instant Problem.

and $\neg f$ are both true at $i$ which is inconsistent; if they are open we might have a "truth gap" at $i$; finally, the choices open/closed and closed/open are judged to be artificial [Allen, 1984; Galton, 1990].

**Non-Instantaneous Holding of Fluents** Formalizing the properties of temporal incidence for non-instantaneous fluents can be problematic. There are two major classes of them. The first class are properties about the holding of a fluent at related times. Instances of this class are:

- *Homogeneity*: If a fluent is true on a piece of time it must hold on any sub-time [Allen, 1984; Galton, 1990].

- *Concatenability**: If a fluent is true on two consecutive pieces of time it must be true on the piece of time obtained by concatenating them. Notice that there may be different views for the meaning of *consecutive*.

The second class are properties relating the holding of contradictory fluents at related times. It includes:

- *Non-holding*: If a fluent is not true on a piece of time, there must be a sub-time where its negation holds [Allen, 1984; Galton, 1990].

- *Disjointness*: Any two periods such that a fluent is true on one and its negation is true on the other, must be disjoint. There may be different views for the meaning of *disjoint* too.

*As opposed to homogeneity, very little attention has been paid to concatenability. Nevertheless, it is a semantical issue that also may have some computational benefits since it allows for a compact representation of a fluent holding on numerous, overlapping periods.

**Non-Atomic Fluents** The holding of negation, conjunction and disjunction of atomic fluents can be a non-trivial issue [Shoham, 1987; Galton, 1990].

After presenting our theories of time and temporal incidence we shall revisit all these issues. Next, we'll discuss theories of time. Starting from different intuitions, several theories of time have been proposed*. They can be classified into three classes according to whether the primitive time units are instants, periods or events†.

## 1.3 Instant-based Theories

Instants are defined as a durationless pieces of time. An alternative, more precise definition identifies instants to pieces of time whose begin and end are not distinct. In *physics* it has been standard practice to model time as an unbounded continuum of instants [Newton, 1936] structured as the real numbers set. Instant-based theories have been used in several AI systems [McCarthy and Hayes, 1969; Bruce, 1972; Kahn and Gorry, 1977; McDermott, 1982; Shoham, 1987].

An instant-based theory is defined on a structure $\langle \mathcal{I}, \prec \rangle$ with a number of properties:

- *Ordering.* The *minimum* properties for instants ordering are those of a *partially ordered set* (POSET):

$$\begin{array}{ll} \textbf{IRREF} & \neg(i \prec i) \\ \textbf{ASYM} & i \prec i' \Rightarrow \neg(i' \prec i) \\ \textbf{TRANS} & i \prec i' \wedge i' \prec i'' \Rightarrow i \prec i'' \end{array}$$

  Additionally, one may want to impose some *"linearity"* of time. If it is imposed only towards the past

$$\textbf{Left-LIN} \quad i' \prec i \wedge i'' \prec i \Rightarrow i' \prec i'' \vee i' = i'' \vee i'' \prec i'$$

  we obtain a branching time structure towards the future [McDermott, 1982] that might be appropriate to model various possible futures. Otherwise, we can impose linearity on both directions

$$\textbf{LIN} \quad i \prec i' \vee i = i' \vee i' \prec i$$

  forcing instants to be arranged in a single line.

- *Boundedness.* The instants ordered structure may have beginning and end or, conversely, have no end towards past and future. The later is caputured by the following axiom:

$$\textbf{SUCC} \quad \forall i \exists i' \, (i' \prec i') \quad \forall i \exists i' \, (i \prec i)$$

  The idea of unbounded time corresponds to a more general view whereas bounded time can be more appropriate in some particular contexts.

- *Dense/Discrete.* Denseness forces to have an instant between any two instants

$$\textbf{DENS} \quad \forall i, i' \, (i \prec i' \Rightarrow \exists i'' \, (i \prec i'' \prec i'))$$

*Some intuitions as well as pointers to some relevant works are from [Lin, 1991].

†Note that the word event is overloaded: in the context of time theories, *event* denotes any temporal proposition, thus it includes both events and fluents as defined in the introduction section.

This property may be important to model continuous change. A consequence is that any stretch of time can be decomposed into sub-times which can have interest in planning where tasks are usually decomposed into subtasks. Another consequence is that one cannot refer to the previous and next instants. Discreteness is enforced by the following axiom:

$$\textbf{DISC} \quad \begin{array}{l} \forall i, i' \; (i < i' \Rightarrow \exists i'' \; (i \prec i'' \wedge \neg \exists i''' \; i \prec i''' \prec i'')) \\ \forall i, i' \; (i < i' \Rightarrow \exists i'' \; (i'' \prec i' \wedge \neg \exists i''' \; i'' \prec i''' \prec i')) \end{array}$$

As it happens, each finite strict partial order is also discrete.

The above principles are sufficient to achieve a certain level of completeness. Two theories are known to be *syntactically complete* [van Benthem, 1983]: the *unbounded dense linear* theory axiomatized by

IRREF, TRANS, LIN, SUCC, DENS

and the *unbounded discrete linear* theory axiomatized by

IRREF, TRANS, LIN, SUCC, DENS

The alternatives to instant-based theories are theories built upon primitive units more related to our experience than instants. One alternative are period-based theories since "periods are usually associated with events that take time". A further step in that direction is directly developing theories based on events.

## 1.4 Period-based Theories

Period-based theories interested researchers in philosophy, linguistics and AI [Walker, 1948; Hamblin, 1972; Newton-Smith, 1980; Dowty, 1979; Allen, 1984; Allen and Hayes, 1989]. For instance, Allen proposes a theory exclusively based on periods and the 13 relations between pairs of them shown in Figure 1.3. This theory has been analysed and reformulated

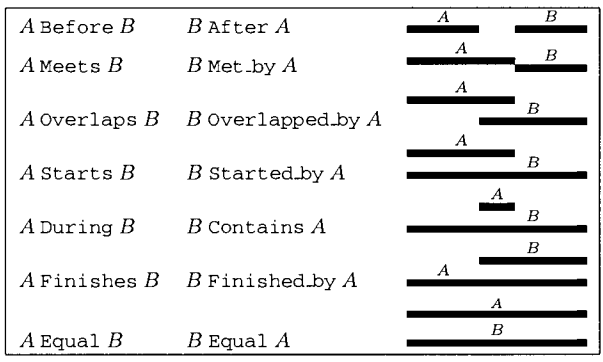



Figure 1.3: The 13 relations between temporal intervals.

in terms of the sole relation `Meets` by Allen & Hayes and Ladkin.

### 1.4.1 Allen's Period Theory

Allen [Allen, 1983] takes an initial structure $\langle \mathcal{P}, \mathbf{AR} \rangle$ where $\mathbf{AR}$ denotes the set of the 13 primitive period relations that correspond to every possible simple qualitative relationship that may exist between two intervals (see Figure 1.3). The behaviour or $\mathbf{AR}$ is informally specified by the following axiom schemas [Allen, 1984]:

1. Given any period, there exists another period related to it by each relationship in $\mathbf{AR}$.
2. The relationships in $\mathbf{AR}$ are mutually exclusive.
3. The relationships have a transitive behaviour, e.g. if $p_1$ `Before` $p_2$ and $p_2$ `Meets` $p_3$ then $p_1$ `Before` $p_3$.

We shall henceforth refer to this theory as $\mathcal{A}$. We propose the following formalization:

$\mathbf{A_1}$ $\forall P \in \mathcal{P}, R \in \mathbf{AR}\ \exists P'(R(P, P'))$
$\mathbf{A_2}$ $\forall P, P' \in \mathcal{P}, R \in \mathbf{AR}\ \forall R' \in \mathbf{AR} - R\ (R(P, P') \Rightarrow \neg R'(P, P'))$
$\mathbf{A_3}$ Allen's transitive table [Allen, 1983]

### 1.4.2 Allen & Hayes's Period Theory

Allen's theory is re-defined in terms of the `Meets` relation in [Allen and Hayes, 1985]* ($\|$ denotes `Meets` and $\oplus$ the or-exclusive logical connective):

$\mathbf{AH_1}$ $\forall p, q, r, s\ (p\|q \wedge p\|s \wedge r\|q \Rightarrow r\|s)$
$\mathbf{AH_2}$ $\forall p, q, r, s\ (p\|q \wedge r\|s \Rightarrow \exists t\ (p\|t\|s) \oplus p\|s \oplus \exists t\ (r\|t\|q))$
$\mathbf{AH_3}$ $\forall p\ \exists q, r\ q\|p\|r$
$\mathbf{AH_4}$ $\forall p, q, r, s\ (p\|q\|s \wedge p\|r\|s \Rightarrow q = r)$
$\mathbf{AH_5}$ $\forall p, q\ (p\|q \Rightarrow \exists r, s, t\ (r\|p\|q\|s \wedge r\|t\|s))$

We call it $\mathcal{AH}$. It has been also by Ladkin [Ladkin, 1987] who (i) claims that axiom $\mathbf{AH_5}$ is redundant (it is, in fact, not true [Galton, 1996a]), (ii) relates $\mathcal{AH}$ to other period-based theories of time, (iii) completely characterizes its models, and (iv) proposes a completion to obtain an axiomatization of the theory of rational intervals which is proved to be *countably categorical* [van Benthem, 1983] and, therefore, complete. The completion is obtained by adding the following *denseness* axiom $\mathbf{N_1}$:

$$\mathbf{N_1} \quad \forall p, q, r, s\ \exists x, y\ \text{Pointless}(p, q, x, y) \wedge \text{Pointless}(x, y, r, s)$$

where pointless is defined as follows:

**Definition 1.4.1.** **("pointless")** *Given the intervals $p, q, r, s$, Pointless$(p, q, r, s)$ iff*

$$\exists u, v, w\ [p, q] \sim_{\mathcal{I}} [u, v] \wedge [r, s] \sim_{\mathcal{I}} [v, w]$$

and $\sim_{\mathcal{I}}$ denotes the period relation of "having the same meeting point".

*Notice that this re-definition is not equivalent to Allen's initial axiomatics. Both are analyzed below in deeper detail.

### 1.4.3 Relation between $\mathcal{A}$ and $\mathcal{AH}$

Surprisingly, $\mathcal{A}$ and $\mathcal{AH}$ are not that similar. On the one hand $\mathcal{A}$ accepts models which do not fit in $\mathcal{AH}$.

**Lemma 1.4.1.** $\mathcal{A} \not\subset \mathcal{AH}$

*Proof.* We may find several counter-examples of models of $\mathcal{A}$ which are not models of $\mathcal{AH}$.

- Counter-example 1 (axiom $\mathbf{AH_4}$): Let us take $M$ as the set of non-empty open intervals on $Q$ plus a second "copy" of an arbitrary interval, let us take for instance (1,2)', which happens to be different from its "original" (1,2) though it keeps every relation that (1,2) has with any other interval. Thus, $M$ is a model of $\mathcal{A}$ but it is not a model of $\mathcal{AH}$ since it does not satisfy $\mathbf{AH_4}$.
- Counter-example 2 (axiom $\mathbf{AH_2}$): Let us take $M$ as two copies of the set of non-empty open intervals on $Q$, being the intervals in one copy not related by any relationship the intervals of the other copy. $M$ is a model of $\mathcal{A}$ but linear ordering is not satisfied since any interval in one copy is not ordered with respect to any interval of the other one.

□

On the other hand, $\mathcal{A}$ is stronger than $\mathcal{AH}$ since it imposes *denseness* Axiom $\mathbf{A_1}$ in $\mathcal{A}$ guarantees decomposability which in period-based theories corresponds to denseness. Contrarily, $\mathcal{AH}$ has been designed to be weak enough to embrace both discrete and dense models.

**Lemma 1.4.2.** $\mathcal{AH} \not\subset \mathcal{A}$

*Proof.* As a counter-example take any discrete model –for instance the intervals formed over the set of integer numbers. It is a model of $\mathcal{AH}$ [Allen and Hayes, 1985] but is ruled out by $\mathbf{A_1}$. For instance, there is no period which `Overlaps` the period [1,2]. □

### 1.4.4 Revised $\mathcal{A}$

A more accurate look at $\mathcal{A}$ reveals that nothing accounts for the intuition that "periods are all contained in a single time dimension"*. Our refinement of $\mathcal{A}$ is based on adding an axiom schema with such a role:

**Definition 1.4.2.** ($\mathcal{A}'$) *$\mathcal{A}'$'axiomatization is $\mathcal{A}$'s plus the following additional axiom schema:*

$$\mathbf{A'_1} \quad \forall P, P' \, \exists R \in \mathbf{AR} \; R(P, P')$$

This refinement produces a rather remarkable change in the accepted models. We demonstrate it by analyzing how this theory relates to Allen & Hayes's theory. We use the definitions of period relations in terms of the single relation `Meets` given in [Allen and Hayes, 1985].

**Lemma 1.4.3.** $\mathcal{A}' \subseteq \mathcal{AH}$

*This is an idea which Allen seems to be in sympathy with since he explicitly refuses alternative structures like McDermott's *branching time* construction.

*Proof.*

**AH$_1$**. Suppose $\neg r \| s$.

Then, by **A'$_1$**, there must be a period relation $R$ such that $R \neq \|$ and $rRs$. It is easy to check that whatever relation we take, it stands in contradiction with the period transitivity table and/or with axiom $A_2$. For example let us assume that $(r$ `Before` $s)$, which is equivalent to $(s$ `After` $r)$. Using the period constraints in the premise and our assumption, we have $(p$ `Meets` $s$ `After` $r$ `Meets` $q)$. By successively applying Allen's transition table we get

$$(p \texttt{ After Met\_by Overlaps Overlapped\_by During Contains Start Started\_by Finishes Finished\_by Equal } q)$$

which is inconsistent with $p \| q$ in the premise.

**AH$_2$**. From the hypothesis (a) $p \| q \wedge r \| s$ we want to show that one of the following alternatives exclusively hold: (b) $p \| s$, (c) $p$ `Before` $s$, (d) $r$ `Before` $q$ –according to the definition of `Before`. By $A'_1$ we know that one of the 13 period relations must hold between $p$ and $s$. Since (b) and (c) are mutually exclusive (by $A_2$) we only need to prove the following three statements:

1. (a) $\wedge \neg(p \| s) \wedge \neg(p$ `Before` $s) \Rightarrow r$ `Before` $q$
2. (a) $\wedge\, p \| s \Rightarrow \neg(r$ `Before` $q)$
3. (a) $\wedge\, p$ `Before` $s \Rightarrow \neg(r$ `Before` $q)$

**1.** Let us assume $\neg(p \| s)$, $\neg(p$ `Before` $s)$ and $\neg(r$ `Before` $q)$. By **A'$_1$** and **A$_2$** one of the other 12 period relations holds. If $r \| q$ then from $r \| q \wedge r \| s$ it follows –by applying the transitivity table– that $q(= \vee$ `Starts` $\vee$ `Started_by`$)s)$, which combined with $p \| q$ by transitivity and conjunction gives $p \| s$ which stands in contradiction with our starting assumption. For any other case it holds that $\exists t'\ (p \| t' \| s)$ (this can be checked by revising the definition of the remaining 12 relations in terms of $\|$) that is equivalent to $p$ `Before` $s$ which is contradictory with the hypothesis $\neg(p$ `Before` $s)$. **2** and **3** are shown by assuming $r$ `Before` $q$ and standing in contradiction by consecutively applying the transitivity table a number of times.

**AH$_3$**. is trivially derived from **A$_1$**.

**AH$_4$**. (By **A$_1$**) $q$ and $r$ must be related by some period relation, namely $R$. Suppose that $R$ is not $=$. Then using the definitions of each period relation one stands in contradiction with one from $\{p \| q, q \| s, p \| r, r \| s\}$.

□

**Lemma 1.4.4.** $\mathcal{AH} \vdash$ **A'$_1$, A$_2$, A$_3$**

*Proof.* **A'$_1$** and **A$_2$**. For any two periods $I, J$ we have (by **AH$_3$**) that $\exists a, b, a', b'\ a \| I \| b$ and $a' \| J \| b'$. By **AH$_2$** we have

1. $\exists t_1\ a \| t_1 \| J \oplus a \| J \oplus \exists t'_1\ a' \| t'_1 \| I$
2. $\exists t_2\ a \| t_2 \| b' \oplus a \| b' \oplus \exists t'_1\ J \| t'_2 \| I$

3. $\exists t_3\ I\|t_3\|J \oplus I\|J \oplus \exists t'_3\ a'\|t'_3\|b$
4. $\exists t_4\ I\|t_4\|b' \oplus I\|b' \oplus \exists t'_4\ J\|t'_4\|b$

We may combine these different mutually exclusive choices. We obtain $3^4$ *a priori* possible alternatives, but not every combination is feasible. We must use the remaining $\mathcal{AH}$ axioms to discard disallowed possibilities. For instance, let us take the first choice in (1): $\exists t_1\ a\|t_1\|J$. (By $\mathbf{AH_5}$) there exists $t_2$ and $b$ such that $t_2 = t_1 + J$ and $a\|t_2\|b'$ which is the first choice in (2) and, due to the $\oplus$, the sole one allowed. Thus, we have (1.2) $a\|t_1\|J\|b'$. This is compatible with every choice in (3) each of which in turn is compatible with some in (4). Finally we get (1.2) with each of the following relations:

$$
\begin{array}{l}
I\|t_3\|J \\
I\|J \\
a'\|t'_3\|b \wedge I\|t_4\|b' \\
a'\|t'_3\|b \wedge I\|b' \\
a'\|t'_3\|b \wedge J\|t'_4\|b
\end{array}
$$

By exhaustively applying this process we obtain an exclusive disjunctive formula where each disjunctive element exactly matches the definition in terms of the single relation `Meets` of one of the 13 period relations. Thus, we prove the mutual exclusivity of period relations ($\mathbf{A_2}$). Their existence ($\mathbf{A'_1}$) is also guaranteed since every auxiliary period has been introduced either through $\mathbf{AH_3}$ or $\mathbf{AH_2}$ and period addition used at some stages is supported by $\mathbf{AH_5}$.

$\mathbf{A_3}$ is easy though a bit tedious to prove by using the transitivity table [Allen and Hayes, 1985]. □

Regarding $\mathbf{A_1}$, the relationships `Meets` and `Before` (and their inverses) can be easily derived from one and two applications respectively of axiom $\mathbf{AH_3}$ towards the future (towards the past), but this is not the case for the remaining ones. To derive them we would require the *denseness* axiom $\mathbf{N_1}$:

**Lemma 1.4.5.** $\mathcal{AH}, \mathbf{N_1} \vdash \mathbf{A_1}$

*Proof.* By using the denseness axiom one may always find the appropriate endpoints which define the period $P'$ that satisfies $R(P, P')$. □

**Theorem 1.4.1.** $Th(\mathcal{AH}, \mathbf{N_1}) \equiv \mathcal{A}'$

*Proof.* Given lemmas 1.4.3, 1.4.4 and 1.4.5, it suffices to prove that $\mathcal{A}' \vdash \mathbf{N_1}$ which is straight forward by applying $\mathbf{A_1}$ with the relationship `Started_by` on the period bounded by the initial points. □

### 1.4.5 Extending $\mathcal{A}$ with Instants

Allen's theory can be properly extended with instants by implementing the idea of instants as period the meeting points. Now the ontology is made of non-empty sets of instants and periods in a structure such as $\langle \mathcal{I}, \mathcal{P}, \texttt{Limits}, \mathbf{AR} \rangle$. where `Limits` is a instant-period relation. $\mathcal{A}'$ axioms are extended with the following:

$$
\begin{array}{ll}
\mathbf{IM_1} & \forall P, P'\ (P\|P' \Rightarrow \exists i\ (\texttt{Limits}(i, P) \wedge \texttt{Limits}(i, P'))) \\
\mathbf{IM_2} & \forall i \exists P, P'\ (P\|P' \wedge \texttt{Limits}(i, P) \wedge \texttt{Limits}(i, P'))
\end{array}
$$

## 1.5 Events

Event-based theories are motivated by the following intuition:

> "time is no more than the totality of temporal relations between the events and processes which constitute the history of our world. Then defining time is a question about the actual relations between these events and processes."

This approach mostly interested philosophers such as [Russell, 1956; Whitehead, 1919; Kamp, 1979] and a few AI people [Tsang, 1987a].

Time is defined as the structure $\langle E, \prec, O\rangle$ where $E$ is a non-empty set of events, $\prec$ is a precedence relation and $O$ is a overlapping relation. The axioms of the theory, called $\mathcal{E}$, are as follows [Kamp, 1979]:

| | | |
|---|---|---|
| $\mathbf{E_1}$ | $e \prec e' \Rightarrow \neg(e' \prec e)$ | **NO SYM**($\prec$) |
| $\mathbf{E_2}$ | $e \prec e' \wedge e' \prec e'' \Rightarrow e \prec e''$ | **TRANS**($\prec$) |
| $\mathbf{E_3}$ | $eOe' \Rightarrow e'Oe$ | **SYM**($O$) |
| $\mathbf{E_4}$ | $eOe$ | **REFL**($O$) |
| $\mathbf{E_5}$ | $e \prec e' \Rightarrow \neg(eOe')$ | **SEP** |
| $\mathbf{E_6}$ | $e \prec e' \wedge e'Oe'' \wedge e'' \prec e''' \Rightarrow e \prec e'''$ | **TRANS**($\prec, O$) |
| $\mathbf{E_7}$ | $e \prec e' \vee eOe' \vee e' \prec e$ | **LIN** |

$\mathbf{E_1}$ and $\mathbf{E_2}$ state partial order for $\prec$. $O$ is reflexive and symmetric ($\mathbf{E_1}$ and $\mathbf{E_2}$) but not transitive. $\mathbf{E_5}$ to $\mathbf{E_7}$ state relations between $\prec$ and $O$: they are mutually exclusive, exhibit a sort of transitivity and establish a linear ordering over events. The properties of $\mathcal{E}$ have extensively studied in [Kamp, 1979] and later in [Tsang, 1987a; Lin, 1991].

Since period-based theories are based on the intuition that periods are the stretches of time occupied by events, the properties of periods and events are very similar. However, event-based theories are conceptually different in the sense that events are not "pure time entities" but the occurrences themselves. In other words, a clear dissociation between occurrences and their times of occurrence is not established.

### 1.5.1 Relation between $\mathcal{E}$ and $\mathcal{AH}$

Since events happen on periods, period-based theories and event-based are very similar. The main distinction is that two events that happen at the same are not necessarily the same events. If we take Allen's relations as the reference, $e \prec e'$ is equivalent to $e$ `Before` $\vee$ `Meets` $e'$ and $O$ is equivalent to $\neg(e \prec e') \wedge \neg(e' \prec e)$. According to Tsang [Tsang, 1987b], $\mathcal{E}$ can be extended to obtain a theory equivalent to $\mathcal{AH}$ by adding the following axioms (where ! and $\cap$ denote period intersection and union respectively):

| | |
|---|---|
| $\mathbf{E_8}$ | $\exists e'\ e' \prec e \wedge \neg(\exists x\ (e' \prec x \wedge x \prec e)$ |
| $\mathbf{E_9}$ | $e \prec e' \wedge e' \prec e'' \Rightarrow e \prec e''$ |
| $\mathbf{E_{10}}$ | $eOe' \Rightarrow e!e'$ |
| $\mathbf{E_{11}}$ | $\exists e''\ e'' = e \cup e$ |

$\mathbf{E_8}$ and $\mathbf{E_9}$ are needed to guarantee period meeting unboundedness ($\mathbf{AH_e}$ axiom), $\mathbf{E_{10}}$ is needed to derive axiom $\mathbf{AH_1}$, and $\mathbf{E_{11}}$ is needed to derive $\mathbf{AH_2}$ and $\mathbf{AH_5}$. Given the theory $\mathbf{E_1} \div \mathbf{E_{11}}$, called $\mathcal{E}^*$ by Tsang, we have:

**Theorem 1.5.1.** $\mathcal{E} \equiv \mathcal{AH}$

## 1.6 Analysing the Time Theories

Let us now analyze these alternatives from the philosophical, notational, computational technical viewpoints:

**Philosophical** Event-based theories clearly are the most attractive from a philosophical point of view as they are directly based on perceived phenomena. However, as noticed by [Lin, 1991], are instant and period -based theories that have gained wide acceptance in AI:

> "The reason for this seems to be that people are accustomed to think that time is an "independent" entity where events take place."

Period-based theories are appealing since they make this distinction while they preserve the starting intuition that our direct experience is with events that take time. In fact, period-based theories capture the most relevant relations between events.

Several philosophical arguments have been put forward against instants such as "Our direct experience is with phenomena that take time", "It takes to many instants to make up a durable experience ?" [Hamblin, 1972; Kamp, 1979], "the point-based, continuous model ... they start with is too rich" [Allen and Hayes, 1989], "... (instant-based models) permit the description of phenomenally impossible states of affairs" [Hamblin, 1972]. One argument in favor of instants must be mentioned though. As with durative events, we seem to have mental experience with instantaneous phenomena as well. It is reflected by many references in natural language expressions (see the examples given in Section 1.2 such as "the time I started moving" or "the temperature of the patient at 9:00"). The claim is not about the existence of instantaneous phenomena but about the fact that we are accustomed to think about the notion of instantaneous.

**Notational vs. Computational** Two types of expressions must be considered: *temporal assertions* and *temporal relations*. We next discuss both, being the second the one that has some computational consequences.

*Expressing temporal assertions:* If our ontology is provided with instants, expressing instantaneous events and instantaneous holding of fluents is straight forward. This becomes a difficult issue in period-based theories because instants cannot be represented as very short periods (as proposed in [Allen, 1984]) because a short period does not *divide* a period into two meeting periods. For example, if $i$ in Figure 1.2 is modelled as a short period then $p_1$ does not meet $p_2$. The same applies to expressing instantaneous fluents (like in the TBS). The proposal of modelling instants as *indivisible* periods, called *moments* [Allen and Hayes, 1985], does not work either for the same reason. The option of representing instants as zero duration periods is problematic too because Allen's transitive table needs to be transformed into a much weaker table, otherwise the distinction between the period relations that are not `Before`, `Equal` and `After` becomes meaningless [Schwalb, 1996].

A sounder, more sophisticated technique is defining instants as sets of periods. Mathematicians proposed a number of set-theoretic constructions of points from intervals such as (i) an instant is identified with the maximal set of intervals that have a non-empty intersection [Whitehead, 1919] (called *nests*), or (ii) an instant is defined as the equivalence class of pairs of meeting intervals that meet "at the same place"*. Now, the points are (i) whether

* Attributed to Bolzano.

there is any simpler, more natural alternative to this class of instants, and (ii) whether these instants can be used to talk about the occurrence of events and holding of fluents. We discuss both in forthcoming sections.

*Expressing temporal relations:* Any temporal relation between two periods can be specified in terms of instant relations between the endpoints of the periods. In particular, every basic period relation in Figure 1.3 can be specified by a conjunction of binary instant relations. Furthermore, some temporal relations are more efficiently expressed in terms of relations between instant than between periods. For example

$$p_1 \texttt{Before Meets Overlaps Finished\_by Contains} \; p_2 \equiv \texttt{begin}(p_1) \prec \texttt{begin}(p_2)$$

However, the instant-based notation may be less efficient because of several reasons. First, a binary relation may become higher arity relation ($n \geq 2$) when stated in terms of instants. For example, given the periods $p_1$ and $p_2$,

$$p_1 \; \texttt{Before After} \; p_2 \equiv \texttt{end}(p_1) \prec \texttt{begin}(p_2) \; \vee \; \texttt{end}(p_2) \prec \texttt{begin}(p_1)$$

Second, instant-based expressions are sometimes more cumbersome. Third, as the number of events grows, the number of conjunctive combinations grows exponentially (as noticed by Tsang [Tsang, 1987a]). For example,

$$p_1 \; \texttt{Overlaps Finishes} \; p_2 \quad \wedge \quad p_2 \; \texttt{Meets Overlaps} \; p_3$$

is represented as the disjunction of each element of the cartesian set expressed in terms of instant relations, namely

$$\begin{array}{llll}
\texttt{begin}(p_1) \prec \texttt{begin}(p_2) \wedge \texttt{end}(p_1) \prec \texttt{end}(p_2) & \wedge & \texttt{end}(p_2) = \texttt{begin}(p_3) \\
\vee & & & \\
\texttt{begin}(p_1) \prec \texttt{begin}(p_2) \wedge \texttt{end}(p_1) \prec \texttt{end}(p_2) & \wedge & \texttt{begin}(p_2) \prec \texttt{begin}(p_3) \wedge \texttt{end}(p_2) \prec \texttt{end}(p_3) \\
\vee & & & \\
\texttt{begin}(p_2) \prec \texttt{begin}(p_1) \wedge \; \texttt{end}(p_1) = \texttt{end}(p_2) & \wedge & \texttt{end}(p_2) = \texttt{begin}(p_3) \\
\vee & & & \\
\texttt{begin}(p_2) \prec \texttt{begin}(p_1) \wedge \; \texttt{end}(p_1) = \texttt{end}(p_2) & \wedge & \texttt{begin}(p_2) \prec \texttt{begin}(p_3) \wedge \texttt{end}(p_2) \prec \texttt{end}(p_3)
\end{array}$$

expressed in terms of instant relations.

Having a compact, low order expression of temporal relations is not only a notational issue but it also has some impact on the computational cost of reasoning with them.

**Technical** Instant-based axiomatizations apparently allow for a better understanding and control of the properties we want for time. There is no general agreement on this point. The interested reader may compare the theories provided in the appendices.

## 1.7 Instants and Periods

According to the above analysis, it seems that an interesting approach would be a theory of time based on both instants and periods. It would enjoy the following advantages:

- *Natural expression*: Instants are used to express instantaneous events and fluents, and periods to express the durable ones.

- *Efficient notation and computation*: Either instant or period relations, according to what is more efficient, can be used to express the temporal relations at hand.

To define such a theory two alternatives have been explored: (i) starting with a concerted instant-period ontology, or (ii) defining instants from periods. Semantical arguments, such as the DIP, led a number of researchers to follow the second alternative. In Section 1.6 we have seen that simple techniques of representing an instant as a period do not work, and only more complex mathematical constructions do. The interest of deriving instants from periods is unclear since (as noted by Allen & Hayes [Allen and Hayes, 1989]) "we may end up in the same place" as if we start with an instants structure, whereas both the axioms and the instants construction are clearly less intuitive. Indeed we show (Section 1.7.2) that Allen & Hayes's theory [Allen and Hayes, 1985; Allen and Hayes, 1989] together with "derived instants" admits the same models than our instant-period theory of time.

Therefore, if we want both instants and periods its preferable to take the route of starting with both as ontological primitives of our time model. To our knowledge, the only proposal in this direction is Galton's theory of time which we discuss below and elsewhere in this collection.

### 1.7.1 Galton's Instants and Periods Theory

In Galton's theory [Galton, 1990] neither periods are a set-theoretic construction from instants nor vice versa. Both have the same ontological status as a primitive. The underlying intuition is

> "...there being an instant at the point where two periods meet".

Time is defined as the structure $\langle I, P, \texttt{Within}, \texttt{Limits}, \text{Allen's relations} \rangle$ where $I$ and $P$ are non-empty sets of instants and periods respectively, `Within` and `Limits` are instant-period relations with the obvious meaning. In addition to the 13 Allen's period relations, the period relation `In` is defined as the disjunction of `During`, `Starts` and `Finishes`. The set of axioms (we call it $\mathcal{G}$) is as follows ($i$ denotes an instant and $p, q, r$ periods):

$$\begin{array}{ll} \mathbf{G_1} & \forall p \exists i \; \texttt{Within}(i,p) \\ \mathbf{G_2} & \texttt{Within}(i,p) \wedge \texttt{In}(p,q) \Rightarrow \texttt{Within}(i,q) \\ \mathbf{G_3} & \texttt{Within}(i,p) \wedge \texttt{Within}(i,q) \Rightarrow \exists r \, (\texttt{In}(r,p) \wedge \texttt{In}(r,q)) \\ \mathbf{G_4} & \texttt{Within}(i,p) \wedge \texttt{Limits}(i,q) \Rightarrow \exists r \, (\texttt{In}(r,p) \wedge \texttt{In}(r,q)) \end{array}$$

We assume that by "...together with the various relations between intervals ..." ([Galton, 1990], p166) Galton means that $\mathcal{A}$ axioms are also included. Neither that nor a characterization of the models of the theory is formally given by Galton. $\mathcal{G}$ avoids DIP-like arguments and turns out to be useful for Galton to prove the relations between his temporal occurrence predicates (such as $\text{HOLDS}_{on}$, $\text{HOLDS}_{in}$, $\text{HOLDS}_{at}$), however they exhibit two major shortcomings. The first regards the relations between instants. They are not sufficient for the needs of a practical reasoner. For instance, the relation `Limits` is not sufficient for distinguishing between the begin and the end of a period. Notice that there is no account for any ordering over instants. Although it is a very basic notion, it is neither explicitly stated nor induced by the period axioms as we discuss next. The second shortcoming regards the connection between instants and periods. Although it is not easy to figure out what are the intended models of $\mathcal{G}$, a careful analysis reveals that the theory is not strong enough to

properly connect instants and periods. It is easy to identify examples of counter-intuitive, accepted models:

**Example 1.7.1.** *Let us take a basic model $M$ composed of an infinite set of periods and Allen's relations satisfying $\mathcal{I}_\mathcal{A}$ axioms plus an infinite set of instants which make $M$ satisfy $I_1$ –for example INT($\mathcal{Q}$) as periods and $\mathcal{Q}$ as instants:*

- *Example model 1: $M$ plus a single instant $i \notin \mathcal{Q}$ which limits a certain period $P$ in $M$ and only that one. In particular it does not limit any of those periods that meet or are met by $P$.*
- *Example model 2: $M$ plus a single instant $i \notin \mathcal{Q}$ which limits a certain period $P$ in $M$ and is not within any period. In particular it is not within any of those periods that overlap $P$.*
- *Example model 3: $M$ plus a single instant $i \notin \mathcal{Q}$ which limits a certain period $P$ in $M$ and also is within $P$.*

The obvious undesirable consequence of $\mathcal{G}$ weakness is that some queries will not receive the expected intuitive answers. In the first example, given the assertions $\texttt{Within}(i,p)$ and $\texttt{Meets}(p,p')$, it is not possible to derive an answer for the query $\texttt{Limits}(i,p')$.

## 1.7.2 $\mathcal{IP}$

$\mathcal{IP}$ [Vila and Schwalb, 1996] has two sorts of symbols, instants ($\mathcal{I}$) and periods ($\mathcal{P}$) which are formed by two infinite disjoint sets of symbols, and three primitive binary relation symbols $\prec$: $\mathcal{I} \times \mathcal{I}$ and $\texttt{begin}, \texttt{end} : \mathcal{I} \times \mathcal{P}$.

The first-order axiomatization of $\mathcal{IP}$ theory is as follows:

$$
\begin{array}{ll}
\mathbf{IP_1} & \neg(i \prec i) \\
\mathbf{IP_2} & i \prec i' \Rightarrow \neg(i' \prec i) \\
\mathbf{IP_3} & i \prec i' \wedge i' \prec i'' \Rightarrow i \prec i'' \\
\mathbf{IP_4} & i \prec i' \vee i \prec i' \vee i = i' \\
\mathbf{IP_{5.1}} & \exists i' \, (i' \prec i) \\
\mathbf{IP_{5.2}} & \exists i' \, (i \prec i') \\
\mathbf{IP_6} & \texttt{begin}(i,p) \wedge \texttt{end}(i',p) \Rightarrow i \prec i' \\
\mathbf{IP_{7.1}} & \exists i \, \texttt{begin}(i,p) \\
\mathbf{IP_{7.2}} & \exists i \, \texttt{end}(i,p) \\
\mathbf{IP_{8.1}} & \texttt{begin}(i,p) \wedge \texttt{begin}(i',p) \Rightarrow i = i' \\
\mathbf{IP_{8.2}} & \texttt{end}(i,p) \wedge \texttt{end}(i',p) \Rightarrow i = i' \\
\mathbf{IP_9} & i \prec i' \Rightarrow \exists \, p \, (\texttt{begin}(i,p) \wedge \texttt{end}(i',p)) \\
\mathbf{IP_{10}} & \texttt{begin}(i,p) \wedge \texttt{end}(i',p) \wedge \\
 & \wedge \texttt{begin}(i,p') \wedge \texttt{end}(i',p') \Rightarrow p = p'
\end{array}
$$

$\mathbf{IP_1}$ – $\mathbf{IP_4}$ are the conditions for $\prec$ to be a *strict linear order* — namely irreflexive, asymmetric, transitive and linear– relation over the instants*. $\mathbf{IP_5}$ imposes unboundedness on this ordered set. $\mathbf{IP_6}$ orders the extremes of a period. This axiom rules out durationless

*Notice that $\mathbf{IP_1}$ is actually redundant since it can be derived from $\mathbf{IP_2}$. We include it for clarity.

periods which are not necessary since we have instants as a primitive. The pairs of axioms $\mathbf{IP_{7_{..}}}$ and $\mathbf{IP_{8_{..}}}$ formalize the intuition that the beginning and end instants of a period always *exist* and are *unique* respectively. Conversely, axioms $\mathbf{IP_9}$ and $\mathbf{IP_{10}}$ close the connection between instants and periods by ensuring the *existence* and *uniqueness* of a period for a given ordered pair of instants.

Next we characterize the models of $\mathcal{IP}$ and determine its relationships with other theories.

### The Models

The models are defined over $\mathcal{IP}$-structures.

**Definition 1.7.1.** **($\mathcal{IP}$-structure)** *An $\mathcal{IP}$-structure is a tuple $\langle \mathcal{I}_d, \mathcal{P}_d, <_d, \mathtt{begin}_d, \mathtt{end}_d \rangle$ where $\mathcal{I}_d$ and $\mathcal{P}_d$ are sets of instants and periods respectively, $<_d$ is a binary relation on $\mathcal{I}_d$ and $\mathtt{begin}_d, \mathtt{end}_d$ are binary relations on $\mathcal{I}_d, \mathcal{P}_d$.*

Periods are merely viewed as ordered pairs of instants.

**Definition 1.7.2.** (pairs) *Given a set $\mathcal{S}$ over which an ordering relation $<$ is defined, we note by $\mathtt{pairs}(\mathcal{S})$ the set of $<$-ordered pairs of distinct elements of $\mathcal{S}$: $\mathtt{pairs}(\mathcal{S}) = \{(x, y) \mid x, y \in \mathcal{S} \wedge x < y\}$. Over a set of pairs we define the following relations: (i) first$(x, (y, z)) \stackrel{def}{=} x = y$ and (ii) second$(x, (y, z)) \stackrel{def}{=} x = z$.*

Now we show — similar to Ladkin [Ladkin, 1987] — that the elements and the pairs of an unbounded linear order $\mathcal{S}$ form a model for $\mathcal{IP}$.

**Theorem 1.7.1.** **(a model)** *Given an infinite set $\mathcal{S}$ and an unbounded strict linear order $<$ on it, the $\mathcal{IP}$-structure $\langle \mathcal{S}, \mathtt{pairs}(\mathcal{S}), <, first, second \rangle$ forms a model of $\mathcal{IP}$.*

*Proof.* (sketch) It is easy to prove that every axiom of $\mathcal{IP}$ is satisfied if we interpret the instants on the set $\mathcal{S}$, the periods on the set $\mathtt{pairs}(\mathcal{S})$, the ordering as $<$, and begin and end relations as first and second respectively. □

Indeed these are the only models of $\mathcal{IP}$.

**Theorem 1.7.2.** **(the models)** *Any model $M = \langle \mathcal{I}_d, \mathcal{P}_d, <_d, \mathtt{begin}_d, \mathtt{end}_d \rangle$ of $\mathcal{IP}$ is* isomorphic *to the structure $\langle \mathcal{I}_d, \mathtt{pairs}(\mathcal{I}_d), <_d, first, second \rangle$ where pairs, first and second are defined as above.*

**Corollary 1.** *Every model of $\mathcal{IP}$ is completely characterized by an infinite set $\mathcal{S}$ and an unbounded strict linear order $<$ on it (not necessarily dense).*

Note that $\mathcal{IP}$ accepts both dense and discrete models of time.

### Dense $\mathcal{IP}$

The sub-theory that embraces dense models only, we call it $\mathcal{IP}_{dense}$, is axiomatized by adding the *denseness* axiom over instants:

$$\mathbf{IP_{11}} \quad i \prec i' \Rightarrow \exists i'' \, (i \prec i'' \prec i')$$

**Theorem 1.7.3.** **(dense models)** *The models of $\mathcal{IP}_{dense}$ are characterized by the set of elements and the set of ordered pairs of distinct elements of an unbounded,* dense, *strict linearly ordered set. Moreover $\mathcal{IP}_{dense}$ is a* complete *axiomatization for the theory of rationals and rational intervals, namely $Th(\mathcal{Q}, INT(\mathcal{Q}))$.*

**Relation between $\mathcal{IP}$ and $\mathcal{AH}$**

To compare our theory with Allen & Hayes theory (called $\mathcal{AH}$) we use the same technique as Ladkin [Ladkin, 1987]. Instants are derived from periods by first defining the notion of pair of meeting periods, second applying the equivalence relation "having the same meeting point" and, finally, associating an instant to each class. Let us call the resulting theory $\mathcal{I}_{\mathcal{AH}_{\sim_\mathcal{I}}}$. Its class of models is the same as $\mathcal{IP}$.

**Theorem 1.7.4.** $\mathcal{IP} \equiv \mathcal{I}_{\mathcal{AH}_{\sim_\mathcal{I}}}$

**Theorem 1.7.5.** $\mathcal{IP}_{dense} \equiv Th(\mathcal{AH}, N_1)$

**Relation between $\mathcal{IP}$ and $\mathcal{G}$**

As we discuss in Section 1.6, the instants in $\mathcal{G}$ do not correspond with the places where periods meet. For instance, nothing forces the instant that exists within a period by axiom $\mathbf{G_1}$ to be a place where two periods meet. They do not correspond to the idea of period endpoints either, which is our approach. As a matter of fact, $\mathcal{G}$ is weaker than $\mathcal{IP}_{dense}$.

**Theorem 1.7.6.** $\mathcal{IP}_{dense} \subset \mathcal{G}$

*Proof.* (sketch) $\mathcal{G}$ axioms are derived from $\mathbf{IP_{10}}$, linearity, extremes ordering, existence of both instants and periods and density. □

The reason of $\mathcal{G}$ weakness is the loose connection between instants and periods. There is no direct relation between $\mathcal{IP}$ and $\mathcal{G}$: $\mathcal{IP}$ accepts discrete models, whereas $\mathcal{G}$ imposes a sort of denseness by axiom $\mathbf{G_1}$. Characterize the models of $\mathcal{G}$ and its relation with $\mathcal{IP}$ is an open issue.

## 1.8 Temporal Incidence

For the sake of showing how a time theory is put at work, we complement this survey of theories of time with a section on temporal incidence, we propose a specific temporal incidence theory that works very well with an instant-period time theory, and, finally, we show on an example how the representational issues above a outlined th a classical example from naive physics.

Classical temporal logics in AI [McCarthy and Hayes, 1969; McDermott, 1982; Allen, 1984; Shoham, 1987; Haugh, 1987; Galton, 1991] mostly agree upon the temporal incidence properties that distinguish *fluents* from *events*. Fluents hold homogeneously whereas events event occurrences are anti-homogeneous. Instant-based approaches allow (i) a direct expression of instantaneous events and fluents, and (ii) an easy specification of temporal incidence properties. In McDermott's framework, for example, homogeneity of fluents is specified by

$$\text{Throughout}(T_1, T_2, F) \Leftrightarrow \forall\, T_1 \leq T \leq T_2\ \text{True}(T, F)$$

These advantages are more obvious in Shoham's work where a much richer categorization of proposition types is defined.

In period-based theories temporal incidence specification is more difficult. For example, Allen's axiom for fluent homogeneity is as follows:

$$\textbf{H.2} \quad \text{HOLDS}(F, I) \Leftrightarrow \forall\, I' \in I\ \exists I'' \in I'\ \text{HOLDS}(F, I'')$$

It is not only a cumbersome axiom, but in fact allows some non-intended models*. Moreover Galton [Galton, 1990] proves that axiom **H.2** conflicts with axiom **H.4** which specifies the *holding of negated fluents*. It is not clear how it can be avoided.

Also, period-based theories have problems to express the holding of a fluent at an instant, either because instants cannot be directly referred or because of the DIP. Allen & Hayes's reply to this issue as follows ([Allen and Hayes, 1989], Section 4):

> "We avoid it by resolutely refusing to allow fluents to hold at points".

They propose the following alternative: "One could define a notion of a fluent $X$ being true at a point $p$ by saying that $X$ is true at $p$ just when there is some interval $I$ containing $p$ during which $X$ is true". It is easy to see that it does not work for modelling continuous fluents (consider the $v = 0$ fluent in the TBS). In Section 1.10 we determine the conditions for the DIP to be a problem and we propose a simple approach that satisfies them.

Let us now address the issue of modelling continuous change. There is a general agreement upon the importance of this issue for common-sense reasoning. In spite of it, no previous work formally accounts for the essential temporal incidence differences between holding of discrete and continuous fluents. Galton's work [Galton, 1990] is the only attempt in that direction, up to our knowledge. Fluents are diversified into instantaneous/durable and states of position/states of motion:

> "A *state of position* can hold at isolated instants; if it holds during a period it holds at its limits (e.g. a quantity taking a particular value). ... A *state of motion* cannot hold at isolated instants (e.g. a body being at rest)."

Galton's approach presents two problems:

1. The utility of Galton's new classes of fluents is not clear since more than one class is needed to model a single continuously changing parameter. Let us illustrate it with the TBS. Consider the fluent $f = (v \neq 0)$. It cannot be modelled as a state of position because $f$ holds on both $p_1$ and $p_2$ which must contain the limiting instant $i$ where $\neg f$ holds ($v = 0$). A state of motion cannot be used either because it cannot hold at isolated instants: we are not allowed to say that $\neg f$ is true at $i$.

2. While states of position are *concatenable*, states of motion are not always. It is rather counter-intuitive: it seems that states of position should not be concatenable since the parameter they represent may have a different value at the meeting point. Since it is not the case for states of motion it seems that they should be concatenable. Next we follow this intuition.

*An instance, due to Shoham, is a model in which time has the structure of real numbers and a property holds only over all its subintervals whose endpoints are rational.

## 1.9 $\mathcal{CD}$

$\mathcal{CD}$ the theory of temporal incidence initially proposed in [Vila and Schwalb, 1996], is based on the following ideas:

1. *We allow fluents to hold at points.* It allows modelling continuously changing fluents and makes the resulting theory much simpler to define. We discuss why it in fact does not originate any problem.

2. *We distinguish between continuous and discrete fluents.* We diversify fluents according to whether the change on the parameter they model is *continuous* or *discrete*.

To present our approach we assume the standard temporal reified first-order language with equality (as in [McDermott, 1982; Allen, 1984; Galton, 1990]) as underlying language. We propose a temporal representation with the following features:

- *Time theory*: We take $\mathcal{IP}_{dense}$. We define the instant-to-period relations (such as `Within`) and period-to-period relations (such as `Meets`) upon $\prec$, `begin` and `end`.

- *Reified propositions*: Reified propositions are classified into *continuous fluents*, *discrete fluents** and *events*.

- *Temporal Incidence Predicates (TIPs)*. We introduce a different TIP for each combination of temporal proposition and temporal primitive (similar to [Kowalski and Sergot, 1986; Galton, 1990]):

| | | |
|---|---|---|
| $\text{HOLDS}^{\sim}_{on}(f,p)$ | $\stackrel{def}{=}$ | The continuous fluent $f$ holds throughout the period $p$ |
| $\text{HOLDS}^{\neg}_{on}(f,p)$ | $\stackrel{def}{=}$ | The discrete fluent $f$ holds throughout the period $p$ |
| $\text{HOLDS}^{\sim}_{at}(f,i)$ | $\stackrel{def}{=}$ | The continuous fluent $f$ holds at the instant $i$ |
| $\text{HOLDS}^{\neg}_{at}(f,i)$ | $\stackrel{def}{=}$ | The discrete fluent $f$ holds at the instant $i$ |
| $\text{OCCURS}_{on}(e,p)$ | $\stackrel{def}{=}$ | The event $e$ occurs on the period $p$ |
| $\text{OCCURS}_{at}(e,i)$ | $\stackrel{def}{=}$ | The event $e$ occurs at the instant $i$ |

**Terminology.** Henceforth we use the following notational shorthands. We use `begin` and `end` in functional form (e.g. $i = \texttt{begin}(p)$). $\text{HOLDS}_{on}$ stands for both $\text{HOLDS}^{\sim}_{on}$ and $\text{HOLDS}^{\neg}_{on}$, and $\text{HOLDS}_{at}$ for $\text{HOLDS}^{\sim}_{at}$ and $\text{HOLDS}^{\neg}_{at}$.

$\mathcal{CD}$ axioms are as follows. Since instants and periods are both primitives, we are not forced to accept any assumption on the relation between the holding of a fluent on a period and its holding at the period endpoints. A fluent holds during a period iff it holds at its *inner* instants:

$$\textbf{CD}_1 \quad \text{HOLDS}_{on}(f,p) \Leftrightarrow (\texttt{Within}(i,p) \Rightarrow \text{HOLDS}_{at}(f,i))$$

From it, nothing can be derived about the holding of $f$ at $\texttt{begin}(p)$ and $\texttt{end}(p)$.

**Continuous Fluents** A continuous fluent may hold both during a period and at a particular instant without any restriction. This is not the case for discrete ones.

* We use the equality relation to express a fluent representing a parameter taking a certain value. E.g. the speed of a ball being positive on $p$ is expressed as HOLDS(speed $= +, p$). We omit necessary axioms imposing the exclusivity among the different values of a parameter.

**Discrete Fluents** The genuine property of discrete fluents is that *they cannot hold at an isolated instant*:

$$\mathbf{CD_2} \quad \text{HOLDS}^{\rightarrow}_{at}(f,i) \Rightarrow \exists p\,(\text{HOLDS}^{\rightarrow}_{on}(f,p) \wedge (\texttt{Within}(i,p) \vee \texttt{begin}(i,p) \vee \texttt{end}(i,p)))$$

Our distinction between continuous and discrete events is different from Galton's distinction between states of position and states of motion. Identifying it as a key property in modelling changing domains is a contribution of this chapter.

**Non-Instantaneous Events** The intuition behind events (both instantaneous and durable) is that of an *accomplishment* that may have relevant consequences over the state of the world. Unlike preceding approaches, *our theory does not include any axiom governing the occurrence of events that take time.* It reflects the intuition that whether two *accomplishments* may happen concurrently depends on the abstraction degree of the analysis. For example, the event "I programmed the program p1" can not occur over two periods that are not disjoint. It is not the case, however, if the event under consideration is merely "programming a program". Therefore, no domain-independent axiom can be stated as part of a general theory of temporal incidence.

**Non-Atomic Fluents** Our theory directly addresses the issue of the holding of non-atomic fluents with the following axioms:

$$\begin{array}{lll}
\textit{Negation:} & \mathbf{CD_3} & \text{HOLDS}_{at}(\neg f,i) \Leftrightarrow \neg\text{HOLDS}_{at}(f,i) \\
\textit{Conjunction:} & \mathbf{CD_4} & \text{HOLDS}_{at}(f \wedge f',i) \Leftrightarrow \text{HOLDS}_{at}(f,i) \wedge \text{HOLDS}_{at}(f',i) \\
\textit{Disjunction:} & \mathbf{CD_5} & \text{HOLDS}_{at}(f \vee f',i) \Leftrightarrow \text{HOLDS}_{at}(f,i) \vee \text{HOLDS}_{at}(f',i)
\end{array}$$

Deriving the properties of non-instantaneous holding of non-atomic fluents from these axioms is straight forward.

## 1.10 Revisiting the Issues

Let us see now how the problems presented in Section 1.2 are addressed using $\mathcal{IP}$ together with $\mathcal{CD}$.

**Instantaneous Events** Since we take instants as primitive, we can directly express instantaneous events using the predicate $\text{OCCURS}_{at}$. In the DIP scenario, for instance, we can write $\text{OCCURS}_{at}(\text{switchoff}, i)$. In Section 1.11 we discuss how to handle *sequences of instantaneous events.*

**Instantaneous Holding** We allow talking about a instantaneous holding of fluents by using $\text{HOLDS}_{at}$ predicates. Axiom $\mathbf{CD_1}$ ensures that we are able to express the holding of contradictory fluents ending or beginning at a certain instant without conflict. Furthermore, we can express the holding of a continuous fluent at an isolated instant. The TBS scenario, for example, is merely represented as follows:

$$\begin{array}{l}
\text{HOLDS}^{\sim}_{on}(\text{speed} = +, p_1) \\
\text{HOLDS}^{\sim}_{at}(\text{speed} = 0, i) \qquad \texttt{end}(p_1) = i = \texttt{begin}(p_2) \\
\text{HOLDS}^{\sim}_{on}(\text{speed} = -, p_2)
\end{array}$$

**The Dividing Instant Problem** The DIP is not a problem for temporal incidence theories where the following two conditions hold:

1. The holding of a fluent over a period does not constrain its holding at the period's endpoints.

2. One can express that a fluent holds at an instant.

These conditions avoid logical contradiction and a truth gap at the dividing instant, respectively. In Figure 1.2 fluent $f$ can be regarded as discrete and the DIP scenario can be formalized as follows:

$$\text{HOLDS}^{\neg}_{on}(\text{light} = \text{on}, p_1) \wedge \texttt{Meets}(p_1, p_2) \wedge \text{HOLDS}^{\neg}_{on}(\text{light} = \text{off}, p_2)$$

Given this information only, the query $\text{HOLDS}^{\neg}_{on}(\text{light} = \text{on}, \texttt{end}(p_1))$ merely gets no answer. The additional information required to answer it is domain-dependent. Some fluents hold on and at the end of a period (e.g. the fluent "being in contact with the floor" for a ball being lifted up), other fluents hold at the beginning and throughout the period (e.g. "not being in contact with the floor" for a ball that falls on the floor). In the light example, the most appropriate might be having three fluents light=on, light=off and light=changing, where the first and the second hold over open periods and the third holds at the dividing instant. Our approach avoids making any commitment about the holding at period's endpoints, whereas provides the means to safely specify what happens *at* the dividing instant. It requires an adequate theory of concatenability that we present below.

**Non-Instantaneous Fluent Holding** A nice feature of our proposal is that the above few axioms are enough to easily derive the fundamental properties of temporal incidence of fluents. For instance, *Allen's Homogeneity* $\text{HOLDS}^{\sim}_{on}(f,p) \Leftrightarrow \texttt{In}(p',p) \Rightarrow \text{HOLDS}^{\sim}_{on}(f,p')$ is easily derived from $\mathbf{CD_1}$. Before presenting the concatenability properties we define few basic notions. Given any two periods $p, p'$ such that $\texttt{Meets}(p,p')$, $\texttt{meetpoint}(p,p') \stackrel{def}{=} \texttt{end}(p)$ and $\texttt{concat}(p,p') \stackrel{def}{=} p''$ s.t. $\texttt{begin}(p'') = \texttt{begin}(p) \wedge \texttt{end}(p'') = \texttt{end}(p')$. Also $\texttt{In} : \mathcal{P} \times \mathcal{P} \stackrel{def}{=} \texttt{Starts} \vee \texttt{During} \vee \texttt{Finishes}$, $\texttt{Disjoint}^{\neg}_{on} : \mathcal{P} \times \mathcal{P} \stackrel{def}{=} \texttt{Before} \vee \texttt{After}$ and $\texttt{Disjoint}^{\sim}_{on} : \mathcal{P} \times \mathcal{P} \stackrel{def}{=} \texttt{Before} \vee \texttt{Meets} \vee \texttt{Met\_by} \vee \texttt{After}$. The properties for concatenability are as follows:

**Theorem 1.10.1. (Concatenability of discrete fluents)**
*If* $\texttt{Meets}(p,p')$ *then*

$$\text{HOLDS}^{\neg}_{on}(f,p) \wedge \text{HOLDS}^{\neg}_{on}(f,p') \Leftrightarrow \text{HOLDS}^{\neg}_{on}(f, \texttt{concat}(p,p'))$$

**Theorem 1.10.2. (Concatenability of continuous fluents)**
*If* $\texttt{Meets}(p,p')$ *then*

$$\text{HOLDS}^{\sim}_{on}(f,p) \wedge \text{HOLDS}^{\sim}_{on}(f,p') \wedge \text{HOLDS}^{\sim}_{at}(f, \texttt{meetpoint}(p,p')) \Leftrightarrow$$
$$\Leftrightarrow \text{HOLDS}^{\sim}_{on}(f, \texttt{concat}(p,p'))$$

Concatenability can be regarded as a special case of *joinability*. Given two periods $p$ and $p'$, $\texttt{join}(p,p')$ is defined as a period $p''$ such that $\texttt{begin}(p'') = \texttt{min}(\texttt{begin}(p), \texttt{begin}(p'))$ and $\texttt{end}(p'') = \texttt{max}(\texttt{end}(p), \texttt{end}(p'))$, where min and max are defined according to $\prec$.

**Theorem 1.10.3. (Joinability of discrete fluents)**
*If* $\neg\texttt{Disjoint}^{\rightarrow}_{on}(p,p')$ *then*

$$\textsc{Holds}^{\rightarrow}_{on}(f,p) \wedge \textsc{Holds}^{\rightarrow}_{on}(f,p') \Leftrightarrow \textsc{Holds}^{\rightarrow}_{on}(f,\texttt{join}(p,p'))$$

**Theorem 1.10.4. (Joinability of continuous fluents)**
*If* $\neg\texttt{Disjoint}^{\sim}_{on}(p,p')$, *or*
$\texttt{Meets}(p,p') \wedge \textsc{Holds}_{at}(f,\texttt{meetpoint}(p,p'))$, *or*
$\texttt{Met\_by}(p,p') \wedge \textsc{Holds}_{at}(f,\texttt{meetpoint}(p',p))$
*then*

$$\textsc{Holds}^{\sim}_{on}(f,p) \wedge \textsc{Holds}^{\sim}_{on}(f,p') \Leftrightarrow \textsc{Holds}^{\sim}_{on}(f,\texttt{join}(p,p'))$$

There are also a number properties relating the holding of contradictory fluents at distinct, related times.

**Theorem 1.10.5. (non-holding of discrete fluents)**

$$\neg\textsc{Holds}^{\rightarrow}_{on}(f,p) \Leftrightarrow \exists p'\ \texttt{In}(p',p) \wedge \textsc{Holds}^{\rightarrow}_{on}(\neg f,p')$$

**Theorem 1.10.6. (non-holding of continuous fluents)**

$$\neg\textsc{Holds}^{\sim}_{on}(f,p) \Leftrightarrow \exists i\ \texttt{Within}(i,p) \wedge \textsc{Holds}^{\sim}_{at}(f,i)$$

**Theorem 1.10.7. (disjointness)**

$$\textsc{Holds}_{on}(f,p) \wedge \textsc{Holds}_{on}(\neg f,p') \Rightarrow \texttt{Disjoint}^{\sim}_{on}(p,p')$$

At this point one may ask for how long can we go enumerating properties of temporal incidence. To answer this question, let us analyze the issue from a more general perspective. The above properties are particular cases of the following scheme ($f$ is a fluent and $\overline{p}$ denotes the collection of periods $p_1,\ldots,p_n$) :

If $\textsc{Holds}(f,\overline{p})$ and $f \models f'$ then $\textsc{Holds}(f',\overline{p}')$
If $\textsc{Holds}(f,\overline{p})$ and $f \models \neg f'$ then $\textsc{Holds}(\neg f',\overline{p}')$

The scope of this chapter goes as far as showing that the most basic properties of this scheme are theorems of our theory.

## 1.11 Example: Modelling Hybrid Systems

In this section we illustrate the application of the proposed theory in qualitative modelling of physical systems. A (qualitative) model that includes both discrete and continuously changing parameters is called a *hybrid model*. Many physical systems, such as most electromechanical devices like photocopiers, cars, stereos, are suitably modelled as a hybrid model. Several approaches have been proposed to represent "qualitative" hybrid models [Nishida and Doshita, 1987; Forbus, 1989; Iwasaki and Low, 1992; Iwasaki *et al.*, 1995], however some semantical problems arise because of the different nature of discrete and continuous change. We shall see that an adequate theory of time and temporal incidence is fundamental to overcome them.

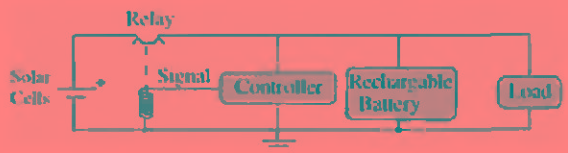



Figure 1.4: The hybrid circuit example.

Let us consider a particular example from [Iwasaki *et al.*, 1995] (we borrow the example, the qualitative model, the intended environment and a tentative solution). Figure 1.4 shows a simple circuit in which electric power is provided to a *load* either by a *solar cells* array or by a *re-chargeable battery*.
A part of the continuous behavior of this system is described as follows:

**C0:** "If the sun is shining and the relay is closed then the solar array acts as a constant current source and the battery accumulates charge."

The discrete events are specified as follows:

**D1:** "If the relay is closed, when the signal from the controller goes high, then the relay opens."

**D2:** "If the relay is open, when the signal from the controller goes low, then the relay closes."

**D3:** "If the signal is low, when the controller detects that the charge level in the battery has reached the threshold $q_2$, then the controller turns on the signal to the relay."

Now, let us consider a particular predicted qualitative behavior. A qualitative behaviour is described as a sequence of states that hold alternatively at an instant and throughout a period. The transition from one state to another is produced either by a continuous or by a discrete change. Quoting Iwasaki *et al* [Iwasaki *et al.*, 1995]: "...we would like to model discrete events as being instantaneous". Problems arise when sequences of them occur. For instance "the signal goes high and *immediately after* the relay closes". The following predicted behavior and the explanation about why sequences of discrete events are problematic are borrowed from [Iwasaki *et al.*, 1995]. Given the initial state of our example where the *signal* is *low*, the *relay* is *closed* and the sun is shining, the intended environment would be as follows:

| | | | |
|---|---|---|---|
| | $Q_{BA} = q_2$ | $signal = low$ | $relay = closed$ |
| | $Q_{BA} = ?$ | $signal = high$ | $relay = closed$ |
| $t_{2.2}$ | $Q_{BA} = ?$ | $signal = high$ | $relay = open$ |
| $(t_{2.2}, .)$ | $Q_{BA} < q_2$ | $signal = high$ | $relay = open$ |

The state $s_{2.1}$ is produced by the signal going high, and $s_{2.2}$ by the relay closing.

It is not clear how to model neither the times of $s_2$, $s_{2.1}$ and $s_{2.2}$ nor the time spans and the discrete events between them. If we assume that discrete events take no time, we encounter the following logical problem: "The antecedent for rules specifying discrete events often includes the negation of the consequence; this leads to a contradiction when events

are treated as implications." An alternative is assuming that discrete changes take a very little period. It is problematic too since the value of every continuous variable that changes concurrently becomes unknown after a sequence of actions. In the example, the charge of the battery would keep continuously increasing for a short period. After a number of discrete events these small variations accumulate and complicate the computation of parameter values.

Several solutions have been proposed to solve this quandary. They are based on complicating the model of time by either introducing *mythical* time ([Nishida and Doshita, 1987], *direct method*), extending the real numbers with infinitessimals ([Nishida and Doshita, 1987], *approximation method*)[Alur *et al.*, 1993], or using non-standard analysis [Iwasaki *et al.*, 1995]. Next we show that none of these is necessary. We use our theory of instants/periods and continuous fluents/discrete fluents/events as follows:

- Discrete events are modelled as instantaneous events.
- Continuous/discrete quantities are modelled as continuous/discrete fluents.

Since $\text{HOLDS}_{on}$ is defined as holding at the inner points only, the value of a fluent that changes because of an instantaneous event is not defined at the time that the event's time unless there is some specific knowledge about it. The sequence of states representing the intended environment becomes simpler:

| | | | | |
|---|---|---|---|---|
| $s_1$ | $(t_1, t_2)$ | $Q_{BA} < q_2$ | $signal = low$ | $relay = closed$ |
| $s_2$ | $t_2$ | $Q_{BA} = q_2$ | $signal = ?$ | $relay = ?$ |
| $s_3$ | $(t_2, \_)$ | $Q_{BA} < q_2$ | $signal = high$ | $relay = open$ |

Indeed, this solution is much simpler than the previously proposed techniques. The formalization of the environment is as follows:

| | |
|---|---|
| $\text{HOLDS}^{\sim}_{on}(Q_{BA} < q_2, p_1)$ | |
| $\text{HOLDS}^{\neg}_{on}(signal = low, p_2)$ | |
| $\text{HOLDS}^{\neg}_{on}(relay = closed, p_3)$ | |
| $\text{HOLDS}^{\sim}_{at}(Q_{BA} = q_2, \text{end}(p_1))$ | |
| $\text{OCCURS}_{at}(turn\_on(signal), \text{end}(p_1))$ | $\text{end}(p_1) = \text{end}(p_2)$ |
| $\text{HOLDS}^{\neg}_{on}(signal = high, p_4)$ | $\texttt{Meets}(p_2, p_4)$ |
| $\text{OCCURS}_{at}(open(relay)), \text{end}(p_3))$ | $\text{end}(p_2) = \text{end}(p_3)$ |
| $\text{HOLDS}^{\neg}_{on}(relay = open, p_5)$ | $\texttt{Meets}(p_3, p_5)$ |

## 1.12 Concluding Remarks

A theory of time and temporal incidence is the foundation for a proper temporal representation, independently of both the temporal qualification method and the underlying representation language. In this chapter we identified the problematic issues that need to be addressed, namely the expression of *instantaneous events* and *fluents*, the *dividing instant problem* (DIP) and the formalization of the properties *for non-instantaneous holding of fluents*.

In this chapter we have surveyed the most relevant theories of time in Artificial Intelligence and we have discussed the pros and cons of each of them. Also, we presented a brief overview of temporal incidence theories and proposes a theory of *temporal incidence* called $\mathcal{CD}$ defined upon a theory of instants and periods (such as $\mathcal{IP}$) whose key insight is the distinction between continuous and discrete fluents.

Handbook of Temporal Reasoning in Artificial Intelligence
Edited by M. Fisher, D. Gabbay and L. Vila


# Chapter 2

# Eventualities

## Antony Galton

The previous chapter has discussed the many different ways in which we can construct formal models of time. These models provide temporal frameworks within which it is possible to represent many of things which can go on in time, but they do not on their own provide a means for representing those things themselves. Such a model of time is like a new diary or calendar which provides the dates in their correct temporal relationships but has nothing written into it yet. This chapter is concerned with the kinds of things that can be written into such a temporal framework.

## 2.1 Introduction

What are these "things"? There are many general words to describe various classes of things which we might want to write into our calendar. Suppose I mark over a week in August the words "On holiday all week". This indicates a *state* which obtains throughout that week. On the other hand, I might mark, on a particular day, "Heathrow depart 10.25 a.m.". This refers to an *event* which happens at a particular point in time. This distinction between states and events is one of the most fundamental, but there are many other distinctions that have been made between many subtly or not-so-subtly different kinds of things that can go on in time. Some words that have been used for this purpose are *situation*, *state of affairs*, *process*, *act*, *action*, *activity*, *accomplishment*, *achievement*, *happening*, *occurrence*, and *eventuality*.

Most of these words are used in everyday language with more or less precise non-technical meanings, but they have also all been used as technical terms, sometimes in different ways by different authors. A need has often been felt for a single most general term to cover the whole range of meanings presented by these terms. Working in a linguistic context, Comrie [Comrie, 1976, p. 13] suggested *situation* for this purpose, but in an Artificial Intelligence context this term has a rather more specific meaning and is therefore inappropriate. The term *eventuality* was suggested by Bach [Bach, 1986] as a catch-all to cover events, states, processes, and the like, and as it has not, as far as I know, been used for any more specific purpose I shall use it in this chapter as the general term to cover all the categories mentioned above.

Two terms that I shall not be using in this context are *fact* and *property*, both of which have been used for what is better referred to as a *state*. The former was used by McDermott [McDermott, 1982] to contrast with *event*: you could have a fact holding at a time or an event happening at a time. Unfortunately this does not fit in with our normal use of the term 'fact':

we might say that it is a fact that I was in London on 24th March, but equally that it is a fact that I went to London on 24th of March. The former is a fact about a state, the latter a fact about an event. I take it that the correct way to talk about facts is as expounded by Jonathan Bennett [Bennett, 1988]. Similarly, it seems odd to call my being in London a *property*, as James Allen did in [Allen, 1984]. It seems more natural to say that being in London was a property that I temporarily possessed on 24th of March, and one might just say that *my* being in London was a property of that *time* (though this sounds a little odd); again, though, this does not establish the desired contrast with events, since one could equally say that my going to London (an event) was a property of the time at which it occurred.

The plan of this chapter is to survey systematically the range of different eventualities that can be described in a series of systems of increasing complexity. We begin with the simplest of all possible systems, a system with just one primitive two-valued state undergoing variation in discrete time. We shall see that even with such a simple system it is already possible to describe an astonishing range of temporal phenomena, which correspond at least roughly with many of the concepts we use in describing temporal phenomena in the real world. We next complicate matters somewhat by introducing more than one state—or equivalently, allowing our single state to become many-valued, i.e., a fluent. The next complication we introduce is to model time as continuous rather than discrete, but still insisting that our primitive fluents undergo only discrete variation. The final stage in the development is to look at the case of continuous change, where the variation in value of a fluent needs to be represented by means of a continuous function of time. We shall close our survey with an examination of a single real-world system, which, by viewing it at different levels of granularity, we can model in any of the ways we have described.

## 2.2 One state in discrete time

### 2.2.1 The simplest possible temporal system

The simplest possible system in which temporal phenomena occur has one state, which may be either "on" or "off". Assuming a discrete time series, the evolution of such a system can be portrayed as a bit-string of indefinite length, the 1s denoting the "on" state, the 0s the "off" state:

$$\cdots 0001101101\ 0010100101\ 0001010000\ 1111101000 \cdots$$

Such a sequence is sometimes known as a *history*. For convenience we shall always group the bits in a history into blocks of ten. We shall call the "on" state of this system $S$; the "off" state is its negation which we shall denote $-S$. Assigning an integer designation to each time in the discrete series, with the usual order, we shall write $Holds(S, n)$ to mean that the "on" state holds at the time denoted $n$, and likewise $Holds(-S, n)$ to indicate that the "off" state holds then. We shall write things like $t = n$ to indicate that a time $t$ that we are interested in is assigned the integer value $n$.

We shall write $[m, n]$ to denote the interval composed of all the atomic intervals $k$ such that $m \leq k \leq n$. Note that in this notation $[m, m]$ is just another way of denoting the atomic interval $m$. We shall extend our use of the predicate $Holds$ to apply to non-atomic (or *extended*) intervals as follows:

$$Holds(S, [m, n]) \leftrightarrow \forall k(m \leq k \leq n \rightarrow Holds(S, k)).$$

Thus state $S$ holds on the interval $[m, n]$ if and only if it holds on each of the atomic intervals of which $[m, n]$ is composed. A simple consequence of this definition is the rule **HOM** which will be introduced in Section 2.2.4.

### 2.2.2 Instantaneous transitions

Even a system as simple as this can furnish us with examples of a wide range of temporal phenomena. The simplest case is that of an **instantaneous transition**. This occurs when the state $S$ changes either from "on" to "off" or from "off' to "on". In the history

(**S1**) 0000000000 1111111111 0000000000

we see one example of each. If the series shown here begins at $t = 1$ then the transition from "off" to "on" occurs between times $t = 10$ and $t = 11$, and the transition from "on" to "off" occurs between times $t = 20$ and $t = 21$. This leads to a little difficulty in describing exactly *when* the transitions occur; if the times in the model are $t = 1, t = 2, t = 3, \ldots$, then neither transition occurs *at* any time, but rather each of them occurs *between* two times. This incidentally provides a compelling reason why we have to distinguish between two types of time, intervals and instants. The times $t = 1, t = 2, t = 3, \ldots$ are **atomic intervals**; we may assign to each of them a notional duration of 1 (indivisible) unit. They may be concatenated together to form longer (non-atomic) intervals, e.g., the interval [11,20] on which the state $S$ is "on" in example **S1**. An instantaneous transition such as the change from "off" to "on" does not occur on any interval, atomic or otherwise, although there are many non-atomic intervals *within* which it occurs: in our example the transition from "off" to "on" occurs within the interval [10,11] and also within any interval containing this, e.g., [8,12] or [7,20]. But the single unique "time" *at* which the transition occurs isn't an interval at all, but rather the point at which two intervals meet. This is an **instant**. We shall denote it 10][11.

We now have two different kinds of eventuality, sharply distinct from one another yet intimately related. On the one hand we have a **state**, our $S$, which on each atomic interval is either "on" or "off" (one might say "true" or "false" if $S$ were regarded as a statement). When $S$ is "on" we say that it **holds**. If $S$ (or $-S$) holds over a sequence of consecutive atomic intervals, we can say that it holds over the longer interval comprising those atomic intervals; in example **S1**, we have $Holds(S, [10, 15])$ and $Holds(-S, [3, 6])$, for example. On the other hand we have **instantaneous events**, the transitions from $-S$ to $S$ and vice versa. We shall denote the former event $Ingr(S)$, and the latter $Ingr(-S)$. Events do not hold, they **occur**. An instantaneous event occurs not on an interval but at the interface between two neighbouring intervals, an instant. We can write, for example **S1**, $Occurs(Ingr(S), 10][11)$ and $Occurs(Ingr(-S), 20][21)$.

### 2.2.3 Can a state hold at an instant?

Does it make sense, in this system, to speak of a state holding *at an instant*, as opposed to over an interval? If it does, it does so only derivatively: states primarily hold on intervals. But there is a sense in which we could say that if $S$ holds on interval $n$ and also on interval $n + 1$ then it holds at the instant $n][n + 1$ which marks their meeting point. If we say this then there will be some instants, such as 10][11 and 20][21 in example **S1**, at which neither $S$ nor $-S$ holds; these are precisely the instants at which there occur the transition events

associated with $S$. Note that there are also *intervals* over which, in another sense, neither $S$ nor $-S$ holds: for example, it is not the case either that $S$ holds over the interval [7,13] or that $-S$ holds over that interval. This is because $S$ holds over some parts of the interval, and $-S$ holds over other parts. The holding of a state like $S$, whether over intervals or at instants, completely reduces to the question of which atomic intervals it holds on. Clearly, in this discrete system we can very well manage without ascribing states to instants at all.

We can also, as a matter of fact, manage without ascribing instantaneous events to instants either, but only by establishing an essentially arbitrary convention. If $Ingr(S)$ occurs at the instant $n][n+1$, then we could decide that we shall eliminate reference to instants by saying that it occurs at the atomic interval $n$; or alternatively we could choose the opposite convention and say that $Ingr(S)$ occurs at $n+1$.* Although this is possible, it can lead to confusion (see the end of Section 2.2.4); it is certainly conceptually quite wrong, since it seems to ascribe the duration of an atomic interval to an event which is durationless.

### 2.2.4 A state holding for a while: the operator $Po$

We may wish to refer to an event which consists of $S$ starting to hold, holding for a while, and then ceasing to hold. This happens with $S$ over the interval [11,20] in example **S1**. We shall write $Po(S)$ to denote this event.† The event $Po(S)$ consists of the state $S$ holding for a while. The reader will naturally wonder in what way this is any different from the state $S$ itself. The answer is that for the event to occur it is essential that the interval over which $S$ holds is bounded by times at which $S$ does not hold. Thus in example **S1**, $Po(S)$ occurs *only* on the interval [11,20], whereas $S$ holds over any subinterval of this, e.g., [12,15], [11,16], [16,17], as well as atomic subintervals such as 13 and 14. Thus we can write

$$\begin{array}{ll} Holds(S,[11,20]) & Occurs(Po(S),[11,20]) \\ Holds(S,[12,15]) & \neg Occurs(Po(S),[12,15]) \\ Holds(S,12) & \neg Occurs(Po(S),12) \end{array}$$

We can formulate two general rules as follows:

(**HOM**) $$Holds(S,i) \wedge i' \sqsubset i \rightarrow Holds(S,i')$$

(**UNI**-Po) $$Occurs(Po(S),i) \wedge i' \sqsubset i \rightarrow \neg Occurs(Po(S),i')$$

The notation $i' \sqsubset i$ means that $i'$ is a *proper subinterval* of $i$, that is, the atomic subintervals which make up $i'$ form a proper subset of the set of atomic subintervals which make up $i$. The designation **HOM** is short for *homogeneous*, the point being that states occupy time in a uniform way so that any part of an interval over which a state holds is also an interval over which that state holds.‡ This contrasts with the designation **UNI**-Po, for *unitary*, the point here being that an occurrence of an event like $Po(S)$ is a single indivisible unit; no proper

*The former alternative recalls Von Wright's [von Wright, 1965] binary proposition-forming operator $T$ such that the proposition $pTq$ (read "$p$ and next $q$") is true at time $t$ if and only if $p$ is true at $t$ and $q$ is true at $t+1$. Using this operator, if $p$ represents the proposition that the state $S$ holds, then the proposition $(\neg p)Tp$ effectively records the occurrence of $Ingr(S)$.

†The designation $Po$ was introduced in [Galton, 1984].

‡The axiom here designated **HOM** was proposed by Hamblin [Hamblin, 1971], and independently by Allen [Allen, 1984]; it has been used by many authors since.

part of the time of such an occurrence is also the time of such an occurrence. More generally we shall say that an event-type $E$ is **unitary** if it satisfies the formula*

**(UNI)** $$Occurs(E, i) \wedge i' \sqsubset i \rightarrow \neg Occurs(E, i')$$

In the sequence

0000000000 1000000000

we see an occurrence of $Po(S)$ having the shortest possible duration. This is not an instantaneous event, since it does have duration, albeit minimal; a truly instantaneous event such as $Ingr(S)$ has no duration at all, occurring as it does at a durationless instant. We could describe the occurrence of $Po(S)$ here as *momentary*. We have, in this example,

$$Occurs(Ingr(S), 10][11),\ Occurs(Po(S), 11),\ Occurs(Ingr(-S), 11][12).$$

Here the event $Ingr(S)$ occurs immediately before the event $Po(S)$, marking its beginning, and the event $Ingr(-S)$ occurs immediately after $Po(S)$ marking its end. That is one reason why it would be confusing to adopt a convention whereby one or other of the events $Ingr(S)$ and $Ingr(-S)$ was said to occur on the atomic interval 11.

We can classify events as **instantaneous** or **durative** depending on whether they occur at instants or on intervals. Amongst durative events we can distinguish **momentary** and **extended** occurrences depending on whether they occupy atomic or non-atomic intervals.

### 2.2.5 Events and their occurrences

We have referred to both **events** and **occurrences**. We do not use these terms interchangeably. An event such as $Po(S)$ or $Ingr(S)$ is an **event type**: it can manifest itself as many different occurrences. For example, in the sequence

**(S2)** 0000001111 1110000011 1000000000 0001111111 0000000000

there are three occurrence of each of these event types. Occurrences are sometimes called **event tokens**. Events and their occurrences have very different ontological status. The event $Po(S)$ exists, in an abstract sense, so long as the state $S$ exists; whether or not any occurrences of this event exist depends on the particular incidence of state $S$ in the history we are considering. In the history

0000000000 0000000000 0000000000 0000000000 0000000000

there is no occurrence of $Po(S)$, although the event type still exists in the sense that although there are in fact no occurrences of it, there could have been some.

An event-type such as $Po(S)$ or $Ingr(S)$ is defined in terms of the state $S$ by giving a necessary and sufficient condition for there to be an occurrence of the event at a given time. We shall call this the **occurrence condition** for the event. The occurrence conditions for $Ingr(S)$ and $Po(S)$ are:

**Ingr** $Ingr(S)$ occurs at $n][n+1$ iff $-S$ holds on $n$ and $S$ holds on $n+1$.

**Po** $Po(S)$ occurs on $[m, n]$ iff $S$ holds throughout $[m, n]$ and $-S$ holds on both $m-1$ and $n+1$.

*Note that for Allen [Allen, 1984], and many subsequent authors, *all* event-types are unitary, **UNI** being postulated as an axiom.

### 2.2.6 A state holding for a certain duration

We can restrict $Po(S)$ to a certain duration as follows. For an integer $d$, let $For(S, d)$ be the event which consists of $S$ holding for exactly $d$ consecutive atomic intervals. Here $d$ is the *duration* of the event. For example, in the sequence **S2** there are two occurrences of $For(S, 7)$ (on the intervals [7,13] and [34,40]) and one occurrence of $For(S, 3)$ (on [19,21]). The occurrence condition for $For(S, d)$ is:

**For** $For(S, d)$ occurs on $[m, n]$ iff $n - m = d - 1$, $S$ holds throughout $[m, n]$ and $-S$ holds on both $m - 1$ and $n + 1$.

The occurrences of the event $For(S, 1)$ are just the *momentary* occurrences of $Po(S)$. In general, the occurrences of $For(S, k)$ form a subset of the occurrences of $Po(S)$. We shall say that $For(S, k)$ is a **subtype** of $Po(S)$, where the exact definition of subtype is as follows:

> Event type $E'$ is a subtype of event type $E$ if any occurrence of $E'$ is necessarily also an occurrence of type $E$.

The word "necessarily" in this definition means that we have to consider all possible sequences in determining whether one event is a sub-event of another.

### 2.2.7 Repetition events

For an event, $E$, let $Consec(E, k)$ be the event which consists of exactly $k$ consecutive non-overlapping occurrences of $E$. In example **S2** there is an occurrence of $Consec(Po(S), 3)$, and an occurrence of $Consec(For(S, 7), 2)$, both occurring on the interval [7,40], and two occurrences of $Consec(Po(S), 2)$, on the intervals [7,21] and [19,40]. There are also occurrences of $Consec(Ingr(S), 2)$ on [7,18] and [19,33].

The definition of consecutive events is complicated by the possibility of overlapping occurrences of an event type. Here we take the view that in order for two occurrences of $E$, say $e_1$ and $e_2$, to count as consecutive, there must be no occurrences of $E$ either beginning or ending in the interval between the end of $e_1$ and the beginning of $e_2$. Thus $e_2$ is the first occurrence of $E$ to begin after the end of $e_1$, and $e_1$ is the last occurrence of $E$ to end before the beginning of $e_2$. This requirement is formalised in the definitions given below.

Separate occurrence conditions for $Consec(E, k)$ have to be given depending on whether $E$ is instantaneous or durative:

**Consec-D** If $E$ is a durative event, then

- $Consec(E, 1)$ occurs on $[m, n]$ iff $E$ occurs on $[m, n]$;
- For $k > 1$, $Consec(E, k)$ occurs on $[m, n]$ iff there are integers $p, q$, where $m \leq p < q \leq n$, such that $E$ occurs on $[m, p]$ but not on any interval ending in $[p + 1, q - 1]$, and $Consec(E, k - 1)$ occurs on $[q, n]$ but not on any interval beginning in $[p + 1, q - 1]$.

**Consec-I** If $E$ is an instantaneous event, then

- $Consec(E, 2)$ occurs on $[m, n]$ iff $E$ occurs on both $m - 1][m$ and $n][n + 1$;

- For $k > 2$, $Consec(E, k)$ occurs on $[m, n]$ iff there is an integer $p$, where $m < p \leq n$, such that $E$ occurs at $m - 1][m$ but not at any instant between $m$ and $p$, and $Consec(E, k - 1)$ occurs on $[p, n]$.*

If $E$ is durative, $Consec(E, 1)$ is the same event as $E$ itself; for instantaneous $E$ the definition does not apply to the case $k = 1$, but we could if we wish supplement the definition by stipulating that $Consec(E, 1) = E$ in this case also.

Note the effect of iterating the operator $Consec$. In the sequence

(**S3**) 0001110001 1100011100 0111000111 0001110001 11000

the event $Consec(Po(S), 3)$ occurs on the intervals [4,18], [10,24], [16,30], [22,36], and [28,42], i.e., five times; but we do not have an occurrence of $Consec(Consec(Po(S), 3), 5)$ here since these occurrences are not all consecutive.

On the other hand, $Consec(Consec(Po(S), 3), 2)$ has two overlapping occurrences, on the intervals [4,36] and [10,42].

An alternative type of repetition event, which we shall denote $Times(E, k)$, *does* allow overlapping occurrences. The occurrence conditions are

**Times-D** If $E$ is durative, then

- $Times(E, 1)$ occurs on $[m, n]$ iff $E$ occurs on $[m, n]$;
- For $k > 1$, $Times(E, k)$ occurs on $[m, n]$ iff there are integers $p, q$, where $m \leq p \leq n$ and $m < q \leq n$, such that $E$ occurs on $[m, p]$ but not on any other subinterval of $[m, n]$ beginning before $q$, and $Times(E, k - 1)$ occurs on $[q, n]$.

**Times-I** If $E$ is instantaneous, then $Times(E, k)$ occurs on $[m, n]$ iff $Consec(E, k)$ does.

In example **S3**, both

$$Times(Consec(Po(S), 3), 5)$$

and

$$Times(Consec(Consec(Po(S), 3), 2), 2)$$

occur on [4,42]. For instantaneous events, $Times$ is the same as $Consec$. This is because these two operators only behave differently from one another when applied to event-types which can have overlapping occurrences. Here we are assuming that distinct occurrences of the same instantaneous event-type must occupy distinct instants, and hence cannot overlap.

The operator $Times$, as we have defined it, has one inevitable shortcoming. It does not cover the case where there are two distinct but strictly simultaneous occurrences of an event type. It is not possible to handle this case in the present framework: this can only be done by introducing a separate category of terms to refer to individual occurrences.

Although a number of authors have discussed repetitions of events, little systematic work has been done on this. Allen [Allen, 1984] suggests a definition of an event TWICE(E) which roughly corresponds to our Consec(E,2), although Allen's definition is faulty because the event thereby defined does not satisfy the principle **UNI** which he lays down as an axiom

*Note that it is not necessary to include a further condition that $Consec(E, k - 1)$ does not occur on any interval beginning in $[m + 1, p - 1]$ since this is implied by the non-occurrence of $E$ in this interval.

to be satisfied by all events. To explore this area further, it would be interesting to devise an algorithm for detecting occurrences of events of the form

$$Op_1(Op_2(Op_3(\ldots Op_n(Po(S), k_n), \ldots k_3), k_2), k_1)$$

where each $Op_i$ is either $Consec$ or $Times$.

A related issue is *periodic* events, where we are not concerned with some definite number of repetitions of a basic event, but rather with an ongoing state of affairs consisting of regularly or irregularly repeated occurrences of a given event type. We discuss this below in Section 2.2.11. For a fuller treatment of repetitive and periodic events, see other chapters.

### 2.2.8 Sequential composition of events

Related to these repetition events are *sequence* events. Given two event-types $E_1$ and $E_2$, an occurrence of their **sequential composition** consists of an occurrence of $E_1$ followed by an occurrence of $E_2$. We can distinguish two varieties of sequential composition, according to whether or not the second event is required to follow the first immediately. We shall use the notation $E_1; E_2$ for the **immediate sequential composition**, which requires the occurrence of $E_2$ to follow immediately on from the occurrence of $E_1$, and $E_1;; E_2$ for the **general sequential composition**, which does not require this (but allows it). The occurrence conditions are complicated by the fact that one or (in the case of general composition) both of the component events might be instantaneous.

There are many ways in which we might choose to define the composition operators. Possible occurrence conditions for immediate sequential composition (**ISC**) and general sequential composition (**GSC**), which might be modified to suit particular special purposes, are:

**ISC** If $E_1$ and $E_2$ are both durative, then $E_1; E_2$ occurs on the interval $[m, n]$ iff there is an integer $k$, where $m \leq k < n$, such that $E_1$ occurs on $[m, k]$ and $E_2$ occurs on $[k + 1, n]$.

If $E_1$ is durative and $E_2$ is instantaneous, then $E_1; E_2$ occurs on $[m, n]$ iff $E_1$ occurs on $[m, n]$ and $E_2$ occurs at $n][n + 1$.

If $E_1$ is instantaneous and $E_2$ is durative, then $E_1; E_2$ occurs on $[m, n]$ iff $E_1$ occurs at $m - 1][m$ and $E_2$ occurs on $[m, n]$.

**GSC** If $E_1$ and $E_2$ are both durative, then $E_1;; E_2$ occurs on the interval $[m, n]$ iff there are integers $k$ and $l$, where $m \leq k < l \leq n$, such that $E_1$ occurs on $[m, k]$ but not on any interval $[p, q]$ such that $k < q < l$, and $E_2$ occurs on $[l, n]$ but not on any interval $[r, s]$ such that $k < r < l$.

If $E_1$ is durative and $E_2$ is instantaneous, then $E_1;; E_2$ occurs on $[m, n]$ iff there is an integer $k$, where $m \leq k \leq n$, such that $E_1$ occurs on $[m, k]$ but not on any interval $[p, q]$ such that $k < q < n$, and $E_2$ occurs at $n][n + 1$ but at no instant $l][l + 1$ where $k \leq l < n$.

If $E_1$ is instantaneous and $E_2$ is durative, then $E_1;; E_2$ occurs on $[m, n]$ iff there is an integer $k$, where $m \leq k \leq n$, such that $E_1$ occurs at $m - 1][m$ but at no instant $l - 1][l$ where $m < l < k$ and $E_2$ occurs on $[k, n]$ but not on any interval $[p, q]$ where $m \leq p < k$.

If $E_1$ and $E_2$ are both instantaneous, then $E_1;;E_2$ occurs on $[m,n]$ iff $E_1$ occurs at $m-1][m$ and $E_2$ occurs at $n][n+1$, and neither event occurs on any instant $l-1][l$ where $m < l < n+1$.

In example **S2** there is an occurrence of the event-type $For(S,7);;For(S,3)$ on [7,21], and hence an occurrence of $(For(S,7);;For(S,3));;For(S,7)$ on [7,40]. From the definitions one can prove that the event-types $(E_1;;E_2);;E_3$ and $E_1;;(E_2;;E_3)$ are always identical in the sense of having the same occurrences as each other in every possible history; hence we may drop the brackets and write $E_1;;E_2;;E_3$. The same goes for the operator ";".

For a durative event $E$ we can define a *repetition* operator *Rep* such that an occurrence of $Rep(E,n)$ comprises $n$ consecutive occurrences of $E$ in immediate succession. It can be defined recursively in terms of ";" as follows:

**Rep** $Rep(E,1) = E$
$Rep(E,n+1) = Rep(E,n);E$

$Rep(E,n)$ is a subtype of $Consec(E,n)$, covering just those occurrences of the latter in which the component occurrences of $E$ follow on from one another without delay. Similarly, we can define $Consec$ itself in terms of ";;" as follows:

$$\begin{aligned} Consec(E,1) &= E \\ Consec(E,n+1) &= E;;Consec(E,n) \end{aligned}$$

and this definition can be shown to be equivalent to the one given earlier.

### 2.2.9 An event in progress: the operator $Prog$

In addition to defining classes of events by laying down their occurrence conditions, our present framework allows us to define new classes of *states* by laying down appropriate **holding conditions**. An example of this is the *progressive* state which obtains at those times when some event is in the process of occurring, i.e., when it is in progress. In the sequence

0000000000 1111000111 0000000000

the event $Consec(Po(S),2)$ occurs on the interval [11,20], so at any time during this interval that event is in progress. We shall say that the state $Prog(Consec(Po(S),2))$ holds throughout this interval.

The holding condition for states of the form $Prog(E)$ is

**Prog** For a durative event $E$, the state $Prog(E)$ holds on the atomic interval $k$ iff $E$ occurs on an interval $[m,n]$ such that $m \leq k \leq n$.

The operator $Prog$, which maps events onto states, is closely related to the operator $Po$ which maps states onto events; they stand to one another approximately as inverses.

Thus, given a state $S$, the state $Prog(Po(S))$ holds at a time $k$ iff $Po(S)$ occurs on an interval $[m,n]$ such that $m \leq k \leq n$. But if $Po(S)$ occurs on $[m,n]$ and $m \leq k \leq n$, then $S$ must hold at $k$. Thus $Prog(Po(S))$ entails $S$ (meaning that the latter must hold whenever the former holds). The converse entailment does not hold however, since e.g., in the sequence

0000000000 1111111111 1111111111 $\cdots$

where $S$ holds at *all* times from $t = 11$ onwards, $Po(S)$ does not occur at all, and hence $Prog(Po(S))$ does not hold.

The two states $S$ and $Prog(Po(S))$, although intimately related, are nonetheless distinct, and in fact represent two different levels of description of what is going on. The state $S$ is *basic* in the sense that the holding or not holding of $S$ over an atomic interval is simply a given fact, specifiable independently of its holding or not holding at any other time. We could say that $S$ is a description at Level 0. The event $Po(S)$ is defined in terms of $S$, so its occurrence or non-occurrence at different times is dependent on the pattern of holding and not holding of $S$. It is one level further up on the dependence hierarchy, Level 1. The state $Prog(Po(S))$ is at a higher level still, since its holding depends on the occurrence of $Po(S)$ which in turn depends on the pattern of holding and not holding of $S$. Thus to say that $Prog(Po(S))$ holds at time $k$ has implications not just for the state of the world at $t = k$ but also for the state of the world at earlier and later times. In fact we can easily prove, from the conditions **Po** and **Prog**, that

$$Holds(Prog(Po(S)), k) \equiv \\ Holds(S, k) \wedge \exists m, n(m < k < n \wedge Holds(-S, m) \wedge Holds(-S, n)).$$

The other respect in which $Po$ and $Prog$ are approximately inverse to one another concerns the relationship between the events $E$ and $Po(Prog(E))$. (Here $E$ has to be durative.) If $Po(Prog(E))$ occurs on $[m, n]$ then $Prog(E)$ holds throughout $[m, n]$ but does not hold at $m - 1$ or $n + 1$. This means that if $m \leq k \leq n$ then $E$ occurs on some interval $[m_k, n_k]$, where $m \leq m_k \leq k \leq n_k \leq n$. It may be that all the $k$s are associated with the same occurrence of $E$; if so, that occurrence must occupy exactly the interval $[m, n]$ and therefore corresponds exactly to the occurrence of $Po(Prog(E))$. This will always be the case if $E = Po(S)$, so we can identify $Po(S)$ and $Po(Prog(Po(S)))$.

More generally, though, there may be overlapping or immediately consecutive occurrences of $E$ spanning the interval $[m, n]$. An example of overlapping occurrences is furnished by the sequence

0000000111 0001110001 1100000000.

Here there are two occurrences of $Consec(Po(S), 2)$, on the intervals [8,16] and [14,22]. This means that $Prog(Consec(Po(S), 2))$ holds throughout [8,22], and since it does not hold at 7 or 23, the event $Po(Prog(Consec(Po(S), 2)))$ occurs on [8,22].

The same sequence gives us an example of immediately consecutive occurrences. The event $For(S, 3); For(-S, 3)$ occurs on [8,13] and [14,19]. Hence we can see that this event holds over [8,19]. On the other hand it does not hold at 7 or 20,* and hence

$$Po(Prog(For(S, 3); For(-S, 3)))$$

occurs on [8,19]. These examples show that in general $Po(Prog(E))$ is not the same as $E$: it is only for events $E$ of a type that does not admit overlapping or immediately consecutive occurrences that this identity holds. Examples of such events are $Po(S)$ and $For(S, n)$.

*$For(S, 3); For(-S, 3)$ does not occur on [20,25], since $For(-S, 3)$ doesn't hold on [23,25]: in order for this to be the case we would need $-(-S))$, i.e., $S$, to hold at 26.

### 2.2.10 Completed events: the operator $Perf$

$Prog$ gives us one way of deriving states from events, enabling us to describe the state of the world at a time in terms of the events that are in progress then. Another way is to describe the world in terms of events that are completed. For this we use the operator $Perf$ which has the following holding condition:

**Perf** For a durative event $E$, the state $Perf(E)$ holds at $k$ iff $E$ occurs on some interval $[m, n]$ such that $n < k$.

For an instantaneous event $E$, the state $Perf(E)$ holds at $k$ iff $E$ occurs at some instant $n][n + 1$ such that $n < k$.

The operator $Perf$ roughly corresponds to the perfect tense in English, so that, for example, if $E$ is the event "John flies across the Atlantic", $Perf(E)$ is the state "John has flown across the Atlantic".

The relationship between $Po$ and $Prog$ is roughly paralleled by that between $Ingr$ and $Perf$. The facts, easily verified, are as follows:

1. If $E$ is an instantaneous event, then the first occurrence of this event is also the only occurrence of $Ingr(Perf(E))$.
2. If $E$ is a durative event, then the event $Ingr(Perf(E))$ marks the completion of the first occurrence of $E$.
3. If state $S$ holds, but has not always held, then so does the state $Perf(Ingr(S))$.

One can also define a temporal mirror-image of $Perf$, which we denote $Pros$, with holding condition

**Pros** For a durative event $E$, the state $Pros(E)$ holds at $k$ iff $E$ occurs on some interval $[m, n]$ such that $k < m$.

For an instantaneous event $E$, the state $Pros(E)$ holds at $k$ iff $E$ occurs at some instant $n - 1][n$ such that $k < n$.

### 2.2.11 Frequency of occurrence

Our final state-forming operator is the *Frequentative* operator $Freq$. Given an event $E$, the state $Freq(E)$ holds if $E$ is occurring repeatedly. In English this is the natural interpretation of the Continuous Tense when used with a verb denoting an event with no or very short duration, as in "John is knocking at the door", which says there is a sequence of knocks rather than a single knock in progress. It is impossible to give a precise formal definition of this notion, but the following definition, for all its crudeness, enables us to express roughly what we want. We shall use $Freq(E, p/q)$ to mean that $E$ is occurring with a frequency of at least $p$ occurrences in $q$ atomic intervals:

**Freq** For an event $E$, the state $Freq(E, p/q)$ holds at $k$ iff $k$ is in an interval $[m, n]$ on which occurs $Times(E, r)$ for some $r$ such that $r/(n - m + 1) \geq p/q$.

This construction is related to, but not the same, as the notion of a *periodic* event type, which occurs at regular intervals, or else in correlation with other specified event types. Such events have been the focus of considerable research in the temporal representation and reasoning community, e.g., by [Terenziani, 1996]. For further discussion, see Chapter 5.

### 2.2.12 Processes and activities

Progressives and frequentatives are examples of *processes*. Here we treat a process as a kind of state, but one which has a texture on larger timescales than the atomic. A very simple process is illustrated by the sequence

0101010101 0101010101 0101010101 0101010101

which simply consists of our basic state alternating between "on" and "off" on consecutive atomic intervals. We can treat this as a state at a higher level of description. Suppose we label it as $Alt(S)$. Then $Alt(S)$ holds throughout the above sequence, and hence on every atomic interval making up the sequence. Its holding on any one such interval is not determined by what is the case at that interval considered in isolation, but by what is the case over a larger context in which that interval occurs. This can lend a somewhat indeterminate character to the process, which becomes particularly apparent if we consider a situation in which the process starts or stops, e.g., the sequence

0000000000 1010101010 1010101010 1111111111

Here $Alt(S)$ clearly holds over the interval [11,30], and clearly does not hold at any time during either [1,9] and [32,40]. But does it hold on the atomic intervals 10 and 31? There is no principled way of choosing whether to regard the holding of $-S$ at 10 as a continuation of the unbroken holding of $-S$ over [1,9], or as the first state in the process $Alt(S)$ which continues over [11,30]; and likewise at the other end.

The process $Alt(S)$ can be regarded as *uniform* in that it looks the same throughout the time that it holds. Some processes are *progressive* in that they involve a systematic change during the time that they hold. An example would be a processes by which state $S$ comes to hold for an ever greater proportion of time. This is illustrated in the sequence

$\cdots$ 0000000010 0000010000 0100001000 1001011011 1011110111 1101111111 $\cdots$

in which determination of the onset and termination of the process is even more problematic.

## 2.3 Systems with finitely-many states in discrete time

In the previous section we saw that even with a single binary state in discrete time an astonishing range of different kinds of eventuality can be described. There are several ways in which we could make the system we are studying more complicated. Two obvious choices are (i) to consider a state having more than two values, and (ii) to consider more than one state. These are in fact equivalent, as we shall show.

### 2.3.1 One many-valued state *vs* many binary states

The first option is to generalise from a state which can assume only the two values "on" and "off" to a state which can assume a range of values, which for the present we shall assume to be finite. A state of this kind is known, following McCarthy and Hayes [McCarthy and Hayes, 1969], as a **fluent**. Indeed, an ordinary two-valued state can be regarded as the limiting case of a fluent in which the range of values is reduced to two—a so-called **boolean fluent**. A general finite-valued fluent $f$ can take values from some pre-assigned set $V_f =$

$\{f_1, f_2, \ldots, f_n\}$. To say that at time $k$ the fluent $f$ takes value $f_i$ we write $Holds(f = f_i, k)$. A history can then be portrayed as a sequence of values from $V_f$, e.g.,

$$\cdots f_3 f_3 f_3 f_4 f_5 f_6 f_6 f_6 f_6 f_2 f_3 f_3 f_1 f_1 f_1 f_1 \cdots$$

The second option is to consider a finite set of binary states $\mathbf{S} = \{S_1, S_2, \ldots, S_n\}$. Each of these states behaves like the single state $S$ considered in the Section 2.2. A history must be presented as a correlated set of sequences, one for each state in $\mathbf{S}$:

| | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $S_1$ : | $\cdots$ | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | $\cdots$ |
| $S_2$ : | $\cdots$ | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | $\cdots$ |
| $S_3$ : | $\cdots$ | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | $\cdots$ |

The states are not necessarily independent of one another. In the example above, $S_1$ and $S_3$ are never "off" together. This might be a coincidence, or it might be that we are using $S_1$ and $S_3$ to model states in the world which necessarily have this property.

We can replace the three states $S_1$, $S_2$, and $S_3$ by a single fluent $f$ which takes eight values according to the rule

$$f = v(S_1) + 2v(S_2) + 4v(S_3),$$

where $v(S_i)$ is 1 or 0 according as $S_i$ is "on" or "off". The triple sequence shown above can then be represented by a single sequence for $f$:

$$\cdots 44466677366664451115 \cdots$$

Since $S_1$ and $S_3$ are never off together, the values $f = 0$ and $f = 2$ do not occur. We can build in this dependency between $S_1$ and $S_2$ by making $f$ six-valued rather than eight-valued.

This shows us how we can take an arbitrary finite-valued fluent $f$ and replace it by means of a finite set of binary states. Suppose we have a fluent $f$ which takes values from the set $\{a, b, c, d, e\}$. Since $2^2 < 5 \leq 2^3$, we shall need three binary states to model this. But three independent states give rise to eight values; to reduce these to five we must introduce appropriate dependencies. There are many ways of doing this. A simple solution is to have states $S_1, S_2, S_3$ such that whenever $S_1$ is "on", $S_2$ and $S_3$ must both be "off". Then a suitable mapping between values of $f$ and values of the $S_i$ is:

| $f$ | $S_1$ | $S_2$ | $S_3$ |
|---|---|---|---|
| $a$ | off | off | off |
| $b$ | off | off | on |
| $c$ | off | on | off |
| $d$ | off | on | on |
| $e$ | on | off | off |

On this scheme we can explain the "meaning" of the three states as follows:

$S_1$ is the state $f = e$.
$S_2$ is the state $(f = c) \vee (f = d)$.
$S_3$ is the state $(f = b) \vee (f = d)$.

Of course, a simpler, but much less economical, way of representing the fluent $f$ by means of a set of binary states is to have one such state for each possible value of $f$, thus $f = a$, $f = b, \ldots$, and $f = e$, and constrain these states to be pairwise incompatible.

We have shown that the two suggested extensions to a one-state system, namely a many-state system and a system with a many-valued fluent, are interconvertible, and therefore in a formal sense equivalent. This means that we can choose whichever system most suits us for any particular purpose and be assured that any results we derive are transferable, *mutandis mutatis*, to the other system. Not only that, but we can also, if we wish, use a "mixed" system in which there are several many-valued fluents instead of, or in addition to, a number of binary states.

All the phenomena we saw in the previous section in connection with a single binary state also exist in the more complicated systems we are now considering. But in addition, there are further interesting phenomena which are worth exploring here.

### 2.3.2 State-spaces, adjacency, and quasi-continuity

The set $V_f$ of values that can be assumed by a fluent $f$ can be thought of as a kind of "space", and change in the value of $f$ as a kind of "motion" through this space. One interesting possibility is that such "motion" resembles ordinary motion in being continuous. Of course, in a discrete space, continuity as we ordinarily understand it is not possible; but if we endow the space with an *adjacency* relation defining the "next-door neighbours" of each of the values of $f$ making up the space, then we can describe a motion through the space as "quasi-continuous" so long as any instantaneous change in the value of $f$ is between next-door neighbours.

As an example, suppose we have a 9-valued fluent, taking integer values in the range 0, ..., 8. Many different adjacency relations can be defined on this space, of which three are illustrated in Figure 2.1.

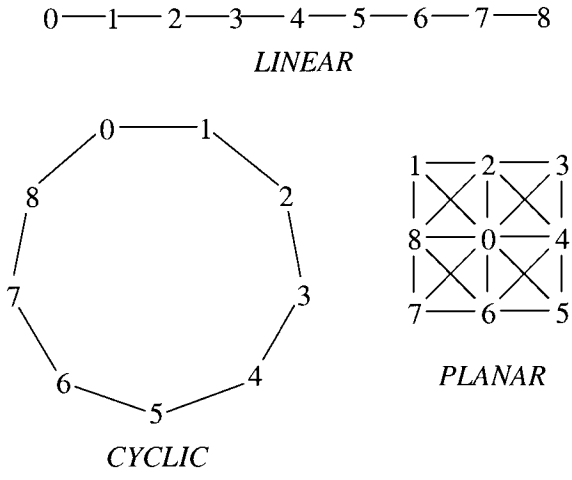



Figure 2.1: *Three adjacency relations on* {0,1,2,3,4,5,6,7,8}.

If change of value of $f$ is constrained to be quasi-continuous in the sense defined above, then each of the three adjacency relations portrayed here defines certain sequences as possible

and others as impossible. For example, the sequence

(**S4**) $\cdots 88888880000000 \cdots$

is possible in the cyclic and planar spaces, but not in the linear one, whereas the sequence

$\cdots 22222224444444 \cdots$

is only possible in the planar space. The sequence

(**S5**) $\cdots 2222222255555555 \cdots$

is ruled out in all three spaces.

### 2.3.3 Instantaneous and Durative Transitions

Example **S4** above illustrates a phenomenon we have already seen in connection with a single state: an instantaneous transition. Now instead of being a transition from "on" to "off", as in the previous section, it is a transition from one value of the fluent, $f = 8$, to another value, $f = 0$. We therefore cannot represent it by means of the operator $Ingr$. Instead we shall introduce a new operator for this.

Example **S5**, if it could occur, would also represent an instantaneous transition. However, in all three of the value-spaces illustrated, it is only possible to get from $f = 2$ to $f = 5$ by passing through some intermediate values. In the linear model, we might have

$\cdots 222222233344455555555 \cdots$

and this sequence is also possible in the other two models. The other models have other possibilities as well, e.g.,

$\cdots 222222222108765555555 \cdots$

which is possible in both the cyclic and planar models, and

$\cdots 2222222228886665555555 \cdots$

which is only possible in the planar model. In all these cases, what we have is a **durative transition** between the states $f = 2$ and $f = 5$.

We shall use the same operator $Trans$ to construct both instantaneous and durative transitions, but give separate occurrence conditions for the two cases, as follows:

**Trans** If $S_1$ and $S_2$ are two mutually incompatible states, then the event $Trans(S_1, S_2)$ has

- an instantaneous occurrence at $m][n$ iff $S_1$ holds at $m$ and $S_2$ holds at $n$, and
- a durative occurrence on $[m, n]$ iff $S_1$ holds at $m - 1$ and $S_2$ holds at $n + 1$ and both $-S_1$ and $-S_2$ hold on $[m, n]$.

The reason for the third conjunct in the condition for a durative occurrence is that it ensures that $Trans(S_1, S_2)$ satisfies the condition **UNI**.

### 2.3.4 Formal and material progressive operators

In everyday language, we might describe the event $Trans(f = f_1, f = f_2)$ in the terms "The value of $f$ changes from $f_1$ to $f_2$", and many kinds of change we observe in the world can be described in this way. Now suppose we want to say instead that "The value of $f$ is changing from $f_1$ to $f_2$". This says that a certain state obtains that can be characterised in terms of an event which is in progress. The obvious way to represent this is using the progressive operator to form the state

$$Prog(Trans(f = f_1, f = f_2)),$$

but this is in some respects problematic. The holding conditions of this complex state can be derived from **Prog** and **Trans** as follows:

$Prog(Trans(f = f_1, f = f_2))$ holds at $m$ iff there are times $l$ and $n$ such that $l < m < n$ and

- $f = f_1$ holds at $l$,
- $f = f_2$ holds at $n$,
- neither $f = f_1$ nor $f = f_2$ holds on any subinterval of $[l + 1, n - 1]$.

In the planar model illustrated in Figure 2.1, consider the sequence

$$\cdots 1112186600\ 0111000445\ 433 \cdots$$

Note that $f = 2$ holds at time $t = 4$, that $f = 5$ holds at $t = 20$, and neither of these states holds at any time during the interval [5,19]. It follows that $Trans(f = 2, f = 5)$ occurs on this interval, and therefore that $Prog(Trans(f = 2, f = 5))$ holds on all of its subintervals. At $t = 10$, $f$ has the value 0. If at this time we ask what is happening, is it a satisfactory answer to say that the value of $f$ is changing from 2 to 5? There are two possible objections to this: first, that at $t = 10$, the value of $f$ is not actually changing at all, since it remains 0 over the interval [9,11]; and second, even if one grants that, on a longer perspective, the value is indeed changing, the immediate change is between 6 and 1 (there being an occurrence of $Trans(f = 6, f = 1)$ on the interval [9,11]), i.e., a movement further from 5 and nearer to 2, so it would be perverse to describe this as part of a transition from 2 to 5.

The problem here arises from trying to use abstract models to explain the meanings of linguistic phenomena which are normally only encountered in concrete contexts. In some concrete exemplifications of the abstract pattern presented above, it would indeed be appropriate to say that the change here represented by $Trans(f = 2, f = 5)$ is in progress at $t = 10$, whereas in others it would not. As an example of the first kind, consider the following. On a certain day, I drive from Bristol to Northampton. I stop for lunch in Swindon and drive on. When I get to Oxford, I realise that I have left my wallet in Swindon. I rush back to where I had lunch and to my relief find that the wallet is safe. I then drive on through Oxford to Northampton. The journey is portrayed in Figure 2.2. Suppose that while I am driving back between Oxford and Swindon someone asks me what I am doing. I reply "I'm driving from Bristol to Northampton", and this is surely a perfectly correct and reasonable reply, even though I am just then travelling in exactly the opposite direction to the way I should be going in order to drive from Bristol to Northampton. It is very much a matter of the perspective one adopts. I am indeed simultaneously driving from Bristol to Northampton

and from Oxford to Swindon; the former description is appropriate when taking a broader perspective, involving not just a longer time-scale but also more ulterior purposes than the narrower perspective within which the latter description is more appropriate.

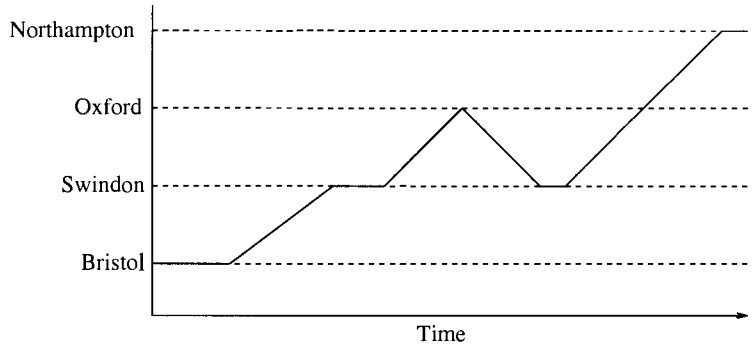



Figure 2.2: *A journey from Bristol to Northampton.*

Now consider a more extreme case. Suppose someone who lives in Bristol moves house to Oxford, and then several years later moves again to Northampton. Suppose further that they never revisit Bristol after having left it, and never visit Northampton before moving house there. Both journeys—from Bristol to Oxford, and from Oxford to Northampton, are undertaken by car. It would surely be stretching things to say, during this person's years-long sojourn in Oxford, that they were in the process of driving from Bristol to Northampton. The only circumstance that could render this in the least plausible is that the person set off from Bristol with the intention of eventually settling in Northampton, always regarding the stay in Oxford as only temporary.

We can define two extremes. On the one hand there is the **formal progressive** represented by our operator $Prog$. The state $Prog(E)$ is true at any time between the start of an occurrence of the event $E$ and its end. It takes no account of whether or not the states or activities obtaining at these times contribute in any way to the final completion of $E$. This represents the broadest perspective, a perspective that is able to overlook all temporary deviations from, or interruptions to, the progress of the event. At the other extreme is what might be called the **material progressive**, which does take into account only those states or activities which make a material contribution to the progress of the event $E$. In our example, "I am driving from Bristol to Northampton", when interpreted in this narrow, material sense, is only true at those times when I am actually driving, and when the driving I am doing is indeed taking me forward along the route from Bristol to Northampton—in other words to the times in Figure 2.2 where the graph of my journey has positive gradient, thus excluding both the time I spend in Swindon having lunch and the time I spend driving back to retrieve my wallet. We frequently use both the material and the formal senses of the progressive.* The

*Nor should one forget that we frequently—perhaps most frequently of all—use the progressive in a "modalised" sense that does not commit us to the eventual completion of the event in question. This leads to the so-called "imperfective paradox" [Dowty, 1979] by which, while on the one hand it cannot be true that I have been driving without it also being true that I have driven, on the other hand it *can* be true that I have been driving from Bristol to Northampton, without it ever being true that I have driven from Bristol to Northampton.

difficulty with the material sense is that it seems to be impossible to give a formal holding condition for it.

### 2.3.5 Formal and material perfect-tense operators

A similar difficulty affects the operator $Perf$, which corresponds to some, but not all, uses of the English perfect tense. At 1.30 p.m. on Monday I truthfully say "I have had lunch". At the same time on Tuesday I truthfully say "I have not had lunch yet". We cannot represent these two statements as $Perf(E)$ and $-Perf(E)$for the same event $E$, since the holding condition for $Perf(E)$ implies that if it holds at a certain time then it must also hold at *any* later time: in short, $Perf(E)$ is *irrevocable*. There are two ways we might avoid this difficulty.

One way is to claim that when I say "I have had lunch" on Monday, what I mean is "I have had Monday's lunch", and this remains true ever after, whereas when I say "I have not had lunch" on Tuesday, what I mean is "I have not had Tuesday's lunch". Formally, we would have to represent these two statements as $Perf(E_1)$ and $-Perf(E_2)$, using different event-types to avoid the incompatibility.

The alternative solution is to say that as with the progressive, we use the perfect in a range of different senses, with at one extreme the **formal perfect** represented by $Perf$, with the property of irrevocability built into that operator, and at the other extreme a **material perfective** which takes account of the material state resulting from the occurrence of an event and only continues to affirm that the event has happened so long as that state persists. In linguistics these two senses of the perfect are recognised as just two amongst a larger number, commonly given as four [Comrie, 1976]. As with the material sense of the progressive, it is impossible to give formal holding conditions for the material perfect.

### 2.3.6 Logical operations on states

We now turn to consider various logical operations on states, fluents and events. Regarding states, we have already met $-S$, the negation of state $S$, which holds on just those atomic intervals on which $S$ does not hold. We have just given, in effect, the holding condition for $-S$, which we here state formally as

**State-Neg** The state $-S$ holds on the atomic interval $n$ iff $S$ does not hold on $n$.

From this condition we can readily derive the condition for $-S$ to hold over an arbitrary interval, as follows:

> The state $-S$ holds on the interval $[m, n]$ iff there is no atomic interval $k$, where $m \leq k \leq n$, on which $S$ holds.

Note in particular that we *cannot* say, for an arbitrary interval, that $-S$ holds over it iff $S$ does not hold over it, the reason being, of course, that $S$ may hold on some of its subintervals, and $-S$ on others.

In Section 2.3.1 we identified a state $S_3$ with a disjunction $(f = b) \vee (f = d)$ of fluent-values. We must be a little more precise about this. Given a fluent $f$ and a possible value for it, say $b$, the expression $f = b$ denotes that state which is "on" on precisely those atomic intervals when $f$ has the value $b$, and "off" on all other atomic intervals. Thus $(f = b) \vee (f = d)$ is a disjunction of states, and we need to define its occurrence condition. In general, for states $S_1$ and $S_2$, we can define their disjunction $S_1 \vee S_2$ by the rule

**State-disj** The state $S_1 \vee S_2$ holds on the atomic interval $n$ iff either $S_1$ or $S_2$ (or both) holds on $n$.

As with negation, we *cannot* say that $S_1 \vee S_2$ holds over an arbitrary interval if and only if either $S_1$ holds over it or $S_2$ holds over it (the "if" part is correct, but not the "only if" part).

State-disjunction enables us to bundle a set of states together in order to refer to states of affairs in more general terms. For example, in the planar adjacency model of Figure 2.1, we could introduce some bundled states as follows:

$$\begin{aligned} Top &= f = 1 \vee f = 2 \vee f = 3 \\ Right &= f = 3 \vee f = 4 \vee f = 5 \\ Bottom &= f = 5 \vee f = 6 \vee f = 7 \\ Left &= f = 7 \vee f = 8 \vee f = 1 \end{aligned}$$

We can then use these states as arguments for our operators, for example $Trans(Left, Right)$. This is a genuinely higher-level event-description than $Trans(f = 1, f = 3)$ and the like: while it is true that any occurrence of $Trans(Left, Right)$ must also be an occurrence of some more primitive event of the form $Trans(f = a, f = b)$, where $f = a$ is one of the disjuncts defining $Left$ and $f = b$ is one of the disjuncts defining $Right$, the converse does not hold. For example, in the sequence

1111111880 0443333333

there is an occurrence of $Trans(f = 1, f = 3)$ on the interval [8,13], but there is not an occurrence of $Trans(Left, Right)$ on that interval, even though $f = 1$ implies $Left$ and $f = 3$ implies $Right$. There *is* an occurrence of $Trans(Left, Right)$ in this sequence, but it occurs on the interval [10,11], being also an occurrence of $Trans(f = 8, f = 4)$.

In general we can say that for an interval $i$, $Occurs(Trans(S_1 \vee S_2, S_3 \vee S_4), i)$ implies

$$\begin{aligned} &Occurs(Trans(S_1, S_3), i) \vee Occurs(Trans(S_1, S_4), i) \vee \\ &\quad Occurs(Trans(S_2, S_3), i) \vee Occurs(Trans(S_2, S_4), i), \end{aligned}$$

but the converse implication does not hold. With other operators, we may not even have such a simple implication as that. For example, in the sequence

0000000000 1111188888 0000000000

there is an occurrence of $Po(Left)$ on [11,20], but no occurrence of the form $Po(f = x)$ on that interval; while on the other hand there is an occurrence of $Po(f = 1)$ on [11,15], and an occurrence of $Po(f = 8)$ on [16,20], but $Po(Left)$ does not occur on either of these intervals.

Turning now to *conjunction*, we define the state $S_1 \wedge S_2$ by the occurrence condition

**State-conj** The state $S_1 \wedge S_2$ holds over the atomic interval $n$ if both $S_1$ and $S_2$ hold over $n$.

This rule generalises straightforwardly to arbitrary intervals:

The state $S_1 \wedge S_2$ holds over the interval $[m, n]$ iff $S_1$ and $S_2$ both hold over $[m, n]$.

As with state-disjunction, the behaviour of state-conjunction in interaction with operators like $Trans$ and $Po$ is complex. We shall not labour the details here; the reader should have no difficulty in constructing appropriate examples to illustrate the various relationships.

### 2.3.7 Logical operations on events

Given two events $E_1$ and $E_2$, there seem to be two ways to form a third event that might reasonably be called their conjunction. These are perhaps best illustrated by means of concrete examples:

- Suppose that Mary and John both take the cross-channel ferry from Dover to Calais. There is an occurrence of the event "Mary travels from Dover to Calais" and also an occurrence of the event "John travels from Dover to Calais", and the two occurrences occupy exactly the same interval of time. We can say that over that interval Mary travelled from Dover to Calais *and* John travelled from Dover to Calais. This gives us one kind of conjunction of two events.

- Consider an occurrence of the event "Mary travels from Dover to Calais". The very same occurrence is also an occurrence of the events "Mary travels from England to France" and of "Mary has a boat trip". Thus we can say that Mary travelled from England to France *and* Mary had a boat trip. If we think of all the possible occurrences of the former event-type as forming one set, and all the possible occurrences of the latter as forming another, then the event we are talking about is in the intersection of the two sets, and it seems natural to construct an event-type whose possible occurrence are precisely all the elements of that intersection. This event would be another kind of conjunction of two events, and would be a subtype of both of them (in fact their maximal common subtype).

At first sight these two kinds of conjunction seem to be sharply distinct, but it proves to be quite hard to characterise the distinction unambiguously. Let us use the notations $E_1 \wedge E_2$ and $E_1 \cap E_2$ for the two kinds of conjunction. What we should like to say is something like the following:

- The event $E_1 \wedge E_2$ occurs on the interval $[m, n]$ iff both the events $E_1$ and $E_2$ occur on $[m, n]$.

- $X$ is an occurrence of $E_1 \cap E_2$ iff $X$ is an occurrence both of $E_1$ and of $E_2$.

Unfortunately, these two definitions provide us with no ground for distinguishing between $E_1 \wedge E_2$ and $E_1 \cap E_2$. This is because the first definition characterises an event-type, as usual, in terms of its occurrence condition, i.e., the condition for there to be an occurrence of the event at a given time, whereas the second definition refers instead to the occurrences of an event as individuals with implicit criteria of identity. We need to know what it means to say that an occurrence described in one way is the very same occurrence as an occurrence described in another way. Intuitively it seems that an occurrence of "Mary travels from Dover to Calais" cannot be identical to an occurrence of "John travels from Dover to Calais", since one of them involves Mary and the other involves John. Implicitly, we are appealing to some criterion of identity for occurrences which entails that occurrences involving different participants cannot be identical. An obvious criterion to use would be that occurrences are identical if they involve the same participants undergoing the same changes. Davidson [Davidson, 1969] has a clever example to show that this can lead to counterintuitive results. Consider a metal ball which rotates through 35 degrees while getting warmer. Is the rotation of the ball the same occurrence as the warming of the ball? Davidson points out that in both cases we are referring to the same movements of the same molecules; it is impossible, even

in principle, to separate out movements which contribute to the rotation from movements which contribute to the warming. By the suggested criterion of identity, they are the same occurrence, whereas intuitively we might have good grounds for regarding them as different.

We here touch on an issue that has been the subject of much philosophical discussion: the characterisation of events generally, of which the matter of event identity is one important aspect. To pursue this further here would take us too far from our main theme; a useful reference is [Casati and Varzi, 1996].

## 2.4 Finite-state systems in continuous time

Up to now we have assumed that the flow of time can be represented by means of a discrete series of indivisible intervals. For many purposes this is a satisfactory model of time, and as we have seen, it certainly allows us to define many temporal phenomena of interest. In many contexts, however, it is more usual to model the flow of time as a continuum, ordered like the real numbers rather than the integers.* "Ordering time like the real numbers" represents a radical departure from the discrete model we have been assuming up to now, since whereas in the discrete model we have atomic intervals to provide the elementary building blocks out of which all other intervals are built, and in terms of which instants (the meeting-points of adjacent atomic intervals) can be defined, in the continuous model there are no such building blocks, since every interval can be endlessly subdivided into smaller ones. In the discrete model, all temporal phenomena can be defined in terms of the holding of states on atomic intervals; in the continuous model there is no similar set of intervals such that all temporal phenomena can be defined in terms of the holding of states on intervals of the set. There are two approaches we might take here: either to take the elementary facts to be the holdings of states over arbitrary intervals (with appropriate dependencies amongst such facts), or to shift the burden of supporting the elementary facts from intervals onto instants. We shall examine the former approach in this section, and the latter in the next.

Since we no longer have atomic intervals, we cannot use designations like $[m, n]$, where $m$ and $n$ are integers, to denote arbitrary intervals in this system. Nor, since at this stage we do not wish to rely on instants to provide the conceptual underpinning of our temporal model, will we use designations like $(x, y)$, where $x$ and $y$ are real numbers. Instead, we shall simply use notations like $i$ and $j$ to represent intervals, these forming a set on which the standard Interval Calculus relations ('meets', 'overlaps', etc; see previous chapter, Fig. 1.3) are defined, and we shall continue to use the predicate $Holds$ for assigning states to *these* intervals, e.g., $Holds(S, i)$. Lacking atomic intervals, we are no longer in a position to derive the rule **HOM** as a simple consequence of any more basic principles. Instead we have to postulate it as an axiom of our system. As before, we shall find no reason to say that a state holds on an instant, although we can still define instants as the meeting points of intervals; that is, whenever interval $i$ meets interval $j$ (i.e., $i\texttt{Meets}j$), we can define an instant $i][j$ at which they meet, the criterion of identity for instants being given by the rule

$$i][j = i'][j' \text{ iff } i \text{ meets } j'.$$

*There is also, of course, an intermediate possibility, which is to order time like the rational numbers. It is not always appreciated what a bizarre model of time is implied by this option: in particular, rational time is unable to provide an intuitively reasonable model of continuous change.

(We could equally put '$i'$ meets $j$' instead of '$i$ meets $j$'': the axiom $\mathbf{AH_1}$ given in Chapter 1, §1.4.2, ensures that these two conditions stand or fall together.) An ordering relation $\prec$ on instants can be defined by the rule

$i][j \prec k][l$ iff there is an interval $r$ such that $i$ meets $r$ and $r$ meets $l$.

If interval $i$ meets interval $j$, we shall use the notation $i + j$ to denote the interval which begins when $i$ begins and ends when $j$ ends; $i + j$ therefore represents the "sum" of the two intervals in an intuitively natural sense.*

Almost all the phenomena discussed in the previous section can be described in the current setting also, although the definitions have to be modified in order to take account of the non-existence of atomic intervals. In some of the following definitions we refer to the "beginning" and "end" of an interval $i$, denoted $beg(i)$ and $end(i)$ respectively; these refer to the instants which mark the meeting points of $i$ with intervals which respectively meet or are met by $i$.

**Ingr$_{\text{CONT}}$** $Ingr(S)$ occurs at $i][j$ iff $-S$ holds on some interval which meets $j$, and $S$ holds on some interval which $i$ meets.

**Po$_{\text{CONT}}$** $Po(S)$ occurs on $i$ iff $S$ holds on $i$ and $-S$ holds both on an interval which meets $i$ and an interval which $i$ meets.

**Consec-D$_{\text{CONT}}$** If $E$ is a durative event, then

- $Consec(E, 1)$ occurs on $i$ iff $E$ occurs on $i$;
- For $n > 1$, $Consec(E, n)$ occurs on $i$ iff there is an initial subinterval $j$ and a final subinterval $k$ of $i$ such that $end(j) \preceq beg(k)$, $E$ occurs on $j$ but not on any interval $l$ such that $end(j) \prec end(l) \prec beg(k)$, and $Consec(E, n-1)$ occurs on $k$ but not on any interval $l$ such that $end(j) \prec beg(l) \prec beg(k)$.

**Consec-I$_{\text{CONT}}$** If $E$ is an instantaneous event, then

- $Consec(E, 2)$ occurs on $i$ iff $E$ occurs at both $beg(i)$ and $end(i)$.
- $Consec(E, n)$ occurs on $i$ iff $E$ occurs at $beg(i)$, and there is a proper final subinterval $j$ of $i$ such that $Consec(E, n-1)$ occurs on $j$ and $E$ does not occur between $beg(i)$ and $beg(j)$.

**Times-D$_{\text{CONT}}$** If $E$ is a durative event, then

- $Times(E, 1)$ occurs on $i$ iff $E$ occurs on $i$;
- For $n > 1$, $Times(E, n)$ occurs on $i$ iff $i$ has a proper initial subinterval $j$ and a proper final subinterval $k$ such that $E$ occurs on $j$ but not on any other subinterval of $i$ beginning earlier than $k$, and $Times(E, n-1)$ occurs on $k$.

**Times-I$_{\text{CONT}}$** If $E$ is an instantaneous event, then $Times(E, n)$ occurs on $i$ iff $Consec(E, n)$ occurs on $i$.

*The existence of such an interval is ensured by axiom $\mathbf{AH_5}$ of Allen and Hayes; see Chapter 1, Section 1.4.2.

**ISC$_{\text{CONT}}$** If $E_1$ and $E_2$ are both durative, then $E_1; E_2$ occurs on the interval $i$ iff there are intervals $j, k$ such that $i = j + k$, $E_1$ occurs on $j$, and $E_2$ occurs on $k$.

If $E_1$ is durative and $E_2$ is instantaneous, then $E_1; E_2$ occurs on $i$ iff $E_1$ occurs on $i$ and $E_2$ occurs at $end(i)$.

If $E_1$ is instantaneous and $E_2$ is durative, then $E_1; E_2$ occurs on $i$ iff $E_1$ occurs at $beg(i)$ and $E_2$ occurs on $i$.

**GSC$_{\text{CONT}}$** If $E_1$ and $E_2$ are both durative, then $E_1;; E_2$ occurs on the interval $i$ iff $i$ contains an initial subinterval $j$ and a final subinterval $k$ such that $end(j) \preceq beg(k)$, $E_1$ occurs on $j$ but not on any interval $l$ such that $end(j) \prec end(l) \prec beg(k)$, and $E_2$ occurs on $k$ but not on any interval $l$ such that $end(j) \prec beg(l) \prec beg(k)$.

If $E_1$ is durative and $E_2$ is instantaneous, then $E_1;; E_2$ occurs on $i$ iff there is an initial subinterval $j$ of $i$ such that $E_1$ occurs on $j$ but not on any interval $k$ such that $end(j) \prec end(k) \prec end(i)$, and $E_2$ occurs at $end(i)$ but at no instant $t$ such that $end(j) \preceq t \prec end(i)$.

If $E_1$ is instantaneous and $E_2$ is durative, then $E_1;; E_2$ occurs on $i$ iff there is a final subinterval $j$ of $i$ such that $E_2$ occurs on $j$ but not on any interval $k$ such that $beg(i) \prec beg(k) \prec beg(j)$, and $E_1$ occurs at $beg(i)$ but at no instant $t$ such that $beg(i) \prec t \preceq beg(j)$.

If $E_1$ and $E_2$ are both instantaneous, then $E_1;; E_2$ occurs on $i$ iff $E_1$ occurs at $beg(i)$, $E_2$ occurs at $end(i)$, and neither event occurs on any instant dividing $i$.

**Trans$_{\text{CONT}}$** If $S_1$ and $S_2$ are two mutually incompatible states, then the event $Trans(S_1, S_2)$ has

- an instantaneous occurrence at $i][j$ iff $S_1$ holds over some interval which meets $j$ and $S_2$ holds over some interval which $i$ meets; and
- a durative occurrence on $i$ iff $S_1$ holds over some interval which meets $i$ and $S_2$ holds over some interval which $i$ meets, and both $-S_1$ and $-S_2$ hold on $i$.

We can similarly modify the holding conditions of states, as follows:

**Prog$_{\text{CONT}}$** For a durative event $E$, the state $Prog(E)$ holds on the interval $i$ iff $E$ occurs on an interval $j$ such that $i$ is a subinterval of $j$.

**Perf$_{\text{CONT}}$** For a durative event $E$, the state $Perf(E)$ holds on the interval $i$ iff $E$ occurs on some interval which is before or meets $i$.

For an instantaneous event $E$, the state $Perf(E)$ holds on the interval $i$ iff $E$ occurs on some instant no later than the beginning of $i$.

**Pros$_{\text{CONT}}$** For a durative event $E$, the state $Pros(E)$ holds on $i$ iff $E$ occurs on some interval which $i$ is before or meets.

For an instantaneous event $E$, the state $Pros(E)$ holds on $i$ iff $E$ occurs at some instant no earlier than the end of $i$.

The one operator we defined over discrete time which does not straightforwardly carry over into continuous time is $For$. The event-type $For(S, n)$ cannot be defined in continuous time unless we stipulate how durations are to be measured. Discrete time comes with its own inherent measure of duration, obtained by counting atomic intervals, but for continuous time we have to define duration separately, over and above the definition of qualitative ordering relation on intervals.

We have changed the notation and the language we use for talking about intervals, but how much has really changed in the transition from discrete to continuous time? This depends on how much we allow ourselves to exploit the infinitely dissectible nature of the time line. A common procedure is actually to negate the potential dissectibility by insisting that the pattern of holding of every state behaves in a way that can, in effect be simulated in discrete time. This is done by introducing a "non-intermingling" principle [Galton, 1996b]. The most satisfactory form of this principle is that of Davis [Davis, 1992], which we may formulate as the following "finite dissection" rule:

**FD** Every interval $i$ can be partitioned into finitely many non-overlapping intervals $i = i_1 + i_2 + \cdots + i_n$ such that for $1 \leq m \leq n$, either $S$ or $-S$ holds on $i_m$.

Suppose now that **FD** holds for every state, and that we have only finitely many primitive states $S_1, \ldots, S_k$. Take an arbitrary interval $i$. For each state $S_k$, there is a partition $i = i_{k,1} + i_{k,2} + \cdots + i_{k,n_k}$ into subintervals over which $S_k$ has constant value. From this partition we can derive a set of instants

$$\mathcal{I}_k = \{i_{k,1}][i_{k,2}, i_{k,2}][i_{k,3}, \ldots, i_{k,n_k-1}][i_{k,n_k}\}.$$

Let

$$\mathcal{I} = \{t_0, t_s\} \cup \bigcup_{k=1}^{n} \mathcal{I}_k,$$

where $t_0$ and $t_s$ are the beginning and end of $i$ respectively. Relabel the elements of $\mathcal{I}$ in ascending order as $t_0 < t_1 < t_2 < \ldots < t_{s-1} < t_s$. Now for $0 < r \leq s$, consider the interval $j_r = (t_{r-1}, t_r)$ which begins at $t_{r-1}$ and ends at $t_r$. Then each of the states $S_1, \ldots, S_k$ has constant value over this interval, since for each of the states it is a subinterval of one of the elements of the partition of $i$ determined by that state. Hence *the world does not change over the interval* $j_r$.

Since we can partition any interval into non-overlapping subintervals over which no change occurs, our temporal model is isomorphic to a discrete time model in which each of these "non change" intervals is an atomic interval (or, if we prefer, is composed of some *finite* sequence of atomic intervals). We conclude, then, that in the presence of *finitely many* primitive states and the *finite dissection* principle, a continuous-time model does not yield any phenomena over and above what is already afforded by discrete-time models.

This does not mean that finite-state, finite-dissection continuous-time (FFC) models are of no value. Suppose, for example, we have a FFC model $\mathcal{M}$. Corresponding to this there will be a discrete model $\mathcal{M}_{\mathbf{D}}$ constructed as described above. Now suppose we wish to add an extra primitive state to the model. We can do this to the FFC model without in any way disturbing what we already have in place: we still have the same intervals with the same relations between them, and all temporal phenomena definable in terms of our initial set of primitive states remain unchanged. All that happens is that we are *adding* some new

information. Call the new FFC model $\mathcal{M}'$. We can construct a discrete model $\mathcal{M}'_{\mathbf{D}}$ from this model also. Now consider the relationship between $\mathcal{M}_{\mathbf{D}}$ and $\mathcal{M}'_{\mathbf{D}}$. Whereas $\mathcal{M}$ and $\mathcal{M}'$ have exactly the same instants and intervals, in general we will expect $\mathcal{M}'_{\mathbf{D}}$ to have *more* atomic intervals than $\mathcal{M}'_{\mathbf{D}}$. An atomic interval $n$ in $\mathcal{M}_{\mathbf{D}}$ might be divided into a sequence $n', n'+1, n'+2, \ldots, n'+m$ in $\mathcal{M}'_{\mathbf{D}}$, since although the primitive states in the former model all remain unchanged over the interval $n$, the new state which has been added to produce the latter model might change its value $m$ times over that interval.

It follows that unless we know in advance some discrete sequence of intervals such that none of the states we will ever need to consider in our model changes value within any of the intervals in the sequence, we would be well advised to opt for a continuous model of time. This allows us much more flexibility when it comes to updating our model by the addition of new states (or even change of information regarding the pattern of incidence of existing states). And of course, if we want to relax either the finite-state constraint, or the finite-dissection rule, then continuous time immediately affords phenomena that prevent the simple transformation to a discrete model.

## 2.5 Continuous state-spaces

Up to now we have considered models in which the number of distinct primitive states or fluents is finite, and in which each fluent can take on only finitely many distinct values. As soon as we relax these constraints, new phenomena appear.

We shall consider a single fluent $f$ which is capable of taking arbitrary real-number values. (We could restrict this to real numbers in a given range, e.g. $(0, 1)$, but this will not make any difference to what we consider below.) Suppose we have intervals $i$, $j$, and $k$ such that $i$ meets $j$ and $j$ meets $k$, that the state $f = 0$ holds over $i$, that $f = 1$ holds over $k$, and that neither $f = 0$ nor $f = 1$ holds over any subinterval of $j$. By $\mathbf{Trans}_{\text{CONT}}$, it follows that $Trans(f = 0, f = 1)$ occurs over the interval $j$. But what exactly happens over the interval $j$?

In the previous sections we have adhered to the principle that the only sense in which a state can be said to hold at an instant is that it holds over an interval within which that instant falls. In the finite-state setting of those sections this is indeed a perfectly reasonable principle. If we apply it to the present case, what do we obtain?

We consider two possibilities. The first is that the finite dissection principle **FD** holds. In that case, we can divide $j$ up into a sequence of contiguous subintervals $j_1, j_2, \ldots, j_n$ over each of which the value of $f$ is constant. For $r = 1, \ldots, n$, let $f$ take the value $v_r$ over interval $j_r$. We have the situation pictured in Figure 2.3(a). This transition comprises a sequence of *discontinuous* steps, since for each $r$ the value of $f$ changes from $v_r$ to $v_{r+1}$ without passing through any intermediate values.

Can we get $f$ to change continuously by relaxing the condition **FD**, while maintaining the requirement that whenever any state holds, it holds over an interval? Only by means of a highly artificial construction! To illustrate, we shall construct a continuous function $f : (0, 1) \rightarrow (0, 1)$. The construction proceeds in a sequence of stages as follows:

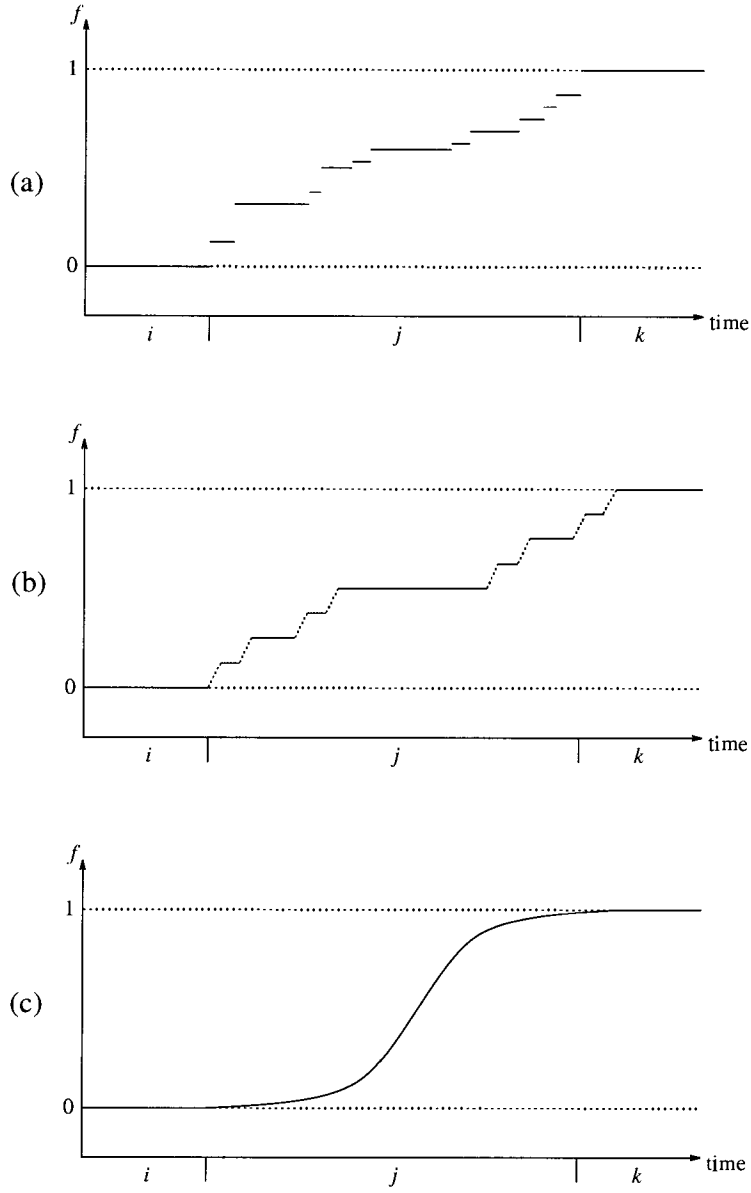



Figure 2.3: *Transitions from* $f = 0$ *to* $f = 1$.

| | |
|---|---|
| *Stage 1* | For $x \in [1/3, 2/3]$, let $f(x) = 1/2$ |
| *Stage 2* | For $x \in [1/9, 2/9]$, let $f(x) = 1/4$ |
| | For $x \in [7/9, 8/9]$ let $f(x) = 3/4$ |
| *Stage 3* | For $x \in [1/27, 2/27]$ let $f(x) = 1/8$ |
| | For $x \in [7/27, 8/27]$ let $f(x) = 3/8$ |
| | For $x \in [19/27, 20/27]$ let $f(x) = 5/8$ |
| | For $x \in [25/27, 26/27]$ let $f(x) = 7/8$ |
| ⋮ | ⋮ |
| *Stage n* | For $x \in [1/3^n, 2/3^n]$ let $f(x) = 1/2^n$ |
| | For $x \in [7/3^n, 8/3^n]$ let $f(x) = 3/2^n$ |
| | ⋮ |
| | For $x \in [\frac{3^n-2}{3^n}, \frac{3^n-1}{3^n}]$ let $f(x) = \frac{2^n-1}{2^n}$ |
| ⋮ | ⋮ |

If we identify $j$ with the temporal interval $(0, 1)$, this gives us the graph shown schematically in Figure 2.3(b). The change represented here is continuous, in the sense that the total change over an interval around any instant can be made as small as we like by choosing the interval to be short enough; note, however, that the set of values actually taken by the fluent is the set of rational numbers in [0,1] of the form $2^q p$ (where $p$ and $q$ are integers). Moreover, for every value of the form $2^q p$, there is some interval (of length $3^{-q}$) over which the fluent takes that value. If all change were of this kind, then just as in the case of FFC systems, we would never need to say that a state holds at an instant.

As we have noted, the example just described is highly artificial. Nobody really believes that change in the real world occurs in this way! A much more natural model for change is illustrated in Figure 2.3(c). Here the fluent $f$ is represented as a *smooth* continuous function. Often such functions will be expressible analytically by means of a formula such as $f(t) = 3x^2 - 2x^3$, which more or less fits the graph in our illustration over the interval $j = (0, 1)$. With this kind of change, we can no longer insist that it makes no sense to say that a state holds at an instant. During the course of a continuous change such as the one illustrated, no state of the form $f = v$ holds over an interval, yet for each $v$ in the range $(0, 1)$, this state holds at some instant during the interval $j$. For example, the state $f = \frac{1}{9}$ holds at an instant exactly one third of the way through the interval—the instant to which it is natural, in this context, to assign the number $\frac{1}{3}$.

For this kind of description to be possible, we need to break away from our previous assumption and take seriously the idea of a state holding at an instant. The obvious possibility is to embrace the standard mathematical account of continuous change, by which the time line is represented by the real numbers, with each real number corresponding to exactly one instant, and a continuously-varying fluent represented by a continuous function on the real numbers. In this model, fluents are *primarily* evaluated at instants rather than over intervals, and hence conceptual priority is given to instants over intervals.

On this basis we define what it is for a state to hold over an interval by means of the equivalence

$$Holds(S, i) \Leftrightarrow \forall t \in i Holds(S, t).$$

Here we use the conventional set-theoretic notation $t \in i$ to express the relation between an instant and an interval within which it falls, without necessarily thereby embracing the idea

that an interval *is* a set of instants. All the definitions in Section 2.4 can now be carried over unchanged into the present setting.

The mathematical language that is used in this kind of model is far removed from the everyday, qualitative terms we use for describing and reasoning about change in the world of experience. The mathematical language is necessary for the precise quantitative work required in many technical and scientific contexts, but it cannot be necessary for *all* our reasoning about change, since otherwise we would be unable to perform such reasoning without it: yet manifestly in our everyday lives we are frequently able to obtain an understanding of situations involving change that is at least adequate to enable us to achieve the more mundane of our everyday goals, without ever broaching on the complexities of a mathematical analysis. An important goal for AI is precisely to provide a model of the world which is capable of supporting this kind of everyday, rough-and-ready, qualitative reasoning.

It seems that a number of points of view are possible here. The most uncompromising would be to insist that we make full use of the available mathematical machinery, translating *all* everyday qualitative descriptions into the language of such machinery. To many this seems like overkill, and in any case the computational complexities involved may be prohibitively expensive. An alternative route is to embrace the conceptual clarity afforded by the finite-state interval-based models we have been looking at in previous sections, acknowledging that such models are unable to capture continuous phenomena, but using them to construct approximations to such phenomena that are adequate for whatever our immediate purposes are. This is very much in the *satisficing* spirit of AI, which prefers an imperfect solution that *works* to a theoretically perfect solution that is unmanageable in practice.

Aside from practicality, there are also philosophical reasons for questioning the viability of the standard mathematical model as a true account of time and change. For on the one hand, an instant is nothing: that is, it has no duration, and exists only as a point of potential division of an interval. As such, instants cannot provide the "substance" from which the extended temporal continuum is constructed. On the other hand, the standard mathematical account of continuity suggests that everything that happens is reducible to the holding of states at instants. It begins to seem paradoxical that anything can ever happen at all!

In order to escape from this paradox, while retaining the full power of the mathematical analysis of continuity, we must look for a way to achieve the same effect in a system in which intervals still play the leading role. One way of doing this might be as follows. Take the primitive notion of temporal incidence to be the holding of a state over an interval. Remember, though, that we have a wide range of states to choose from, and in particular, the state which holds over an interval may be specified as a disjunction of more primitive states. To achieve the effect we require, we must generalise the notion of disjunction to allow infinitely many disjuncts, in effect introducing existential quantification over states. Thus if $f$ is a real-valued fluent, we shall want to introduce a state such as $0 < f < 1$, which can be regarded as the disjunction of an infinite set of states $\bigvee\{f = x \mid 0 < x < 1\}$, or as a quantified state of the form $\exists x(0 < x < 1 \wedge f = x)$. In a similar way, we can form infinite conjunctions, in effect introducing universal quantification over states.

We can now define what it means to say that state $S$ holds at instant $t$ as follows. First, consider the set of all intervals within which $t$ falls, which can be defined as $\mathcal{I}_t = \{i+j \mid t = i][j\}$. For each such interval, we can find states which hold over that interval; if need be, we

form a disjunction to make sure we cover the whole interval. Now let

$$S_t = \bigwedge_{i \in \mathcal{I}_t} \{S \mid Holds(S, i)\}.$$

This is the conjunction of all the states holding over any interval within which $t$ falls. We can then stipulate that a state $S$ holds at $t$ if and only if it is entailed by $S_t$. To illustrate, we look at a case of continuous change and a case of discontinuous change.

1. *Continuous change.* Let the fluent $f$ take the value 0 over the interval (0,1) and 1 over the interval (3,4), with intermediate values over the interval (1,3). Thus the event $Trans(f = 0, f = 1)$ occurs on (1,3). Assume that the value of $f$ changes at a uniform rate over that interval. What is the value of the fluent at $t = 2$? We note that on the interval $(2 - \epsilon, 2 + \epsilon)$, where $\epsilon > 0$, the state $1.5 - \frac{1}{2}\epsilon < f < 1.5 + \frac{1}{2}\epsilon$ holds. Thus

$$S_2 = \bigwedge_{\epsilon > 0} \{1.5 - \tfrac{1}{2}\epsilon < f < 1.5 + \tfrac{1}{2}\epsilon\} = (f = 1.5).$$

   We conclude that the value of $f$ at $t = 2$ is 1.5.

   This example has an air of circularity: the point is to show how we can derive the state holding at an instant from a knowledge of states holding over intervals. But how do we know that $f$ takes values in the range $(1.5 - \frac{1}{2}\epsilon, 1.5 + \frac{1}{2}\epsilon)$ over the interval $(2 - \epsilon, 2 + \epsilon)$? The simplest way of determining this is to calculate that $f$ has the value $1.5 - \frac{1}{2}\epsilon$ at $t = 2 - \epsilon$, and similarly at the other end of the interval (noting also that $f$ is increasing monotonically throughout the interval). Thus although it is possible, *given* a knowledge of what states hold over what intervals, to derive the states holding at instants, it is not at all clear where the knowledge, supposed given, could have come from, other than a prior knowledge of what states hold at what instants, the very information we were trying to derive!

2. *Discontinuous change.* Suppose $f = 0$ over the interval (0,1), and $f = 1$ over the interval $(1, 2)$. What can we say about the value of $f$ at $t = 1$? This is the classic Dividing Instant Problem. We note that on the interval $(2 - \epsilon, 2 + \epsilon)$, where $\epsilon > 0$, the state $f = 0 \vee f = 1$ holds. Hence

$$S_2 = \bigwedge_{\epsilon > 0} \{f = 0 \vee f = 1\} = (f = 0 \vee f = 1).$$

   This corresponds to the solution to the Dividing Instant Problem according to which the value of $f$ at the instant of transition is indeterminate; and this certainly seems the most appropriate answer to give in the case of discontinuous change envisaged here.

It is possible that some such approach as this might work, despite the appearance of circularity noted above, but it does not appear to have been investigated in any detail. For the present, we must leave it as an open question; the main lesson to be learnt from this section is that continuity—a subject which received much attention from ancient and medieval philosophers as well as modern mathematicians—is difficult, and that the modern mathematical approach is surely not the last word on the subject.

## 2.6 Case study: A game of tennis

In this section we shall illustrate the systems described in the previous section by showing how they can be used to describe a single objective situation at different levels of detail.* The situation we shall consider is a tennis match. A tennis match between two players consists of a certain number of *sets* each consisting of a certain number of *games*. Each game consists of a certain number of *points*. Thus we have a hierarchical structure which lends itself well to a description at different levels. We shall by-pass a consideration of the upper levels, and concentrate on a single game in the match.

As already mentioned, the game consists of a sequence of points. Physically, a point consists in the players attempting to hit the ball from one end of the court to the other until one of a number of termination conditions obtains—e.g., the ball hits the net or lands outside a certain designated area of the court, or it lands inside the area but the player whose turn it is to hit it fails to do so. Each such condition determines which player wins the point. The game is won by the first player to have at least two more points than his opponent *and* at least four points altogether. The bizarre conventions regarding the naming of scores are shown in Figure 2.4, in which the possible courses of the game are shown as paths through a finite-state automaton. In this figure $S$ represents a point won by the server, who is the first to hit the ball after each point, and $R$ represents a point won by his opponent, who returns the serve.

Figure 2.4 most naturally lends itself to a description of the game as a finite-state system operating in discrete time. All possible games are covered by the diagram. An example of a particular game might be presented as

(0–0)(15–0)(15–15)(15–30)(15–40)(30–40)(deuce)(advantage R)(deuce)(advantage S)(game to S)

The individual "times" here are, essentially, points, in the sense of covering all the play following one point and leading up to the next.

The example of the tennis game differs from the finite-state systems we considered earlier in one important respect, which is that whereas in the earlier systems it was assumed that the primitive facts defining the system were states, with all events defined in terms of them, in the tennis example it is more natural to take certain events as the primitive elements, and define states in terms of them. At the level of detail shown in Figure 2.4, there are two primitive events, namely "S wins a point" and "R wins a point". The state represented by the score 15–30 is defined as, essentially "Since the start of the game, S has won one point and R has won two points". The sequence of states illustrated above can also be represented as a sequence of events:

SRRRSRRSSS

and representations of this kind are interchangeable, in the present example, with the state-based representations we had earlier. This interchangeability relies on the finite-state system being deterministic; with a non-deterministic system it is not possible to recover the state sequence from the event sequence. And if there is more than one primitive event type that can effect the transition between a particular pair of states, then it is also impossible to

* A detailed technical account of levels of detail in temporal representation can be found in Chapter 3; here we handle the issue in an informal way only.

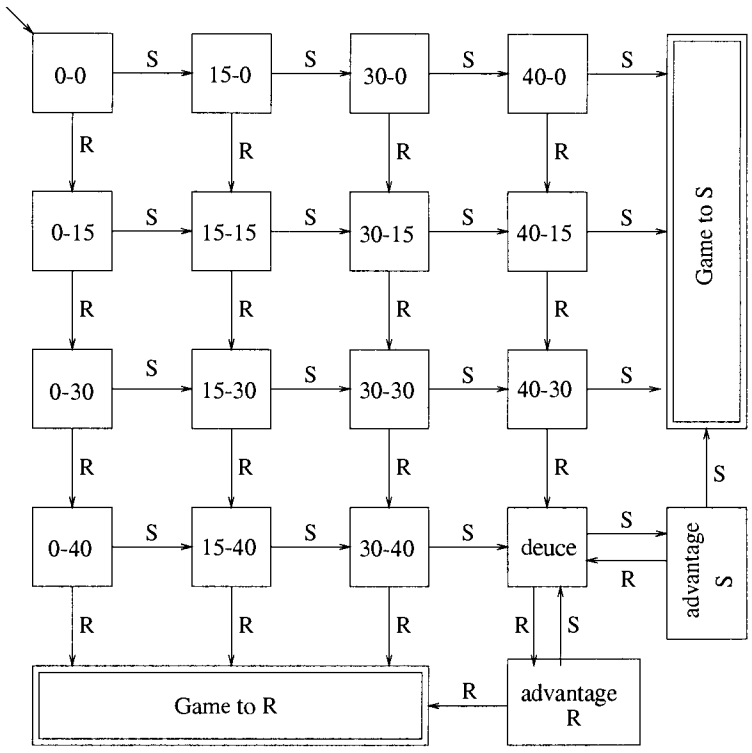



Figure 2.4: *How a game is won in tennis.*

recover the event sequence from the state sequence. In such cases the full history would have to be given by a mixed state and event sequence such as

$$(0\text{–}0)\xrightarrow{S}(15\text{–}0)\xrightarrow{R}(15\text{–}15)\xrightarrow{R}(15\text{–}30)\xrightarrow{R}(15\text{–}40)\xrightarrow{S}(30\text{–}40)\xrightarrow{S}(\text{deuce})\xrightarrow{R}$$
$$(\text{advantage R})\xrightarrow{S}(\text{deuce})\xrightarrow{S}(\text{advantage S})\xrightarrow{S}(\text{game to S})$$

We can define higher-level states on this system by forming disjunctions of the primitive states shown in the diagram. An example is "R is winning", which is the disjunction

$$(0\text{–}15) \vee (0\text{–}30) \vee (0\text{–}40) \vee (15\text{–}30) \vee (15\text{–}40) \vee (30\text{–}40) \vee (\text{advantage R}).$$

In the sequence above, this state holds at times 4, 5, 6, and 8. Thus there is an occurrence of the event $Po$(R is winning) (i.e., "R was winning for a while") on the interval [4,6]. One might say "after the third point, $R$ started winning", i.e., $Occurs(Ingr(\text{R is winning}), 3][4)$. Another high-level state is "Break point", used by tennis commentators to describe the situation in which R only needs to win the next point in order to win:

$$(0\text{–}40) \vee (15\text{–}40) \vee (30\text{–}40) \vee (\text{advantage R}).$$

In the automaton, the arcs representing transitions from state to state are labelled "S" or "R" according to whether the server or his opponent wins the next point. These transitions can

be regarded as instantaneous event-types. More exactly, the event-type "R wins the point" can be given a disjunctive occurrence condition of the form

$$\begin{aligned} Occurs(\text{R's point}, n][n+1) \quad \Leftrightarrow \quad & (Holds(\text{0–0}, n) \wedge Holds(\text{0-15}, n+1)) \vee \\ & (Holds(\text{15–0}, n) \wedge Holds(\text{15–15}, n+1)) \vee \\ & \vdots \\ & (Holds(\text{deuce}, n) \wedge Holds(\text{advantage R}, n+1)) \vee \\ & (Holds(\text{advantage R}, n) \wedge Holds(\text{game to R}, n+1)) \end{aligned}$$

Note that we are defining the occurrence condition for R to score a point in terms of transitions between particular scores—which is of course the reverse of the real logical dependence, in which the current score is determined by the previous score together with who won the last point. This order of dependence could be captured by a set of holding conditions such as

$$\begin{aligned} Holds(\text{15–30}, n) \quad \Leftrightarrow \quad & (Holds(\text{0–30}, n-1) \wedge Occurs(\text{S's point}, n-1][n)) \vee \\ & (Holds(\text{15–15}, n-1) \wedge Occurs(\text{R's point}, n-1][n)) \end{aligned}$$

At this level of granularity, it is in the nature of the game that none of the primitive states can hold over two or more consecutive atomic intervals. In fact, apart from "deuce" and the two advantage states, none of them can occur more than once in the entire game. We can move to a finer granularity, while still operating with a finite-state system in discrete time, by tracking the events within a point. Each point consists of a sequence of *shots*; a shot is an event in which a player hits, or attempts to hit, the ball. A shot may be considered "good" or "bad". To win the point a player needs to deliver a "good" shot which the opponent responds to with a "bad" shot. The different ways in which a point may be won by a succession of shots are shown in the finite-state automaton in Figure 2.5.

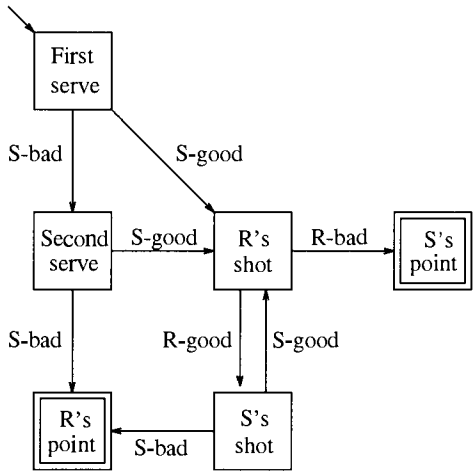



Figure 2.5: *How a point is won in tennis.*

Writing $X^+$ and $X^-$ to represent good and bad points for player $X$, we can represent the history of a particular point in the form

$$S^- S^+ R^+ S^+ R^+ S^+ R^+ S^-$$

and of course each point in a game can be analysed in this form also, giving us a two-level representation as follows:

| $\underbrace{S^+R^-}_{S}$ | $\underbrace{S^+R^+S^+R^-}_{S}$ | $\underbrace{S^-S^+R^-}_{S}$ | $\underbrace{S^+R^+S^+R^+S^-}_{R}$ | $\underbrace{S^-S^-}_{R}$ | $\underbrace{S^+R^+S^+R^+S^+R^-}_{S}$ | |
|---|---|---|---|---|---|---|
| 0–0 | 15–0 | 30–0 | 40–0 | 40–15 | 40–30 | Game to S |

At the next level of detail, we do not introduce any new states or events, but we do take into account how long each shot takes. For this purpose we move to a continuous model of time. The game shown above can now be represented as shown in Figure 2.6. As is to be expected from our earlier remarks, we do not see any new qualitative phenomena in this representation; the extra information conveyed is all quantitative, concerning the relative durations of what previously were treated as atomic intervals.

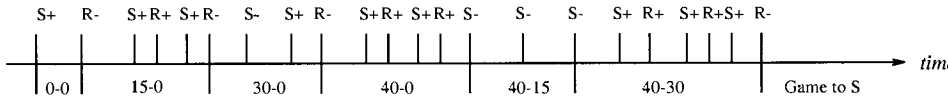



Figure 2.6: *A game of tennis in continuous time.*

At the final level of detail, not only time itself, but also the fluents defined over time, take values from continuous ranges. In the tennis example, this is the point at which we turn from the rather abstract, conventionalised point of view characterised in terms of scores, advantages, and so on, to a lower-level, physicalistic point of view from which the game is characterised in terms of the motions of bodies—the players and the ball—about the court.

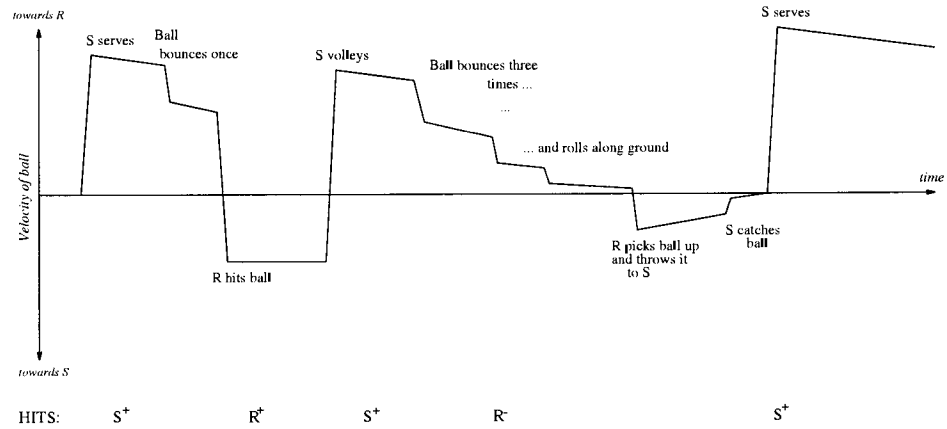



Figure 2.7: *Speed of ball along court plotted as a function of time.*

There are innumerable different fluents we could consider here, of varying relevance to the higher-level descriptions of the game. For the sake of illustration, we shall take just one fluent, which generates a good deal of interest in tennis circles, namely the speed of the ball. What is of particular interest is the speed at which the ball is served at the start of each point, but of course the ball *has* a speed at every instant of the game.

In Figure 2.7 we present a history of one point in terms of ball-speed; more precisely, what is plotted (on the vertical axis) is the component of velocity parallel to the long axis of the court, with the direction from server to opponent taken as positive. The correlation with the previous level of description is shown by the inclusion of terms like $S^+$ to indicate the type of hit as characterised at that level. It has to be admitted that the graph shown here is entirely imaginary, no real data of this kind being available to the author, but I hope it is not too implausible!

The main lesson to be learnt from this case study is that there is no question of there being one correct way to describe what goes on in time. In the preceding sections we looked at a number of different frameworks within which such descriptions can be given, which are incompatible in the sense that if there is one "correct" model then only one of the frameworks can merit that title. But as we have seen in this section, each of the frameworks can provide a valid way of representing a particular piece of (possible or actual) history, depending on what aspects one wants to focus on.

Handbook of Temporal Reasoning in Artificial Intelligence
Edited by M. Fisher, D. Gabbay and L. Vila


Chapter 3

# Time Granularity

Jérôme Euzenat & Angelo Montanari

A temporal situation can be described at different levels of abstraction depending on the accuracy required or the available knowledge. Time granularity can be defined as the resolution power of the temporal qualification of a statement. Providing a formalism with the concept of time granularity makes it possible to model time information with respect to differently grained temporal domains. This does not merely mean that one can use different time units, e.g., months and days, to represent time quantities in a unique flat temporal model, but it involves more difficult semantic issues related to the problem of assigning a proper meaning to the association of statements with the different temporal domains of a layered temporal model and of switching from one domain to a coarser/finer one. Such an ability of providing and relating temporal representations at different "grain levels" of the same reality is both an active research theme and a major requirement for many applications (e.g., integration of layered specifications and agent communication).

After a presentation of the general requirements of a multi-granular temporal formalism, we discuss the various issues and approaches to time granularity proposed in the literature. We focus our attention on the main existing formalisms for representing and reasoning about quantitative and qualitative time granularity: the set-theoretic framework developed by Bettini et al. [Bettini *et al.*, 2000] and the logical approach systematically investigated by Montanari et al. [Montanari, 1996; Franceschet, 2002] for quantitative time granularity, and Euzenat's relational algebra granularity conversion operators [Euzenat, 2001] for qualitative time granularity. We present in detail the achieved results, we outline the open issues, and we point out the links that connect the different approaches. In the last part of the chapter, we describe some applications exploiting time granularity, and we briefly discuss related work in the areas of formal methods, temporal databases, and data mining.

## 3.1 Introduction

The usefulness of the addition of a notion of time granularity to representation languages is widely recognized. As an example, let us consider the problem of providing a logical specification of a wide-ranging class of real-time reactive systems whose components have dynamic behaviors regulated by very different — even by orders of magnitude — time constants (*granular systems* for short) [Montanari, 1996]. This is the case, for instance, of a pondage power station that consists of a reservoir, with filling and emptying times of days or weeks, generator units, possibly changing state in a few seconds, and electronic control

devices, evolving in milliseconds or even less [Corsetti *et al.*, 1991a]. A complete specification of the power station must include the description of these components and of their interactions. A natural description of the temporal evolution of the reservoir state will probably use days: "During rainy weeks, the level of the reservoir increases 1 meter a day". The description of the control devices behavior may use microseconds: "When an alarm comes from the level sensors, send an acknowledge signal in 50 microseconds". We say that systems of such a type have *different time granularities*. It is not only somewhat unnatural, but also sometimes impossible, to compel the specifier of these systems to use a unique time granularity, microseconds in the previous example, to describe the behavior of all the components. For instance, the requirement that "the filling of the reservoir must be completed within $m$ days" can be hardly assumed to be equivalent to the requirement that "the filling of the reservoir must be completed within $n$ microseconds", for a suitable $n$ (we shall discuss in detail the problems involved in such a rewriting in the next section). Since a good language must allow the specifier to easily and precisely describe all system requirements, different time granularities must be a feature of a specification language for granular systems.

A complementary point of view on time granularity is also possible: besides an important feature of a representation language, time granularity can be viewed as a formal tool to investigate the definability of meaningful timing properties, such as density and exponential grow/decay, as well as the expressiveness and decidability of temporal theories [Montanari *et al.*, 1999]. In this respect, the number and organization of layers (single vs. multiple, finite vs. infinite, upward unbounded vs. downward unbounded) of the underlying temporal structure plays a major role: certain timing properties can be expressed using a single layer; others using a finite number of layers; others only exploiting an infinite number of layers. In particular, finitely-layered metric temporal logics can be used to specify timing properties of granular systems composed by a finite number of differently-grained temporal components, which have been fixed once and for all ($n$-layered temporal structures). Furthermore, if provided with a rich enough layered structure, they suffice to deal with conditions like "$p$ holds at all even times of a given temporal domain" that cannot be expressed using flat propositional temporal logics [Emerson, 1990] (as a matter of fact, a 2-layered structure suffices to capture the above condition). $\omega$-layered metric temporal logics allow one to express relevant properties of infinite sequences of states over a single temporal domain that cannot be captured by using flat or finitely-layered temporal logics. This is the case, for instance, of conditions like "$p$ holds at all times $2^i$, for all natural numbers $i$, of a given temporal domain".

The chapter is organized as follows. In Section 3.2, we introduce the general requirements of a multi-granular temporal formalism, and then we discuss the different issues and approaches to time granularity proposed in the literature. In Sections 3.3 and 3.4, we illustrate in detail the two main existing formal systems for representing and reasoning about quantitative time granularity: the set-theoretic framework for time granularity developed by Bettini et al. [Bettini *et al.*, 2000] and the logical approach systematically explored by Montanari et al. [Montanari, 1996; Franceschet, 2002]. In Section 3.5, we present the relational algebra granularity conversion operators proposed by [Euzenat, 2001] to deal with qualitative time granularity and we briefly describe the approximation framework outlined by Bittner [Bittner, 2002]. In Section 3.6, we describe some applications exploiting time granularity, while in Section 3.7 we briefly discuss related work. The concluding remarks provide an assessment of the work done in the field of time granularity and give an indication

of possible research directions.

## 3.2 General setting for time granularity

In order to give a formal meaning to the use of different time granularities in a representation language, two main problems have to be solved: the qualification of statements with respect to time granularity and the definition of the links between statements associated with a given time granularity, e.g., *days*, and statements associated with another granularity, e.g., *microseconds* [Montanari, 1996]. Sometimes, these problems have an obvious solution that consists in using *different time units* — say, months and minutes — to measure time quantities in a *unique model*. In most cases, however, the treatment of different time granularities involves more difficult semantic problems. Let consider, for instance, the sentence: "every month, if an employee works, then he gets his salary". It could be formalized, in a first-order language, by the following formula:

$$\forall t_m, emp(work(emp, t_m) \rightarrow get\_salary(emp, t_m)),$$

with an obvious meaning of the used symbols, once it is stated that the subscript $m$ denotes the fact that $t$ is measured by the time unit of *months*.
Another requirement can be expressed by the sentence: "an employee must complete every received job within 3 days". It can be formalized by the formula:

$$\forall t_d, emp, job(get\_job(emp, job, t_d) \rightarrow job\_done(emp, job, t_d + 3)),$$

where the subscript $d$ denotes that $t$ is measured by the time unit of *days*.
Assume now that the two formulas are part of the specification of the same office system. We need a *common model* for both formulas. As done before, we could choose the finest temporal domain, i.e., the set of (times measured by) *days*, as the common domain. Then, a term labeled by $m$ would be translated into a term labeled $d$ by multiplying its value by 30. However, the statement "every month, if an employee works, then he gets his salary" is clearly different from the statement "every day, if an employee works, then he gets his salary". In fact, working for a month means that one works for 22 days in the month, whereas getting a monthly salary means that there is one day when one gets the salary for the month. Similarly, stating that "every day of a given month it rains" does not mean, in general, that "it rains for all seconds of all days of the month". On the contrary, if one states that "a car has been moving for three hours at a speed greater than 30 km per hour", he usually means that for all seconds included in the considered three hours the car has been moving at the specified speed. The above examples show that the interpretations of temporal statements are likely to change when switching from one time granularity to another one. The addition of the concept of time granularity is thus necessary to allow one to build granular temporal models by referring to the natural scale in any component of the model and by properly constraining the interactions between differently-grained components.

Further difficulties arise from the *synchronization problem* of temporal domains [Corsetti *et al.*, 1991a]. Such a problem can be illustrated by the following examples. Consider the sentence "tomorrow I will eat". If one interprets it in the domain of hours, its meaning is that there will be several hours, starting from the next midnight until the following one, when it will be true that I eat, *no matter in which hour of the present day this sentence is claimed.*

Thus, if the sentence is claimed at 1 a.m., it will be true that "I eat" at some hours $t$ whose distance $d$ from the current hour is such that $23 \leq d < 47$. Instead, if the same sentence is claimed at 10 p.m. of the same day, $d$ will be such that $2 \leq d < 26$. Consider now the sentence "dinner will be ready in one hour". If it is interpreted in the domain of minutes, its meaning is that dinner will be ready in 60 minutes *starting from the minute when it is claimed*. Therefore, if the sentence is claimed at minute, say, 10, or 55, of a given hour, it will be always true that "dinner is ready" at a minute $t$ whose distance $d$ from the current minute is *exactly* 60 minutes. Clearly, the two examples require two different semantics.

Thus, when the granularity concept is applied to time, we generally assume a set of differently-grained domains (or layers) with respect to which the situations are described and some operators relating the components of the multi-level description. The resulting system will depend on the language in which situations are modeled, the properties of the layers, and the operators. Although these elements are not fully independent, we first take into consideration each of them separately.

### 3.2.1 Languages, layers, operators

The distinctive features of a formal system for time granularity depend on some basic decisions about the way in which one models the relationships between the representations of a given situation with respect to different granularity layers.

**Languages.** The first choice concerns the language. One possibility is to use the same language to describe a situation with respect to different granularity layers. As an example, the representations associated with the different layers can use the same temporal logic or the same algebra of relations. In such a way, the representations of the same situation at different abstraction levels turn out to be *homogeneous*. Another possibility is to use different languages at different levels of abstraction, thus providing a set of *hybrid* representations of the same situation. As an example, one can adopt a metric representation at the finer layers and a qualitative one at the coarser ones.

**Layers.** Any formal system for time granularity must feature a number of different (granularity) layers. They can be either explicitly introduced by means of suitable linguistic primitives or implicitly associated with the different representations of a given situation.

**Operators.** Another choice concerns the operators that the formal system must encompass to deal with the layered structure. In this respect, one must make provision for at least two basic operators:

**contextualization** to select a layer;

**projection** to move across layers.

These operators are independent of the specific formalism one can adopt to represent and to reason about time granularity, that is, each formalism must somehow support such operators. They are sufficient for expressing fundamental questions one would like to ask to a granular representation:

- converting a representation from a given granularity to another one (how would a particular representation appear under a finer or coarser granularity?);
- testing the compatibility of two representations (is it possible that they represent the same situation under different granularities?);
- comparing the relative granularities of two representations (which is the coarser/finer representation of a given situation?).

**Internal vs. external layers.** Once the relevance of these operators is established, it must be decided if the granularity applies within a formalism or across formalisms. In other terms, it must be decided if an existing formalism will be extended with these new operators or if these operators will be defined and applied from the outside to representations using existing formalisms. Both these alternatives have been explored in the literature:

- Some solutions propose an internal extension of existing formalisms to explicitly introduce the notion of granularity layer in the representations (see Sections 3.4.1 and 3.4.2 [Ciapessoni *et al.*, 1993; Montanari, 1996; Montanari *et al.*, 1999]), thus allowing one to express complex statements combining granularity with other notions. The representations of a situation with respect to different granularity layers in the resulting formalism are clearly homogeneous.
- Other solutions propose an external apprehension which allows one to relate two descriptions expressed in the same formalism or in different formalisms (see Sections 3.3, 3.4.3, and 3.5 [Euzenat, 1995b; Fiadeiro and Maibaum, 1994; Franceschet, 2002; Franceschet and Montanari, 2004]). This solution has the advantage of preserving the usual complexity of the underlying formalism, as far as no additional complexity is introduced by granularity.

### 3.2.2 Properties of languages

The whole spectrum of languages for representing time presented in this book is available for expressing the sentences subject to granularity. Here we briefly point out some alternatives that can affect the management of granularity.

**Qualitative and quantitative languages.** There can be many structures on which a temporal representation language is grounded. These structures can be compared with that of mathematical spaces:

**set-theory** when the language takes into account containment (i.e. set-membership);

**topology** when the language accounts for connexity and convexity;

**metric spaces** when the language takes advantage of a metric in order to quantify the relationship (distance) between temporal entities.

**vector spaces** when the language considers alignment and precedence (with regard to an alignment). As far as time is considered as totally ordered, the order comes naturally.

A quantitative representation language is generally a language which embodies properties of metric and vector spaces. Such a language allows one to precisely define a displacement operator (of a particular distance along an axis). A qualitative representation language does not use a metric and thus one cannot precisely state the position of objects. For instance, Allen's Interval Algebra (see Chapter 1) considers notions from vector (before) and topological (meets) spaces.

**Expressive power.** The expressive power of the languages can vary a lot (this is true in general for classical temporal representation languages, see Chapter 6). It can roughly be:

**exact and conjunctive** when each temporal entity is localized at a particular known position ($a$ is ten minutes after $b$) and a situation is described by a conjunction of such sentences;

**propositional** when the language allows one to express conjunction and disjunction of propositional statements ($a$ is before or after $b$); this also applies to constrained positions of entities ($a$ is between ten minutes and one hour after $b$);

**first-order** when the language contains variables which allow one to quantify over the entities (there exists time lap $x$ in between $a$ and $b$);

**"second-order"** when the language contains variables which allow one to quantify over layers (there exists a layer $g$ under which $a$ is after $b$).

### 3.2.3 Properties of layers

As it always happens when time information has to be managed by a system, the properties of the adopted model of time influence the representation. The distinctive feature of the models of time that incorporate time granularity is the coexistence of a set $\mathcal{T}$ of temporal domains. Such a set is called *temporal universe* and the temporal domains belonging to it are called (temporal) *layers*. Layers can be either overlapping, as in the case of `Days` and `Working Days`, since every working day is a day (cf. Section 3.3), or disjoint, as in the case of `Days` and `Weeks` (cf. Section 3.4).

**Structure of time.** It is apparent that the temporal structure of the layers influences the semantics of the operators. Different structures can obviously be used. Moreover, one can either constrain the layers to share the same structure or to allow different layers to have different structures.
For each layer $T \in \mathcal{T}$, let $<$ be a linear order over the set of time points in $T$. We confine our attention to the following temporal structures:

**continuous** $T$ is isomorphic to the set of real numbers (this is the usual interpretation of time);

**dense** between every two different points there is a point

$$\forall x, y \in T \; \exists z \in T (x < y \rightarrow x < z < y);$$

**discrete** every point having a successor (respectively, a predecessor) has an immediate one

$$\forall x \in T((\exists y \in T(x < y) \rightarrow \exists z \in T(x < z \wedge \forall w \in T\ \neg(x < w < z))) \wedge$$
$$(\exists y \in T(y < x) \rightarrow \exists z \in T(z < x \wedge \forall w \in T\ \neg(z < w < x)))).$$

Most formal systems for time granularity assume layers to be discrete, with the possible exception of the most detailed layer, if any, whose temporal structure can be dense, or even continuous (an exception is [Endriss, 2003]). The reason of this choice is that each dense layer is already at the finest level of granularity, and it allows any degree of precision in measuring time. As a consequence, for dense layers one must distinguish granularity from metric, while, for discrete layers, one can define granularity in terms of set cardinality and assimilate it to a natural notion of metric. Mapping, say, a set of rational numbers into another set of rational numbers would only mean changing the unit of measure with no semantic effect, just in the same way one can decide to describe geometric facts by using, say, kilometers or centimeters. If kilometers are measured by rational numbers, indeed, the same level of precision as with centimeters can be achieved. On the contrary, the key point in time granularity is that saying that something holds for all days in a given interval does not imply that it holds at every second belonging to the interval [Corsetti *et al.*, 1991a]. For the sake of simplicity, in the following we assume each layer to be discrete.

**Global organization of layers.** Further conditions can be added to constrain the global organization of the set of layers. So far, layers have been considered as independent representation spaces. However, we are actually interested in comparing their grains, that is, we want to be able to establish whether the grain of a given layer is finer or coarser than the grain of another one. It is thus natural to define an order relation $\prec$, called *granularity* relation, on the set of layers of $\mathcal{T}$ based on their grains: we say that a layer $T$ is finer (resp. coarser) than a layer $T'$, denoted by $T \prec T'$ (resp. $T' \prec T$), if the grain of $T$ is finer (resp. coarser) than that of $T'$. There exist at least three meaningful cases:

**partial order** $\prec$ is a reflexive, transitive, and anti-symmetric relation over layers;

**(semi-)lattice** $\prec$ is a partial order such that, given any two layers $T, T' \in \mathcal{T}$, there exists a layer $T \wedge T' \in \mathcal{T}$ such that $T \wedge T' \prec T$ and $T \wedge T' \prec T'$, and any other layer $T$" with the same property is such that $T$" $\prec T \wedge T'$;

**total order** $\prec$ is a partial order such that, for all $T, T' \in \mathcal{T}$, either $T = T'$ or $T \prec T'$ or $T' \prec T$.

We shall see that the set of admissible operations on layers depends on the structure of $\prec$.

Beside the order relation $\prec$, one must consider the cardinality of the set $\mathcal{T}$. Even though a finite number of layers suffices for many applications, there exist significant properties that can be expressed only using an infinite number of layers (cf. Section 3.4.2). As an example, an infinite number of arbitrarily fine (discrete) layers makes it possible to express properties related to temporal density, e.g., the fact that two states are distinct, but arbitrarily close.

**Pairwise organization of layers.** Even in the case in which layers are totally ordered, their organization can be made more precise. For instance, consider the case of a situation described with respect to the totally ordered set of granularities including `years`, `months`, `weeks`, and `days`. The relationships between these layers differ a lot. Such differences can be described through the following notions:

**homogeneity** when the (temporal) entities of the coarser layer consist of the same number of entities of the finer one;

**alignment** when the entities of the finer layer are mapped in only one entity of the coarser one.

These two notions allow us to distinguish four different cases:

**year-month** the situation is very neat between years and months since each year contains the same number of months (homogeneity) and each month is mapped onto only one year (alignment);

**year-week** a year contains a various number of weeks (non homogeneity) and a week can be mapped into more than one year (non alignment);

**month-day** while every day is mapped into exactly one month (alignment), the number of days in a month is variable (non homogeneity);

**working week-day** one can easily imagine working weeks beginning at 5 o'clock on Mondays (this kind of weeks exists in industrial plants): while every week is made of the same duration or amount of days (homogeneity), some days are mapped into two weeks (non alignment).

**How the objects behave.** There are several options with regard to the behavior of the objects considered by the theories. The objects can

**persist** when they remain the same across layers (in the logical setting, this is modeled by the Barcan formula);

**change category** when, moving from one layer to another one, they are transformed into objects of different size (e.g., transforming intervals into points, or vice versa, or changing an object into another of a bigger/lower dimension, see Section 3.6.4);

**vanish** when an object associated with a fine layer disappears in a coarser one.

### 3.2.4 Properties of operators

The operator that models the change of granularity is the *projection* operator. It relates the temporal entities of a given layer to the corresponding entities of a finer/coarser layer. In some formal systems, it also models the change of the interpretation context from one layer to another. The projection operator is characterized by a number of distinctive properties, including:

**reflexivity** (see Section 3.5.2 self-conservation p. 105 and Section 3.4.1 p. 85) constrains an entity to be able to be converted into itself;

**symmetry** (see Section 3.5.2 inverse compatibility p. 106 and Section 3.4.1 p. 85) states that if an entity can be converted into another one, then this latter entity can be converted back into the original one;

**order-preservation** (for vectorial systems, see Section 3.3 p. 69, Section 3.5.2 p. 105, and Section 3.4.1 p. 86) constrains the projection operators to preserve the order of entities among layers;

**transitivity** (see below) constrains consecutive applications of projection operators in any "direction" to yield the same result as a direct projection;

**oriented transitivity** (see Section 3.5.2 p. 106 and Section 3.4.1 downward transitivity p. 85 and upward transitivity p. 86) constrains successive applications of projection operators in the same "direction" to yield the same result as a direct projection;

**downward/upward transitivity** (see Section 3.4.1 pp. 85-86 and [Euzenat, 1993]) constrains two consecutive applications of the projection operators (first downward, then upward) to yield the same result as a direct downward or upward projection;

Some properties of projection operators are related to pairwise properties of layers:

**contiguity** (see Section 3.4.1 p. 86), or "contiguity-preservation", constrains the projections of two contiguous entities to be either two contiguous (sets of) entities or the same entity (set of entities);

**total covering** (see Section 3.3 p. 69 and Section 3.4.1 p. 86) constrains each layer to be totally accessible from any other layer by projection;

**convexity** (see Section 3.4.1 p. 86) constrains the coarse equivalent of an entity belonging to a given layer to cover a convex set of entities of such a layer;

**synchronization** (see Sections 3.3 and 3.4.1), or "origin alignment", constrains the origin of a layer to be projected on the origin of the other layers. It is called synchronization because it is related to "synchronicity" which binds all the layers to the same clock;

**homogeneity** (see Section 3.4.1 p. 86) constrains the temporal entities of a given layer to be projected on the same number of entities of a finer layer;

Such properties are satisfied when they are satisfied by all pairs of layers.

### 3.2.5 Quantitative and qualitative models

In the following we present in detail the main formal systems for time granularity proposed in the literature. We found it useful to make a distinction between quantitative and qualitative models of time granularity. Quantitative models are able to position temporal entities (or occurrences) within a metric frame. They have been obtained following either a set-theoretic approach or a logical one. In contrast, qualitative models characterize the position of temporal entities with respect to each other. This characterization is often topological or vectorial. The main qualitative approach to granularity is of algebraic nature.

The set-theoretic approach is based upon naive set theory and algebra. According to it, the single temporal domain of flat temporal models is replaced by a temporal universe, which is defined as a set of inter-related temporal layers, which is built upon its finest layer. The finest layer is a totally ordered set, whose elements are the smallest temporal units relevant to the considered application (*chronons*, according to the database terminology [Dyreson and

Snodgrass, 1994; Jensen *et al.*, 1994]); any coarser layer is defined as a suitable partition of this basic layer. To operate on elements belonging to the same layer, the familiar Boolean algebra of subsets suffices. Operations between elements belonging to different layers require a preliminary mapping to a common layer. Such an approach, originally proposed by Clifford and Rao in [Clifford and Rao, 1988], has been successively refined and generalized by Bettini et al. in a number of papers [Bettini *et al.*, 2000]. In Section 3.3, we shall describe the evolution of the set-theoretic approach to time granularity from its original formulation up to its more recent developments.

According to the logical approach, the single temporal domain of (metric) temporal logic is replaced by a temporal universe consisting of a possibly *infinite* set of inter-related differently-grained layers and logical tools are provided to qualify temporal statements with respect to the temporal universe and to switch temporal statements across layers. Logics for time granularities have been given both non-classical and classical formalizations. In the non-classical setting, they have been obtained by extending metric temporal logics with operators for temporal contextualization and projection [Ciapessoni *et al.*, 1993; Montanari, 1996; Montanari and de Rijke, 1997], as well as by combining linear and branching temporal logics in a suitable way [Franceschet, 2002; Franceschet and Montanari, 2003; Franceschet and Montanari, 2004]. In the classical one, they have been characterized in terms of (extensions of) the well-known monadic second-order theories of $k$ successors and of their fragments [Montanari and Policriti, 1996; Montanari *et al.*, 1999; Franceschet *et al.*, 2003]. In Section 3.4, we shall present in detail both approaches.

The study of granularity in a qualitative context is presented in Section 3.5. It amounts to characterize the variation of relations between temporal entities that are induced by granularity changes. A number of axioms for characterizing granularity conversion operators have been provided in [Euzenat, 1993; Euzenat, 1995a], which have been later shown to be consistent and independent [Euzenat, 2001]. Granularity operators for the usual algebras of temporal relations have been derived from these axioms. Another approach to characterizing granularity in qualitative relations, associated with a new way of generating systems of relations, has recently come to light [Bittner, 2002]. The relations between two entities are characterized by the relation (in a simpler relation set) between the intersection of the two entities and each of them. Temporal locations of entities are then approximated by subsets of a partition of the temporal domain, so that the relation between the two entities can itself be approximated by the relation holding between their approximated locations. This relation (that corresponds to the original relation under the coarser granularity) is obtained directly by maximizing and minimizing the set of possible relations.

## 3.3 The set-theoretic approach

In this section, we present several contributions to the development of a general framework for time granularity coming from both the area of knowledge-based systems and that of database systems. We qualify their common approach as set-theoretic because it relies on a temporal domain defined as an ordered set, it builds granularities by grouping subsets of this domain, and it expresses their properties through set relations and operations over sets. In the area of knowledge representation and reasoning, the addition of a notion of time granularity to knowledge-based systems has been one of the most effective attempts at dealing with the widely recognized problem of managing periodic phenomena. Two relevant set-theoretic ap-

proaches to time granularity are the formalism of collection expressions proposed by Leban et al. [Leban *et al.*, 1986] and the formalism of slice expressions developed by Niezette and Stevenne [Niézette and Stevenne, 1992]. In the database area, time granularity emerged as a formal tool to deal with the intrinsic characteristics of calendars in a principled way. The set-theoretic approach to time granularity was originally proposed by Clifford and Rao [Clifford and Rao, 1988] as a suitable way of structuring information with a temporal dimension, independently of any particular calendric system, and, later, it has been systematically explored by Bettini et al. in a series of papers [Bettini *et al.*, 1998a; Bettini *et al.*, 1998b; Bettini *et al.*, 1996; Bettini *et al.*, 1998c; Bettini *et al.*, 1998d; Bettini *et al.*, 2000]. As a matter of fact, the set-theoretic framework developed by Bettini et al. subsumes all the other ones. In the following, we shall briefly describe its distinctive features. A comprehensive presentation of it is given in [Bettini *et al.*, 2000]

### 3.3.1 Granularities

The basic ingredients of the set-theoretic approach to time granularity have been outlined in Clifford and Rao's work. Even though the point of view of the authors has been largely revised and extended by subsequent work, most of their original intuitions have been preserved.

The temporal structure they propose is a temporal universe consisting of a finite, totally ordered set of temporal domains built upon some base discrete, totally ordered, infinite set which represents the smallest observable/interesting time units.

Let $T^0$ be the chosen base temporal domain. A temporal universe $\mathcal{T}$ is a finite sequence $\langle T^0, T^1, \ldots, T^n \rangle$ such that, for $i, j = 0, 1, \ldots, n$, if $i \neq j$, then $T^i \cap T^j = \emptyset$, and, for $i = 0, 1, \ldots, n-1$, $T^{i+1}$ is a constructed intervallic partition of $T^i$. We say that $T^{i+1}$ is a constructed intervallic partition of $T^i$ if there exists a mapping $\psi_i^{i+1} : T^{i+1} \rightarrow 2^{T^i}$ which satisfies the following two properties: (i) $\psi_i^{i+1}(x)$ is a (finite) convex subset of $T^i$ (convexity) , and (ii) $\bigcup_{x \in T^{i+1}} \psi_i^{i+1}(x) = T^i$ (total covering). If we add the conditions that, for each $x \in T^{i+1}$, $\psi_i^{i+1}(x) \neq \emptyset$ and, for every pair $x, y \in T^{i+1}$, with $x \neq y$,$\psi_i^{i+1}(x) \cap \psi_i^{i+1}(y) = \emptyset$, the temporal domain $T^{i+1}$, under the mapping $\psi_i^{i+1}$, can be viewed as a partition of $T^i$. Furthermore, the resulting mapping $\psi_i^{i+1}$ allows us to inherit a total order of $T^{i+1}$ from the total order of $T^i$ as follows (order-preservation). Given a finite closed interval $S$ of $T^i$, let $first(S)$ and $last(S)$ be respectively the first and the last element of $S$ with respect to the total order of $T^i$. A total order of $T^{i+1}$ can be obtained by stating that, for all $x, y \in T^{i+1}$, $x < y$ if and only if $last(\psi_i^{i+1}(x)) < first(\psi_i^{i+1}(y))$.

In [Bettini *et al.*, 1998c; Bettini *et al.*, 1998d; Bettini *et al.*, 2000], Bettini et al. have generalized that simple temporal structure for time granularity. The framework they propose is based on a *time domain* $\langle T, \leq \rangle$, that is, a totally ordered set, which can be dense or discrete. A *granularity* $g$ is a function from an index set $I_g$ to the powerset of $T$ such that:

$$\forall i, j, k \in I_g (i < k < j \wedge g(i) \neq \emptyset \wedge g(j) \neq \emptyset \Rightarrow g(k) \neq \emptyset) \qquad \text{(conservation)}$$

$$\forall i, j \in I_g (i < j \Rightarrow \forall x \in g(i)\ \forall y \in g(j)\ x < y) \qquad \text{(order preservation)}$$

Typical examples of granularities are the business weeks which map week numbers to sets of five days (from Monday to Friday) and ignore completely Saturday and Sunday. $I_g$ can be any discrete ordered set. However, for practical reasons, and without loss of generality, we shall consider below that it is either $\mathbb{N}$ or an interval of $\mathbb{N}$.

The *origin* of a granularity is $g_0 = g(min_<(I_g))$ and its *anchor* is $a \in g_0$ such that $\forall x \in g_0(a \leq x)$. The *image* of a granularity $g$ is $Im(g) = \cup_{i \in I_g} g(i)$ and its *extent* is $Ext(g) = \{x \in T : \exists a, b \in Im(g)(a \leq x \leq b)\}$. Two granules $g(i)$ and $g(j)$ are said to be *contiguous* if and only if $\nexists x \in T(g(i) \leq x \leq g(j))$.

### 3.3.2 Relations between granularities

One of the important aspects of the work by Bettini et al. is the definition of many different relationships between granularities:

$$g \trianglelefteq h \equiv \forall j \in I_h, \exists S \subseteq I_g(h(j) = \cup_{i \in S} g(i)) \qquad (g \text{ groups into } h)$$

$$g \preceq h \equiv \forall i \in I_g, \exists j \in I_h(g(i) \subseteq h(j)) \qquad (g \text{ is finer than } h)$$

$$g \sqsubseteq h \equiv \forall i \in I_g, \exists j \in I_h(g(i) = h(j)) \qquad (g \text{ is a subgranularity of } h)$$

$$g \leftrightarrow h \equiv \exists k \in \mathbb{N}\, \forall i \in I_g(g(i) = h(i+k)) \qquad (g \text{ is shift-equivalent to } h)$$

$$g \trianglelefteq h \text{ and } g \preceq h \qquad (g \text{ partitions } h)$$

$$g \widehat{\subseteq} h \equiv Im(g) \subseteq Im(h) \qquad (g \text{ is covered by } h)$$

$$g \trianglelefteq h \text{ and } \exists r, p \in \mathbb{Z}^+(r \leq |I_h| \wedge \forall i \in I_h(h(i) = \cup_{x=0}^{k} g(j_x) \wedge h(i+r) \neq \emptyset \Rightarrow h(i+r) = \cup_{x=0}^{k} g(j_x + p)))$$

($g$ groups periodically into $h$)

Apart from the case of shift-equivalence, all these definitions state, in different ways, that $g$ is a more precise granularity than $h$. As an example, the groups into relation groups together intervals of $g$. In fact, it can groups a subset of the elements within the interval, but in such a case the excluded elements cannot belong to any other granule of the less precise granularity. Finer than requires that all the granules of $g$ are covered by a granule of $h$. So $h$ can group granules of $g$, but never forget one. However, it can introduce granules that were not taken into account by $g$ (between two $g$-granules). Sub-granularity can only do exactly that (i.e., it cannot group $g$-granules). Shift-equivalence is, in spirit, the relation holding between two granularities that are equivalent up to index renaming. It is here restricted to integer increment. Partition, as we shall see below, is the easy-behaving relationship in which the less precise granularity is just a partition of the granules of the more precise one.

It is noteworthy that all these relationships consider only aligned granularities, that is, the granules of the more precise granularity are either preserved of forgotten, but never broken, in the less precise one.

These relations are ordered by strength as below.

**Proposition 3.3.1.** $\forall h, g(g \sqsubseteq h \Rightarrow g \preceq h \Rightarrow g \widehat{\subseteq} h)$

It also appears that the shift-equivalence is indeed the congruence relation induced by the subgranularity relation.

**Proposition 3.3.2.** $\forall h, g(g \leftrightarrow h$ *iff* $g \sqsubseteq h$ *and* $h \sqsubseteq g)$

It is an equivalence relation and if we consider the quotient set of granularity modulo shift-equivalence, then $\sqsubseteq$ but also $\preceq$ and $\trianglelefteq$ define partial orders (and thus partition as well) and $\widehat{\subseteq}$ is still a pre-order.

### 3.3.3 Granularity systems and calendars

For the purpose of using the granularities, it is more convenient to study granularity systems, i.e., sets of granularities related by different constraints.

A *calendar* is a set $S$ of granularities over the same time domain that includes a granularity $g$ such that $\forall h \in S(g \trianglelefteq h)$. Considering sets of granularities in which items can be converted, there are four important design choices:

**The choice of the absolute time set** $\mathcal{A}$ dense, discrete or continuous.

**Restriction on the use of the index set** if it is common to all granularities, otherwise, the restriction hold between them; the authors offer the choice between $\mathbb{N}$ or $\mathbb{N}^+$. More generally, the choice can be done among index sets isomorphic to these.

**Constraints on the granularities** no gaps within a granule, no gaps between granules, no gaps on left/right (i.e., the granularity covers the whole domain), with uniform extent.

**Constraints between granularities** which can be expressed through the above-defined relationships.

They define, as their reference granularity frame, the *General Granularity on Reals* by:

- Absolute time is the set $\mathbb{R}$;
- index set is $\mathbb{N}^+$;
- no restrictions on granules;
- no two granularities are in shift-equivalent.

Two particular units $g_\top$ and $g_\bot$ can be defined such that:

$$\forall i \in \mathbb{N}^+, g_\bot(i) = \emptyset \text{ and } g_\top(i) = \begin{cases} T & \text{if } i = 1; \\ \emptyset & \text{otherwise.} \end{cases}$$

It is shown [Bettini *et al.*, 1996] that under sensible assumptions (namely, order-preservation or convexity-contiguity-totality), the set of units is a lattice with respect to $\preceq$ in which $g_\top$ (resp. $g_\bot$) is the greatest (resp. lowest) element. In [Bettini *et al.*, 2000], it is proved that this applies to any granularity system having no two granularity shift-equivalence (i.e., $\leftrightarrow = \emptyset$). This is important because any granularity system can be quotiented by shift-equivalence.

Finally, two conversion operators on the set of granularities are defined. The *upward conversion* between granularities is defined as:

$$\forall i \in I_g, \uparrow_g^h i = \begin{cases} j & \text{if } \exists j \in I_h(g(i) \subseteq h(j)); \\ undefined & \text{otherwise.} \end{cases}$$

Notice that the upward operator is thus only defined in the aligned case expressed by the "finer than" relationship.

**Proposition 3.3.3.** *if* $g \preceq h$, *then* $\uparrow_g^h$ *is always defined.*

The *downward conversion* between granularities is defined as:

$$\forall j \in I_h, \downarrow_g^h j = \begin{cases} \langle i, k \rangle & \text{if } h(j) = \cup_{x=i}^{i+k-1} g(x); \\ undefined & \text{otherwise.} \end{cases}$$

The result is thus the set of elements covered by $h(j)$. Obviously, here the "groups into" relation between the granularities ensures the totality of the downward conversion.

**Proposition 3.3.4.** *if $g \preceq h$, then $\downarrow_g^h$ is always defined.*

### 3.3.4 Algebra for generating granularities

As it is usual in the database tradition, the authors investigate the many ways in which granularities can be freely generated by applying operations to other granularities. This can be used for defining the free generated system from a set of base granularities over the same temporal domain and a set of operations. With these operations will naturally come corresponding conversion operators.

Two set of operations are identified: grouping (or group-oriented) operations, which create a granularity by grouping granules of another granularity, and selection (or granule-oriented) operations, which create a granularity by selecting granules of another granularity.

These operations are informally described below. Interested readers must refer to [Bettini *et al.*, 2000] which adds new notions (label-aligned subgranularities) for facilitating their introduction.

Grouping operations are the following:

$group_m(g)$ groups $m$ granules of a granularity $g$ into one granule of granularity $group_m(g)$;

$alter_{l,k}^m(g, g')$ modifies granularity $g$ such that any $l^{th}$ granule having $k$ additional granules of $g'$ ($g'$ must partition $g$, $k$ can be negative);

$shift_m(g)$ creates a granularity shift-equivalent to $g$ modulo $m$;

$combine(g, h)$ creates a new granularity whose granules group granules of $h$ belonging to the same granule of $g$;

$anchor - group(g, h)$ creates a new granularity by adding to each granule of $h$ all following granules of $g$ before the next granule of $h$.

Selection operations are the following:

$subset_m^n(g)$ selects the granules of $g$ whose index are between $m$ and $n$;

$select - up(g, h)$ selects the granules of $g$ that contain at least one granule of $h$;

$select - down_k^l(g, h)$ selects the $l$ granules of $g$ starting with the $k^{th}$ in each granule of $h$;

$select - by - intersect_k^l(g, h)$ selects the $k$ granules of $g$ starting with the $l^{th}$ in each ordered set of granules intersecting any granule of $h$;

$union(g, h)$, $intersection(g, h)$, $difference(g, h)$ are defined as the corresponding operations on the set of granules of two subgranularities of the same reference granularity.

A consequence of the choice of these operations is that the operators never create finer granularities from coarser ones (they either group granules for a coarser granularity or select a subset of the granules of one existing granularity). This can be applied, for instance, generating many granularities starting with the `second` granularity (directly inspired from [Bettini *et al.*, 2000]):

$$
\begin{aligned}
\texttt{minute} &= group_{60}(\texttt{second}) \\
\texttt{hour} &= group_{60}(\texttt{minute}) \\
\texttt{USEasthour} &= shift_{-5}(\texttt{hour}) \\
\texttt{day} &= group_{24}(\texttt{hour}) \\
\texttt{week} &= group_{7}(\texttt{day}) \\
\texttt{busi-day} &= select - down_1^5(\texttt{day}, \texttt{week}) \\
\texttt{month} &= alter^{12*400}_{2+12*399,1}(\texttt{day}, alter^{12*100}_{2+12*99,-1}(\texttt{day}, alter^{12*4}_{2+12*3,1}(\texttt{day}, \\
&\quad alter^{12}_{11,-1}(\texttt{day}, alter^{12}_{9,-1}(\texttt{day}, alter^{12}_{6,-1}(\texttt{day}, alter^{12}_{4,-1}(\texttt{day}, \\
&\quad alter^{12}_{2,-3}(\texttt{day}, group_{31}(\texttt{day}))))))) \\
\texttt{year} &= group_{12}(\texttt{month}) \\
\texttt{academicyear} &= anchor - group(\texttt{day}, \\
&\quad select - by - intersect_1^1(\texttt{busi-day}, select - down_9^1(\texttt{month}))
\end{aligned}
$$

As a matter of fact, these granularities can be generated in a more controlled way. Indeed, the authors distinguish three layers of granularities:

$L_1$ containing the bottom granularity and all the granularities obtained by applying *group*, *alter*, and *shift* on granularities of this layer;

$L_2$ including $L_1$ and containing all the granularities obtained by applying *subset*, *union*, *intersection*, and *difference* on granularities of this layer and selections with first operand belonging to this layer;

$L_3$ including $L_2$ and containing all the granularities obtained by applying *combine* on granularities of this layer and *anchor* − *group* with the second operand on granularities of this layer.

Granularities of $L_1$ are full-integer labelled granularities, those of $L_2$ may not be labelled by all integers, but they contain no gaps within granules. These aspects, as well as the expressiveness of the generated granularities, are investigated in depth in [Bettini *et al.*, 2000].

### 3.3.5 Constraint solving and query answering

Wang et al. [Wang *et al.*, 1995] have proposed an extension of the relational data model which is able to handle granularity. The goal of this work is to take into account possible granularity mismatch in the context of federated databases.

An extended temporal model is a relational database in which each tuple is timestamped under some granularity. Formally, it is a set of tables such that each table is a quadruple

$\langle R, \phi, \tau, g\rangle$ such that $R$ is a set of tuples (a relational table), $g$ is a granularity, $\phi : \mathbb{N} \longrightarrow 2^R$ maps granules to tuples, $\tau : R \longrightarrow 2^{\mathbb{N}}$ maps tuples to granules such that $\forall t \in R, t \in \phi(i) \Rightarrow i \in \tau(t)$ and $\forall i \in \mathbb{N}, i \in \tau(t) \Rightarrow t \in \phi(i)$.

In [Bettini *et al.*, 2000], the authors develop methods for answering queries in database with granularities. The answers are computed with regard to hypotheses tied to the databases. These hypotheses allow the computation of values between two successive timestamps. The missing values can, for instance, be considered constant (persistence) or interpolated with a particular interpolation function. These hypotheses also apply to the computation of values between granularity.

The hypotheses ($H$) provide the way to compute the closure ($\overline{D}^H$) of a particular database ($D$). Answering a query $q$ against a database with granularities $D$ and hypotheses $H$ consists in answering the query against the closure of the database ($\overline{D}^H \models q$). Instead of computing this costly closure, the authors proposes to *reduce* the database with regard to the hypotheses (i.e., to find the minimal database equivalent to the initial one modulo closure) and to add to the query formulas allowing the computation of the hypotheses.

The authors also define quantitative temporal constraint satisfaction problems under granularity whose variables correspond to points and arcs are labelled by an integer interval and a granularity. A pair of points $\langle t, t'\rangle$ satisfies a constraint $[m, n]g$ (with $m, n \in \mathbb{Z}$ and $g$ a granularity) if and only if $\uparrow^g t$ and $\uparrow^g t'$ are defined and $m \leq \mid \uparrow^g t - \uparrow^g t' \mid \leq n$. These constraints cannot be expressed as a classical TCSP (see Chapter 7). As a matter of fact, if the constraint [0 0] is set on two entities under the hour granularity, two points satisfy it if they are in the same hour. In terms of seconds, the positions should differ from 0 to 3600. However, [0 3600] under the second granularity does not corresponds to the original constraint since it can be satisfied by two points in different hours.

The satisfaction problem for granular constraint satisfaction is NP-hard (while STP is polynomial) [Bettini *et al.*, 1996]. Indeed the modulo operation involved in the conversions can introduce disjunctive constraints (or non convexity). For instance, next business day is the convex constraint ([1 1]), which converted in hours can yield the constraint [1 24]$\vee$[49 72] which is dependent on the exact day of the week.

The authors propose an arc-consistency algorithm complete for consistency checking when the granularities are periodical with regard to some common finer granularity. They also propose an approximate (i.e., incomplete) algorithm by iterating the saturation of the networks of constraints expressed under the same granularity and then converting the new values into the other granularities.

The work described above mainly concerns aligned systems of granularity (i.e., systems in which the upward conversion is always defined). This is not always the case, as the week/month example illustrates it. Non-aligned granularity has been considered by several authors. Dyreson and collaborators [Dyreson and Snodgrass, 1994] define comparison operators across granularities and their semantics (this covers the extended comparators of [Wang *et al.*, 1995]): comparison between entities of different granularities can be considered under the coarser granularity (here coarser is the same as "groups into" above and thus requires alignment) or the finer one. They define upward and downward conversion operators across comparable granularities and the conversion across non-comparable granularities is carried out by first converting down to the greatest lower bound and then up (assuming the greatest lower bound exists and thus that the structure is a lower semi-lattice): $\downarrow^g_{g\wedge g'} \uparrow^{g'}_{g\wedge g'} x$. Comparisons across granularities (with both semantics) are implemented in terms of the

conversion operators.

### 3.3.6 Alternative accounts of time granularity

The set-theoretic approach has been recently revisited and extended in several directions. In the following, we briefly summarize the most promising ones.

An alternative string-based *m*odel for time granularities has been proposed by Wijsen [Wijsen, 2000]. It models (infinite) granularities as (infinite) words over an alphabet consisting of three symbols, namely, ■ (filler), □ (gap), and ≀ (separator), which are respectively used to denote time points covered by some granule, to denote time points not covered by any granule, and to delimit granules. Wijsen focuses his attention on (infinite) periodical granularities, that is, granularities which are left bounded and, ultimately, periodically groups time points of the underlying temporal domain. Periodical granularities can be identified with ultimately periodic strings, and they can be finitely represented by specifying a (possibly empty) finite prefix and a finite repeating pattern. As an example, the granularity BusinessWeek ■■■■■□□ ≀ ■■■■■□□ ≀ ... can be encoded by the empty prefix $\varepsilon$ and the repeating pattern ■■■■■□□≀. Wijsen shows how to use the string-based model to solve some fundamental problems about granularities, such as the equivalence problem (to establish whether or not two given representations define the same granularity) and the minimization problem (to compute the most compact representation of a granularity). In particular, he provides a straightforward solution to the equivalence problem that takes advantage of a suitable *aligned form* of strings. Such a form forces separators to occur immediately after an occurrence of ■, thus guaranteeing a one-to-one correspondence between granularities and strings.

The idea of viewing time granularities as ultimately periodic strings establishes a natural connection with the field of formal languages and automata. An automaton-based approach to time granularity has been proposed by Dal Lago and Montanari in [Dal Lago and Montanari, 2001], and later revisited by Bresolin et al. in [Bresolin *et al.*, 2004; Dal Lago *et al.*, 2003a; Dal Lago *et al.*, 2003b]. The basic idea underlying such an approach is simple: we take an automaton $\mathcal{A}$ recognizing a *single* ultimately periodic word $u \in \{\square, \blacksquare, \blacktriangleleft\}^\omega$ and we say that $\mathcal{A}$ represents the granularity $G$ if and only if $u$ represents $G$. The resulting framework views granularities as strings generated by a specific class of automata, called Single-String Automata (SSA), thus making it possible to (re)use well-known results from automata theory. In order to compactly encode the redundancies of the temporal structures, SSA are endowed with counters ranging over discrete finite domains (Extended SSA, ESSA for short). Properties of ESSA have been exploited to efficiently solve the equivalence and the granule conversion problems for single time granularities [Dal Lago *et al.*, 2003b]. The relationships between ESSA and Calendar Algebra have been systematically investigated by Dal Lago et al. in [Dal Lago *et al.*, 2003a], where a number of algorithms that map Calendar Algebra expressions into automaton-based representations of time granularities are given. Such an encoding allows one to reduce problems about Calendar Algebra expressions to equivalent problems for ESSA. More generally, the operational flavor of ESSA suggests an alternative point of view on the role of automaton-based representations: besides a formalism for the direct specification of time granularities, automata can be viewed as a low-level formalism into which high-level time granularity specifications, such as those of Calendar Algebra, can be mapped. This allows one to exploit the benefits of both formalisms, using a high level language to define granularities and their properties in a natural and flexible

way, and the automaton-based one to efficiently reason about them. Finally, a generalization of the automaton-based approach from single periodical granularities to (possibly infinite) sets of granularities has been proposed by Bresolin et al. in [Bresolin *et al.*, 2004]. To this end, they identify a proper subclass of Büchi automata, called Ultimately Periodic Automata (UPA), that captures regular sets consisting of only ultimately periodic words. UPA allow one to encode single granularities, (possibly infinite) sets of granularities which have the same repeating pattern and different prefixes, and sets of granularities characterized by a finite set of non-equivalent patterns, as well as any possible combination of them.

The choice of Propositional Linear Temporal Logic (Propositional LTL) as a logical tool for granularity management has been recently advocated by Combi et al. in [Combi *et al.*, 2004]. Time granularities are defined as models of Propositional LTL formulas, where suitable propositional symbols are used to mark the endpoints of granules. In this way, a large set of regular granularities, such as, for instance, repeating patterns that can start at an arbitrary time point, can be captured. Moreover, problems like checking the consistency of a granularity specification or the equivalence of two granularity expressions can be solved in a uniform way by reducing them to the validity problem for Propositional LTL, which is known to be in PSPACE. An extension of Propositional LTL that replaces propositional variables by first-order formulas defining integer constraints, e.g., $x \equiv_k y$, has been proposed by Demri in [Demri, 2004]. The resulting logic, denoted by PLTL$^{mod}$(Past LTL with integer periodicity constraints), generalizes both the logical framework proposed by Combi et al. and the automaton-based approach of Dal Lago and Montanari, and it allows one to compactly define granularities as periodicity constraints. In particular, the author shows how to reduce the equivalence problem for ESSA to the model checking problem for PLTL$^{mod}$(-automata), which turns out to be in PSPACE, as in the case of Propositional LTL. The logical approach to time granularity is systematically analyzed in the next section, where various temporal logics for time granularity are presented.

## 3.4 The logical approach

A first attempt at incorporating time granularity into a logical formalism is outlined in [Corsetti *et al.*, 1991a; Corsetti *et al.*, 1991b]. The proposed logical system for time granularity has two distinctive features. On the one hand, it extends the syntax of temporal logic to allow one to associate different granularities (temporal domains) with different subformulas of a given formula; on the other hand, it provides a set of translation rules to rewrite a subformula associated with a given granularity into a corresponding subformula associated with a finer granularity. In such a way, a model of a formula involving different granularities can be built by first translating everything to the finest granularity and then by interpreting the resulting (flat) formula in the standard way.

A major problem with such a method is that there exists no a standard way to define the meaning of a formula when moving from a time granularity to another one. Thus, more information is needed from the user to drive the translation of the (sub)formulas. The main idea is that when we state that a predicate $p$ holds at a given time point $x$ belonging to the temporal domain $T$, we mean that $p$ holds in a subset of the interval corresponding to $x$ in the finer domain $T'$. Such a subset can be the whole interval, a scattered sequence of smaller intervals, or even a single time point. For instance, saying that "the light has been switched on at time $x_{min}$", where $x_{min}$ belong to the domain of minutes, may correspond to state

that a predicate *switching_on* is true at the minute $x_{min}$ and exactly at one second of $x_{min}$. Instead, saying that an employee works at the day $x_d$ generally means that there are several minutes, during the day $x_d$, where the predicate *work* holds for the employee. These minutes are not necessarily contiguous. Thus, the logical system must provide the user with suitable tools that allow him to qualify the subset of time intervals of the finer temporal domain that correspond to the given time point of the coarser domain.

A substantially different approach is proposed in [Ciapessoni *et al.*, 1993; Montanari, 1994; Montanari, 1996], where Montanari et al. show how to extend syntax and semantics of temporal logic to cope with metric temporal properties possibly expressed at different time granularities. The resulting metric and layered temporal logic is described in detail in Subsection 3.4.1. Its distinctive feature is the coexistence of three different operators: a contextual operator, to associate different granularities with different (sub)formulas, a displacement operator, to move within a given granularity, and a projection operator, to move across granularities.

An alternative logical framework for time granularity has been developed in the classical logic setting [Montanari, 1996; Montanari and Policriti, 1996; Montanari *et al.*, 1999]. It imposes suitable restrictions to languages and structures for time granularity to get decidability. From a technical point of view, it defines various theories of time granularity as suitable extensions of monadic second-order theories of $k$ successors, with $k \geq 1$. Monadic theories of time granularity are the subject of Subsection 3.4.2.

The temporal logic counterparts of the monadic theories of time granularity, called temporalized logics, are briefly presented in Subsection 3.4.3. This way back from the classical logic setting to the temporal logic one passes through an original class of automata, called temporalized automata.

A coda about the relationships between logics for time granularity and interval temporal logics concludes the section.

### 3.4.1 A metric and layered temporal logic for time granularity

Original metric and layered temporal logics for time granularity have been proposed by Montanari et al. in [Ciapessoni *et al.*, 1993; Montanari, 1994; Montanari, 1996]. We introduce these logics in two steps. First, we take into consideration their purely metric fragments in isolation. To do that, we adopt the general two-sorted framework proposed in [Montanari, 1996; Montanari and de Rijke, 1997], where a number of metric temporal logics, having a different expressive power, are defined as suitable combinations of a temporal component and an algebraic one. Successively, we show how flat metric temporal logic can be generalized to a many-layer metric temporal logic, embedding the notion of time granularity [Montanari, 1994; Montanari, 1996]. We first identify the main functionalities a logic for time granularity must support and the constraints it must satisfy; then, we axiomatically define metric and layered temporal logic, viewed as the combination of a number of differently-grained (single-layer) metric temporal logics, and we briefly discuss its logical properties.

#### The basic metric component

The idea of a logic of positions (topological, or metric, logic) was originally formulated by Rescher and Garson [Rescher and Garson, 1968; Rescher and Urquhart, 1971]. In [Rescher

and Garson, 1968], the authors define the basic features of the logic and they show how to give it a temporal interpretation. Roughly speaking, metric (temporal) logic extends propositional logic with a parameterized operator $\Delta_\alpha$ of positional realization that allows one to constrain the truth value of a proposition at position $\alpha$. If we interpret the parameter $\alpha$ as a displacement with respect to the current position, which is left implicit, we have that $\Delta_\alpha q$ is true at a position $x$ if and only if $q$ is true at a position $y$ at distance $\alpha$ from $x$. Metric temporal logics can thus be viewed as two-sorted logics having both formulas and parameters; formulas are evaluated at time points while parameters take values in a suitable algebraic structure of temporal displacements. In [Montanari and de Rijke, 1997], Montanari and de Rijke start with a very basic system of metric temporal logic, and they build on it by adding axioms and/or by enriching the underlying structures. In the following, we describe the metric temporal logic of two-sorted frames with a linear temporal order (*MTL*); we also briefly consider general metric temporal logics allowing quantification over algebraic and temporal variables and free mixing of algebraic and temporal formulas (*Q-MTL*).

The *two-sorted temporal language* for *MTL* has two components: the algebraic component and the temporal one. Given a non-empty set $A$ of constants, let $T(A)$ be the set of terms over $A$, that is, the smallest set such that $A \subseteq T(A)$, and if $\alpha, \beta \in T(A)$ then $\alpha + \beta$, $-\alpha, 0 \in T(A)$. The first-order (algebraic) component is built up from $T(A)$ and the predicate symbols $=$ and $<$. The temporal component of the language is built up from a non-empty set $\mathcal{P}$ of proposition letters. The set of formulas over $\mathcal{P}$ and $A$, $F(\mathcal{P}, A)$, is the smallest set such that $\mathcal{P} \subseteq F(\mathcal{P}, A)$, and if $\phi, \psi \in F(\mathcal{P}, A)$ and $\alpha \in T(A)$, then $\neg\phi$, $\phi \wedge \psi$, $\top$ (true), $\bot$ (false), and $\Delta_\alpha \phi$ (and its dual $\nabla_\alpha \phi := \neg\Delta_\alpha \neg\phi$) belong to $F(\mathcal{P}, A)$. $\Delta_\alpha$ is called the (parameterized) *displacement operator*.

A *two-sorted frame* is a triple $\mathbf{F} = (T, \mathbf{D}; \mathrm{DIS})$, where $T$ is the set of (time) points over which temporal formulas are evaluated, $\mathbf{D}$ is the algebra of metric displacements in whose domain $D$ terms take their values, and $\mathrm{DIS} \subseteq T \times D \times T$ is an accessibility relation, called displacement relation, relating pairs of points and displacements. The components of two-sorted frames satisfy the following properties. First, $\mathbf{D}$ is an ordered Abelian group, that is, a structure $\mathbf{D} = (D, +, -, 0, <)$, where $+$ is a binary function of displacement composition, $-$ is a unary function of inverse displacement, and 0 is the zero displacement constant, such that:

| | | |
|---|---|---|
| $(i)$ | $\alpha + \beta = \beta + \alpha$ | (commutativity of $+$); |
| $(ii)$ | $\alpha + (\beta + \gamma) = (\alpha + \beta) + \gamma$ | (associativity of $+$); |
| $(iii)$ | $\alpha + 0 = \alpha$ | (zero element of $+$); |
| $(iv)$ | $\alpha + (-\alpha) = 0$ | (inverse), |

and $<$ is an irreflexive, asymmetric, transitive, and linear relation that satisfies the comparability property (*viii*) below:

| | |
|---|---|
| $(v)$ | $\neg(\alpha < \alpha)$; |
| $(vi)$ | $\neg(\alpha < \beta \wedge \beta < \alpha)$; |
| $(vii)$ | $\alpha < \beta \wedge \beta < \gamma \rightarrow \alpha < \gamma$; |
| $(viii)$ | $\alpha < \beta \vee \alpha = \beta \vee \beta < \alpha$. |

Furthermore, there are two conditions expressing the relations between $+$ and $-$, and $<$:

| | |
|---|---|
| $(ix)$ | $\alpha < \beta \rightarrow \alpha + \gamma < \beta + \gamma$; |
| $(x)$ | $\alpha < \beta \rightarrow -\beta < -\alpha$. |

As for the displacement relation, we first require DIS to respect the converse operation of the Abelian group in the following sense:

Symmetry: $\forall i, j, \alpha\,(\mathrm{DIS}(i, \alpha, j) \rightarrow \mathrm{DIS}(j, -\alpha, i))$.

Furthermore, we require DIS to be reflexive, transitive, quasi-functional (q-functional for short) with respect to both its third and second argument, and totally connected:

Reflexivity: $\forall i\,\mathrm{DIS}(i, 0, i)$;
Transitivity: $\forall i, j, k, \alpha, \beta\,(\mathrm{DIS}(i, \alpha, j) \wedge \mathrm{DIS}(j, \beta, k) \rightarrow \mathrm{DIS}(i, \alpha + \beta, k))$;
Q-functionality - 1: $\forall i, j, j', \alpha\,(\mathrm{DIS}(i, \alpha, j) \wedge \mathrm{DIS}(i, \alpha, j') \rightarrow j = j')$;
Q-functionality - 2: $\forall i, j, \alpha, \beta\,(\mathrm{DIS}(i, \alpha, j) \wedge \mathrm{DIS}(i, \beta, j) \rightarrow \alpha = \beta)$;
Total connectedness: $\forall i, j \exists \alpha\,\mathrm{DIS}(i, \alpha, j)$.

From the ordering $<$ on the algebraic component of the frames, an ordering $\ll$ on the temporal component can be defined as follows:

$$i \ll j \text{ iff for some } \alpha > 0, \mathrm{DIS}(i, \alpha, j). \tag{3.1}$$

According to Definition 3.1, we have that $i$ and $j$ are $\ll$-related if there exists a positive displacement between them. It is possible to show that $\ll$ is a strict linear order [Montanari and de Rijke, 1997] (it is worth noting that, without the properties of quasi-functionality with respect to the second argument and total connectedness, Definition 3.1 does not produce a strict linear order).

The interpretation of the language for *MTL* on two-sorted frames based on an ordered Abelian group is fairly straightforward: the first-order (algebraic) component is interpreted on the ordered Abelian group, and the temporal component on the temporal domain. Basically, a two-sorted frame $\mathbf{F}$ can be turned into a *two-sorted model* by adding an interpretation for the algebraic terms and a valuation for proposition letters. An interpretation for algebraic terms is given by a function $g : A \rightarrow D$ that is automatically extended to all terms from $T(A)$. A valuation is simply a function $V : \mathcal{P} \rightarrow 2^T$. We say that $\alpha = \beta$ (resp. $\alpha < \beta$) is *true* in a model $\mathbf{M} = (T, \mathbf{D}; \mathrm{DIS}; V, g)$ whenever $g(\alpha) = g(\beta)$ (resp. $g(\alpha) < g(\beta)$). *Truth* of temporal formulas is defined by means of the standard semantic clauses for proposition letters and Boolean connectives, plus the following clause for the displacement operator:

$$\mathbf{M}, i \Vdash \Delta_\alpha \phi \quad \text{iff} \quad \text{there exists } j \text{ such that } \mathrm{DIS}(i, g(\alpha), j) \text{ and } \mathbf{M}, j \Vdash \phi.$$

Let $\Gamma$ denote a set of formulas. To avoid messy complications we only consider one-sorted consequences $\Gamma \models \phi$; for algebraic formulas '$\Gamma \models \phi$' means 'for all models $\mathbf{M}$, if $\mathbf{M} \models \Gamma$, then $\mathbf{M} \models \phi$'; for temporal formulas it means 'for all models $\mathbf{M}$, and time points $i$, if $\mathbf{M}, i \Vdash \Gamma$, then $\mathbf{M}, i \Vdash \phi$'.

The following example shows that the language of *MTL* allows one to express meaningful temporal conditions.

**Example 3.4.1.** *Let us consider a communication channel C that collects messages from $n$ different sources $S_1, \ldots, S_n$ and outputs them with delay $\delta$. To exclude that two input events can occur simultaneously, we add the constraint (notice that preventing input events from occurring simultaneously also guarantees that output events do not occur simultaneously):*

$$\forall i, j\, \neg(\mathtt{in}(i) \wedge \mathtt{in}(j) \wedge i \neq j),$$

*which is shorthand for:*

$$\neg(\mathtt{in}(1) \wedge \mathtt{in}(2)) \wedge \ldots \wedge \neg(\mathtt{in}(n-1) \wedge \mathtt{in}(n)).$$

*The behavior of C is specified by the formula:*

$$\forall i\,(\mathtt{out}(i) \leftrightarrow \Delta_{-\delta}\mathtt{in}(i)),$$

*which is shorthand for a finite conjunction.*

Validity in *MTL* can be axiomatized as follows. For the displacement component, one takes the axioms and rules of identity, ordered Abelian groups, and strict linear order, together with any complete calculus for first-order logic. For the temporal component, one takes the usual axioms of propositional logic plus the axioms:

| | | |
|---|---|---|
| (AxND) | $\nabla_\alpha(p \rightarrow q) \rightarrow (\nabla_\alpha p \rightarrow \nabla_\alpha q)$ | (normality); |
| (AxSD) | $p \rightarrow \nabla_\alpha \Delta_{-\alpha} p$, | (symmetry); |
| (AxRD) | $\nabla_0 p \rightarrow p$ | (reflexivity); |
| (AxTD) | $\nabla_{\alpha+\beta} p \rightarrow \nabla_\alpha \nabla_\beta p$ | (transitivity); |
| (AxQD) | $\Delta_\alpha p \rightarrow \nabla_\alpha p$ | (q-functionality - 1). |

Its rules are modus ponens and

| | | |
|---|---|---|
| (D-NEC) | $\vdash \phi \implies \vdash \nabla_\alpha \phi$ | (necessitation rule for $\nabla_\alpha$); |
| (REP) | $\vdash \phi \leftrightarrow \psi \implies \vdash \chi(\phi/p) \leftrightarrow \chi(\psi/p)$ | (replacement), |
| | where $(\phi/p)$ denotes substitution of $\phi$ for the variable $p$; | |
| (LIFT) | $\vdash \alpha = \beta \implies \vdash \nabla_\alpha \phi \leftrightarrow \nabla_\beta \phi$ | (transfer of identities). |

Axiom (AxN) is the usual distribution axiom; axiom (AxS) expresses that a displacement $\alpha$ is the converse of a displacement $-\alpha$; axioms (AxR), (AxT), and (AxQ) capture reflexivity, transitivity, and quasi-functionality with respect to the third argument, respectively. A suitable adaptation of two truth preserving constructions from standard modal logic to the *MTL* setting allows one to show there are no *MTL* formulas that express total connectedness and quasi-functionality with respect to the second argument of the displacement relation [Montanari and de Rijke, 1997]. The rules (D-NEC) and (REP) are familiar from modal logic. Finally, the rule (LIFT) allows one to transfer provable algebraic identities from the displacement domain to the temporal one.

A *derivation in MTL* is a sequence of formulas $\sigma_1, \ldots, \sigma_n$ such that each $\sigma_i$, with $1 \leq i \leq n$, is either an axiom or obtained from $\sigma_1, \ldots, \sigma_{n-1}$ by applying one of the derivation rules of *MTL*. We write $\vdash_{MTL} \sigma$ to denote that there is a derivation in *MTL* that ends in $\sigma$. It immediately follows that $\vdash_{MTL} \alpha = \beta$ iff $\alpha = \beta$ is provable from the axioms of the algebraic component only: whereas we can lift algebraic information from the displacement domain to the temporal domain using the (LIFT) rule, there is no way in which we can import temporal information into the displacement domain. As with consequences, we only consider one-sorted inferences '$\Gamma \vdash \phi$'.

**Theorem 3.4.1.** *MTL is sound and complete for the class of all transitive, reflexive, totally-connected, and quasi-functional (in both the second and third argument of their displacement relation) frames.*

The proof of soundness is trivial. The completeness proof is much more involved [Montanari and de Rijke, 1997]. It is accomplished in two steps: first, one proves completeness with respect to totally connected frames via same sort of generated submodel construction; then, a second construction is needed to guarantee quasi-functionality with respect to the second argument.

Propositional variants of *MTL* are studied in [Montanari and de Rijke, 1997]. As an example, one natural specialization of *MTL* is obtained by adding discreteness. As in the case of the ordering, the discreteness of the temporal domain necessarily follows from that of the domain of temporal displacements, which is expressed by the following formula:

$$\forall\alpha\,\exists\beta,\beta'(\alpha<\beta\wedge\forall\gamma(\alpha<\gamma\rightarrow(\beta=\gamma\vee\beta<\gamma))\wedge$$
$$\beta'<\alpha\wedge\forall\delta(\delta<\alpha\rightarrow(\beta'=\delta\vee\beta'<\delta)))$$

**Proposition 3.4.1.** *Let* $\mathbf{F}=(T,\mathbf{D};\mathrm{DIS})$ *be a two-sorted frame based on a discrete ordered Abelian group* $\mathbf{D}$. *For all* $i,j\in T$, *there exist only finitely many* $k$ *such that* $i\ll k\ll j$.

For some applications, both *MTL* and its propositional variants are not expressive enough, and thus they must be extended. In particular, they lack quantification and constrain displacements to occur as parameters of the displacement operator only. The following example shows how the ability of freely mixing temporal and displacement formulas enables one to exploit more complex ways of interaction between the two domains, rather than to only lift information from the algebraic domain to the temporal one.

**Example 3.4.2.** *Let us consider the operation of a traffic light controller C [Henzinger* et al., *1994]. When the request button is pushed, the controller makes a pedestrian light turn green within a given time bound after which the light remains green for a certain amount of time. Moreover, assume that C takes a unit of time to switch the light and that the time needed for its internal operations is negligible.*

*We require that C satisfies the following conditions:*

*(i) whenever a pedestrian pushes the request button ('request is true'), then the light is green within 5 time units and remains green for at least 10 time units (this condition guarantees that no pedestrian waits for more than 5 time units, and that he or she is given at least 10 time units to cross the road);*

*(ii) whenever request is true, then it is false within 20 time units (this condition ensures that the request button is reset);*

*(iii) whenever request has been false for 20 time units, the light is red (this condition should prevent the light from always being green).*

*By taking advantage of the possibility of quantifying displacement variables and of using displacement formulas, the behavior of C can be specified by the conjunction of the following formulas:*

$$\begin{aligned}&\text{request}\ \rightarrow\ \exists x(0<x\leq 5\wedge\forall y(x\leq y<x+10\rightarrow\nabla_y\text{lightIsGreen}));\\&\text{request}\ \rightarrow\ \exists z(0\leq z\leq 20\wedge\Delta_z\neg\,\text{request});\\&\forall x(0\leq x<20\rightarrow\nabla_x\neg\text{request})\rightarrow\nabla_{20}\text{lightIsRed},\end{aligned}$$

*together with a formula stating that at each time point the traffic light is either red or green:*

$$\text{lightIsGreen} \leftrightarrow \neg\ \text{lightIsRed}.$$

*Different implementations of C, all satisfying the given specification, can be obtained by making different assumptions about the value of temporal parameters, e.g., by varying the delay between requests and resets. It is worth noting that, even if there are no restrictions on the frequency of requests, the above specification is appropriate only if that frequency is low; otherwise, it may happen that switching the light to red is delayed indefinitely. A solution to this problem is discussed in [Montanari, 1996].*

Systems of quantified metric temporal logic (*Q-MTL* for short) are developed in [Montanari and de Rijke, 1997]. The language of *Q-MTL* extends that of *MTL* by adding algebraic variables (and, possibly, temporal variables) and by allowing quantification over algebraic (and temporal) variables and free mixing of algebraic formulas and temporal propositional symbols. *Q-MTL* models can be obtained from ordered two-sorted frames $\mathbf{F} = (T, \mathbf{D}; \mathrm{DIS})$ by adding an interpretation function $g$ for the algebraic terms and a valuation $V$ for proposition letters, and by specifying the way one evaluates mixed formulas at time points. An axiomatic system for *Q-MTL* (we refer to the simplest system of quantified metric temporal logic; other cases are considered in [Montanari and de Rijke, 1997]) is obtained from that for *MTL* by adding a number of axiom schemata governing the behavior of quantifiers and substitutions:

(AxF) $\forall x\,(\phi \rightarrow \psi) \leftrightarrow (\forall x\,\phi \rightarrow \forall x\,\psi)$ (functionality);
(AxEVQ) $\phi \rightarrow \forall x\,\phi$, for $x$ not in $\phi$
(elimination of vacuous quantifiers);
(AxUI) $\forall x\,\phi \rightarrow \phi(\alpha/x)$, with $\alpha$ free for $x$ in $\phi$
(universal instantiation),

the Barcan formula for the displacement operator:

(AxBFD) $\forall x\,\nabla_\alpha \phi \rightarrow \nabla_\alpha \forall x\,\phi$, with $x \notin \alpha$ (Barcan formula for $\nabla_\alpha$),
where $x \notin \alpha$ stands for $x \neq \alpha$ and $x$ does not occur in $\alpha$,

the axioms relating the algebraic terms and the displacement operator (axiom (AxAD4) can actually be derived from the other axioms):

(AxAD1) $\alpha = \beta \rightarrow \forall x \nabla_x \alpha = \beta$; (AxAD2) $\alpha \neq \beta \rightarrow \forall x \nabla_x \alpha \neq \beta$;
(AxAD3) $\alpha < \beta \rightarrow \forall x \nabla_x \alpha < \beta$; (AxAD4) $\alpha \not< \beta \rightarrow \forall x \nabla_x \alpha \not< \beta$,

and the rule:

(UG) $\vdash \phi \implies \vdash \forall x\,\phi$ (universal generalization).

The completeness of *Q-MTL* can be proved by following the general pattern of the completeness proof for *MTL*, but the presence of mixed formulas complicates some of the details. Basically, it makes use of a variant of Hughes and Cresswell's method for proving axiomatic completeness in the presence of the Barcan formula [Hughes and Cresswell, 1968].

### The addition of time granularity

Metric and Layered Temporal Logic (*MLTL* for short) is obtained from *MTL* by adding a notion of time granularity [Ciapessoni *et al.*, 1993; Montanari, 1994; Montanari, 1996]. In

the following, we first show how to extend two-sorted frames to incorporate granularity; then, we present syntax, semantics, and axiomatization of *MLTL*; finally, we briefly describe the way in which the synchronization problem (cf. Section 3.2) can be dealt with in *MLTL*.

The main change to make to the model of time when moving from *MTL* to *MLTL* is the replacement of the temporal domain $T$ by a temporal universe $\mathcal{T}$ consisting of a set of *disjoint* linear temporal domains/layers, that share the same displacement domain $D$. Formally, $\mathcal{T} = \{T^i : i \in M\}$, where $M$ is an initial segment of $\mathbb{N}$, possibly equal to $\mathbb{N}$. The set $\bigcup_{i \in M} T^i$ collects all time points belonging to the different layers of $\mathcal{T}$. $\mathcal{T}$ is assumed to be *totally ordered* by the granularity relation $\prec$. As an example, if $\mathcal{T} = \{\texttt{years}, \texttt{months}, \texttt{weeks}, \texttt{days}\}$, we have that $\texttt{days} \prec \texttt{weeks} \prec \texttt{months} \prec \texttt{years}$. A finer characterization of the relations among the layers of a temporal universe is provided by the *disjointedness* relation, denoted by $\subset$, which is quite similar to the *groups-into* relation defined in Section 3.3. It defines a partial order over $\mathcal{T}$ that rules out pairs of layers like `weeks` and `months` for which a point of a finer layer (`weeks`) can be astride two points of the coarser one (`months`). As an example, given $\mathcal{T} = \{\texttt{years}, \texttt{months}, \texttt{weeks}, \texttt{days}\}$, we have that $\texttt{months} \subset \texttt{years}, \texttt{days} \subset \texttt{months}$, and $\texttt{days} \subset \texttt{weeks}$. This means that `years` are pairwise disjoint when viewed as sets of `months`; the same holds for `months` when viewed as sets of `days`.

The links between points belonging to the same layer are expressed by means of (a number of instances of) the *displacement* relation, while those between points belonging to different layers are given by means of a *decomposition* relation that, for every pair $T^i, T^j \in \mathcal{T}$, with $T^j \prec T^i$, associates each point of $T^i$ with the set of points of $T^j$ that compose it. We assume that the decomposition relation turns every point $x \in T^i$ into a set of contiguous points (decomposition interval) of $T^j$ (*convexity*). This condition excludes the presence of 'temporal gaps' within the set of components of a given point, as it happens, for instance, when `business months` are mapped on `days`. In general, the cardinalities of the sets of components of two distinct points $x, y \in T^i$ with respect to $T^j$ may be different (*non homogeneity*). This is the case, for instance, with pairs of layers like `real months` and `days`: different months can be mapped on a different number of days (28, 29, 30, or 31). In some particular contexts, however, it is convenient to work with temporal universes where, for every pair of layers $T^i, T_j$, with $T^j \prec T^i$, the decomposition intervals have the same cardinality (*homogeneity*). For instance, this is the case of temporal universes that replace `real months` by `legal months`, which, conventionally, are 30-days long. We constrain the decomposition relation to respect the ordering of points within layers (*order preservation*). If $T^j \subset T^i$, e.g., `seconds` and `minutes`, then the intervals are disjoint; otherwise, the intervals can possibly meet at their endpoints, e.g., `weeks` and `months`. We further require that the union of the intervals of $T^j$ associated with the points of $T^i$ covers the whole $T^j$ (*total covering*). From order preservation and total covering, it follows that, for all pairs of layers $T^i, T^j$, with $T^j \prec T^i$, the decomposition relation associates contiguous points of $T^i$ with contiguous sets of points of $T_j$ (*contiguity*). This excludes the presence of 'temporal gaps' between the decomposition intervals of consecutive points of the coarser layer, as it happens, for instance, when `business weeks` are mapped on `days`. Finally, we require that, for every $i, j, k$, if $T^j \subset T^k \subset T^i$, then the decomposition of $T^i$ into $T^j$ can be obtained from the decomposition of $T^i$ into $T^k$ and that of $T^k$ into $T^j$ (*downward transitivity*). The same holds for $T^k \subset T^j \subset T^i$ (*downward/upward transitivity*). In the following, we shall also consider the inverse relation of *abstraction*, that, for every pair $T^i, T^j \in \mathcal{T}$, with

$T^j \prec T^i$, associates a point $x \in T^j$ with a point $y \in T^i$ if $x$ belongs to the the decomposition of $y$ with respect to $T^j$. Every point $x \in T^j$ can be abstracted into either one or two adjacent points of $T^i$. If $T^j \subset T^i$, $x$ is abstracted into a unique point $y$, which is called the *coarse grain equivalent* of $x$ with respect to $T^i$.

Besides the algebraic and temporal components, the *temporal language* for *MLTL* includes a *context sort*. Moreover, the displacement operator is paired with a contextual operator and a projection operator. Formally, given a non-empty set $C$ of context constants, denoting the layers of the temporal universe, and a set $Y$ of context variables, the set $T(C \cup Y)$ of context terms is equal to $C \cup Y$. The set $T(A \cup X)$ of algebraic terms denoting temporal displacements is built up as follows. Let $A$ be a set of algebraic constants and $X$ be a set of algebraic variables. $T(A \cup X)$ is the smallest set such that $A \subseteq T(A \cup X)$, $X \subseteq T(A \cup X)$, and if $\alpha, \beta \in T(A \cup X)$ then $\alpha + \beta, -\alpha, 0 \in T(A \cup X)$. Finally, given a non-empty set of proposition letters $\mathcal{P}$, the set of formulas $F(\mathcal{P}, A, X, C, Y)$ is the smallest set such that $\mathcal{P} \in F(\mathcal{P}, A, X, C, Y)$, if $\phi, \psi \in F(\mathcal{P}, A, X, C, Y)$, $x \in X$, $y \in Y$, $c, c', c'' \in T(C \cup Y)$, and $\alpha, \beta \in T(X \cup A)$, then $\alpha = \beta, \alpha < \beta, c' \prec c'', c' \subset c'', \neg\phi, \phi \wedge \phi, \Delta_\alpha \phi$ (and $\nabla_\alpha \phi$), $\Delta^c \phi$ (and its dual $\nabla^c \phi := \neg\Delta^c\neg\phi$), $\Diamond\phi$ (and its dual $\Box\phi := \neg\Diamond\neg\phi$), $\forall x\phi$, and $\forall y\phi$ belong to $F(\mathcal{P}, A, X, C, Y)$. $\Delta^c$ is called the (parameterized) *contextual operator*. When applied to a formula $\phi$, it restricts the evaluation of $\phi$ to the time points of the layer denoted by $c$. The combined use of $\Delta_\alpha$ and $\Delta^c$ makes it possible to define a derived operator $\Delta^c_\alpha$ of *contextualized* (or *local*) *displacement*: $\Delta^c_\alpha \phi := \Delta^c \Delta_\alpha \phi$ (and its dual $\nabla^c_\alpha \phi := \nabla^c \nabla_\alpha \phi$). In such a case, the context term $c$ can be viewed as the sort of the algebraic term $\alpha$ (*multi-sorted algebraic terms*). $\Diamond$ is called the *projection operator*. When applied to a formula $\phi$, it allows one to evaluate $\phi$ at time points which are descendants (decomposition) or ancestors (abstraction) of the current one. Restrictions to specific sets of descendants or ancestors can be obtained by pairing the projection operator with the contextual one.

The *two-sorted frame* for time granularity is a tuple

$$\mathbf{F} = ((\mathcal{T}, \prec, \subset), \mathbf{D}; \text{DIS}, \text{CONT}, \updownarrow)$$

where $\mathcal{T}$ is the temporal universe, $\prec$ and $\subset$ are the granularity and disjointedness relations, respectively, $\mathbf{D}$ is the algebra of metric displacements, $\text{DIS} = \bigcup_{i \in M} \text{DIS}_i$ is the displacement relation, $\text{CONT} \subseteq \bigcup_{i \in M} T^i \times \mathcal{T}$ is the relation of contextualization, and $\updownarrow \subseteq \bigcup_{i \in M} T^i \times \bigcup_{i \in M} T^i$ is the projection relation. $\mathcal{T}$ is totally (resp. partially) ordered by $\prec$ (resp. $\subset$). For every layer $T^i$, the ternary relation $\text{DIS}_i \subseteq T^i \times D \times T^i$ relates pairs of time points in $T^i$ to a displacement in $D$. We assume that all $\text{DIS}_i$ satisfy the same properties. The relation CONT associates each time point with the layer it belongs to. In its full generality, such a relation allows one point to belong to more than one layer (overlapping layers). However, since we restricted ourselves to the case in which $\mathcal{T}$ is totally ordered by "$\prec$', we assume that $\mathcal{T}$ defines a partition of $\bigcup_{i \in M} T^i$. This amounts to constrain CONT to be a total function with range equal to $\mathcal{T}$. The projection relation $\updownarrow$ associates each point with its direct or indirect descendants (downward projection) and ancestors (upward projection). More precisely, for any pair of points $x, y$, $\updownarrow (x, y)$ means that either $x$ downward projects on $y$ or $x$ upward projection on $y$. Different temporal structures for time granularity can be obtained by imposing different conditions on the projection relation. Here is the list of the basic properties of the projection relation, where we assume variables $x, y, z$ to take value over (subsets of) $\bigcup_{i \in M} T^i$ and variables $\alpha, \beta$ to take value over $D$:

**reflexivity** every point $x$ projects on itself

$$\forall x \ \updownarrow(x,x)$$

**uniqueness** the projection relation does not link distinct points belonging to the same layer

$$\forall x,y,T^i((x \in T^i \wedge y \in T^i \wedge x \neq y) \rightarrow \neg \updownarrow(x,y))$$

**refinement - case 1** for any pair of layers $T^i, T^j$, with $T^j \prec T^i$, any point of $T^i$ projects on at least two points of $T^j$

$$\forall T^i, T^j, x \ \exists y,z((T^j \prec T^i \wedge x \in T^i) \rightarrow \\ (y \in T^j \wedge z \in T^j \wedge y \neq z \wedge \updownarrow(x,y) \wedge \updownarrow(x,z)))$$

**refinement - case 2** for any pair of layers $T^i, T^j$, with $T^j \prec T^i$, and every point $x \in T^i$, there exists at least one point $y \in T^j$ such that $x$ projects on $y$ and no other point $z \in T^i$ projects on it

$$\forall T^i, T^j, x \ \exists y((T^j \prec T^i \wedge x \in T^i) \rightarrow (y \in T^j \wedge \\ \updownarrow(x,y) \wedge \forall z((z \in T^i \wedge z \neq x) \rightarrow \neg \updownarrow(z,y))))$$

**separation** for any pair of layers $T^i, T^j$, with $T^j \subset T^i$, the decomposition intervals of distinct points of $T^i$ are disjoint

$$\forall T^i, T^j, x, y, x', y'((T^j \subset T^i \wedge x \in T^i \wedge y \in T^i \wedge x \neq y \wedge \\ x' \in T^j \wedge y' \in T^j \wedge \updownarrow(x,x') \wedge \updownarrow(y,y')) \rightarrow x' \neq y')$$

**symmetry** if $x$ downward (resp. upward) projects on $y$, then $y$ upward (resp. downward) projects on $x$

$$\forall x,y(\updownarrow(x,y) \rightarrow \updownarrow(y,x))$$

By pairing symmetry and separation, it easily follows that, whenever $T^j \subset T^i$, each point of the finer layer is projected on a unique point of the coarser one (*alignment*).

**downward transitivity** if $T^k \subset T^j \subset T^i$, $x \in T^i$ projects on $y \in T^j$, and $y$ projects on $z \in T^k$, then $x$ projects on $z$

$$\forall T^i, T^j, T^k, x, y, z((T^k \subset T^j \subset T^i \wedge x \in T^i \wedge \\ y \in T^j \wedge z \in T^k \wedge \updownarrow(x,y) \wedge \updownarrow(y,z)) \rightarrow \updownarrow(x,z))$$

Notice that we cannot substitute $\prec$ for $\subset$ in the above formula. Consider a temporal universe consisting of `months`, `weeks`, and `days`. The week from December 29, 2003, to January 4, 2004, belongs to the decomposition of December 2003 (as well as of January 2004) and the 3rd of January 2003 belongs to the decomposition of such a week, but not to that of December 2003.

**downward/upward transitivity - case 1** if $T^j \subset T^k \subset T^i$, $x \in T^i$ projects on $y \in T^j$, and $y$ projects on $z \in T^k$, then $x$ projects on $z$

$$\forall T^i, T^j, T^k, x, y, z((T^j \subset T^k \subset T^i \wedge x \in T^i \wedge \\ y \in T^j \wedge z \in T^k \wedge \updownarrow(x,y) \wedge \updownarrow(y,z)) \rightarrow \updownarrow(x,z))$$

As in the case of downward transitivity, we cannot substitute $\prec$ for $\subset$ in the above formula. Consider a temporal universe consisting of `years`, `months`, and `weeks`. The week from December 29, 2003, to January 4, 2004, belongs both to the decomposition of the year 2003 (as well as of the year 2004) and to the decomposition of the month of January 2004, but such a month does not belong to the decomposition of the year 2003.

**order preservation** the linear order of layers is preserved by the projection relation. For every pair $T^i, T^j$, the projection intervals are ordered, but they can possibly meet (weak order preservation)

$$\forall T^i, T^j, x, y, x', y'((x \in T^i \wedge y \in T^i \wedge x' \in T^j \wedge y' \in T^j \wedge \\ \updownarrow(x, x') \wedge \updownarrow(y, y') \wedge x \ll y) \rightarrow (x' \ll y' \vee x' = y'))$$

where $x \ll y$ iff for some $i \in M$ and $\alpha > 0$, $\mathrm{DIS}_i(x, \alpha, y)$. Weak order preservation encompasses both the case of two months that share a week and the case of two months that belong to the same year.

From refinement (cases 1 and 2), symmetry and weak order preservation, it follows that, for any pair of layers $T^i, T^j$, with $T^j \prec T^i$, any point of $T^j$ projects on either one or two points of $T^i$ (*abstraction*). Moreover, from refinement (case 2), symmetry, and weak order preservation, it follows that it is never the case that, given any pair of layers $T^i, T^j$, with $T^j \prec T^i$, two consecutive points of $T^j$ are both projected on the same two points of $T^i$.

If $T^j \subset T^i$, the projection intervals of the elements of $T^i$ over $T^j$ are ordered and disjoint, that is, we must substitute $x' \ll y'$ for $x' \ll y' \vee x' = y'$ (*strong order preservation*).

**convexity** for any ordered pair of layers $T^i, T^j$ (either $T^i \prec T^j$ or $T^j \prec T^i$), the projection relation associates any point of $T^i$ with an interval of contiguous points of $T^j$

$$\forall T^i, T^j, x, y, w, z((x \in T^i \wedge y \in T^j \wedge z \in T^j \wedge w \in T^j \wedge \\ y \ll w \wedge w \ll z \wedge \updownarrow(x, y) \wedge \updownarrow(x, z)) \rightarrow \updownarrow(x, w))$$

In some situations, the layers of the temporal universe can be assumed to (pairwise) satisfy the property of homogeneity.

**homogeneity** for every pair of (discrete) layers ordered by granularity, the projection relation associates the same number of points of the finer layer with every point of the coarser one

$$\forall T^i, T^j, x, y, x', x'' \exists y', y''((T^j \prec T^i \wedge x \in T^i \wedge y \in T^i \wedge x' \in T^j \wedge x'' \in T^j \wedge \\ x' \neq x'' \wedge \updownarrow(x, x') \wedge \updownarrow(x, x'')) \rightarrow (y' \in T^j \wedge y'' \in T^j \wedge y' \neq y'' \wedge \updownarrow(y, y') \wedge \updownarrow(y, y'')))$$

and

$$\forall T^i, T^j, x, y, y' \exists x'((T^j \prec T^i \wedge x \in T^i \wedge y \in T^i \wedge \\ y' \in T^j \wedge \updownarrow(y, y')) \rightarrow (x' \in T^j \wedge \updownarrow(x, x')))$$

Other interesting properties of the projection relation can be derived from the above ones, including *total covering*, *contiguity*, *seriality* (any point $x$ can be projected on any layer $T^i$), *upward transitivity* (if $T^k \subset T^j \subset T^i$, $x \in T^k$ projects on $y \in T^j$, and $y$ projects on $z \in T^i$, then $x$ projects on $z$), and *downward/upward transitivity - case 2* (if $T^j \subset T^i \subset T^k$, $x \in T^i$ projects on $y \in T^j$, and $y$ projects on $z \in T^k$, then $x$ projects on $z$).

To turn a two-sorted frame $\mathbf{F}$ into a *two-sorted model* $\mathbf{M}$, we first add the interpretations for context and algebraic terms, and the valuation for atomic temporal formulas. The interpretation for context terms is given by a function $h : C \cup Y \rightarrow \mathcal{T}$; that for algebraic terms

is given by a function $g : A \cup X \rightarrow D$, which is automatically extended to all terms from $T(A \cup X)$. The valuation $V$ for propositional variables is defined as in *MTL*. An atomic formula of the form $\alpha = \beta$ (resp. $\alpha < \beta$) is *true* in a model $\mathbf{M} = (\mathbf{F}; V, g, h)$ whenever $g(\alpha) = g(\beta)$ (resp. $g(\alpha) < g(\beta)$). Analogously, $c \prec c'$ (resp. $c \subset c'$) is *true* in $\mathbf{M}$ whenever $h(c) \prec h(c')$ (resp. $h(c) \subset h(c')$). Next, the *truth* of the temporal formulas $\Delta_\alpha \phi$, $\Delta^c \phi$, and $\Diamond \phi$ is defined by the following clauses:

$$\begin{aligned}
\mathbf{M}, i \Vdash \Delta_\alpha \phi \quad &\text{iff} \quad \text{there exists } j \text{ such that } \mathrm{DIS}(i, g(\alpha), j) \text{ and } \mathbf{M}, j \Vdash \phi; \\
\mathbf{M}, i \Vdash \Delta^c \phi \quad &\text{iff} \quad \mathrm{CONT}(i, h(c)) \text{ and } \mathbf{M}, i \Vdash \phi; \\
\mathbf{M}, i \Vdash \Diamond \phi \quad &\text{iff} \quad \text{there exists } j \text{ such that } \updownarrow (i, j) \text{ and } \mathbf{M}, j \Vdash \phi.
\end{aligned}$$

The semantic clauses for the dual operators $\nabla_\alpha$, $\nabla^c$, and $\Diamond$, as well as for the derived operator $\Delta^c_\alpha$, can be easily derived from the above ones. Note that $\Delta^c \phi$ (resp. $\nabla^c \phi$) conventionally evaluates to false (resp. true) outside the context $c$. Finally, to evaluate the quantified formula $\forall x\, \phi$, with $x \in X$ (resp. $\forall y\, \phi$, with $y \in Y$), at a point $i$, we write $g =_x g'$ (resp. $h =_y h'$) to state that the assignments $g$ and $g'$ (resp. $h$ and $h'$) agree on all variables except maybe $x$ (resp. $y$). We have that $(\mathbf{F}; V, g, h), i \Vdash \forall x \phi$ iff $(\mathbf{F}; V, g', h), i \Vdash \phi$, for all assignments $g'$ such that $g =_x g'$. Analogously for $\forall y\, \phi$.

The notions of satisfiability, validity, and logical consequence given for *MTL* can be easily generalized to *MLTL*. Furthermore, the layered structure of *MLTL*-frames makes it possible to define the notions of *local* satisfiability, validity, and logical consequence by restricting the general notions of satisfiability, validity, and logical consequence to a specific layer.

The following examples show how *MLTL* allows one to specify temporal conditions involving different time granularities (the application of *MLTL* to the specification of complex real-time systems is discussed in [Montanari, 1996]). In the simplest case (case (i)), *MLTL* specifications are obtained by contextualizing formulas and composing them by means of logical connectives. The projection operator is needed when displacements over different layers have to be composed (case (ii)). Finally, contextual and projection operators can be paired to specify nested quantifications (cases (iii)-(vi)).

**Example 3.4.3.** *Consider the temporal conditions expressed by the following sentences:*

*(i) men work every month and eat every day;*

*(ii) in 20 seconds 5 minutes will have passed from the occurrence of the fault;*

*(iii) some days the plant works every hour;*

*(iv) some days the plant remains inactive for several hours;*

*(v) every day the plant is in production for some hours;*

*(vi) the plant is monitored by the remote system every minute of every hour.*

*They can be expressed in* MLTL *by means of the following formulas:*

*(i)* $\forall x_{man}(\forall \alpha \nabla_\alpha^{month} \mathtt{work}(x_{man}) \wedge \forall \beta \nabla_\beta^{day} \mathtt{eat}(x_{man}))$;

*(ii)* $\Delta_{20}^{second} \Diamond \Delta_{-5}^{minute} \mathtt{fault}$;

*(iii)* $\exists\alpha\Delta_{\alpha}^{day}\Box\nabla^{hour}\texttt{work(plant)}$;

*(iv)* $\exists\alpha\Delta_{\alpha}^{day}\Diamond\Delta^{hour}\texttt{inactive(plant)}$;

*(v)* $\forall\alpha\nabla_{\alpha}^{day}\Diamond\Delta^{hour}\texttt{in\_production(plant)}$;

*(vi)* $\forall\alpha\nabla_{\alpha}^{hour}\Box\nabla^{minute}\texttt{monitor(remote-system, plant)}$.

*As a matter of fact, it is possible to give a stronger interpretation of condition (ii), which is expressed by the formula:*

*(ii')* $\Delta_{20}^{second}\Diamond\Delta_{-5}^{minute}\texttt{fault} \wedge \forall\alpha(0 \leq \alpha < 20 \rightarrow \neg\Delta_{\alpha}^{second}\Diamond\Delta_{-5}^{minute}\texttt{fault})$.

The problem of finding an axiomatization of validity in *MLTL* is addressed in [Ciapessoni *et al.*, 1993; Montanari, 1996]. The idea is to pair axioms and rules of *(Q-)MTL*, which are used to express the properties of the displacement operator with respect to every context, with additional axiom schemata and rules governing the behavior of the contextual and projection operators as well as the relations between these operators and the displacement one. First, the axiomatic system for *MLTL* must constrain $\prec$ to be a total order and $\subset$ to be a partial order that refines $\prec$, that is, for every pair of contexts $c, c'$ we have that if $c \subset c'$, then $c \prec c'$, but not necessarily vice versa. Moreover, it must express the basic logical properties of the contextual and projection operators:

| | | |
|---|---|---|
| (AxNC) | $\nabla^c(\phi \rightarrow \psi) \rightarrow (\nabla^c\phi \rightarrow \nabla^c\psi)$ | (normality of $\nabla^c$); |
| (AxNP) | $\Box(\phi \rightarrow \psi) \rightarrow (\Box\phi \rightarrow \Box\psi)$ | (normality of $\Box$); |
| (AxNEC) | $\Delta^c\phi \rightarrow \phi$ | ("necessity" for $\Delta^c$); |
| (AxIC) | $\nabla^c\nabla^c\phi \equiv \nabla^c\phi$ | (idempotency of $\nabla^c$); |
| (AxCCD) | $\nabla^c\nabla_\alpha\phi \equiv \nabla_\alpha\nabla^c\phi$ | (commutativity of $\nabla^c$ and $\nabla_\alpha$), |

together with the rules:

| | | |
|---|---|---|
| (C-NEC) | $\vdash \phi \longrightarrow \vdash \nabla^c\phi$ | (necessitation rule for $\nabla^c$); |
| (P-NEC) | $\vdash \phi \longrightarrow \vdash \Box\phi$ | (necessitation rule for $\Box$). |

Notice that the projection operators $\Diamond$ and $\Box$ behave as the usual modal operators of possibility and necessity, while the contextual operators $\Delta^c$ and $\nabla^c$ are less standard (a number of theorems that account for the behavior of the contextual operators are given in [Montanari, 1996]). The set of axioms must also include the Barcan formula for the contextual and projection operators:

| | | |
|---|---|---|
| (AxBFC) | $\forall x\nabla^c\phi \rightarrow \nabla^c\forall x\phi$, with $x \neq c$ | (Barcan formula for $\nabla^c$); |
| (AxBFP) | $\forall x\Box\phi \rightarrow \Box\forall x\phi$ | (Barcan formula for $\Box$), |

as well as the counterparts of axioms (AxAD1)-(AxAD4) for the contextual operator. Similar axioms must be used to constrain the relationships between context terms, ordered by $\prec$ or $\subset$, and the displacement and contextual operators. Finally, we add a number of axioms that express the properties of the temporal structure, that is, the structural properties of the contextualization and projection relations. As an example, the axiom $\forall c_1, c_2, c_3((c_3 \subset c_2 \subset c_1 \wedge \nabla^{c_1}\Box\nabla^{c_3}\phi) \rightarrow \nabla^{c_1}\Box\nabla^{c_2}\Box\nabla^{c_3}\phi)$ can be added to constrain the projection relation to be downward transitive. Different classes of structures (e.g., homogeneous and non-homogeneous) can be captured by different sets of axioms. A sound axiomatic system

for *MLTL* is reported in [Montanari *et al.*, 1992]. No completeness proof is given. In principle, one can try to directly prove it by building a canonical model for *MLTL*. However, even though there seem to be no specific technical problems to solve, the process of canonical model construction is undoubtedly very demanding in view of the size and complexity of the *MLTL* axiom system. As a matter of fact, one can follow an alternative approach, based on the technique proposed by Finger and Gabbay in [Finger and Gabbay, 1996], which views temporal logics for time granularity as combinations of simpler temporal logics, and specifies what constraints such combinations must satisfy to guarantee the transference of logical properties (including completeness results) from the component logics to the combined ones. In Section 3.4.3 we shall present temporal logics for time granularity which are obtained as suitable combinations of existing linear and branching temporal logics.

We conclude the section with a discussion of two classical problems about granularity conversions. The first problem has already been pointed out at the beginning of the section: given the truth value of a formula with respect to a certain layer, can we constrain (and how) its truth value with respect to the other layers? In [Montanari and Policriti, 1996], Montanari and Policriti give an example of a proposition which is true at every point of a given layer, and false with respect to every point of another one. It follows that, in general, we can record the links explicitly provided by the user, but we cannot impose any other constraint about the truth value of a formula with respect to a layer different from the layer it is associated with. Accordingly, *MLTL* makes it possible to write formulas involving granularity changes, but the proposed axiomatic systems do not impose any general constraint on the relations among the truth values of a formula with respect to different layers. Nevertheless, from a practical point of view, it makes sense to look for general rules expressing *typical* relations among truth values. In [Ciapessoni *et al.*, 1993], Ciapessoni et al. introduce two consistency rules that respectively allow one to project simple *MLTL* formulas, that is, *MLTL* formulas devoid of any occurrence of the displacement, contextual, and projection operators, from coarser to finer layers (downward temporal projection) and from finer to coarser ones (upward temporal projection). For any given pair of layers $T^i, T^j$, with $T^j \prec T^i$, any point $x \in T^i$, and any simple formula $\phi$, *downward temporal projection* states that if $\phi$ holds at $x$, then there exists at least one $y \in T^j$ such that $\updownarrow(x, y)$ and $\phi$ holds at $y$, while *upward temporal projection* states that if $\phi$ holds at every $y \in T^j$ such that $\updownarrow(x, y)$, then $\phi$ holds at $x$. Formally, downward temporal projection is defined by the formula $\forall c_1, c_2(c_2 \subset c_1 \rightarrow \nabla^{c_1}(\phi \rightarrow \Diamond\Delta^{c_2}\phi))$, while upward temporal projection is defined by the formula $\forall c_1, c_2(c_2 \subset c_1 \rightarrow \nabla^{c_1}(\Box\nabla^{c_2}\phi \rightarrow \phi))$. It is not difficult to show that the two formulas are *inter-deducible* [Montanari, 1996]. (Downward) temporal projection captures the *weakest semantics* that can be attached to a statement with respect to a layer finer than the original one, provided that the statement is not wholistic. In most cases, however, such semantics is too weak, and additional user qualifications are needed. Various domain-specific categorizations of statements have been proposed in the literature [Roman, 1990; Shoham, 1988], which allow one to classify statements according to their behavior under temporal projection, e.g., events, properties, facts, and processes. In [Montanari, 1994], Montanari proposes some specializations of the *MLTL* projection operator $\Diamond$ that allow one to define different types of temporal projection, distinguishing among statements that hold at one and only one point of the decomposition interval (*punctual*), statements that hold at every point of such an interval (*continuous and pervasive*), statements that hold over a scattered sequence of sub-intervals of the decomposition interval (*bounded sequence*), and so on.

The second problem is the synchronization problem. We introduced this problem in Section 3.2, where we showed that the interpretations of the statements "tomorrow I will eat" and "dinner will be ready in one hour" with respect to a layer finer than the layer they explicitly refer to differ a lot. It is not difficult to show that even the same statement may admit different interpretations with respect to different finer layers (a detailed example can be found in [Montanari, 1996]). In general, the synchronization problem arises when logical formulas which state that a given fact holds at a point $y$ of a layer $T^i$ at distance $\alpha$ from the current point $x$ need to be interpreted with respect to a finer layer $T^j$. There exist at least two possible interpretations for the original formula with respect to $T^j$ (for the sake of simplicity, we restrict our attention to facts encoded by simple *MLTL* formulas, with a punctual interpretation under temporal projection, and we assume the temporal universe to be homogeneous). The first interpretation maps $x$ (resp. $y$) into an arbitrary point $x'$ (resp. $y'$) of its decomposition interval, thus allowing the distance $\alpha'$ between $x'$ and $y'$ to vary. If $x$ precedes $y$, we get the minimum (resp. maximum) value for $\alpha'$ when $x'$ is the last (resp. first) element of the decomposition interval for $x$ and $y'$ is the first (resp. last) element of the decomposition interval for $y$. The second interpretation forces the mapping for $y$ to conform to the mapping for $x$. As an example, if $x$ is mapped into the first element of its decomposition interval, then $y$ is mapped into the first element of its decomposition interval as well. As a consequence, there exists only one possible value for the distance $\alpha'$. The first interpretation can be easily expressed in *MLTL* (it is the interpretation underlying the semantics of the projection operator). In order to enable *MLTL* to support the second interpretation, two extensions are needed: (i) we must replace the notion of current point by the notion of *vector* of current points (one for each layer); (ii) we must define a new projection operator that maps the current point of $T^i$ into the current point of $T^j$, for every pair of layers $T^i, T^j$. Such extensions are accomplished in [Montanari, 1994]. In particular, it is possible to show that the new projection operator is second-order definable in terms of the original one, and that both projection operators are (second-order) definable in terms of a third simpler projection operator that maps every point into the first elements of its decomposition (and abstraction) intervals.

### 3.4.2 Monadic theories of time granularity

We move now from the temporal logic setting to the classical one, focusing our attention on monadic theories of time granularity. First, we introduce the relational structures for time granularity; then we present the theories of such structures and we analyze their decision problem. At the end, we briefly study the definability and decidability of meaningful binary predicates for time granularity with respect to such theories and some fragments of them.

#### Relational structures for time granularity

We begin with some preliminary definitions about finite and infinite sequences and trees (we assume the reader to be familiar with the notation and the basic notions of the theory of formal languages). Let $A$ be a finite set of symbols and $A^*$ be its Kleene closure. The length of a string $x \in A^*$, denoted by $|x|$, is defined in the usual way: $|\epsilon|=0$, $|xa| = |x| + 1$. For any pair $x, y \in A^*$, we say that $x$ is a *prefix* of $y$, denoted by $x <_{pre} y$, if $xw = y$ for some $w \in A^+ (= A^* \setminus \{\epsilon\})$. The *prefix* relation $<_{pre}$ is a partial ordering over $A^*$. If $A$ is totally ordered, a total ordering over $A^*$ can be obtained from the one over $A$ as follows. Let $<$ be

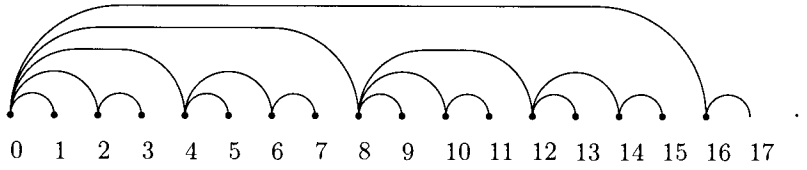



Figure 3.1: The structure of the relation $\mathtt{flip}_2$.

the total ordering over $A$. For every $x, y \in A^*$, we say that $x$ lexicographically precedes $y$ with respect to $<$, denoted $x <_{lex} y$, if either $x <_{pre} y$ or there exist $z \in A^*$ and $a, b \in A$ such that $za \leq_{pre} x$, $zb \leq_{pre} y$, and $a < b$. The *lexicographical* relation $<_{lex}$ is a total ordering over $A^*$.

A *finite sequence* is a relational structure $s = \langle I, < \rangle$, where $I$ is an initial segment of the natural numbers $\mathbb{N}$ and $<$ is the usual ordering over $\mathbb{N}$. Given a *finite* set of monadic predicate symbols $\mathcal{P}$, a $\mathcal{P}$-labeled finite sequence is a relational structure $s_\mathcal{P} = \langle s, (\overline{P})_{P \in \mathcal{P}} \rangle$, where $s = \langle I, < \rangle$ and, for every $P \in \mathcal{P}$, $\overline{P} \subseteq I$ is the set of elements labeled with $P$ (note that $\overline{P} \cap \overline{Q}$, with $P, Q \in \mathcal{P}$, can obviously be nonempty). An *infinite sequence* ($\omega$-sequence for short) is a relational structure $s = \langle \mathbb{N}, < \rangle$ and a $\mathcal{P}$-labeled $\omega$-sequence $s_\mathcal{P}$ is an $\omega$-sequence $s$ expanded with the sets $\overline{P}$, for $P \in \mathcal{P}$. For the sake of simplicity, hereafter we shall use the symbol $P$ to denote both a monadic predicate and its interpretation; accordingly, we shall rewrite $s_\mathcal{P}$ as $\langle s, (P)_{P \in \mathcal{P}} \rangle$. In the following, we shall take into consideration three binary relations over $\mathbb{N}$, namely, $\mathtt{flip}_k$, $\mathtt{adj}$, and $2\times$. Let $k \geq 2$. The binary relation $\mathtt{flip}_k$ is defined as follows. Given $x, y \in \mathbb{N}$, $\mathtt{flip}_k(x, y)$, also denoted $\mathtt{flip}_k(x) = y$, if $y = x - z$, where $z$ is the least power of $k$ with non-null coefficient in the $k$-ary representation of $x$. Formally, $\mathtt{flip}_k(x) = y$ if $x = a_n \cdot k^n + a_{n-1} \cdot k^{n-1} + \ldots + a_m \cdot k^m$, $0 \leq a_i \leq k - 1$, $a_m \neq 0$, and $y = a_n \cdot k^n + a_{n-1} \cdot k^{n-1} + \ldots + (a_m - 1) \cdot k^m$. For instance, $\mathtt{flip}_2(18, 16)$, since $18 = 1 \cdot 2^4 + 1 \cdot 2^1$, $m = 1$, and $16 = 1 \cdot 2^4 + 0 \cdot 2^1$, while $\mathtt{flip}_2(16, 0)$, since $16 = 1 \cdot 2^4$, $m = 4$, and $0 = 0 \cdot 2^4$. Note that there exists no $y$ such that $\mathtt{flip}_2(0, y)$. The structure of $\mathtt{flip}_2$ is depicted in Figure 3.1. The relation $\mathtt{adj}$ is defined as follows: $\mathtt{adj}(x, y)$, also denoted $\mathtt{adj}(x) = y$, if $x = 2^{k_n} + 2^{k_{n-1}} + \ldots + 2^{k_0}$, with $k_n > k_{n-1} > \ldots > k_0 > 0$, and $y = x + 2^{k_0} + 2^{k_0 - 1}$. For instance, $\mathtt{adj}(12, 18)$, since $12 = 2^3 + 2^2$, $k_0 = 2$, and $18 = 12 + 2^2 + 2^1$, while there exists no $y$ such that $\mathtt{adj}(13, y)$, since $13 = 2^3 + 2^2 + 2^0$ and $k_0 = 0$. Finally, for any pair $x, y \in \mathbb{N}$, it holds that $2 \times (x, y)$ if $y = 2x$.

Finite and infinite ($k$-ary) trees are defined as follows. Let $k \geq 2$ and $T_k$ be the set $\{0, \ldots k - 1\}^*$. A set $D \subseteq T_k$ is a ($k$-ary) *tree domain* if:

1. $D$ is *prefix closed*, that is, for every $x, y \in T_k$, if $x \in D$ and $y <_{pre} x$, then $y \in D$;
2. for every $x \in T_k$, either $xi \in D$ for every $0 \leq i \leq k - 1$ or $xi \notin D$ for every $0 \leq i \leq k - 1$.

Note that, according to the definition, the whole $T_k$ is a tree domain. A *finite tree* is a relational structure $\kappa = \langle D, (\downarrow_i)_{i=0}^{k-1}, <_{pre} \rangle$, where $D$ is a finite tree domain, for every $0 \leq i \leq k-1$, $\downarrow_i$ is the $i$-th *successor relation* over $D$ such that $\downarrow_i (x, y)$, also denoted $\downarrow_i (x) = y$, if $y = xi$, and $<_{pre}$ is the prefix ordering over $D$ defined as above. The elements of $D$ are

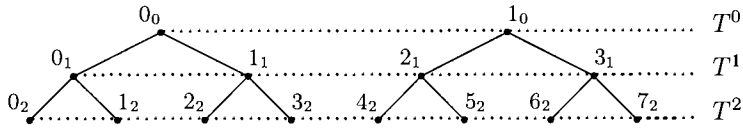



Figure 3.2: The 2-refinable 3-layered structure.

called *nodes*. If $\downarrow_i (x) = y$, then $y$ is said the $i$-th *son* of $x$. The lexicographical ordering $<_{lex}$ over $D$ is defined with respect to the natural ordering $<$ over $\{0, \ldots k-1\}$ such that $0 < 1 < \ldots < k-1$. A *path* $P$ in $\kappa$ is a subset of $D$ whose nodes can be written as a sequence $x_0, x_1, \ldots$ such that, for every $i > 0$, there exists $0 \leq j \leq k-1$ with $x_i = \downarrow_j (x_{i-1})$. We shall denote by $P(i)$ the $i$-th element $x_i$ of the path $P$. A *full path* is a maximal path with respect to set inclusion. A *chain* is any subset of a path. The *root* of $\kappa$ is the node $\epsilon$. A *leaf* of $\kappa$ is an element $x \in D$ devoid of sons. A node which is not a leaf is called an *internal node*. The *depth* of a node $x \in D$ is the length of the (unique) path from the root $\epsilon$ to $x$. The *height* of $\kappa$ is the maximum of the depths of the nodes in $D$. $\kappa$ is *complete* if every leaf has the same depth. A $\mathcal{P}$-labeled finite tree is a relational structure $\kappa = \langle D, (\downarrow_i)_{i=0}^{k-1}, <_{pre}, (P)_{P \in \mathcal{P}} \rangle$, where the tuple $(D, (\downarrow_i)_{i=0}^{k-1}, <_{pre})$ is a finite tree and, for every $P \in \mathcal{P}$, $P \subseteq D$ is the set of nodes labeled with $P$. As for infinite trees, we are interested in *complete* infinite trees over the tree domain $T_k$. The complete *infinite tree* over $T_k$ is the tuple $\kappa = \langle T_k, (\downarrow_i)_{i=0}^{k-1}, <_{pre} \rangle$. Paths, full paths, and chains are defined as for finite trees. A $\mathcal{P}$-labeled infinite tree is an expansion of the complete infinite tree over $T_k$ with monadic predicates $P$, for $P \in \mathcal{P}$.

Relational structures for time granularity consists of a (possibly infinite) number of distinct layers/domains (we shall use the two terms interchangeably). We focus our attention on $n$-layered structures, which include a fixed finite number $n$ of layers, and $\omega$-layered structures, which feature an infinite number of layers.

Let $n \geq 1$ and $k \geq 2$. For every $0 \leq i < n$, let $T^i = \{j_i \mid j \geq 0\}$. The *$n$-layered temporal universe* is the set $\mathcal{U}_n = \bigcup_{0 \leq i < n} T^i$. The ($k$-refinable) *$n$-layered structure* ($n$-LS for short) is the relational structure $\langle \mathcal{U}_n, (\downarrow_j)_{j=0}^{k-1}, < \rangle$. Such a structure can be viewed as an infinite sequence of complete ($k$-ary) trees of height $n-1$, each one rooted at a point of the coarsest layer $T^0$ (see Figure 3.2). The sets $T^i$, with $0 \leq i < n$, are the layers of the trees. For every $0 \leq j \leq k-1$, $\downarrow_j$ is the $j$-th *successor relation* over $\mathcal{U}_n$ such that $\downarrow_j (x, y)$ (also denoted $\downarrow_j (x) = y$) if $y$ is the $j$-th son of $x$. Hereafter, to adhere to the common terminology in the field, we shall substitute the term projection for the term successor. Note that for all $x$ belonging to the finest layer $T^{n-1}$ there exist no $0 \leq j \leq k-1$ and $y \in \mathcal{U}_n$ such that $\downarrow_j (x) = y$. Finally, $<$ is a total ordering over $\mathcal{U}_n$ given by the *pre-order* (root-left-right in the binary trees) visit of the nodes (for elements belonging to the same tree) and by the total linear ordering of trees (for elements belonging to different trees). Formally, for any pair $a_b, c_d \in \mathcal{U}_n$, we have that $\downarrow_j (a_b) = c_d$ if $b < n-1$, $d = b+1$, and $c = a \cdot k + j$. The total ordering $<$ is defined as follows:

1. if $x = a_0$, $y = b_0$, and $a < b$ over $\mathbb{N}$, then $x < y$;

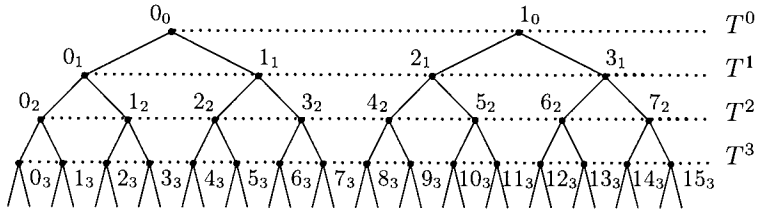



Figure 3.3: The 2-refinable downward unbounded layered structure.

2. for all $x \in \mathcal{U}_n \setminus T^{n-1}$, $x <\downarrow_0 (x)$, and $\downarrow_j (x) <\downarrow_{j+1} (x)$, for all $0 \leq j < k-1$;
3. if $x \in \mathcal{U}_n \setminus T^{n-1}$, $x < y$, and not $\texttt{ancestor}(x, y)$, then $\downarrow_{k-1} (x) < y$;
4. if $x < z$ and $z < y$, then $x < y$,

where $\texttt{ancestor}(x, y)$ if there exists $0 \leq j \leq k-1$ such that $\downarrow_j (x) = y$ or there exist $0 \leq j \leq k-1$ and $z$ such that $\downarrow_j (z) = y$ and $\texttt{ancestor}(x, z)$. A *path* over the $n$-LS is a subset of the domain whose elements can be written as a sequence $x_0, x_1, \ldots x_m$, with $m \leq n-1$, in such a way that, for every $i = 1, \ldots m$, there exists $0 \leq j < k$ for which $x_i =\downarrow_j (x_{i-1})$. A *full path* is a maximal path with respect to set inclusion. A *chain* is any subset of a path. A $\mathcal{P}$-labeled $n$-LS is a relational structure $\langle \mathcal{U}_n, (\downarrow_i)_{i=0}^{k-1}, <, (P)_{P \in \mathcal{P}} \rangle$, where the tuple $(\mathcal{U}_n, (\downarrow_i)_{i=0}^{k-1}, <)$ is the $n$-LS and, for every $P \in \mathcal{P}$, $P \subseteq \mathcal{U}_n$ is the set of points labeled with $P$.

As for $\omega$-layered structures, we focus our attention on the ($k$-refinable) downward unbounded layered structure (DULS for short), which consists of a coarsest domain together with an infinite number of finer and finer domains, and the ($k$-refinable) upward unbounded layered structure (UULS for short), which consists of a finest temporal domain together with an infinite number of coarser and coarser domains. Let $\mathcal{U} = \bigcup_{i \geq 0} T^i$ be the *$\omega$-layered temporal universe*. The DULS is a relational structure $\langle \mathcal{U}, (\downarrow_i)_{i=0}^{k-1}, < \rangle$. It can be viewed as an infinite sequence of complete ($k$-ary) infinite trees, each one rooted at a point of the coarsest domain $T^0$ (see Figure 3.3). The sets $T^i$, with $i \geq 0$, are the layers of the trees. The definitions of the projection relations $\downarrow_j$, with $0 \leq j \leq k-1$, and the total ordering $<$ over $\mathcal{U}$ are close to those for the $n$-LS. Formally, for any pair $a_b, c_d \in \mathcal{U}$, we have that $\downarrow_j (a_b) = c_d$ if and only if $d = b+1$ and $c = a \cdot k + j$, while the total ordering $<$ is defined as follows:

1. if $x = a_0$, $y = b_0$, and $a < b$ over $\mathbb{N}$, then $x < y$;
2. for all $x \in \mathcal{U}$, $x <\downarrow_0 (x)$, and $\downarrow_j (x) <\downarrow_{j+1} (x)$, for all $0 \leq j < k-1$;
3. if $x < y$ and not $\texttt{ancestor}(x, y)$, then $\downarrow_{k-1} (x) < y$;
4. if $x < z$ and $z < y$, then $x < y$.

A *path* over the DULS is a subset of the domain whose elements can be written as an infinite sequence $x_0, x_1, \ldots$ such that, for every $i \geq 1$, there exists $0 \leq j < k$ for which $x_i =\downarrow_j (x_{i-1})$. A *full path* is a maximal (infinite) path with respect to set inclusion. A *chain* is

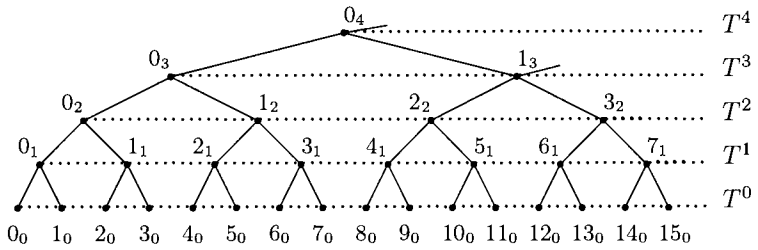



Figure 3.4: The 2-refinable upward unbounded layered structure.

any subset of a path. A $\mathcal{P}$-labeled DULS is a relational structure $\langle \mathcal{U}, (\downarrow_i)_{i=0}^{k-1}, <, (P)_{P \in \mathcal{P}} \rangle$, where the tuple $(\mathcal{U}, (\downarrow_i)_{i=0}^{k-1}, <)$ is the DULS and, for every $P \in \mathcal{P}$, $P \subseteq \mathcal{U}$ is the set of points labeled with $P$.

The UULS is a relational structure $\langle \mathcal{U}, (\downarrow_i)_{i=0}^{k-1}, < \rangle$. It can be viewed as a complete ($k$-ary) infinite tree generated from the leaves (Figure 3.4). The sets $T^i$, with $i \geq 0$, are the layers of the tree. For every $0 \leq j \leq k-1$, $\downarrow_j$ is the $j$-th *projection relation* over $\mathcal{U}$ such that $\downarrow_j (x, y)$ (also denoted by $\downarrow_j (x) = y$) if $y$ is the $j$-th son of $x$. The total ordering $<$ over $\mathcal{U}$ is induced by the *in-order* (left-root-right in the binary tree) visit of the treelike structure. Formally, for every $a_b, c_d \in \mathcal{U}$, $\downarrow_j (a_b) = c_d$ if $b > 0$, $d = b - 1$, and $c = a \cdot k + j$. The total ordering $<$ is defined as follows:

1. for all $x \in \mathcal{U} \setminus T^0$, $\downarrow_0 (x) < x$, $x < \downarrow_1 (x)$, and $\downarrow_j (x) < \downarrow_{j+1} (x)$, for every $0 < j < k - 1$;
2. if $x < y$ and not $\texttt{ancestor}(x, y)$, then $\downarrow_{k-1} (x) < y$;
3. if $x < y$ and not $\texttt{ancestor}(y, x)$, then $x < \downarrow_0 (y)$;
4. if $x < z$ and $z < y$, then $x < y$.

A *path* over the UULS is a subset of the domain whose elements can be written as an infinite sequence $x_0, x_1, \ldots$ such that, for every $i \geq 1$, there exists $0 \leq j < k$ such that $x_{i-1} = \downarrow_j (x_i)$. A *full path* is a maximal (infinite) path with respect to set inclusion. A *chain* is any subset of a path. It is worth noting that every pair of paths over the UULS may differ on a finite prefix only. A $\mathcal{P}$-labeled UULS is obtained by expanding the UULS with a set $P \subseteq \mathcal{U}$, for any $P \in \mathcal{P}$.

### Theories of time granularity

We are now ready to introduce the theories of time granularity. They are systems of monadic second-order (MSO for short) logic that allow quantification over arbitrary sets of elements. We shall study the properties of the full systems as well as of some meaningful fragments of them. We shall show that some granularity theories can be reduced to well-know classical MSO theories, such as the MSO theory of one successor and the MSO theory of two successors, while other granularity theories are proper extensions of them.

**Definition 3.4.1.** *(The language of monadic second-order logic)*

*Let $\tau = c_1, \ldots, c_r, u_1, \ldots, u_s, b_1, \ldots, b_t$ be a finite alphabet of symbols, where $c_1, \ldots, c_r$ (resp. $u_1, \ldots, u_s$, $b_1, \ldots, b_t$) are constant symbols (resp. unary relational symbols, binary relational symbols), and let $\mathcal{P}$ be a finite set of uninterpreted unary relational symbols. The second-order language with equality $\mathrm{MSO}[\tau \cup \mathcal{P}]$ is built up as follows:*

1. atomic formulas *are of the forms $x = y$, $x = c_i$, with $1 \leq i \leq r$, $u_i(x)$, with $1 \leq i \leq s$, $b_i(x, y)$, with $1 \leq i \leq t$, $x \in X$, $x \in P$, where $x$, $y$ are individual variables, $X$ is a set variable, and $P \in \mathcal{P}$;*

2. formulas *are built up from atomic formulas by means of the Boolean connectives $\neg$ and $\wedge$, and the quantifier $\exists$ ranging over both individual and set variables.*

In the following, we shall write $\mathrm{MSO}_{\mathcal{P}}[\tau]$ for $\mathrm{MSO}[\tau \cup \mathcal{P}]$; in particular, we shall write $\mathrm{MSO}[\tau]$ when $\mathcal{P}$ is meant to be the empty set. The first-order fragment of $\mathrm{MSO}_{\mathcal{P}}[\tau]$ will be denoted by $\mathrm{FO}_{\mathcal{P}}[\tau]$, while its path (resp. chain) fragment, which is obtained by interpreting second-order variables over paths (resp. chains), will be denoted by $\mathrm{MPL}_{\mathcal{P}}[\tau]$ (resp. $\mathrm{MCL}_{\mathcal{P}}[\tau]$). We focus our attention on the following theories:

1. $\mathrm{MSO}_{\mathcal{P}}[<]$ and its first-order fragment interpreted over finite and $\omega$-sequences;

2. $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{flip}_k]$ (as well as its first-order fragment), $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{adj}]$, and $\mathrm{MSO}_{\mathcal{P}}[<, 2\times]$ interpreted over $\omega$-sequences;

3. $\mathrm{MSO}_{\mathcal{P}}[<_{pre}, (\downarrow_i)_{i=0}^{k-1}]$ and its first-order, path, and chain fragments interpreted over finite and infinite trees;

4. $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ and its first-order, path, and chain fragments interpreted over the $n$-LS, the DULS, and the UULS.

We preliminarily introduce some notations and basic properties that will help us in comparing the expressive power and logical properties of the various theories. Most definitions and results are given for full MSO theories with uninterpreted unary relational symbols, but they immediately transfer to their fragments, possibly devoid of uninterpreted unary relational symbols.

Let $\mathcal{M}(\varphi)$ be the set of models of the formula $\varphi$. We say that $\mathrm{MSO}_{\mathcal{P}}[\tau_1]$ can be *embedded* into $\mathrm{MSO}_{\mathcal{P}}[\tau_2]$, denoted $\mathrm{MSO}_{\mathcal{P}}[\tau_1] \rightarrow \mathrm{MSO}_{\mathcal{P}}[\tau_2]$, if there is an *effective* translation $tr$ of $\mathrm{MSO}_{\mathcal{P}}[\tau_1]$-formulas into $\mathrm{MSO}_{\mathcal{P}}[\tau_2]$-formulas such that, for every formula $\varphi \in \mathrm{MSO}_{\mathcal{P}}[\tau_1]$, $\mathcal{M}(\varphi) = \mathcal{M}(tr(\varphi))$. For instance, it is easy to prove that $\mathrm{FO}_{\mathcal{P}}[<_{pre}, (\downarrow_i)_{i=0}^{k-1}] \rightarrow \mathrm{MPL}_{\mathcal{P}}[<_{pre}, (\downarrow_i)_{i=0}^{k-1}] \rightarrow \mathrm{MCL}_{\mathcal{P}}[<_{pre}, (\downarrow_i)_{i=0}^{k-1}] \rightarrow \mathrm{MSO}_{\mathcal{P}}[<_{pre}, (\downarrow_i)_{i=0}^{k-1}]$ (the same holds for their counterparts devoid of $\mathcal{P}$), where all theories are interpreted over trees. The condition '$X$ is a path' can indeed be written in the monadic chain logic, and the condition '$X$ is a chain' can be expressed in the MSO logic. It is also easy to show that the monadic path logic over paths is as expressive as the monadic path logic over full paths. Moreover, we say that $\mathrm{MSO}_{\mathcal{P}}[\tau_1]$ is *as expressive as* $\mathrm{MSO}_{\mathcal{P}}[\tau_2]$, written $\mathrm{MSO}_{\mathcal{P}}[\tau_1] \rightleftarrows \mathrm{MSO}_{\mathcal{P}}[\tau_2]$, if both $\mathrm{MSO}_{\mathcal{P}}[\tau_1] \rightarrow \mathrm{MSO}_{\mathcal{P}}[\tau_2]$ and $\mathrm{MSO}_{\mathcal{P}}[\tau_2] \rightarrow \mathrm{MSO}_{\mathcal{P}}[\tau_1]$. It is immediate to see that if $\mathrm{MSO}_{\mathcal{P}}[\tau_1] \rightarrow \mathrm{MSO}_{\mathcal{P}}[\tau_2]$ and $\mathrm{MSO}_{\mathcal{P}}[\tau_2]$ is decidable (resp. $\mathrm{MSO}_{\mathcal{P}}[\tau_1]$ is undecidable), then $\mathrm{MSO}_{\mathcal{P}}[\tau_1]$ is decidable (resp. $\mathrm{MSO}_{\mathcal{P}}[\tau_2]$ is undecidable) as well.

Besides decidability issues, we are interested in definability ones. Let $\beta$ be a relational symbol. We say that $\beta$ is *definable* in $\mathrm{MSO}_{\mathcal{P}}[\tau]$ if $\mathrm{MSO}_{\mathcal{P}}[\tau \cup \{\beta\}] \rightarrow \mathrm{MSO}_{\mathcal{P}}[\tau]$. If the

addition of $\beta$ to a decidable theory $\mathrm{MSO}_{\mathcal{P}}[\tau]$ makes the resulting theory $\mathrm{MSO}_{\mathcal{P}}[\tau \cup \{\beta\}]$ undecidable, we can conclude that $\beta$ is not definable in $\mathrm{MSO}_{\mathcal{P}}[\tau]$. The opposite does not hold in general: the predicate $\beta$ may not be definable in $\mathrm{MSO}_{\mathcal{P}}[\tau]$, but the extension of $\mathrm{MSO}_{\mathcal{P}}[\tau]$ with $\beta$ may preserve decidability. In such a case, we obviously cannot reduce the decidability of $\mathrm{MSO}_{\mathcal{P}}[\tau \cup \{\beta\}]$ to that of $\mathrm{MSO}_{\mathcal{P}}[\tau]$.

The decidability of $\mathrm{MSO}_{\mathcal{P}}[<]$ over finite sequences has been proved in [Büchi, 1960; Elgot, 1961], while its decidability over $\omega$-sequences has been shown in [Büchi, 1962] ($\mathrm{MSO}_{\mathcal{P}}[<]$ over $\omega$-sequences is the well-known MSO theory of one successor $S1S$).

**Theorem 3.4.2.** (*Decidability of* $\mathrm{MSO}_{\mathcal{P}}[<]$ *over sequences*)
$\mathrm{MSO}_{\mathcal{P}}[<]$ *over finite (resp. infinite) sequences is non-elementarily decidable.*

The theory $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{flip}_k]$ ($S1S^k$ for short), interpreted over $\omega$-sequences, has been studied by Monti and Peron in [Monti and Peron, 2000]. Such a theory properly extends $S1S$. Moreover, the unary predicate $\mathtt{pow}_k$ such that $\mathtt{pow}_k(x)$ if $x$ is a power of $k$ can be easily expressed as $\mathtt{flip}_k(x) = 0$. Hence, $S1S^k$ is at least as expressive as the well-known (decidable) extension of $\mathrm{MSO}_{\mathcal{P}}[<]$ with the predicate $\mathtt{pow}_k$ [Elgot and Rabin, 1966]. The decidability of $S1S^k$ has been proved by showing that it is the logical counterpart of the class of $\omega$-sequences languages ($\omega$-languages for short) recognized by systolic ($k$-ary) tree automata. The class of the languages of finite sequences recognized by systolic tree automata was originally investigated by Culik II et al. in [Culik II *et al.*, 1984]. In [Monti and Peron, 2000], Monti and Peron extend the notion of systolic tree automaton to deal with $\omega$-languages. They prove that the class of systolic tree $\omega$-languages is a proper extension of the class of regular $\omega$-languages (that is, $\omega$-languages recognized by Büchi automata), that maintains the closure properties of regular $\omega$-languages as well as the decidability of the emptiness problem. The correspondence between systolic tree $\omega$-languages and $S1S^k$ is established by means of a generalization of Büchi's Theorem.

**Theorem 3.4.3.** (*Decidability of* $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{flip}_k]$ *over* $\omega$*-sequences*)
$\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{flip}_k]$ *over* $\omega$*-sequences is non-elementarily decidable.*

The theories $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{adj}]$ and $\mathrm{MSO}_{\mathcal{P}}[<, 2\times]$, interpreted over $\omega$-sequences, have been investigated in [Monti and Peron, 2001]. $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{adj}]$ is a proper extension $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{flip}_2]$. Unfortunately, unlike $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{flip}_2]$, it is undecidable.

**Theorem 3.4.4.** (*Undecidability of* $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{adj}]$ *over* $\omega$*-sequences*)
$\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{adj}]$ *over infinite sequences is undecidable.*

Since $\mathrm{MSO}_{\mathcal{P}}[<, 2\times]$ is at least as expressive as $\mathrm{MSO}_{\mathcal{P}}[<, \mathtt{adj}]$, its decision problem is undecidable as well.

**Theorem 3.4.5.** (*Undecidability of* $\mathrm{MSO}_{\mathcal{P}}[<, 2\times]$ *over* $\omega$*-sequences*)
$\mathrm{MSO}_{\mathcal{P}}[<, 2\times]$ *over* $\omega$*-sequences is undecidable.*

The theories $\mathrm{MSO}_{\mathcal{P}}[<_{pre}, (\downarrow_i)_{i=0}^{k-1}]$, interpreted over infinite ($k$-ary) trees, are the well-known MSO theories of $k$ successors ($SkS$ for short). The decidability of $SkS$ over finite trees has been shown in [Doner, 1970; Thatcher and Wright, 1968]. The decidability of the MSO theory of the infinite binary tree $S2S$ has been proved in [Rabin, 1969]. Such a result can be easily generalized to the MSO theory of the infinite $k$-ary tree $SkS$, for any $k > 2$ (and even to $S\omega S$ over countably branching trees) [Thomas, 1990].

**Theorem 3.4.6.** (*Decidability of* $\mathrm{MSO}_{\mathcal{P}}[<_{pre}, (\downarrow_i)_{i=0}^{k-1}]$ *over trees*)
$\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ *over finite (resp. infinite) trees is non-elementarily decidable.*

The decidability of $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ over the $n$-LS has been proved in [Montanari and Policriti, 1996] by reducing it to $S1S$. Such a reduction is accomplished in two steps. First, the $n$-layered structure is flattened by embedding all its layers into the finest one; then, metric temporal information is encoded by means of a finite set of unary relations. This second step is closely related to the technique exploited in [Alur and Henzinger, 1993] to prove the decidability of a family of real-time logics*. It relies on the *finite-state character* of the involved metric temporal information, which can be expressed as follows: every temporal property that partition an infinite set of states/time points into a finite set of classes can be finitely *modeled* and hence it is decidable.

**Theorem 3.4.7.** (*Decidability of* $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ *over the* $n$-LS)
$\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ *over the $n$-LS is non-elementarily decidable.*

The decidability of $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ over both the DULS and the UULS has been shown in [Montanari *et al.*, 1999]. The decidability of the theory of the DULS has been proved by embedding it into $SkS$. The infinite sequence of infinite trees of the $k$-refinable DULS can indeed be appended to the rightmost full path of the infinite $k$-ary tree. The encoding of the 2-refinable DULS into the infinite binary tree is shown in Figure 3.5. Suitable definable predicates are then used to distinguish between the nodes of the infinite tree that correspond to elements of the original DULS, and the other nodes. As an example, in the case depicted in Figure 3.5 we must differentiate the auxiliary nodes belonging to the rightmost full path of the tree from the other ones. Finally, for $0 \leq j \leq k-1$, the $j$-th projection relation $\downarrow_j$ can be interpreted as the $j$-th successor relation and the total order $<$ can be naturally mapped into the lexicographical ordering $<_{lex}$ (it is not difficult to show that $<_{lex}$ can be defined in $SkS$).

**Theorem 3.4.8.** (*Decidability of* $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ *over the* DULS)
$\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ *over the DULS is non-elementarily decidable.*

The decidability of the theory of the UULS has been proved by reducing it to $S1S^k$. For the sake of simplicity, we describe the basic steps of this reduction in the case of the 2-refinable UULS (the technique can be generalized to deal with any $k > 2$). An embedding of $MSO[<, \downarrow_0, \downarrow_1]$ into $S1S^2$ can be obtained as follows. First, we replace the 2-refinable ULLS by the so-called *concrete* 2-refinable ULLS, which is defined as follows:

- for all $i \geq 0$, the $i$-th layer $T^i$ is the set $\{2^i + n2^{i+1} : n \geq 0\} \subseteq \mathbb{N}$;

*The relationships between the theories of $n$- and $\omega$-layered structures and real-time logics have been explored in detail by Montanari et al. in [Montanari *et al.*, 2000]. Logic and computer science communities have traditionally followed a different approach to the problem of representing and reasoning about time and states. Research in logic resulted in a family of (metric) tense logics that take *time* as a primitive notion and define (*timed*) *states* as sets of atomic propositions which are true at given time points, while research in computer science concentrated on the so–called (real-time) temporal logics of programs that take *state* as a primitive notion, and define *time* as an attribute of states. Montanari et al. show that the theories of time granularity provide a unifying framework within which the two approaches can be reconciled. States and time-points can indeed be uniformly referred to as elements of the (decidable) theories of the DULS and the UULS. In particular, they show that the theory of timed state sequences, underlying real-time logics, can be naturally recovered as an abstraction of such theories.

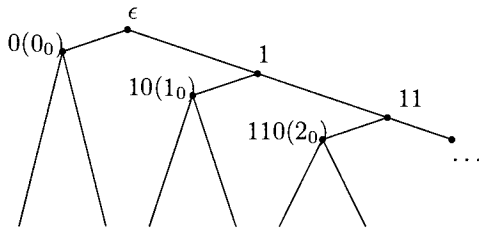



Figure 3.5: The encoding of the 2-refinable DULS into $\{0,1\}^*$.

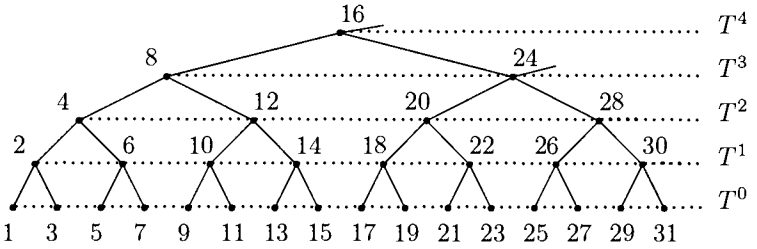



Figure 3.6: The concrete 2-refinable UULS.

- for every element $x = 2^i + n2^{i+1}$ belonging to $T^i$, with $i \geq 1$, $\downarrow_0 (x) = 2^i + n2^{i+1} - 2^{i-1} = 2^{i-1} + 2n2^i$ and $\downarrow_1 (x) = 2^i + n2^{i+1} + 2^{i-1} = 2^{i-1} + (2n+1)2^i$;
- $<$ is the usual ordering over $\mathbb{N}$.

A fragment of this concrete structure is depicted in Figure 3.6. Notice that all odd numbers are associated with layer $T^0$, while even numbers are distributed over the remaining layers. Notice also that the labeling of the concrete structure does not include the number $0^*$. It is easy to show that the two structures are isomorphic by exploiting the obvious mapping that associates each element of the 2-refinable UULS with the corresponding element of the concrete structure, preserving projection and ordering relations. Hence, the two structures satisfy the same $MSO[<, \downarrow_0, \downarrow_1]$-formulas. Next, we can easily encode the concrete 2-refinable UULS into $\mathbb{N}$. Both relations $\downarrow_0$ and $\downarrow_1$ can indeed be defined in terms of $\mathtt{flip}_2$ as follows. For any given even number $x$,

$$\begin{array}{lll} \downarrow_0 (x) = y & \text{iff} & y < x \wedge \mathtt{flip}_2(y) = \mathtt{flip}_2(x) \wedge \\ & & \neg\exists z(y < z \wedge z < x \wedge \mathtt{flip}_2(z) = \mathtt{flip}_2(x)); \\ \downarrow_1 (x) = y & \text{iff} & \mathtt{flip}_2(y) = x \wedge \neg\exists z(y < z \wedge \mathtt{flip}_2(z) = x). \end{array}$$

By exploiting such a correspondence, it is possible to define a translation $\tau$ of $MSO[<, \downarrow_0, \downarrow_1]$ formulas (resp. sentences) into $S1S^2$ formulas (resp. sentences) such that, for any formula

*In [Montanari *et al.*, 2002a], Montanari et al. show that it is convenient to consider 0 as the label of the first node of an imaginary additional finest layer, whose remaining nodes are not labeled. In such a way the node with label 0 turns out to be the left son of the node with label 1.

(resp. sentence) $\phi \in MSO[<, \downarrow_0, \downarrow_1]$, $\phi$ is satisfiable by (resp. true in) the UULS if and only if $\tau(\phi) \in S1S^2$ is satisfiable by (resp. true in) $\langle \mathbb{N}, <, \mathtt{flip}_2 \rangle$.

**Theorem 3.4.9.** (*Decidability of* $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ *over the* UULS)
$\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ *over the UULS is non-elementarily decidable.*

In [Montanari and Puppis, 2004b], Montanari and Puppis deal with the decision problem for the MSO logic interpreted over an $\omega$-layered temporal structure devoid of both a finest layer and a coarsest one (we call such a structure totally unbounded, TULS for short). The temporal universe of the TULS is the set $\mathcal{U}_n = \bigcup_{i \in \mathbb{Z}} T^i$, where $\mathbb{Z}$ is the set of integers; the layer $T^0$ is a distinguished intermediate layer of such a structure. It is not difficult to show that $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$ over both the DULS and the UULS can be embedded into $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}, L_0]$ over the TULS ($L_0$ is a unary relational symbol used to identify the elements of $T^0$). The solution to the decision problem for $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}, L_0]$ proposed by Montanari and Puppis extends Carton and Thomas' solution to the decision problem for the MSO theories of residually ultimately periodic words [Carton and Thomas, 2002]. First, they provide a tree-like characterization of the TULS and, taking advantage of it, they define a non-trivial encoding of the TULS into a vertex-colored tree that allows them to reduce the decision problem for the TULS to the problem of determining, for any given Rabin tree automaton, whether it accepts such a vertex-colored tree. Then, they reduce this latter problem to the decidable case of regular trees by exploiting a suitable notion of tree equivalence [Montanari and Puppis, 2004a].

**Theorem 3.4.10.** (*Decidability of* $\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}, L_0]$ *over the* TULS)
$\mathrm{MSO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}, L_0]$ *over the TULS is non-elementarily decidable.*

Notice that, taking advantage of the above-mentioned embedding, such a result provides, as a by-product, an alternative (uniform) decidability proof for the theories of the DULS and the UULS.

The definability and decidability of a set of binary predicates in monadic languages interpreted over the $n$-LS, the DULS, and the UULS have been systematically explored in [Franceschet *et al.*, 2003]. The set of considered predicates includes the equi-level (resp. equi-column) predicate constraining two time points to belong to the same layer (resp. column) and the horizontal (resp. vertical) successor predicate relating a time point to its successor within a given layer (resp. column), which allow one to express meaningful properties of time granularity [Montanari, 1996]. The authors investigate definability and decidability issues for such predicates with respect to $\mathrm{MSO}[<, (\downarrow_i)_{i=0}^{k-1}]$ and its first-order, chain, and path fragments $\mathrm{FO}[<, (\downarrow_i)_{i=0}^{k-1}]$, $\mathrm{MPL}[<, (\downarrow_i)_{i=0}^{k-1}]$, and $\mathrm{MCL}[\tau]$ of $\mathrm{MSO}[<, (\downarrow_i)_{i=0}^{k-1}]$ (as well as their $\mathcal{P}$-variants $\mathrm{FO}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$, $\mathrm{MPL}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$, and $\mathrm{MCL}_{\mathcal{P}}[<, (\downarrow_i)_{i=0}^{k-1}]$). Figure 3.7 summarizes the relationships between the expressive powers of such formal systems (an arrow from $\mathcal{T}$ to $\mathcal{T}'$ stands for $\mathcal{T} \rightarrow \mathcal{T}'$). From Theorems 3.4.7, 3.4.8, 3.4.9, and 3.4.10, it immediately follows that all the formalisms in Figure 3.7, when interpreted over the $n$-LS, the DULS, the UULS, and the TULS are decidable.

The outcomes of the analysis of the equi-level, equi-column, horizontal successor, and vertical successor predicates can be summarized as follows. First, the authors show that all these predicates are not definable in the MSO language over the DULS and the UULS, and that their addition immediately leads the MSO theories of such structures to undecidability.

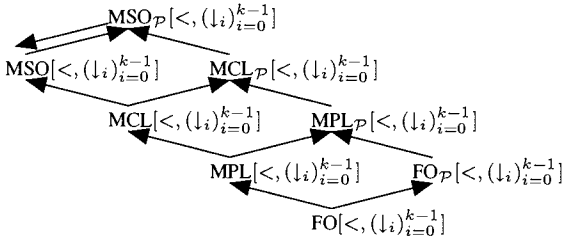



Figure 3.7: A hierarchy of monadic formalisms over layered structures.

As for the $n$-LS, the status of the horizontal (equi-level and horizontal successor) and vertical (equi-column and vertical successor) predicates turns out to be quite different: while horizontal predicates are easily definable, vertical ones are undefinable and their addition yields undecidability. Then, the authors study the effects of adding the above predicates to suitable *fragments* of the MSO language, such as its first-order, path, and chain fragments, possibly admitting uninterpreted unary relational symbols. They systematically explore all the possibilities, and give a number of positive and negative results. From a technical point of view, (un)definability and (un)decidability results are obtained by reduction from/to a wide spectrum of undecidable/decidable problems. Even though the complete picture is still missing (some decidability problems are open), the achieved results suffice to formulate some general statements. First, all predicates can be added to monadic first-order, path, and chain fragments, devoid of uninterpreted unary relational symbols, over the $n$-LS and the UULS preserving decidability. In the case of the DULS, they prove the same result for the equi-level and horizontal successor predicates, while they do not establish whether the same holds for the equi-column and vertical successor predicates. Moreover, they prove that the addition of the equi-column or vertical successor predicates to monadic first-order fragments over the $\omega$-layered structures, with uninterpreted unary relational symbols, makes the resulting theories undecidable. The effect of such additions to the $n$-layered structure is not known. As for the equi-level predicate, they only prove that adding it to the monadic path fragment over the DULS, with uninterpreted unary relational symbols, leads to undecidability. Finally, as far as the MSO language over the UULS is concerned, they establish an interesting connection between its extension with the equi-level (resp. equi-column) predicate and systolic $\omega$-languages over $Y$-trees (resp. trellis) [Gruska, 1990].

### 3.4.3 Temporalized logics and automata for time granularity

In the previous section, we have shown that monadic theories of time granularity are quite expressive, but they have not much computational appeal because their decision problem is *non-elementary*. This roughly means that it is possible to algorithmically check the truth of sentences, but the complexity of the algorithm grows very rapidly and it cannot be bounded. Moreover, the corresponding automata (Büchi sequence automata for the theory of the $n$-LS, Rabin tree automata for the theory of the DULS, and systolic tree automata for the theory of the UULS) do not directly work over layered structures, but rather over collapsed structures into which layered structures can be encoded. Hence, they are not natural and intuitive tools to specify and check properties of time granularity. In this section, we outline a different approach that connects monadic theories of time granularity back

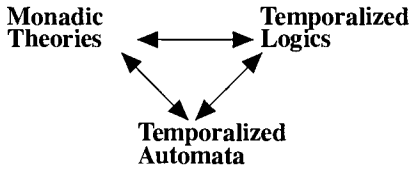



Figure 3.8: From monadic theories to temporalized logics via temporalized automata.

to temporal logic [Franceschet and Montanari, 2001a; Franceschet and Montanari, 2001b; Franceschet and Montanari, 2004]. Taking inspiration of methods for logic combinations (a short description of these methods can be found in [Franceschet *et al.*, 2004]), Franceschet and Montanari reinterpret layered structures as *combined structures*. This allows them to define suitable combined temporal logics and combined automata over layered structures, respectively called temporalized logics and temporalized automata, and to study their expressive power and computational properties by taking advantage of the transfer theorems for combined logics and combined automata. The outcome is rewarding: the resulting combined temporal logics and automata directly work over layered structures; moreover, they are expressively equivalent to monadic systems, and they are elementarily decidable.

Finding the temporal logic counterpart of monadic theories is a difficult task, involving a non-elementary blow up in the length of formulas. Ehrenfeucht games have been successfully exploited to deal with such a correspondence problem for first-order theories [Immerman and Kozen, 1989] and well-behaved fragments of second-order monadic ones, e.g., the path fragment of the monadic second-order theory of infinite binary trees [Hafer and Thomas, 1987]. As for the theories of time granularity, in [Franceschet and Montanari, 2003] Franceschet and Montanari show that an expressively complete and elementarily decidable combined temporal logic counterpart of the path fragment of the MSO theory of the DULS can be obtained by means of suitable applications of Ehrenfeucht games. Ehrenfeucht games have also been used by Montanari et al. to extend Kamp's theorem to deal with the first-order fragment of the MSO theory of the UULS [Montanari *et al.*, 2002a]. Unfortunately, these techniques produce rather involved proofs and they do not naturally lift to the full second-order case. A little detour is needed to deal with such a case. Instead of trying to establish a direct correspondence between MSO theories of time granularity and temporal logics, Franceschet and Montanari connect them via automata [Franceschet and Montanari, 2004] (cf. Figure 3.8). Firstly, they define the class of temporalized automata, which can be proved to be the automata-theoretic counterpart of temporalized logics, and they show that relevant properties, such as closure under Boolean operations, decidability, and expressive equivalence with respect to temporal logics, transfer from component automata to temporalized ones. Then, on the basis of the established correspondence between temporalized logics and automata, they reduce the task of finding a temporal logic counterpart of the MSO theories of the DULS and the UULS to the easier one of finding temporalized automata counterparts of them. The mapping of MSO formulas into automata (the difficult direction) can indeed greatly benefit from automata closure properties.

As a by-product, the alternative characterization of temporalized logics for time gran-

ularity as temporalized automata allows one to reduce logical problems to automata ones. As it is well-known in the area of automated system specification and verification, such a reduction presents several advantages, including the possibility of using automata for both system modeling and specification, and the possibility of checking the system on-the-fly (a detailed account of these advantages can be found in [Franceschet and Montanari, 2001b]).

### 3.4.4 Coda: time granularity and interval temporal logics

As pointed out in [Montanari, 1996], there exists a natural link between structures and theories of time granularity and those developed for representing and reasoning about time intervals. Differently-grained temporal domains can indeed be interpreted as different ways of partitioning a given discrete/dense time axis into consecutive disjoint intervals. According to this interpretation, every time point can be viewed as a suitable interval over the time axis and projection implements an intervals-subintervals mapping. More precisely, let us define *direct constituents* of a time point $x$, belonging to a given domain, the time points of the immediately finer domain into which $x$ can be refined, if any, and *indirect constituents* the time points into which the direct constituents of $x$ can be directly or indirectly refined, if any. The mapping of a given time point into its direct or indirect constituents can be viewed as a mapping of a given time interval into (a specific subset of) its subintervals.

The existence of such a natural correspondence between interval and granularity structures hints at the possibility of defining a similar connection at the level of the corresponding theories. For instance, according to such a connection, temporal logics over DULSs allow one to constrain a given property to hold true densely over a given time interval, where $P$ densely holds over a time interval $w$ if $P$ holds over $w$ and there exists a direct constituent of $w$ over which $P$ densely holds. In particular, establishing a connection between structures and logics for time granularity and those for time intervals would allow one to transfer decidability results from the granularity setting to the interval one. As a matter of fact, most interval temporal logics, including Moszkowski's Interval Temporal Logic (ITL) [Moszkowski, 1983], Halpern and Shoham's Modal Logic of Time Intervals (HS) [Halpern and Shoham, 1991], Venema's CDT Logic [Venema, 1991a], and Chaochen and Hansen's Neighborhood Logic (NL) [Chaochen and Hansen, 1998], are highly undecidable. Decidable fragments of these logics have been obtained by imposing severe restrictions on their expressive power, e.g., the *locality* constraint in [Moszkowski, 1983].

Preliminary results can be found in [Montanari *et al.*, 2002b], where the authors propose a new interval temporal logic, called Split Logic (SL for short), which is equipped with operators borrowed from HS and CDT, but is interpreted over specific interval structures, called *split-frames*. The distinctive feature of a split-frame is that there is at most one way to chop an interval into two adjacent subintervals, and consequently it does not possess *all* the intervals. They prove the decidability of SL with respect to particular classes of split-frames which can be put in correspondence with the first-order fragments of the monadic theories of time granularity. In particular, *discrete* split-frames with maximal intervals correspond to the $n$-layered structure, discrete split-frames (with unbounded intervals) can be mapped into the upward unbounded layered structure, and *dense* split-frames with maximal intervals can be encoded into the downward unbounded layered structure.

## 3.5 Qualitative time granularity

Granularity operators for qualitative time representation have been first provided in [Euzenat, 1993; Euzenat, 1995a]. These operators are defined in the context of relational algebras and they apply to both point and interval algebras. They have the advantage of being applicable to fully qualitative and widespread relational representations. They account for granularity phenomena occurring in actual applications using only qualitative descriptions.

After a short recall of relation algebras (Section 3.5.1), a set of six constraints applying to the granularity operators is defined (Section 3.5.2). These constraints are applied to the well-known temporal representation of point and interval algebras (Section 3.5.3). Some general results of existence and relation of these operators with composition are also given (Section 3.5.4).

### 3.5.1 Qualitative time representation and granularity

The qualitative time representation considered here is a well-known one:

1. it is based on an algebra of binary relations $\langle 2^{\Gamma}, \cup, \circ, ^{-1} \rangle$ (see Chapter 1); we focus our attention on the point and interval algebras [Vilain and Kautz, 1986; Allen, 1983]);
2. this algebra is augmented with a neighborhood structure (in which $N(r, r')$ means that the relationships $r$ and $r'$ are neighbors) [Freksa, 1992];
3. last, the construction of an interval algebra [Hirsh, 1996] is considered (the conversion of a quadruple of base relationships $R$ into an interval relation is given by $\Rightarrow R$ and the converse operation by $\Leftarrow r$ when it is defined).

In such an algebra of relations, the situations are described by a set of possible relationships holding between entities (here points or intervals).

As an example, imagine several witnesses of an air flight incident with the witness from the ground (g) saying that "the engine stopped working (W) and the plane went [immediately] down", the pilot (p) saying that "the plane worked correctly (W) until there has been a misfiring period (M) and, after that, the plane lost altitude", and the (unfortunately out of reach) "blackbox" flight data recorder (b) revealing that the plane had a short misfiring period (M) and a short laps of correct behavior before the plane lost altitude (D).
If these descriptions are rephrased in the interval algebra (see Figure 3.9), this would correspond to three different descriptions: $g = \{WmD\}$, $p = \{WmM, MmD\}$ and $b = \{WmM, MbD\}$. Obviously, if any two of these descriptions are merged, the result is an inconsistent description. However, such inconsistencies arise because the various sources of information do not share the same precision and not because of intrisically contradictory descriptions. It is thus useful to find in which way the situations described by $g$ and $p$ can be coarse views of that expressed by $b$.

The qualitative granularity is defined through a couple of operators for converting the representation of a situation into a finer or coarser representation of the same situation. These operators apply to the relationships holding between the entities and transform these relationship into other plausible relationships at a coarser (with upward conversion denoted by $\uparrow$) or finer (with downward conversion denoted by $\downarrow$) granularity. When the conversion is not oriented, i.e., when we talk about a granularity change between two layers, but it is not necessary to know which one is the coarser, a neutral operator is used (denoted by $\rightarrow$).

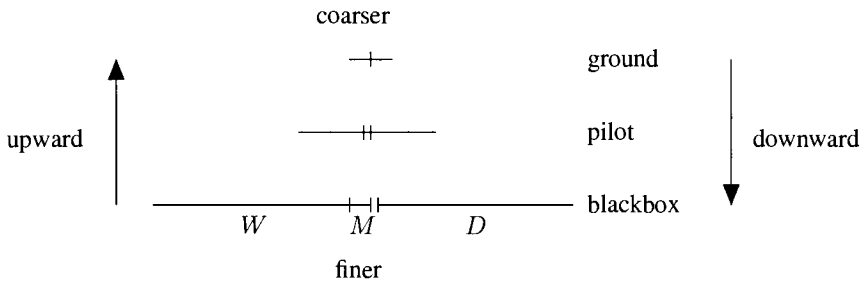



Figure 3.9: The air flight incident example.

Before turning to precisely define the granularity conversion, the assumptions underlying them must be clear. First of all, the considered language is qualitative and relational. Each layer represents a situation in the unaltered language of the relational algebra. This has the advantage of considering any description of a situation as being done under a particular granularity. Thus the layers are external to the language. The descriptions considered here are homogeneous (i.e., the language is the same for all the layers). The temporal structure is given by the algebra itself. The layers are organised as a partial order $\langle \mathcal{T}, \prec \rangle$ (sometimes it is known that a layer is coarser than another). In the example of Figure 3.9, it seems clear that $b \prec p \prec g$. It is not assumed that they are aligned or decomposed into homogeneous units, but the constraints below can enforce contiguity. The only operators considered here are the projection operators. The contextualisation operator is not explicit since (by opposition to logical systems) it cannot be composed with other operators. However, sometimes the notation ${}_g \rightarrow_{g'}$ is used, providing a kind of contextualisation (by specifying the concerned granularities). The displacement operator is useless since the relational language is not situated (or absolute, i.e., it does not evaluate the truth of a formula at a particular moment, but rather evaluates the truth of a temporal relationship between two entities).

### 3.5.2 Generic constraints on granularity change

Anyone can think about a particular set of projection operators by imagining the effects of coarseness. But here we provide a set of properties which should be satisfied by any system of granularity conversion operators. In fact, the set of properties is very small. Next section shows that they are sufficient for restricting the number of operators to only one (plus the expected operators corresponding to identity and conversion to everything).

Constraints below are given for unit relations (singletons of the set of relations). The operators on general relations are defined by:

$$\rightarrow R = \cup_{r \in R} \rightarrow r \tag{3.2}$$

#### Self-conservation

Self-conservation states that whatever be the conversion, a relationship must belong to its own conversion (this corresponds to the property named reflexivity when the conversion is a

relation).

$$r \in\!\rightarrow r \qquad \text{(self-conservation)} \tag{3.3}$$

It is quite a sensible and minimal property: the knowledge about the relationship can be less precise, but it must have a chance to be correct. Moreover, in a qualitative system, it is possible that nothing changes through granularity if the (quantitative) granularity step is small enough. Not requiring this property would disable the possibility that the same situation looks the same under different granularity. Self-conservation accounts for this.

### Neighborhood compatibility

A property considered earlier is the *order preservation* property — stated in [Hobbs, 1985] as an equivalence: $\forall x, y, x < y \equiv (\rightarrow x) < (\rightarrow y)$. This property takes for granted the availability of an order relation ($<$) structuring the set of relationships. It states that

$$\text{if } x > y \text{ then } \neg(\rightarrow x <\!\rightarrow y) \qquad \text{(order preservation)}$$

However, order preservation has the shortcoming of requiring the order relation. Its algebraic generalization could be reciprocal avoidance:

$$\text{if } xry \text{ then } \neg(\rightarrow xr^{-1} \rightarrow y) \qquad \text{(reciprocal avoidance)}$$

Reciprocal avoidance is over-generalized and conflicts with self-conservation in case of auto-reciprocal relationships (i.e. such that $r = r^{-1}$). The neighborhood compatibility, while not expressed in [Euzenat, 1993], has been taken into account informally: it constrains the conversion of a relation to form a conceptual neighborhood (and hence the conversion of a conceptual neighborhood to form a conceptual neighborhood).

$$\forall r, \forall r', r'' \in\!\rightarrow r, \exists r_1, \ldots r_n \in\!\rightarrow r : \\ r_1 = r', r_n = r'' \text{ and } \forall i \in [1, n-1] N(r_i, r_{i+1}) \\ \text{(neighborhood compatibility)} \tag{3.4}$$

This property has already been reported by Freksa [Freksa, 1992] who considers that a set of relationships must be a conceptual neighborhood in order to be seen as a coarse representation of the actual relationship. It is weaker than the two former proposals because it does not prevent the opposite to be part of the conversion. But in such a case, it constrains a path between the relation and its converse to be in the conversion too. Neighborhood compatibility seems to be the right property, partly because, instead of the former ones, it does not forbid a very coarse granularity under which any relationship is converted in the whole set of relations. It also seems natural because granularity can hardly be imagined as discontinuous (at least in continuous spaces).

### Conversion-reciprocity distributivity

An obvious property for conversion is symmetry. It states that the conversion of the relation between a first object and a second one must be the reciprocal of the conversion of the

relation between the second one and the first one. It is clear that the relationships between two temporal occurrences are symmetric and thus granularity conversion must respect this.

$$\rightarrow r^{-1} = (\rightarrow r)^{-1} \qquad \text{(distributivity of } \rightarrow \text{ on } {}^{-1}\text{)} \qquad (3.5)$$

**Inverse compatibility**

Inverse compatibility states that the conversion operators are consistent with each other, i.e., that if the relationship between two occurrences can be seen as another relationship under some granularity, then the inverse operation from the latter to the former can be achieved through the inverse operator. Stated otherwise, this property corresponds to symmetry when the operator is described as a relation.

$$r \in \bigcap_{r' \in \uparrow r} \downarrow r' \text{ and } r \in \bigcap_{r' \in \downarrow r} \uparrow r' \qquad \text{(inverse compatibility)} \qquad (3.6)$$

For instance, if someone in situation (p) of Figure 3.9 is able to imagine that, under a finer granularity (say situation b), there is some time between the misfiring period and the loss of altitude, then (s)he must be ready to accept that if (s)he were in situation (b), (s)he could imagine that there is no time between them under a coarser granularity (as in situation p).

**Idempotency**

A property which is usually considered first (especially in quantitative systems) is the full transitivity:

$${}_g{\rightarrow_{g'}}\ {}_{g'}{\rightarrow_{g''}}\ r = {}_g{\rightarrow_{g''}}\ r \qquad \text{(transitivity)}$$

This property is too strong; it would for instance imply that:

$${}_g{\uparrow^{g'}}\ {}_{g'}{\downarrow_g}\ r = r$$

Of course, it cannot be achieved because this would mean that there is no loss of information through granularity conversion: this is obviously false. If it were true anyway, there would be no need for granularity operators: everything would be the same under any layer. On the other hand, other transitivity such as the oriented transitivity (previously known as cumulated transitivity) can be expected:

$${}_g{\uparrow^{g'}_{g'}}{\uparrow^{g''}}\ r = {}_g{\uparrow^{g''}}\ r \text{ and } {}^g{\downarrow^{g'}_{g'}}{\downarrow_{g''}}\ r = {}^g{\downarrow_{g''}}\ r \qquad \text{(oriented transitivity)}$$

However, in a purely qualitative calculus, the precise granularity (g) is not relevant and this property becomes a property of idempotency of operators:

$$\uparrow\uparrow r = \uparrow r \text{ and } \downarrow\downarrow r = \downarrow r \qquad \text{(idempotency)} \qquad (3.7)$$

At first sight, it could be clever to have non idempotent operators which are less and less precise with granularity conversion. However, if this applies very well to quantitative data, it does not apply for qualitative: the qualitative conversion applies equally for a large granularity conversion and for a small one which is ten times less. If, for instance, in a particular situation, a relationship between two entities is $r$, in a coarser representation it is $r'$ and in an even coarser representation it is $r''$, then $r''$ must be a member of the upward conversion of $r$.

This is because $r''$ is indeed the result of a qualitative conversion from the first representation to the third. Thus, qualitatively, $\uparrow\uparrow=\uparrow$.

If there were no idempotency, converting a relationship directly would give a different result than when doing it through ten successive conversions.

**Representation independence**

Since the operation allowing one to go from a relational space to an interval relational space has been provided (by $\Leftarrow$ and $\Rightarrow$), the property constraining the conversion operators can also be given at that stage: representation independence states that the conversion must not be dependent upon the representation of the temporal entity (as an interval or as a set of bounding points). Again, this property must be required:

$$\rightarrow r = \Leftarrow\rightarrow\Rightarrow r \text{ and } \rightarrow r = \Rightarrow\rightarrow\Leftarrow r \qquad \text{(representation independence)} \qquad (3.8)$$

It can be though of as a distributivity:

$$\Rightarrow\rightarrow r = \rightarrow\Rightarrow r \text{ and } \Leftarrow\rightarrow r = \rightarrow\Leftarrow r$$

Note that, since $\Leftarrow$ requires that the relationship between bounding points allows the result to be an interval, there could be some restrictions on the results (however, these restrictions correspond exactly to the vanishing of an interval which is out of scope here).

The constraints (3.3, self-conservation) and (3.7, idempotence), together with the definition of the operators for full relations (3.2), characterise granularity operators as closure operators.

Nothing ensures that these constraints lead to a unique couple of operators for a given relational system.

**Definition 3.5.1.** *Given a relational system, a couple of operators up-down satisfying 3.3-3.7 is a coherent granularity conversion operator for that system.*

For any relation algebra there are two operators which always satisfy these requirements: the identity function ($Id$) which maps any relation into itself (or a singleton containing itself) and the non-informative function ($Ni$) which maps any relation into the base set of the algebra. It is noteworthy that these functions must then be their own inverse (i.e., they are candidates for both $\uparrow$ and $\downarrow$ at once). These solutions are not considered anymore below.

The framework provided so far concerns two operators related by the constraints, but there is no specificity of the upward or downward operator (this is why constraints are symmetric). By convention, if the system contains an equivalence relation (defined as $e$ such that $e = e \circ e = e^{-1}$ [Hirsh, 1996]), the operators which maps this element to a strictly broader set is denoted as the downward operator. This meets the intuition because the coarser the view the more indistinguishable the entities (and they are then subject to the equivalence relation).

### 3.5.3 Results on point and interval algebras

From these constraints, it is possible to generate the possible operators for a particular relation algebra. This is first performed for the point algebra and the interval algebra in which

it turns out that only one couple of non-trivial operators exists. Moreover, these operators satisfy the relationship between base and interval algebra.

### Granularity for the point algebra

**Proposition 3.5.1.** *Table 3.1 defines the only possible non auto-inverse upward/downward operators for the point algebra.*

| relation: $r$ | $\uparrow r$ | $\downarrow r$ |
|---|---|---|
| $<$ | $<=$ | $<$ |
| $=$ | $=$ | $<=>$ |
| $>$ | $>=$ | $>$ |

Table 3.1: Upward and downward granularity conversions for the point algebra.

These operators fit intuition very well. For instance, if the example of Figure 3.9 is modeled through bounding points ($x^-$ for the left endpoint and $x^+$ for the right endpoint) of intervals $W^+$, $M^-$, $M^+$ and $D^-$, it is represented in (b) by $W^+ = M^-$ (the engine stops working when it starts misfiring), $M^- < M^+$ (the beginning of the misfire is before its end), $M^+ < D^-$ (the end of the misfiring period is before the beginning of the loss of altitude) in (p) by $M^+ = D^-$ (the misfiring period ends when the loss of altitude begins) and in (g) by $M^- = M^+$ (the misfiring period does not exist anymore). This is possible by converting $M^+ < D^-$ into $M^+ = D^-$ ($=\in\uparrow<$) and $M^- = M^+$ into $M^- < M^+$ ($<\in\downarrow=$).

### Granularity for the interval algebra

Since the temporal interval algebra is a plain interval algebra, the constraint 3.8 can be applied for deducing its granularity operators. This provides the only possible operators for the interval algebra. Table 3.2 shows the automatic translation from points to intervals:

| $r$ | $\uparrow r$ | | | | $\uparrow r$ | $\downarrow r$ | | | | $\downarrow r$ |
|---|---|---|---|---|---|---|---|---|---|---|
| $b$ | $<=$ | $<=$ | $<=$ | $<=$ | $bm$ | $<$ | $<$ | $<$ | $<$ | $b$ |
| $d$ | $>=$ | $<=$ | $>=$ | $<=$ | $dsfe$ | $>$ | $<$ | $>$ | $<$ | $d$ |
| $o$ | $<=$ | $<=$ | $>=$ | $<=$ | $osmef^{-1}$ | $<$ | $<$ | $>$ | $<$ | $o$ |
| $s$ | $=$ | $<=$ | $>=$ | $=$ | $se$ | $<=>$ | $<$ | $>$ | $<$ | $osd$ |
| $f$ | $>=$ | $<=$ | $>=$ | $=$ | $fe$ | $>$ | $<$ | $>$ | $<=>$ | $o^{-1}fd$ |
| $m$ | $<=$ | $<=$ | $=$ | $<=$ | $m$ | $<$ | $<$ | $<=>$ | $<$ | $bmo$ |
| $e$ | $=$ | $<=$ | $>=$ | $=$ | $e$ | $<=>$ | $<$ | $>$ | $<=>$ | $of^{-1}d^{-1}s$ $es^{-1}dfo^{-1}$ |

Table 3.2: Transformation of upward and downward operators between points into interval relation quadruples.

The conversion table for the interval algebra is given below. The corresponding operators enjoy the same properties as the operators for the point algebra.

**Proposition 3.5.2.** *The upward/downward operators for the interval algebra of Table 3.3 satisfy the properties 3.3 through 3.7.*

| $r$ | $\uparrow r$ | $\downarrow r$ | $r^{-1}$ | $\uparrow r^{-1}$ | $\downarrow r^{-1}$ |
|---|---|---|---|---|---|
| $b$ | $bm$ | $b$ | $b^{-1}$ | $b^{-1}m^{-1}$ | $b^{-1}$ |
| $d$ | $dfse$ | $d$ | $d^{-1}$ | $d^{-1}s^{-1}f^{-1}e$ | $d^{-1}$ |
| $o$ | $of^{-1}sme$ | $o$ | $o^{-1}$ | $o^{-1}s^{-1}fem^{-1}$ | $o^{-1}$ |
| $s$ | $se$ | $osd$ | $s^{-1}$ | $s^{-1}e$ | $d^{-1}s^{-1}o^{-1}$ |
| $f$ | $fe$ | $dfo^{-1}$ | $f^{-1}$ | $f^{-1}e$ | $d^{-1}f^{-1}o$ |
| $m$ | $m$ | $bmo$ | $m^{-1}$ | $m^{-1}$ | $o^{-1}m^{-1}b^{-1}$ |
| $e$ | $e$ | $of^{-1}d^{-1}ses^{-1}dfo^{-1}$ | | | |

Table 3.3: Upward and downward granularity conversion for the interval algebra.

**Proposition 3.5.3.** *The upward/downward operators for the interval algebra of Table 3.3 are the only ones that satisfy the property 3.8 with regard to the operators for the point algebra of Table 3.1.*

If one wants to generate possible operators for the interval algebra, many of them can be found. But the constraint that this algebra must be the interval algebra (in the sense of [Hirsh, 1996]) of the point algebra restricts drastically the number of solutions.

The reader is invited to check on the example of Figure 3.9, that what has been said about point operators is still valid: the situation (b) is described by $W\{m\}M$ (the working period meets the misfiring one), $M\{b\}D$ (the misfiring period is anterior to the loss of altitude), in (p) by $M\{m\}D$ (the misfiring period meets the loss of altitude) and in (g) where the misfiring period does not appear anymore by $W\{m\}D$ (the working period meets the loss of altitude). This is compatible with the idea that, under a coarser granularity, $b$ can become $m$ ($m \in \uparrow b$) and that under a finer granularity $m$ can become $b$ ($b \in \downarrow m$).

The upward operator does not satisfy the condition 3.4 for B-neighborhood (in which objects are translated continuously [Freksa, 1992]) as it is violated by $d$, $s$, and $f$ and C-neighborhood (in which the objects are continuously expanded or contracted by preserving their center of gravity [Freksa, 1992]) as it is violated by $o$, $s$, and $f$. This is because the corresponding neighborhoods are not based upon independent limit translations while this independence has been used for translating the results from the point algebra to the interval algebra.

It is noteworthy that the downward operator corresponds exactly to the closure of relationships that Ligozat [Ligozat, 1990] introduced in his own formalism. This seems natural since this closure, just like the conversion operators, provides all the adjacents relationships of a higher dimension.

### 3.5.4 General results of existence and composition

We provide here general results about the existence of granularity operators in algebra of binary relations. Then, the relationships between granularity conversion and composition, i.e., the impact of granularity changes on inference results, are considered.

### Existence results for algebras of binary relations

The question of the general existence of granularity conversion operators corresponding to the above constraints can be raised. Concerning granularity conversion operators different from $Id$ and $Ni$, two partial results have been established [Euzenat, 2001]. The first one shows that there are small algebras with no non-trivial operators:

**Proposition 3.5.4.** *The algebra based on two elements $a$ and $a^{-1}$ such that $N(a, a^{-1})$ has no granularity conversion operators other than identity and non-informative map.*

A more interesting result is that of the existence of operators for a large class of algebras. In the case of two auto-inverse operators (e.g., $=$ and $\neq$), there must exist conversion operators as shown by proposition 3.5.5. Proposition 3.5.5 exhibits a systematic way of generating operators from minimal requirements (but does not provide a way to generate all the operators). It only provides a sufficient, but not necessary, condition for having operators.

**Proposition 3.5.5.** *Given a relation algebra containing two relationships $a$ and $b$ such that $N(a, b)$ (it is assumed that neighborhood is converse independent, i.e., $N(a^{-1}, b^{-1})$), there exists a couple of upward/downward granularity operators defined by :*

**if $a$ and $b$ are auto-inverse** $\downarrow a = \{a, b\}$, $\uparrow b = \{a, b\}$, *the remainder being identity;*

**if $a$ only is auto-inverse** $\downarrow a = \{a, b, b^{-1}\}$, $\uparrow b = \{a, b\}$, $\uparrow b^{-1} = \{a, b^{-1}\}$, *the remainder being identity;*

**if $a$ and $b$ are not auto-inverse** $\downarrow a = \{a, b\}$, $\uparrow b = \{a, b\}$, $\downarrow a^{-1} = \{a^{-1}, b^{-1}\}$, $\uparrow b^{-1} = \{a^{-1}, b^{-1}\}$, *the remainder being identity.*

There can be, in general, many possible operators for a given algebra. Proposition 3.5.5 shows that the five core properties of Section 3.5.2 are consistent. Another general question about them concerns their independence. It can be answered affirmatively:

**Proposition 3.5.6.** *The core properties of granularity operators are independent.*

This is proven by providing five systems satisfying all properties but one [Euzenat, 2001].

### Granularity and composition

The composition of symbolic relationships is a favored inference means for symbolic representation systems. One of the properties which would be interesting to obtain is the independence of the results of the inferences from the granularity level (equation 3.9). The distributivity of $\rightarrow$ on $\circ$ denotes the independence of the inferences from the granularity under which they are performed.

$$\rightarrow (r \circ r') = (\rightarrow r) \circ (\rightarrow r') \qquad \text{(distributivity of } \rightarrow \text{ over } \circ\text{)} \qquad (3.9)$$

This property is only satisfied for upward conversion in the point algebra.

**Proposition 3.5.7.** *The upward operator for the point algebra satisfies property 3.9.*

It does not hold true for the interval algebra. Let three intervals $x$, $y$ and $z$ be such that $xby$ and $ydz$. The application of composition of relations gives $x\{b\ o\ m\ d\ s\}z$ which, once upwardly converted, gives $x\{b\ m\ e\ d\ f\ s\ o\ f^{-1}\}z$. By opposition, if the conversion is first applied, it returns $x\{b\ m\}y$ and $y\{d\ f\ s\ e\}z$ which, once composed, yields $x\{b\ o\ m\ d\ s\}z$. The interpretation of this result is the following: by first converting, the information that there exists an interval $y$ forbidding $x$ to finish $z$ is lost; however, if the relationships linking $y$ to $x$ and $z$ are preserved, then the propagation will take them into account and recover the lost precision: $\{b\ m\ e\ d\ f\ s\ o\ ^{-1}\} \circ \{b\ o\ m\ d\ s\} = \{b\ o\ m\ d\ s\}$. In any case, this cannot be enforced since, if the length of $y$ is so small that the conversion makes it vanish, the correct information at that granularity is the one provided by applying first the composition: $x$ can meet the end of $z$ under such a granularity. However, if equation 3.9 cannot be achieved for upward conversion in the interval algebra, upward conversion is super-distributive over composition.

**Proposition 3.5.8.** *The upward operator for the interval algebra satisfies the following property:*

$$(\uparrow r) \circ (\uparrow r') \subseteq \uparrow (r \circ r') \qquad \text{(super-distributivity of } \uparrow \text{ over } \circ\text{)}$$

A similar phenomenon appears with the downward conversion operators (it appears both for points and intervals). Let $x$, $y$ and $z$ be three points such that $x > y$ and $y = z$. On the one hand, the composition of relations gives $x > z$, which is converted to $x > z$ under the finer granularity. On the other hand, the conversion gives $x > y$ and $y<=>z$ because, under a more precise granularity, $y$ could be close but not really equal to $z$. The composition then provides no more information about the relationship between $x$ and $z$ ($x<=>z$). This is the reverse situation as before: it takes into account the fact that the non-distinguishability of two points cannot be ensured under a finer grain. Of course, if everything is converted first, then the result is as precise as possible: downward conversion is sub-distributive over composition.

**Proposition 3.5.9.** *The downward operators for the interval and point algebras satisfy the following property:*

$$\downarrow (r \circ r') \subseteq (\downarrow r) \circ (\downarrow r') \qquad \text{(sub-distributivity of } \downarrow \text{ over } \circ\text{)}$$

These two latter properties can be useful for propagating constraints in order to get out of them the maximum of information quickly. For instance, in the case of upward conversion, if no interval vanishes, every relationship must be first converted and then composed.

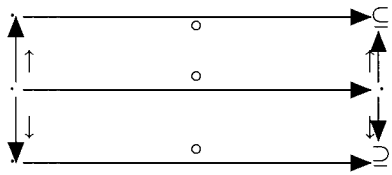



Figure 3.10: A diagrammatic summary of Propositions 3.5.9 and 3.5.8.

These properties have been discovered independently in the qualitative case [Euzenat, 1993] and in the set-theoretic granularity area through an approximation algorithm for quantitative constraints [Bettini *et al.*, 1996].

### 3.5.5 Granularity through discrete approximation

The algebra of relations can be directly given or derived as an interval algebra. It can also be provided by axiomatizing properties of objects or generated from properties of artefacts. Bittner [Bittner, 2002] has taken such an approach for generating sets of relations depending on the join of related objects. He has adapted a framework for qualitatively approximating spatial position to temporal representation. This framework can be used in turn for finding approximate relations between temporal entities which can be seen as relations under a coarser granularity.

#### Qualitative temporal relations

This work is based on a new analysis of the generation of relations between two spatial areas. These relations are characterized through the "intersection" (or meet) between the two regions. More precisely, the relation is characterized by the triple:

$$\langle x \wedge y \not\approx \bot, x \wedge y \approx x, x \wedge y \approx y \rangle$$

The items in these triples characterize the non emptiness of $x \wedge y$ (1st item) and its relation to $x$ and $y$ (2nd and 3rd items). So the values of this triple are relations (this approach is inspired from [Egenhofer and Franzosa, 1991]). These values are taken out of a set of possible relations $\Omega$. This generates several different sets of relations depending on the kind of relations used:

- boundary insensitive relations (RCC5);
- one-dimensional boundary insensitive relations between intervals ($\text{RCC}_1^9$);
- one-dimensional boundary insensitive relations between non convex regions ($\text{RCC}_1^9$);
- boundary sensitive relations (RCC8);
- one-dimensional boundary sensitive relations ($\text{RCC}_1^{15}$).

Some of these representations are obviously refinement of others. In that sense, we obtain a granular representation of a temporal situation by using more or less precise qualitative relationships. This can also be obtained by using other kinds of temporal representations (RCC8 is less precise than Allen's algebra of relations).

As an example, $\text{RCC}_1^9$ considers regions $x$ and $y$ corresponding to intervals on the real line. The set $\Omega$ is made of FLO, FLI, T, FRI, FRO. FLO indicates that no argument is included in the other (O) and there is some part of the first argument left (L) of the second one, FLI indicates that the second argument is included in the first one and there is some part of the first argument left (L) of the second one, T corresponds to the equality of the intersection with the interval, and FRI and FRO are the same for the right hand bound. This provides the relations of Table 3.4.

| $x \wedge y \not\sim \perp$ | $x \wedge y \sim x$ | $x \wedge y \sim y$ | Allen |
|---|---|---|---|
| FLO | FLO | FLO | b m |
| FRO | FRO | FRO | $b^{-1}$ $m^{-1}$ |
| T | FLO | FLO | o |
| T | FRO | FRO | $o^{-1}$ |
| T | T | FLI | d s |
| T | T | FRI | d f |
| T | FLI | T | $d^{-1}$ $f^{-1}$ |
| T | FRI | T | $d^{-1}$ $s^{-1}$ |
| T | T | T | e |

Table 3.4: The relations of $RCC_1^9$.

The relations in these sets are not always jointly exhaustive and pairwise disjoint. For instance, $RCC_1^9$ is exhaustive but not pairwise disjoint, simply because d and $d^{-1}$ appear in two lines of the table.

### Qualitative temporal locations

The framework as it is developed in [Bittner and Steel, 1998] considers a space, here a temporal domain, as a set of places $T_0$. Any spatial or temporal occurrence will be a subset of $T_0$. So, with regard to what has been considered in Section 3.3, the underlying space is aligned and structured.

An approximation is based on the partition of $T_0$ into a set of cells $K$ (i.e., $\forall k, k' \in K, k \subseteq T_0, k \cap k' = \emptyset$ and $\cup_{k \in K} k = T_0$). The localization of any temporal occurrence is then approximated by providing its relation to each cell. The location of $x \subseteq T_0$ is a function $\rho_x : K \rightarrow \Omega'$ from the set of cells to a set of relations $\Omega'$ (which may but have not to correspond to $\Omega$ or a $RCC_q^p$ defined above). The resulting approximation is thus dependent on the partition $K$ and the set of relations $\Omega'$.

From this, we can state that two occurrences $x$ and $y$ are indistinguishable under granularity $\langle K, \Omega' \rangle$ if and only if $\rho_x = \rho_y$. This formulation is typical from the set-theoretic approach to temporal granularity used in a strictly qualitative domain.

We can also define the interpretation of an area of the set of cells $(X : K \rightarrow \Omega)$ as the set of places it approximates:

$$[X] = \{x \subseteq T_0 | \rho_x = X\}$$

### Relations between approximations and granularity

It is clear that the approximation of a region $x$ can be considered as its representation $\uparrow x$ under the granularity $\langle K, \Omega' \rangle$ (i.e., $\rho_x$). In the same vein, the interpretation of approximation $[X]$ corresponds to the conversion of this region to the finer granularity $\downarrow X$. In that respect we are faced with two discrete and aligned granularities.

The following question can be raised: given a relation $r \in RCC_q^p$ between $x$ and $y$, the approximations $\uparrow x$ and $\uparrow y$, and $\uparrow r$ holding between $\uparrow x$ and $\uparrow y$, what can be said of the

relationship between $r$ and $\uparrow r$? The approximate relation $\uparrow r$ holding between $X$ and $Y$ is characterized as $SEM(X,Y)$ and defined as:

$$SEM(X,Y) = \{r \in RCC_q^p | x \in [X], y \in [Y], xRy, \text{ and } r \in R\}$$

The author goes on to define a syntactic operator ($SYN(X,Y)$) for determining the relationships between approximate regions. This operator must be as close as possible to $SEM(X,Y)$. It is defined by replacing in the equations defining the relations of the considered set, the region variables ($x$ and $y$) by approximation variables ($X$ and $Y$) and the meet operation by upper or lower bounds for the meet operation. This provides a pair of values for the relations between $X$ and $Y$ depending on whether they have been computed with the upper and lower meet.

It is now possible to obtain the relations between granular representations of the entities by considering that $x \uparrow r\ y$ can be obtained in the usual way (but for obtaining $\uparrow r$ we need to consider all the possible granularities, i.e., all the possible $K$ and all the possible $\Omega'$). $X \downarrow r\ Y$ is what should be obtained by $SEM(X,Y)$ and approximated by $SYN(X,Y)$.

Hence, a full parallel can be made between the above-described work on qualitative granularity and this work on discrete approximation in general. Unfortunately, the systems developed in [Bittner, 2002] do not include Allen's algebra. The satisfaction of the axioms by this scheme has not been formally established. However, one can say that self-conservation and idempotence are satisfied. Neighborhood compatibility depends on a neighborhood structure, but $SYN(X,Y)$ is very often an interval in the graph of relations (which is not very far from a neighborhood structure). It could also be interesting to show that when $RCC_1^{15}$ relations correspond to Allen's ones, the granularity operators correspond.

In summary, this approximation framework has the merit of providing an approximated representation of temporal places interpreted on the real line. The approximation operation itself relies on aligned granularities. This approach is entirely qualitative in its definition but can account for orientation and boundaries.

## 3.6 Applications of time granularity

Time granularity come into play in many classes of applications with different constraints. Thus, the contributions presented below not only offer an application perspective, but generally provide their own granular formalism. The fact that there are no applications to multi-agent communication means that the agents currently developed communicate with agents of the same kind. With the development of communicating programs, it will become necessary to consider the compatibility of two differently grained descriptions of what they perceive.

### 3.6.1 Natural language processing, planning, and reasoning

The very idea of granularity in artificial intelligence comes from the field of natural language understanding [Hobbs, 1985]. In [Gayral, 1992] Gayral and Grandemange take into account the same temporal unit under a durative or instantaneous aspect. Their work is motivated by problems in text understanding. A mechanism of upward/downward conversion is introduced and modeled in a logical framework. It only manages symbolic constraints and it converts the entities instead of their relationships. The representation they propose is based

on a notion of composition and it allows the recursive decomposition of beginning and ending bounds of intervals into new intervals. The level of granularity is determined during text understanding by the election of a distinguished individual (which could be compared with a focus of attention) among the set of entities and the aspect (durative vs. instantaneous) of that individual. Unlike most of the previously-described approaches, where granularity is considered orthogonal to a knowledge base, in Gayral and Grandemange's work the current granularity is given relatively to the aspect of a particular event. A link between the two notions can be established by means of the decomposition relation between entities (or history [Euzenat, 1993]). Time granularity in natural language processing and its relation with the durative/instantaneous aspects have been also studied by other authors. As an example, Becher et al. model granularity by means of time units and two basic relations over them: precedence and containment (alike the set-theoretic approach, Section 3.3) [Becher *et al.*, 1998]. From a model of time units consisting of a finite sequence of rational numbers, the authors build an algebra of relations between these units, obtaining an algebraic account of granularity.

In [Badaloni and Berati, 1994], Badaloni and Berati use different time scales in an attempt to reduce the complexity of planning problems. The system is purely quantitative and it relies on the work presented in Section 3.3. The NatureTime [Mota *et al.*, 1997] system is used for integrating several ecological models in which the objects are modeled under different time scales. The model is quantitative and it explicitly defines (in Prolog) the conversions from a layer to another. This is basically used during unification when the system unifies the temporal extensions of the atoms. Combi et al. [Combi *et al.*, 1995] applied their multi-granular temporal database to clinical medicine. The system is used for the follow-up of therapies in which data originate from various physicians and the patient itself. It allows one to answer (with possibility of undefined answers) to various questions about the history of the patient. In this system (like in many other) granularity usually means "converting units with alignment problems".

### 3.6.2 Program specification and verification

In [Ciapessoni *et al.*, 1993], Ciapessoni et al. apply the logics of time granularity to the specification and verification of real-time systems. The addition of time granularity makes it possible to associate coarse granularities with high-level modules and fine granularities with the lower level modules that compose them. In [Fiadeiro and Maibaum, 1994], Fiadeiro and Maibaum achieve the same practical goal by considering a system in which granularity is defined a posteriori (it corresponds to the granularity of actions performed by modules, while in the work by Ciapessoni et al. the granularity framework is based on a metric time) and the refinement (granularity change) takes place between classical logic theories instead of inside a specialized logical framework (as in Section 3.4.1). It is worth pointing out that both contributions deal with refinement, in a quite different way, but they do not take into account upward granularity change. Finally, in [Broy, 1997], Broy introduces the notion of temporal refinement into the description of software components in such a way that the behavior of these components is temporally described under a hierarchy of temporal models.

### 3.6.3 Temporal Databases

Time granularity is a long-standing issue in the area of temporal databases (see Chapter 14). As an evidence of the relevance of the notion of time granularity, the database community has released a "glossary of time granularity concepts" [Bettini *et al.*, 1998a]. As we already pointed out, the set-theoretic formalization of granularity (see Section 3.3) has been settled in the database context. Moreover, besides theoretical advances, the database community contributed some meaningful applications of time granularity. As an example, in [Bettini *et al.*, 1998b] Bettini et al. design an architecture for dealing with granularity in federated databases involving various granularities. This work takes advantage of extra information about the database design assumptions in order to characterize the required transformations. The resulting framework is certainly less general than the set-theoretic formalization of time granularity reported in Section 3.3, but it brings granularity to concrete databases applications. Time granularity has also been applied to data mining procedures, namely, to procedures that look for repeating collection of events in federated databases [Bettini *et al.*, 1998d] by solving simple temporal reasoning problems involving time granularities (see Section 3.3). An up-to-date account of the system is given in [Bettini *et al.*, 2003].

### 3.6.4 Granularity in space

(Spatial) granularity plays a major role in geographic information systems. In particular, the granularity for the Region Connection Calculus [Randell *et al.*, 1992; Egenhofer and Franzosa, 1991] has been presented in that context [Euzenat, 1995b]. Moreover, the problem of generalization is heavily related to granularity [Muller *et al.*, 1995]. Generalization consists in converting a terrain representation into a coarser map. This is the work of cartographers, but due to the development of computer representation of the geographic information, the problem is now tackled in a more formal, and automated, way.

In [Topaloglou, 1996], Topaloglou et al. have designed a spatial data model based on points and rectangles. It supports aligned granularities and it is based on numeric constraints. The treatment of granularity consists in tolerant predicates for comparing objects of different granularities which allow two objects to be considered as equals if they only deviate from the granularity ratio.

In [Puppo and Dettori, 1995; Dettori and Puppo, 1996], Puppo and Dettori outline a general approach to the problem of spatial granularity. They represent space as a cell complex (a set of elements with a relation of containment and the notion of dimension as a map to integers) and generalization as a surjective mapping from one complex cell into another. One can consider the elements as simplexes (points of dimension 1, segments of dimension 2 bounded by two points, and triangles of dimension 3 bounded by three segments). This notion of generalization takes into account the possible actions on an object: preservation, if it persists with the same dimension under the coarser granularity, reduction, if it persists at a lower dimension, and immersion, if it disappears (it is then considered as immersed in another object). The impact of these actions on the connected objects is also taken into account through a set of constraints, exactly like it has been done in Section 3.5.2. This should be totally compatible with the two presentations of granularity given here. Other transformations, such as exaggeration (when a road appears larger than it is under the map scale) and displacement, have been taken into account in combination with generalization, but they do not fit well in the granularity framework given in Section 3.2. Last, it must be noted that

these definitions are only algebraic and that no analytical definitions of the transformations have been given.

Other authors have investigated multi-scale spatial databases, where a simplified version of the alignment problem occurs [Rigaux and Scholl, 1995]. It basically consists in the requirement that each partition of the space is a sub-partition of those it is compared with (a sort of spatial alignment).

Finally, some implementations of multi-resolution spatial databases have been developed with encouraging results [Devogele *et al.*, 1996]. As a matter of fact, the addressed problem is simpler than that of generalization, since it consists in matching the elements of two representations of the same space under different resolutions. While generalization requires the application of a (very complex) granularity change operator, this problem only requires to look for compatibility of representations. Tools from databases and generalization can be used here.

## 3.7 Related work

We would like to briefly summarize the links to time granularity coming from a variety of research fields and to provide some additional pointers to less-directly related contributions which have not been fully considered here due to the lack of space. Relationships with research in databases have been discussed in Sections 3.3 and 3.6.3. Granularity as a phenomenon that affects space has been considered in Section 3.6.4. The integration of a notion of granularity into logic programming is dealt with in [Mota *et al.*, 1997; Liu and Orgun, 1997] (see Section 3.6.1 and see also Chapter 13). Work in qualitative reasoning can also be considered as relevant to granularity [Kuipers, 1994] (see Chapter 20).

The relationships between (time) granularity and formal tools for abstraction have been explored in various papers. As an example, Giunchiglia et al. propose a framework for abstraction which applies to a structure $\langle L, A, R\rangle$, where $L$ is a language, $A$ is a set of axioms, and $R$ is a set of inference rules [Giunchiglia *et al.*, 1997]. They restrict abstraction to $A$, because the granularity transformations are constrained to remain within the same language and the same rules apply to any abstraction. One distinctive feature of this work is that it is oriented towards an active abstraction (change of granularity) in order to increase the performance of a system. As a matter of fact, using a coarse representation reduces the problem size by getting rid of details. The approaches to time granularity we presented in this chapter are more oriented towards accounting for the observed effects of granularity changes instead of creating granularity change operators which preserve certain properties.

# Concluding remarks

We would like to conclude this chapter by underlining the relevance and complexity of the notion of time granularity. On the one hand, when some situations can be seen from different viewpoints (of designers, observers, or agents), it is natural to express them under different granularities. On the other hand, problems immediately arise from using multiple granularity, because it is difficult to assign a proper (or, at least, a consistent) meaning to these granular representations.

As it can be seen from above, a lot of work has already been devoted to granularity. This

research work has been developed in various domains (e.g., artificial intelligence, databases, and formal specification) with various tools (e.g., temporal logic, set theory, and algebra of relations). It must be clear that the different approaches share many concepts and results, but they have usually considered different restrictions. The formal models have provided constraints on the interpretations of the temporal statements under a particular granularity, but they did not provide an univocal way to interpret them in a specific application context.

On the theoretical side, further work is required to formally compare and/or integrate the various proposals. On the application side, if the need for granularity handling is acknowledged, it is not very developed in the solutions. There are reasons to think that this will change in the near future, drained by applications such as federated databases and agent systems, providing new problems to theoretical research.

Handbook of Temporal Reasoning in Artificial Intelligence
Edited by M. Fisher, D. Gabbay and L. Vila


# Chapter 4

# Modal Varieties of Temporal Logic

## Howard Barringer & Dov Gabbay

This chapter provides a rudimentary introduction to a number of different systems of temporal logic that have been developed from a modal logic basis. We assume that the reader has some familiarity with propositional and first order logic but assume no background in modal logic, although some reference to modal logic does occasionally occur. Our purpose is to take a tour through a few key "modal" forms of temporal logic, from linear to branching and from points to intervals, present their salient properties and features, for example, syntactic and semantic expressiveness, inference systems, satisfiability and decidability results, and provide sufficient insight into these families of logics to support the interested reader in undertaking further study or to use such logics in practice. The field is vast and there are many other important systems of temporal logic we could have developed; space is limited however and we have therefore focussed primarily on the development on the modal forms of linear time temporal logics.

## 4.1 Introduction

Most interesting systems, be they computational, physical, biological, mental, social, and so on, are dynamic and evolve with time. In order to help understand such evolving systems, specialised languages and logics have been developed over the centuries to reason about and model their dynamic, or temporal, behaviour. Although Aristotle and many later philosophers made major contributions to the debate on the relation between time, truth and possibility, it should be emphasised that much of the formal development of temporal logic has occurred within the last half century, following Prior's seminal work on tense logics. Furthermore, it is the application of temporal logics for reasoning about computational systems that has been the major stimulus for the explosion of research in temporal logics and reasoning over the past three decades. The field is vast and an introductory chapter such as this can only dance lightly across some aspects of the area. Hence here we have chosen to focus on a particular kind of temporal logic, one in which time is effectively abstracted away and special logical operators are used to shift one's attention from one moment to another; we refer to these as modal varieties of temporal logic.

So let us begin our brief tour by asking rhetorically "What is required in order to reason about a system that evolves over time?" Put quite simply, one needs

1. a logic $\mathcal{L}_S$ to reason about the state of the system at some moment in time, and

2. a logic $\mathcal{L}_T$ to reason about how the states of the system at different moments in time are related.

The combination of these two logics, say $\mathcal{L}_T(\mathcal{L}_S)$, which we denote this way since $\mathcal{L}_T$ will be built based upon $\mathcal{L}_S$, we might then think of as a temporal logic. Our interest in this chapter is, however, focussed on the temporal aspects of the logic $\mathcal{L}_T(\mathcal{L}_S)$ and for the sake of simplicity we will in general be treating the logic $\mathcal{L}_S$ as a propositional one.

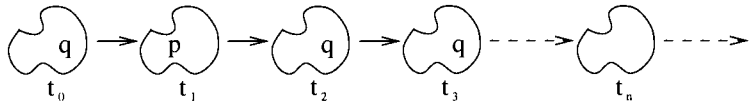
To address the question of what the logic $\mathcal{L}_T$ might be, we need to model the notion of a system whose states are time-dependent and, in the process of doing so, show how the different temporal states are connected. For a fairly general situation a set of system states together with some connection matrix would suffice. As a simple introductory example, let us construct a temporal logic $\mathcal{L}_T(\mathcal{L}_S)$ that can describe some properties of, and hence be used to reason about certain aspects of, a system $S$ that evolves in discrete times steps, $t_0, t_1, t_2, \ldots, t_n, \ldots$. We will draw the time points, $t_n$, from the set of natural numbers and the connection matrix will comply with the usual arithmetic ordering on the numbers. We will take the state logic $\mathcal{L}_S$ to be a propositional logic based on a stock of propositions, PROP, that are used to characterise the state of the system. The logic $\mathcal{L}_T$ however, is a one-sorted first-order logic, where quantification is restricted to individuals ranging over the sort of natural numbers; $\mathcal{L}_T$ will also possess necessary arithmetic functions and relations, e.g. $+$, $-$, $<$, etc. Importantly, however, we equip $\mathcal{L}_T$ with a two-place predicate *holds* that determines whether a formula of $\mathcal{L}_S$ (describing the makeup of the state) holds at a particular time point $t$. Effectively, the embedding of $\mathcal{L}_S$ in $\mathcal{L}_T$ makes formulae of $\mathcal{L}_S$ terms of $\mathcal{L}_T$. Assuming $p$ and $q$ are propositions from PROP, the following formulae of $\mathcal{L}_T$ are examples of the use of the *holds* predicate:

| | |
|---|---|
| $holds(p, t)$ | is true if and only if the proposition $p$ is true at $t$ |
| $holds(p \wedge q, t)$ | is true if and only if the formula $p \wedge q$ is true at $t$ |
| $holds(\neg p \wedge q, t+1)$ | is true if and only if $\neg p \wedge q$ is true at $t+1$, i.e. $p$ is false and $q$ is true at the $t+1$ |

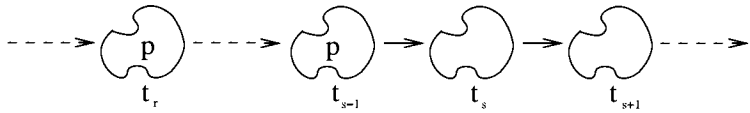
We can then construct formulae such as

(i) $\forall t \cdot holds(p, t) \Rightarrow holds(\neg p \wedge q, t+1)$
(ii) $\forall r \cdot holds(p, r) \Rightarrow \exists s \cdot (r < s \wedge holds(\neg p, s) \wedge \forall t \cdot s < t \Rightarrow holds(\neg p, t))$

The two diagrams below represent an example state evolution that satisfies, respectively, each of the properties above. In the pictorial representation of the state space the absence of $p$ (resp. $q$) from the "state" is used to indicate that $p$ (resp. $q$) is false in that state, whereas, of course, presence of $p$ (resp. $q$) indicates that the proposition holds true. The formula in (i) captures the property that whenever proposition $p$ is true, $p$ will be false in the next moment of time and $q$ will be true. Hence we see in the first evolution that since $p$ is given true at $t_1$, it is false at $t_2$ but $q$ must be true at $t_2$.




The formula in (ii) characterises a system evolution such that if ever the proposition $p$ becomes true, say at time $r$, then there will be a time beyond $r$ from which the proposition $p$ will always be false.




The definition of the *holds* predicate must ensure that the following equivalences hold.

$$\begin{array}{lcl} holds(p \wedge q, t) & \Leftrightarrow & holds(p,t) \wedge holds(q,t) \\ holds(p \vee q, t) & \Leftrightarrow & holds(p,t) \vee holds(q,t) \\ holds(\neg p, t) & \Leftrightarrow & \neg holds(p,t) \\ \vdots & \Leftrightarrow & \end{array}$$

Rather than continue with the use of a meta predicate "holds", we can let the temporal logic $\mathcal{L}_T$ be based on unary predicates $p()$ for each proposition $p$ of $\mathcal{L}_S$ with the proviso that for every time point $t$

$$p(t) \textbf{ iff } holds(p,t)$$

must hold for all $t$. The example formulae from above become:

$$\forall t \cdot p(t) \Rightarrow \neg p(t+1) \wedge q(t+1)$$
$$\forall r \cdot p(r) \Rightarrow \exists s \cdot (r < s \wedge \neg p(s) \wedge \forall t \cdot s < t \Rightarrow \neg p(t))$$

This approach is easily extended to, say, predicate logic for $\mathcal{L}_S$ which results in $\mathcal{L}_T$ being a two-sorted logic - one sort for the time points, and the other capturing the sort of $\mathcal{L}_S$.

One awkwardness with the above form of temporal logic is that it can often be rather difficult to "see the wood for the trees", i.e. to recognise the temporal relations, or connections, being characterised by a given formula. It is not that *natural* a representation. Indeed, although the logic is certainly well able to model a rich expressive set of temporal phenomena, *model* is the key word and the resulting logical expressions do not clearly resemble the concise linguistic temporal patterns we can speak and write. Consider the following statement:

> The dollar has been falling and will continue to fall whilst there is uncertainty in the state of the government.

Let the unary predicate $p$ denote that the dollar is falling and let $q$ denote certainty in the state of the government. A first-order modelling of the above statement will go something like the following, where the individual $n$ represents the current time point (i.e. now):

$$\begin{array}{l} \exists t \cdot t < n \wedge \forall s \cdot t < s < n \Rightarrow p(s) \\ \quad \wedge \\ p(n) \\ \quad \wedge \\ (\forall t \cdot n < t \Rightarrow p(t) \\ \quad \vee \\ \exists t \cdot n < t \wedge q(t) \wedge \forall s \cdot n < s < t \Rightarrow p(s)) \end{array}$$

The first conjunct captures the fact that the dollar has been falling, the second conjunct that the dollar is still falling, and the third conjunct captures that the dollar will continue to fall whilst there is uncertainty in the state of the government. This latter expression captures the

fact that either the dollar continues to fall in perpetuity or there is a future in which there is certainty in the government ($q$) but up to then the dollar had been falling.

As is clear, the formula is cluttered with quantifications and constraints on the time point variables*. The modal varieties of temporal logic, especially tense logic†, overcome these problems through the use of modalities (temporal/tense modal operators). The since-until tense logic (of Kamp), for example, could be used to capture the falling dollar statement as below

$$p \textbf{ since } \mathtt{true} \ \wedge \ p \ \wedge \ p \textbf{ unless } q$$

which is abundantly clearer than the first order logic representation‡. The formula $p$ **unless** $q$ captures the property that either $p$ will hold into the future until $q$ holds or if $q$ never holds at some future point then $p$ will hold forever into the future.

In the following pages of this chapter, we will develop basic systems of the modal varieties of temporal logic. At the start of this subsection we observed that a set of states together with a connection matrix (i.e. a binary relation on states) would suffice to model the notion of time dependent states, where the connection matrix defines the temporal flows. Such a structure is rather similar to a modal frame and in Section 4.2 we introduce the notion of temporal frames and different characterisations of temporal flows.

In order to abstract away from the detail of the temporal model, modal temporal operators (or better, temporal modalities) are defined that can be used to express properties over the network of time-dependent states. Initially we introduce just four primitive temporal modalities:

| | | | |
|---|---|---|---|
| $\boxplus$ | in every future moment | $\boxminus$ | in every past moment |
| $\diamondplus$ | in some future moment | $\diamondminus$ | in some past moment |

For the natural number model of time we were using above, we will define that the formula $\boxplus p$ holds at time point $s$ if and only if for every $t$, $s < t$ implies that $p$ holds at time $t$, i.e. $p$ holds in all future moments of $s$. Whereas, the formula $\diamondplus p$ holds at time point $s$ if and only if there is some time $t$ such that $s < t$ and $p$ holds at $t$, i.e. $p$ holds in some future moment of $s$. Similarly for the past time operators $\boxminus$ and $\diamondminus$. So instead of defining $\mathcal{L}_T$ as a sorted predicate logic, $\mathcal{L}_T$ is, effectively, defined as a multi-modal language - the syntactic and semantic details of these operators are defined in Section 4.2.2. From this basis, Section 4.3 introduces a minimal temporal logic and considers correspondence properties in Section 4.3.1, e.g. if the formula $\diamondplus \diamondplus \varphi \Rightarrow \diamondplus \varphi$ is valid for a temporal frame, then the forwards accessibility relation is transitive, etc..

Although we show in Section 4.3.1 that the temporal operators $\boxplus$, $\diamondplus$, and past time mirrors, are sufficient to characterise a large number of different frame properties§, we show in Section 4.4 that this set is not expressive enough and a range of richer temporal logics is then presented. Section 4.4 restricts attention to linear temporal logics, so in Section 4.5 we explore briefly some aspects of branching time logics. The presentation on modal temporal systems up to that stage primarily focuses on point-based temporal structures and so in Section 4.6 we explore some elements of interval temporal logics using, however, point-based models. We conclude with some pointers to further reading.

---

*And it gets worse when one adds in the necessary formulae that characterise the temporal sort

†Developed for the study of tense in natural language

‡One should note that some advantages will occur through the use of a well-understood first order logic

§some second-order properties as well

## 4.2 Temporal Structures

We begin by introducing the general notion of a *temporal frame*, $\mathcal{F}$, a structure comprising a set of time points, $T$, and two binary relations, $R_f$ and $R_b$, on $T$, i.e. $\subseteq T \times T$.

$$\mathcal{F} = (T, R_f, R_b)$$

$R_f$ relates time points that are connected in a forwards direction of time, i.e. given time points $t_1$ and $t_2$ ($\in T$) if $t_1 R_f t_2$ then $t_1$ is an earlier time than $t_2$. Similarly, $R_b$ relates time points that are connected in a backwards direction of time, i.e. given time points $t_1$ and $t_2$ ($\in T$) if $t_1 R_b t_2$ then $t_1$ is a later time than $t_2$. These binary relations thus determine the temporal flow, that is, the progression of time from one moment to another. The temporal frame, on the other hand, represents a specific model of time and is characterised by the properties of its set of time points $T$ and its binary (flow) relations. For example, for a model of discrete time, where time is clocked forward in explicit jumps, one could simply choose the set $T$ to be a discrete set (of appropriate size).

In many models, or views, of time (or temporal flow) there is no need to have both relations, $R_f$ and $R_b$, distinguished as it is usual that one is the inverse of the other and hence one *earlier than* relation will suffice*. Where this is the case, without loss of generality, we will denote temporal frames as just pairs $\mathcal{F} = (T, R_f)$.

For the temporal flows of time that we will mainly consider in this chapter, i.e. just linear and branching acyclic flows, we add the further constraint that the binary relations should be asymmetric and transitive. The reason for requiring $R_f(R_b)$ to be transitive should be clear given that we intend to model abstractions of real time, i.e. as 1/1/00 is earlier than 2/1/00† which is earlier than 3/1/00, 1/1/00 is also earlier than 3/1/00. Then transitivity combined with asymmetry precludes any cycles in the temporal flow.

The following four examples illustrate a few simple cases of temporal frames.

**Example 4.2.1 (Natural Number Time).** *The temporal frame*

$$\mathcal{F}_{\mathbb{N}} = (\mathbb{N}, <)$$

*where $<$ is the usual less than ordering on numbers represents a model of natural number time, i.e. time that has a beginning, is discrete, linear and future serial (without future end). This model is often used as a temporal abstraction of computation, where each "time point" corresponds to some discrete computation state.*

**Example 4.2.2 (Real Time).** *For those not in favour of the Big Bang theory, the temporal frame*

$$\mathcal{F}_{\mathbf{R}} = (\mathbf{R}, <)$$

*can represent a continuous linear flow of time, without beginning or end; again $<$ is the usual less than ordering on real numbers.*

**Example 4.2.3 (Days of the Week).** *The following frame could be used for a crude model of the days of the week.*

$$\mathcal{F}_{\text{Days}} = (\text{Days}, <_{\text{Days}})$$

*That is, if in our general abstractions of time, $\forall t_1, t_2 \in T \cdot t_1 R_f t_2 \equiv t_2 R_b t_1$ we only need one relation.

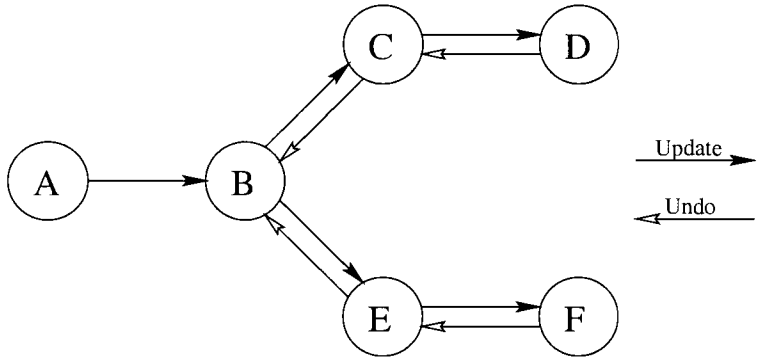
†Assuming the English date format dd/mm/yy.

*where*

$$\begin{array}{lcl} \text{Days} & = & \{\textbf{Sun}, \textbf{Mon}, \textbf{Tue}, \textbf{Wed}, \textbf{Thu}, \textbf{Fri}, \textbf{Sat}\} \\ <_{\text{Days}} & = & \{ \quad (\textbf{Sun}, \textbf{Mon}), (\textbf{Sun}, \textbf{Tue}), (\textbf{Sun}, \textbf{Wed}), \ldots, (\textbf{Sun}, \textbf{Sat}) \\ & & \quad\; (\textbf{Mon}, \textbf{Tue}), (\textbf{Mon}, \textbf{Wed}), (\textbf{Mon}, \textbf{Thu}), \ldots, (\textbf{Mon}, \textbf{Sat}) \\ & & \quad\; (\textbf{Tue}, \textbf{Wed}), (\textbf{Tue}, \textbf{Thu}), (\textbf{Tue}, \textbf{Fri}), \ldots, (\textbf{Tue}, \textbf{Sat}) \\ & & \quad\; \vdots \\ & & \quad\; (\textbf{Fri}, \textbf{Sat}) \\ & & \} \end{array}$$

*So with our week beginning on Sunday and ending on Saturday, we have the usual, or expected, orderings that Monday comes before Tuesday, Tuesday before Thursday, etc., but note that Saturday no longer comes before Sunday!*

**Example 4.2.4 (Database Updates).** *As a rather different example* where one might wish not to maintain $R_f$ and $R_b$ as inverses, consider a typical situation in database updating. First assume a discrete set of database states and then take the forwards temporal flow as database updates. As there may well be certain updates that can't be undone (i.e. rolled back), one might choose to model this as discontinuities in the backwards temporal flow.*




*To keep the above picture uncluttered the full transitive relations for Update and Undo have not been shown, only their principal elements. More formally, it represents the temporal frame*

$$\mathcal{F}_{DB} = (States, Update, Undo)$$

*where*

$$\begin{array}{lcl} States & = & \{A, B, C, D, E\} \\ Update & = & \{ \quad (A,B), (A,C), (A,D), (A,E), (A,F) \\ & & \quad\; (B,C), (B,D), (C,D), \\ & & \quad\; (B,E), (B,F), (E,F) \\ & & \} \\ Undo & = & \{ \quad (D,C), (D,B), (C,B) \\ & & \quad\; (F,E), (F,B), (E,B) \\ & & \} \end{array}$$

* And perhaps a little contrived from a purely temporal standpoint

*As can be clearly seen, the update from state A to B represents a commit which can not be undone.*

So far temporal frames appear very similar to (multi-)modal frames; in that context the set of possible worlds is the set of time points, and the accessibility relation is the earlier/later than relation. The principal difference is that we have restricted the accessibility relation to be transitive. In light of this it is not unreasonable to ask whether the frame of Example 4.2.4 is really a temporal frame, rather than a multi-modal frame? Not wishing to embark upon such philosophical discussion here, in what follows we will assume not. Indeed, we will adopt the following minimal requirements for temporal frames.

**Invertibility:** now is in the past of every future moment and now is in the future of every past moment;

**Antisymmetry:** a future moment of some moment in time can not also be a past moment of that moment, and vice-versa;

**Transitivity:** a future moment of a future moment of now is also a future moment of now, and similarly for the past.

Figure 4.1: Minimal Constraints for Temporal Frames

The above constraints characterise what we generally take for granted when we reason about time, at least the temporal flow in which we live. Obviously, one might dream up models of circular time, where one's past can be reached by going into the future (indeed it might even be possible one day by backward time travel, though if it were we might have had some sign about it already!). We will stick with the intuitive, natural, minimal constraints and henceforth we shall use "$<$" to denote the "earlier than/later than" temporal binary relation on time points.

### 4.2.1 Temporal Frame Properties

In the above section we presented a few specific illustrations of different temporal frames by choosing particular sets of time points and temporal relations. The set of time points in each had well known properties, e.g. the discreteness of the natural numbers and integers, the continuity of the reals. It is more useful, however, to give formal, logical, characterisations of the frame properties and then characterise *classes* of temporal frames according to their properties. For example, we may be interested in the class of all asymmetric, transitive, weakly dense temporal frames (which includes the frames $(\mathbb{Q}, <)$, $(\mathbf{R}, <)$, etc.). Table 4.1 presents a number of interesting frame properties together with a formal characterisation in predicate logic (first order for all but the wellfoundedness property).
The first three properties, namely **irreflexivity**, **asymmetry** and **transitivity**, are clear. For the others, however, a brief explanation is in order. **Future (past) seriality** characterises the property that time has no ending (beginning). **Maximal (minimal) points** characterises that time does have some ends (beginnings). Beware that maximal (minimal) points do not characterise future (past) finiteness of time. The time may be branching and the property requires

*This and the mirror property are often referred to as right (resp. left) linearity

| Property | Formal Characterisation |
|---|---|
| **irreflexivity** | $\forall t \in T \cdot \neg(t < t)$ |
| **asymmetry** | $\forall s, t \in T \cdot \neg(s < t \wedge t < s)$ |
| **transitivity** | $\forall s, t, u \in T \cdot s < t \wedge t < u \Rightarrow s < u$ |
| **future serial** | $\forall s \in T \cdot \exists t \in T \cdot s < t$ |
| **past serial** | $\forall s \in T \cdot \exists t \in T \cdot t < s$ |
| **maximal points** | $\exists s \in T \cdot \neg\exists t \in T \cdot s < t$ |
| **minimal points** | $\exists s \in T \cdot \neg\exists t \in T \cdot t < s$ |
| **connectedness** | $\forall s, t \in T \cdot s < t \vee s = t \vee t < s$ |
| **weak future connectedness*** | $\forall s, t, u \in T \cdot (s < t \wedge s < u) \Rightarrow (t < u \vee t = u \vee u < t)$ |
| **weak past connectedness** | $\forall s, t, u \in T \cdot (t < s \wedge u < s) \Rightarrow (t < u \vee t = u \vee u < t)$ |
| **successors** | $\forall s \in T \cdot \exists t \in T \cdot s < t \wedge \neg\exists u \in T \cdot s < u \wedge u < t$ |
| **predecessors** | $\forall s \in T \cdot \exists t \in T \cdot t < s \wedge \neg\exists u \in T \cdot t < u \wedge u < s$ |
| **weakly dense** | $\forall s, u \in T \cdot s < u \Rightarrow \exists t \in T \cdot s < t \wedge t < u$ |
| **weakly dense with breaks** | $\forall s, t, u \in T \cdot s < t < u \Rightarrow \exists v \in T \cdot s < v < t \vee t < v < u$ |
| **wellfoundedness** | $\forall P \cdot \exists t \in T \cdot P(t) \Rightarrow$ $\exists t \in T \cdot (P(t) \wedge \forall s \in T \cdot (s < t \Rightarrow \neg P(s)))$ |

Table 4.1: Temporal Frame Properties

only that some point is a dead end (beginning) and thus some paths through the structure may be serial. **Connectedness** characterises that every pair of different time points are ordered by the temporal relation. Finiteness of time, on the other hand, has to be characterised by a second order property (**wellfoundedness** in the table).

The temporal frame of Example 4.2.1 (Natural Number Time) satisfies the properties irreflexivity, asymmetry, transitivity, future seriality, minimal points, connectedness, weak future connectedness, weak past connectedness, successors, predecessors and wellfoundedness.

The temporal frame of Example 4.2.2 (Real Time) satisfies irreflexivity, asymmetry, transitivity, future seriality, past seriality, connectedness, weak future connectedness, weak past connectedness and weak density.

Note that the temporal frame $(\mathbb{Q}, <)$ that is based on the rationals also satisfies the list given just above for the real time frame. However, it should be remembered that the temporal frame $(\mathbf{R}, <)$ also satisfies a completeness property[†], which is not the case for the rationals.

## 4.2.2 Temporal Language, Models and Interpretations

Let us now introduce our base temporal logic language. As indicated in the previous section, we wish to use temporal modalities $\boxplus$, $\boxminus$, $\diamondplus$ and $\diamondminus$. More formally, let the temporal language $\mathcal{L}_{(\boxplus, \boxminus)}$ be the set of formulas defined inductively by the following formation rules:

[†] i.e. $\forall P \cdot \exists s \in T \cdot P(s) \wedge \exists s \in T \cdot \neg P(s) \wedge (\forall s \in T \cdot \forall t \in T \cdot (P(s) \wedge \neg P(t) \Rightarrow s < t)) \Rightarrow \exists s \in T \cdot (P(s) \wedge \forall t \in T \cdot (s < t \Rightarrow \neg P(t))) \vee \exists s \in T \cdot (P(s) \wedge \forall t \in T \cdot (t < s \Rightarrow \neg P(t)))$

(i) $p$ is in $\mathcal{L}_{(\boxplus, \boxminus)}$ for any atomic proposition $p$ drawn from the stock, PROP;

(ii) If $\varphi$ and $\psi$ are formulae in $\mathcal{L}_{(\boxplus, \boxminus)}$, then so are the Boolean combinations $\neg\varphi$, $\varphi \wedge \psi$, $\varphi \vee \psi$, $\varphi \Rightarrow \psi$ and $\varphi \Leftrightarrow \psi$ in $\mathcal{L}_{(\boxplus, \boxminus)}$;

(iii) If $\varphi$ is a formula of $\mathcal{L}_{(\boxplus, \boxminus)}$, then $\boxplus\varphi$, $\boxminus\varphi$, $\diamondplus\,\varphi$ and $\diamondminus\,\varphi$ are also formulae in $\mathcal{L}_{(\boxplus, \boxminus)}$.

Thus, assuming that Lhb, Lhs, Lgdg are atomic propositions and hence drawn from the stock PROP, the following are examples of formulae in $\mathcal{L}_{(\boxplus, \boxminus)}$

| | | |
|---|---|---|
| Lhb | $\diamondplus$ Lhb | Lhb $\wedge$ $\neg$Lhs |
| $\diamondplus$ (Lhb $\wedge$ $\diamondplus$ Lgdg) | $\diamondminus$ Lhs | $\diamondplus$ (Lgdg $\wedge$ $\diamondminus$ (Lhb $\wedge$ $\diamondminus$ Lhs)) |
| $\diamondminus$ (Lhs $\wedge$ $\neg$Lhb) | $\diamondplus$ ($\neg$Lgdg $\wedge$ $\neg$Lhb $\wedge$ $\neg$Lhs) | $\neg\boxminus$Lhs |
| $\neg\boxplus\,\diamondplus$ Lhb | $\diamondminus\,\boxminus\neg$Lhs | $\boxplus$(Lgdg $\Rightarrow$ $\diamondminus$ Lhs) |

**Semantics** A model for a logical formula must provide the information necessary to interpret fully that formula. We have seen how temporal frames, and properties placed upon them, can provide an underlying temporal structure, or network of time points, over which the temporal connectives of the language will be interpreted. In addition to the temporal frame, a valuation function is required for the propositions of the language. In the case of propositional logic, where formulae are effectively interpreted in a single world, the valuation function is just a truth valuation function. For the temporal logic case, propositions may have different interpretations at different points in time - i.e. they are not statically interpreted. Thus the valuation function must provide the set of time points at which any given proposition holds true. A model for temporal logic formulae is therefore taken as a structure combining a temporal frame with a valuation function.

$$M = (\underbrace{T, <}_{\text{temporal frame}}, \underbrace{V}_{\substack{\text{valuation function} \\ \text{PROP} \rightarrow 2^T}})$$

We can now define the interpretation of formulae of the temporal language $\mathcal{L}_{(\boxplus, \boxminus)}$ in the above model structure. Let $\models$ be a satisfaction relation between a model time-point pair $(M, s)$ and a temporal formula $\varphi$, i.e. $M, s \models \varphi$ means that $\varphi$ is true in model $M$ at time point $s$. As is to be expected the interpretation is defined inductively over the structure of the temporal formulae.

- For $\varphi$ being an atomic proposition $p$ drawn from PROP, we have

  $$M, s \models p \textbf{ iff } s \in V(M)(p)$$

  We use the notation $V(M)$ to denote the valuation function of the model $M$. $V(M)(p)$ thus yields the set of time points at which the proposition $p$ holds true. Thus the model $M$ satisfies $p$ at time point $s$ if and only if $s$ is a time point at which the proposition holds true according to the valuation.

  As it is only the interpretation of proposition symbols which requires access to the valuation, for notational convenience, the reference to the model $M$ is dropped from the interpretations for propositional and temporal connectives making it clear, especially in the case of temporal connectives, that their interpretation is dependent upon the particular time point in the model $M$.

- Assuming that $\psi$ and $\phi$ are formulae of $\mathcal{L}_{(\boxplus, \boxminus)}$, for the propositional connectives the inductive cases are standard as below.

$$\begin{array}{lll} s \models \phi \wedge \psi & \textbf{iff} & s \models \phi \text{ and } s \models \psi \\ s \models \phi \vee \psi & \textbf{iff} & s \models \phi \text{ or } s \models \psi \\ s \models \neg\phi & \textbf{iff} & \text{it is not the case that } s \models \phi \\ \vdots & \vdots & \vdots \end{array}$$

- The interesting cases are for the temporal connectives, $\boxplus$, $\boxminus$, $\diamondplus$ and $\diamondminus$.

$$\begin{array}{lll} s \models \boxplus\varphi & \textbf{iff} & \text{for every } t, s < t \text{ implies } t \models \varphi \\ s \models \boxminus\varphi & \textbf{iff} & \text{for every } t, t < s \text{ implies } t \models \varphi \\ s \models \diamondplus\varphi & \textbf{iff} & \text{there exists some } t, s < t \text{ and } t \models \varphi \\ s \models \diamondminus\varphi & \textbf{iff} & \text{there exists some } t, t < s \text{ and } t \models \varphi \end{array}$$

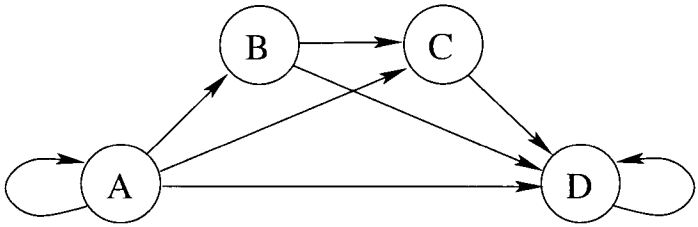
Note that:

$\boxplus$, always in the future, has the usual modal interpretation. Namely, for $\boxplus\phi$ to be true at point $s \in T$ in a model $(T, <, V)$, then $\phi$ must be true at all points $t \in T$ reachable by $<$ from $s$.

$\diamondplus$, sometime in the future, also has the normal modal interpretation. So, $\diamondplus\,\phi$ is true at $s \in T$ in model $(T, <, V)$ if and only if there is a point $t \in T$ reachable from $s$ by $<$ (i.e. later than $s$) at which $\phi$ is true.

The past time connectives $\boxminus$ and $\diamondminus$ have mirror definitions.

**Example 4.2.5 (Interpretation exercises).** *Assume a model $M$ with time points $\{A, B, C, D\}$ and the relation $<$ given as below*




*and the valuation $V$ is*

$$p \mapsto \{A, C, D\}, q \mapsto \{A, B\}, r \mapsto \{B, C, D\}$$

1. *Consider the interpretation of the formula $\diamondplus\boxplus r$ at node (time point) $A$. For it to be true at $A$ in the given model, we must be able to move to a node (time point) in the network, say $n$ at which $\boxplus r$ is true. For the latter formula to be true, all (future) reachable nodes must have $r$ true. Choose the node $n$ to be node $B$. This clearly satisfies the constraints.*

   *Notice that the formula $\boxplus r$ is not true at $A$. $A$ is reachable from itself, and as $r$ is not true at $A$, it will contradict the requirement that $r$ is true at all (future) reachable nodes.*

2. *Now consider the interpretation of* $\boxminus(p \wedge q)$ *at node A. Here we require that the formula* $p \wedge q$ *be true in all nodes which precede, i.e. can reach node A. There is only one node preceding node A and that is just node A. Since both p and q are true in A, so is* $\boxminus(p \wedge q)$.
3. *For* $\diamondminus r$ *to be true at A, there needs to be a node that can reach A at which r evaluates true. The only node which can reach A is A itself. r is not true at A and therefore neither is* $\diamondminus r$.
4. *Consider* $\boxplus r$ *at D. This requires that r be true in reachable nodes from D. Similar to the above, the only reachable node from D is D itself. However in this case r evaluates true at D, therefore* $\boxplus r$ *is true at D.*
5. *For* $\diamondminus \boxplus r$ *to be true at D we must be able to find a node in D's past at which* $\boxplus r$ *is true. One could choose node B or C. Note that node A will* not *do as r is not necessarily true in the future, A is in the future of itself.*
6. *Finally consider* $\boxminus(p \vee q)$ *at node D. Since all nodes in the model satisfy* $p \vee q$*, all nodes that can reach D must also satisfy* $p \vee q$*. Therefore* $\boxminus(p \vee q)$ *is true at node D.*

Now that we have defined the interpretation of temporal formulae against our model structures we are in a position to define other (standard) notions, e.g. validity. Like other modal logics we define three notions of validity:

**Definition 4.2.1 (Model Validity).** *A formula* $\varphi$ *is said to be model valid if it holds true at every time point of the model* $M$*. Formally,*

$$M \models \varphi \text{ iff for every } t \in T(M),\ M, t \models \varphi$$

*We'll refer to formulae being* $M$*-valid.*

**Definition 4.2.2 (Frame Validity).** *A formula* $\varphi$ *is said to be frame valid if it is model valid for every model of the frame, i.e. it holds true for possible valuation and at every possible time point of the temporal frame.*

$$F \models \varphi \text{ iff for every } M = (F, V),\ M \models \varphi$$

*As above, we refer to a formula being* $F$*-valid. Clearly frame validity implies model validity, but not vice-versa.*

And finally it is useful to define validity for a class of frames (satisfying some property, e.g. discrete frames). Obviously a formula is valid (in the unrestricted sense) if it is valid for all possible frames.

**Definition 4.2.3 (Class Validity).** *A formula* $\varphi$ *is said to be valid wrt a class* $C$ *of frames* $F$ *if it is frame valid for each* $F$ *in* $C$*, i.e.*

$$C \models \varphi \text{ iff for every frame } F \in C,\ F \models \varphi$$

**Example 4.2.6 (Exercise in validity).** *Consider the model* $M = (\mathbb{N}, <, V)$ *where* $V(p) = \mathbb{N}$*, i.e. atomic propositions are true everywhere, the following five formulae are all* $M$*-valid*

$$p \quad \boxplus p \quad \diamondminus p \boxminus p \quad \boxplus \diamondminus p \quad \boxplus \boxminus p$$

*One might be led to the false conclusion that since every proposition is true everywhere in this particular model, every temporal formula will also be true. This isn't the case because our model has an asymmetry between the future and the past whereas in our language* $\mathcal{L}_{(\boxplus, \boxminus)}$ *the past time connectives are proper mirrors of their future counterparts. Indeed the*

*formula $\Diamond p$ is not valid on $M$. Why not? Because at time point 0, $\Diamond p$ is false (and for all other points it is true).*

*Consider a dense temporal frame $F$, such as $(\mathbf{R}, <)$, i.e. for any two points $u$, $v$, there is always a point $w$ such that $u < w$ and $w < v$ ($w$ lies strictly between $u$ and $v$). The temporal formula $\Diamond \phi \Rightarrow \Diamond \Diamond \phi$ is frame valid for $F$.*

*Indeed for the class of weakly dense frames, i.e. for any two points $u$, $v$, there is always a point $w$ such that $u < w$ and $w < v$, the above formula $\Diamond \phi \Rightarrow \Diamond \Diamond \phi$ is class valid.*

*The usual notion of validity, namely a formula $\phi$ is valid iff it is true in all models, still applies. Indeed, the formula would be valid for all possible frames, etc. Tautologies, e.g. $\varphi \Rightarrow \varphi$ are clearly examples of valid temporal formulae. A more interesting temporal example is the formula $\Diamond \Diamond \phi \Rightarrow \Diamond \phi$. It is temporally valid since we have defined temporal frames to be those that satisfy a minimal number of constraints (Figure 4.1), in particular, temporal frames are transitive; the given formula is valid on every transitive frame (see later Section 4.3.1 on temporal correspondences). However, we generally apply validity in a restricted context, namely relative to a particular frame, or class of frames.*

## 4.3 A Minimal Temporal Logic

In the above section we introduced minimal constraints that should hold on a frame for the frame to be referred to as temporal, namely invertibility, antisymmetry and transitivity of the ordering relation. In this section we reflect these constraints in our temporal language $\mathcal{L}_{(\boxdot, \boxminus)}$ and define as a result the minimal temporal logic*, $K_T$. Similar to the minimal constraints on temporal frames, we require:-

**A** now is in the past of every future moment and now is in the future of every past moment.

**B** a future moment of a future moment of now is also a future moment of now. Similarly for past.

**K** If, in all future moments $\phi \Rightarrow \psi$ is true and in all future moments $\phi$ holds true Then, in all future moments $\psi$ will hold true. Similarly for the past.

The conditions **A** and **B** are indeed met by our given notion of temporal frame $(T, <)$. In particular, requirement **A** (invertibility) is clearly satisfied by virtue of just using the relation $<$ as the accessibility (or earlier/later than) relation. Of course if two relations had been given (as in some presentations of temporal logic), say $\overset{+}{\rightarrow}$ and $\overset{-}{\leftarrow}$ then we would require the model constraint,

$$\forall t_1, t_2 \in T \cdot t_1 \overset{+}{\rightarrow} t_2 \Leftrightarrow t_1 \overset{-}{\leftarrow} t_2$$

Requirement **B** is simply a transitivity condition. Again, $<$ is given as a transitive relation and also usually taken as irreflexive. The requirement **K** is a normality condition, again similar to that usually required of modal logics.

We'll present temporal logics via axioms and inference rules†. In fact, all the temporal systems we'll consider will be classical and hence include all "propositional" tautologies.

*This is similar in spirit to the way that a minimal modal logic, system K, is developed.

†In actual fact we use finite sets of axiom and inference rule schemata, and an axiomatisation is obtained by taking all substitution instances over the appropriate alphabets.

The requirements **A** and **B** correspond to the formulae Cf, Cp and 4f, 4p, respectively, treated as axioms. The normality constraint **K** is captured by the axioms $Kf$, $Kp$. Thus the axiom schemata (where $\phi$, $\psi$, $\varphi$, etc., are meta-variables) together with "necessitation" inference rules for $\boxplus$ and $\boxminus$, and Modus Ponens gives what is referred to as the **Minimal Temporal Logic**—$K_T$.

**Axioms:**

Tautologies

$Cf$ $\quad \varphi \Rightarrow \boxplus \diamondplus \varphi$

$Cp$ $\quad \varphi \Rightarrow \boxminus \diamondminus \varphi$

$4f$ $\quad \diamondplus \diamondplus \varphi \Rightarrow \diamondplus \varphi$

$4p$ $\quad \diamondminus \diamondminus \varphi \Rightarrow \diamondminus \varphi$

$Kf$ $\quad \boxplus(\varphi \Rightarrow \psi) \Rightarrow (\boxplus \varphi \Rightarrow \boxplus \psi)$

$Kp$ $\quad \boxminus(\varphi \Rightarrow \psi) \Rightarrow (\boxminus \varphi \Rightarrow \boxminus \psi)$

**Inference Rules:**

$MP$ $\quad \dfrac{\vdash \varphi \qquad \vdash \varphi \Rightarrow \psi}{\vdash \psi}$

$\boxplus - Gen$ $\quad \dfrac{\vdash \varphi}{\vdash \boxplus \varphi}$

$\boxminus - Gen$ $\quad \dfrac{\vdash \varphi}{\vdash \boxminus \varphi}$

**Example 4.3.1 (Transitivity).** *Above we stated that the axiom $4f$ characterises transitive frames - let us formally establish that property. Firstly we consider the easier case, namely given an arbitrary transitive frame we prove that the formula $4f$ is indeed valid for that frame. Then we'll prove that if $4f$ is frame valid, then the frame's accessibility relation is transitive.*

**only if case:** *Let $F = (T, <)$ be an arbitrary transitive frame. We show that $4f$ is frame valid. Let $V$ be an arbitrary valuation and $s$, $t$ and $u$ arbitrary members of $T$ such that $s < t$ and $t < u$. Assume that $((T, <), V), u \models \varphi$. Therefore by the interpretation definition for $\diamondplus$, we have $((T, <), V), t \models \diamondplus \varphi$. By similar reasoning, we have $((T, <), V), s \models \diamondplus \diamondplus \varphi$. By the definition of $\Rightarrow$, for the formula $4f$ to be true at $s$, we must show that $((T, <), V), s \models \diamondplus \varphi$. By the transitivity of the frame $F$, as $s < t$ and $t < u$, we have that $s < u$. Since we were given that $\varphi$ held at time point $u$, we thus have $\diamondplus \varphi$ holding at $s$. Hence $4f$ holds at time point $s$. Since both $V$ and $s$ were arbitrary, $4f$ is frame valid. Hence the desired result.*

**if case:** *Now we are given that for some frame $F = (T, <)$, the formula $4f$ is $F$-valid. Thus for arbitrary valuation function $V$ and arbitrary time point $s$, we have $((T, <), V), s \models \diamondplus \diamondplus \varphi \Rightarrow \diamondplus \varphi$. We need only consider the case when the antecedent of $4f$ is true. Without loss of generality, assume a valuation $V$ such that $\varphi$ is only true at time point $u$. Since $\diamondplus \diamondplus \varphi$ holds at $s$, there must also be a time point $t$, such that $s < t$ and $t < u$ such that $\diamondplus \varphi$ holds at $t$. By the frame validity of $4f$, it must also be the case that $\diamondplus \varphi$ holds at time point $s$. Therefore by the interpretation definition for $\diamondplus$, it must be the case that $s < u$. Therefore we have established that the accessibility relation $<$ is transitive. Hence the result.*

A similar proof can be produced for $4p$. We leave that as an exercise for the reader and also the proof to establish that the axioms $Cf(Cp)$ correspond to the invertibility properties.

### More modalities

For notational convenience, in Table 4.2 we define the following additional temporal modalities in terms of the existing modalities and connectives. Of course this expansion in the number of temporal modalities does not increase the semantic expressiveness of the temporal logic, but it does improve the syntactic compactness and structural expressiveness of the language. Later in section 4.4 we consider extending the semantic expressiveness of the language.

| | | |
|---|---|---|
| $\boxplus\varphi$ | $\stackrel{def}{=}$ | $\boxminus\varphi \wedge \varphi \wedge \boxplus\varphi$ |
| $\diamondplus\varphi$ | $\stackrel{def}{=}$ | $\diamondminus\varphi \vee \varphi \vee \diamondplus\varphi$ |
| $\Box\varphi$ | $\stackrel{def}{=}$ | $\varphi \wedge \boxplus\varphi$ |
| $\Diamond\varphi$ | $\stackrel{def}{=}$ | $\varphi \vee \diamondplus\varphi$ |
| $\blacksquare\varphi$ | $\stackrel{def}{=}$ | $\varphi \wedge \boxminus\varphi$ |
| $\blacklozenge\varphi$ | $\stackrel{def}{=}$ | $\varphi \vee \diamondminus\varphi$ |

Table 4.2: More temporal modalities

So now we've introduced five variations of the "box" modality and five variations of "diamond" modality. It is useful to remember the variations as follows:

- $\boxplus$ takes one everywhere reachable in the *strict* future (not including the present)
- $\boxminus$ takes one everywhere reachable in the *strict* past (not including the present)
- $\Box$ takes one everywhere reachable in the present and future
- $\blacksquare$ takes one everywhere reachable in the present and past
- $\boxplus$ takes one everywhere that is reachable, be it in the past, present or future.

Similarly for the diamond modality taking one somewhere ....

### Duals and Mirrors

The observant reader will have noted that our temporal logic is propositionally classical - look back at the interpretation definition for the boolean connectives again if you're not convinced (page 128). This therefore leads to notions of temporal duality that are similar to those for classical propositional and predicate calculus, i.e. where $\wedge$ and $\vee$ are duals, and $\forall$ and $\exists$ are duals. Indeed we refer to $\diamondplus$ as the dual temporal modality of $\boxplus$ and vice-versa. This is because the temporal formula $\boxplus\varphi \equiv \neg \diamondplus \neg\varphi$ is valid (i.e. true for all models). We'll establish one direction of the equivalence here and leave the other as an easy exercise. For convenience we omit the detailed reference to the model structure. Picking an arbitrary time point $s$, if $\boxplus\varphi$ is true at $s$, then by definition $\varphi$ is true at all $t$, s.t. $s < t$. Therefore there is

no $w$, $s < w$ s.t. $\neg\varphi$ is true, in other words, by the interpretation definition of $\Diamond$, $\neg \Diamond \neg\varphi$ holds at time $s$. Hence the result. The other direction is just as straightforward.

Clearly we have the following pairs of temporal modalities as duals of each other: $\boxminus$ and $\Diamond$; $\blacksquare$ and $\blacklozenge$; $\Box$ and $\Diamond$; $\boxplus$ and $\Diamond$.

Note that the complement of a temporal formula consisting of a prefix of temporal modalities, say $T_i$, applied to formula $\phi$, can always be written as the string of the duals of the prefix $T_i$ applied to $\neg\phi$. So for example, the complement of $\Diamond \Diamond \boxplus \Diamond \boxplus \boxminus\phi$ would be $\boxminus \boxminus \Diamond \boxminus \Diamond \Diamond \neg\phi$.

We also introduce the notion of a *Mirror* image of a temporal formula. It is obtained by interchanging the past connectives with their future counterparts, and vice-versa. For example assuming that $\varphi$ is some boolean combination of atomic propositions,

$$\Diamond \Box\varphi \Rightarrow \Diamond \varphi$$

has mirror image

$$\Diamond \blacksquare\varphi \Rightarrow \Diamond \varphi$$

The mirror image of a formula can be thought of as just the formula's reflection about now. The inductive definition of mirror is left as an exercise.

### 4.3.1 Temporal Correspondences

In Section 4.2.1 we showed how classes of temporal frames can be specified by first-order properties over the frame, and in Example 4.3.1 we established that the axiom $4f$ of the minimal temporal logic $K_T$ does indeed determine transitive frames. In this section we refer back to more of the frame properties listed in Table 4.1 and show how, for many, each can be characterised by a temporal formula. The existence of such characteristic formulae leads to one way to develop different forms of temporal logic. The minimal temporal logic $K_T$ placed minimal constraints on the frames. By adding further axioms, each corresponding to specific properties such as seriality, or weak density, etc., we can obtain richer forms of temporal logic. Does this always work? That is to say, as axioms are added to the system, is the resulting logic complete with respect to the union of the properties represented by the axioms? In the final subsection we provide an example where this is not the case.

**transitivity** The formula $\Diamond \Diamond \varphi \Rightarrow \Diamond \varphi$ (or its mirror) has already been considered. A different formulation is $\Diamond \varphi \Rightarrow \boxminus \Diamond \varphi$ (or its mirror). We'll establish one direction of the proof, namely showing the formula is valid on transitive frames, leaving the other direction as an exercise. So consider an arbitrary transitive frame $F = (T, <)$, with valuation $V$ and some time point $t$. If $\Diamond \varphi$ is false at $t$, then the original formula is true. More interestingly, if $\Diamond \varphi$ is true at $t$, then there is some point $u$ beyond $t$ such that $\varphi$ holds true at $u$. For $\boxminus \Diamond \varphi$ to hold true at $t$, we require, by the interpretation definition of $\boxminus$ that for every time point $s$ strictly before $t$ we have $\Diamond \varphi$ holding true. This is almost trivially the case since by the transitivity of the accessibility relation, $<$, from $s < t$ and $t < u$ we have that $s < u$ - and hence the desired result.

**weak future connectedness** We characterised this frame property as

$$\forall s, t, u \in T \cdot (s < t \wedge s < u) \Rightarrow (t < u \vee t = u \vee u < t)$$

Three possible temporal logic formulas determining such frames are as follows.

$$\diamondsuit \varphi \wedge \diamondsuit \psi \Rightarrow \diamondsuit ( \diamondsuit \varphi \wedge \psi) \vee \diamondsuit (\varphi \wedge \psi) \vee \diamondsuit (\varphi \wedge \diamondsuit \psi)$$

or $\boxplus ( \Box \varphi \Rightarrow \psi) \vee \boxplus ( \Box \psi \Rightarrow \varphi)$

or $\boxplus \varphi \Rightarrow \boxminus \boxplus \varphi$ together with Cp and Cf

The correspondence with the first given formula should be clear. Take a future connected frame. If $\varphi$ and $\psi$ are both true in the future, say at points $v$ and $w$, respectively, then if $w < v$ we must have by definition of $\diamondsuit$ that $( \diamondsuit \varphi \wedge \psi)$ holds at $w$ and hence $\diamondsuit ( \diamondsuit \varphi \wedge \psi)$ holds at $u$. The other two cases are similar. They are the only possibilities. The argument for the other direction is as straightforward but omitted for sake of space! Proofs of the other two formulae are also left as exercises.

We refer to temporal formulae corresponding to weak future (resp. past) connectedness, as WFC (and WPC). When both of these formulae are added as axioms to $K_T$, the resulting temporal logic is complete with respect to total orders.

**connectedness** This is a stronger property than WFC, or WPC, and captures that for every pair of time points, either one comes before the other, or vice-versa, or they are both the same point, i.e. $\forall t, u \in T \cdot (t < u \vee t = u \vee u < t)$*. Interestingly, there is no temporal formula that characterises the class of connected frames. The proof of this is by contradiction. Suppose $\varphi$ is such a characteristic formula, then by definition it is valid on the connected frames $F_1 = (T_1, <_1)$ and $F_2 = (T_2, <_2)$ whose sets of time points $T_1$ and $T_2$ are disjoint. By definition of frame validity, $\varphi$ is valid on the frame $F_3 = (T_1 \cup T_2, <_1 \cup <_2)$. Unfortunately, $F_3$ is not connected since $T_1$ and $T_2$ are disjoint sets. Thus $\varphi$ can not be a characteristic formula and there is no such characterisation.

**future seriality** This means that time is endless in the future direction, i.e. $\forall s \in T \cdot \exists t \in T \cdot s < t$. The temporal formulae

$$\boxplus \varphi \Rightarrow \diamondsuit \varphi$$

or even

$$\diamondsuit \mathbf{true}$$

characterise this frame property. Considering the first formula. Suppose there was an end to time, i.e. there is a point in time from which there are no other reachable points. Since the given formula is valid, it must be true at that end-point. Furthermore, at

* Some authors refer to this property as "comparability" since every pair of elements are comparable. We prefer the term "connectedness" because it fits better with the notion of fully connected networks of time points.

that end-point it must be the case that $\boxplus\varphi$ is also true (vacuously), since there are no other points. That therefore means that $\diamondplus\,\varphi$ must be true at the end-point. But this is contradictory, since there are no points which can be reached from the end-point. Hence the original assumption that the frame was bounded in the future is therefore false.

The alternative formulation, $\diamondplus\,\mathbf{true}$, is perhaps more straightforward.

**past seriality** Similar to above, the mirror image of a formula characterising future seriality will determine past serial frames, i.e. those with no beginning. Thus, $\boxminus\varphi \Rightarrow \diamondminus\,\varphi$.

**maximal and minimal points** The addition of an axiom for future (past) seriality clearly forces the frames for which the temporal logic is valid to be endless in the future (past). The absence of such axioms, however, does not imply boundedness. The constraint for maximal (minimal) points will help, namely:

$$\exists s \in T \cdot \neg\exists t \in T \cdot s < t$$

The temporal formula

$$\boxplus\mathbf{false} \vee \diamondplus\,\boxplus\mathbf{false}$$

captures this constraint. Informally, either there is no future (first disjunct) or we can move to a future point that is the end (second disjunct). The mirror captures minimal points.

**weakly dense** (not to be confused with *weekly dense*!) This property is such that if two points $s$ and $u$ are related, i.e. $s < u$, then one can always find a time point in between. The frame property is:

$$\forall s, u \in T \cdot s < u \Rightarrow \exists t \in T \cdot s < t \wedge t < u$$

In our temporal logic this is neatly characterised by:

$$\diamondplus\,\varphi \Rightarrow \diamondplus\,\diamondplus\,\varphi$$

To see this fact, consider a frame that has some discreteness embedded within it. In particular there will be two points $e$ and $b$ such that $e < b$ and for which there are no points $m$ in between $e$ and $b$. For the formula $\diamondplus\,\varphi \Rightarrow \diamondplus\,\diamondplus\,\varphi$ to be valid on this frame, the formula must be true for all models based upon that frame. Choose a valuation such that $\varphi$ evaluates to true only at point $b$ and hence the evaluation of $\diamondplus\,\varphi$ at point $e$ must be true. Thus $\diamondplus\,\diamondplus\,\varphi$ must be true also at $e$, and therefore there is a point beyond $e$ such that $\diamondplus\,\varphi$ is true. Take that point to be the nearest future time point to $e$, i.e. time point $b$. But $\diamondplus\,\varphi$ is false at $b$ since $b$ is the only point that makes $\varphi$ true. And hence the formula $\diamondplus\,\varphi \Rightarrow \diamondplus\,\diamondplus\,\varphi$ is not valid on this frame. Clearly the formula is valid on weakly dense frames.

**weakly dense with breaks** The frame used as a counterexample immediately above was indeed a weakly dense frame with one break. These frames are characterised by the following property.

$$\forall s, t, u \in T \cdot s < t < u \Rightarrow \exists v \in T \cdot s < v < t \vee t < v < u$$

Informally, pick any three ordered time points, there may be a discreteness on only one side of the middle point for clearly if there were no points in between either $s$ and $t$, or $t$ and $u$, the property would be false. Here is a temporal formula corresponding to this property.

$$\Diamond\!\!\!\!+ (\varphi \wedge \Diamond\!\!\!\!+ \psi) \Rightarrow \Diamond\!\!\!\!+ ( \Diamond\!\!\!\!+ \varphi \vee \Diamond\!\!\!\!+ \Diamond\!\!\!\!+ \psi)$$

We leave the proof of correspondence to the reader. Clearly this formula holds on a frame with single gaps between dense regions, so consider the validity on a frame with two consecutive gaps between otherwise dense regions.

**immediate successors** The constraint

$$\forall s \in T \cdot \exists u \in T \cdot s < u \wedge \neg \exists t \in T \cdot s < t \wedge t < u$$

defines frames whose points have immediate successors, i.e. a necessary condition (but not sufficient) for obtaining discrete frames. The temporal formula

$$\varphi \wedge \boxdot \varphi \Rightarrow \Diamond\!\!\!\!+ \boxdot \varphi$$

corresponds to this property. Suppose a frame $F$ at point $s$ does not have an immediate successor, i.e. $\forall u \in T \cdot s < u \Rightarrow \exists t \in T \cdot s < t < u$, the given formula is not valid on that frame. Consider a valuation that makes $\varphi$ true for time point $s$ and all its preceding points, and false elsewhere. Since the right neighbourhood of $s$ is dense, $\Diamond\!\!\!\!+ \boxdot \varphi$ will be false. Hence the formula can not be valid on such a frame. Clearly the formula is valid on frames with immediate successors.

The mirror image of this formula characterises immediate predecessors.

**irreflexivity** This was the first frame property presented in table 4.1 of the previous section. Unfortunately there is no axiom that corresponds to this particular property, however, when other constraints are placed, one can result in frames that are irreflexive, amongst other properties. For example, add Löb's axiom to transitive, weakly future connected frames. If it were the case that for some point $s \in T$, $s < s$, there would be a potentially infinite chain of stability - but this is contradicted by Löb's well-foundedness property implying that there is always some first point at which a property becomes true. Hence there can not be any such cycles in the relation, thus the frames must also now be irreflexive.

As one further example of correspondence, if we add to our minimal temporal logic $K_T$, axioms for weak future (past) connectedness, WFC (WPC), and a weakened version of immediate successors (to cater for possible boundedness), the resulting temporal logic is complete with respect to discrete total orders (see Figure 4.2).

| | | | |
|---|---|---|---|
| $Cf$ | $\varphi$ | $\Rightarrow$ | $\boxplus \Diamond \varphi$ |
| $Cp$ | $\varphi$ | $\Rightarrow$ | $\boxminus \Diamond \varphi$ |
| $4f$ | $\Diamond \Diamond \varphi$ | $\Rightarrow$ | $\Diamond \varphi$ |
| $4p$ | $\Diamond \Diamond \varphi$ | $\Rightarrow$ | $\Diamond \varphi$ |
| $Kf$ | $\boxplus(\varphi \Rightarrow \psi)$ | $\Rightarrow$ | $(\boxplus \varphi \Rightarrow \boxplus \psi)$ |
| $Kp$ | $\boxminus(\varphi \Rightarrow \psi)$ | $\Rightarrow$ | $(\boxminus \varphi \Rightarrow \boxminus \psi)$ |
| $WFC$ | $\Diamond \varphi \wedge \Diamond \psi$ | $\Rightarrow$ | $\Diamond(\Diamond \varphi \wedge \psi) \vee \Diamond(\varphi \wedge \psi) \vee \Diamond(\varphi \wedge \Diamond \psi)$ |
| $WPC$ | $\Diamond \varphi \wedge \Diamond \psi$ | $\Rightarrow$ | $\Diamond(\Diamond \varphi \wedge \psi) \vee \Diamond(\varphi \wedge \psi) \vee \Diamond(\varphi \wedge \Diamond \psi)$ |
| $IS$ | $(\varphi \wedge \boxminus \varphi)$ | $\Rightarrow$ | $(\boxplus \mathbf{false} \vee \Diamond \boxminus \varphi)$ |
| $IP$ | $(\varphi \wedge \boxplus \varphi)$ | $\Rightarrow$ | $(\boxminus \mathbf{false} \vee \Diamond \boxplus \varphi)$ |

Figure 4.2: Axioms for discrete temporal logic over total orders

### 4.3.2 A consistent but incomplete logic

We have been busy demonstrating correspondences between frame properties and temporal axioms. We now present an example (due to Thomason 1972) of a temporal logic which is consistent, i.e. it has models, but which is not determined by any class of frames. Consider the smallest temporal logic containing:

| | | | |
|---|---|---|---|
| Löb | $\Diamond \varphi$ | $\Rightarrow$ | $\Diamond(\varphi \wedge \neg \Diamond \varphi)$ |
| WFC | $\Diamond \varphi \wedge \Diamond \psi$ | $\Rightarrow$ | $\Diamond(\Diamond \varphi \wedge \psi) \vee \Diamond(\varphi \wedge \psi) \vee \Diamond(\varphi \wedge \Diamond \psi)$ |
| FS | $\Diamond \mathbf{true}$ | | |
| STAB | $\boxplus \Diamond \varphi$ | $\Rightarrow$ | $\Diamond \boxplus \varphi$ |

This is a consistent logic, but there are no frames for which it is valid.

To show that the above logic has a model, i.e. is consistent, consider $M = (\mathbb{N}, <, V)$ where $V(p) = \{\}$ for all $p$ in PROP. The frame $(\mathbb{N}, <)$ validates all axioms apart from STAB*. However, it can be shown, for any $\varphi$, that the set of points at which $\varphi$ is true in $M$ is either finite, or cofinite. Therefore, $\varphi$ either eventually stabilises as false (the finite case) or eventually stabilises as true. This corresponds to either $\boxplus \Diamond \varphi$ being false everywhere or $\Diamond \boxplus \varphi$ being true everywhere. Thus STAB is $M$-valid. We now establish that there is no frame which validates the above logic. Note first that Löb (i.e. Löb's axiom) determines transitive, wellfounded frames; WFC ensures that the frame is weakly future connected; and FS guarantees there are no future end-points. Suppose a frame validates the logic. The set of reachable points (i.e. via $<$), $P_s$, from $s$ is connected and forms a strict total ordering which by the future seriality, FS, has no final element. Take a subset $Q$ of $P_s$ such that neither $Q$ nor $P_s - Q$ has an end point. Make the valuation of $\varphi$ be that subset $Q$. Thus $\boxplus \Diamond \varphi$ is true at $s$ in $M$, but $\Diamond \boxplus \varphi$ is clearly false at $s$. But this contradicts the assumption that the frame validates the logic.

*In modal logic, this axiom is often referred to as McKinsey's axiom; see [Goldblatt, 1991] where the axiom is shown to be the smallest formula (not equivalent to one) that is not canonical - the McKinsey axiom is not valid in the canonical frame for the smallest normal modal logic containing it.

## 4.4 A Range of Linear Temporal Logics

The penultimate example of the previous section presented a temporal logic for discrete total orders, i.e. frames that are antisymmetric, transitive, (weakly) future and past connected, together with immediate successors and predecessors. If we remove the requirement for discreteness the resulting system is the smallest temporal logic closest to what we can call a linear temporal logic (often referred to as tense logic). Strictly speaking, it is the smallest temporal logic for total orders but it does not determine a linear order. A frame validating the axioms of this particular logic for total orders may be the union of a set of disjoint frames, each being a total order. The problem is essentially that which was raised when attempting to characterise connectedness. Indeed our logics can not distinguish the individual total orders in the set. Since we can't tell the difference between such parallel flows and a single linearly-ordered set of time points, we will treat the logic over such frames as a linear temporal logic.

In this section we will first define a few examples of linear temporal logics using the $\mathcal{L}_{(\boxplus, \boxminus)}$ language, then begin to explore the expressiveness of the temporal modalities. We will conclude the section with the introduction of a temporal logic based on fixed point operators - the temporal $\mu$-calculus.

In the presentations of temporal logics that follow, we will focus attention on the temporal axioms additional to the minimal temporal logic. But recall that an axiomatisation of the logic is built from a base of propositional tautologies together with inference rules for Modus Ponens and temporal necessitation (backwards and forwards), as in Figure 4.3.

**Axioms:**

| | | |
|---|---|---|
| $Taut$ | $\varphi$ | Propositional tautologies |
| $Cf$ | $\varphi \Rightarrow \boxplus \Diamond \varphi$ | |
| $Cp$ | $\varphi \Rightarrow \boxminus \Diamond \varphi$ | |
| $4f$ | $\Diamond \Diamond \varphi \Rightarrow \Diamond \varphi$ | |
| $4p$ | $\Diamond \Diamond \varphi \Rightarrow \Diamond \varphi$ | |
| $Kf$ | $\boxplus(\varphi \Rightarrow \psi) \Rightarrow (\boxplus\varphi \Rightarrow \boxplus\psi)$ | |
| $Kp$ | $\boxminus(\varphi \Rightarrow \psi) \Rightarrow (\boxminus\varphi \Rightarrow \boxminus\psi)$ | |
| $WFC$ | $\Diamond \varphi \wedge \Diamond \psi \Rightarrow$ $\Diamond(\Diamond \varphi \wedge \psi) \vee \Diamond(\varphi \wedge \psi) \vee \Diamond(\varphi \wedge \Diamond \psi)$ | |
| $WPC$ | $\Diamond \varphi \wedge \Diamond \psi \Rightarrow$ $\Diamond(\Diamond \varphi \wedge \psi) \vee \Diamond(\varphi \wedge \psi) \vee \Diamond(\varphi \wedge \Diamond \psi)$ | |

**Inference Rules:**

$$MP \quad \frac{\vdash \varphi \qquad \vdash \varphi \Rightarrow \psi}{\vdash \psi}$$

$$\boxplus - Gen \quad \frac{\vdash \varphi}{\vdash \boxplus\varphi}$$

$$\boxminus - Gen \quad \frac{\vdash \varphi}{\vdash \boxminus\varphi}$$

Figure 4.3: The Smallest Linear Temporal Logic

### 4.4.1 Linear temporal logics

We take as the starting point for linear systems the minimal system $K_T$ with just weak connectedness (future and past). This logic is the smallest *linear* temporal logic.

Now by adding further axioms, each being a temporal formula corresponding to particular constraints on temporal frames, one can define linear temporal logics over discrete structures, dense structures, finite but unbounded structures, infinite structures, etc. For example, by adding to the above logic formulae corresponding to past and future seriality as axioms, namely

PS $\Diamond_P \mathbf{true}$ FS $\Diamond_F \mathbf{true}$

we obtain a linear temporal logic that is infinite in the past and infinite in the future. But note this is only in the sense that one can make infinitely many moves via the temporal relation into the past, respectively future, from any given point in time. Within the class of frames determined by this logic is the frame that has the set of points in the real open interval $(0, 1)$ together with the usual $<$ ordering on points as well as the frame that has the natural numbers with the usual ordering. To constrain the logic to, say, weakly dense models, the following formulae should be added as axioms.

WDF $\Diamond_F \varphi \Rightarrow \Diamond_F \Diamond_F \varphi$ WDP $\Diamond_P \varphi \Rightarrow \Diamond_P \Diamond_P \varphi$

On the other hand, to move towards natural number time, we need to add constraints for discreteness, i.e. immediate predecessors and successors.

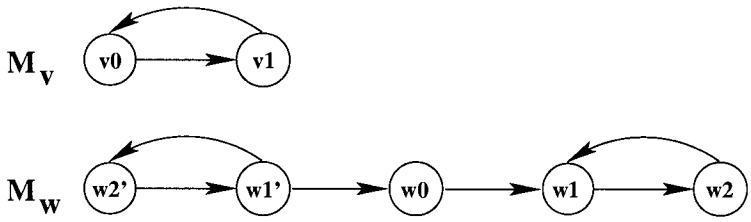
IP $(\varphi \wedge \Diamond_P \varphi) \Rightarrow \Diamond_F \boxminus \varphi$ IS $(\varphi \wedge \Diamond_F \varphi) \Rightarrow \Diamond_P \boxplus \varphi$

Note that if the temporal flow is not constrained to be serial in the past and/or future the above axioms would require weakening to allow for the beginning, respectively end, of time, i.e.

WIP $(\varphi \wedge \Diamond_P \varphi \wedge \Diamond_F \mathbf{true}) \Rightarrow \Diamond_F \boxminus \varphi$
WIS $(\varphi \wedge \Diamond_F \varphi \wedge \Diamond_P \mathbf{true}) \Rightarrow \Diamond_P \boxplus \varphi$

#### Modalities for Next and Previous

Even though the logic above determines discrete frames, this does not mean that the logic is expressive enough to be able to define a general next-time (respectively, previous-time) temporal modality, one that would move forwards (backwards) one step in time to the next (previous) moment in time. The proof that such a temporal modality is not expressible, i.e. definable, in $\mathcal{L}_{(\boxdot, \boxminus)}$, proceeds as follows. We pose two models, which clearly could be distinguished by a logic with a next time connective. By distinguished, we mean that a formula can be given which has a different truth value at related time points of the different models. We then establish that the models can not be distinguished by the logic without next, i.e. that the models are zig-zag equivalent. If that is the case, then clearly "next" can not be definable in the logic without next, otherwise the models would still be distinguishable. So consider the two models $M_v$ and $M_w$




where $V(p) = \{\text{v1}\}$, $W(p) = \{\text{w2'}, \text{w2}\}$ and $V(x) = W(x) = \{\}$ for all $x \neq p$. We claim that for any formula $\varphi$, $M_V, \text{v0} \models \varphi$ iff $M_W, \text{w0} \models \varphi$. To prove this, however, we need to establish a stronger result, namely

i. v1 $\models \varphi$ **iff** w2' $\models \varphi$
ii. v1 $\models \varphi$ **iff** w2 $\models \varphi$
iii. v0 $\models \varphi$ **iff** w1' $\models \varphi$
iv. v0 $\models \varphi$ **iff** w0 $\models \varphi$
v. v0 $\models \varphi$ **iff** w1 $\models \varphi$

which effectively shows that the point v0 is equivalent to w1' , w0 and to w1 , and that point v1 is equivalent to w2' and to w2 .

Without loss of generality, assume the temporal language has only one atom, namely $p$. We will establish the results by inducting over the depth of temporal connectives. (Note, of course, that the depth of $\boxplus\varphi$ is one plus the depth of $\varphi$ and the depth of a purely propositional formula is zero.)

**Basis:** The valuation of any purely propositional formula $\varphi$ at some point t is only dependent on the valuations of propositions at t. Thus:

i. v1 $\models \varphi$ **iff** w2' $\models \varphi$
ii. v1 $\models \varphi$ **iff** w2 $\models \varphi$
iii. v0 $\models \varphi$ **iff** w1' $\models \varphi$
iv. v0 $\models \varphi$ **iff** w0 $\models \varphi$
v. v0 $\models \varphi$ **iff** w1 $\models \varphi$

since v0 has the same valuation for $p$ as w1' , w0 and w1 , and v1 has the same valuation for $p$ as w2' and w2 .

**Inductive step:** Assume the result holds for all formulae $\varphi$ with temporal connective depth less than or equal to $k$. We now show that the result holds for all formulae with depth $k+1$.

Wlog, consider formulae only of shape $\boxplus\varphi$ where $\varphi$ has maximum depth $k$. The argument for other temporal connectives will be similar. Then the other cases to be considered can be handled using those forms.

By definition v0 $\models \boxplus\varphi$ implies both v0 $\models \varphi$ and v1 $\models \varphi$. Therefore by the inductive assumption, it implies w2' $\models \varphi$, w1' $\models \varphi$, w0 $\models \varphi$, w1 $\models \varphi$ and w2 $\models \varphi$. But from these we can obtain, w1' $\models \boxplus\varphi$, w0 $\models \boxplus\varphi$ and w1 $\models \boxplus\varphi$, which is as required.

Similarly v1 $\models \boxplus\varphi$ implies both w2' $\models \boxplus\varphi$ and w2 $\models \boxplus\varphi$.

The argument for the converses proceeds along similar lines.

Now we need to introduce next-time and previous-time modalities into our logic and show that the models $M_V$ and $M_W$ can be distinguished. For convenience, let us first introduce a relation $\mathcal{N}$ from $<$ of the temporal frame.

$$m\mathcal{N}n \text{ iff } m < n \wedge \neg\exists t \cdot m < t < n$$

Thus, for discrete frames $\mathcal{N}$ relates adjacent time points, but for dense frames (with no gaps) this relation is empty. It is, in effect, a one-step relation*. A temporal modality for next-time, $\bigcirc$, is thus defined as:

$$F, s \models \bigcirc\varphi \text{ iff for all } t \cdot s\mathcal{N}t \text{ implies } F, t \models \varphi$$

We read this temporal modality as "in the next moment", or "tomorrow", etc. Similarly, we define a temporal modality for taking a step backwards in time, $\bullet$, which we read as "in the previous moment", or "yesterday", and so forth.

$$F, s \models \bullet\varphi \text{ iff for all } t \cdot t\mathcal{N}s \text{ implies } F, t \models \varphi$$

Let the language $\mathcal{L}_{(\bigcirc, \bullet)}$ be $\mathcal{L}_{(\boxdot, \boxminus)}$ extended in the obvious way with the next and previous time modalities. Are the two models posed above distinguishable with this logic? For $\mathcal{L}_{(\boxdot, \boxminus)}$ we had shown that the point v0 was indistinguishable from the points w1' , w0 and w1 . But for the formula $\bigcirc p \wedge \bullet p$, for example, clearly distinguishes them. It is true at point v0 , but is clearly false in w1' , w0 and w1 . We have thus established that $\mathcal{L}_{(\bigcirc, \bullet)}$ is more expressive than $\mathcal{L}_{(\boxdot, \boxminus)}$.

**Example 4.4.1 (Next time relationships).** *Consider a temporal frame $F = (\mathbf{Z}, <)$. The following formulae (and the corresponding mirror formulae)*†

| | | | |
|---|---|---|---|
| *i.* | $\neg\bigcirc\varphi$ | $\Leftrightarrow$ | $\bigcirc\neg\varphi$ |
| *ii.* | $\bigcirc(\varphi \Rightarrow \psi)$ | $\Rightarrow$ | $(\bigcirc\varphi \Rightarrow \bigcirc\psi)$ |
| *iii.* | $\Box\varphi$ | $\Leftrightarrow$ | $\varphi \wedge \bigcirc\Box\varphi$ |
| *iv.* | $\Diamond\varphi$ | $\Leftrightarrow$ | $\varphi \vee \bigcirc\Diamond\varphi$ |
| *v.* | $\bigcirc\bullet\varphi$ | $\Leftrightarrow$ | $\bullet\bigcirc\varphi$ |

*are all* valid *on $F$. We sketch the proof of (i) and (iii), respectively, and leave the others as an exercise. Considering the $\Rightarrow$ direction of (i), for any point $s$, $s \models \neg\bigcirc\varphi$ implies that it is not the case that $s \models \bigcirc\varphi$. By definition of $\bigcirc$, this implies that it is also not the case that $s+1 \models \varphi$. In other words, it is the case that $s+1 \models \neg\varphi$. Therefore, by definition of $\bigcirc$, $s \models \bigcirc\neg\varphi$. The $\Leftarrow$ direction of (i) is as straightforward. For the $\Rightarrow$ direction of (iii), $s \models \Box\varphi$ means that $t \models \varphi$, for every $t$ such that $s \leq t$. Thus $s \models \varphi$ and for every $t$ such that $s+1 \leq t$ $t \models \varphi$. Hence $s \models \bigcirc\Box\varphi$. Similarly, for the $\Leftarrow$ direction of (iii). If $s \models \varphi$ and $s+1 \models \Box\varphi$ then clearly for every $t$ such that $s \leq t$, $t \models \varphi$. Hence the result.*

The above equivalences are of particular interest for they indicate that one might be able to define natural number based temporal logic in terms of $\bigcirc$, $\Box$, instead of using $\boxplus$. Indeed, the formula $\neg\bigcirc\varphi \Rightarrow \bigcirc\neg\varphi$ characterises, in a certain sense, linearity‡. The other direction characterises seriality and discreteness.

---

*Some presentations of linear discrete temporal logics actually start with such a next time relation as the frame relation, then define the usual frame relation (used in $\boxplus$ definitions, etc) as the transitive closure of the step relation.

†Recall that $\Box\varphi \stackrel{def}{=} \varphi \wedge \boxplus$ and that $\Diamond\varphi \stackrel{def}{=} \varphi \vee \dot{\Diamond}\varphi$, i.e. they are the reflexive versions.

‡Actually, branching models would be acceptable, however, the logic would not be able to distinguish the different paths.

**For a while**

We will now introduce another temporal modality that is easily definable over our model structures; the future (past) version captures the property that a formula holds true within the future (past) vicinity of the current point. Such a modality would be useful in natural language representation, for example, for expressing temporal adverbials such as "for a while". Formally, we define

$$M, s \models \widetilde{\boxplus}\varphi \textbf{ iff } \exists t \cdot s < t \wedge \forall u \cdot s < u \wedge u < t \Rightarrow M, u \models \varphi$$

and then read $\widetilde{\boxplus}\varphi$ as "$\varphi$ holds uninterruptably for a while immediately in the future". The past time mirror of this connective has obvious definition. An interesting question is whether this temporal modality can be defined in either $\mathcal{L}_{(\boxdot, \boxminus)}$ or $\mathcal{L}_{(\bigcirc, \bullet)}$? If it can't we have shown yet another weakness in the expressibility of this particular temporal modal logic. The answer, not unsurprisingly, is that such modalities are not definable in $\mathcal{L}_{(\boxdot, \boxminus)}$. We will proceed here to sketch the proof for the first question, is $\widetilde{\boxplus}$ expressible in $\mathcal{L}_{(\boxdot, \boxminus)}$, leaving the second as an exercise for the interested reader.

We follow a similar approach to that above in showing that $\bigcirc$ was not expressible in $\mathcal{L}_{(\boxdot, \boxminus)}$. So we need to find a pair of models that can not be distinguished by any formula in $\mathcal{L}_{(\boxdot, \boxminus)}$, then show that a formula of $\mathcal{L}_{(\boxdot, \boxminus, \widetilde{\boxplus}, \widetilde{\boxminus})}$ is able to distinguish them. We pose two models $M_V$ and $M_W$ based on the temporal frame $F = (\mathbf{R}^+, <)$ with the valuations $V$ and $W$, respectively, for proposition letter $p$ (again, without loss of generality, assume only one proposition letter).

$$V(p) = \{0, 1, 2, \ldots\} \qquad W(p) = \{\tfrac{1}{2}, \tfrac{1}{4}, \tfrac{1}{8}, \ldots\} \cup V(p)$$

An inductive proof over the structure of formulae of $\mathcal{L}_{(\boxdot, \boxminus)}$will easily establish that for any $\mathcal{L}_{(\boxdot, \boxminus)}$ formula $\varphi$

$$M_V, 0 \models \varphi \textbf{ iff } M_W, 0 \models \varphi$$

However, it is fairly easy to see that the formula $\widetilde{\boxplus}\neg p$ does indeed distinguish these two models at time point 0, i.e.

i. $M_V, 0 \models \widetilde{\boxplus}\neg p$ ii. $M_W, 0 \not\models \widetilde{\boxplus}\neg p$

The formula $\widetilde{\boxplus}\neg p$ holds at 0 in $M_V$ since $p$ is false in the open interval $(0, 1)$. However, in $M_W$ the same is not the case. For any point $t$ within the open interval $(0, 1)$, we have infinitely many points $s$, $0 < s < t$, such that $M_W, s \models p$, i.e. there is no point $s$, $0 < s < 1$ such that $p$ is stable over $(0, s]$.

**The tense modalities: Until and Since**

As one further extension, we define the until and since temporal connectives. These are generally referred to as tense logic modalities because their introduction came principally from a logical formalisation of tense in natural language.

$M, s \models \varphi \mathcal{U}^+ \psi$ iff there is $u \cdot s < u$ and $M, u \models \psi$ and for all $t \cdot s < t < u$ implies $M, t \models \varphi$

$M, s \models \varphi \textbf{ since}^- \psi$ iff there is $u \cdot u < s$ and $M, u \models \psi$ and for all $t \cdot u < t < s$ implies $M, t \models \varphi$

The formula $\varphi \mathcal{U}^+ \psi$ is read as $\varphi$ will hold until $\psi$ holds (similarly for since). As we will indicate, they have formally been shown to be very expressive. Indeed first of all notice that a language based on until and since, i.e. $\mathcal{L}_{(\mathcal{U}^+, \mathcal{S}^-)}$ contains both $\mathcal{L}_{(\boxdot, \boxdot)}$ and $\mathcal{L}_{(\boxdot, \boxdot, \tilde{\boxplus}, \tilde{\boxminus})}$. The following equivalences are straightforward to establish:

$$\begin{array}{lcllcl} \diamondsuit \varphi & \Leftrightarrow & \mathbf{true}\, \mathcal{U}^+ \varphi & \diamondsuit \varphi & \Leftrightarrow & \mathbf{true}\ \mathbf{since}^-\ \varphi \\ \tilde{\boxplus} \varphi & \Leftrightarrow & \varphi \mathcal{U}^+ \mathbf{true} & \tilde{\boxminus} \varphi & \Leftrightarrow & \varphi\ \mathbf{since}^-\ \mathbf{true} \end{array}$$

However, we really need to establish that $\mathcal{L}_{(\mathcal{U}^+, \mathcal{S}^-)}$ is strictly more expressive than the previous language $\mathcal{L}_{(\boxdot, \boxdot, \tilde{\boxplus}, \tilde{\boxminus})}$. The approach is as before. We must provide two models that can be distinguished by formulas of $\mathcal{L}_{(\mathcal{U}^+, \mathcal{S}^-)}$ but not by formulas of $\mathcal{L}_{(\boxdot, \boxdot, \tilde{\boxplus}, \tilde{\boxminus})}$. Consider models $M_V$ and $M_W$ formed from the frame $(\mathbf{R}, <)$ with valuations $V$ and $W$ for two propositions, $p$ and $q$, as:

$$\begin{array}{lcl} V(p) & = & \{\pm 1, \pm 2, \pm 3, \ldots\} \\ W(p) & = & \{\pm 2, \pm 3, \ldots\} \\ V(q) & = & W(q) = \{\ldots (-5,-4), (-3,-2), (-1,+1), (+2,+3), (+4,+5) \ldots\} \\ & & \text{i.e. the union of open intervals} \end{array}$$

We can show, via an inductive argument over the structure of formulae, that for any $\varphi$ constructed from just $\boxplus$, $\boxminus$, $\tilde{\boxplus}$, $\tilde{\boxminus}$ temporal connectives,

$$M_V, 0 \models \varphi \text{ iff } M_W, 0 \models \varphi$$

But it should be clear that

$$M_V, 0 \models q \mathcal{U}^+ p \text{ and } M_W, 0 \not\models q \mathcal{U}^+ p$$

because in model $M_V$ $p$ is true at time point 1 and $q$ holds over the interval $(0, 1)$. However, in model $M_W$ the first point in the future of 0 at which $p$ holds is time point 2 and $q$ does not hold over the open interval $(0, 2)$ for it is false over the closed interval $[1, 2]$.

### Since and Until in Linear Discrete Frames

If we restrict attention to linear discrete frames, a collection of connectives, which have been shown most useful for describing properties of computational systems, can be defined as below.

$$\begin{array}{rcl|rcl} \odot \varphi & \stackrel{def}{=} & \mathbf{false}\, \mathcal{U}^+ \varphi & \bullet \varphi & \stackrel{def}{=} & \mathbf{false}\ \mathbf{since}^-\ \varphi \\ \bigcirc \varphi & \stackrel{def}{=} & \neg \odot \neg \varphi & \bullet \varphi & \stackrel{def}{=} & \neg \bullet \neg \varphi \\ \diamondsuit \varphi & \stackrel{def}{=} & \varphi \vee \mathbf{true}\, \mathcal{U}^+ \varphi & \blacklozenge \varphi & \stackrel{def}{=} & \varphi \vee \mathbf{true}\ \mathbf{since}^-\ \varphi \\ \Box \varphi & \stackrel{def}{=} & \neg \diamondsuit \neg \varphi & \blacksquare \varphi & \stackrel{def}{=} & \neg \blacklozenge \neg \varphi \\ \varphi\ \mathbf{until}\ \psi & \stackrel{def}{=} & \psi \vee \varphi \wedge (\bigcirc \psi \vee \varphi \mathcal{U}^+ \psi) & \varphi\ \mathbf{since}\ \psi & \stackrel{def}{=} & \psi \vee \varphi \wedge (\bullet \psi \vee \varphi\ \mathbf{since}^-\ \psi) \\ \varphi \mathcal{W} \psi & \stackrel{def}{=} & \varphi\ \mathbf{until}\ \psi \vee \Box \varphi & \varphi \mathcal{Z} \psi & \stackrel{def}{=} & \varphi\ \mathbf{since}\ \psi \vee \blacksquare \varphi \end{array}$$

The above definitions are fairly self-explanatory, but let us dwell on a few of them. First of all we have defined a "strong" version of the next-time modality, $\odot$. We refer to this as a strong version of next because $\odot \varphi$ is existential in nature, i.e. if it holds, then there is a next moment of time and $\varphi$ holds there. It is defined in terms of $\mathcal{U}^+$ by noting that $\varphi$ holds eventually in the strict future, i.e. beyond now, but that **false** holds strictly between now and

when $\varphi$ holds. Because **false** never holds, there can't be any points in between now and when $\varphi$ holds, so it must be a next moment in time. The universal version, or weak version, of next, e.g. as in $\bigcirc\varphi$, may be vacuously true - in the situation that there is no next moment - or true if $\varphi$ holds in all next moments (the model may be branching in the general case). The universal version of next is obtained from the existential one in the obvious way. Thus when interpreted in linear discrete frames, the formula $\odot\mathbf{true}$ characterises that there is a next moment, whereas $\bigcirc\mathbf{false}$ can only be true at the end of time. The past mirrors of strong and weak next, i.e. strong and weak previous, temporal modalities are defined in a similar manner.

A non-strict modality for eventually in the future, i.e. allowing $\Diamond\varphi$ to be satisfied by $\varphi$ holding now, is defined by noting that either $\varphi$ holds now or, via the strict until, $\varphi$ holds in the future (with **true** holding in between). The non-strict always in the future is simply the dual of the non-strict eventually in the future.

A non-strict version of until, $\varphi$ **until** $\psi$, can be satisfied by $\psi$ holding now or $\varphi$ holding now together with either $\psi$ holding at the next moment or $\varphi$ will hold strictly until $\psi$. We can note that from this definition the following equivalence holds

$$\varphi \,\mathbf{until}\, \psi \Leftrightarrow \psi \vee \varphi \wedge \bigcirc(\varphi \,\mathbf{until}\, \psi)$$

In computational and specification contexts it has also been found useful to define a weak version of the until connective, $\mathcal{W}$ - read as unless, which is universal in nature and doesn't force $\psi$ to be true, but in the situation that $\psi$ is never true, $\varphi$ must be true for ever.

### Future-time Linear Discrete Temporal Logic

Restricting attention to a future fragment of the since-until temporal language over linear discrete frames, e.g. $(\mathbb{N}, <)$, we obtain the logic that was used in the early work of Manna and Pnueli for the global description of program properties, see for example [Pnueli, 1977; Manna and Pnueli, 1992; Manna and Pnueli, 1995]. The following is an axiomatisation for the logic.

The given axiomatisation follows, effectively, the approach that we've taken before - namely start with a minimal system and add constraints to restrict to the frame(s) of interest. Thus we take all tautologies as axioms. The first axiom about next is essentially the K axiom of modal logic. The second axiom about next provides commutativity of next and negation; it also determines future seriality and discreteness (in one direction) and future linearity (for the other), as explained below. The essence of the final axiom for unless is that it determines the formula $\varphi\,\mathcal{W}\,\psi$ as a solution to the implication $\xi \Rightarrow \psi \vee \varphi \wedge \bigcirc\xi$. Correspondingly, the inference rule of interest is $\mathcal{W}$-introduction. This captures the fact that $\varphi\,\mathcal{W}\,\psi$ is a maximal solution to the above-mentioned implication*. We will give discussion on that matter in Section 4.4.5 where we consider the more general fixed point temporal logic. The other future-time modalities that we introduced earlier can be defined as below.

$$\begin{array}{llll}
\Box\varphi & \stackrel{def}{=} \varphi\,\mathcal{W}\,\mathbf{false} & \boxplus\varphi & \stackrel{def}{=} \bigcirc\Box\varphi \\
\Diamond\varphi & \stackrel{def}{=} \neg\Box\neg\varphi & \oplus\varphi & \stackrel{def}{=} \bigcirc\Diamond\varphi \\
\varphi \,\mathbf{until}\, \psi & \stackrel{def}{=} \neg(\neg\psi\,\mathcal{W}\,(\neg\psi \wedge \wedge\varphi)) & \varphi\,\mathcal{U}^{+}\,\psi & \stackrel{def}{=} \bigcirc(\varphi \,\mathbf{until}\, \psi)
\end{array}$$

*Solutions are ordered by implication; thus **false** is the minimum element of the ordering and the maximum element is **true**.

**Axioms**

$\vdash w$ tautology
$\vdash \bigcirc(\varphi \Rightarrow \psi) \Rightarrow (\bigcirc\varphi \Rightarrow \bigcirc\psi)$
$\vdash \bigcirc\neg\varphi \Leftrightarrow \neg\bigcirc\varphi$
$\vdash \varphi \mathcal{W} \psi \Rightarrow \psi \vee \varphi \wedge \bigcirc(\varphi \mathcal{W} \psi)$

**Inference Rules**

Modus Ponens $\dfrac{\vdash \varphi \qquad \vdash \varphi \Rightarrow \psi}{\vdash \psi}$

$\bigcirc$ – Gen $\dfrac{\vdash \varphi}{\vdash \bigcirc\varphi}$

$\mathcal{W}$ – Intro $\dfrac{\vdash \xi \Rightarrow \psi \vee \varphi \wedge \bigcirc\xi}{\vdash \xi \Rightarrow \varphi \mathcal{W} \psi}$

Note:

$i \models \varphi \mathcal{W} \psi$ iff there is $k \cdot i \leq k$ and $k \models \psi$ and
for all $j \cdot i \leq j < k$ implies $j \models \varphi$
or for all $k \cdot i \leq k$ implies $k \models \varphi$

Figure 4.4: Axiomatisation for Future-time Linear Discrete Temporal Logic

The definition of **until** may look slightly odd at first, however, it has in fact been defined as the dual of $\mathcal{W}$, just as $\Diamond$ is defined as the dual of $\Box$. Some further equivalences and explanation are given below in the paragraph "Until-Unless duality" .

**Theorem 4.4.1.** *The logic $\mathcal{L}_{(\mathcal{W},\bigcirc)}$ is sound with respect to temporal frames $(\mathbb{N}, <)$, i.e.*

*if $F \vdash \varphi$ then $F \models \varphi$*

We will not work through the soundness proof leaving that as an exercise. However, let us just note the interesting properties of the axiom $\neg\bigcirc\varphi \Leftrightarrow \bigcirc\neg\varphi$.

Consider $\neg\bigcirc\varphi \Rightarrow \bigcirc\neg\varphi$ as an axiom on the class of discrete frames. We show that the frames must be future linear. First remember that $\bigcirc$ has a universal interpretation, namely: $s \models \bigcirc\varphi$ if and only if for all $t$, $s\mathcal{N}t$ implies $t \models \varphi$. Thus, by definition, $s \models \neg\bigcirc\varphi$ implies that it is not the case that $\varphi$ holds for all successors of $s$, i.e. there is at least one successor $t$ where $\varphi$ is false. But $s \models \bigcirc\neg\varphi$ implies, by definition, that $\varphi$ is false in all successors of $s$. Therefore, as the given implication is axiomatic, our language can not distinguish between the successors: the branching model is thus zig-zag equivalent to a linear model. Hence the axiom determines, in essence, future linearity.

Consider next $\bigcirc\neg\varphi \Rightarrow \neg\bigcirc\varphi$ as an axiom on the class of discrete frames. We show that the frames validating the formula must be future serial. Suppose $s$ is an endpoint. For any valuation of $\varphi$, $s \models \bigcirc\neg\varphi$ is true: there is no successor to the point $s$ and thus $\bigcirc\varphi$ is vacuously true. But by the axiom, $\neg\bigcirc\varphi$, must also hold at $s$. But this formula is false at $s$ as $\bigcirc\varphi$ must be true at an endpoint, which therefore contradicts the assumption that $s$ is an endpoint. The class of frames is thus future serial.

Propositional logic has the following deduction theorem

$$\frac{\varphi_1, \ldots, \varphi_n \vdash \psi}{\varphi_1, \ldots, \varphi_{n-1} \vdash \varphi_n \Rightarrow \psi}$$

where $\varphi_1, \ldots, \varphi_n \vdash \psi$ means that $\psi$ is a theorem under the assumptions that $\varphi_1, \ldots, \varphi_n$ are also theorems. Suppose this were to hold in modal systems.

Assume $\vdash \varphi$. By the necessitation rule, $\vdash \Box\varphi$ is also a theorem, thus $\varphi \vdash \Box\varphi$. Therefore by the (proposed) deduction theorem, $\vdash \varphi \Rightarrow \Box\varphi$. But this is not valid — consider the model $M = (\mathbb{N}, <, V)$ such that $V(p) = \{0\}$. Clearly this invalidates $p \Rightarrow \Box p$.

The problem is that theoremhood in modal systems corresponds to truth in all worlds of all models of a class of frames. The movement of a premise assumption across the turnstile effectively weakens the assumption to it being true at a world, rather than at all worlds of a model. Thus modal deduction must ensure that the box is introduced. Hence

$$\frac{\varphi_1, \ldots, \varphi_n \vdash \psi}{\varphi_1, \ldots, \varphi_{n-1} \vdash \Box\varphi_n \Rightarrow \psi}$$

The given counterexample then deduces that $\vdash \Box p \Rightarrow \Box p$

**Theorem 4.4.2.** *The logic $\mathcal{L}_{(\mathcal{W},\bigcirc)}$ is complete with respect to temporal frames $(\mathbb{N}, <)$, i.e.*

*if $F \models \varphi$ then $F \vdash \varphi$*

Completeness proofs are generally notoriously much harder to establish than their soundness counterparts. In modal logic there are several standard techniques for establishing completeness. We don't have the space to investigate such methods. However, one approach that has been used for this particular temporal logic over the natural numbers builds on the fact that the logic is decidable. By that we mean an algorithm can be given which will give a yes/no answer to the problem "Is $\varphi$ (in $\mathcal{L}_{(\mathcal{W},\bigcirc)}$) valid for the frame $(\mathbb{N}, <)$?" Again, we don't have space to give detailed proof of the decision procedure, however, we will outline the mechanism later. Completeness can be shown by establishing that the steps taken by the semantic procedure can be encoded as a proof using the given axiomatisation.

**Until-Unless duality** The definition of until as the dual of the unless gives the following immediate equivalences.

$$\begin{array}{lcl} \neg(\varphi \textbf{ until } \psi) & \Leftrightarrow & (\neg\psi)\,\mathcal{W}\,(\neg\psi \wedge \neg\varphi) \\ \neg(\varphi\,\mathcal{W}\,\psi) & \Leftrightarrow & (\neg\psi) \textbf{ until } (\neg\psi \wedge \neg\varphi) \end{array}$$

The first equivalence above is by definition; the second one is derived from the first via renaming of propositions. An inference rule for the introduction of **until** can be obtained from the $\mathcal{W}$-intro inference rule and defines $\varphi$ **until** $\psi$ as the minimal solution to the implication $\psi \vee (\varphi \wedge \bigcirc X) \Rightarrow X$.

| | |
|---|---|
| $\psi \vee (\varphi \wedge \bigcirc X) \Rightarrow X$ | Assumption |
| $\neg X \Rightarrow \neg(\psi \vee (\varphi \wedge \bigcirc X))$ | PR |
| $\neg X \Rightarrow (\neg\psi \wedge \neg\varphi) \vee (\neg\psi \wedge \bigcirc\neg X)$ | PR |
| $\neg X \Rightarrow (\neg\psi)\,\mathcal{W}\,(\neg\psi \wedge \varphi)$ | $\mathcal{W}$-Intro |
| $\neg((\neg\psi)\,\mathcal{W}\,(\neg\psi \wedge \varphi)) \Rightarrow X$ | PR |
| $\varphi$ **until** $\psi \Rightarrow X$ | by **until** definition to yield conclusion |

thus establishing the rule

$$\textbf{until}\text{ -Intro} \quad \frac{\vdash \psi \vee \varphi \wedge \bigcirc \xi \Rightarrow \xi}{\vdash \varphi \textbf{ until } \psi \Rightarrow \xi}$$

Furthermore, it is relatively straightforward to establish that $\varphi$ **until** $\psi \Leftrightarrow \varphi \mathcal{W} \psi \wedge \Diamond \psi$, which could have been taken as an alternative definition. The $\Rightarrow$ direction of the equivalence is trivially proved by showing that $\varphi \mathcal{W} \psi \wedge \Diamond \psi$ is a solution to the implication $\psi \vee (\varphi \wedge \bigcirc X) \Rightarrow X$. In a similar way, the $\Leftarrow$ direction can be proved by showing that $\neg(\varphi$ **until** $\psi) \wedge \varphi \mathcal{W} \psi$ is a solution to the implication $X \Rightarrow \neg\psi \wedge \bigcirc X$, thus proving that $\neg(\varphi$ **until** $\psi) \wedge \varphi \mathcal{W} \psi \Rightarrow \Box \neg \psi$ and hence that $\varphi \mathcal{W} \psi \wedge \Diamond \psi \Rightarrow \varphi$ **until** $\psi$.

**Incorporating the Past** It is relatively straightforward to adapt the axiomatisation of $\mathcal{L}_{(\mathcal{W},\bigcirc)}$ to an axiomatisation for $\mathcal{L}_{(\mathcal{W},\mathcal{Z},\bigcirc,\bullet)}$ which is sound and complete with respect to the frame $(\mathbf{Z}, <)$. The temporal modality $\mathcal{Z}$ is the past time mirror of the unless modality $\mathcal{W}$ and is pronounced "zince" (the existential, or strong, version being the since modality **since**). First of all include the past time mirrors of the $\mathcal{L}_{(\mathcal{W},\bigcirc)}$ axioms and rules, then introduce the "cancellation" axioms for next and last. Note that the cancellation axioms simply capture the fact that now is in the past, in fact yesterday, of tomorrow, and vice-versa. The result is listed in Figure 4.5.

And now what must be done in order to obtain an axiomatisation of the logic $\mathcal{L}_{(\mathcal{W},\mathcal{Z},\bigcirc,\bullet)}$ that is complete with respect to the frame $(\mathbb{N}, <)$, i.e. where the past is always bounded (non serial past). Clearly, the requirement that $\bullet\neg\varphi \Rightarrow \neg\bullet\varphi$ can not hold. However, the commutativity in this direction must hold at all points other than the beginning point. The beginning point can be characterised by remembering that it is the only point at which "yesterday false" can be true. Similarly, beware the axiom $\bullet\bigcirc\varphi \Rightarrow \bigcirc\bullet\varphi$: at the beginning point, $\bullet\bigcirc$**false** is true and $\bigcirc\bullet$**false** is clearly false; however, at all other points the implication holds. Finally, one must remember to ensure that the frame is bounded in the past, i.e. add that as an axiom. Such past linear-time temporal logics were introduced for specification purposes, see for example [Barringer and Kuiper, 1984], [Lichtenstein *et al.*, 1985], [Koymans and Roever, 1985].

### Decidability of $\mathcal{L}_{(\mathcal{W},\bigcirc)}$

Importantly for automation purposes, the propositional temporal logics we introduce in this chapter are all decidable, albeit of varying space and time complexity. The temporal logic $\mathcal{L}_{(\mathcal{W},\bigcirc)}$ over the natural numbers in PSPACE-complete. Given a formula $\varphi$ of $\mathcal{L}_{(\mathcal{W},\bigcirc)}$ the decision procedure works by attempting to construct a model for $\neg\varphi$. If the process is unsuccessful, i.e. no models can be constructed, then clearly the original formula $\varphi$ is valid. If on the other hand a model, or set of models, can be constructed, then clearly $\varphi$ is invalid. The correctness of the decision result relies upon the small model property, or finite model property, of $\mathcal{L}_{(\mathcal{W},\bigcirc)}$. This property establishes that if a formula $\varphi$ has a model, then it has model that can be represented finitely. For example, although the formula $\Box\Diamond p$ has a model in natural number time where the proposition $p$ has as valuation the set of points $\{i | i \text{ is prime}\}$, its models can in fact be represented by a two state structure in which one of the states has $p$ true and must be visited infinitely often on infinite paths through the structure, with either of the states may be a starting state.

**Axioms:**

| | |
|---|---|
| $\vdash w$ | tautologies |
| $\vdash \bigcirc(\varphi \Rightarrow \psi) \Rightarrow (\bigcirc\varphi \Rightarrow \bigcirc\psi)$ | K for next |
| $\vdash \bullet(\varphi \Rightarrow \psi) \Rightarrow (\bullet\varphi \Rightarrow \bullet\psi)$ | K for last |
| $\vdash \bigcirc\neg\varphi \Leftrightarrow \neg\bigcirc\varphi$ | commutativity of not & next |
| $\vdash \bullet\neg\varphi \Leftrightarrow \neg\bullet\varphi$ | commutativity of not & last |
| $\vdash \varphi \mathcal{W} \psi \Rightarrow \psi \vee \varphi \wedge \bigcirc(\varphi \mathcal{W} \psi)$ | unless "definition" as fixpoint |
| $\vdash \varphi \mathcal{Z} \psi \Rightarrow \psi \vee \varphi \wedge \bullet(\varphi \mathcal{Z} \psi)$ | zince "definition" as fixpoint |
| $\vdash \bigcirc\bullet\varphi \Leftrightarrow \varphi$ | next-last cancellation |
| $\vdash \bullet\bigcirc\varphi \Leftrightarrow \varphi$ | last-next cancellation |

**Inference Rules:**

MP
$$\frac{\vdash \varphi \qquad \vdash \varphi \Rightarrow \psi}{\vdash \psi}$$

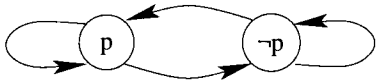
$\bigcirc$ – Gen
$$\frac{\vdash \varphi}{\vdash \bigcirc\varphi}$$

$\bullet$ – Gen
$$\frac{\vdash \varphi}{\vdash \bullet\varphi}$$

$\mathcal{W}$ – Intro
$$\frac{\vdash \xi \Rightarrow \psi \vee \varphi \wedge \bigcirc\xi}{\vdash \xi \Rightarrow \varphi \mathcal{W} \psi}$$

$\mathcal{Z}$ – Intro
$$\frac{\vdash \xi \Rightarrow \psi \vee \varphi \wedge \bullet\xi}{\vdash \xi \Rightarrow \varphi \mathcal{Z} \psi}$$

Figure 4.5: Axiomatisation for Future and Past




Note that such an abstraction of satisfying models, if used as a recogniser, would indeed recognise the model where $p$ is made true only on prime indices.

There are numerous descriptions of the basic decision/satisfiability procedure for linear temporal logic in the literature and we refer the reader to any of these for detailed proofs and constructions (for example, [Gough, 1984; Lichtenstein and Pnueli, 1985; Vardi and Wolper, 1986]). In essence, the model construction proceeds by building states, each labelled by subformulas of the formula under test that hold there, and the next-time relation between states. The determination of the next-time relation comes from the the fact that any formula of $\mathcal{L}_{(\mathcal{W},\bigcirc)}$ can be separated into a disjunction of conjunction of states formulas and next-time formulas (see also the Section 4.4.2).

### 4.4.2 More on Expressiveness

We have seen, for certain temporal frames, $\mathcal{L}_{(\boxplus,\boxminus)} < \mathcal{L}_{(\boxplus,\boxminus,\tilde{\boxplus},\tilde{\boxminus})} < \mathcal{L}_{(\mathcal{U}^+,\mathcal{S}^-)}$, i.e. starting from a minimal logic based on $\boxplus$ and $\boxminus$, we've been able to add new temporal modalities, not expressible in the previous ones, that have gained expressiveness within particular classes of frames. We've presented this in a fairly natural and intuitive way, however, there was little logical strategy. So questions of expressive completeness, or even connective completeness, should be considered. In boolean logic, we know there are only 16 different binary connectives (there are only 16 combinations $2\times 2$ tables of boolean values). It is easy to show that each of these 16 binary connectives can be expressed in terms of, say, just negation and conjunction, or negation and disjunction, or implication and falsity, etc. We have a notion of functional completeness. Turning to our temporal connectives, we introduced $\tilde{\boxplus}$ and $\tilde{\boxminus}$ as new temporal connectives and then showed that they could not be expressed in terms of $\boxplus$ and $\boxminus$. Similarly for the $\mathcal{U}^+$ and $\mathcal{S}^-$ connectives. So it is natural to ask whether there is some similar notion to functional completeness for our temporal logics. First we must note that functional completeness is the wrong notion — it clearly is inapplicable. However, what we're trying to get at is a formal notion of expressiveness. For that some well understood base is required. We fix on a first order language which can be interpreted over essentially the same models as our temporal languages, and then formally compare expressiveness with respect to the known base.

Let $\mathcal{L}_1$ be such a first order language with $=, <$ and unary predicates $q_i, i \in \mathbb{N}$. We interpret $\mathcal{L}_1$ in models similar to those for $\mathcal{L}_{(\mathcal{U}^+,\mathcal{S}^-)}$, i.e. structures $(T, <, Q_i)$ where $Q_i \subseteq T$ for $i \in \mathbb{N}$.

There is a natural transformation of $\mathcal{L}_{(\mathcal{U}^+,\mathcal{S}^-)}$ formulae $\varphi$ into $\mathcal{L}_1$ formulae, $\varphi^*(t)$ s.t. $\varphi$ is true at $t$ iff $\varphi^*(t)$ is true. The transformation follows the semantic descriptions given earlier. Thus $T_f^t$ is defined as:

$$
\begin{array}{lcl}
T_f^t(p) & \stackrel{def}{=} & p(t) \\
T_f^t(\varphi \wedge \psi) & \stackrel{def}{=} & T_f^t(\varphi) \wedge T_f^t(\psi) \\
\vdots & & \\
T_f^t(\varphi \mathcal{U}^+ \psi) & \stackrel{def}{=} & \exists s \cdot t < s \wedge T_f^t(\psi) \wedge (\forall u \cdot t < u \wedge u < s \Rightarrow T_f^t(\varphi)) \\
etc & &
\end{array}
$$

Variable s, above, must be chosen so not to capture others.

**Definition 4.4.1.** *The logic $\mathcal{L}_{(\mathcal{U}^+, \mathcal{S}^-)}$ is said to be expressively complete wrt $\mathcal{L}_1$ if there exists a transformation from $\mathcal{L}_1$ to $\mathcal{L}_{(\mathcal{U}^+, \mathcal{S}^-)}$.*

Remember that the transformation must preserve truth, i.e. letting $\varphi_1(t)$ be a formula of $\mathcal{L}_1$ and $\varphi$ be the temporal logic formula resulting from the transformation, then $\varphi_1(t)$ is true iff $\varphi$ is true at $t$.

**Theorem 4.4.3.** *(due to Kamp/Gabbay/G.P.S.S.) The logic $\mathcal{L}_{(\mathcal{U}^+, \mathcal{S}^-)}$ is expressively complete wrt a 1st order language over complete linear orders.*

The expressive completeness of the since until language was first due to Kamp and was presented in his PhD thesis of 1968 [Kamp, 1968]. The particular work is not the easiest of reads, which therefore made the details of the result rather inaccessible. However, it is a very important and surprising result. What is so special about $\mathcal{U}^+$ and its mirror is that this restricted form of quantification can capture the arbitrary, unrestricted, form of quantification allowed by the 1st order language.

Others have since produced more understandable proofs, but ones which are still not that easy. Indeed Gabbay, in [Gabbay, 1981], produced a more interesting result which was based on a syntactic separation property of temporal languages (discussed in the next section).

Gabbay, Pnueli, Shelah and Stavi ([Gabbay *et al.*, 1980]) produced yet another proof of expressive completeness, more readable than Kamp's original.

Define a first order formula $\varphi(t)$ as a *future formula* if all quantification is restricted to the future of $t$, i.e. $\exists y \cdot t < y$

**Theorem 4.4.4.** *(due to Gabbay, Pnueli, Shelah and Stavi) The logic $\mathcal{L}_{(\mathcal{U}^+)}$ is expressively complete wrt the future formulae of $\mathcal{L}_1$.*

### Gabbay's Separation Result

Consider $\mathcal{L}_{(\mathcal{U}^+, \mathcal{S}^-)}$ over frames $(\mathbb{N}, <)$. Define the subsets of $\mathcal{L}$, $wff^0$ – pure present formulae, $wff^0$ – pure past formulae, and $wff^+$ – pure future formulae. The separation result was produced simply as a means to establishing a more approachable proof of expressive completeness of temporal languages. Indeed Gabbay succeeded in providing a more general route than Kamp, as well as being more accessible.

**Theorem 4.4.5. Separation Theorem** *(due to Gabbay). Any formula $\varphi$ of $\mathcal{L}_{(\mathcal{U}^+, \mathcal{S}^-)}$ can be written as a boolean combination of formulae from $wff^- \cup wff^0 \cup wff^+$.*

**Theorem 4.4.6.** *(due to Gabbay). Given $\mathcal{L}$ as a temporal language with $\diamondsuit$, $\overleftarrow{\diamondsuit}$ and the separation property, $\mathcal{L}$ is expressively complete (over complete linear orders).*

However, the separation result is of more interest than just as a tool for improving difficult, or even unreachable, proofs; it is a result of importance in its own right. For example, it is a basis for creating models which satisfy temporal formulae, and consequentially a basis for temporal logic programming. Because of this, we look at the proof of separation in a little more depth.

The key idea behind the proof of the separation theorem is the use of a systematic procedure that extracts nested occurrences of $\mathcal{U}^+$, resp. $\mathcal{S}^-$, from within a $\mathcal{S}^-$, resp. $\mathcal{U}^+$,

formula until there are no nestings of $\mathcal{U}^+$ within $\mathcal{S}^-$ and vice versa. For $\mathcal{U}^+$ within $\mathcal{S}^-$, there are eight basic cases to handle. Let $\varphi$ and $\psi$ stand for arbitrary formulae and letters $a$ and $b$, etc., denote propositional atoms. The eight cases are:-

1. $\varphi\,\mathcal{S}^-\,(\psi \wedge a\,\mathcal{U}^+\,b)$
2. $\varphi\,\mathcal{S}^-\,(\psi \wedge \neg(a\,\mathcal{U}^+\,b))$
3. $(\varphi \vee a\,\mathcal{U}^+\,b)\,\mathcal{S}^-\,\psi$
4. $(\varphi \vee \neg(a\,\mathcal{U}^+\,b))\,\mathcal{S}^-\,\psi$
5. $(\varphi \vee a\,\mathcal{U}^+\,b)\,\mathcal{S}^-\,(\psi \wedge a\,\mathcal{U}^+\,b)$
6. $(\varphi \vee \neg(a\,\mathcal{U}^+\,b))\,\mathcal{S}^-\,(\psi \wedge a\,\mathcal{U}^+\,b)$
7. $(\varphi \vee a\,\mathcal{U}^+\,b)\,\mathcal{S}^-\,(\psi \wedge \neg(a\,\mathcal{U}^+\,b))$
8. $(\varphi \vee \neg(a\,\mathcal{U}^+\,b))\,\mathcal{S}^-\,(\psi \wedge \neg(a\,\mathcal{U}^+\,b))$

Other nested $\mathcal{U}^+$ forms reduce to one of the 8 schema for atomic $\mathcal{U}^+$ formula. For example, consider $\varphi\,\mathcal{S}^-\,(a\,\mathcal{U}^+\,(p\,\mathcal{U}^+\,q))$, however, this can be viewed as a formula of shape 1 above. Replace the sub-formula $p\,\mathcal{U}^+\,q$ by $pq$ say, and note that the formula $\psi$ in 1 is **true**.

For each of the above shapes, one can provide an equivalent formula of form $E_1 \vee E_2 \vee E_3$ where each $E_i$ is a boolean combination of pure past, present and pure future formulae. An inductive proof can then establish that separation can occur for all formulae.

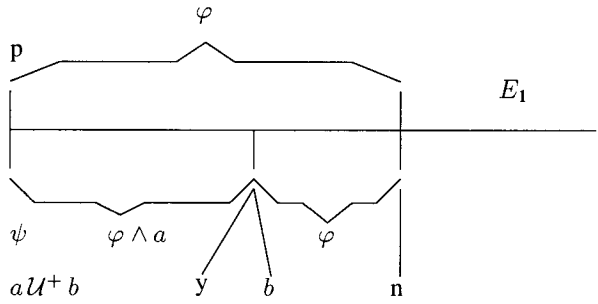
In the following we establish the first of the above eliminations. Let $E \stackrel{def}{=} \varphi\,\mathcal{S}^-\,(\psi \wedge a\,\mathcal{U}^+\,b)$. We can write $E$ as the disjunction of $E_1E_2E_3$ such that the $E_i$ contain no nested $\mathcal{U}^+$, in fact in a separated form,

$$\models E \Leftrightarrow E_1 \vee E_2 \vee E_3$$

In order to construct the formulae $E_i$, consider a model for $\varphi\,\mathcal{S}^-\,(\psi \wedge a\,\mathcal{U}^+\,b)$.

$$\begin{array}{lcll} E_1 & : & y < n & : \quad \varphi\,\mathcal{S}^-\,(b \wedge \varphi \wedge (\varphi \wedge a)\,\mathcal{S}^-\,\psi) \\ E_2 & : & y = n & : \quad b \wedge (\varphi \wedge a)\,\mathcal{S}^-\,\psi \\ E_3 & : & y > n & : \quad a\,\mathcal{U}^+\,b \wedge a \wedge (\varphi \wedge a)\,\mathcal{S}^-\,\psi \end{array}$$

Let n be the point in the model at which we are evaluating the formula (which we will assume to be true). By definition of the $\mathcal{S}^-$ connective, there is a point before n, say p, at which $\psi \wedge a\,\mathcal{U}^+\,b$ holds. Consider then the formula $a\,\mathcal{U}^+\,b$ at point p. Clearly there is a point beyond p at which $b$ holds, name that point y. y must either be earlier than, equal to, or later than the point n. We thus have three cases, the first of which is illustrated below.



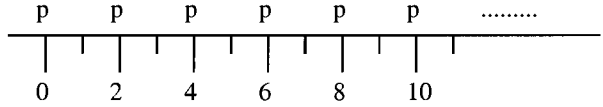

So consider the first case, i.e. y<n. We can describe the situation, from n, as $\varphi$ holding back to y, at which $b \wedge \varphi$ hold and from there $\varphi \wedge a$ hold back to p at which $\psi$ holds. This is a pure past formula. The situation when y=n can be given as a boolean combination of pure past and present time formulas. The case for y>n as a boolean combination of pure past, present, and pure future formulas. We are thus done, since the original formula is equivalent to one of these three, i.e. the disjunction of the three cases, which is a boolean combination of pure past, present and pure future formulas.

### 4.4.3 Is $\mathcal{L}_{(\mathcal{U}^+, \mathcal{S}^-)}$ expressive enough?

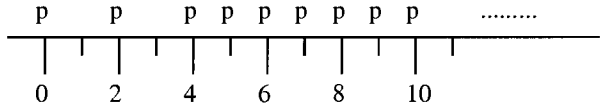
In the previous section we've identified the expressiveness of the $\mathcal{L}_{(\mathcal{U}^+, \mathcal{S}^-)}$ logic as begin equivalent to the first order language of linear order: we've precisely answered the question of how expressive the language is. However, when considering applications of temporal logic, questions relating to whether this logic is expressive enough are still likely to arise. Of course, we can continue increasing expressiveness until we can distinguish every feature of some proposed model - we very much doubt, however, whether such a logic would be useful! Indeed there is always some trade-off. For us, one important trade-off is between expressiveness and complexity of the decision process (or, worse still, how undecidable a logic may be). So one natural question arises: how much richer, or more expressive, can we make our temporal language whilst maintaining decidability? Further interesting extensions can be made, and there is a need for such. Wolper was one of the first to propose a decidable extension of the until temporal language that fulfilled a need in program specification. We proceed by highlighting his example, then introduce two alternative extensions to the until language of more or less equivalent expressiveness.

#### Regular properties

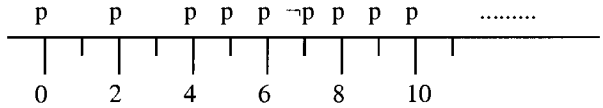
Consider the statement "the clock ticked on every even moment of time": can such a statement be expressed in some way in the until/since version of linear discrete temporal logic? To make a little more precise whilst capturing the essence of the example let us rephrase the problem as:- construct a temporal formula of the logic $\mathcal{L}_{(\mathcal{W}, \bigcirc)}$ that characterises that atom $p$ holds in every even moment of the frame $(\mathbb{N}, <)$. As a first attempt, consider the formula $\Box(p \Rightarrow \bigcirc\bigcirc p) \wedge p$. If this formula is true at the first moment in time, i.e. time point 0 in the frame, it will clearly give that the atomic proposition $p$ will be true there and at every even moment thereafter, as in the diagram below.




However, this formula asserts that the subformula $p \Rightarrow \bigcirc\bigcirc p$ holds at every moment in time. Therefore if $p$ happens to be true at some odd moment in time, it will be true on every odd moment thereafter as well, as depicted in the next diagram.




Such a situation is rather undesirable because no constraints should be placed on $p$ on odd moments of time, for example, the situation in the next diagram should be perfectly acceptable.



Well, no matter how hard one tries to characterise of this particular evenness property, failure is guaranteed. Not unsurprisingly, there are related properties which are expressible. For example, consider the stronger constraint that the atomic proposition $p$ is to be true on just the even moments of time (and hence nowhere else). The following formula of $\mathcal{L}_{(\mathcal{W},\mathcal{Z},\bigcirc,\bullet)}$ will characterise the constraint*.

$$\Box((p \Rightarrow \bigcirc\neg p) \wedge (\neg p \Rightarrow \bigcirc p) \wedge (\bullet\mathbf{false} \Rightarrow p))$$

Wolper [Wolper, 1983] introduced an extension of linear discrete temporal logic, aptly named Extended Temporal Logic, or ETL, in which one could define new temporal connectives based on regular grammars. Wolper further established that the logic was complete (although see [Banieqbal and Barringer, 1986]) with respect to $\omega$-regular languages, precisely those accepted by finite state Buchi-automata. Importantly, ETL was thus a decidable extension. The essence of ETL is to define n-ary grammar operators, with production rules of the form $V_i = u_j V_k$ where $V_i, V_k$, etc., are non-terminal symbols and letters $u_j$ denote terminal symbols. With respect to linear discrete models, a grammar operator, say $g_i(p_1, \ldots, p_n)$ holds at a point $t$ in the model if and only if there is an expansion $u_{i0}u_{i1}\ldots$ of the non-terminal $V_i$ such that each parameter, $p_{im}$, obtained by substitution of $p_j$ for $u_j, j = 1, \ldots, n$, holds at $t + m$.

**Motivational examples** Consider the production rule $V_1 = u_1 V_1$ and the evaluation of the grammar operator $g_1(p)$ at $s$. It defines that the proposition $p$ must hold on every moment $t$, $t \geq s$, in other words $g_1(p)$ is equivalent to $\Box p$. On the other hand, consider the evaluation of $\neg g_1(\neg p)$ at $s$, which can only be satisfied on a model that does not have $p$ false in every moment $t$, $t \geq s$ - in other words it corresponds to $\Diamond p$.

Given the production rules, $V_1 = u_1 V_1$, $V_1 = u_2 V_2$ and $V_2 = u_3 V_2$, the formula $g_1(p, q, \mathbf{true})$ will correspond to $p\,\mathcal{W}\,q$: there is an unwinding of $V_1$ that has just $u_1$ for ever, and there are unwindings that have finite iteration of $u_1$ fby $u_2$ fby infinite iteration of $u_3$. On the other hand $\neg g_1(\neg q, \neg(p \vee q), \mathbf{true})$ corresponds to $\neg(\neg q\,\mathcal{W}\,\neg(p \vee q))$, i.e. $p$ **until** $q$.

Finally consider the production rules, $V_1 = u_1 V_2$ and $V_2 = u_2 V_1$, then the evenness property, i.e. $p$ should be true in every even moment from now, can be characterised by the formula $g(p, \mathbf{true})$.

Rather than examine ETL in detail, in the next sections we prefer to introduce Quantified Propositional Temporal Logic and the Fixed Point Temporal Logic, both syntactically more convenient extensions of linear temporal logic.

### 4.4.4 Quantified Propositional Temporal Logic

Consider the temporal logic $\mathcal{L}_{(\Box,\blacksquare,\bigcirc,\bullet)}$, i.e. the propositional temporal logic built using just the temporal modalities, $\Box$, $\bigcirc$ and their past time counterparts. This is not the most expressive temporal logic we have seen so far, however, this is rich enough for our purposes.† The extension we now make is to introduce quantification over atomic propositions, which then yields quantified propositional temporal logic, QPTL. The following are examples of QPTL formulae:

$$p \qquad \bigcirc(p \Rightarrow q) \qquad \bigcirc\Box(p \Rightarrow \bigcirc q)$$
$$\exists x \cdot (p \Rightarrow x) \qquad \exists x \cdot \Box(p \equiv \bigcirc x) \qquad \exists x \cdot \Box(x \Rightarrow p \wedge \bigcirc\bigcirc x) \wedge x$$

*The past is required in order to characterise the first point in time.

†In fact, our extension will subsume the other logics we have seen so far.

The interpretation of QPTL formulae is pretty routine. The first three examples, which have no quantification, are interpreted as before. Thus $p$ is true in a model $M = (T, <, V)$ at time point $t$ if $t$ is a member of $V(p)$, etc. On the other hand, the formula $\exists x \cdot (p \Rightarrow x)$ is true at time point $t$ if one can make an assignment to the proposition $x$ that will make the formula $p \Rightarrow x$ true at time point $t$, i.e. make $x$ true at $t$. The formula $\exists x \cdot \Box(p \Leftrightarrow \bigcirc x)$ will be true at $t$ if an assignment can be made to proposition $x$ that will make $\Box(p \equiv \bigcirc x)$ true at $t$, which requires that the valuation of $x$ at every point $s > t$ is the same as the valuation of proposition $p$ at each points $s - 1$. More formally, we have the following semantic definition for existential quantification.

$$M \models_t \exists x \cdot \varphi(x) \quad \text{iff} \quad \text{there exists } M' = (T(M), <, V') \text{ s.t.}$$
$$V' \text{ differs from } V(M) \text{ just on } x \text{ and } M' \models_t \varphi(x)$$

Universal quantification is then defined in the usual way in terms of existential quantification, namely:

$$\forall x \cdot \varphi(x) \stackrel{def}{=} \neg \exists x \cdot \neg \varphi(x)$$

Consider now the evaluation of the formula $\exists x \cdot \Box(x \Rightarrow p \wedge \bigcirc\bigcirc x) \wedge x$ in model $M = (\mathbb{N}, <, V)$ where $V(p) = \{0, 2, 4, 6, \ldots\} \cup \{3, 7\}$. Let the given formula be denoted by $\exists x \cdot \varphi(x)$. For $M \models_0 \exists x \cdot \varphi(x)$ to hold we must find an assignment for $x$, $V_x$, such that for $M$ updated by $V_x$, i.e. $M'$, we have $M' \models_0 \varphi(x)$. From the definition of formula $\varphi(x)$, since $x$ must be true at time point 0 and the fact that whenever $x$ is true, say at $t$, it must also be true at $t + 2$, an assignment for $x$ must have $V(x) = \{0, 2, 4, 6, \ldots\}$. For the formula $\varphi(x)$ to be true, we must also have $p$ true whenever $x$ is true, i.e. $V(x) \subseteq V(p)$, which is the case. Therefore $M \models_0 \exists x \cdot \Box(x \Rightarrow p \wedge \bigcirc\bigcirc x) \wedge x$. Of course, the formula we've just evaluated characterises the evenness property of the above section.

**Theorem 4.4.7.** *The temporal logic $\mathcal{L}_{tw,\bigcirc}$ is contained in QPTL.*

In order to establish this result we need to show how any $\mathcal{L}_{tw,\bigcirc}$ formula, say $\varphi$ can be represented by a formula in QPTL, say $\psi$, such that $M \models_t \varphi$ if and only if $M \models_t \psi$ for all $t$ in $M(T)$. The proof follows by induction over the structure of formulae. We need to consider only the $\mathcal{W}$ case, since other formula have direct correspondents. Assume that $tr(A)$, $tr(B)$ are QPTL formulae equivalent to $\mathcal{L}_{tw,\bigcirc}$ formulae $A$ and $B$. We assert that the QPTL formula

$$\exists x \cdot \Box(x \Leftrightarrow (tr(B) \vee tr(A) \wedge \bigcirc x)) \wedge x \wedge maximal(x)$$
$$\text{where } maximal(x) \stackrel{def}{=} \forall y \cdot \Box(y \Leftrightarrow (tr(B) \vee tr(A) \wedge \bigcirc y)) \Rightarrow \Box(y \Rightarrow x)$$

is true on just the models that $A \mathcal{W} B$ is true, and vice-versa. The first clause of the translation corresponds, in a sense, to the axiom that defines $A \mathcal{W} B$ is a solution to the equation $x = B \vee A \wedge \bigcirc x$, and the formula $maximal(x)$ corresponds to the unless introduction inference rule characterising that the formula $A \mathcal{W} B$ is a maximal solution to the equation.

**Theorem 4.4.8.** *QPTL is decidable, with non-elementary complexity.*

An axiomatisation for QPTL is given in [Kesten and Pnueli, 1995] and shown to be complete with respect to left bounded linear discrete models, the logic is equipped with both past and future temporal modalities.

### 4.4.5 A Fixed Point Temporal Logic

Another approach to extending the expressiveness of, say, future time temporal logic has been to start with a language with just one temporal modality (next) and then allow recursively defined formulas. Such languages have been found most natural for describing computations (particularly in a compositional fashion). We will exemplify this approach here in the context of linear discrete temporal logic. Our presentation follows that of [Banieqbal and Barringer, 1987], but see [Vardi, 1988; Kaivola, 1995] for alternative presentations. We construct a temporal language $\nu$TL (over $(\mathbb{N}, <)$) from a propositional logic, e.g.

| | | | |
|---|---|---|---|
| $p$ | atomic propositions | $x$ | proposition (recursion) variables |
| $\neg$ | negation | $\bigcirc$ | next time temporal modality |
| $\wedge$ | and | $\nu$ | fixed point constructor |
| $\vdots$ | $\vdots$ | | |

Open well formed formulae, owff, are defined inductively by

$p \in$ owff, $x \in$ owff
if $\varphi, \psi \in$ owff, then so are $\neg\varphi, \varphi \wedge \psi, \ldots, \bigcirc\varphi$
if $\chi(x) \in$ owff, with proposition variable $x$ free, then $\nu x.\chi(x) \in$ owff

Closed well formed formulae, wff, are then open well formed formulas that have no free proposition variables. We take the language $\nu$TL to be the set of closed well formed formulae.

In order to define the semantics of $\nu$TL formulas we adopt a slightly different viewpoint. In essence, we are just changing our view so that we think of the sets of models that satisfy a temporal formula. This enables us to apply standard techniques to construct solutions to recursive definitions. For ease of presentation we will restrict our logic to a linear discrete temporal frame structure $(\mathbb{N}, <)$, thus a model $M = (\mathbb{N}, <, V)$ over which we have the successor function $(+1)$. Let $U_M$ denote the set of all such possible models and let $\mathcal{M}^i_f$ be the set of models which satisfy the $\nu$TL formula $f$ at time $i$. Then, by induction over the structure of formulae we have the following definition.

$$\mathcal{M}^i_p = \{M | M, i \models p\} \qquad \mathcal{M}^i_{\bigcirc\varphi} = \mathit{Shift}(\mathcal{M}^i_\varphi)$$
$$\mathcal{M}^i_{\varphi\wedge\psi} = \mathcal{M}^i_\varphi \cap \mathcal{M}^i_\psi \qquad \mathcal{M}^i_{\neg\varphi} = \overline{\mathcal{M}^i_\varphi} \text{ i.e. } U_M - \mathcal{M}^i_\varphi$$

where we define the set theoretic function *Shift* as

$$\mathit{Shift}(\mathcal{M}) = \{(\mathbb{N}, <, V') | (\mathbb{N}, <, V) \in \mathcal{M} \text{ and } V' = \mathit{shift}(V)\}$$
$$\mathit{shift}(V)(p) = \{i + 1 | i \in V(p)\} \text{ for any proposition } p$$

Thus $\mathcal{M}^i_p$ is just the set of models which have proposition $p$ true at time point $i$. Note that no constraint is placed on the valuation of any other proposition, or on the valuation of $p$ at any point other than $i$. Slightly more interestingly, $M^i_{\bigcirc\varphi}$ is defined as the set of models that satisfy $\varphi$ at $i + 1$, although this is achieved through the auxiliary function *Shift* that literally shifts the evaluation of every proposition forwards by one moment. Thus the set of models $\mathcal{M}^i_{\bigcirc p}$ will be the $Shift$ of the set of models each having $p$ true at $i$, resulting in the set of models that have $p$ true at $i + 1$. The definitions of the boolean connectives follows their set theoretic counterparts, from which it follows, of course, that $\mathcal{M}^i_{\neg\varphi} = \{M | M, i \not\models \varphi\}$.

Let us now consider the fixed point form $\nu x.f(x)$. First note that if $g$ is a $\nu$TL formula then so is $f(g) = f[g/x]$, the result of substituting g for the free occurrences of $x$ in $f(x)$. From the definition above, $\mathcal{M}^i_{f(g)}$ depends solely on $\mathcal{M}^i_g$, hence one can construct a function $F$ on sets of models, corresponding to $f$, such that, for any given formula $g$ we have,

$$\mathcal{M}^i_{f(g)} = F(\mathcal{M}^i_g)$$

Consider the sets $\mathcal{M}^i$ such that $\mathcal{M}^i = F(\mathcal{M}^i)$. If there is a maximum set among them, i.e. one which contains all the others, then we say that $\nu x.f(x)$ exists and that $\mathcal{M}^i_{\nu x.f(x)}$ is that set. It follows that since we then have

$$\mathcal{M}^i_{\nu x.f(x)} = F(\mathcal{M}^i_{\nu x.f(x)}) = \mathcal{M}^i_{f(\nu x.f(x))}$$

we have that

$$\nu x.f(x) = f(\nu x.f(x))$$

Thus $\nu x.f(x)$ is a solution to $x = f(x)$ and every other solution $y$ satisfies $y \Rightarrow \nu x.f(x)$. We consider the necessary conditions for its existence a little below.

**Examples** Consider the following examples of fixed point formulae.

1. $\nu x.p \wedge \bigcirc x$.

   If the above is true at $i$ in some model $M$, then $M$ has $p$ true at $i$ and all points beyond in the future. Let $f(x) = p \wedge \bigcirc x$ and thus $F(\mathcal{M}^i) = \{M | i \in V(M)(p), M \in Shift(\mathcal{M}^i)\}$. Since we identify $\mathcal{M}^i$ with $F(\mathcal{M}^i)$, the maximal set $\mathcal{M}^i$ that is a solution must have a valuation for $p$ that is the set of all $j \geq i$, i.e. $p$ is true at $i$ and all points beyond. This is the same as the previous semantics of $\Box p$. Thus we can indeed define $\Box\varphi \stackrel{def}{=} \nu x.\varphi \wedge \bigcirc x$.

2. $\nu x.p \wedge \bigcirc\bigcirc x$.

   If this is true at $i$ then $p$ has to be true at all points $i + 2 * j$ for $j \geq 0$, i.e. $p$ has to be true at all even moments beyond $i$.

3. $\neg\nu x.\neg p \wedge \bigcirc x$.

   If this is true at $i$, then there is a $j \geq i \in V(p)$, otherwise $p$ would be false everywhere beyond (and including) $i$, i.e. $p$ is true sometime in the future.

4. $\nu x.q \vee p \wedge \bigcirc x$.

   If this formula is true at $i$, it characterises models that have $p$ holding forwards from $i$ until at least $q$ (which may never occur, in which case $p$ holds for ever from $i$).

**A small extension** The first of the above examples showed that the formula $\nu x.p \wedge \bigcirc x$ corresponds to the $\mathcal{L}_{(\mathcal{W},\bigcirc)}$ formula $\Box p$. In a similar way that we defined $\Diamond\varphi$ as $\neg\Box\neg\varphi$, so we can define a minimal fixed point formula. Indeed, let

$$\mu x.f(x) \stackrel{def}{=} \neg\nu x.\neg f(\neg x)$$

when the right-hand side formula exists. Thus we have

$$\neg f(\mu x.f(x)) \Leftrightarrow \neg f(\neg \nu x.\neg f(\neg x)) \Leftrightarrow \nu x.\neg f(\neg x)$$

And therefore $\mu x.f(x)$ is a fixed point of $f$, i.e. $f(\mu x.f(x)) = \mu x.f(x)$. Indeed, it can be shown that $\mu x.f(x)$ is a minimal solution. If $f(g) = g$ then $\neg f(\neg(\neg g)) = \neg g$, hence $\neg g \Rightarrow \nu x.\neg f(\neg x)$, i.e. $\neg \nu x.\neg f(\neg x) \Rightarrow g$ and thus $\mu x.f(x) \Rightarrow g$. Therefore $\mu x.f(x)$ is minimal.

Note we can define:

$$\begin{array}{llll} \Box\varphi & \stackrel{def}{=} & \nu x.\varphi \wedge \bigcirc x & \Diamond\varphi \stackrel{def}{=} \mu x.\varphi \vee \bigcirc\varphi \\ \varphi \mathcal{W} \psi & \stackrel{def}{=} & \nu x.\psi \vee \varphi \wedge \bigcirc x & \varphi \textbf{ until } \psi \stackrel{def}{=} \mu x.\psi \vee \varphi \wedge \bigcirc x \end{array}$$

**Existence of fixed points** Consider the following $\nu$TL formula, $\nu x.p \wedge \bigcirc \neg x$. Let $M$ be a model satisfying $p$ and $Shift(M) = M$. We must have that $M$ satisfies $x$ if and only if $M$ satisfies $\neg x$. But this is a contradiction. Therefore the given formula denotes no set of models, i.e. there is no solution to the equation $y = p \wedge \bigcirc \neg y$. On the other hand, the equation $y = p \wedge x \wedge \bigcirc \neg y$ has solutions but no unique maximal solution - **false** is a solution. So, the issue to be explored is under what conditions do maximal and minimal fixed point formula exist?

We call $f$ monotone if whenever $x \Rightarrow y$ we have that $f(x) \Rightarrow f(y)$. For such functions, the fixed points always exist and can be constructed by approximation. Given a formula $x$, we define, for ordinal $\alpha$ and $\beta$, $f^{\alpha}_{\wedge}$ and $f^{\alpha}_{\vee}$ inductively as follows.

$$f^{\alpha}_{\wedge}(x) = \begin{cases} f(f^{\alpha-1}_{\wedge}(x)) & \text{if } \alpha \text{ is not limiting} \\ \bigwedge_{\beta<\alpha} f^{\beta}_{\wedge}(x) & \text{if } \alpha \text{ is limiting} \end{cases}$$

$$f^{\alpha}_{\vee}(x) = \begin{cases} f(f^{\alpha-1}_{\vee}(x)) & \text{if } \alpha \text{ is not limiting} \\ \bigvee_{\beta<\alpha} f^{\beta}_{\vee}(x) & \text{if } \alpha \text{ is limiting} \end{cases}$$

And then $\nu x.f(x) = f^{\alpha}_{\wedge}(\mathbf{true})$ for some ordinal $\alpha$, and $\mu x.f(x) = f^{\alpha}_{\vee}(\mathbf{false})$ for some ordinal $\beta$.* Let us exemplify the construction. Consider the evaluation of formula $\nu x.p \wedge \bigcirc x$ in model $M$ at time point $i$. There is some ordinal $\alpha$ such that $M, i \models f^{\alpha}_{\wedge}(\mathbf{true})$ where $f(x) = p \wedge \bigcirc x$: clearly $f(x)$ in $x$. Let $\mathcal{M}^0$ be the set of models $M^0$ that satisfy $f^0(\mathbf{true})$ at time point $i$, i.e. the formula **true**. And then $\mathcal{M}^1$ be the set of models $M^1$ that satisfy $f^1(\mathbf{true})$, namely $p \wedge \bigcirc\mathbf{true}$, at time point $i$. Consider the first limiting ordinal $\omega$: the set of models satisfying $f^{\omega}(\mathbf{true})$ will be intersection of the sets $\mathcal{M}^n$ for all $n \in \mathbb{N}$. Each model in this set will have $p$ true at every time point $j \geq i$. Further iteration over this set of models does not cause change in the set. Therefore we have indeed found the set of models satisfying the maximal fixed point of $f$. These are, of course, precisely those models satisfying $\Box p$ at $i$.

As one further example, consider the maximal and the minimal solutions to the equation $f(x) = q \vee p \wedge \bigcirc x$. First note that $f(x)$ is monotone in $x$. The construction of the maximal solution of $f$ requires iteration again to the first limiting ordinal $\omega$. The set of models includes those that either have $q$ true at some point $k \in \mathbb{N}$ and then $p$ at all points $j \geq i$ and $j < k$, or have $p$ true at all points $j \geq i$. The latter set of models is present in the original set (**true**)

*This actually presents an alternative way to define the semantics of the fixed point formulae in the case that the function $f$ is monotone.

and never gets removed by the iteration. On the other hand, the construction of the minimal solution to $f$ starts from the empty set and adds all models satisfying the property that $q$ is true at some future point and $p$ is true up to that point. The model with $p$ true everywhere (from $i$) and $q$ never true (from $i$ onwards) is never added. The minimal fixed point formula thus corresponds to $p$ **until** $q$ and the maximal fixed point formula is the weak version, namely $p \mathcal{W} q$.

The examples we've shown so far have no nesting of fixed points. However, our language allows such formulae. Suppose therefore that $f(x, y)$ is monotone in both variables $x$ and $y$; it can be shown that

$$\text{if } x \Rightarrow x' \text{ then } \nu y.f(x, y) \Rightarrow \nu y.f(x', y)$$

It follows that a general condition for monotonicity of $f(x)$ is that $x$ must occur under an even number of negations. If this is the case for all bound variables of a fixed point formula, then the fixed point does exist. For example, $\nu x.(a \wedge \bigcirc \neg \nu y.(\neg x \wedge \bigcirc y))$ is defined, $x$ appears under two negations, the innermost being applied direct to $x$, then another encompassing negation applied to the immediately surrounding $\nu$ formula. The use of negation applied to bound variables, in such cases, can be avoided by the use of minimal fixed point formulae. The example just given can be rewritten as $\nu x.(a \wedge \bigcirc \mu y.(x \vee \bigcirc y))$. Indeed, if a formula $f$ has no negation symbols applied to bound variables, then the formula $\neg f$ can also be written without negation applied to bound variables.

**Decidability** The propositional temporal fixed point logic, $\nu$TL, over linear discrete frames $(\mathbb{N}, <)$, is decidable. A decision procedure for the logic was given in [Banieqbal and Barringer, 1987] and relied on the property that if a formula is satisfiable then it is satisfiable on an eventually periodic model, which only requires iteration (of recursion formulas) up to the first limiting ordinal $\omega$ to show satisfaction. An alternative approach to the problem was adopted in [Vardi, 1988]. See also the decidability results on the propositional $\mu$-calculus [Streett and Emerson, 1984].

**Axioms:**

$\vdash w$ tautology
$\vdash \bigcirc(\varphi \Rightarrow \psi) \Rightarrow (\bigcirc\varphi \Rightarrow \bigcirc\psi)$
$\vdash \bigcirc\neg\varphi \Leftrightarrow \neg\bigcirc\varphi$
$\vdash \nu x.\chi(x) \Leftrightarrow \chi(\nu x.\chi(x))$

**Inference Rules:**

Modus Ponens

$\bigcirc$ – Gen $\dfrac{\vdash \varphi}{\vdash \bigcirc\varphi}$

$\nu$ – Intro $\dfrac{\vdash \xi \Rightarrow \chi(\xi)}{\vdash \xi \Rightarrow \nu x.\chi(x)}$

Figure 4.6: Axiomatisation for $\nu$TL.

**Proof System for $\nu$TL** Consider a fixed point temporal logic over linear discrete frames $(\mathbb{N}, <)$. The proof system given in Figure 4.6 follows the approach for the $\mathcal{L}_{(W,\bigcirc)}$ system. Soundness is straightforward; on the other hand, completeness remained an open problem for several years, although a solution has been given in [Kaivola, 1995].

## 4.5 Branching Time Temporal Logic

We have just examined a range of temporal logics that all possess, one way or another, a constraint for linearity - each model will be structurally indistinguishable (in a zig-zag sense) from some linearly ordered structure. Let us restrict attention to a future-time linear discrete temporal logic, i.e. one that can only reason forwards into the future, so there are no past time temporal operators. Clearly the removal of the linearity constraint for such a logic will yield a temporal logic whose models may have a branching structure, i.e. each time point may have more than one immediate successor. Such a logic will also admit models where different branches rejoin at some future time, see Figure 4.7 for example. This may

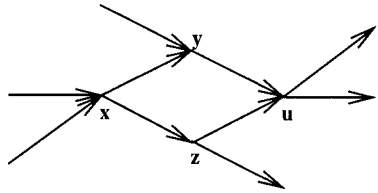



Figure 4.7: A Non Linear Structure

be quite acceptable for the desired use of the logic, however, there are obviously situations where it would not be so. Can we constrain the logic such that the determined models are pure tree structures, i.e. with only a linear past? This is a straightforward exercise using past time modalities for we simply keep a formula such as WPC present as an axiom, but how can it be expressed without the past? The first order formula

$$\forall y, z. \neg(y < z \vee z < y \vee y = z) \Rightarrow \neg\exists u. y < u \wedge z < u$$

certainly rules out backward branching, but this is not expressible with only future time modalities. Weakening the above formula by moving the two unrelated points into the future of a reference point, i.e. now, just as was done with weak future (past) connectedness, will in part solve the problem. For example, in Figure 4.7 the point $u$ is preceded by two distinct unrelated points, $y$ and $z$, i.e. there is no way to move from $y$ to $z$ or vice-versa; $y$ and $z$ can both be reached, however, from $x$. Indeed, the formulation will be quite adequate for all those situations where there is an initial point to the temporal flow. Thus we have

$$\forall x, y, z. x < y \wedge x < z \wedge \neg(y < z) \wedge \neg(z < y) \wedge \neg(y = z) \Rightarrow \neg\exists u. y < u \wedge z < u$$

which has corresponding temporal formula

$$\begin{aligned} &\Diamond Y \wedge \Diamond Z \wedge \neg \Diamond (Y \wedge \Diamond Z) \wedge \neg \Diamond (Z \wedge \Diamond Y) \wedge \neg \Diamond (Y \wedge Z) \Rightarrow \\ &\quad \neg( \Diamond U \wedge \Diamond (Y \wedge \Diamond U) \wedge \Diamond (Z \wedge \Diamond U)) \end{aligned}$$

and hence the structure of Figure 4.7 would not be admissible. It should be emphasised that our concern over ruling out all branching past temporal flows without resort to past time modalities is, perhaps, somewhat academic since those backward branching models not ruled out will not be distinguishable from backward linear models using a future only language. Thus we have a future-time temporal logic determining forward branching tree structures.

Now let us consider the effect of the temporal modalities introduced in the previous section when interpreted over future branching discrete tree structures*. Assume a frame structure $(T, <)$ where $T$ denotes a set of discrete nodes and $<$ is an asymmetric, transitive relation from which a successor relation $\mathcal{N}$ is defined. Additionally, assume that the frame satisfies the first of the above constraints.

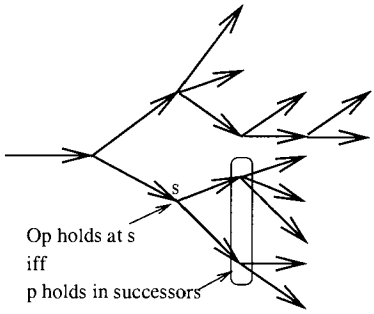



Figure 4.8: Branching tree

Recall that the next time modality $\bigcirc$ was defined as a universal, or weak, next, namely:

$$M, s \models \bigcirc\varphi \quad \textbf{iff} \quad \forall t. s\mathcal{N}t \Rightarrow M, t \models \varphi$$

and thus, on a branching model, $\varphi$ has to be true in all the successors of $s$ for $\bigcirc\varphi$ to hold at $s$. So in Figure 4.8, $\varphi$ must be true in both the successors of point $s$ in order for $\bigcirc\varphi$ to hold at $s$. Given the universal interpretation to $\bigcirc$ it follows that the $\boxplus$ modality then reaches all possible future states from the given valuation state, whereas its dual modality, $\diamondplus$, will find some future state, as illustrated in figure 4.9 below. Suppose, however, one wished to express a property about every state on some path through a computation tree, or that some property at some future state of every possible computation path, as in Figure 4.10. Is the language of $\mathcal{L}(\mathcal{W}, \bigcirc)$ expressive enough to do so? Well, the answer is no. Within the language we do have the dual of the $\bigcirc$ modality, i.e. $\odot$ an existential, or strong, next defined by $\neg\bigcirc\neg$: $\odot\varphi$ is true at node $s$ if and only there is at least one successor $t$ where $\varphi$ holds. In order to obtain a modality to capture that a particular property holds at each moment along some path, we would clearly need to use the strong version of next. The languages of QPTL, the fixed point temporal logic, and ETL are all, indeed, rich enough to do so, when interpreted over branching structures. Because we have already shown how to translate from one to the other† we'll just demonstrate expression of these different modalities with $\nu$TL.

*Such a structure is often used as a basis for a model of the possible computation states of a program

†Of course, our presentation only considered these languages over linear structures, however, interpretation over non linear structures is a straightforward exercise

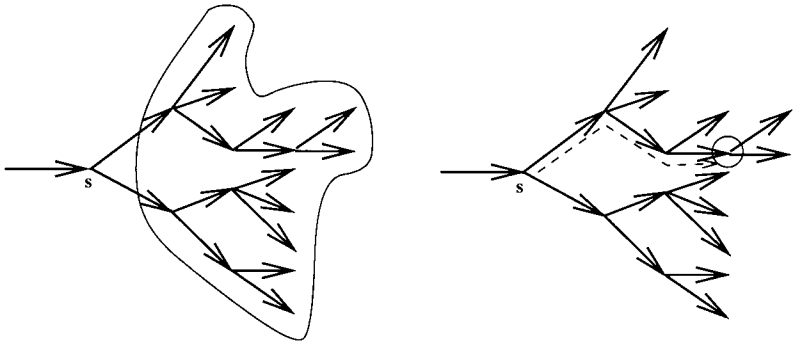



Figure 4.9: $\boxplus$ and $\diamondplus$ on trees

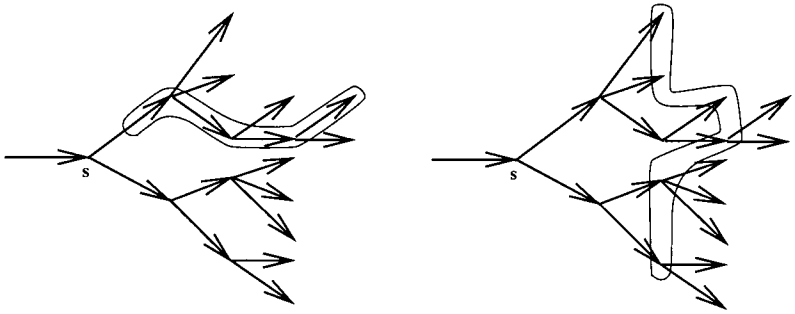



Figure 4.10: Paths and cuts on trees

The fixed point formula

$$\nu x.\varphi \wedge \odot x$$

could be used to express the suggested property. $M, s \models \nu x.\varphi \wedge \odot x$ if and only if there is some infinite path $\sigma$ starting from $s$ (i.e. $\sigma_0 = s$) such that $M, \sigma_i \models \varphi$ for every $i > 0$. The weaker formula, on the other hand,

$$\nu x.[\varphi \wedge (\odot \mathbf{true} \Rightarrow \odot x)]$$

would be satisfied on a future finite path, provided $\varphi$ holds at each moment along it of course. The "cut", or wave front, property can be expressed by using the dual of $\nu x.\varphi \wedge \odot x$, namely by the formula

$$\mu x.[\varphi \vee (\odot \mathbf{true} \wedge \bigcirc x)]$$

which requires the minimal fixed point to hold in every successor of some point $s$ if it doesn't get satisfied at $s$ itself, and hence eventually (because it's minimal) along every path from $s$.The recognition of the usefulness of a language with such modalities led researchers to develop the branching time temporal logic CTL [Clarke and Emerson, 1981b], i.e. Computation Tree Logic, and an important number of extensions. It was a key development which spearheaded the now widely accepted use of model-checking as an automated verification technique, [Clarke *et al.*, 1983; Clarke *et al.*, 1986; Clarke *et al.*, 1999; Clarke and Schlingloff, 2001].

## 4.6 Interval-based Temporal Logic

So far, the temporal logics that we've introduced have been point-based, be they over discrete or dense structures, linear or branching temporal flows, etc.; propositions are given truth values at each individual time-point in the model. Although for the majority of applications of temporal logics in computational contexts point-based logics are quite appropriate, there are applications, particularly those dealing with representation of natural phenomena and their interactions over time, where an interval-based valuation of propositions can be more natural and intuitive. For example, "the door is closing" is a proposition that if true will always be true over some interval of time. Of course, because intervals can be, and very often are, modelled by sets of points, there is an argument that takes the line that point-based models are quite sufficient. We don't wish to explore or contribute to that particular debate*, but will take as a starting point to interval logics an extension to the point-based linear discrete temporal logic $\mathcal{L}_{(\mathcal{W},\bigcirc)}$ that enables, model theoretically (and formulaically), the sequential composition of models (and hence the composition of temporal formulas viewed as denoting intervals). This approach will lead naturally to a brief introduction to Moszkowski's work on ITL [Moszkowski, 1986]. It would have then been appropriate to review the modal system of Halpern and Shoham, [Halpern and Shoham, 1991], where intervals are adopted as the primitive underlying temporal object (from which points can be derived), then finally make comparison with Allen's interval temporal logic [Allen, 1983; Allen, 1984; Allen and Hayes, 1989; Allen, 1991b; Allen and Ferguson, 1994]; however, space does not permit further exposition here and we merely strongly recommend this area to the interested readers.

*Interested readers may follow this up through, for example, Galton's chapter in [Galton, 1987]

### 4.6.1 Introducing the chop

For simplicity of exposition, consider $\mathcal{L}_{(\mathcal{W},\bigcirc)}$ over possibly bounded natural number time, i.e. temporal frames $F = (N, <)$ for $N = \mathbb{N}$ or $\{0..i | i \in \mathbb{N}\}$ together with the successor relation $\mathcal{N}$. Furthermore, it will be easier to consider such natural number time models $M$ represented by sequences $\sigma$ of states (providing valuations to atomic propositions). We thus use the notation $\sigma, i \models \varphi$ to denote that the formula $\varphi$ is satisfied in the sequence (model) $\sigma$ at index (time point) $i$. We will add to the language of $\mathcal{L}_{(\mathcal{W},\bigcirc)}$ a modality that will correspond to the fusion of two sequences. Consider two sequences $\sigma_1$ and $\sigma_2$ the first of which is finite and the second may be infinite such that the end state of the first sequence $\sigma_1$ is the beginning state of the second sequence $\sigma_2$; we then define the fusion of $\sigma_1$ with $\sigma_2$ as

$$\sigma_1 \circ \sigma_2 \stackrel{def}{=} \iota\sigma \text{ s.t. } \sigma_1 < \sigma \text{ and } \sigma_2 = \sigma^{(|\sigma_1|-1)}$$

If one views a sequence as a point-based representation of an interval then the fusion of two sequences corresponds to their join where the last and first elements are fused together, i.e. they are the same. In a computational context this corresponds to the sequential composition of two particular computations. We now introduce a temporal modality, $\mathcal{C}$ that will achieve the effect of fusion. Informally, the formula $\phi\,\mathcal{C}\,\psi$ will be true for some sequence model at $i$ if and only if the sequence can be cut, or chopped, at $j > i$ such that $\phi$ holds on the prefix sequence up to and including $j$ and $\psi$ holds on the suffix sequence from $j$ onwards. More formally

$$\begin{aligned} \sigma, i \models \phi\,\mathcal{C}\,\psi \quad \textbf{iff} \quad & \text{either there exists } \sigma_1, \sigma_2 \text{ s.t.} \\ & \qquad \sigma = \sigma_1 \circ \sigma_2 \text{ and } \sigma_1, i \models \phi \text{ and } \sigma_2, 0 \models \psi \\ & \text{or } |\sigma| = \omega \text{ and } \sigma, i \models \phi \end{aligned}$$

This particular chop temporal modality was first introduced in order to ease the presentation of compositional temporal specifications, see [Barringer *et al.*, 1984], and was motivated by the fusion operator used in PDL [Harel *et al.*, 1980; Harel *et al.*, 1982] (see also [Chandra *et al.*, 1981a]) and by its use in Moszkowski's ITL [Moszkowski, 1986]. Its use arose in the following way. Suppose the formula $\phi$ characterises the (temporal) behaviour of some program $P$, and $\psi$ characterises program $Q$, then the formula $\phi\,\mathcal{C}\,\psi$ will characterise the temporal behaviour of the sequential composition of $P$ and $Q$, i.e. $P;Q$, the program that first executes $P$ and then, when complete, executes $Q$. An iterated version of chop, $\mathcal{C}^*$, was also introduced in [Barringer *et al.*, 1984]; informally $\phi\,\mathcal{C}^*\,\psi$ denoted the maximal solution to the implication $x = \psi \vee \phi\,\mathcal{C}\,x$ and was used to obtain a compositional temporal semantics for loops. A sound and complete axiomatisation of the logic $\mathcal{L}_{(\mathcal{W},\,c\,,\bigcirc)}$ was presented in [Rosner and Pnueli, 1986].

**Examples** To give a flavour of the linear temporal logic with chop we examine a few formulae and models in which they're satisfied. To provide a little more intuitiveness with the temporal formula, we define the following special proposition **fin**, which is true only at the end of an interval*.

$$\textbf{fin} \stackrel{def}{=} \bigcirc\textbf{false}$$

*As we are working with a future only logic we are unable to write down a formula that uniquely determines the beginning point. This is not a major problem, for we can either extend the logic with such a proposition **beg** or allow past time modalities. We will not bother to do so in this brief exposition.

Recall that the modality $\bigcirc$ has universal interpretation, hence it is vacuously true at an "end of time", but since **false** is true nowhere, **fin** is uniquely true at an end point*.

(i) $(p\,\mathcal{C}\,q) \Rightarrow p \wedge q$
(ii) $(p \wedge \mathbf{fin}\,\mathcal{C}\,q) \Rightarrow p \wedge q$
(iii) $(\Box p)\,\mathcal{C}\,(p\ \mathbf{until}\ q) \Rightarrow p\ \mathbf{until}\ q$
(iv) $(p\ \mathbf{until}\ q) \Rightarrow [((\Box p)\,\mathcal{C}\,q) \vee ((\Box p)\,\mathcal{C}\bigcirc q) \vee q]$

Which of the above formulae are valid for the linear discrete frames? The first is obviously invalid: consider a frame with just two points and construct a model with $p$ is true only at the first point and $q$ true in the second - $p\,\mathcal{C}\,q$ is clearly true at the first point in the model, but $p$ and $q$ aren't both true at that point. The second formula, however, is valid over linear discrete frames. Any model $\sigma$ which satisfies the formula $p \wedge \mathbf{fin}\,\mathcal{C}\,q$ at, say, index $i$ will have $p$ true at $i$. It must also have $q$ true at $i$ because $\sigma$ can be decomposed as two sequences, $\sigma_1$ and $\sigma_2$ such that the length of $\sigma_1$ is $i+1$ (**fin** is true at $i$), thus the $i^{\text{th}}$ state of $\sigma_1$ is also its last state and hence the first state of $\sigma_2$; therefore $q$, which is true on $\sigma_2$, will also be true on $\sigma_1$ at $i$ and hence $\sigma$ at $i$.

The third and fourth formulas above are also valid and we leave as a simple exercise for the reader.

## ITL

Moszkowski's Interval Temporal Logic (ITL), as in [Moszkowski, 1986], is essentially a point-based discrete linear-time temporal logic defined over finite sequences. In other words, the basis is the above choppy logic defined to be restricted to finite sequences. The finiteness of the intervals enabled Moszkowski, however, to develop ITL more easily as a low-level programming language - Tempura. Temporal modalities (constructs) are defined to mimic imperative programming features, such as assignment, conditional, loops, etc., and model (interval) construction became the interpretation mechanism. For example, an assignment modality can be defined in the following way.

$$
\begin{array}{lcl}
\mathbf{empty} & \stackrel{def}{=} & \neg \odot \mathbf{true} \\
e_1\ \mathbf{gets}\ e_2 & \stackrel{def}{=} & \Box(\neg \mathbf{empty} \Rightarrow ((\odot e_1) = e_2)) \\
\mathbf{stable}\ e & \stackrel{def}{=} & e\ \mathbf{gets}\ e \\
\mathbf{fin}\phi & \stackrel{def}{=} & \Box(\mathbf{empty} \Rightarrow \phi) \\
e_1 \rightarrow e_2 & \stackrel{def}{=} & \exists V.((\mathbf{stable}\ V) \wedge (V = e_1) \wedge \mathbf{fin}(e_2 = V))
\end{array}
$$

**empty** is true on empty intervals, i.e. at all points in a model where there are no future states (it corresponds to our earlier use of the **fin**). $e_1\ \mathbf{gets}\ e_2$ is true at some state in an interval when the value of expression $e_1$ is the value the expression $e_2$ in the following state of the interval (thus $x\ \mathbf{gets}\ x-2$ is true at state $s$ is the value of $x$ has decreased by 2 in the next state). $\mathbf{stable}\ e$ is true whenever the value of the expression $e$ is stable over the interval, i.e. $e$'s value remains the same. $\mathbf{fin}\,\phi$ is true anywhere within an interval if the formula $\phi$ is true at the end state of the interval. Finally, the temporal assignment, $e_1 \rightarrow e_2$ holds for an interval of states if the value of $e_2$ at the end of the interval is the value of $e_1$ at the start

*Similarly, if the temporal logic were equipped with the $\bullet$ modality, also with universal interpretation, we could define the proposition **beg**.

- the stability requirement on the value $V$ is necessary as quantification in ITL is defined for non-rigid variables. In similar ways, other imperative style programming constructs can be defined as temporal formulas, with the result that, syntactically, a Tempura formula can appear just like an imperative program.

This ITL/Tempura approach therefore provided a uniform approach to the specification and implementation of programs. Within the formal methods community of computer science there are many critics of this standpoint, however, putting aside some of these more philosophical considerations, Moszkowski's ITL marked the start of an important avenue of temporal logic applications, namely, executable temporal logic. The collection [Barringer *et al.*, 1996] and the volume introduced by [Fisher and Owens, 1995a] describe alternative approaches to executable temporal logics.

## 4.7 Conclusion and Further Reading

The range of modal varieties of temporal logic is now vast and, as will become abundantly clear to the interested reader as she delves further into the field, this chapter has barely entered the field. Its focus has primarily been limited to the development of discrete linear-time temporal logics from a modal logic basis, although it introduced the idea of branching and interval structures; of course, these latter areas warrant whole chapters, or even books, to themselves. And then there is the issue of temporal logics over dense structures, such as used in [Barringer *et al.*, 1986; Gabbay and Hodkinson, 1990] and the whole rich field of real-time, or metric, temporal logics, for example see [Alur and Henzinger, 1991; Bellini *et al.*, 2000] for two brief surveys. There are, today, a number of excellent expositions on temporal logic, from historical accounts such as [Ohrstrom and Hasle, 1995], early seminal monographs such as [Prior, 1967; Rescher and Urquhart, 1971], various treatise and handbooks such as [van Benthem, 1983; Benthem, 1984; Benthem, 1988b; Benthem, 1988a; van Benthem, 1991; Blackburn *et al.*, 2001; Gabbay *et al.*, 1994a; Gabbay *et al.*, 1994b; Goldblatt, 1987], shorter survey or handbook chapters such as [Burgess, 1984; Stirling, 1992; Benthem, 1995; Emerson, 1990], expositions on the related dynamic and process logics such as [Harel, 1979; Harel, 1984; Harel *et al.*, 1980; Harel *et al.*, 1982] to application-oriented expositions such as [Gabbay *et al.*, 2000] and then [Kroger, 1987; Manna and Pnueli, 1992; Manna and Pnueli, 1995] for specific coverage of linear-time temporal logic in program specification and verification, and [Clarke *et al.*, 1999] on model checking, [Moszkowski, 1986; Fisher and Owens, 1995b; Barringer *et al.*, 1996] on executable temporal logic.

We trust the reader will enjoy delving more deeply into these logics and their applications through the remaining chapters of this volume and the abundant associated, technical, literature in the field.

This Page Intentionally Left Blank

Handbook of Temporal Reasoning in Artificial Intelligence
Edited by M. Fisher, D. Gabbay and L. Vila


Chapter 5

# Temporal Qualification in Artificial Intelligence

Han Reichgelt & Lluis Vila

We use the term *temporal qualification* to refer to the way a logic is used to express that temporal propositions are true or false at different times. Various methods of temporal qualification have been proposed in the AI community. Beginning with the simplest approach of adding time as an extra argument to all temporal predicates, these methods move to different levels of representational sophistication. In this chapter we describe and analyze a number of approaches by looking at the syntactical, semantical and ontological decisions they make. From the ontological point of view, there are two issues: (i) whether time receives full ontological status or not and (ii) what the temporally qualified expressions represent: *temporal types* or *temporal tokens*. Syntactically, time can be explicit or implicit in the language. Semantically a line is drawn between methods whose semantics is based on standard first-order logic and those that move beyond the semantics of standard first-order logic to either higher-order semantics, possible-world semantics or an *ad hoc* temporal semantics.

## 5.1 Introduction

Temporal reasoning in artificial intelligence deals with *relationships* that hold at some times and do not hold at other times (called *fluents*), *events* that occur at certain times, *actions* undertaken by an actor at the right time to achieve a goal and *states* of the world that are true or hold for a while and then *change* into a new state that is true at the following time. Consider the following illustrative example that will be used throughout the chapter:

> "On 1/4/04, SmallCo sent an offer for selling goods $g$ to BigCo for price $p$ with a 2 weeks expiration interval. BigCo received the offer three days later* and it has been effective since then. A properly formalized offer becomes effective as of it is received by the offered and continues to be so until it is accepted by the offered or it expires (as indicated by its expiration interval). Anybody who makes an offer is committed to the offer as long as the offer is effective. Anybody who receives an offer is obliged to send a confirmation to the offerer within two days."

* A more realistic and up-to-date examples might be an e-trading scenario where the messages are received 2 or 3 seconds after being sent. However, the essential representation issues and results would not be affected.

This narrative contains instances of the temporal phenomena mentioned above:

- *fluents* such as "$x$ being effective from time $t$ to time $t'$". In this case, the beginning and the end and the duration are not fully determined but the beginning is. This fluent may also hold on a set of non-overlapping intervals of time.
- *Actions* such as "an agent $x$ sending an object or message $y$ to agent $z$ at time $t$". This also may happen more than once for the same $x$, $y$ and $z$, with $t$ being the only distinctive feature.
- *Events* such as "$x$ receiving $y$ on time $t$". Both executed actions and events potentially are causes of some change in the domain. In this case, the event causes the offer to be effective as of the reception time.
- *States* such as the state before "1/Apr/04" and the state right after receiving the offer where the offer is effective and various obligations hold.

Additionally, we observe other kinds of temporal phenomena such as:

- Temporal features of an object or the object itself. For instance "the manager of SmallCo" can be a different person at different times or even "SmallCo" could denote different companies at different times depending on our timeframe.
- Temporal relations between events and fluents such as "The offer is effective as of it is received by the offered and will be so until it is accepted by the offered or it expires" or "sending an object causes the receiving party to receive it between 1 and 4 days later."
- Temporal relations between fluents such as "the offerer is committed to the offer as long as the offer is effective" or "an offer cannot be effective and expired at the same time".

Notice that references to time objects may appear in a variety of styles: absolute ("1/Apr/04"), relative ("two days later"), instantaneous (`now`), durative ("the month of march"), precise ("exactly 2 days"), vague ("around 2 days"), etc.

This example illustrates the issues that must be addressed in designing a formal language for temporal reasoning*, namely the *model of time* i.e. the set or sets of time objects (points, intervals, etc.) that time is made of with their structure, the *temporal ontology* i.e. the classification of different temporal phenomena (fluents, events, actions, etc.), the *temporal constraints language*, i.e. the language for expressing constraints between time objects, the *temporal qualification* method and the *reasoning system*. Research done on models of time, temporal ontologies, and temporal constraints is reviewed in the various chapters of this volume. In this chapter we will focus on **Temporal Qualification**:

> By a temporal qualification method we mean the way a logic (which we shall call the *underlying logic* of our temporal framework) is used to express the above temporal phenomena that happen at specific times.

* The presentation is biased towards the standard definition of first-order logic (FOL), although nothing prevents the situation of the elements described here in the context of a different logic, including non-standard semantics for FOL, modal logics and higher-order logics.

One may either adopt a well-known logic equipped with a well-defined model and proof theory as the underlying logic or define a language with a non-standard model theory and develop a proof theory for it.

The temporal qualification method is a central issue in defining a temporal reasoning framework and it is closely related to the other issues mentioned above. As said, most of these issues are discussed in detail in other chapters of this volume. We discuss them here up to level needed to make our presentation self-contained and to be able to discuss the advantages and shortcomings of each temporal qualification approach.

### 5.1.1 Temporal Reasoning Issues

#### The Model of Time

Modeling time as a mathematical structure requires deciding (i) the class or classes of the basic objects that time is composed of, such as instants, intervals, etc. (i.e. the *time ontology*) and (ii) the properties of these time sets, such as dense vs. discrete, bounded vs. unbounded, partial vs total order, etc. (i.e. the *time topology*).

This issue is discussed in chapter *Theories of Time and Temporal Incidence* in this handbook and we shall remain silent on what the best model of time is. When introducing a temporal qualification method we shall merely assume we are given a *time structure*

$$\langle \mathcal{T}_1, \ldots, \mathcal{T}_{n_t}, \mathcal{F}_{time}, \mathcal{R}_{time} \rangle$$

where each $\mathcal{T}_i$ is a non-empty set of time objects, $\mathcal{F}_{time}$ is a set of functions defined over them, and $\mathcal{R}_{time}$ is a set of relations over them. For instance, when formalizing our example we shall take a time structure with three sets: a set of *time points* that is isomorphic to the natural numbers (where the grain size is one day), the *set of ordered pairs* of natural numbers and a set of *temporal spans* or durantions that is isomorphic to the integers. $\mathcal{F}_{time}$ contains functions on these sets $\mathcal{R}_{time}$ contains relations among them

The decision about the model of time to adopt is independent of the temporal qualification method although it has an impact on the formulas one can write and the formulas one can prove. The temporal qualification method one selects will determine how the model of time adopted will be embedded in the temporal reasoning system. The completeness of a proof theory depends on the availability of a theory that captures the properties of the model of time and allows the proof system to infer all statements valid in the time structure. Such a theory, the *theory of time*, may have the form of a set of axioms written in the temporal language that will include symbols denoting functions in $\mathcal{F}_{time}$ and relations in $\mathcal{R}_{time}$. For example, the transitivity of ordering relationship (denoted by $<_1$) over $\mathcal{T}_1$ can be captured by the axiom

$$\forall\, t_1, t_2, t_3\, [t_1 \leq_1 t_2 \wedge t_2 \leq_1 t_3 \rightarrow t_1 \leq_1 t_3]$$

However, depending on the time structure and the expressive power of underlying logic it may be impossible to write a complete set of axioms in our language.

An alternative way to capture the theory of time is through an appropriate set of inference rules, typically at least one for each temporal function and relation, which indicate how these expressions can be used in generating proofs. Of course, this choice requires much more effort than the previous one.

**Temporal Constraints Language**

A second issue that needs to be addressed in designing a temporal reasoning system is the temporal constraints language, the language used to denote constraints between temporal objects. Temporal constraints are logical combinations of atoms built from *time constants* (possibly of different nature, such as "1/Apr/04" or "2 days") denoting time objects in $\mathcal{T}_1, \ldots, \mathcal{T}_{n_t}$, *time functions* that denote functions in $\mathcal{F}_{time}$ and *time predicates* symbols denoting relations in $\mathcal{R}_{time}$.

**Temporal Ontology and the Theory of Temporal Incidence**

As discussed in two previous chapters in this book ("Eventualities" by A. Galton and partly "Theories of Time and Temporal Incidence" by Ll. Vila) temporal statements can be classified in various classes (as illustrated by the different temporal phenomena in our example), each associated with a pattern of temporal incidence. Different temporal ontologies have been proposed in different contexts, such as natural language understanding and commonsense reasoning. In most cases, the result of such ontological studies is a classification of temporal relations into a number of classes $\mathcal{E}_1, \ldots, \mathcal{E}_{e_n}$ (e.g. fluents, events, etc.) that we call *temporal entities*. Each class is usually accompanied by a temporal incidence pattern that is characterized by one or more axioms written in our logical language through some sort of *temporal incidence meta-predicates*. We call the set of these axioms the *theory of temporal incidence*.

For instance, to formalize our example we decide to have a temporal ontology with the following temporal entities: $\mathcal{E}_{events}$ is the class of *events* or *accomplishments* such as "sending a legal object on time t" that occur either at a time point, i.e. one day or during a time span (several days) and $\mathcal{E}_{fluents}$ is the class of temporal relationships such as "the offer being effective as of t" that hold homogeneously throughout a number of days. Whereas the occurrence of an event over an interval is *solid*, i.e. if it occurs on a interval it does not hold on any interval that overlaps with it, the holding of a fluent over an interval is *homogeneous*, i.e. if it holds during an interval it also holds over any subinterval. For example, if we had the meta-predicate HOLDS[3], then for each fluent $R^k \in \mathcal{E}_{fluents}$

$$\forall\, t_1, t_2, x_1, \ldots, x_k \,[\, \text{HOLDS}(t1, t2, R^k(x_1, \ldots, x_k)) \rightarrow \\ \forall\, t_3, t_4 \,[\, (t_3, t_4) \subseteq (t_1, t_2) \rightarrow \text{HOLDS}(t3, t4, R^k(x_1, \ldots, x_k))]\,]$$

Although these issues are out of the scope of this chapter, we must bear in mind that the temporal qualification method determines how the temporal incidence axioms are written and formulas derived from them.

## 5.1.2 Temporal Qualification Issues

We are now in a position to focus on the issues that are determined by a temporal qualification method. In fact, it can be argued that any method of temporal qualification method can be regarded as the set of decisions made with respect to these issues:

- *The distinction between temporal and atemporal individuals.* As illustrated by the example, a distinction ought to be made between atemporal individuals (i.e. individuals that are independent of time such as the color green, the number 3, ...) and individu-

als whose existence depends on time such as "contract c1-280-440" or "the SmallCo company".

- *The distinction between temporal and atemporal functions.* The introduction of time also leads to the need to make a semantic distinction between temporal functions and classical functions, possibly co-existing in the same logic. We define a *temporal function* as a function whose value can be different at different times, for example "the manager of". We shall call $\mathcal{F}_t$ the set of temporal function symbols and $\mathcal{F}_\infty$ the set of atemporal function symbols.

- *The distinction between temporal and atemporal relations.* Similarly, a temporal logic ought to make a semantic distinction between relations whose truth-value can be different at different times, such as "agent $a_1$ sends an offer to $a_2$ to sell $g$" and those whose truth-value is independent of time such as "a contract is a legal document" and "an offer is properly formalized". Notice that the *time relations* mentioned above are in fact atemporal relations. We shall call $\mathcal{R}_t$ the set of temporal function symbols and $\mathcal{R}_\infty$ the set of atemporal predicate symbols.

- *The distinction between temporal occurrences and temporal types of occurrences.* By a temporal occurrence (namely **temporal token**) we mean a particular temporal relation that is true at a specific time (e.g. "at time t agent $a_1$ sends an offer to $a_2$ to sell $g$") as opposed to a **temporal type** that denotes the set of all the occurrences of a temporal relation (e.g. the set of all specific sending events of type "agent $a_1$ sends an offer to $a_2$ to sell $g$").

- *The specification of time and temporal incidence theories.* As we explained above, the time and temporal incidence theories are fairly independent of the temporal qualification method but our temporal qualification method ought to provide the flexibility and expressiveness needed to specify the axioms in one's time and temporal incidence theories.

- *The specification of nested temporal relations.* A "nested" temporal relation relates objects or other relations that in turn are temporal. For example, "an agent is committed for a period to send a confirmation of a certain offer". The committment, the send action and the offer are all temporal relations.

- *The specification of relations between temporal relations or their occurrences.* The paradigmatic example of this is the causal relation between two temporal relations where the first causes the latter to hold. Other examples are incompatibility between temporal relations and correlations between temporal relations.

Although in this chapter we focus on temporal qualification in AI, temporal qualification is an issue in any formal temporal representation. In this section we give a brief overview of temporal qualification in different areas, ending with temporal qualification in AI where we introduce the approaches that will be discussed in detail in the following sections.

### 5.1.3 Temporal Qualification in Logic

**Classical Logics.** Classical logics have proven useful for reasoning about domains that are atemporal (such as mathematics) or in domains where time is not a relevant feature

and can be abstracted away (e.g. a diagnostic system in a domain where the times of the relevant symptoms do not affect the result of the diagnosis). However, in many domains time cannot be disregarded if we want our logical system to be correct and complete. Logicians have studied different theories to model time and designed various temporal logics. In such logics, statements are no longer timelessly true or false but are true or false at a certain time. Temporality may be inherent in any component of the formula: functions, predicates or logical connectives. Moreover, as soon as we have a time domain, it is natural to *quantify* over time individuals.

A simple approach to formulating a temporal logic is as a particular first-order logic (FOL) with a time theory. Temporal functions and predicates are supplemented with an additional argument representing the time at which they are evaluated and time is characterized by a set of first-order axioms. Standard FOL syntax and semantics are preserved and, therefore, standard FOL proof theory is also valid. However, time axioms complicate matters. On the one hand, as discussed above, the completeness of the theorem prover depends on the existence of a complete first-order axiomatization for the intended time structures. On the other hand, the time axioms may easily lead to an explosion of the search space to be explored by the theorem prover.

It is convenient to move to a **many-sorted logic** [Cohn, 1987; Walther, 1987; Manzano, 1993; Cimatti *et al.*, 1998a] since it naturally allows one to distinguish between time and non-time individuals. Many-sorted logics do not extend FOL's expressive power (it is well-known that a many-sorted first-order logic can be translated to standard FOL) but it provides several advantages. The notation is more efficient as formulas are more readable, more "elegant" and some $\rightarrow$ can be dropped yielding more compact formulas. Semantics also can be regarded as a simple extension of FOL. Many-sorted logic therefore preserves the most interesting logical properties of FOL while it provides some potential for making reasoning more efficient. A formula parser can perform "sort checking" and some of the reasoning involving the sortal axioms can be moved into the unification algorithm. Although this leads to a more expensive unification step, this is typically more than off-set by the reduction in the search space that can be achieved through the elimination of the sortal axioms from the theory.

**Modal Logics.** An alternative way to incorporate time is by complicating the model theory, along the lines of modal logic. Using the common Kripke-style possible world semantics for modal logics, each *possible world* represents a different time while the *accessibility relationship* becomes a temporal ordering relationship between possible worlds. Different modal temporal logics are obtained by (i) imposing different properties on the accessibility relationship, and (ii) choosing different domain languages (e.g. propositional, first-order, ...). In order to provide an efficient notation, modal varieties of temporal logic use a number of temporal modal operators, operators that are applied to propositions in the domain logic and change the time with respect to which the proposition is to be interpreted. Traditionally, four primitive *modal temporal operators* are defined: $F$ (at some future time), $P$ (at some past time), $G$ (at any future time) and $H$ (at any past time). Hence $F\varphi$ denote that the formula $\varphi$ is true at some future time. Other common temporal modalities are $p$ UNTIL $q$ ($p$ is true until $q$ is true), $p$ SINCE $q$ ($p$ has been true since $q$ has been true) or AT$(t)$ $p$ ($p$ is true at time $t$).

### 5.1.4 Temporal Qualification in Databases

From a purely logical point of view, classical database applications [Ahn, 1986; Tansel *et al.*, 1993; Chomicki, 1994] have followed the first approach outlined in the previous section. In addition to the original relations and a data domain for the values of the attributes, the temporal database includes a temporal domain. Typically, temporal databases use an instant-based approach to time. Some kind of mathematical structure is imposed on instants: usually one that is isomorphic to the natural numbers. A temporal database can be abstractly defined in a number of different ways [Chomicki, 1994].

**The Model-theoretic View.** A database is abstractly viewed as a two-sorted first-order language. Each relation $P$ of arity $n$ gives rise to a predicate $R$ with arity $n+1$, where the additional argument is a time argument. Its intended meaning is as follows:

$$(a_1, \ldots, a_n, t) \in R \text{ if and only if } P(a_1, \ldots, a_n) \text{ holds at time } t$$

All $a_i$ are constant symbols denoting elements in a data domain. The set of constant symbols is possibly extended with some symbols denoting elements in the temporal domain. The theory may also add some time function and relation symbols, such as a function symbol $t+1$ to denote the time immediately following $t$ or the relation $<$ to denote temporal ordering.

Some databases require multiple temporal dimensions. The usual case is that a single temporal domain is assumed. The relational predicates are then given two temporal arguments to indicate that the relation holds between two points in time (interval timestamps), or a number of time arguments used to model multiple kinds of time. For example, in the so-called *bi-temporal databases*, one set of temporal arguments refers to the *valid time* (the time when the relation is true in the real world) and another to the *transaction time* (the time when the relation was recorded in the database) [Snodgrass and Ahn, 1986]. The different interpretations of multiple temporal attributes databases are captured by *integrity constraints*. For example, a constraint may state that the beginning of an interval always precedes its end or that transaction time is not before valid time.

**The Timestamp View.** Moving to concrete databases (database that are to be implemented and therefore must allow for a finite representation), the most useful view is the timestamp view. In this view, each tuple is supplemented with a timestamp formula possibly representing an infinite set of times. A timestamp formula is a first-order formula with one free variable in the language of the temporal domain, e.g. $0 < t < 3 \vee 10 < t$. Different temporal databases result from different decisions about what subsets can be defined by timestamp formulas. An interesting temporal domain is the Presburger arithmetic as it allows one to describe periodic sets and therefore has obvious application in calendars and repeating events.

It is not clear whether timestamps could be defined in a language richer than the first-order theory of the time domain [Chomicki, 1994]. However, there are some approaches that extend the timestamp view by associating timestamps not to tuples but to attribute values [Tansel, 1993]. Such approaches increase data expressiveness and temporal flexibility but pay for this through increased query complexity, and hence decreased efficiency.

**Temporal Query Languages.** While the temporal arguments approach has been predominant in temporal databases a wide variety of languages have been explored for querying

them. These range from logic programs with a single instantaneous temporal argument to temporal logics with modal operators such as SINCE, UNTIL, etc. Readers interested in temporal query languages are referred to the relevant chapter on this subject in this volume.

**Temporal Qualification in Computer Systems**

Computer systems can be regarded as a sequence of states. Each state is characterized by a set of propositions stating what is true at that time. Interesting reasoning tasks such as system specification, verification and synthesis can be stated in terms of logical properties that must hold at some times/states in the future when the system starts at a certain initial state.

In this context, it is appropriate to model time as an ordered, discrete sequence of time points and the dominant temporal qualification approach is modal logics. The reasons are that temporal modal operators allow one to easily express relative temporal references (e.g.,"the value of variable $a$ is $x$ until this assignment statement is executed"). Modal operators also provide a very efficient notation for various levels of nested temporal references (e.g. "p will have been true until then"). Also, the semantics fits the discrete time model very well. Since modal temporal logic is discussed at length in other chapters in this volume, we will not expand on this discussion here and merely refer the reader to these other chapters.

### 5.1.5 Temporal Qualification in AI

It has been recognized that AI problems such as natural language understanding, commonsense reasoning, planning, autonomous agents, etc. make greater demands on the expressive power of temporal logics than many other areas in computer science. For example, the temporal reasoning that autonomous agents have to undertake typically requires both relative and absolute temporal references. Autonomous agents also often require reasoning about different possible futures and, if they are to engage in abductive reasoning, they may have to consider different possible pasts in order to determine which past is the best explanation for the current state of affairs.

All techniques that have been employed in temporal databases and/or computer science have also been applied in AI:

- The method of *temporal arguments* has been an appealing method to many AI researchers because of its simplicity, the ability to use standard FOL theorem proving techniques, and the fact that its expressiveness is not as limited as has commonly been claimed [Bacchus *et al.*, 1991] if we allow temporal arguments in functions as well as in predicates.

- *Temporal Modal logics* have been appealing to those interested in formalizing natural language (the so-called *tense logics*) and formal knowledge representation.

However, it is a third family of techniques that attracted much of the attention from AI researchers, specially during the 80s and 90s, namely the *reified approach*. In the reified approach, one "reifies" temporal propositions and introduces names for them. One then uses temporal incidence predicates to express that the named proposition is true at a certain time, or over a certain interval. Classical examples of this approach are the *situation calculus* [McCarthy and Hayes, 1969; Reiter, 2001; Shanahan, 1987], McDermott's logic

for plans [McDermott, 1982], Allen's interval logic [Allen, 1984], *event calculus* [Kowalski and Sergot, 1986; Shanahan, 1987], the *time map manager* [Dean and McDermott, 1987], Shoham's logic for time and change [Shoham, 1987] Reichgelt's temporal reified logic [Reichgelt, 1989] and token reified logics [Vila and Reichgelt, 1996].

The attraction of the reified approach is to a large extent due to the fact that the inclusion of names for such entities as actions, events, properties and states in the formalism allows one to predicate and quantify over such entities, something that is not allowed in either the method of temporal arguments or in temporal modal logic. This expressive power is important in many AI applications. Even our seemingly simple example includes examples of propositions that require quantification. The proposition "An offer remains valid until it either expires or is withdraw" is most naturally regarded as involving a quantification over expiration and withdrawal events. Other examples of propositions that are best regarded as involving quantification over events and/or states include propositions such as "whenever company X is in need of cleaning services, it issues a tender document", or "State-funded agencies can only issue contracts after an open and transparent tendering process".

Although reified logics have proven very popular, they have come under attack from different angles. First, temporal reified systems have often been presented without a *precise formal semantics*. While temporal reified logics in general remain first-order, the introduction of names for events and states, and some meta-predicates to assert their temporal occurrence, means that one cannot simplistically rely on the standard semantics for first-order logic to provide a rich enough semantics for a temporal reified logic. In some cases, like Shoham's reified logic, the apparent increased expressive power is not superior to that of the standard, easy-to-define method of temporal arguments [Bacchus *et al.*, 1991]. Second, in the cases in which the expressiveness advantage is clear, the price to pay is a logic that may end being far too complex. Third, reified temporal logics also received criticisms from the ontological point view, Galton [Galton, 1991], for example, considers them *"philosophically suspect and technically unnecessary"*, as they seem to advocate the introduction of *temporal types* in the ontology. One way to escape from this criticism is to move to an ontology of temporal propositions based on *temporal tokens*. A temporal token is not to be interpreted as a type of temporal propositions but as a particular temporal instance of a temporal proposition. Such ontology has been used as the basis for some alternative temporal qualification methods such as *temporal token arguments* or *temporal token reification*.

### 5.1.6 Outline

In the following sections we describe in detail the most relevant methods of temporal qualification in AI that we briefly introduced in the previous subsection. We look at the syntactical, semantical and ontological decisions they make. As we have seen, syntactically we distinguish between those that represent times as additional arguments and those that introduce specific temporal operators. Semantically, the main distinction is between those methods that stay within standard first-order logics and those that move to some sort of non-standard semantics, either defined from scratch or by adapting some known non-classical semantics such as modal logics. Finally, from the ontological point of view, we distinguish between the methods that only give full ontological status to time from the ones that, in addition, include in the ontology denotations for temporal propositions, either as *temporal types* or as *temporal tokens*.

Each method is illustrated by formalizing our trading example. The reader should recall

that we assume we are given the following:

- **A model of time.** The *time structure* composed of the three time subdomains and a number of functions and relations (see Section 5.1.1)
- **Temporal Entities and Temporal Incidence Theory.** We have two temporal entities $\mathcal{E}_{events}$ and $\mathcal{E}_{fluents}$ (see Section 5.1.1).

We analyze the advantages and shortcomings of each method according to a set of representational, computational and engineering criteria. Among the representation criteria, we shall first look at the **expressiveness** of the language. In particular, it is important for our temporal qualification method to be able to represent the various types of propositions and axioms indicated in previous sections. The comparison will be informal and illustrated by our example. Second, we shall look at the **notational efficiency**. For a host of reasons, it is important that one is able to formalize knowledge into formulas that are compact, readable and elegant. Third, it is desirable to have an **ontology** that is clean and not unnecessarily complex. One wants to make sure that one avoids undesirable entities in one's ontology. For example, an ontology that requires one to postulate the existence of both types and tokens is suspect. On the other hand, one also wants to make sure that the entities that one postulates in one's ontology are rich enough to enable one to express whatever temporal knowledge one wants to express. A second type of criteria are **theorem proving** criteria such as soundness and completeness of the proof theory, efficiency of any theorem provers, as well as the possibility of using implementation technique to improve the efficiency of the theorem prover. Finally, we also bear in mind what one might call "engineering criteria", such as **modularity** of the method. Often temporal reasoning is but one aspect of the reasoning that the system is expected to undertake. For example, an autonomous agent needs to be able to reason not only about time but also about the intentions of other agents that it is likely to have to deal with. It would therefore be advantageous if the method of temporal qualification allows one to extend the reasoning system to include reasoning about other modalities as well.

## 5.2 Temporal Modal Logic

One possible approach to temporal qualification in AI is the adoption of modal temporal logic (*MTL*). We already briefly discussed modal temporal logic in Section 5.1.3. Moreover, the chapter in this handbook by Barringer and Gabbay is devoted to modal varieties of temporal logic, and our discussion of this approach is therefore extremely condensed.

### 5.2.1 Definition

Temporal modal logics are a special case of modal logic. Starting with a normal first order logic, one adds a number of modal operators, sentential operators which, in the case of temporal modal logic, change the time at which the proposition in its scope is claimed to be true. In other words, the problem of temporal qualification is dealt with by putting a modal operator in front of a non-modal proposition. For example, one may introduce a modal operator $P$ ("was true in the Past"). When applied to a formula $\phi$, the modal operator would change the claim that $\phi$ is true at this moment in time to one which states that $\phi$ was true some time in the past. Thus, the statement "SmallCo sent offer o1 to BigCo some time in the past" would be represented as $P\ send(sco, o1, bco)$.

Modal temporal logic, as traditionally defined by philosophical logicians, is not particularly expressive. In its simplest form, modal temporal logic only allows existential and universal quantification over the past and the future. In other words, in its simplest form, modal temporal logic contains only four modal operators, namely $P$ ("was true in the past"), $H$ ("has always been true"), $F$ ("will be true sometimes in the future") and $G$ ("is always going to be true"). Clearly, this is insufficient for Artificial Intelligence, or indeed Computer Science. For example, none of the propositions in our example could be expressed in such an expressive poor formalism. It is for this reason that a number of authors (e.g., Fischer, 1991; Reichgelt, 1989) have introduced a number of additional modal operators, such as UNTIL, SINCE and a model operator scheme AT, which takes a name for a temporal unit as argument and returns a modal operator. Alternatively, one can, as Barringer and Gabbay do in an earlier chapter in this handbook, introduce a unary predicate $p()$ for each proposition $p$ in the original -propositional- language and stipulate that $p(t)$ holds if $p$ is true at time point $t$. Thus, $p(t)$ is essentially a different notation for $\text{AT}(t)p$. One advantage of the $AT$ operator is that it is easier to see how it can be used in a full first-order logic.

Modal temporal logic inherits its model theory from generic modal logic. The standard model theory for such logics relies on the notion of a possible world, as introduced in this context by Kripke (1963). In Kripke semantics, primitive expressions, such as constants and predicates, are evaluated with respect to a possible world. Non-modal propositions can then be assigned truth values with respect to possible worlds using the standard way of doing this in first-order logic (e.g., $p \vee q$ is true in a possible world $w$ if either $p$ is true in $w$ or $q$ is true in $w$ or both are). The semantics for modal propositions is defined with the help of an accessibility relation between possible worlds. In modal temporal logic, an intuitive way of defining possible worlds is as points in time, and the accessibility relation between possible worlds as an ordering relation between possible worlds. We then say that for example the proposition $Pp$ is true in a possible world $w$ if there is a possible world $w'$, which is temporally before $w$ and in which $p$ is true. With this in mind, the definition of the semantics for other modal operators is relatively natural.

The only complication to this picture is caused by an introduction of a possible $AT$ operator scheme. Since this operator requires a name for a temporal unit as an argument, the language has to be complicated to include names for such temporal units, and the semantics has to be modified to ensure that such temporal units receive their proper denotation. Obviously, the most appropriate way to deal with this complication is to assign possible worlds as the designation of names for temporal units, and to include an additional clause in the semantics that states that the proposition $\text{AT}(t)p$ is true if $p$ is true in the possible world denoted by $t$.

### 5.2.2 Analysis

We defined a number of representational desiderata on any temporal logic. One of the criteria is the **notational efficiency** (conciseness, naturalness, readability, elegance, ...). Compared to other temporal formalisms discussed in this chapter, modal temporal logic scores well on this criterion since the temporal operators produce concise and natural temporal expressions. Another issue is the **modularity** with respect to other knowledge modalities such as knowledge and belief operators. It is straightforward to combine the syntax and semantics of a modal temporal logic with a modal logic to represent, say, knowledge. Syntactically, such a change merely involves adding a knowledge modal operator; semantically, it involves

adding an accessibility relation for this new modality. The model theory now contains two accessibility operators, one used for temporal modalities, the other for epistemic modalities.

As far as cleanness of the **ontology** is concerned, the main concern is the notion of a possible world. There is a significant amount of philosophical literature on whether possible worlds are ontologically acceptable or suspect. Without wanting to delve into this literature, it seems to us that a possible world can simply be regarded as a model for a non-modal first order language, and that this makes the notion ontologically unproblematic. There are of course additional arguments about the identity of individuals across possible worlds, but it again seems to us that this problem can be solved relatively easily by insisting that the same set of individuals be used for each possible world.

Where modal temporal logic is less successful is in its ability to represent the various sentences and axioms in our example. To formalize the statement "An offer becomes effective when is received by the offered and continues to be so until it is accepted by the offered or the offer expires" we introduce several predicates. Let $E(x)$ denote "the offer x is effective", $R(x)$ denote "the offer x is received" $A(x)$ denote "the offer x is accepted" and $X(x)$ denote "the offer x expires". The classic since-until tense logic can be used to express the example as

$$\forall\, x_a, y_a, x_o$$
$$[\, E(o(x_a, y_a, x_o))\ \text{SINCE}\ R(y_a, o(x_a, y_a, x_o)) \wedge$$
$$E(o(x_a, y_a, x_o))\ \text{UNTIL}\ (A(y_a, x_o(x_a, y_a, x_o)) \vee E(x_o(x_a, y_a, x_o)))]$$

The problem is that modal temporal logic does not allow one to quantify over occurrences of a particular event. Thus, a proposition like "every time a company makes an offer, it is committed to that offer until it either expires or has been accepted" would be impossible to express.

Although the semantics for modal temporal logics is well understood, it has to be admitted that the implementation of **automated theorem provers** for modal temporal logic is not straightforward. One could of course try to adopt a theorem prover developed for general modal logic. However, such theorem provers in general do not allow for particularly complex accessibility relationships between possible worlds. Most merely allow accessibility relations to be serial, transitive, reflexive or some combination of these. However, such properties are clearly not enough if one were to introduce intervals as one's temporal units. In other words, using a general theorem prover as a reasoning mechanism for modal temporal logic is only likely to be successful if one uses points as one's temporal units. A more promising approach would be to develop theorem provers specifically for temporal modal logic, ,a topic of ongoing research and discussed in other chapters in this volume.

## 5.3 Temporal Arguments

The oldest and probably most widely used approach to temporal qualification is the method of temporal arguments (*TA* ) as introduced in Section 5.1.3. The idea of the temporal arguments approach is to start with a traditional logical theory but to add additional arguments to predicates and function symbols to deal with time. In order to reflect the fact that the domain now contains both "normal individuals" and times, the theory is often formulated as an instance of a many-sorted first-order logic with equality.

### 5.3.1 Definition

For a given time structure $\langle \mathcal{T}_1, \ldots, \mathcal{T}_{n_t}, \mathcal{F}_{time}, \mathcal{R}_{time} \rangle$ with its FOL axiomatization and a classification of temporal entities $\{\mathcal{E}_1, \ldots, \mathcal{E}_{e_n}\}$, with each class accompanied by a temporal incidence axiomatization. We define the temporal arguments method as a many-sorted logic with the time sorts $T_1, \ldots, T_{n_t}$, one for each time set, and a number of non-time sorts $U_1, \ldots, U_n$.

**Syntax.** The vocabulary is composed of the following symbols:

- a set of function symbols $\mathbf{F} = \{f^{\langle D_1, \ldots, D_n \mapsto R \rangle}\}$. If $n = 0$, $f$ denotes a single individual from sort $R$, otherwise $f$ denotes a function $D_1 \times \ldots \times D_n \mapsto R$ and depending of the nature of the $D_i$, we distinguish between:
  - *Time functions* $\mathbf{F}_{time}$ whose domain and range are time sorts.
  - *Temporal functions* $\mathbf{F}_t$ whose range is a non-time sort and whose domain includes both time and non-time sorts.
  - *Atemporal functions* $\mathbf{F}_\infty$ whose domain and range are domain sorts.

  Time, temporal and atemporal terms are defined in the usual way.
- a set of predicates $\mathbf{P} = \{P^{\langle D_1, \ldots, D_n \rangle}\}$. If $n = 0$, $P$ denotes a propositional atom, otherwise $P$ denotes a relation defined over $D_1, \ldots, D_n$ and depending on whether $D_i$ are time or a non-time sorts we distinguish between:
  - *Time predicates* $\mathbf{P}_{time}$ whose arguments are all time sorts.
  - *Temporal predicates* $\mathbf{P}_t$ whose arguments include both time and domain sorts.
  - *Atemporal predicates* $\mathbf{P}_\infty$ whose arguments do not include any time sort.
- a set of variable symbols for each sort.

We have three classes of basic formula: atomic temporal formulas, atomic atemporal formulas and temporal constraints.

We also have the standard connectives and quantifiers.

**Semantics.** The semantics is the standard semantics of many-sorted logics. Notice that time gets full ontological status as we have one or more time sorts, but that temporal entities and temporal formulas receive no special treatment.

### 5.3.2 Formalizing the Example

Having assumed the models of time and temporal incidence indicated in 5.1, we define the following **sorts**: $T_{\text{point}}$ for time points, $T_{\text{int}}$ for time intervals, and $T_{\text{span}}$ for time spans or durations, $A$ for agents, $O$ for legal objects, $G$ for trading goods, $S$ for legal status and \$ for money. Our **vocabulary** includes the following symbols:

- a set of constants for each sort: day constants = $\{1/8/04, \texttt{now}, \ldots\}$, time interval constants = $\{3/04, 2004, \ldots\}$, time span constants = $\{3d, 2w, 1y, \ldots\}$, the constant now, agent constants = $\{John, jane, bco, sco, \ldots\}$, legal object constants = $\{o_1, o_2, \ldots\}$, etc.

- the following sets of function symbols:
  - $\mathbf{F}_{time} = \{\texttt{Next}^{\langle T_{\text{point}} \mapsto T_{\text{point}}\rangle}, +_T^{\langle T_{\text{point}}, T_{\text{span}} \mapsto T_{\text{point}}\rangle}, -_T^{\langle T_{\text{point}}, T_{\text{point}} \mapsto T_{\text{span}}\rangle}, \texttt{begin}^{\langle T_{\text{int}} \mapsto T_{\text{point}}\rangle}, \texttt{end}^{\langle T_{\text{int}} \mapsto T_{\text{point}}\rangle}, \texttt{duration}^{\langle T_{\text{int}} \mapsto T_{\text{span}}\rangle}, \texttt{interval}^{\langle T_{\text{point}}, T_{\text{point}} \mapsto T_{\text{int}}\rangle}, \ldots\}$
  - $\mathbf{F}_t = \{manager^{\langle T_{\text{int}}, A \mapsto A\rangle}\}$
  - $\mathbf{F}_\infty = \{sale^{\langle G, P \mapsto O\rangle}, offer^{\langle A, A, O, T_{\text{span}} \mapsto O\rangle}\}$
- the following sets of predicates:
  - $\mathbf{P}_{time} = \{\leq^{\langle T_{\text{point}}, T_{\text{point}}\rangle}, =^{\langle T_{\text{point}}, T_{\text{point}}\rangle}, \texttt{Meets}, overlaps, \ldots^{T_{\text{int}} \times T_{\text{int}}}, \ldots\}$
  - $\mathbf{P}_t = \mathbf{P}_{event} \cup \mathbf{P}_{fluent}$
    - $\mathbf{P}_{event} = \{send^{\langle T_{\text{point}}, A, A, O\rangle}, Receive^{\langle T_{\text{point}}, A, O\rangle}, Accept^{\langle T_{\text{point}}, A, O\rangle}, Expire^{\langle T_{\text{point}}, O\rangle}\}$
    - $\mathbf{P}_{fluent} = \{effective^{\langle T_{\text{point}}, T_{\text{point}}, O\rangle}, accepted^{\langle T_{\text{point}}, T_{\text{point}}, O\rangle}, Expired^{\langle T_{\text{point}}, T_{\text{point}}, O\rangle}\}$
  - $\mathbf{P}_\infty = \{Correct\_form^{\langle O\rangle}, \leq_{\$}^{\langle P, P\rangle}$ (that denotes the $\leq$ relation between prices)$\}$
- and a set of variable symbols for each sort.

The statements in the example can be formalized as follows:

1. "On 1/4/04 SmallCo sent an offer to BigCo for selling goods $g$ for price $p$ with a 2 weeks expiration interval."
   $send(1/4/04, sco, bco, offer(sco, bco, sale(g, p), 2w))$
2. "BigCo received the offer three days later and it has been effective since then."
   $Receive(1/4/04 + 3d, bco, offer(sco, bco, sale(g, p), 2w)) \wedge$
   $effective(1/4/04 + 3d, \text{now}, offer(sco, bco, sale(g, p), 2w))$
3. "A properly formalized offer becomes effective when it is received by the offered ..."
   $\forall\, t_1 : T_{\text{point}}, x_a, y_a : A, x_o : O, ts : T_{\text{span}},$
   $[\, Correct\_form(offer(x_a, y_a, x_o, ts)) \wedge Receive(t_1, y_a, offer(x_a, y_a, x_o, ts) \rightarrow$
   $\exists t_2 : T_{\text{point}}\, [\, effective(t_1, t_2, offer(x_a, y_a, x_o, ts) \wedge t_1 \leq t_2\, ]$
   $]$
4. "...(an effective offer) continues to be so until it is accepted by the offered or the offer expires (as indicated by its expiration interval)."
   $\forall\, t_1, t_2 : T_{\text{point}}, x_a, y_a : A, x_o : O, ts : T_{\text{span}}$
   $[\, effective(t_1, t_2, offer(x_a, y_a, x_o, ts)) \wedge t_1 \leq t_2 \rightarrow$
   $\exists t_3 : T_{\text{point}}\, [Accept(t_3, y_a, offer(x_a, y_a, x_o, ts)) \wedge t_1 < t_3 \leq t_1 + ts] \vee$
   $(t_2 = t_1 + ts \wedge Expire(t2, offer(x_a, y_a, x_o, ts)))$
   $]$
5. "Anybody who makes an offer is committed to the offer as long as the offer is effective."
   $\forall\, t_1, t_2 : T_{\text{point}}, x_a : A$
   $[\, effective(t_1, t_2, offer(x_a, \_, \_, \_)) \rightarrow Committed(t_1, t_2, x_a, offer(x_a, \_, \_, \_))\, ]$

6. "Anybody who receives an offer is obliged to send a confirmation to the offerer within two days."
   $\forall\, t : T_{\text{point}}, x_a, y_a : A, x_o : O$
   $[\, \mathit{Receive}(t, y_a, x_a, x_o) \rightarrow \mathit{Obliged}(t, t + 2d, y_a, ???)\, ]$

The "???" in the last formula indicates that it is not clear how to express that $y_a$ is obliged to "send a confirmation of $x_o$ to $x_a$" since in standard FOL we cannot predicate or quantify over propositions*. In addition to this example, there are few further general statements whose formalization is interesting to consider:

1. Time axioms: "The ordering between instants is transitive":
   $\forall\, t_1, t_2, t_3 : T_{\text{point}}\, [\, t_1 \leq t_2 \wedge t_2 \leq t_3 \rightarrow t_1 \leq t_3\, ]$

2. Temporal Incidence axioms such as "Fluents hold homogeneously":
   $\forall\, t_1, t_2, t_3, t_4 : T_{\text{point}}, x_1 : S_1, \ldots, x_n : S_n,$
   $[\, P(t_1, t_2, x_1, \ldots, x_n) \wedge t_1 \leq t_3 \leq t_4 \leq t_2 \wedge t_1 \neq t_4 \rightarrow P(t_3, t_4, x_1, \ldots, x_n)\, ]$
   This an "axiom schema" that is a shorthand for a potentially large set of axioms, one for each fluent predicate $P$ in the language.

The previous examples are instances of relations holding between temporal entities, which can be important in some applications. In common-sense reasoning and planning, for instance, it is important to specify the CAUSE relationship:

> "Whenever an offer is effective it *causes* the agent who made the offer to be committed to it as long as the offer is effective."

Again, it is not clear how to express this piece of knowledge in the method of temporal arguments since it requires the predicate **causes** to take as argument the proposition $\mathit{effective}(t_1, t_2, \mathit{offer}(x_a, y_a, x_o, ts))$ which is beyond standard many-sorted FOL. A similar problem arises when we attempt to formalize like the following properties:

- "Whenever a cause occurs its effects hold."
- "Causes precede their effects."

### 5.3.3 Theorem Proving

Defining a temporal logic as a standard many-sorted logic has the advantage that we can use the various reasoning systems available for many-sorted logics [Cohn, 1987; Walther, 1987; Manzano, 1993; Cimatti *et al.*, 1998b]. For a desired time model, it may be impossible to define a set of axioms that completely captures that model. For instance, we have taken the "set of integers" as our duration subdomain. But it is well-known that there is no complete axiomatization of the integers in first-order logic if the language includes addition. Therefore, it is important to choose a temporal structure that can be characterized fully in first-order logic, such as *"unbounded linear orders"*, *"totally ordered fields"* or some of the theories discussed in chapter "Theories of Time and Temporal Incidence".

Having a complete axiomatics and therefore a complete proof theory, though, is merely the beginning of the story. We must bear in mind that, while many-sorted logics often allow

*The reader might come up with the idea of turning temporal predicates into terms in order to be able to take them as proper predicate arguments. This is the idea of temporal reified logics that we discuss below.

one to delete sortal axioms, such as "All offers are legal documents", the inclusion of a number of time sorts and predicate symbols with a specific meaning (as determined by the properties of the model of time adopted) requires one to add a potentially large number of axioms that capture the nature of the temporal incidence theory. These axioms can be a heavy load for our theorem prover as they often lead to a significant increase in the size of the search space. This problem may lead to the unavoidable effort in developing a specialized temporal theorem prover.

### 5.3.4 Analysis

The method of temporal arguments has a number of advantages over other approaches to temporal qualification. First, the **ontology** that one is committed to is relatively straightforward. In addition to "normal" objects, one merely has to add time objects to one's ontology. Compared to the ontologies that underlie the other approaches to temporal quantification, the ontology is both parsimonious and clean. Moreover, again in contrast with some of the approaches discussed in this chapter, the system does not make any ontological commitments itself, and one is therefore completely free to make the ontology as parsimonious as the application allows.

Second, despite it seeming simplicity, the **expressive power** of languages embodying the temporal arguments approach exceeds that of many other approaches to temporal quantification. The inclusion of additional temporal arguments in predicate and function symbols allows one both to express information about individuals and their properties at specific times and to quantify over times. Moreover, it is straightforward to include purely temporal axioms explicitly in one's theories. However, this is not to say that the method of temporal arguments gives one all the desired expressive power. For example, as we indicated in the previous section, since it stays within the expressive limitations of first-order logic, it is not possible to express temporal incidence properties for all temporal entities in class (fluents, events and so on) or any other property or relation about temporal entities such as "event e at time t causes fluent f to be true at time t".

Third, the **notation** is perhaps not as efficient as some of the alternatives, specifically modal logic. Many of the modal temporal operators are a notational shortcut for existential or universal quantification. For example, the modal operator $F$ provides an existential quantification over future times. Since no such notational shortcuts exist in systems based on the method of temporal arguments, the expression of sentences becomes more tedious in such systems. This is true in particular of sentences that require embedded temporal quantification, such as "The contract will have been signed by then".

Fourth, as we already indicated in the previous section, the fact that the method of temporal arguments is based on a standard first-order logic means that one can use the tried and tested **theorem proving** methods for such systems, which is not the case of methods based on a temporal logic with a non-standard temporal semantics. Moreover, setting up the system as an instance of a multi-sorted logic allows one to take advantage of the more efficient theorem provers developed for such logic. However, it is important to mention that the fact that one is forced to include explicit axioms describing temporal structures in one's theories has detrimental effects on the performance of the actual theorem provers. Many of the additional axioms lead to an combinatorial explosion of the search space and therefore significantly increase the time required to find a proof. For example, some axioms, such as for every point in time, there is a later point in time, are recursive and, unless carefully

controlled, lead to an infinite search space.

Finally, since the arguments that are added to the predicate and function symbols denote time, the method of temporal arguments does not easily lend itself to the **modular** inclusion of other modalities, such as epistemic or deontic modalities.

The methods that we discuss below have been developed to overcome some of the shortcomings associated with the method of temporal arguments. One way of increasing the expressive power of the formalism without moving to a higher-order logic is through the addition of some vocabulary and a complication in the ontology. The *temporal token arguments* is one such approach.

## 5.4 Temporal Token Arguments

The temporal token argument method (*TTA* ) was introduced in early AI temporal databases such as the *Event Calculus* [Kowalski and Sergot, 1986] and Dean's *Time Map Manager* [Dean and McDermott, 1987] and later presented in [Galton, 1991] in deeper detail. It is based on the simple idea, common in the database community, of introducing a key to identify every tuple in a relation. Here, a tuple of a temporal relation represents an instance of that relation holding at a particular time or time span. Therefore, we introduce a key that identifies a temporal instance of the relation, namely a *temporal token*, which shall receive full ontological status.

### 5.4.1 Definition

Given a time structure $\langle \mathcal{T}_1, \ldots, \mathcal{T}_{n_t}, \mathcal{F}_T, \mathcal{R}_T \rangle$ and a set of *temporal entities* $\{\mathcal{E}_1, \ldots, \mathcal{E}_{n_e}\}$, we define a standard many-sorted first order language with the following *sorts*: one time sort $T_1, \ldots, T_{n_t}$ for each set of time objects, a number of non-time sorts $U_1, \ldots, U_n$ and one token sort $E_1, \ldots, E_{n_e}$ whose union is called *tokens* for each temporal entity.

**Syntax.** The syntax is very similar to the temporal arguments method but instead of having extra time arguments in our temporal predicates, the extra argument is a single *temporal token* term. Token terms also appear as arguments to (i) time functions, and (ii) the temporal incidence predicates introduced below. The vocabulary is extended accordingly:

- *Function symbols*: In addition to the function symbol sets introduced in our discussion of the method of temporal arguments, we have a set of **time-token** functions that map tokens to their relevant times.

- *Predicate symbols*: *Temporal predicates* no longer have any time argument, but instead have a single token argument from the sort of the temporal entity denoted by the temporal predicate. Thus, *effective*$(t_1, t_2, \textit{offer}(\_))$ becomes *effective*$(tt_1, \textit{offer}(\_))$ where $tt_1$ is a constant symbol of the new $E_{\text{fluent}}$ sort.

  *Time predicates* and *Atemporal predicates* remain the same. However, we incorporate one new **Temporal Incidence Predicate (TIP)** for each temporal entity $\mathcal{E}_i$. TIPs take as sole their argument a term of the temporal sort $E_i$. Given our temporal ontology we have 2 TIPs: HOLDS$(tt_1)$ expresses that the fluent token $tt_1$ holds throughout the time interval denoted by the term `interval`$(tt_1)$ and OCCURS(*event token*) for event occurrences.

**Semantics.** The standard many-sorted first-order semantics is preserved with both time domains, non-time domains and temporal token domains with the usual interpretation of function and predicate symbols. Time and temporal incidence theories are incorporated as a set of first order axioms.

**Token Incidence Theory.** The specific semantics of temporal tokens may yield some additional temporal incidence axioms. An example is the so-called "maximality of fluent tokens". For efficiency reasons, one is interested in adopting the following convention:

> "A fluent token denotes a *maximal* piece of time where that fluent is true."

A consequence of this is the following property "Any two intervals associated with the same fluent are either identical or disjoint." Thus, in practice in can be interesting to define some additional incidence predicates such as $\text{HOLDS}_{at}{}^2$ and $\text{HOLDS}_{on}{}^2$ which are shorthands for

$$\text{HOLDS}_{at}(\textit{fluent}, t) \equiv \exists f : E_{\text{fluent}}\ (fluent(f) \wedge \text{HOLDS}(f) \wedge i \in \texttt{interval}(f))$$
$$\text{HOLDS}_{on}(\textit{fluent}, I) \equiv \exists f : E_{\text{fluent}}\ (fluent(f) \wedge \text{HOLDS}(f) \wedge I \subseteq \texttt{interval}(f))$$

respectively, where $f$ is a variable of the *fluent token* sort $E_{\text{fluent}}$ and $fluent(f)$ denotes the atomic proposition fluent with the extra temporal token argument $f$.

## 5.4.2 Formalizing the Example

We illustrate the approach by formalizing the example. We make the same assumptions as before and we will frequently refer to the formalization of this example in *TA* method.

In addition to the **sorts** defined in the *TA* example, we introduce sorts for tokens of each temporal entity: $E_{\text{event}}$ for event tokens and $E_{\text{fluent}}$ for fluent tokens. In turn, our vocabulary will include event token constants and fluent token constants. Besides the usual functions, we have the following **time-token functions**: $t : E_{\text{token}} \mapsto T_{\text{point}}$, $\texttt{begin} : E_{\text{token}} \mapsto T_{\text{point}}$, $\texttt{end} : E_{\text{token}} \mapsto T_{\text{point}}$ and $\texttt{interval} : E_{\text{token}} \mapsto T_{\text{int}}$.

In addition to the time and atemporal predicates from the previous formalization, the *temporal predicates* now are as follows:

- *Events*: $send^{\langle E_{\text{event}}, A, A, O\rangle}$ (where the last argument denotes the event token of this particular send event), $Receive^{\langle E_{\text{event}}, A, A, O\rangle}$, and $Accept^{\langle E_{\text{event}}, A, O\rangle}$.
- *Fluents*: $effective^{\langle E_{\text{fluent}}, O\rangle}$ (where the first argument denotes the fluent token of a particular period where the legal object $O$ is effective), $accepted^{\langle E_{\text{fluent}}, O\rangle}$ and $Expired^{\langle E_{\text{fluent}}, O\rangle}$.

As in the *TA* method, we have four classes of basic formula: atomic atemporal formula, atomic temporal formula, temporal constraints and temporal incidence formula.

The statements in the example can be formalized as follows:

1. "On 1/4/04, SmallCo sent an offer to BigCo for selling goods $g$ for price $p$ with a 2 weeks expiration interval."
   $send(s_1, sco, bco, offer(sco, bco, sale(g, p), 2w)) \wedge \text{OCCURS}(s_1) \wedge t(s_1, 1/4/04)$

2. "BigCo received the offer three days later and it has been effective since then."
$$\begin{array}{l} \mathit{Receive}(r_1, bco, \mathit{offer}(sco, bco, \mathit{sale}(g,p), 2w)) \ \wedge \ \textsc{Occurs}(r_1) \ \wedge \\ t_(r_1) = 1/4/04 + 3d \ \wedge \\ \mathit{effective}(e_1, \mathit{offer}(sco, bco, \mathit{sale}(g,p), 2w)) \ \wedge \ \textsc{Holds}(e_1) \ \wedge \\ t_(r_1) = \mathtt{begin}(e_1) \ \wedge \ \mathtt{end}(e_1) = now \end{array}$$

3. "A properly formalized offer becomes effective when is received by the offered ..."
$$\begin{array}{l} \forall\, tt_1 : E_{\text{event}}, ts : T_{\text{span}}, x_a, y_a : A, \ x_o : O \\ \quad [\ \mathit{Correct\_form}(\mathit{offer}(x_a, y_a, x_o, ts)) \ \wedge \\ \quad \mathit{Receive}(tt_1, y_a, x_a, \mathit{offer}(x_a, y_a, x_o, ts)) \ \wedge \ \textsc{Occurs}(tt_1) \ \rightarrow \\ \quad \exists\, tt_2 : E_{\text{fluent}} \\ \qquad [\ \mathit{effective}(tt_2, \mathit{offer}(x_a, y_a, x_o, ts)) \ \wedge \ \textsc{Holds}(tt_2) \ \wedge \ tt_1 \ \mathtt{Meets} \ tt_2 \ ] \\ \quad ] \end{array}$$

4. "...(an effective offer) continues to be so until it is accepted by the offered or the offer expires (as indicated by its expiration interval)."
$$\begin{array}{l} \forall\, tt_1 : E_{\text{fluent}}, x_a, y_a : A, \ x_o : O, \ ts : T_{\text{span}} \\ \quad [\ \mathit{effective}(tt_1, \mathit{offer}(x_a, y_a, x_o, ts)) \ \wedge \ \textsc{Holds}(tt_1) \ \rightarrow \\ \quad \exists tt_2 : E_{\text{event}} \\ \qquad [\mathit{Accept}(tt_2, y_a, \mathit{offer}(x_a, y_a, x_o, ts)) \ \wedge \ \textsc{Occurs}(tt_2) \ \wedge \\ \qquad \mathtt{begin}(tt_1) < t_(tt_2) \leq \mathtt{begin}(tt_1) + ts] \\ \quad \vee \\ \quad (\mathtt{end}(tt_1) = \mathtt{begin}(tt_1) + ts \ \wedge \\ \quad \exists tt_2 : E_{\text{event}} \\ \qquad [\mathit{Expire}(tt_2, \mathit{offer}(x_a, y_a, x_o, ts)) \ \wedge \ \textsc{Occurs}(tt_2) \ \wedge \\ \qquad \mathtt{end}(tt_1) = t_(tt_2)]) \\ \quad ] \end{array}$$

5. "Anybody who makes an offer is committed to the offer as long as the offer is effective."
$$\begin{array}{l} \forall\, tt_1 : E_{\text{fluent}}, x_a, y_a : A, x_o : O, ts : T_{\text{span}} \\ \quad [\ \mathit{effective}(tt_1, \mathit{offer}(x_a, y_a, x_o, ts)) \wedge \textsc{Holds}(tt_1) \ \rightarrow \\ \quad \exists\, tt_2 : E_{\text{fluent}} \\ \qquad [\ \mathit{Committed}(tt_2, x_a, \mathit{offer}(x_a, y_a, x_o, ts)) \wedge \textsc{Holds}(tt_2) \ \wedge \\ \qquad \mathtt{interval}(tt_1) = \mathtt{interval}(tt_2) \ ] \ ] \end{array}$$

6. "Anybody who receives an offer is obliged to send a confirmation to the offerer within two days."
$$\begin{array}{l} \forall\, tt_1 : E_{\text{event}}, x_a, y_a : A, x_o : O, ts : T_{\text{span}} \\ \quad [\ \mathit{Receive}(tt_1, y_a, \mathit{offer}(x_a, y_a, x_o, ts)) \wedge \textsc{Occurs}(tt_1) \rightarrow \\ \quad \exists tt_2 : E_{\text{event}} \\ \qquad [\mathit{Obliged}(x_a, tt_2) \wedge \mathit{send}(tt_2, y_a, x_a, \mathit{conf}(x_o)) \ \wedge \\ \qquad t_(tt_1) \leq t_(tt_2) \leq t_(tt_1) + 2d \ ] \ ] \end{array}$$

Observe that we express that $x_a$ is obliged to a temporal proposition by using a temporal token of that proposition. In general, the additional flexibility of temporal tokens allows us (i) to talk about temporal occurrences that may or may not happen, and (ii) to express that an agent is obliged to that event. This is not possible in the *TA* method.

The more general statements are formalized as follows:

- Time axioms are expressed as usual:
  $\forall\, t_1, t_2, t_3 : T_{\text{point}}[\, t_1 \leq t_2 \wedge t_2 \leq t_3 \rightarrow t_1 \leq t_3\,]$

- Temporal Incidence axioms become more compact since we can quantify over all the instances of a given entity (e.g. all fluents) independently of their particular meaning. It is no longer necessary to have an "axiom schema". For instance the "homogeneity of fluent holding" is stated by:
  $\forall\, tt : E_{\text{fluent}}, I : T_{\text{int}}\, [\, \mathit{holds}(tt) \wedge I \subseteq \texttt{interval}(tt) \rightarrow \text{HOLDS}_{on}(tt, I)\,]$

- "It is necessary for an offer to be properly written to be effective".
  $\forall\, tt : E_{\text{fluent}}, x_o : O\, [\mathit{effective}(tt, x_o) \wedge \text{HOLDS}(tt) \rightarrow \mathit{Correct\_form}(x_o)]$

- "Whenever an offer is effective it *causes* the agent who made the offer to be committed to it for as long as the offer is effective."
  $\forall\, tt_1 : E_{\text{fluent}}, x_a, y_a : A, x_o : O, ts : T_{\text{span}}$
  $[\, \mathit{effective}(tt_1, \mathit{offer}(x_a, y_a, x_o, ts)) \wedge \text{HOLDS}(tt_1) \rightarrow$
  $\exists tt_2 : E_{\text{fluent}}$
  $[\, \text{CAUSE}(tt_1, tt_2) \wedge \mathit{Committed}(tt_2, \mathit{offer}(x_a, y_a, x_o, ts)) \wedge \texttt{interval}(tt_1, tt_2)\,]\,]$

- "Whenever a cause occurs its effects hold."
  $\forall\, tt_1 : E_{\text{event}}, tt_2 : E_{\text{fluent}}$
  $[\text{OCCURS}(tt_1) \wedge \text{CAUSE}(tt_1, tt_2) \rightarrow \text{HOLDS}(tt_2)]$

- "Causes precede their effects."
  $\forall\, tt_1 : E_{\text{event}}, tt_2 : E_{\text{fluent}}$
  $[\, \text{CAUSE}(tt_1, tt_2) \rightarrow (\text{OCCURS}(tt_1) \rightarrow \text{HOLDS}(tt_2) \wedge t_(tt_1) \leq \texttt{begin}(tt_2)\,]$

### 5.4.3 Analysis

*TTA* has several advantages. The extra objects, i.e. the temporal tokens, introduced in the language gives the **notation** increased flexibility and helps overcome some of the expressiveness problems that we identified in the *TA* method. First, as the example shows, as temporal tokens are used as argument of other predicates they are useful to express nested temporal references. Second, different levels of time are supported by diversifying the time-token functions. For instance, we may have $\texttt{begin}_{\text{v}}(tt_1)$ to refer to *valid time* and $\texttt{begin}_{\text{t}}(tt_1)$ to refer to *transaction time*. Third, at the implementation level, a different temporal constraint network instance is maintained for each time level. Every temporal term will be mapped to a node in its corresponding constraint network.

However, the increased notation flexibility causes the notation to be more baroque and sometimes awkward (compare the formalization of our example here with the formalizations obtained by other methods). To improve notational *conciseness* we can define some syntactic sugar that allows the omission of token symbols whenever they are not strictly necessary.

Another advantage of this approach is its **modularity**. A clear separation is made between the temporal and other information as a atomic temporal formulas are linked to time through time-token functions like `begin` and `end`. However, token symbols can also be used as the link to other modalities as the deontic modalities of commitment and obligation illustrated by the example.

## 5.5 Temporal Reification

Temporal reification (*TR* ) was motivated by the desire to extend the expressive power of the temporal arguments approach while remaining within the limits of first order logic. It is achieved by: (i) complicating the underlying ontology and (ii) representing temporal propositions as terms in order to be able to predicate and quantify over them.

In essence, in reified temporal logic, both time objects and temporal entities receive full ontological status and one introduces in the language terms referring to them. The *Temporal Incidence Predicates* are used to associate a temporal entity with its time of occurrence and allow a direct and natural axiomatization of the given temporal incidence properties, as illustrated in the example below.

**Syntax.** Reified temporal logics are in fact relatively straightforward to construct from a standard first order language. First, it is useful to move to a sorted logic in which we make a distinction between temporal entities, normal individuals and temporal units. Second, for each $n$-place function symbol in the first order language, one introduces a corresponding $n$-place function symbol in the reified language. Its sortal signature is that it maps $n$ normal individuals into a normal individual. For each $n$-place predicate in the original language, one also introduces a $n$-place function symbol in the language. However, its sortal signature is different. It takes as input $n$ normal individuals and maps them into a temporal entity.

**Semantics.** Interestingly, not many authors worried about providing a clear model-theoretic semantics for their formalism, either because they were not interested in doing so, or because they believed that reified temporal logic would simply inherit its semantics from first order predicate calculus. It was not until [Shoham, 1987] that the semantics of reified temporal logics became an issue. Shoham observed that reified temporal logic are very similar to formalizations of the model theory for modal temporal logic in a first order logic and proposed to formulate the semantics for reified temporal logic in these terms. It is not clear that the actual framework proposed by Shoham actually achieved this. For example, [Vila and Reichgelt, 1996] argue that Shoham's formalism is more appropriately regarded as being a hybrid between a modal temporal logic and a system in the tradition of the temporal arguments method. As a matter of fact, Shoham's is subsumed by the *TA* method [Bacchus *et al.*, 1991]. Nevertheless, Shoham's insight was the inspiration for Reichgelt [Reichgelt, 1989] who indeed formulated a reified temporal logic.

### 5.5.1 Formalizing the Example

We make the same assumptions and we shall be continuously referring to the formalization of this example made with the temporal arguments method in our attempt to formalize the example in reified temporal logic.

Besides the **sorts** $T_{\text{point}}$ for time instants, $T_{\text{int}}$ for time intervals, $T_{\text{span}}$ for durations, $A$ for agents, etc. we now have additional sorts, one for each temporal entity: $E_{\text{event}}$ for events and $E_{\text{fluent}}$ for fluents. Notice that, although we use the same names that *TTA* , there are ontological differences since here they denote temporal types whereas in *TTA* they denote temporal tokens.

Our **vocabulary** is composed of:

- For each sort, a set of constant symbols, including event constants and fluent constants.
- We have time, temporal and atemporal function symbols as in the temporal arguments approach except that the set of temporal functions (where we have functions like $offer^{\langle A,A,O,T\mapsto O\rangle}$) is extended with new temporal functions produced by temporal reification, one for each temporal relation (which in the temporal arguments is represented by a temporal predicate):
  - $E_{\text{event}} = \{send^{\langle A,A,O\mapsto E_{\text{event}}\rangle}, Receive^{\langle A,O\mapsto E_{\text{event}}\rangle}, Accept^{\langle A,O\mapsto E_{\text{event}}\rangle}, Expire^{\langle O\mapsto E_{\text{event}}\rangle}\}$
  - $E_{\text{fluent}} = \{effective^{\langle O\mapsto E_{\text{fluent}}\rangle}, \ldots\}$
- the following sets of predicates:
  - $\mathbf{P}_{time} = \leq^{\langle T,T\rangle}, =^{\langle T,T\rangle}, \ldots$
  - $\mathbf{P}_{\infty} = \leq_{\$}^{\langle P,P\rangle}$ (that denotes the $\leq$ relation between prices).
- and a set of variable symbols for each sort.

The statements in the example may be formalized as follows:

1. "On 1/4/04, SmallCo sent an offer to BigCo for selling goods $g$ for price $p$ with a 2 weeks expiration interval."
   $\text{OCCURS}(1/4/04, send(sco, bco, offer(sco, bco, sale(g,p), 2w)))$

2. "BigCo received the offer three days later and it has been effective since then."
   $\text{OCCURS}(1/4/04 + 3d, Receive(bco, offer(sco, bco, sale(g,p), 2w))) \wedge$
   $\text{HOLDS}(1/4/04 + 3d, now, effective(offer(sco, bco, sale(g,p), 2w)))$

3. "A properly formalized offer becomes effective when is received by the offered ..."
   $\forall\, t_1 : T_{\text{point}}, x_a, y_a : A,$
   $[\, Correct\_form(offer(x_a, y_a, \_, \_)) \wedge \text{OCCURS}(t_1, Receive(y_a, offer(x_a, y_a, \_, \_))) \rightarrow$
   $\exists t_2 : T_{\text{point}}\, [\, \text{HOLDS}(t_1, t_2, effective(offer(x_a, y_a, \_, \_)))]$
   $]$

4. "... (an effective offer) continues to be so until it is accepted by the offered or the offer expires (as indicated by its expiration interval)."
   $\forall\, t_1, t_2 : T_{\text{point}}, x_a, y_a : A, x_o : O, ts : T_{\text{span}}$
   $[\, \text{HOLDS}(t_1, t_2, effective(offer(x_a, y_a, x_o, ts))) \rightarrow$
   $\exists t_3 : T_{\text{point}}\, [t_1 < t_3 \leq t_1 + ts \wedge \text{OCCURS}(t_3, Accept(y_a, x_o))] \vee$
   $(t_2 = t_1 + ts \wedge \text{OCCURS}(t_2, Expire(offer(x_a, y_a, x_o, ts))))$
   $]$

5. "Anybody who makes an offer is committed to the offer as long as the offer is effective."
   $\forall\, t_1, t_2 : T_{\text{point}}, x_a : A, x_o : O$
   $[\, \text{HOLDS}(t_1, t_2, effective(offer(x_a, \_, \_, \_))) \rightarrow$
   $\text{HOLDS}(t_1, t_2, Committed(x_a, offer(x_a, \_, \_, \_)))\, ]$

6. "Anybody who receives an offer is obliged to send a confirmation to the offerer within two days."
$\forall\, t : T_{\text{point}}, x_a, y_a : A, x_o : O,$
$[\text{OCCURS}(t, \mathit{Receive}(y_a, \mathit{offer}(x_a, y_a, x_o, \_))) \rightarrow$
$\text{HOLDS}(t, t + 2d, \mathit{Obliged}(y_a, \mathit{send}(y_a, x_a, \mathit{conf}(\mathit{offer}(x_a, y_a, x_o, \_)))))]$

The last formula is legal but the resulting formalization is somewhat obscure. It expresses that the obligation holds between $t$ and $t+2d$. However, it fails to express that the obligation is to send the confirmation between $t$ and $t+2d$. The more general statements are formalized as follows:

- Time axioms are expressed as usual:
$\forall\, t_1, t_2, t_3 : T_{\text{point}}\ [\, t_1 \leq t_2 \wedge t_2 \leq t_3 \rightarrow t_1 \leq t_3\,]$

- Temporal incidence axioms become more compact since we can quantify over all the instances of a given entity (e.g. all fluents) independently of their particular meaning (and it is no longer necessary to have an "axiom schema"). For instance the "homogeneity of fluent holding" is stated as:
$\forall\, t_1, t_2, t_3, t_4 : T_{\text{point}}, f : E_{\text{fluent}}$
$[\, \text{HOLDS}(t_1, t_2, f) \wedge t_1 \leq t_3 \leq t_4 \leq t_2 \wedge t_1 \neq t_4 \rightarrow \text{HOLDS}(t_3, t_4, f)\,]$

- "It is necessary for an offer to be properly written to be effective".
$\forall\, t, t' : T_{\text{point}}, x_o : O\ [\text{HOLDS}(\mathit{effective}(t, t', x_o)) \rightarrow \mathit{Correct\_form}(x_o)]$

- "Whenever an offer is effective it *causes* the agent who made the offer to be committed to it for as long as the offer is effective."
$\forall\, t_1, t_2 : T_{\text{point}}, x_a, y_a : A, x_o : O, ts : T_{\text{span}}$
$[\, \text{CAUSE}(\mathit{effective}(t_1, t_2, \mathit{offer}(x_a, y_a, x_o, ts)),$
$\mathit{Committed}(t_1, t_2, x_a, \mathit{offer}(x_a, y_a, x_o, ts)))\,]$

- "Whenever a cause occurs its effects hold."
$\forall\, e : E_{\text{event}}, f : E_{\text{fluent}}\ [\, \text{OCCURS}(e) \wedge \text{CAUSE}(e, f) \rightarrow \text{HOLDS}(f)\,]$

- "Causes precede their effects."
$\forall\, e : E_{\text{event}}, f : E_{\text{fluent}}$
$[\, \text{CAUSE}(e, f) \rightarrow (\text{OCCURS}(e) \rightarrow \text{HOLDS}(f) \wedge t_{(e)} < \texttt{begin}(f))\,]$

### 5.5.2 Full Temporal Reified Logic

In the previous section we have restricted ourselves to reification of atomic propositions. However, as the following examples illustrate, it may be necessary to reify also non-atomic propositions (as first discussed in [McDermott, 1982; Allen, 1984]):

1. "The offer was sent between $t_1$ and $t_2$ but is not effective from $t_1$ to $t_2$".
$\text{HOLDS}(t_1, t_2, \mathit{sent}(o_1) \wedge \neg \mathit{effective}(o_1))$

2. "From $t_1$ to $t_2$ all offers offered by agent $a_1$ have been frozen."
$\text{HOLDS}(t_1, t_2, \forall x_a : A, x_o : O, ts : T_{\text{span}}\ [\mathit{frozen}(\mathit{offer}(a_1, y_a, x_o, ts))]$

3. "As of 1/may/04, when an offer is sent, the offerer will have to pay a tax within the next 3 days."
   $\text{HOLDS}(1/may/04, +\infty,$
   $\forall x_a, y_a : A, x_o : O, ts : T_{\text{span}}$
   $[send(x_a, y_a, \mathit{offer}(x_a, y_a, x_o, ts)) \rightarrow Obligation(pay(x_a, tax), t?, t?)]$

In order to deal with such examples, we need to expand our language and include a function symbol for each logical connective or quantifier. Thus, as example 1 above shows, the language has to contain a function symbol $\wedge$ which takes as input two fluents and returns another fluent. *Reichgelt's reified temporal logic* illustrates this approach. It provides a full formalization of Shoham's insight that reified temporal logic can be regarded as a formalization of modal temporal logic. Reichgelt's reified temporal logic therefore takes as its starting point modal temporal logic, and formaluates the semantics for such logics in a first-order language. The resulting system, however, becomes rather baroque as it needs to include terms to refer both to the semantic entities that are introduced in modal temporal logic, and terms to refer to the expressions in the modal temporal logic. Thus, a full reified logic would need to codify such statements as "Fp(a) is true at time t if and only if there is a time t' later than t at which the individual denoted by a is an element of the set denoted by P" and this requires the full reified logic to have expressions to refer to times ("t, t' "), expressions to refer to individuals ("the individual denoted by a") and denotations of predicates ("the set denoted by P"), as well as expressions to refer to expressions in the modal temporal logic that is used as its starting point ("the expression a"). The semantics for a full reified logic becomes correspondingly complex, as it needs to include normal individuals and points in time, as well as entities corresponding to the linguistic entities that make up the underlying modal temporal logic. Reichgelt's logic is therefore more of academic interest, rather than of any practical use. However, the system shows that one can indeed use Shoham's proposal to regard reified temporal logics as a formalization of the semantics of modal temporal logic in a complicated, sorted but classical first-order logic.

### 5.5.3 Advantages and Shortcomings of Temporal Reified method

As illustrated by the example, the temporal reification method provides a fairly natural and efficient notation and an expressive power clearly superior to the methods of temporal arguments as it allows one quantify over temporal relations satisfactorily.

However temporal reified approaches have been criticized on two different direction. On the one hand, because the **ontologies** they commit one to. In the example $\text{OCCURS}(1/4/04+3d, \mathit{Receive}(bco, \mathit{offer}(sco, bco, \mathit{sale}(g,p), 2w))) \wedge$
$\text{HOLDS}(1/4/04+3d, now, \mathit{effective}(\mathit{offer}(sco, bco, \mathit{sale}(g,p), 2w)))$ we observe that, in both cases, the non-time arguments to the temporal incidence predicate stand for a type of event or fluent, respectively. There are two objections against the introduction of event and state types. The first is ontological. Thus, taking his lead from [Davidson, 1967], and following a long tradition in ontology, A. Galton [Galton, 1991] argues that a logic which forces one to reify event *tokens* instead of event *types*, would be preferable on ontological grounds. Using Occam's razor, Galton argues that one should not multiply the entities in one's ontology without need, and that, unless one is a die-hard Platonist, one would prefer an ontology based on particulars rather than universals. A second argument against the introduction of types is that the resulting logic may have expressiveness shortcomings. Haugh [Haugh, 1987] talks

about the "individuation and counting of the events of a particular type". One cannot, for instance, refer to the set of multiple effects originated by a single event causing them. Also, one cannot quantify over causes and the related set of the effects each produces in order to assert general constraints between them.

On the other hand, temporal reification has been criticized as an unnecessary technical complication, specially in the case that it is not defined as a standard many-sorted logic and we have to develop a new model theory and a complete proof theory. Some researchers look at the *temporal token arguments* method a the ideal alternative since it avoid both criticisms and seem to retain the expressiveness adavantages, in particular in quatifying over predicates as shown in the *TTA* section.

## 5.6 Temporal Token Reification

The temporal token reification approach is motivated by the attempt of achieving the expressiveness advantages of temporal reification and the ontological and technical advantages of temporal tokens shown by the temporal token arguments approach which avoids having to reify temporal types.

The primary intuition behind *Temporal Token Reification* (*TTR* ) is that one reifies temporal tokens rather than temporal types. However, rather than making names for event tokens an additional argument to a predicate (like in the temporal token arguments approach), it proposes to introduce "meaningful" names for temporal tokens. This allows one to talk and quantify about "parts of a token" as well as over all tokens and thus express express general temporal properties.

### 5.6.1 Definition

The logical language of *TTR* is a many-sorted FOL with the same sorts as *TTA* : $T_1, \ldots, T_{n_t}$, one for each time set, a number of non-time sorts $U_1, \ldots, U_n$ and one token sort $E_1, \ldots, E_{n_e}$ for each temporal entity.

**Syntax** The vocabulary is defined accordingly:

- *Function symbols:* In addition to the time and atemporal function symbols of *TTA* , we have a set of additional a $m + n$-place function symbol for each $n$-place temporal relation, where the first $m$ arguments are of a time sort and the last $n$ arguments of some non-time or token sort. The output is an entity of some type $E_i$.

  We also have the usual *time-token function* symbols, whose input argument is of sort $E_i$ and whose output argument is of sort $T_j$. For instance, `begin` denotes the starting point of a temporal token and their definition is straightforward. Thus

  $$\texttt{begin}(f(\ldots, t, t')) = t$$

  where $f(\ldots, t, t')$ is a term referring to a temporal token.

Finally, the language contains the 1-place function symbol TYPE. It takes as argument the name of a temporal token and returns a function from pairs of points in time into the set of event or state tokens respectively. Hence,

$$\text{TYPE}(f(\ldots, t, t'))$$

is basically syntactic sugar for

$$\lambda x \lambda y f(\ldots, x, y)$$

- *Predicate symbols:* As *TTA* , *TTR* makes *TIPs* 1-place. It contains one TIP each $E_i$ with its only argument being the name for an temporal entity. For instance, the predicates HOLDS or OCCURS simply state that a fluent token indeed holds, or that an event token indeed occurs.

**Semantics** The semantics of the *TTR* is relatively straightforward as well and *TTR* function and predicate symbols are mapped onto the appropriate functions and relations respecting the signature of the symbol.

## 5.6.2 Formalizing the Example

To formalize the example, we use the same sorts and the vocabulary as in the temporal reification example with the following additions:

- $\mathbf{F}_{time} = \{\texttt{end}^{\langle E_{\text{fluent}}, T_{\text{point}}, T_{\text{point}} \mapsto T_{\text{point}} \rangle}, \texttt{begin}^{\langle E_{\text{fluent}}, T_{\text{point}}, T_{\text{point}} \mapsto T_{\text{point}} \rangle}\}$ where $f(\ldots, t, t')$ is a term referring to a fluent-token.
- $\mathbf{F}_t = \mathbf{F}_{event} \cup \mathbf{F}_{fluent}$
  - $\mathbf{F}_{event} = \{\mathit{send}^{\langle T_{\text{point}}, A, A, O \mapsto E_{\text{event}} \rangle}, \mathit{Receive}^{\langle T_{\text{point}}, A, O \mapsto E_{\text{event}} \rangle}, \mathit{Accept}^{\langle T_{\text{point}}, A, O \mapsto E_{\text{event}} \rangle}, \mathit{Expire}^{\langle T_{\text{point}}, O \mapsto E_{\text{event}} \rangle}\}$
  - $\mathbf{F}_{fluent} = \{\mathit{effective}^{\langle O, T_{\text{point}}, T_{\text{point}} \mapsto E_{\text{fluent}} \rangle}\}$
- $\mathbf{P}_t = \emptyset$
- $\mathbf{P}_\infty = \{\leq_{\$}^{\langle P, P \rangle}\}$ (that denotes the $\leq$ relation between prices).
- and a set of variable symbols for each sort.

The statements in the example can be formalized as follows:

1. "On 1/4/04, SmallCo sent an offer for selling goods $g$ to BigCo for price $p$ with a 2 weeks expiration interval."
   $\text{OCCURS}(\mathit{send}(1/4/04, \mathit{sco}, \mathit{bco}, \mathit{offer}(\mathit{sco}, \mathit{bco}, \mathit{sale}(g, p), 2w)))$
2. "BigCo received the offer three days later and it has been effective since then."
   $\text{OCCURS}(\mathit{Receive}(1/4/04 + 3d, \mathit{bco}, \mathit{offer}(\mathit{sco}, \mathit{bco}, \mathit{sale}(g, p), 2w))) \wedge$
   $\text{HOLDS}(\mathit{effective}(1/4/04 + 3d, \texttt{now}, \mathit{offer}(\mathit{sco}, \mathit{bco}, \mathit{sale}(g, p), 2w)))$

3. "A properly formalized offer becomes effective when is received by the offered..."
$\forall\, t_1 : T_{\text{point}}, x_a, y_a : A, x_o : O, ts : T_{\text{span}}$
$[\ \mathit{Correct\_form}(\mathit{offer}(x_a, y_a, x_o, ts)) \wedge$
$\text{OCCURS}(\mathit{Receive}(t_1, y_a, \mathit{offer}(x_a, y_a, x_o, ts))) \rightarrow$
$\exists t_2\ [\text{HOLDS}(\mathit{effective}(t_1, t_2, \mathit{offer}(x_a, y_a, x_o, ts))) \wedge t_1 \leq t_2]\ ]$

4. "...(an effective offer) continues to be so until it is accepted by the offered or the offer expires (as indicated by its expiration interval)."
$\forall\, t_1, t_2 : T_{\text{point}}, x_a, y_a : A, x_o : O, ts : T_{\text{span}}$
$[\ \text{HOLDS}(\mathit{effective}(t_1, t_2, \mathit{offer}(x_a, y_a, x_o, ts))) \wedge t_1 \leq t_2 \rightarrow$
$\exists t_3 : T_{\text{point}}[\mathit{Accept}(t_3, y_a, \mathit{offer}(x_a, y_a, x_o, ts)) \wedge t_1 < t_3 \leq t_1 + ts]\ \vee$
$(t_2 = t_1 + ts \wedge \text{OCCURS}(\mathit{Expire}(t2, \mathit{offer}(x_a, y_a, x_o, ts))))\ ]$

5. "Anybody who makes an offer is committed to the offer as long as the offer is effective."
$\forall\, t_1, t_2 : T_{\text{point}}, x_a : A$
$[\text{HOLDS}(\mathit{effective}(t_1, t_2, \mathit{offer}(x_a, \_, \_, \_))) \rightarrow$
$\text{OCCURS}(\mathit{Committed}(t_1, t_2, x_a, \mathit{offer}(x_a, \_, \_, \_)))]$

6. "Anybody who receives an offer is obliged to send a confirmation to the offerer within two days."
$\forall\, t_1 : T_{\text{point}}, x_a : A, x_o : O,$
$[\text{OCCURS}(\mathit{Receive}(t_1, y_a, \mathit{offer}(x_a, y_a, x_o, \_))) \rightarrow$
$\text{HOLDS}(\mathit{Obliged}(t, t + 2d, y_a, \mathit{send}(y_a, \mathit{conf}(\mathit{offer}(x_a, y_a, x_o, \_)))))$
$]$

The additional statements are formalized as follows:

- Time axioms: "The ordering between instants is transitive":
$\forall\, t_1, t_2, t_3 : T_{\text{point}}\ [\ t_1 \leq t_2 \wedge t_2 \leq t_3 \rightarrow t_1 \leq t_3\ ]$

- Temporal Incidence axioms such as "Fluents hold homogeneously":
$\forall f : E_{\text{fluent}}, t_1, t_2, t_3, t_4 : T_{\text{point}}$
$[\ \text{HOLDS}(\text{TYPE}(f)(< t_1, t_2 >)) \wedge t_1 \leq t_3 \leq t_4 \leq t_2 \wedge t_1 \neq t_4 \rightarrow$
$\text{HOLDS}(\text{TYPE}(f)(< t_3, t_4 >))\ ]$

- "It is necessary for an offer to be properly written to be effective".
$\forall\, t_1, t_2 : T_{\text{point}}, x_o : O$
$[\text{HOLDS}(\mathit{effective}(t_1, t_2, x_o)) \rightarrow \mathit{Correct\_form}(x_o)]$

- "Whenever an offer is effective it *causes* the agent who made the offer to be committed to it for as long as the offer is effective."
$\forall\, t_1, t_2 : T_{\text{point}}, x_a, y_a : A, x_o : O, ts : T_{\text{span}}$
$[\text{CAUSE}(\mathit{effective}(t_1, t_2, \mathit{offer}(x_a, y_a, x_o, ts)),$
$\mathit{Committed}(t_1, t_2, x_a, \mathit{offer}(x_a, y_a, x_o, ts)))]$

- "Whenever a cause occurs its effects hold."
$\forall\, e : E_{\text{event}}, f : E_{\text{fluent}}$
$[\ \text{OCCURS}(e) \wedge \text{CAUSE}(e, f) \rightarrow \text{HOLDS}(f)\ ]$

- "Causes precede their effects."
  $\forall\, e : E_{\text{event}}, f : E_{\text{fluent}}$
  $[\, \text{CAUSE}(e, f) \rightarrow (\text{OCCURS}(e) \rightarrow \text{HOLDS}(f) \wedge t_{(e)} \leq \texttt{begin}(f))\, ]$

## 5.7 Concluding Remarks

In this chapter we have identified the relevant issues around the temporal qualification method which is central in the definition of a temporal reasoning system in AI. We have described the most relevant temporal qualification methods, illustrated them with a rich example and analysed advantages and shortcomings with respect to a number of representational and reasoning efficiency criteria. The various methods are schematically presented in Figure 5.1.

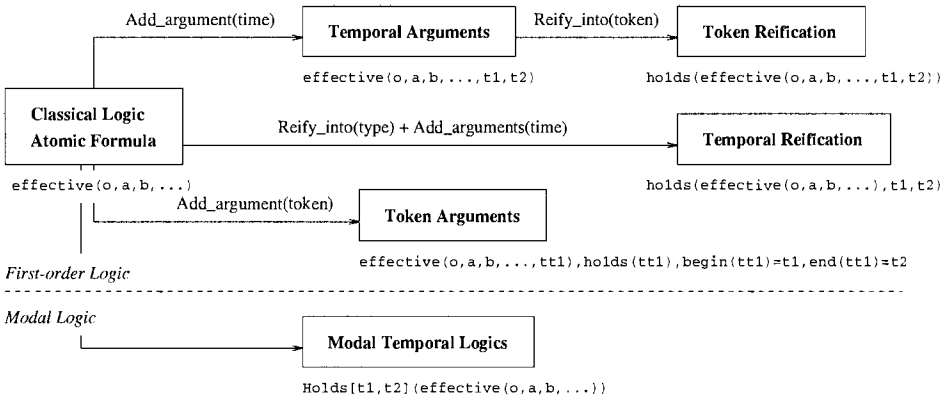



Figure 5.1: Temporal qualification methods in AI.

Temporal arguments is the classical and most straightforward method that turns out to be more expressive than has traditionally been recognized. It is enough for many applications except for those where one needs to represent nested temporal references or one needs to quantify over temporal propositions. In fact, the subsequent methods are a response to this limitation in a more or less sophisticated manner. *Temporal Token Arguments*, while using a language very similar to that of the method of temporal arguments, moves to a token-based ontology and introduces names for temporal token in the language. This provides a good deal of represenation flexibility. The other two approaches are based on reification. Reification allows one to quantify over temporal entities, resulting in significantly increased expressiveness. The increased expressiveness allows one to express statements like "receiving an offer causes to be obliged to send a confirmation" or " causes never preced their effects" which is not possible in the temporal argumentn method.

Technically, the temporal reification methods are not necessarily complex. However, the system becomes highly complex if one insists on reification of non-atomic formulas, as shown in [Reichgelt, 1987; Reichgelt, 1989]. However, in many cases, this is not necessary: some temporal reified logics can be defined as a many-sorted logic with the appropriate time and temporal incidence axiomatizations. However, it is important to be aware that these axioms can be a source of high inefficiency for the theorem prover.

# Constraint Manipulation

This Page Intentionally Left Blank

Handbook of Temporal Reasoning in Artificial Intelligence
Edited by M. Fisher, D. Gabbay and L. Vila


Chapter 6

# Computational Complexity of Temporal Constraint Problems

Thomas Drakengren & Peter Jonsson

This chapter surveys results on the computational complexity of temporal constraint reasoning. The focus is on the satisfiability problem, but also the problem of entailed relations is treated. More precisely, results for formalisms based upon relating time points and/or intervals with qualitative and/or metric constraints are reviewed. The main purpose of the chapter is to distinguish between tractable and NP-complete cases.

## 6.1 Introduction

The purpose of this chapter is to survey results on the computational complexity of temporal constraint reasoning. To keep the presentation reasonably short, we make a few assumptions:

1. We assume that time is linear, dense and unbounded. This implies that, for instance, we do not consider branching, discrete or finite time structures.

2. We focus on the satisfiability problem, that is, the problem of deciding whether a set of temporal formulae has a model or not. However, we also treat the problem of entailed relations, in the context of Allen's algebra.

3. Initially, we follow standard mathematical praxis and allow temporal variables to be unrelated, *i.e.*, we allow problems where variables may not be explicitly tied by any constraint. In the final section, we study some cases where this assumption is dropped.

Our main purpose is to distinguish between problems that are solvable in polynomial time and problems that are not*. As a consequence, we will not necessarily present the most efficient algorithms for the problems under consideration. We will instead emphasize simplicity and generality, which means that we will use standard mathematical tools whenever possible.

This chapter begins, in Section 6.2, with an in-depth treatment of *disjunctive linear relations* (DLR), here serving two purposes:

*Assuming $P \neq NP$, of course.

1. DLRs will be used as a unifying formalism for temporal constraint reasoning, since it subsumes most approaches that have been proposed in the literature.

2. DLRs will be used extensively for dealing with metric time.

We continue in Section 6.3 by introducing Allen's interval algebra, and presenting all tractable subclasses of that algebra. We also provide some results on the complexity of computing entailed relations.

Section 6.4 is concerned with *point-interval* relations, in which time points are related to intervals. A complete enumeration of all maximal tractable subclasses is given, together with algorithms for solving the corresponding problems.

In Section 6.5, the problem of handling metric time is studied. Extensions to Horn DLRs are considered, as well as methods based on arc and path consistency.

Finally, Section 6.6 contains some "non-standard" techniques in temporal constraint reasoning. We consider, for instance, temporal reasoning involving durations, and the implications of not allowing unrelated variables.

## 6.2 Disjunctive Linear Relations

### 6.2.1 Definitions

**Definition 6.2.1.** Let $X = \{x_1, \ldots, x_n\}$ be a set of real-valued variables, and $\alpha, \beta$ linear polynomials (polynomials of degree one) over $X$, with rational coefficients. A *linear relation* over $X$ is a mathematical expression of the form $\alpha R \beta$, where $R \in \{<, \leq, =, \neq, \geq, >\}$.

A *disjunctive linear relation* (DLR) over $X$ is a disjunction of a nonempty finite set of linear relations. A DLR is said to be *Horn* if at most one of its disjuncts is not of the form $\alpha \neq \beta$.

The *satisfiability* problem for a finite set $D$ of DLRs, denoted DLRSAT($D$), is the problem of checking whether there exists an assignment $M$ of variables in $X$ to real numbers, such that all DLRs in $D$ are satisfied in $M$. Such an $M$ is said to be a *model* of $D$. The satisfiability problem for finite sets $H$ of Horn DLRs is denoted HORNDLRSAT($H$). □

**Example 6.2.1.**

$$x + 2y \leq 3z + 42.3$$

*is a linear relation,*

$$(x + 2y \leq 3z + 42.3) \vee (x > \frac{3}{12})$$

*is a disjunctive linear relation, and*

$$(x + 2y \leq 3z + 42.3) \vee (x \neq \frac{3}{12})$$

*is a Horn disjunctive linear relation.*

In principle, the framework of DLRs makes it unnecessary to distinguish between qualitative and metric information. Nevertheless, when it comes to identifying tractable subclasses, the distinction is still convenient.

### 6.2.2 Algorithms and Complexity

In this section, we present the two main results for computing with DLRs. We also provide a polynomial-time algorithm for checking the satisfiability of Horn DLRs.

**Proposition 6.2.2.** The problem DLRSAT is NP-complete.

*Proof.* The satisfiability problem for propositional logic, which is known to be NP-complete, can easily be coded as DLRs. For the details, see [Jonsson and Bäckström, 1998].

**Proposition 6.2.3.** HORNDLRSAT is solvable in polynomial time.

*Proof.* See [Jonsson and Bäckström, 1998] or [Koubarakis, 1996].

We will present a polynomial-time algorithm for HORNDLRSAT in Algorithm 6.2.9. In order to understand it, some auxiliary concepts are needed.

**Definition 6.2.4.** A linear relation $\alpha R\beta$ is said to be *convex* if $R$ is not the relation $\neq$.

Let $\gamma$ be a DLR. We let $\mathcal{C}(\gamma)$ denote the DLR where all nonconvex relations in $\gamma$ have been removed, and $\mathcal{NC}(\gamma)$ the DLR where all convex relations in $\gamma$ have been removed.

We say that $\gamma$ is *convex* if $\mathcal{NC}(\gamma) = \emptyset$, and that $\gamma$ is *disequational* if $\mathcal{C}(\gamma) = \emptyset$. If $\gamma$ is convex or disequational we say that $\gamma$ is *homogeneous*, and otherwise it is said to be *heterogeneous*. We extend these definitions to sets of relations in the obvious way; for example, if $\Gamma$ is a set of DLRs and all $\gamma \in \Gamma$ are Horn, then $\Gamma$ is Horn. □

The algorithm for deciding satisfiability of Horn DLRs is based on linear programming techniques, so we begin by providing the basic facts for that. The linear programming problem is defined as follows.

**Definition 6.2.5.** Let $A$ be an arbitrary $m \times n$ matrix of rational numbers and let $x = (x_1, \ldots, x_n)$ be an $n$-vector of variables over the real numbers. Then an instance of the *linear programming* (LP) problem is defined by $\{\min\ c^{\mathrm{T}}x$ subject to $Ax \leq b\}$, where $b$ is an $m$-vector of rational numbers, and $c$ an $n$-vector of rational numbers. The computational problem is as follows:

1. Find an assignment to the variables $x_1, \ldots, x_n$ such that the condition $Ax \leq b$ holds, and $c^{\mathrm{T}}x$ is minimal subject to these conditions, or
2. Report that there is no such assignment, or
3. Report that there is no lower bound for $c^{\mathrm{T}}x$ under the conditions.

□

Analogously, we can define an LP problem where the objective is to maximize $c^{\mathrm{T}}x$ under the condition $Ax \leq b$. We have the following theorem.

**Theorem 6.2.6.** The linear programming problem is solvable in polynomial time.

*Proof.* Several polynomial-time algorithms have been developed for solving LP. Well-known examples are the algorithms by [Khachiyan, 1979] and [Karmarkar, 1984].

**Definition 6.2.7.** Let $A$ be a satisfiable set of DLRs and let $\gamma$ be a DLR. We say that $\gamma$ *blocks* $A$ if $A \cup \{d\}$ is unsatisfiable for any $d \in \mathcal{NC}(\gamma)$. □

**Lemma 6.2.8.** Let $A$ be an arbitrary $m \times n$ matrix of rational numbers, $b$ an $m$-vector of rational numbers and $x = (x_1, \ldots, x_n)$ an $n$-vector of variables over the real numbers. Let $\alpha$ be a linear polynomial over $x_1, \ldots, x_n$ and $c$ a rational number. Deciding whether the system $S = \{Ax \leq b, \alpha \neq c\}$ is satisfiable or not is a polynomial-time problem.

*Proof.* Consider the following instances of LP:

$$\text{LP1} = \{\min\ \alpha \text{ subject to } Ax \leq b\}$$

$$\text{LP2} = \{\max\ \alpha \text{ subject to } Ax \leq b\}$$

If either LP1 or LP2 has no solutions, then $S$ is not satisfiable. If both LP1 and LP2 yield the same optimal value $c$, then $S$ is not satisfiable, since every solution $y$ to LP1 and LP2 satisfies $\alpha(y) = c$. Otherwise $S$ is obviously satisfiable. Since we can solve the LP problem in polynomial time by Theorem 6.2.6, the result follows.

**Algorithm 6.2.9.** (**Alg-**HORNDLRSAT($\Gamma$))

**input** Set $\Gamma$ of DLRs

```
1  A ← {γ | γ ∈ Γ is convex}
2  if A is not satisfiable then
3      reject
4  if ∃β ∈ Γ that blocks A then
5      if β is disequational then
6          reject
7      else
8          Alg-HORNDLRSAT((Γ − {β}) ∪ C(β))
9  accept
```

□

**Theorem 6.2.10.** Algorithm 6.2.9 correctly solves HORNDLRSAT in polynomial time.

*Proof.* The test in line 2 can be performed in polynomial time using linear programming, and the test in line 4 can be performed in polynomial time by Lemma 6.2.8. Thus, the algorithm runs in polynomial time. The correctness proof can be found in [Jonsson and Bäckström, 1998].

### 6.2.3 Subsumed Formalisms

Several formalisms can easily be expressed as DLRs, but more importantly, most proposed tractable temporal formalisms are subsumed by the Horn DLR formalism.

For the following definitions, let $x, y$ be real-valued variables, $c, d$ rational numbers, and $\mathcal{A}$ Allen's algebra [Allen, 1983] (see Section 6.3 for its definition). It is trivial to see that the

DLR language subsumes Allen's algebra. Furthermore, it subsumes the universal temporal language by Kautz and Ladkin, defined as follows.

**Definition 6.2.11. (Universal temporal language)** The *universal temporal language* [Kautz and Ladkin, 1991] consists of $\mathcal{A}$, augmented with formulae of the form $-cr_1(x - y)r_2d$, where $r_1, r_2 \in \{<, \leq\}$, and $x, y$ are endpoints of intervals. □

DLRs also subsume the *qualitative algebra* (QA) by [Meiri, 1996]. In QA, a qualitative constraint between two objects $O_i$ and $O_j$ (each may be a point or an interval), is a disjunction of the form

$$(O_i r_1 O_j) \vee \ldots \vee (O_i r_k O_j)$$

where each one of the $r_i's$ is a *basic relation* that may exist between two objects. There are three types of basic relations.

1. *Interval-interval* relations that can hold between a pair of intervals. These relations correspond to Allen's algebra.
2. *Point-point* relations that can hold between a pair of points. These relations correspond to the point algebra [Vilain, 1982].
3. *Point-interval* and *interval-point* relations that can hold between a point and an interval and vice-versa. These relations were introduced by [Vilain, 1982].

Obviously, DLRs subsume QA. Meiri also considers QA extended with metric constraints of the following two forms, $x_1, \ldots, x_n$ being time points or endpoints of intervals.

1. $(c_1 \leq x_1 \leq d_1) \vee \ldots \vee (c_1 \leq x_n \leq d_1)$;
2. $(c_1 \leq x_n - x_1 \leq d_1) \vee \ldots \vee (c_1 \leq x_n - x_{n-1} \leq d_1)$.

Also this extension to QA can easily be expressed as DLRs. It has been shown that the satisfiability problems for all of these formalisms are NP-complete [Vilain *et al.*, 1990; Kautz and Ladkin, 1991; Meiri, 1996]. In retrospect, the different restrictions imposed on these formalisms seem quite artificial when compared to DLRs, especially since they do not reduce the computational complexity of the problem.

Next, we review some of the formalisms that are subsumed by Horn DLRs.

**Definition 6.2.12. (Point algebra formulae, pointisable algebra)** A *point algebra formula* [Vilain, 1982] is an expression $xRy$, where $x$ and $y$ are variables, and $R$ is one of the relations $<, \leq, =, \neq, \geq$ and $>$.

The *pointisable algebra* [van Beek and Cohen, 1990] is the set of relations in $\mathcal{A}$ which can be expressed as point algebra formulae. □

We denote satisfiability problem for point algebra formulae by $\text{PASAT}(H)$, for a set $H$ of point algebra formulae.

**Definition 6.2.13. (Continuous endpoint formula, continuous endpoint algebra)** A *continuous endpoint formula* [Vilain *et al.*, 1990] is a point algebra formula $xRy$ where $R$ is not the relation $\neq$.

The *continuous endpoint algebra* [Vilain *et al.*, 1990] is the set of relations in $\mathcal{A}$ which can be expressed as continuous endpoint formulae. □

The following formalism subsumes those of the previous two definitions.

**Definition 6.2.14. (ORD-Horn algebra)** An *ORD clause* is a disjunction of relations of the form $xRy$, where $R \in \{\leq, =, \neq\}$. The *ORD-Horn* subclass $\mathcal{H}$ [Nebel and Bürckert, 1995] is the set of relations in $\mathcal{A}$ that can be written as ORD clauses containing only disjunctions, with at most one relation of the form $x = y$ or $x \leq y$, and an arbitrary number of relations of the form $x \neq y$. □

**Definition 6.2.15. (Koubarakis formula)** A *Koubarakis formula* [1992] is a formula of one of the following forms:

1. $(x - y)Rc$
2. $xRc$
3. A disjunction of formulae of the form $(x - y) \neq c$ or $x \neq c$,

where $R \in \{\leq, \geq, \neq\}$. □

**Definition 6.2.16. (Simple temporal constraint)** A *simple temporal constraint* [Dechter *et al.*, 1991] is a formula on the form $c \leq (x - y) \leq d$. □

**Definition 6.2.17. (Simple metric constraint)** A *simple metric constraint* [Kautz and Ladkin, 1991] is a formula on the form $-cR_1(x - y)R_2d$ where $R_1, R_2 \in \{<, \leq\}$. □

**Definition 6.2.18. (PA/single-interval formula)** A *PA/single-interval* formula [Meiri, 1996] is a formula on one of the following forms:

1. $cR_1(x - y)R_2\, d$, where $R_1, R_2 \in \{<, \leq\}$
2. $xRy$ where $R \in \{<, \leq, =, \neq, \geq, >\}$

□

**Definition 6.2.19. (TG-II formula)** A *TG-II* formula [Gerevini *et al.*, 1993] is a formula on one of the following forms:

1. $c \leq x \leq d$,
2. $c \leq x - y \leq d$
3. $xRy$ where $R \in \{<, \leq, =, \neq, \geq, >\}$

□

Besides these classes, other temporal classes that can be expressed as Horn DLRs have been identified by different authors. Examples include the approach by [Barber, 1993], the subclass $\mathcal{V}^{23}$ for relating points and intervals [Jonsson *et al.*, 1999] (see Section 6.4), and the temporal part of TMM by [Dean and Boddy, 1988].

Not all known tractable classes can be modeled as Horn DLRs (in any obvious way*), however. Examples of this are [Golumbic and Shamir, 1993] and Drakengren and Jonsson [1997a; 1997b].

*Linear programming is a P-complete problem, so in principle, all polynomial-time computable problems can be transformed into Horn DLRs.

| Basic relation | | Example | Endpoints |
|---|---|---|---|
| $x$ before $y$ | $\prec$ | xxx<br>yyy | $x^+ < y^-$ |
| $y$ after $x$ | $\succ$ | | |
| $x$ meets $y$ | m | xxxx<br>yyyy | $x^+ = y^-$ |
| $y$ met-by $x$ | $\mathsf{m}^{-1}$ | | |
| $x$ overlaps $y$ | o | xxxx<br>yyyy | $x^- < y^- < x^+$,<br>$x^+ < y^+$ |
| $y$ overl.-by $x$ | $\mathsf{o}^{-1}$ | | |
| $x$ during $y$ | d | xxx<br>yyyyyyy | $x^- > y^-$,<br>$x^+ < y^+$ |
| $y$ includes $x$ | $\mathsf{d}^{-1}$ | | |
| $x$ starts $y$ | s | xxx<br>yyyyyyy | $x^- = y^-$,<br>$x^+ < y^+$ |
| $y$ started by $x$ | $\mathsf{s}^{-1}$ | | |
| $x$ finishes $y$ | f | xxx<br>yyyyyyy | $x^+ = y^+$,<br>$x^- > y^-$ |
| $y$ finished by $x$ | $\mathsf{f}^{-1}$ | | |
| $x$ equals $y$ | $\equiv$ | xxxx<br>yyyy | $x^- = y^-$,<br>$x^+ = y^+$ |

Table 6.1: The thirteen basic relations. The endpoint relations $x^- < x^+$ and $y^- < y^+$ that are valid for all relations have been omitted.

## 6.3 Interval-Interval Relations: Allen's Algebra

### 6.3.1 Definitions

Allen's interval algebra [Allen, 1983] is based on the notion of *relations between pairs of intervals*. An interval $x$ is represented as a tuple $\langle x^-, x^+ \rangle$ of real numbers with $x^- < x^+$, denoting the left and right endpoints of the interval, respectively, and relations between intervals are composed as disjunctions of *basic interval relations*, which are those in Table 6.1. Denote the set of basic interval relations **B**. Such disjunctions are represented as *sets* of basic relations, but using a notation such that, for example, the disjunction of the basic intervals $\prec$, m and $\mathsf{f}^{-1}$ is written $(\prec\ \mathsf{m}\ \mathsf{f}^{-1})$. Thus, we have that $(\prec\ \mathsf{f}^{-1}) \subseteq (\prec\ \mathsf{m}\ \mathsf{f}^{-1})$. The disjunction of all basic relations is written $\top$, and the empty relation is written $\bot$ (this is also used for relations between interval endpoints, denoting "always satisfiable" and "unsatisfiable", respectively). The algebra is provided with the operations of *converse, intersection and composition* on intervals, but we shall need only the converse operation explicitly. The converse operation* takes an interval relation $i$ to its converse $i^{-1}$, obtained by inverting each basic relation in $i$, *i.e.*, exchanging $x$ and $y$ in the endpoint relations shown in Table 6.1.

By the fact that there are thirteen basic relations, we get $2^{13} = 8192$ possible relations between intervals in the full algebra. We denote the set of all interval relations by $\mathcal{A}$. Subclasses of the full algebra are obtained by considering subsets of $\mathcal{A}$. There are $2^{8192} \approx 10^{2466}$ such subclasses. Classes that are closed under the operations of intersection, converse and composition are said to be *algebras*.

The problem of *satisfiability* (ISAT) of a set of interval variables with relations between them is that of deciding whether there exists an assignment of intervals on the real line for

*The notation varies for this operation. However, we believe that the standard notation for inverse relations is the best and simplest choice.

the interval variables, such that all of the relations between the intervals are satisfied. This is defined as follows.

**Definition 6.3.1.** (ISAT($\mathcal{I}$)) Let $\mathcal{I} \subseteq \mathcal{A}$ be a set of interval relations. An instance of ISAT($\mathcal{I}$) is a labelled directed graph $G = \langle V, E \rangle$, where the nodes in $V$ are interval variables and $E$ is a subset of $V \times \mathcal{I} \times V$. A labelled edge $\langle u, r, v \rangle \in E$ means that $u$ and $v$ are related by $r$.

A function $M$ taking an interval variable $v$ to its interval representation $M(v) = \langle x^-, x^+ \rangle$ with $x^-, x^+ \in \mathbb{R}$ and $x^- < x^+$, is said to be an *interpretation* of $G$.

An instance $G = \langle V, E \rangle$ is said to be *satisfiable* if there exists an interpretation $M$ such that for each $\langle u, r, v \rangle \in E$, $M(u) r M(v)$ holds, *i.e.*, the endpoint relations required by $r$ (see Table 6.1) are satisfied by the assignments of $u$ and $v$. Then $M$ is said to be a *model* of $G$.

We refer to the *size* of an instance $G$ as $|V| + |E|$. □

## 6.3.2 Complexity Results

A complete classification of the computational complexity of ISAT($X$) has been presented by Krokhin *et al.* [2003]. The classification provides no new tractable subclasses; interestingly, it turns out that all existing tractable subclasses of Allen's algebra had been published in earlier papers [Nebel and Bürckert, 1995; Drakengren and Jonsson, 1997b; Drakengren and Jonsson, 1997a]. For the complete classification, the lengthy proof uses results from a number of earlier publications, *cf.* [Krokhin *et al.*, 2001; Drakengren and Jonsson, 1998; Nebel and Bürckert, 1995].

Next, we present the main result and the tractable subclasses; after that we present the polynomial-time algorithms for the tractable subclasses.

**Theorem 6.3.2.** Let $X$ be a subset of $\mathcal{A}$. Then ISAT($X$) is tractable iff $X$ is a subset of the ORD-Horn algebra (Definition 6.2.14), or of one of the 17 subalgebras defined below. Otherwise, ISAT($X$) is NP-complete.

*Proof.* See [Krokhin *et al.*, 2003].

**Definition 6.3.3.** (**Subclasses** $A(r, b)$ [Drakengren and Jonsson, 1997b])
Let $b \in \{\mathsf{s}, \mathsf{s}^{-1}, \mathsf{f}, \mathsf{f}^{-1}\}$, and $r$ one of the relations

$$(\prec\ \mathsf{d}^{-1}\ \mathsf{o}\ \mathsf{m}\ \mathsf{s}\ \mathsf{f}^{-1})$$
$$(\prec\ \mathsf{d}^{-1}\ \mathsf{o}\ \mathsf{m}\ \mathsf{s}^{-1}\ \mathsf{f}^{-1})$$
$$(\prec\ \mathsf{d}\ \mathsf{o}\ \mathsf{m}\ \mathsf{s}\ \mathsf{f})$$
$$(\prec\ \mathsf{d}\ \mathsf{o}\ \mathsf{m}\ \mathsf{s}\ \mathsf{f}^{-1})$$

containing $b$. First define the subclasses $A_1(b)$, $A_2(r, b)$ and $A_3(r, b)$ by

$$A_1(b) = \{r' \cup (b\ b^{-1}) | r' \in \mathcal{A}\},$$

$$A_2(r, b) = \{r' \cup (b) | r' \subseteq r\}$$

and

$$A_3(r, b) = \{r' \cup (\equiv) | r' \in A_2(r, b)\} \cup \{(\equiv)\}.$$

Then set

$$B = A_1(b) \cup A_2(r, b) \cup A_3(r, b)$$

and finally define the subclass $A(r, b)$ by

$$A(r, b) = B \cup \{x^{-1} | x \in B\} \cup \{(\,)\}.$$

□

For an explicit enumeration of the sets $A(r, b)$, see [Drakengren and Jonsson, 1997b].

**Definition 6.3.4.** (**Subclass** $A_{\equiv}$ [Drakengren and Jonsson, 1997b]) Define the subclass $A_{\equiv}$ to contain every relation that contains $\equiv$, and the empty relation $(\,)$. □

**Definition 6.3.5.** (**Subclasses** $S(b)$, $E(b)$ [Drakengren and Jonsson, 1997a]) Set $r_s = (\succ$ d $o^{-1}$ $m^{-1}$ f), and $r_e = (\prec$ d o m s). Note that $r_s$ contains all basic relations $b$ such that whenever $IbJ$ for interval variables $I$, $J$, $I^- > J^-$ has to hold in any model, and symmetrically, $r_e$ is equivalent to $I^+ < J^+$ holding in any model.

First, for $b \in \{\succ, \mathsf{d}, \mathsf{o}^{-1}\}$, define $S(b)$ to be the set of relations $r$, such that either of the following holds:

$$\begin{array}{rcl} (b\ b^{-1}) & \subseteq & r \\ (b) & \subseteq & r \subseteq r_s \cup (\equiv \mathsf{s}\,\mathsf{s}^{-1}) \\ (b^{-1}) & \subseteq & r \subseteq {r_s}^{-1} \cup (\equiv \mathsf{s}\,\mathsf{s}^{-1}) \\ & & r \subseteq (\equiv \mathsf{s}\,\mathsf{s}^{-1}). \end{array}$$

Then, by switching the starting and ending points of intervals, $E(b)$ is defined, for $b \in \{\prec, \mathsf{d}, \mathsf{o}\}$, to be the set of relations $r$, such that either of the following holds:

$$\begin{array}{rcl} (b\ b^{-1}) & \subseteq & r \\ (b) & \subseteq & r \subseteq r_e \cup (\equiv \mathsf{f}\,\mathsf{f}^{-1}) \\ (b^{-1}) & \subseteq & r \subseteq {r_e}^{-1} \cup (\equiv \mathsf{f}\,\mathsf{f}^{-1}) \\ & & r \subseteq (\equiv \mathsf{f}\,\mathsf{f}^{-1}). \end{array}$$

□

**Definition 6.3.6.** (**Subclasses** $S^*$, $E^*$ [Drakengren and Jonsson, 1997a]) Let $r_s$ and $r_e$ be as in Definition 6.3.5, and define $S^*$ to be the set of relations $r$, such that either of the following holds:

$$\begin{array}{rcl} (\equiv \mathsf{f}\,\mathsf{f}^{-1}) & \subseteq & r \\ (\mathsf{f}\,\mathsf{f}^{-1}) & \subseteq & r \subseteq r_s \cup {r_s}^{-1} \\ (\equiv \mathsf{f}) & \subseteq & r \subseteq r_s \cup (\equiv \mathsf{s}\,\mathsf{s}^{-1}) \\ (\equiv \mathsf{f}^{-1}) & \subseteq & r \subseteq {r_s}^{-1} \cup (\equiv \mathsf{s}\,\mathsf{s}^{-1}) \\ (\mathsf{f}) & \subseteq & r \subseteq r_s \\ (\mathsf{f}^{-1}) & \subseteq & r \subseteq {r_s}^{-1} \\ (\equiv) & \subseteq & r \subseteq (\equiv \mathsf{s}\,\mathsf{s}^{-1}) \\ & & r = \bot \end{array}$$

Symmetrically, replacing f by s (and their inverses), $(\equiv \mathsf{s}\,\mathsf{s}^{-1})$ by $(\equiv \mathsf{f}\,\mathsf{f}^{-1})$, and $r_s$ by $r_e$, we get the subclass $E^*$. □