ELSEVIER

European Journal of Operational Research 134 (2001) 664–676

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

www.elsevier.com/locate/dsw

Theory and Methodology

Petri net-based modelling of workflow systems: An overview

Khodakaram Salimifard *, Mike Wright

Management Science Department, Lancaster University, Bailrigg, Lancaster LA1 4YX, UK

Received 19 May 1999; accepted 11 August 2000

Abstract

Despite their wide range of applications, workflow systems still suffer from lack of an agreed and standard modelling technique. It is a motivating research area and some researchers have proposed different modelling techniques. Petri nets, among the other techniques, are one of the mainly used modelling techniques for both qualitative and quantitative analysis of workflow and workflow systems. We have briefly presented a way of mapping workflow into Petri nets, which can be used as a basis for such systems. A lot of available papers on Petri net-based modelling of workflow have been reviewed and classified. © 2001 Elsevier Science B.V. All rights reserved.

Keywords: Petri nets; Workflow; Workflow management system; Modelling; Business process

1. Introduction

Workflow management [60,66] is an organisational process which has always existed [64], but recently it has become the focal point for many researchers and has attracted many commercial companies. The core of workflow systems is the notion of business process [23], which is a set of activities with a common goal. Workflow involves a set of partially ordered activities to be undertaken using human or other resources of the enterprise in order to meet some predefined requirements. Workflow automation allows organisations to become more flexible and adaptable by continuously customising their work to be able to cope with the rapidly changing business environment.

Petri nets are widely studied [72] and successfully applied in different discrete-event dynamic systems, e.g. [35,50], which are characterised by parallelism and synchronisation. The strong mathematical foundation of Petri nets and the availability of a wide range of supporting tools have made them popular among academic researchers. Petri net-based modelling and analysis of workflow and workflow systems is an active research area in academia, although a small number of vendors [30,56] have developed Petri net-based *workflow management system* (WfMS).

This paper presents an overview of existing papers on modelling workflow and workflow sys-

* Corresponding author. Tel.: +44-1524-593867; fax: +44-1524-844885.

E-mail addresses: k.salimifard@lancaster.ac.uk (K. Salimifard), m.wright@lancaster.ac.uk (M. Wright).

PII: S 0 3 7 7 - 2 2 1 7 (0 0) 0 0 2 9 2 - 7

tems, which are based on Petri nets. The rest of the paper is organised as follows. In Section 2 a brief introduction to workflow and its terminology is presented. Petri nets and their formal definition are presented in Section 3. In Section 4 a brief review of published papers on Petri net-based modelling of workflow is discussed. Finally, a conclusion derived from the papers reviewed and proposals for future work are presented in Section 5.

2. Workflow and workflow management system

Although the concept of workflow has been developed from the notion of factory automation [64], its history goes back to the 1970s, when the first prototype of workflow [97] was developed. During the 1970s and 1980s, the notion of Office Information Systems captured a great amount of consideration. Backtalk [75], OfficeTalk-P [42], Office Information System [41], and OfficeTalk-D [40] were introduced as early prototypes for workflow systems. Due to the unavailability of sufficient technology at the time, some workflow applications have not been successful [26]. A brief overview of the history and development of workflow technology is presented in [37]. Most existing WfMSs have evolved from some computer-based systems such as database management system, document imaging or electronic mail. Since the late 1980s, workflow has become a motivating and challenging research field. Some active research groups (for example [74,93,69]) and organisations (for example [96,87,19]) are doing research on workflow. An enormous number of vendors have developed workflow products [11,47,54]. A comprehensive resource listing of workflow software, research and links is available on the web [92].

2.1. Workflow system characteristics

Workflow management is a new approach to the problem of controlling, monitoring, optimising and supporting business processes [7]. A *workflow* itself is a process in which documents, information or tasks are passed from one participant to another. It is a set of activities involving the co-ordinated execution of multiple tasks performed by different processing entities [27] and covers the flow of information and control within and between organisations. The Workflow Management Coalition [94] defines a workflow as a computerised facilitation or automation of a business process, in whole or part. A specific software system, so-called WfMS, controls automated aspects of workflow. It defines, manages and executes workflow through the execution of software whose order of execution is driven by a computer representation of the workflow logic [94,95]. In WfMS, the process itself is of essential importance and can be defined explicitly. Each step of the workflow is executed for a specific case independent of the other cases. Although most workflow processes are used within a department, *intra-organisational* workflow links activities in different departments, and *inter-organisational* workflow covers distributed processes between different organisations.

As depicted in Fig. 1, a WfMS is composed of two main components: *the workflow model* (process definition) and *the workflow execution module* (workflow enactment service) [38]. The former is used in *build-time* to generate a computerised definition of a business process. It provides (graphical) modelling tools and helps the designer to design, test and validate workflow processes. The execution module is composed of a set of software responsible for creating and controlling instances of the processes during *run-time*. It provides interfaces with authorised participants enabling them to perform specific steps of the process.

Using the run-time controller, WfMS manages the execution and sequencing of the various tasks of the workflow process. It retrieves and monitors the status of different instances of processes from the run-time controller. WfMS initialises an instance of a workflow as a new case, which is generated by a customer. An instance of each *task* of the process for the new case, known as a *work-item*, is placed in the *work-list* of all persons (*actors*) eligible to perform it. Once a work-item as an *activity* is assigned to an actor, all the other instances of the same work-item are removed from the work-list of the other eligible actors. Assigning

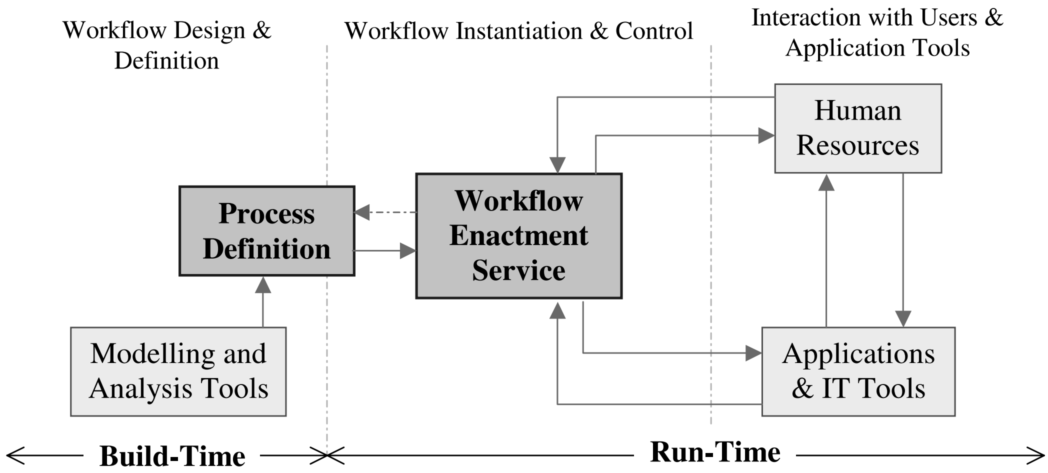

Fig. 1. Workflow system components and phases.

a work-item to an eligible actor can be forced by WfMS (so-called *push* method) or, in a more flexible system, actors are allowed to select a work-item from their work-list (so-called *pull* method). Once execution of an active instance of activity is finished, WfMS creates instances of the next work-items. More details on WfMS can be found in [7,66,96].

Based on the level of persistence in process definition and routing of tasks, three types of workflow are introduced [7,64]. *Production workflow* is characterised by a fixed definition of tasks and order of execution. In *administrative workflow*, cases follow a well-defined procedure but alternative routing of a case is possible. *Ad hoc workflow* [86] covers cases derived from a predefined or template process and allows that for each case, the template be modified to meet some specific needs.

3. Petri nets

Petri nets are a class of modelling tools, which were originated by Petri [78]. They have a well-defined *mathematical foundation* and an easy-to-understand *graphical* feature. The graphical nature of Petri nets makes them self-documenting and a powerful design tool, which facilitates visual communication between the people who are engaged in the design process. On the other hand, they are based on a strong mathematical formalism, which makes it possible to set up mathematical models describing the behaviour of the system [72]. Moreover, validation of the model can be carried out using Petri net analysis techniques. Petri nets are specially suited for modelling and analysing of discrete event dynamic systems whose behaviour is characterised by parallelism and synchronisation [35].

A Petri net is a directed graph consisting of three structural components, *places*, *transitions* and *arcs*. Places, which are drawn as circles, represent possible states or conditions of the system while transitions, which are shown by bars or boxes, describe events that may modify system states. The relationships between places and transitions are represented by a set of arcs, which are the only connectors between a place and a transition in either direction. There is no connection between two nodes of the same type. The dynamic behaviour of a system can be represented using *tokens*, which graphically appear as black dots in places. Adapted from [72], a formal definition of Petri nets is presented.

A Petri net is a 4-tuple $N = \langle P, T, F, M_0 \rangle$, where:

$P = \{p_i : i = 1, \ldots, |P|\}$ is a finite set of places,

$T = \{t_j : j = 1, \ldots, |T|\}$ is a finite set of transitions, $P \cap T = \Phi$,

$F \subseteq (P \times T) \cup (T \times P)$ is the set of directed arcs representing flow relations, joining places and transitions together,

$M_0 : P \to \{0, 1, 2, \ldots\}$ is the initial marking.

A marking $M = \{M(p_1), M(p_2), \ldots, M(p_{|P|})\}$ representing a *state* of the modelled system, is a distribution of tokens over the set of places. Starting from an initial marking M_0, a new marking M is *reachable* if it can be reach by means of a change to the state of the system. This is modelled by the firing of transitions. For a transition $t \in T, {}^{\cdot}t = \{p \in P : (p, t) \in F\}$ is the set of its *input places* and $t^{\cdot} = \{p \in P : (t, p) \in F\}$ is the set of its *output places*. A transition t is *enabled* in M if for all $p \in {}^{\cdot}t$, $M(p) \geqslant 1$. An enabled transition may *fires*, which changes the current marking M into a new marking M'. The effect of firing a transition t can be expressed as $M[t\rangle M'$ or $M \xrightarrow{t} M'$. When a transition fires, it *consumes* one token from each of its input places and *produces* one token in each of its output places.

The aforementioned formalism is just a classical Petri net [72]. There are many extensions of Petri nets, each having its own properties. Some researchers have added new characteristics to Petri nets to enable them to model specific real situations. In a *coloured Petri net* or CPN [59], for example, tokens are distinguishable and each of them carries a *data value*, which belongs to a given *data type*. Using a CPN, the description and analysis of the modelled system is more compact and amenable. It is possible to include data manipulation and to create a hierarchical representation of the system. In *timed Petri nets* and their subclasses, a notion of time is applied to Petri nets, which may be associated with either transitions [67], places [29], tokens [15], or arcs [88]. These types of Petri net are mainly used in analysing the temporal behaviour of systems.

Having a Petri net-based model of a system, it is possible to study two classes of properties of the system. *Behavioural* or *marking dependent properties* are those depending on the initial marking of the model. That is, for different initial markings, we may have different results. *Structural properties* relate to the structure of the modelled system and they are independent of the initial marking.

Definitions, properties and analysis techniques of Petri nets are not completely covered and are beyond the scope of this paper. The interested reader is referred to, for example, [31,34,59,72,77] for theoretical aspects and [35,67] for applications of Petri nets.

4. Workflow modelling

Based on their modelling goals, modellers may use different techniques in workflow modelling. For example, Object-Oriented approach [24], workflow loops based on the language/action perspective [73], modified version of the Entity–Relationship (ER) model [27] and transaction model [23] have been successfully used. Existing workflow products suffer from severe limitations [38]. It is criticised [18] that they have (i) no needed functionality, (ii) no clear set of definitions, and (iii) no general conceptual model.

Recently, Petri nets have been mainly considered as a modelling tool for workflow and workflow systems [1,8,37]. There are a number of reasons [13,39,76] for using a Petri net-based approach in modelling and analysing workflow systems. Petri nets allow a graphical representation to ease the understanding of the modelled system and at the same time, they can be used in formal analysis, verification and validation of the model [2,9,11,32]. In a brief summary, different modelling techniques can be applied to workflow modelling, but Petri nets are the only formal techniques able to be used for both structural modelling and a wide range of qualitative and quantitative analysis.

A number of academic researchers and software vendors have focused on using Petri nets for modelling workflow and workflow systems. A general approach towards the modelling, validation and verification of business processes is described in [32] and a classification of Petri net extensions applied to workflow modelling is presented in [58]. The following subsections cover two main approaches in application of Petri nets in workflow modelling.

4.1. Petri net-based workflow modelling

Despite some criticism of the application of Petri nets to workflow modelling [52] and the impossibility of representing some business process requirements in Petri nets [51], several authors (see for example [1]) have applied different variants of Petri nets. For the first time, Zismann [97] applied Petri nets to represent office procedures as the early idea of workflow systems. His work was followed by Ellis [42] who introduced an extension of classical Petri nets, so-called Information Control Net (ICN) and applied it to model office information system. Since the early 1980s, workflow modelling based on Petri nets has captured a great amount of interest and is still an active and motivating research area. There are a number of special sessions in international conferences mainly focused on Petri net-based modelling of workflow, for example [8].

Van der Aalst has shown [11,13] that workflow primitives can be mapped onto Petri nets. He has discussed [7] that a workflow has three different dimensions: *process*, *case* and *resource* dimensions. It is shown that for process and case dimensions, a Petri net is a suitable modelling tool. He has pointed out that (human) resource dimension of workflow can be modelled by Petri nets. Depending on the modelling objectives, different subclasses of Petri nets can be used in workflow modelling. If modelling concentrates on the structural aspects of a single workflow, a classical Petri net is a suitable modelling technique, while Petri nets with a notion of time have to be used when analysing time-dependent behaviour of workflow. For a workflow management system, where a number of instances of the same or different workflows have to be modelled and traced, it is necessary to choose a subclass of Petri nets in which it is possible to distinguish these different instances. In this case a CPN can be used.

In general, each workflow *task* is represented by a corresponding transition. Places represent *pre-* and *post-conditions* or *resources* needed for a task to be performed. Arcs represent *logical relationships* between tasks and the *flow* of the work. A nontrivial example adapted from [9] is represented.

This workflow consists of five tasks: *submit_claim*, *check_insurance*, *contact_garage*, *send_letter*, and *pay_damage*. The process is modelled in Fig. 2. It is started with the place p_0 and ended with the final place p_7. Each process task is represented by a transition. It is assumed that two tasks *contact_garage* and *check_insurance* may be performed simultaneously. Therefore, two additional transitions, t_2 and t_5 model *fork* and *join* to represent the *parallel execution* of tasks. There is a token in p_0, thus t_1 is the only *enabled* transition. Firing t_1 consumes the token in p_0 and produces a new token in p_1, which means a customer has claimed for damage payment. Then t_2 is enabled. If it fires a token is produced in p_2 and p_3, then t_3 and t_4 are enabled at the same time. This *state* represents the possibility of parallel execution of two tasks modelled by these transitions. t_5 is enabled whenever there is one token in both p_4 and p_5. It guarantees the correct execution of parallel tasks. If t_5 fires, both tokens in p_4 and p_5 are consumed and a token is produced in p_6, enabling both t_6 and t_7. Since there is just one token in p_6, these two transitions are in *conflict* and only one of them

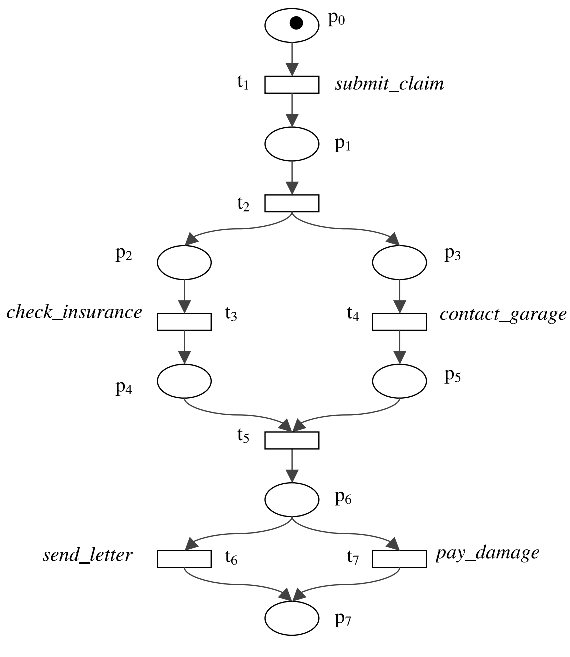

Fig. 2. Petri net model of workflow.

may fire. Firing each of them consumes the token from p_6 and produces a token in p_7. It means a workflow process for one *instance* of *insurance_claim* has finished.

Starting from an initial marking, M_0, all possible states, i.e. the *reachability tree*, of the model can be obtained. It can be used as a basis for a range of marking-dependent analysis, for example, whether the model is safe, deadlock-free or livelock-free (for more details see [12,34,72]). Validation of the model, an important issue in modelling, can easily be done. The analysis of Petri net models can be automated by using a suitable Petri net software package such as *GreatSPN* [48], *TimeNET* [82] or *DesignCPN* [36].

There is no notion of time or resources in the aforementioned model. In addition, this model just represents a single instance of a workflow process, while in a real situation there is more than one process and more than one instance of each process being executed. All of them may compete for resources and obviously it takes time to execute any task of each process. Petri nets are able to represent these important issues. Suppose, for example, task *check_insurance* is performed by an actor who play the role *role1*. An additional place *Role1* represents this assumption as depicted in Fig. 3. Now, the task *check_insurance*, modelled by t_3, can be performed if t_3 is enabled, i.e. there is a token in both places p_2 and *Role1*. During the execution of the task *check_insurance*, which takes τ_3 time units, the actor playing *role1* is busy and not available to be assigned to another job. Elapsing τ_3 time units, firing of t_3 is finished, a new token is produced in p_4, and actor *role1* is released. Therefore, a token is put in place *Role1*. The duration of an activity modelled by a transition can be either deterministic or stochastic [35,67,88].

If the analysts are interested in the time-dependent behaviour of the system, an extension of timed Petri nets [48,82] can be used. CPN [59] and its extensions are able to represent different instances of the workflow process within one model. Therefore, depending on the modelling goals, a different class of Petri nets can be used.

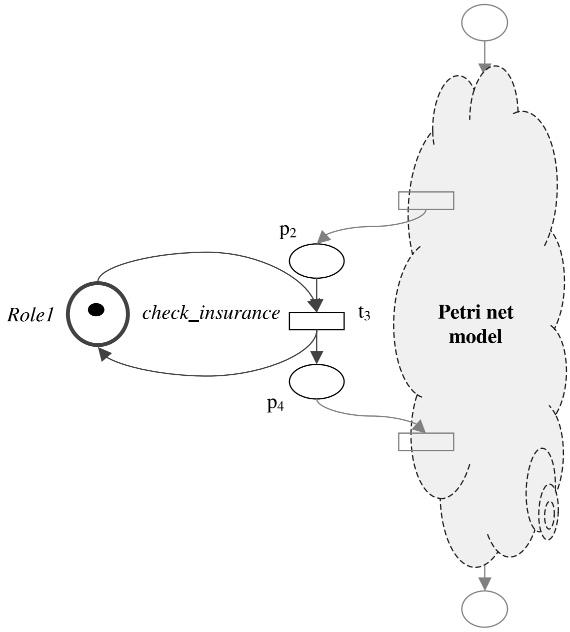

Fig. 3. Resource modelling of workflow.

4.2. Petri net-based workflow software

Successful use of Petri nets for workflow modelling in academic research encouraged software developers to provide Petri net-based workflow modelling software tools. Protos [79] is a Petri net-based software package. It is possible to export the processes modelled with Protos into some WfMS products such as COSA [30], and FLOWer [45]. Structware [84] is a tool that allows the designer to design and simulate business processes based on Petri net models. The resulting model, then, can be exported to Staffware [83]. ExSpect [43] is a business process modelling tool, able to model and analyse workflow processes in terms of Petri nets. The designer can import the process model into COSA [30] which is another process modelling tool based on Petri nets. It is also an enactment tool to model and execute complex workflow processes. COSA can interface with ExSpect, Woflan, and Protos. Woflan [91] is a Petri net-based software tool for analysing workflow process. Assuming the availability of the resources needed and fair triggering [11,13,61], Woflan provides a wide range of analyses such as soundness property [12], to check the correctness of workflow process [10]. It interfaces with the other commercial workflow products such as COSA [30], Staffware [83] and Protos [79]. INCOME [57] is an extended and commercial version of [76]. It supports modelling the behaviour of workflow processes.

4.3. Modelling and qualitative analysis of workflow

Qualitative analysis concentrates on properties such as deadlock freeness, the correct flow of the work, absence of overflow, and the presence of certain mutual exclusion wherever shared resources are used. In fact, the main objective of qualitative analysis is to prove that the model is valid and to find answers for some questions such as: Is there any deadlock in the system? Is sequence of the states correct? Does any process end improperly? Is there any nonsolvable conflict for access to resources among different activities?

Since applying ICN [42] in workflow domain, Petri net-based qualitative analysis of workflow has been the focal point of many papers. Some authors have successfully applied Petri nets, mainly concentrated on structural modelling and analysis of the qualitative properties of workflow processes. In [21] a modified version of an ordinary Petri net, called a *Temporal Constraint Petri Net* (TCPN), has been used to model workflow applications at conceptual level and as a test-bed to verify the system properties. Each task is decomposed into some primitives and therefore the states between the primitives can be represented. Task dependencies are categorised into *control-flow dependency*, *value dependency* and *external dependency*. A logical verification is used to verify the main properties of the workflow model.

In [16,17] a high level Petri net is used to model a business process. The model determines what process is performed, how and by whom. A type of free choice Petri net [34], which is named as *Business Procedure net* (BP-net) is used in [14]. It is proposed for representation, validation and verification business processes. Then this model has been modified into a *workflow net* (WF-net) [2,9,12] as a sound model for workflow. It is shown that the WF-net is well structured. The author proposed hierarchical WF-net for modelling complicated processes. Kindler [63] has shown that the soundness property proposed in [12] is not solely sufficient to guarantee the correctness of the process model. In order to obtain a strong sound workflow model, he has added 1-safe property to WF-net.

Verification of workflow processes is the focal point of [85]. The process is represented as a Petri net model. A technique combining Petri net and process algebra is used for the formal verification of workflow processes.

A high level Petri net has been suggested in [18] in which a control activity is used for the routing of jobs (work-items) inside a procedure, the synchronisation of work and the duplicating of job tokens. The behaviour of the workflow process has been analysed using ordinary Petri nets [28]. The process is defined as a composition of four basic entities. Each entity of the process is modelled as an ordinary Petri net. These nets have been combined to obtain the global Petri net-based model of the process.

In the INCOME/WF project [76] a Petri net-based WfMS, which is a class of high-level Petri net, has been combined with an ER model for modelling and prototyping workflow. The method is based on the so-called *nested relation/transition net* (NR/T net), which can be used for integrated modelling of concurrent processes and related complex objects in distributed environments. INCOME is a new version of INCOME/WF and is commercially available at [57].

A Petri net representation of a *workflow authorisation model* (WAM) has been proposed in [20]. This model ensures that in a workflow system, authorised resources are granted privileges on required tasks only while the task is being executed. In order that authorisation is granted to the right resources in the right time period and being invoked during the time the task is being undertaken, a coloured timed Petri net (CTPN) is used to synchronise the authorisation with the flow of the work. Exception handling of workflow has been considered in [39] and another type of high level Petri net, namely an ICN, has been used to model procedures independent of their implementations.

Distinguishing between two main components of a workflow system, i.e. the *workflow model* and the *workflow execution module*, it is stressed [22] that a simpler workflow model leads to a more flexible and adaptive workflow execution module. Based on this idea, a workflow model based on Elementary Net System [80] is proposed. It allows the designer to model the workflow process and generate a large class of behaviour from the process model. The proposed model supports process modification and run-time exception handling.

Inter-organisational workflow and the availability of the Internet have affected workflow technology. An Internet-based workflow has been considered in [89]. An *SGML* net, a variant of high level Petri nets, has been proposed to capture the process of generating and manipulating structured documents. The SGML net has a Petri net structure as a base layer. Tokens are document instances with an inner structure. Each place is associated with an SGML document type representing the type of the place. A marking of a place is a set of document instances restored in the place. Each arc is inscribed with a so-called *document template*, which is used to manipulate document instances in the place adjacent to the arc. The Firing rule for the SGML is modified to prevent violation between respective places.

It is shown [5] that a Petri net-based workflow can be used in a distributed environment. To check the correctness of inter-organisational workflows, the notion of *global soundness* based on the soundness property of a workflow model [9,2] is introduced. The model itself is based on message passing between workflows and a message sequence chart is used to bridge different workflows. Kindler et al. [62] stressed that sometimes it is not possible to use the technique introduced in [5] to check the global soundness of inter-organisational workflows. The authors proposed an algebraic technique based on Petri nets for automatically analysing a single workflow process in isolation. This technique checks the local soundness of the process model and, using a composition theorem, it guarantees the global soundness of the process.

Workflow fragmentation as a result of distributed workflow systems is considered in [49]. Using a high level Petri net, *predicate/transition net* (Pr/*T* -net) [46], places (predicates) represent relation schemes between fragments and transitions represent a class of operations on the relations in the adjacent places. Fragmentation has been successfully modelled and interpreted by WfMS. A workflow may be moved between different departments and organisations using different types of workflow engines.

Merz et al. [68] have integrated a CPN-based workflow model into Common Open Service Market (COSM), which is a distributed environment. CPN has been used for the formal specification of a workflow process and concurrent activity management and control. In order to verify the coordination between workflow activities in the distributed architecture, simulation has been used.

In current workflow systems it is difficult to handle ad hoc and radical changes. As a solution to the problem, the reconfigurable WF-net [25], which is an extension of the WF-net introduced in [9], is proposed. It models the configuration modification of a workflow process. This net describes the structure of the process for some sort of predefined mode of operation. It can dynamically modify its configuration by rewriting some of its places. The proposed net is considered only with respect to an individual case, which means that a concurrent processing of multiple cases cannot be modelled. In [6,86] the notion of *adaptive workflow* is introduced and two types of change, *ad hoc* and *evolutionary* change, have been addressed. Ad hoc changes are related to rare events, which are case based, while evolutionary changes are often the outcomes of reengineering the system. To deal with these changes *generic process models*, which are illustrated by Petri nets, have been proposed. It has been shown that this method can handle both types of changes.

Some authors have concentrated on object oriented modelling and have proposed the notion of object oriented Petri nets in workflow modelling. Wikarski [90] has introduced the notion of Modular Process Nets to the formal representation of business processes. It is a low level Petri net model combined with an object oriented approach allowing several types of abstraction. The author has introduced the notion of *sensor transition* to be able to model the communication between different nets modelling different processes. Resource management is explicitly considered in [65,53]. The author proposed an object-oriented extension of Petri nets called a Higher-Order Object Net (HOON) to model workflow processes and their related resources as different object models. All the object models are interacting in a client/server distributed environment.

In [70] a CPN-based object oriented approach for modelling business processes called Business Process Petri nets (BPP-nets) is presented. A system containing processes and resources is mapped into functional units and modelled as objects. Methods are represented by transitions and objects communicated to each other through some interfaces. A token in BPP-net may have a complex internal structure. To cope with changing workflow, BPP-nets support dynamic adaptation of the model.

Van der Aalst et al. [4] have launched a repository for workflow modelling patterns. All possible patterns of workflow models including basic control patterns, branching and synchronisation patterns, structural patterns, state-based patterns, and cancellation patterns are taken into account. In [3] four workflow patterns, i.e., synchronisation merge, implicit termination, multiple instance requiring synchronisation, and deferred choice are described independently from current workflow languages. Based on the selected patterns, the author evaluated leading workflow products but none of them support all the patterns. As a result, new requirements for workflow languages have been indicated. The proposed patterns can be used as a check list in selecting a workflow product.

4.4. Quantitative analysis of workflow based on Petri nets

Having a functionally approved system does not guarantee that the system will meet customer requirements. Is the process finished by a predetermined time? What is the resource utilisation for the process? What is the expected time an activity has to wait before its processing starts? These questions have to be answered before the system is installed. Quantitative analysis consists of calculating *performance*, *responsiveness* and *resource utilisation* indices. Despite the importance of quantitative properties of workflow compared with papers that have focused on qualitative analysis, the quantitative analysis of workflow has not captured the attention of many researchers.

Some extensions of Petri nets, in which the notion of time is considered, are suitable for such analysis. A generalised stochastic Petri net (GSPN) [67], amongst the other types of Petri nets, is the one that mostly has been used. In [81] a high-level Petri net is used to model a business process as a workflow process definition. The process has been mapped into a GSPN. A simple GSPN, which is a so-called *load equivalence aggregation* (LEA) model, has been developed to evaluate the performance of the human resources. Then, the model is simulated using a coloured GSPN (CGSPN).

A GSPN representation of a workflow model has been proposed in [44]. Each process element of workflow is described by a single transition and a GSPN represents the precedence relation among process elements. A method based on a continuous time Markov chain (CTMC) has been used to obtain upper bounds of the execution performance. In [21], each place and transition is associated with a time interval to denote the lower and upper bounds of the time delay for the associated place or transition. Tokens are associated with a timestamp, which represents the arrival time of the token in each place of which it arrives.

In [33] an approach based on place/transition nets to modelling business processes is presented. In order to study the performance of the process, the place/transition model is enhanced with the notions of time and cost. The resulting model is simulated and some performance related measures in terms of time or cost is obtained.

Petri net-based performance analysis of a workflow process is mainly considered in [55]. All the routing constructs of the process are mapped into a high-level stochastic Petri net. Time duration, which may follow any arbitrary distribution, is associated with transitions representing activities. Assuming a liberal resource schedule, i.e. infinite availability of resources, the throughput time of the process is analytically computed.

In addition to Petri net-based quantitative analysis of workflow, simulation technique is another alternative. In [71], for example, an approach for integrating simulation modelling and analysis capabilities within the WfMS is developed. Simulation modelling is used for studying the efficiency of workflow designs as well as studying the general performance and reliability of WfMS.

5. Final remarks and conclusion

In this paper we presented an introduction to workflow and workflow systems. We mainly focused on the process definition of workflow, based on the terminology defined by WfMC. A general introduction to Petri nets is covered. Using a nontrivial example, a general methodology to map workflow into Petri nets has been explained. We summarised and classified those available papers, which have concentrated on Petri net-based modelling of workflow. Most of the reviewed papers have used different versions of high-level Petri nets and mostly have focused on just the modelling and qualitative analysis of workflow. Quantitative properties of workflow such as performance and responsiveness indices have rarely been considered.

Finally, the modelling tools should be chosen according to the problem in hand. For structural analysis of workflow processes, ordinary Petri nets are proposed, which are supported with plenty of analytical methods. Behavioural analysis of workflow needs a modified version of high-level Petri nets. Extensions of timed CPNs are necessary to represent a workflow process, with the required details, while being able to calculate quantitative measures related to the dynamic behaviour of the process. Workflow models which use available tools other than those tools introduced in Section 4.2, have no practical benefit for the WfMS, since there is no link to an execution environment. Petri net-based modelling tools should be embedded in the workflow system, if the designer wishes to use the advantages available from using Petri nets.

Acknowledgements

The authors would like to thank Behrouz Zarei and Peter Kueng for their helpful comments and suggestions on earlier version of the manuscript. The authors also wish to thank anonymous referees for their feedback and comments. This research is sponsored by Persian Gulf University, Bushehr, Iran and MSRT Iran.

References

[1] W.M.P. van der Aalst, J. Desel, A. Oberweis (Eds.), Business Process Management: Models, Techniques and Emperical Studies, Lecture Notes in Computer Science, vol. 1806, Springer, Berlin, 2000.

[2] W.M.P. van der Aalst, Workflow verification: Finding control-flow errors using Petri net-based techniques, in [1], pp. 161–183.

[3] W.M.P. van der Aalst, A.H.M. ter Hofstede, B. Kiepuszewski, A.P. Barros, Advanced workflow patterns, in: O. Etzion, P. Scheuermann (Eds.), Seventh International Conference on Cooperative Information Systems (CoopIS 2000), Lecture Notes in Computer Science, vol. 1901, Springer, Berlin, 2000, 18–29.

[4] W.M.P. van der Aalst, A.H.M. ter Hofstede, B. Kiepuszewski, A.P. Barros, Workflow pattern, http://www.mincom.com/mtrspirt/workflow.

[5] W.M.P. van der Aalst, Interorganisational workflows: An approach based on message sequence charts and Petri nets, Systems Analysis, Modelling, Simulation 34 (3) (1999) 335–367.

[6] W.M.P. van der Aalst, How to handle dynamic change and capture management information? An approach based on generic workflow models, Technical Report, UGA-CS-TR-99-01, University of Georgia, Department of Computer Science, Athens, USA, 1999.

[7] W.M.P. van der Aalst, The application of Petri nets to workflow management, The Journal of Circuits Systems and Computers 8 (1) (1998) 21–66.

[8] W.M.P. van der Aalst, G. De Michelis, C.A. Ellis (Eds.), WFM'98, Proceedings of Workflow Management: Net-Based Concepts, Models, Techniques and Tools, Lisbon, Portugal, UNINOVA, Lisbon, June 1998.

[9] W.M.P. van der Aalst, Verification of Workflow Nets, Lecture Notes in Computer Science, vol. 1248, Springer, Berlin, 1997, pp. 407–426.

[10] W.M.P van der Aalst, H.M.W. Verbeek, D. Hauschildt, A Petri net-based tool to analyze workflows, in: Proceedings of PNSE'97, Workshop on Petri Nets in System Engineering: Modelling, Verification and Validation, Hamburg, Germany, 25–26 September 1997.

[11] W.M.P. van der Aalst, Petri net-based workflow management software, in: A. Sheth (Ed.), Proceedings of NFS

Workshop on Workflow and Business Process Automation in Information Systems, State-of-the-Art and Future Directions, Athens, GA, 1996, pp. 114–118.

[12] W.M.P. van der Aalst, Structural characterisations of sound workflow nets, Computing Science Reports 96/23, Eindhoven University of Technology, Eindhoven, 1996.

[13] W.M.P. van der Aalst, Three good reasons for using a Petri net-based workflow management system, in: S. Navathe, T. Wakayama (Eds.), Proceedings of the International Working Conference on Information and Process Integration in Enterprises (IPIC'96), Cambridge, MA, 1996, pp. 179–201.

[14] W.M.P. van der Aalst, A class of Petri net for modelling and analysing business processes, Computing Science Reports 95/26, Eindhoven University of Technology, Eindhoven, The Netherlands, 1995.

[15] W.M.P. van der Aalst, Interval timed coloured Petri nets and their analysis, in: M.A. Marsan (Ed.), Application and Theory of Petri nets 1993, Lecture Notes in Computer Science, vol. 691, Springer, Berlin, 1993, pp. 453–472.

[16] W.M.P. van der Aalst, K.M. van Hee, Business process redesign: A Petri net-based approach, Computers in Industry 29 (1-2) (1996) 15–26.

[17] W.M.P. van der Aalst, K.M. van Hee, Framework for business process redesign, in: J.R. Callahan (Ed.), Proceedings of the Fourth Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE95), IEEE Computer Society Press, Silver Spring, MD, 1995, pp. 36–45.

[18] W.M.P. van der Aalst, K.M. van Hee, and G.J. Houben, Modelling workflow management systems with high-level Petri nets, in: G. De Michelis, C. Ellis, G. Memmi (Eds.), Proceedings of the Second Workshop on Computer-Supported Cooperative Work, Petri Nets and Related Formalisms, 1994, pp. 31–50.

[19] AIIM, Association for Information and Image Management International, Silver Spring, USA. http://www.aiim.org.

[20] V. Atluri, W. Huang, An authorization model for workflows, Lecture Notes in Computer Science 1146 (1996) 44–64.

[21] N.R. Adam, V. Atluri, W.K. Huang, Modelling analysis of workflow using Petri nets, Journal of Intelligent Systems 10 (1998) 131–158.

[22] A. Agostini, G. De Michelis, K. Petruni, Keeping workflow models as simple as possible, in: Proceedings of the Workshop on Computer-Supported Cooperative Work, Petri Nets and Related Formalism within the 15th International Conference on Application and Theory of Petri Nets, Zaragoza, Spain, June 1995.

[23] G. Alonso, D. Agrawal, A. El Abbadi, M. Kamath et al., Advanced transaction models in workflow context, Research Report, IBM Research Division, San Jose, USA.

[24] A.P. Barros, A.H.M. ter Hofstede, Towards the construction of workflow suitable conceptual modelling techniques, Information Systems Journal 8 (4) (1998) 313–337.

[25] E. Badouel, J. Oliver, Reconfigurable nets, a class of high level Petri nets supporting dynamic changes within workflow systems, Publication Internet Number 1163, IRISA, France, 1998.

[26] J. Bair (Ed.), Office Information Systems: Why Some Works and Other Fails, Stanford University Conference Proceedings, Center for Information Technology, Stanford University, 1981.

[27] F. Casati, S. Ceri, B. Pernici, G. Pozzi, Conceptual modelling of workflows, in: Proceedings of 14th Object-Oriented and Entity-Relationship Approach, Gold Coast, Australia, Lecture Notes in Computer Science, vol. 1021, 12–15 December 1995, pp. 341–354.

[28] C. Coves, D. Crestani, F. Prunet, Design and analysis of workflow processes with Petri nets, in: Proceedings of IEEE International Conference on Systems, Man and Cybernetics, vol. 1, 1998, pp. 101–106.

[29] J.E Coolahan Jr, N. Roussopoulos, Timing requirements for time-driven systems using augmented Petri nets, IEEE Transactions on Software Engineering SE 9 (1983) 603–616.

[30] COSA SOLUTIONS, Software-Ley GmbH, Pullheim, Germany, http://www.cosa.de.

[31] R. David, H. Alla, Petri Nets and Grafcet: Tools for Modelling Discrete Event Systems, Prentice-Hall International, Hertfordshire, UK, 1992.

[32] J. Desel, Validation of process models by construction of process nets, in [1], pp. 110–128.

[33] J. Desel, T. Erwin, Modelling, Simulation and Analysis of Business Processes, in [1], pp. 129–141.

[34] J. Desel, J. Esparza, Free Choice Petri Nets, Cambridge Tracts in Theoretical Computer Science 40, Cambridge University Press, Cambridge, 1995.

[35] F. Dicesare, G. Harhalakis, J.M. Proth, M. Silva, F.B. Vernadat, Practice of Petri Nets in Manufacturing, Chapman & Hall, London, 1993.

[36] Design/CPN, Tutorial for X-Windows, Version 2, Meta Software Corporation, 1993.

[37] C.A. Ellis, G.J. Nutt, Workflow: The process spectrum, in: NSF Workshop on Workflow and Process Automation in Information Systems, Athens, Georgia, 1996, pp. 140–145.

[38] C.A. Ellis, K. Keddara, Dynamic change within workflow systems, Technical Report, University of Colorado, July 1993.

[39] C.A. Ellis, G.J. Nutt, Modelling and enactment of workflow systems, in: M. Ajmone, Marsan (Ed.), Proceedings of 14th International Conference on Application and Theory of Petri Nets, Lecture Notes in Computer Science, vol. 691, Springer, Berlin, June 1993, pp. 1–16.

[40] C.A. Ellis, Officetalk-D: An experimental office information system, in: Proceedings of the First ACM Conference on Office Information System, June 1982.

[41] C.A. Ellis, G.J. Nutt, Office information system and computer science, ACM Computing Surveys 12/1 (1980) 27–60.

[42] C.A. Ellis, Information control nets: A mathematical model of office information flow, in: Proceedings of the Conference on Simulation, Measurement and Modelling of Computer Systems, ACM Press, 1979.

[43] ExSpect, Bakkenist Management Consultants, Diemen, The Netherlands, http://www.exspect.com.
[44] A. Ferscha, Business workflow analysis using generalized stochastic Petri nets, in: Ninth Austrian–Hungarian Informatics Conference, 1994, pp.222–234.
[45] FLOWer, Pallas Athena, Plasmolen, the Netherlands. http://www.pallas-athena.com/flower.
[46] H.J. Genrich, Predicate/transition nets, in: W. Brauer, W. Reisig, G. Rozenberg (Eds.), Advances in Petri Nets, Part I, Lecture Notes in Computer Science, vol. 254, 1986, pp. 207–247.
[47] D. Georgakopoulos, M. Hornick, A. Sheth, An overview of workflow management: From process modelling to workflow automation infrastructure, International Journal on Distributed and Parallel Databases 3 (1995) 119–153.
[48] GreatSPN 2.0 User Manual, Dipartmento di Informatica, Università di Torino, Italy, 1995.
[49] V. Guth, K. Lenz, A. Oberweis, Distributed workflow execution based on fragmentation of Petri nets, in: R. Traunmuller, E. Csuhaj-Varju (Eds.), Proceedings of 15th IFIP World Computer Congress Telecooperation, The Global Office, Teleworking and Communication Tools, Vienna, Budapest, September 1998.
[50] K.M. van Hee, Information Systems Engineering, Cambridge University Press, Cambridge, 1994.
[51] A.P. Holt, Management-oriented models of business processes, in [1], pp. 99–109.
[52] Y. Han, HOON: A formalism supporting adaptive workflows, Technical Report UGA-CS-TR-97-005, Department of Computer Science, University of Georgia, November 1997.
[53] Y. Han, J. Himmighofer, T. Schaaf, D. Wikarski, Management of workflow resource to support runtime adaptibility and system evolution, in: Proceedings of Practical Aspect of Knowledge Management, The First International Conference, 30–31 October, 1996.
[54] K. Hayes, K. Laver, Workflow management software: The business opportunity, Technical Report, Ovum Ltd, London, 1991.
[55] K.M. van Hee, H.A. Reijers, Using formal analysis techniques in business process redesign, in [1], pp. 142–160.
[56] INCOME/WF Project, University of Frankfurt, Germany, http://lwi2.wiwi.uni-frankfurt.de/projekte/iwf_proj.htm.
[57] INCOME, Promatis AG, Karlsbad, Germany, http://www.get-income.com.
[58] G.K. Janssens, J. Verelst, B. Weyn, Techniques for modelling workflows and their support of reuse, in [1], pp. 1–15.
[59] K. Jensen, Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use, Vol. 1: Basic Concepts, EATCS Monographs on Theoretical Computer Science, Springer, Berlin, 1992.
[60] S. Jablonski, C. Bussler, Workflow Management: Modelling, Concepts, Architecture and Implementation, International Thomson Computer Press, 1996.
[61] S. Joosten, Trigger modelling for workflow analysis, in: G. Chroust, A. Benczur (Eds.), Proceedings of CON'94, Workflow Management – Challenges, Paradigms and Products, 1994, pp. 147–236.
[62] E. Kindler, A. Martens, W. Reisig, Inter-operability of workflow applications: Local criteria for global soundness, in [1], pp. 235–253.
[63] E. Kindler, Database Theory – Petri Net Theory – Workflow Theory, Humboldt Universitat zu Berlin, Institut fur Informatik, Germany, 1998.
[64] T.M. Koulopoulos, The Workflow Imperatives: Building Real World Business Solutions, Van Nostrand Reinhold, New York, 1995.
[65] M. Lowe, D. Wikarski, Y. Han, Higher-order object nets and their application to workflow modeling, Technische Universitat Berlin, Forschungsberichte der FB Informatik 95–34, 1995.
[66] P. Lawrence (Ed.), Workflow Handbook, Wiley, New York, 1997.
[67] M.A. Marsan, G. Balbo, G. Conte, S. Donateli, G. Franceschinis, Modelling With Generalized Stochastic Petri Nets, Wiley, Chichester, 1995.
[68] M. Merz, D. Moldt, K. Muller, W. Lamersdorf, Workflow modelling and execution with coloured Petri nets in COSM, in: Proceedings of the Workshop on Applications of Petri Nets to Protocols within the 16th International Conference on Application and Theory of Petri Nets, 1995.
[69] METEOR, Large Scale Distributed Information System, University of Georgia, USA. http://lsdis.cs.uga.edu/meteor/meteo.html.
[70] D. Moldt, R. Valk, Object Oriented Petri Nets in Business Process Modelling, in [1], pp. 254–273.
[71] J. Miller, A. Sheth, K. Kochut, X. Wang, A. Murugan, Simulation modelling within workflow technology, in: Proceedings of the 1995 Winter Simulation Conference, Arlington, VA, December 1995.
[72] T. Murata, Petri nets: Properties, analysis and applications, in: Proceedings of the IEEE, vol. 77, 1989, pp. 541–580.
[73] R. Medina-Mora, T. Winograd, P. Flores, Action workflow as the enterprise integration technology, bulletin of the technical committee on data engineering, IEEE Computer Society 16 (2) (1993).
[74] MWMS: Milan Workflow Management System, Universitya degli di Milano, Italy. http://mercurio.sm.dsi.unimi.it/~gdemich/mwms.html.
[75] G.J. Nutt, C.A. Ellis, Balktalk: An Office Environment Simulator, ICC'79 Conference Record, June 1979.
[76] A. Oberweis, R. Schatzle, W. Stucky, W. Weitz, G. Zimmermann, INCOME/WF – A Petri Net-Based Approach to Workflow Management, Institute fur Wirtschaftsinformatik, Germany, June 1996.
[77] J.L. Peterson, Petri Net Theory and the Modelling of Systems, Prentice-Hall, Englewood Cliffs, NJ, 1981.
[78] C.A. Petri, Kommunikation mit Automaten, Ph.D. Thesis, Universität Bonn, Germany, 1962.
[79] Protos 4.0, Pallas Athena Ltd., The Netherlands, http://www2.pallas.nl/protos/uk/protos/.

[80] G. Rozenberg, Behaviour of Elementary Net System, Lecture Notes in Computer Science, vol. 254, Springer, Berlin, 1987, pp. 60–94.
[81] A.K. Shomig, H. Rau, A Petri net approach for the performance analysis of business process, Report No. 116, Lehrstuhl für Informatik III, Universität Wurzburg, Germany, 1995.
[82] TimeNET 2.0 User Manual, Institut fur Technische Informatik, Technische Universität Berlin, 1989.
[83] Staffware, Staffware Plc., Maidenhead, UK, http://www.staffware.com.
[84] Structware, IvyTeam, Zug, Switzerland. http://www.ivy-frame.ch.
[85] M. Voorhoeve, Compositional modelling and verification of workflow processes, in [1], pp 184–200.
[86] M. Voorhoeve, W.M.P. van der Aalst. Ad-hoc workflow: Problems and solutions, in: R. Wagner (Ed.), Proceedings of the Eighth DEXA Workshop on Database and Expert Systems Applications, Toulouse, France, September 1997, pp. 36–41.
[87] WARIA, Workflow and Reengineering International Association, Florida, USA. http://www.waria.com.
[88] B. Walter, Timed Petri nets for modelling and analysing protocols with real-time characteristics, in: Proceedings of the Third IFIP Workshop on Protocol Specification, Testing and Verification, North-Holland, Amsterdam.
[89] W. Weitz, Workflow modelling for internet-based commerce: An approach based on high level Petri nets, Lecture Notes in Computer Science 1402 (1998) 166–178.
[90] D. Wikarski, Introduction to modular process nets, Technical Report TR-96-019, International Computer Science Institute (ICSI), Berkeley, CA, USA, April 1996.
[91] Woflan, Department of Mathematic and Computing Science, Eindhoven University of Technology, The Netherlands. http://www.win.tue.nl/~woflan.
[92] Workflow Software, http://www.workflowsoftware.com.
[93] Workflow Systems, Department of Computer Science, University of Twente, The Netherlands, http://wwwis.cs.utwente.nl:8080/MAICS.
[94] Workflow Management Coalition, The Workflow Reference Model, Document Number TC00-1003, Issue 1.1, January 1995.
[95] Workflow Management Coalition, Terminology and Glossary, Document Number TC-1011, Issue 2.0, June 1996.
[96] Workflow Management Coalition, http://www.aiim.org/wfmc/mainframe.htm.
[97] M.D. Zisman, Representation, specification and automation of office procedures, Ph.D. Thesis, University of Pennsylvania, Wharton School of Business, 1977.