Distributed AI for Ambient Intelligence: Issues and Approaches

Theodore Patkos, Antonis Bikakis, Grigoris Antoniou, Maria Papadopouli, and Dimitris Plexousakis

Institute of Computer Science, FO.R.T.H.
Vassilika Vouton, P.O. Box 1385, GR 71110, Heraklion, Greece
{patkos,bikakis,antoniou,mgp,dp}@ics.forth.gr

Abstract. Research in many fields of AI, such as distributed planning and reasoning, agent teamwork and coalition formation, cooperative problem solving and action theory has advanced significantly over the last years, both from a theoretical and a practical perspective. In the light of the development towards ambient, pervasive and ubiquitous computing, this research will be tested under new, more demanding realistic conditions, stimulating the emergence of novel approaches to handle the challenges that these open, dynamic environments introduce. This paper identifies shortcomings of state-of-the-art techniques in handling the complexity of the Ambient Intelligence vision, motivated by the experience gained during the development and usage of a context-aware platform for mobile devices in dynamic environments. The paper raises research issues and discusses promising directions for realizing the objectives of near-future pervasive information systems.

Keywords: Ambient Intelligence, Distributed AI, Context Awareness, Action Theories, Multi-agent Cooperation.

1 Introduction

The vision of Ambient Intelligence assumes a shift in computing towards a multiplicity of communicating devices disappearing into the background, providing an intelligent, augmented environment, where the emphasis is on the human factor. Realizing this vision requires the integration of expertise from a multitude of disciplines; distributed intelligence, dynamic networks and ubiquitous communications, human-computer interaction and intuitive user-friendly interfacing, robotics, and hardware design are embraced under the influence of the Ambient Intelligence paradigm. This paradigm implies a seamless medium of interaction, advanced networking technology and efficient knowledge management, in order to deploy an environment, where entities describe themselves, are aware of each other and can figure out ways to interoperate at syntactic and semantic levels.

Arranging a physical environment, where mobile and stationary devices communicate and cooperate to achieve common objectives, has proven to be a laborious task for the research community. Although much success has been achieved in

B. Schiele et al. (Eds.): AmI 2007, LNCS 4794, pp. 159–176, 2007.

defining theoretical frameworks for the fields of distributed AI, agent teamwork and coalition formation, planning and reasoning about actions and cooperative problem solving over the last years, the advent of ubiquitous and context-aware computing has produced new challenges, pushing research in these fields to its limit. Existing approaches have difficulty in meeting the real-world challenges imposed by developing ambient information systems, since they typically rely on restricted models and simplifying assumptions, which do not hold in realistic conditions. Already works are being published that question the logic and rational behind some of our larger expectations; Rogers, for instance, in [1] argues that the progress in Ubiquitous Computing research has been hampered by intractable computational and ethical problems and that the field needs to broaden its scope, setting and addressing other goals that are more attainable.

It is our intention in this paper to investigate the challenging new issues that emerge from this domain and focus on research in the field of Distributed Artificial Intelligence under the context of symbolic planning. In particular, we identify and describe three general classes of problems: (a) the challenge of handling the complexity of generating and executing planning tasks in a dynamic and uncertain environment; (b) the challenge of cooperation between devices that have varied skills, but pursue common goals, distributing their resources and capabilities accordingly; and (c) the challenge of sharing enhanced plan representations among participating entities and evaluating their execution. Our selection of this subfield of AI is justified by its high relevance to the distributed nature and goal-directed behavior of ambient computing environments, and also by its significance in the research towards ambient computing engineering. Table 1 summarizes the requirements of each subject and the techniques that are studied in the rest of this analysis. Although, there are previous works that reflect on limitations of each individual approach under various conditions, to the best of our knowledge, this is the first attempt to frame a range of problems under the Ambient Intelligence domain assessing techniques towards this near-future reality, while at the same time proposing enabling directions of research.

The motivation behind this problem statement paper has been our involvement in a running project that concerns the development of a context-aware platform. The paper is more of a survey and analysis of existing methodologies that have been studied as potential integrals of our platform, rather than a focused evaluation report on a specific implementation. Our objective is to identify gaps in the capabilities of current AI techniques and to suggest the most productive lines of research. As such, the contribution is of both theoretical and practical significance; from a theoretical standpoint we raise numerous research issues and challenges that need to be addressed for understanding the domain and enabling ambient computing systems to operate effectively, while from a practical perspective, we discuss shortcomings of state-of-the-art techniques when applied to real-world conditions and suggest ways to overcome their restrictions.

The rest of this paper is organized as follows. We first briefly present our ongoing work in developing a semantics-based framework that supports services for mobile users in a dynamic environment. The next three sections deal with

Table 1. Overview of Problem Analysis

Domain	*Requirements*	*Methods*
Dynamic and Uncertain Environment	Time and Concurrency Continuous Change Non-Determinism Sensing, Knowledge and Belief Natural and Exogenous Actions	STRIPS Situation Calculus Fluent Calculus Event Calculus Action Languages MDPs
Distributed Planning and Coalition Formation	Coordination Cooperation Commitment	GPGP CPS POCL Planning Logic-based Approaches
Plan Representation and Monitoring	Common Plan Interpretation Profile Modelling Online Plan Refinement	HTNs Skeletal Plans Model-based Diagnosis

each of the three problem classes mentioned above. Each section reveals problems that emerge during the development of our platform, projects them to more generic ambient computing requirements and overviews related state-of-the-art research. We conclude in Section 6 with a discussion on enabling research directions towards a pragmatic Ambient Intelligence implementation.

2 Contextual Pedestrian Guide

To motivate the need for a theoretical consideration of the field, we concentrate on the design and analysis phases of a running project that involves the development of semantics-based context-aware services utilizing technologies and formalisms from the Semantic Web and Ambient Intelligence domains. The objective of the project is to explore the intelligent pedestrian navigation by implementing a Context Pedestrian Guiding platform (CG) [2] for users in indoor environments. We focus on modelling and representing context for efficient processing and dissemination of context-based knowledge, in order to develop services for mobile users. A working prototype of the system has been installed in the premises of our research facilities.

2.1 System Design

The platform's architecture, depicted in Figure 1, is designed to support user-centered and device-independent functionality, in order to provide the technical feasibility for building a multitude of context-based services. The framework, intended for indoor environments, is based on a centralized configuration and achieves a high level of transparency in inter-device communication, context management and service deployment.

An RDF-based context model has been designed for use in the CG platform, aiming at addressing issues concerning semantic context representation,

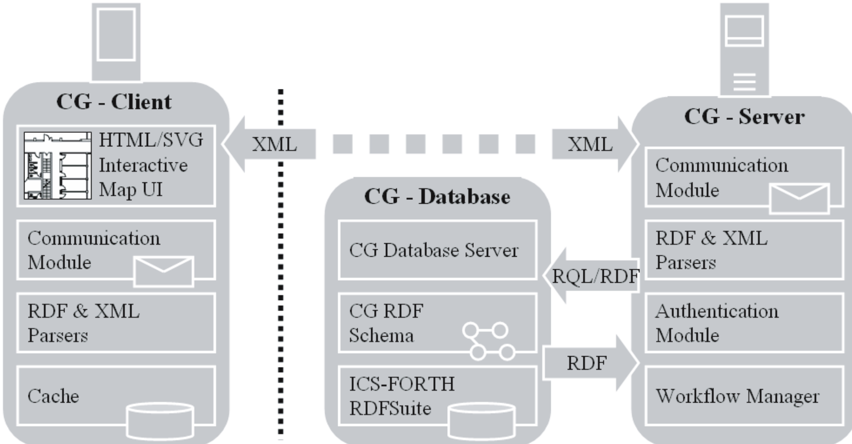

Fig. 1. The CG platform architecture

knowledge sharing and context classification. The ontology captures the relevant notions of physical and semantic entities situated in the system and provides a formal vocabulary for structuring location and other context-aware services. The generic CG RDF Schema follows the abstract design principles of other large-scale pervasive computing ontology frameworks, such as the SOUPA project [3], and defines the high-level concepts of Person, Geographical Space, Location Coordinates, CG-Client and Event, as well as the relations among them. The model captures static information (i.e., device ID for CG-Client devices) and dynamic contextual data (events) that have both spatial and temporal extensions, such as user login, change of location in the building and others.

A CG-Client module characterizes the devices with which a user can experience services provided by the CG platform. Different mobile devices can connect as CG-Clients in the system, such as PDAs or laptops, taking advantage of the flexible and portable Web-based model used for its development. CG-Clients connect with the CG-Server, a core component of our centralized architecture that acts as a middleware for processing, managing and transforming stored and inferred knowledge preserved in the CG-Database, while also offering the medium for synthesizing and publishing the desired services. The CG-Server queries the RDF database using the RQL query language.

2.2 CG Services

The system currently supports several services that enable users to obtain and superimpose context-related information on maps, while additional services are in the process of deployment on top of the framework, as will be described next. Our first concern was to implement a number of semantics-based services, such as personal/room information retrieval, room reservation and note keeping, for which a number of automatically generated RQL queries assist users in retrieving relevant RDF-annotated information. A newly published note, for instance, is semantically matched with the preferences of other users, in order to recognize whether its content is of interest to them.

Our present focus concerns the development of personalized, navigational and groupware services. In particular, we wish to enhance the platform with semantic profiles for purposes of context-awareness, privacy and customization of service provisioning depending on the device used. The use of a standardized semantic profile format will provide a unified and secure interface that is not going to be just a repository of static knowledge, but also a semantic directory of dynamic and context specific information. It will encapsulate personal rule definitions about users preferences, device specifications, service invocation and privacy policies. Moreover, we study semantics-based path planning algorithms that, in collaboration with the context ontology and the user's profile information, will assist users in navigating in the premises, considering not only ground plans and landmarks, but also the state of the user's context. Finally, we design and develop groupware services, such as group-calendars, information sharing applications, group notifications, bulletin boards etc., exploiting Semantic Web technologies to support the definition of associations with the context model and user profiles.

3 Complex, Dynamic and Uncertain Environments

Based on the identification of the requirements that a context-based platform, such as the one briefly described above, introduces and the experience gained with its deployment, we conduct a problem statement on the generic challenges of the Ambient Intelligence field and an evaluation of current research solutions.

3.1 Domain-Specific Challenges

The majority of services provided by our platform requires a certain degree of reasoning and plan management skills by the participating agents[1]. For instance, an agent wishing to print a document must deliberate on whether to use the slow inkjet printer located near the user or the faster high resolution laser printer of the adjacent room. It is our intention to enhance our platform with an increasing number of context dimensions and, therefore, the computational complexity of the planning tasks is a reality that we cannot afford to overlook. This challenge motivated us to consider mathematical methods for addressing aspects related to planning, actions and causality. Reasoning about action is a well-formalized and well-characterized subfield of AI, but has not yet been fully integrated in the ambient computing domain. When research on planning within the AI community established the *classical planning problem* for the development of techniques for agents to generate courses of action in order to reach a desirable world state,

[1] It is common trend in ambient computing system engineering to have devices participate in communication and negotiation tasks represented by some notion of intelligent software agents. In the rest of the paper we often interchange the terms device and agent, according to context.

it adopted a number of simplifying assumptions to delimit the domain. Some of them were very restrictive; the planning agent was considered omniscient, its actions deterministic, atomic and simultaneous, the environment static and the only source of change in the environment was the agent itself, while no other exogenous event occurred. Of course, these simplifications do not persist in realistic planning situations and must be relaxed or completely eliminated when adapting planners to Ambient Intelligence systems.

3.2 Problem Statement

While studying the Ambient Intelligence domain under the planning context, it is important to understand the class of problems that planning methods face and the challenges that these problems introduce. The ambient environment is an open and highly dynamic environment. Mobile devices connect and disconnect to the network, contributing services with durations that vary according to their expected presence in the environment and the availability of their resources. In fact, even actions, goals and sensor observations have a temporal dimension, whose duration may only be partially known in advance. The assumption of complete world knowledge can no longer persist; agents do not know *a priori* all other entities that are present at a specific time instance nor can they communicate directly with all of them. As a consequence, they have limited perception to acquire knowledge about the world they live in and have to generate plans preserving a level of uncertainty on both the state of the world, the available actions to achieve certain state of affairs and the outcome of those actions. Even the fact of committing different agents to certain tasks cannot be guaranteed to hold, since agents might disconnect before plan generation completes or new and more beneficial opportunities might arise, underscoring the importance of monitoring plan execution and replanning in opportunistic ways. The non-deterministic nature of the environment is emphasized by the recognition that not only agents, but also exogenous events occur, in unpredictable and concurrent manner, affecting the state of the world.

These observations essentially sketch a very challenging dynamic and uncertain environment with increased planning complexity. The question is whether current efforts can attack this problem and provide computationally efficient and scalable solutions under these conditions.

3.3 State-of-the-Art Approaches on Acting in Complex Domains

The AI research community has made significant progress on exploring techniques for extending the classical planning paradigm, relaxing some of its assumptions and much success has been achieved in defining theoretical foundations for modelling more complex domains. Reasoning about action and planning is a fundamental area of research within AI that studies the logical characterization of the concepts of action, change and planning of action sequences to accomplish a given task. Knowledge representation and reasoning has resulted in various formalisms, motivated by theorem proving, state space search and associated logic techniques.

The first effort towards an axiomatization of domains using action theories was accomplished by the STRIPS [4] action representation language that, despite being very restrictive in terms of expressivity, the simplicity of its semantics resulted in a quick and wide adoption. This simplicity justifies the fact that STRIPS now constitutes the core of PDDL (Planning Domain Description Language), the standardized syntax used in the International Planning Competitions, and is also utilized in a multitude of planning formulations.

The Situation Calculus [5], a second-order language designed for representing dynamically changing worlds, is the most influential formalism for reasoning about action and change. All changes are the result of named *actions* performed by agents, while a sequence of actions, denoting a possible world history, is represented by a first-order term called *situation*. Relations and functions whose truth values vary from situation to situation are called *relational* and *functional fluents*, respectively. Actions, situations and fluents are the main ingredients of the Situation Calculus formalism that give a complete treatment of reasoning about action, proposing solutions to many fundamental problems, such as the frame problem. The pure Situation Calculus has been extended to accommodate time [6] and concurrent actions [7], indeterminism and actions with uncertain effects [8], sensing and knowledge-producing actions [9], that is actions whose effects are to change a state of knowledge, ramifications [10] and other parameters.

Although the Situation Calculus suffices to solve the representational frame problem, concerning the efforts to specify the non-effects of actions, it fails to address the inferential frame problem for actually computing these non-effects. The Fluent Calculus [11] that extends the Situation Calculus by the notion of *state*, expressing the fluents that hold in a situation, proposes a more general solution and is more expressive under certain circumstances. For instance, apart from solutions to concurrent, continuous [12] and non-deterministic actions [13], the Fluent Calculus has recently been shown to provide the desired expressivity to solve the frame problem in the context of belief and belief revision [14]. Both calculi have well-established programming languages based on Prolog for cognitive agents to perform automated reasoning (Golog and Flux), their implementations though do not scale well to long action sequences and are restricted to small-scale applications with moderate computational effort.

The Event Calculus is also a widely adopted formalism for reasoning about action and change that addresses more naturally certain phenomena, such as continuous time, concurrent, partially-ordered and triggered events. A number of different dialects have been proposed that are summarized in [15] and [16]. The basic idea of the Event Calculus is to state that fluents are true at particular time-points if they have been initiated by an action occurrence at some earlier-time point and not terminated by another action occurrence in the meantime. The calculus supports default reasoning, i.e., reasoning in which a conclusion is reached based on limited information and later retracted when new information is available, using circumscription.

In addition to these calculi, a family of action theories have been developed that are independent of a specific axiomatization, by attempting formal validation methods for assessing correctness and ensuring that the encoding of any domain will yield correct results. Their semantics are based on the theory of causal explanation, which distinguishes between the claim that a formula is true and the stronger claim that there is a *cause* for it to be true, leading to "causal rules". Gelfond and Lifschitz [17] give an overview of such causal languages, such as action languages $\mathcal{A}$ and $\mathcal{C}$ [18]. The action language $\mathcal{C}+$ [19] that has evolved from $\mathcal{C}$, in an important recent formalism that provides a uniform model for supporting, in addition to conditionals and concurrent actions, also indirect effects and non-deterministic actions. Other similar languages are being proposed (i.e., G$\mathcal{C}+$ [20]) and, although this collection of languages had initially established results for a restricted class of domains, it is now characterized by its high expressivity, natural language-like syntax and clear formal semantics.

Furthermore, when discussing planning in dynamic, stochastic environments, we must not neglect decision-theoretic planning tools that extend the classical AI planning paradigm by modelling problems, in which actions have uncertain effects, agents have incomplete information about the world and solutions are of varying quality. Much recent research on decision-theoretic planning has explicitly adopted the Markov Decision Process (MDP) framework as an underlying model. MDPs have two general impediments that make AI researchers skeptical; how they can model a wide range of planning problems and if they can scale to solve planning problems of reasonable size. This is because the generality of the framework comes at a high price in terms of space for storing the transition matrices and policies, in terms of time required to generate a solution policy and in terms of ease of specification. To obviate the need for explicit state space and action enumeration, certain recent works focus on representing the dynamics of first-order MDPs using stochastic actions and objective functions of generalizations of the Situation Calculus [21] and the Fluent Calculus [22].

The above mentioned formalisms are prominent in handling certain aspects of the planning challenges given the demands raised by ambient computing. The literature is rich with notable related surveys and comparative analyses (i.e., [23,24,25,26]). Still, a common conclusion inferred by studying these formalisms and confirmed by different sources is that combining separate phenomena, such as non-determinism, natural actions or continuous change, in a unified model is no trivial task. It is a fact that most extensions to the problem have been investigated in isolation and combining co-existing models has proven to be very challenging [13]. Moreover, we lack a rich set of good heuristic techniques for generating effective contingent plans and it seems unlikely to find optimal solutions to non-trivial problems; new and dramatically different approaches are needed to deal with them [24]. This conclusion justifies the fact that, despite the rapid progress accomplished on reasoning about actions and planning, researchers have delayed applying theory to practice, when confronting the complexity of realistic domains. To our opinion, although Situation Calculus guided research in the field for years and still is the origin for novel ideas since many inspiring researchers work with it, its reasoning capabilities are limited to short action sequences, thus reducing its applicability for real-life large-scale implementations. We trust the Event Calculus to present the most complete solution for addressing key issues of commonsense reasoning for Ambient Intelligence. It is highly usable, comprehensive and handles inherently most of the important aspects of reasoning about actions, as shown in recent studies (i.e., [27]).

4 Distributed Planning and Multi-agent Coordination

4.1 Domain-Specific Challenges

Extending our platform with new services has made evident the importance of seamless collaboration between numerous devices that must work together to achieve common objectives. Even the more ordinary services like the management of a presentation in a meeting room may involve continuous cooperation between devices, as diverse as the room's projector, the lighting dimmer, the lecturer's laptop and many others. Complicated services require more sophisticated models of teamwork between devices, which differ in capabilities, characteristics and resource limitations. The establishment of common objectives among them entails a comprehensive understanding of the problem domain between participating entities and a confirmed desire to distribute their knowledge and to contribute their capabilities, in order to generate plans for achieving these objectives. Moreover, the assumption that the availability of devices is known beforehand is not valid in our platform, and, therefore, we have to come up with ways to arrange the formation of coalitions at execution time.

4.2 Problem Statement

The Ambient Intelligence domain is frequently characterized as a sensor and device-rich environment supported by software capable of fully leveraging its capacity and providing distributed services. A multitude of appliances, however diverse in capabilities and characteristics, needs to be seamlessly integrated into the users' everyday lives, placing them and their tasks on the center of attention. This scheme requires devices to coordinate their actions, cooperate in generating plans and collaborate during their execution.

If all agents in a system were omniscient of the goals, actions and interaction of their fellow community members and could also have unlimited computational resources, it would be possible to know exactly the present and future actions and intentions of each. Obviously, this in not the case in most real-world systems and in ambient computing structures, in particular. Despite the fact that our platform currently utilizes a centralized knowledge base, there are many situations where services would benefit by distributing reasoning tasks among different devices. A typical example is when a user's connectivity with the network is undermined due to coverage range or bandwidth restrictions, while he can still engage in ad hoc communication with other mobile devices in its vicinity, which contribute resources and services. This decentralized self-organizing infrastructure is a nontrivial challenge for the realization of the AmI vision.

4.3 State-of-the-Art Approaches on Teamwork in Cooperative Problem Solving

Coordination does not imply either cooperation or reciprocation; appliances in a smart space, though, are expected to collaborate in achieving common objectives, after a particular level of trustfulness has been established. Therefore, agents in AmI environments are designed to engage in Cooperative Distributed Planning [28], according to which they are endowed with shared objectives and representations, with the purpose to jointly develop and execute a plan in a coherent and effective manner.

Of the most prominent approaches for multi-agent action coordination is Durfee and Lesser's Partial Global Planning (PGP) framework [29] and its extension Generalized PGP [30]. According to this framework, each agent is not aware of the presence of other agents or their capabilities at the start of the plan generation process, but incrementally obtains a partial conception of their plans. Agents dynamically determine sets of long-term and short-term decisions for achieving their objectives, evaluating a number of predictions and ratings. In GPGP, coordination and task assignment are achieved by negotiation using a family of domain-independent coordination algorithms, while planning is based on a set of heuristics. The framework focuses on dynamically revising plans in cost-effective ways given an uncertain world, rather than optimizing plans for static and predictable environments and can lead to sub-optimal results with reduced costs. Despite its advantages, the framework is restricted to domains where temporary incoherence among agents is affordable as information about unanticipated changes to plans is propagated. Therefore, it is inappropriate for domains with irreversible actions where guarantees about coordination must be made prior to any execution [31]. In addition, PGP assumes explicit communication between agents in order to achieve coordination and is not scalable when agents' commitments to partial subgoals are not highly persistent.

In fact, this notion of commitment is essential when attempting to coordinate multi-agent interactions and has been given much importance in the Joint Intention model developed by Cohen and Levesque [32], influencing many works since. This model specifies how a group of agents can act together by sharing certain mental beliefs about the cooperative behavior and forming individual and joint commitments for pursuing goals. The Cooperative Problem Solving (CPS) abstract formal model of [33] is based on the Joint Intention theory and is the first comprehensive attempt to formalize intentions by making a distinction between commitments and conventions (a means for monitoring a commitment) in a mathematical framework. CPS describes a teamwork process evolving in 4 stages, namely recognition, team formation, plan formation, and team action. Although structured around some restricting assumptions, the framework provides principles, which lay a solid ground for designing models that face challenges of teamwork formation in more complicated systems.

Many researchers explore the multi-agent plan coordination problem by extending the partial-order, causal-link (POCL) definition of a plan described in [34]. According to this definition, a plan has temporal and causal partial order constraints on plan steps and as new steps are added (i.e., when loosely-coupled agents attempt to combine their plans) these constraints might get violated and inconsistencies or plan flaws might arise. There are works that investigate ways to transform inconsistent plans into consistent. In [35] a general plan-space search algorithm is described that repairs flaws by exploiting a hierarchical plan representation to reduce the complexity of the plan coordination problem. The same type of flaws, namely causal link threatening and parallel action interference, as well as redundant actions between agents that coordinate their plans, are handled in [36] using the propositional satisfiability approach. There, the authors have developed branch and bound algorithms for multi-agent coordination and cooperation scenarios, where actions have time extents and individual plans are iteratively expanded to an optimal, with respect to length, joint plan. A different approach is followed in [37] that casts the multi-agent plan coordination problem as a type of Distributed Constraint Optimization Problem and examines the efficiency of applying techniques from this domain for solving the problem, such as ADOPT, an algorithm that exploits local communication and parallel computations. Still, this problem is shown to be NP-Hard [38] and attempts to develop computationally-efficient algorithms that produce optimally-coordinated plans are limited, focusing mostly on restricted forms of it (see for example [39]).

Other efforts logically analyze the ability of agents to cooperate in executing complex actions for reaching certain states of affairs. In [40] a complete axiomatization for a dynamic logic that models agent capabilities in executing collective plans in environments that require concurrent actions of several agents is presented. Such formalisms, which are based on cooperation logics, such as Coalition Logic [41] and Coalition Logic for Propositional Control [42], represent important tools not only for verification purposes by proving that a desired goal of a system can be achieved, but also for explicitly capturing how a plan can be executed. Still, they require the designer to sacrifice expressivity, in order to handle the complexity of real-world dynamic and multidimensional problems.

As the previous discussion has shown, despite the well-studied tools and techniques in the repertoire of distributed AI, many of its accomplishments fail to function efficiently when applied to real-world conditions, a conclusion confirmed by many recent studies as well ([31,43,44]). However interesting the existing approaches on distributed planning and coalition formation may be, the common feeling is that they are not mature enough to guarantee successful teamwork in a pragmatic context-aware environment. It is anticipated that more comprehensive methods for coordination and distributed planning in open and dynamic environments must arise that will overcome issues mentioned above.

5 Plan Representation and Monitoring

5.1 Domain-Specific Challenges

The overall implementation of our context-aware platform was structured around a number of observations that specify the basic motivations of the domain, such as the desire to provide context-aware services to users that operate on a variety of mobile devices. In order for our architecture to support this scheme, we seek ways to represent the capabilities of these devices so that services can be adapted according to them. Profiles of devices must be flexible enough to capture both the complex actions that they can perform, but also decompositions of them to more low-level actions, so that planning agents can understand and combine these actions to distribute responsibilities during service execution. There are obstacles that we need to overcome for achieving this type of interactions in our framework, since we assume that the existence of visitors possessing partially unknown mobile devices is going to be a common situation for the system.

5.2 Problem Statement

In order for any Ambient Intelligence framework to support the challenges discussed so far, we also need to seek ways to represent device and service profiles, plans and goals in a manner that is mutually apprehended and correctly interpreted by all participating entities. Instead of exclusively focussing on improving our planners, other facets of the problem should also come into consideration that can refine the planning task. Autonomous agents need to recognize which planning problems and opportunities to consider in the first place. They need to be able to weight alternative incomplete plans and to decide among competing alternatives on execution time. They need to be able to function intelligently in an ambient environment in a way that comports with the inherent bounds on their computational resources. They need to share a common plan representation or a common ontology for describing their plans, goals and actions. And they need to be able to exchange freely their plans and describe their action behaviors, privacy policies and authenticity certificates, in terms of expressive planning languages.

5.3 State-of-the-Art Approaches on Plan Representation and Monitoring

Much of the research has been focusing on representing plans using some form of abstraction and decomposition, with hierarchical task networks (HTN) [45] being the most notable approach. HTN plan representations allow a distributed planning agent to successively refine its planning decisions as it learns more about other agents' plans. This abstraction-based methodology has been widely used in a variety of domains, and attempts have also been presented to adapt it to the Ambient Intelligence domain (see [46] for an illustrative example). An interesting application of HTN-like action representation is demonstrated in [47], where the intention is to model joint complex actions of multi-agent partially ordered plans under temporal and precedence constraints in dynamic environments. Compared with classical planners, HTN planners have more sophisticated knowledge representation and reasoning capabilities and are more expressive, because they can even be used to encode undecidable problems. Logics-based formalisms are evenly expressive and is hard to say which formalism is more effective. Both types are suitable in different situations, as illustrated in [23], and combining them is a useful topic for future research.

A rather similar approach, originally proposed in [48], is representing procedural knowledge in a library of skeletal plans. Skeletal plans are plan schemata at various levels of detail, which capture the essence of the procedure, but leave room for execution-time flexibility in the achievement of particular goals and are also being tested in real-world systems, such as [49]. Although being reusable in various contexts and able to capture in restricted form both complex and primitive actions, skeletal plan representations are domain-specific and it is questionable if they can confront the openness of ubiquitous systems.

In fact, there is a significant gap in the field for methods and tools capable of representing both complex and primitive actions in an expressive manner and with formal semantics, with only few exceptions being present (i.e., McIlraith et. al in [50,51]). Attention should be given in representations that allow a system to capture and share the capabilities of the diverse devices that operate in it and permit them to distribute common plan configurations.

Creating a conflict-free plan for a multi-agent system in a dynamic and unpredictable environment that will remain conflict free during execution, is a laborious task. As agents execute their plans, they monitor their execution by making partial observations that are used to detect plan deviations. Part of related work concentrates on how to explain such erroneous actions by applying *Model-based Diagnosis* [52]. This technique is used to infer abnormalities of internal components of a given system from its input-output behavior and studies, such as [53] and [54], adapt and extend Model-based Diagnosis to multi-agent planning systems. Each component has several health modes; a normal mode and several fault modes. Once the health mode for each component is specified, the behavior of the total system is defined and the output can be inferred. In fact, since in a plan several instances of the same action might occur, an error occurring in one instance might be used to predict the occurrence of the same error in an action instance to be executed later on. The result of Model-based Diagnosis is a suitable assignment of health modes to the components, called a *diagnosis*, such that the actually observed output is consistent with this health mode qualification or can be *explained* by this qualification. Using this technique we can adjust the plan execution within the margins of the plan by determining events that cause constraint violation, thus avoiding replanning the tasks. In [53], in particular, actions are described in an object-oriented manner, rather than the traditional state-based approach. This view of actions as components, and preconditions and effects of actions as objects, allows the plan itself to be viewed as the structural description of a system.

6 Conclusions and Future Lines of Research

As research in AI is oriented towards addressing real-world problems, exploring relaxations of the simplifying assumptions, the efficiency and viability of existing methods will be tested. We have attempted an assessment of state-of-the-art approaches and principles to determine where there is room for improvement and where a different research perspective is necessary, in the context of ambient computing systems. It is indeed evident that there are still serious obstacles that have not been hurdled. Nevertheless, in this paper we argue that regression steps are not justified. To pursue the vision of intelligent environments we need to rethink about the capabilities of our existing tools and focus on those fields that seem more prominent in meeting the new challenges and ideas that the main tenet of Ambient Intelligence introduces. As a starting point, two abstract research directions are presented next that, although being moderately novel in classical AI, are highly relevant in the cross section of Ambient Intelligence and AI research. Departing from the typical planning picture, we attempt to contrast the state-of-the-art in symbolic planning with requirements imposed by ambient computing scenarios, citing also initial attempts that follow these directions.

6.1 The Continual Planning Paradigm

Our remarks concerning a highly dynamic, uncertain environment, where planning and reasoning tasks are performed by devices that may have limited resources available and where knowledge concerning world state is dynamically acquired, have led us to abandon traditional multi-agent planning approaches. As argued in [25], autonomous agents in dynamic environments need to be able to form incomplete plans now, adding detail later and deciding upon the amount of detail to include at each time instance. A new paradigm has been proposed that interleaves plan generation and execution and adopts strategies that allow agents not to plan too much detail too far into the future so that they can confront with the diversity of upcoming events. This technique of performing a tightly coupled control loop between plan generation and execution, first proposed in [28], is called *Continual Planning* and requires an iterative collaboration of the tasks of environment monitoring, plan adaptation and replanning.

More specifically, [28] suggests that an agent should plan continually when aspects of the world can change dynamically beyond the control of the agent, when aspects of the world are revealed incrementally, when time pressures require execution to begin before a complete plan can be generated or when objectives can evolve over time. Continual planning suggests having a single plan and continuously monitor its execution, in contrast to the traditional approach of handling uncertainty by planning for all the contingencies that might arise. Since this approach views planning as an ongoing, dynamic process it can benefit the system not only by reacting to unhealthy situations, but also by adapting to opportunities to improve the plan, repairing it when necessary. Only recently have frameworks started adapting to this real-time planning and execution approach ([47,55]), while the need to explore this direction of research in ambient applications has also been underscored by many recent reports (i.e., [43,44]).

6.2 Context-Aware Planning Paradigm

We have already stressed the fact that in order to attain the goal of planning in dynamic and partially known environments we have to endow agents with cognitive capabilities. When agents weight alternative subplans to reach a certain state of affairs or even when they consider the most profitable next step to a plan, it is often useful to interleave reasoning in the process. Until recently, most approaches only modelled conventional aspects, such as precedence constraints or plan length, relying on replanning techniques to deal with plan failure, when new constraints were added. Nonetheless, the impact of planning can be increased if we consider a richer set of aspects that are truly related to the domain we model. Temporal constraints, such as action durations and temporal instantiation of action are a common alternative for accomplishing scheduling tasks in real time, while a number of studies deliberate on action durations and deadlines of goals. This is a more pragmatic approach requiring reasoning about whether or not goals can be achieved by their deadlines and leads in associating priority values to goals and to the actions which contribute to those goals.

Reasoning over a richer collection of metrics and focusing on the most relevant ones might prove an essential aid in improving planning, both from the computational standpoint, by reducing the search space, and from the perspective of efficiency, since more accurate plans will be generated. Indeed, we believe that the Ambient Intelligence domain raises many opportunities for exploiting reasoning to influence the efficiency of planning. The most important leverage is to exploit the context in which an agent is situated, affecting its plan-generation capabilities. Context awareness has become a hot topic in recent computing research and reasoning on context can be an important source of information both for allowing agents to determine the current state of the world and for restricting possible future states, enabling them to constraint and better prioritize their goals. Moreover, it is anticipated that privacy and trust issues are going to be of utmost importance in designing Ambient Intelligence systems, strengthening the need to include trustfulness values, along with importance values, to actions and subplans. This will enrich the plan evaluation process by allowing users to prioritize goals not only by their significance and success probability, but also by taking into account preference and privacy profiles. We expect this new paradigm of *context-aware planning* to play a determinant role in ambient IS, guiding computationally efficient planning solutions in realistic conditions. Studies, such as [55] and [56] adopt this new trend, incorporating resource allocation and user preference metrics in plan refinement and execution.

References

1. Rogers, Y.: Moving on from Weiser's Vision of Calm Computing: Engaging UbiComp Experiences. In: Dourish, P., Friday, A. (eds.) UbiComp 2006. LNCS, vol. 4206, pp. 404–421. Springer, Heidelberg (2006)
2. Patkos, T., Bikakis, A., Antoniou, G., Papadopouli, M., Plexousakis, D.: A Semantics-based Framework for Context-Aware Services: Lessons Learned and Challenges. In: UIC 2007, pp. 839–848 (2007)
3. Chen, H., Finin, T., Joshi, A.: The SOUPA Ontology for Pervasive Computing. In: Tamma, V., Stephen Cranefield, T.F., Willmott, S. (eds.) Ontologies for Agents: Theory and Experiences, Springer, Heidelberg (2005)
4. Fikes, R., Nilsson, N.J.: STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving. In: IJCAI, pp. 608–620 (1971)

5. Levesque, H., Pirri, F., Reiter, R.: Foundations for the Situation Calculus. In: Linkoping Electronic Articles in Computer and Information Science, vol. 3 (1998)
6. Pinto, J.A.: Temporal Reasoning in the Situation Calculus. PhD thesis (1994)
7. Reiter, R.: Natural Actions, Concurrency and Continuous Time in the Situation Calculus. In: KR 1996. Principles of Knowledge Representation and Reasoning: Proceedings of the Fifth International Conference, Cambridge, Massachusetts, USA, pp. 2–13 (November 1996)
8. Pinto, J., Sernadas, A., Sernadas, C., Mateus, P.: Non-determinism and Uncertainty in the Situation Calculus. Int. J. Uncertain. Fuzziness Knowl.-Based Syst. 8(2), 127–149 (2000)
9. Scherl, R.B., Levesque, H.J.: Knowledge, Action, and the Frame Problem. Artif. Intell. 144(1-2), 1–39 (2003)
10. Papadakis, N., Plexousakis, D.: Actions with Duration and Constraints: The Ramification Problem in Temporal Databases. In: ICTAI 2002. Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence, pp. 83–90. IEEE Computer Society, Washington, DC, USA (2002)
11. Thielscher, M.: From Situation Calculus to Fluent Calculus: State Update Axioms as a Solution to the Inferential Frame Problem. Artif. Intell. 111(1-2), 277–299 (1999)
12. Thielscher, M.: The Concurrent, Continuous Fluent Calculus. Studia Logica 67(3), 315–331 (2001)
13. Thielscher, M.: Modeling Actions with Ramifications in Nondeterministic, Concurrent, and Continuous Domains - and a Case Study. In: Proceedings of the 17th National Conference on Artificial Intelligence and 12th Conference on Innovative Applications of Artificial Intelligence, AAAI Press / The MIT Press, pp. 497–502 (2000)
14. Scherl, R.B.: Action, Belief Change and the Frame Problem: A Fluent Calculus Approach. In: Proceedings of the Sixth Workshop on Nonmonotonic Reasoning, Action, and Change at IJCAI 2005, Edinburgh, Scotland (August 2005)
15. Shanahan, M.: The Event Calculus Explained. In: Veloso, M.M., Wooldridge, M.J. (eds.) Artificial Intelligence Today. LNCS (LNAI), vol. 1600, pp. 409–431. Springer, Heidelberg (1999)
16. Miller, R., Shanahan, M.: Some Alternative Formulations of the Event Calculus. In: Computational Logic: Logic Programming and Beyond, Essays in Honour of Robert A. Kowalski, Part II, pp. 452–490. Springer, London, UK (2002)
17. Gelfond, M., Lifschitz, V.: Action Languages. Electronic Transactions on AI 3 (1998)
18. Giunchiglia, E., Lifschitz, V.: An Action Language Based on Causal Explanation: Preliminary Report. In: AAAI 1998/IAAI 1998: Proceedings of the Fifteenth National/Tenth Conference on Artificial intelligence/Innovative Applications of Artificial Intelligence, American Association for Artificial Intelligence, pp. 623–630 (1998)
19. Giunchiglia, E., Lee, J., Lifschitz, V., McCain, N., Turner, H.: Nonmonotonic Causal Theories. Artif. Intell. 153(1-2), 49–104 (2004)
20. Finzi, A., Lukasiewicz, T.: Game-Theoretic Reasoning About Actions in Nonmonotonic Causal Theories. In: Baral, C., Greco, G., Leone, N., Terracina, G. (eds.) LPNMR 2005. LNCS (LNAI), vol. 3662, pp. 185–197. Springer, Heidelberg (2005)
21. Boutilier, C., Reiter, R., Price, B.: Symbolic Dynamic Programming for First-Order MDPs. In: IJCAI 2001. Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence, Seattle, Washington, USA, pp. 690–700 (2001)

22. Großmann, A., Hölldobler, S., Skvortsova, O.: Symbolic Dynamic Programming within the Fluent Calculus. In: Ishii, N. (ed.) Proceedings of the IASTED International Conference on Artificial and Computational Intelligence, pp. 378–383. ACTA Press, Tokyo, Japan (2002)
23. Ghallab, M., Nau, D., Traverso, P.: Automated Planning: Theory and Practice. Morgan Kaufmann, San Francisco (2004)
24. Bresina, J., Dearden, R., Meuleau, N., Ramkrishnan, S., Smith, D., Washington, R.: Planning under Continuous Time and Resource Uncertainty: A Challenge for AI. In: UAI 2002. Proceedings of the 18th Annual Conference on Uncertainty in Artificial Intelligence, pp. 77–84. Morgan Kaufmann, San Francisco, CA (2002)
25. Pollack, M., Horty, J.F.: There's More to Life than Making Plans. The AI Magazine 20(4), 71–84 (1999)
26. Blythe, J.: Decision-Theoretic Planning. AI Magazine 20(2), 37–54 (1999)
27. Mueller, E.: Automating Commonsense Reasoning Using the Event Calculus. Communications of the ACM (in press, 2007)
28. des Jardins, M., Durfee, E.H., Ortiz Jr., C.L., Wolverton, M.: A Survey of Research in Distributed, Continual Planning. AI Magazine 20(4), 13–22 (1999)
29. Durfee, E.H., Lesser, V.R.: Partial Global Planning: A Coordination Framework for Distributed Hypothesis Formation. IEEE Transactions on Systems, Man, and Cybernetics 21(5), 1167–1183 (1991)
30. Lesser, V.R.: Evolution of the GPGP/TÆMS Domain-Independent Coordination Framework. In: AAMAS 2002. Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 1–2. ACM Press, New York (2002)
31. Durfee, E.H.: Distributed Problem Solving and Planning. In: Luck, M., Mařík, V., Štěpánková, O., Trappl, R. (eds.) ACAI 2001 and EASSS 2001. LNCS (LNAI), vol. 2086, pp. 118–149. Springer, London, UK (2001)
32. Cohen, P.R., Levesque, H.J.: Teamwork. Technical Report 504, Menlo Park, CA (1991)
33. Wooldridge, M., Jennings, N.R.: Towards a Theory of Cooperative Problem Solving. In: Perram, J., Müller, J.P. (eds.) MAAMAW 1994. LNCS, vol. 1069, pp. 40–53. Springer, London, UK (1996)
34. Weld, D.S.: An Introduction to Least Commitment Planning. AI Magazine 15(4), 27–61 (1994)
35. Cox, J.S., Durfee, E.H.: An Efficient Algorithm for Multiagent Plan Coordination. In: AAMAS 2005. Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 828–835. ACM Press, New York (2005)
36. Dimopoulos, Y., Moraitis, P.: Multi-agent Coordination and Cooperation Through Classical Planning. In: IAT 2006. Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT 2006 Main Conference Proceedings) (IAT 2006), pp. 398–402. IEEE Computer Society, Washington, DC, USA (2006)
37. Cox, J.S., Durfee, E.H., Bartold, T.: A Distributed Framework for Solving the Multiagent Plan Coordination Problem. In: AAMAS 2005. Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 821–827. ACM Press, New York (2005)
38. Yang, Q.: Intelligent Planning: A Decomposition and Abstraction Based Approach. Springer, London (1997)

39. Valk, J.M., de Weerdt, M.M., Witteveen, C.: Algorithms for Coordination in Multi-Agent Planning. In: Vlahavas, I., Vrakas, D. (eds.) Intelligent Techniques for Planning, pp. 194–224. Idea Group Inc., London (2005)
40. Sauro, L., Gerbrandy, J., van der Hoek, W., Wooldridge, M.: Reasoning About Action and Cooperation. In: AAMAS 2006. Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 185–192. ACM Press, New York, USA (2006)
41. Pauly, M.: A Modal Logic for Coalitional Power in Games. Journal of Logic and Computation 12, 146–166 (2002)
42. van der Hoek, W., Wooldridge, M.: On the Logic of Cooperation and Propositional Control. Artif. Intell. 164(1-2), 81–119 (2005)
43. de Weerdt, M., ter Mors, A., Witteveen, C.: Multi-agent Planning: An Introduction to Planning and Coordination. In: Handouts of the European Agent Summer School, pp. 1–32 (2005)
44. Chen, H., Finin, T.: Beyond Distributed AI, Agent Teamwork in Ubiquitous Computing. In: AAMAS 2002. Workshop on Ubiquitous Agents on Embedded, Wearable, and Mobile Devices (July 2002)
45. Erol, K., Hendler, J., Nau, D.S.: HTN Planning: Complexity and Expressivity. In: AAAI 1994. Proceedings of the Twelfth National Conference on Artificial Intelligence, Menlo Park, CA, USA, American Association for Artificial Intelligence, vol. 2, pp. 1123–1128 (1994)
46. Amigoni, F., Gatti, N., Pinciroli, C., Roveri, M.: What Planner for Ambient Intelligence Applications? IEEE Transactions on Systems, Man, and Cybernetics, Part A 35(1), 7–21 (2005)
47. Hadad, M., Kraus, S., Gal, Y., Lin, R.: Temporal Reasoning for a Collaborative Planning Agent in a Dynamic Environment. Annals of Mathematics and Artificial Intelligence 37(4), 331–379 (2003)
48. Friedland, P., Iwasaki, Y.: The Concept and Implementation of Skeletal Plans. J. Autom. Reasoning 1(2), 161–208 (1985)
49. Miksch, S., Seyfang, A.: Continual Planning with Time-Oriented, Skeletal Plans. In: ECAI, pp. 511–515 (2000)
50. Baier, J.A., McIlraith, S.A.: On Planning with Programs That Sense. In: KR, pp. 492–502 (2006)
51. McIlraith, S., Fadel, R.: Planning with Complex Actions. In: NMR 2002. Proc. International Workshop on Non-Monotonic Reasoning, pp. 356–364 (2002)
52. Console, L., Torasso, P.: A Spectrum of Logical Definitions of Model-Based Diagnosis. Comput. Intell. 7(3), 133–141 (1991)
53. Witteveen, C., Roos, N., van der Krogt, R., de Weerdt, M.: Diagnosis of Single and Multi-agent Plans. In: AAMAS 2005. Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 805–812. ACM Press, New York (2005)
54. Kalech, M., Kaminka, G.A.: On the Design of Social Diagnosis Algorithms for Multi-agent Teams. In: IJCAI, pp. 370–375 (2003)
55. Coddington, A., Luck, M.: A Motivation-based Planning and Execution Framework. International Journal on Artificial Intelligence Tools 13(1), 5–25 (2004)
56. Look, G., Peters, S., Shrobe, H.: Plan-driven Ubiquitous Computing. In: Dey, A.K., Schmidt, A., McCarthy, J.F. (eds.) UbiComp 2003. LNCS, vol. 2864, pp. 66–73. Springer, Heidelberg (2003)