European Network of Excellence in AI Planning

Technological Roadmap

on

AI Planning and Scheduling

Edited by

Susanne Biundo, Ruth Aylett, Michael Beetz,
Daniel Borrajo, Amedeo Cesta, Tim Grant,
Lee McCluskey, Alfredo Milani, Gérard Verfaillie

http://www.planet-noe.org

PLANET, the European Network of Excellence in AI Planning

Technological Roadmap
on
AI Planning and Scheduling

Susanne Biundo, Ruth Aylett, Michael Beetz,
Daniel Borrajo, Amedeo Cesta, Tim Grant,
Lee McCluskey, Alfredo Milani, Gèrard
Verfaillie

(Editors)

Project Period: 1.08.2001 – 31.10.2003

Contents

Preface

At the beginning of the new century, Information Technologies had become basic and indispensable constituents of the production and preparation processes for all kinds of goods and services and with that are largely influencing both the working and private life of nearly every citizen. This development will continue and even further grow with the continually increasing use of the Internet in production, business, science, education, and everyday societal and private undertaking.

Recent years have shown, however, that a dramatic enhancement of *software capabilities* is required, when aiming to continuously provide advanced and competitive products and services in all these fast developing sectors. It includes the development of *intelligent systems* – systems that are more autonomous, flexible, and robust than today's conventional software.

Intelligent Planning and Scheduling is a key enabling technology for intelligent systems. It has been developed and matured over the last three decades and has successfully been employed for a variety of applications in commerce, industry, education, medicine, public transport, defense, and government.

This document reviews the state-of-the-art in key application and technical areas of Intelligent Planning and Scheduling. It identifies the most important research, development, and technology transfer efforts required in the coming 3 to 10 years and shows the way forward to meet these challenges in the short-, medium- and longer-term future.

The roadmap has been developed under the regime of PLANET – the *European Network of Excellence in AI Planning*. This network, established by the European Commission in 1998, is the co-ordinating framework for research, development, and technology transfer in the field of Intelligent Planning and Scheduling in Europe.

A large number of people have contributed to this document including the members of PLANET non-European international experts, and a number of independent expert peer reviewers. All of them are acknowledged in a separate section of this document.

Intelligent Planning and Scheduling is a far-reaching technology. Accepting the challenges and progressing along the directions pointed out in this roadmap will enable a new generation of intelligent application systems in a wide variety of industrial, commercial, public, and private sectors.

Susanne Biundo

Director of PLANET

Executive Summary

This roadmap document is an integrated collection of the individual roadmaps developed in the Technical Coordination Units of PLANET and edited by the respective TCU chairs.

Intelligent Manufacturing Systems

Chapter 2 gives an overview of the application of AI Planning and Scheduling technologies to industrial –primarily manufacturing– domains. It gives some basic definitions of terms such as AI planning and scheduling, and the same words as used in intelligent manufacturing research. It considers the current state of the art in AI P&S and in Intelligent Manufacturing as it relates to planning and scheduling. An overview of applications to which AI P&S has been applied in manufacturing is given, and the potential contribution of the technology is discussed.

The requirements of manufacturing are also analyzed and the issues involved in meeting these requirements with AI P&S technology are discussed. The new areas that need to be tackled if planning and scheduling technology is to be successfully taken up in manufacturing domains are examined and the community-building actions that should take place alongside this research programme is also evaluated. The Roadmap attempts to evaluate the obstacles which have impeded the penetration of PLANET technologies into manufacturing applications as well as to examine the ways in which those obstacles might be eroded. Finally, references are given for further information.

Ruth Aylett

Workflow Management

Chapter 3 presents an R&D Roadmap for AI Planning and Scheduling applied to business process management (BPM) produced by the Workflow Management Technical Coordination Unit of PLANET. The purpose of the Workflow Management Roadmap is to coordinate R&D by establishing end-user requirements on short medium and long time scales and proposing research and technology transfer goals and activities that will enable the requirements to be satisfied. The current version is only a first step towards such a Roadmap, which in any case should be a living document updated regularly. BPM and AI P&S are two disciplines with many parallels, but which have largely been pursued by disjoint communities. A necessary precursor to producing a Roadmap is to align the two disciplines so that specialists in each can understand each other. One of the main achievements to date has been to develop an understanding of how the "world view", vocabulary, challenges, etc. of

Business Process / Workflow Management relate to AI Planning and Scheduling. This has been possible because of the active participation of a number of workflow and process management experts from end-user organizations and consultancy companies.

Currently, there is a growing interest in the application of Artificial Intelligence Planning and Scheduling techniques to real world problems. We have recently seen impressive applications of AI P&S in space, robot, elevator control, military missions planning, etc. However, there are still many open tasks that can be (semi-)automated using AI P&S technology. One of such tasks is the production, and execution of models of organizations (workflow management). Here, we will describe what AI P&S techniques and models could be applied to workflow management and what actions can be carried out in order to achieve it.

Daniel Borrajo

Planning and Scheduling for the Web

Chapter 4 contains the Roadmap of the Technical Coordination Unit on Planning and Scheduling for the Web, it summarises results from Web TCU initiatives and events as well as discussions and contributions from TCU members.

Defining a roadmap for Planning and Scheduling for the Web represents a particular challenge because the target industry for technology transfer, i.e. the web industry, is not long established, not well defined, and it is moving and reshaping itself very fast. On the other hand many sub-areas in this field can benefit from Artificial Intelligence Planning and Scheduling technologies, and, in some cases, technology transfer has already started. For these reasons this document will consider and explore a set of topics, guidelines and objectives, which will need to be updated and deeply revised in short term as soon as new challenges, hot topics, requirements and issues emerge from the web and IT industry.

Alfredo Milani

Robot Planning

The next generations of autonomous robots will be characterized by being more general than the current one – in at least three respects. First, the robots will be able to successfully carry out multiple, diverse, and possibly interfering tasks in changing and partly unknown environments. Second, they will be able to improve their performance by autonomously adapting their control software for the kinds of tasks they are given and the environments they are to operate in. Third, they will be able to perform novel tasks without learning the achievement of these tasks in long and tedious learning sessions.

As a research community, we believe that these aspects of generality cannot be achieved without the robots being capable of planning their course of action based on foresight and without them being able to autonomously learn better control routines. To achieve more generality in these respects, parts of the control programs, called the robot plans, are to be represented explicitly, such that the robot can reason about them and revise them. This approach to autonomous robot control is called robot

planning or –more generally– plan-based control. The objective is to use plans and the respective reasoning mechanisms for them as resources for increasing the generality and the performance of robot control programs.

In this Technological Roadmap we show and analyze in Chapter 5 the potential for impact of robot planning on autonomous robot and agent control. Our goal is to show appropriate ways of how this potential can be realized, and lay down what the research community, application developers, and funding agencies can do to stimulate and accelerate scientific and technological breakthroughs. We propose a set of challenge application scenarios and a technological milestones for near-term, middle-term, and long-term research and technology development projects that must be achieved to control autonomous robots over extended periods of time, in natural and unmodified human environments, and to perform complex jobs with them.

Michael Beetz

Aerospace Applications

The purpose of the Aerospace Applications Roadmap is to identify the AI P&S research and technology transfer efforts needed in the coming three to ten years specifically to meet the needs of the aerospace domain.

The approach taken in Chapter 6 is to establish the current state-of-the-art and the desired future state of IP&S technology to meet aerospace needs. The roadmap is then a feasible path from the current state to the desired future state. Feasibility is based on the contributors' judgement of how quickly AI P&S research and development could deliver technology mature enough to apply, assuming that current levels of R&D funding are maintained.

The AA TCU's approach is partly formal and partly market-driven. One of the authors with extensive domain experience (Tim Grant) performed a formal object-oriented analysis to identify planning and scheduling applications in the aerospace domain. Selected representatives of aerospace organisations (e.g. the potential IP&S user community) were invited to review the results of the domain analysis, to confirm the applications, to provide the domain-specific names and characteristics of the confirmed applications, and to identify their existing applications and future needs for P&S technologies. The needs were matched with a formal survey of the current and foreseen developments in IP&S technology to derive the Aerospace Applications Road-Map.

Amedeo Cesta & Tim Grant

Knowledge Engineering

The Knowledge Engineering for Planning Technical Co-ordination Unit (KE TCU) of PLANET has been active for the past 5 years. It aims to identify and explore the current major problems involved in developing knowledge-based planning systems, and to examine the potential of future developments (such as the Semantic Web) to contribute to solutions of these problems. We also aim to synthesize related research and techniques from work in more general research areas into a relevant contribution to this area.

To achieve these aims the TCU has created and is maintaining the Roadmap in Chapter 7 which summarizes past work, and includes a collection of activities and problems that need to be tackled for the future. The Roadmap distinguishes Planning from the more general field of Knowledge-Based Systems in that the knowledge elicited is largely knowledge about actions, and how objects are effected by actions. Also, the ultimate use of the planning domain model is to be part of a system involved in the synthetic task of plan construction (in contrast to classification or diagnosis tasks common in KBS). Our Roadmap examines topics inherited from these areas, and attempts to point out where we can learn from past work; it also contains other related research areas such as Machine Learning, KE Tools and Formal Methods.

Looking towards the future, we see the areas of knowledge sharing through the use of ontologies, and the development of the Semantic Net as very important to support KE in Planning. As more and more planning technology finds its way into applications, knowledge engineering issues are recognized as crucial to an application's success. The KE TCU therefore covers fundamental issues which are relevant to all of the application-oriented TCUs.

Lee McCluskey

On-line Planning and Scheduling

This PLANET TCU goes on with the predceding TCU on *Dynamic Scheduling*, which has been extended to AI Planning and Scheduling. Differently from other PLANET TCUs which are dedicated to some specific application areas, like robot planning, intelligent manufacturing, or planning and scheduling for the web, this TCU is not application-oriented, but rather methodology-oriented. Its objective is to study all the consequences on AI Planning and Scheduling methods and tools of the online setting which is present in numerous applications. Chapter 8 is dedicated to this TCU.

It started from the observation that most of the AI Planning and Scheduling methods and tools have been developed for an use in an offline setting (planning and scheduling phase followed by the execution phase), although most of the applications appear in a very different online setting (planning and scheduling phase and execution phase going off concurrently).

Following this observation, it appeared quickly that dealing with an online setting is not a simple question of adaptation of the existing methods and tools and that it implies at least to revisit, but sometimes to rethink and redesign these methods and tools, in order to be able to meet the requirements of a online setting.

Gèrard Verfaillie

1. Introduction

1.1. Intelligent planning and scheduling

Artificial Intelligence Planning and Scheduling is a key enabling technology for intelligent systems. It provides the potential to considerably increase the autonomy, flexibility, and robustness of a wide variety of application systems. These include systems to control autonomous virtual and physical agents, systems for the design and monitoring of production, management, and business processes, and systems supporting web-based services and electronic commerce.

Planning and Scheduling are the processes by which an organization or an agent designs a course of actions prior to performing them. *Planning* focuses on selecting suitable actions and renewable resources (such as machines, people, vehicles, tools, and facilities) and on ordering them in an appropriate sequence so as to achieve some objective. *Scheduling* focuses on assigning time and consumable resources (such as money, fuel, and raw materials) to the resulting plans in order to meet defined objectives and constraints.

Intelligent Planning and Scheduling (IP&S) technologies are distinguished from conventional planning and scheduling techniques, like PERT, CPM, dynamic programming, and project network planning, by the explicit use of knowledge specific to the agent or organization and its field of operations. This *domain knowledge* includes knowledge of the resources and other objects of the domain, their attributes and interrelationships, and knowledge about actions, possible states, state transitions, and related constraints. Intelligent plans and schedules are *knowledge-rich*. The preparation of intelligent plans and schedules is known as *generation*, and their use is known as *execution*.

The explicit representation and use of domain knowledge distinguishing IP&S technologies can be exploited by suitably designed software to

- speed up plan/schedule generation;

- produce high-quality plans/schedules according to various criteria;

- guarantee correctness and safety of plans/schedules;

- perform interactive plan/schedule generation and execution;

- provide substantial information about plans/schedules to users.

Functions supported by IP&S software include the acquisition, engineering, and validation of domain knowledge, the generation, evaluation, animation, and execution of knowledge-rich plans and schedules, and the recognition of plans and schedules of other entities from their observed behavior.

1.2.   Application perspectives and benefits

In recent years, Intelligent Planning and Scheduling have surpassed the effectiveness of the long-dominant operations research techniques for solving a large variety of planning and scheduling problems. This allows IP&S users to

- tackle problems orders of magnitude larger;

- solve problems orders of magnitude more quickly;

- find significantly better solutions.

Through these techniques IP&S has allowed businesses and organizations to cut costs considerably while at the same time increase their ability to handle the dynamic nature and complexity of today's business environments. Furthermore, IP&S techniques provide a framework for controlling complex (autonomous) systems and agents. They are able to task individuals/systems, monitor their progress, and re-plan or re-schedule where new issues arise.

At present, IP&S technologies are providing great benefits for real-world operational applications in commerce, industry, education, medicine, public transport, defense, and government. Example users of IP&S technologies are Boeing (aircraft manufacturing), British Telecom (mobile workforce scheduling), the European Space Agency (spacecraft mission planning), NASA (autonomous spacecraft operations), Electric Boat (submarine building), and military organizations.

Proven benefits of IP&S technologies include 15% to 20% reduction in time needed to build an aircraft (equivalent to savings in the order of 1 million USD), 30% reduction in overtime and sub-contracting, and a shortening of planning time by factors of 40 to 100.
During the 1991 Gulf War, IP&S technologies speeded up the logistical build-up of the allied military forces by several months. The savings paid for all US research and development into Artificial Intelligence over the preceding 40 years.

1.3.   The PLANET network of excellence

The *European Network of Excellence in AI Planning* (PLANET) is the co-ordinating framework for research, development, and technology transfer in the field of AI Planning and Scheduling in Europe. Established in 1998, the network is funded by the European Union under the IST programme until late 2003.
The overall objectives of PLANET are to

- increase the awareness of the technology and promote pan-European and international collaboration;

- promote the orientation of research more closely towards application requirements;

- provide an internationally recognized forum for discussion and exchange in the area of Intelligent Planning and Scheduling;

- support high-quality training and teaching in the field.

PLANET undertakes various activities to achieve these objectives. They include

- maintaining a comprehensive information and communication infrastructure with dedicated Web-sites, online repositories, and mailing-list;

- organizing scientific workshops and industrial information days;

- supporting the exchange of people in academia and industry;

- developing a planning and scheduling curriculum;

- organizing the *PLANET International Summer Schools on IP&S*;

- publishing the *PLANET News* magazine;

- developing the IP&S technological roadmap.

PLANET maintains a number of *Technical Co-ordination Units*. They are devoted to methodologies and application areas that are key to substantial future progress in research, development, exploitation, and industrial uptake of Intelligent Planning and Scheduling technology. These are:

- On-line Planning and Scheduling

- Knowledge Engineering for Planning and Scheduling

- Intelligent Manufacturing

- Workflow Management

- Robot Planning

- Planning and Scheduling for the Web

- Aerospace Applications

The PLANET network is co-ordinated by Prof. Susanne Biundo of the University of Ulm. It has 68 member sites – leading universities, research centers, and industrial companies – in 18 countries and associates more than 200 researchers and practitioners in the field of AI Planning and Scheduling.

1.4. The IP&S technological roadmap

Over the past 20 years Intelligent Planning and Scheduling matured to a key enabling technology for next generation software systems. By making explicit use of domain knowledge IP&S systems can reason about causal dependencies between tasks or activities and manage to cope with uncertainty and dynamicity inherent in many application domains. With that IP&S technologies enable the implementation of flexible systems, which generate, adapt, and modify courses of actions dynamically

according to arising needs, thereby releasing human users from the burden to provide predefined plans for all possible contingencies.

These features of IP&S systems allow for the development of advanced, intelligent systems. They provide particular benefit in areas, where the autonomy, flexibility, and robustness of software systems is essential if one is to compete, like in manufacturing, robotics, workflow management, web-based services, and aerospace.

This roadmap reviews the state-of-the-art in key application and technology areas of IP&S. It identifies the most important research, development, and technology transfer efforts required in the coming 3 to 10 years and shows the way forward to meet these challenges in the short-, medium- and longer-term future.

The roadmap is a concerted effort of the PLANET consortium. This document is an integrated collection of the individual roadmaps that were created in the various Technical Co-ordination Units of PLANET and edited by the respective TCU chairs. A large number of people contributed to this document: members and chair persons of the TCUs, core members of the international planning and scheduling community, and independent expert peer reviewers. All of them are acknowledged in Section B.

The document is structured into sections according to the main areas addressed. Each section begins with an introduction into the area and the respective PLANET Technical Co-ordination Unit. After that the most important research, development, and technology transfer needs – together with the required efforts – are discussed along a number of *roadmap themes*. Each section ends with a summary of recommended actions and conclusions.

2. Intelligent Manufacturing Systems

2.1. Introduction

This document evaluates the fields of AI Planning and Scheduling as they apply or could be applied to Intelligent Manufacturing Systems. By manufacturing we mean the industries concerned with the production of physical goods whether by batch or continuous process and whether for intermediate or end users. Thus cars, food, chemicals, machinery and electronics are all included among others. All phases of manufacturing are to be considered from design of plant and products, to operations to plant decommissioning, not only within individual enterprises but also between them, as in supply-chain management and the concept of the virtual enterprise. This requires an engagement with manufacturing concepts such as just-in-time (JIT), concurrent engineering, flexible manufacturing and holonics.

By *Intelligent* Manufacturing we mean the application of AI and Knowledge-based technologies in general to manufacturing problems as just described. This not only includes a large number of technologies outside the scope of PLANET –such as machine learning, expert systems, data mining and neural networks– but it appears that these same technologies have so far proved more popular than AI Planning and even AI Scheduling in such applications. Looking at the reason for the relative neglect of the PLANET technologies and for ways to overcome this neglect is one theme of this document.

We also distinguish the scope of this document from that being produced by the PLANET TCU on Workflow (cf. p.23). The distinction is a little artificial since any business activity can be considered in terms of workflow including all those related to manufacturing. Nevertheless, for the purposes of this document, we see the concerns of the Workflow TCU as operating with processes in which information constitutes the main flow and people or software are the main executors of actions, while those of PIMS (PLANET Intelligent Manufacturing Systems TCU) operate with processes in which physical material is the main ingredient and machines or plant form a major execution resource. The headings used by the 1998 workshop organized by the American Association for AI Special Interest Group in Manufacturing (SIGMAN) [Lug98] provide a good guide. These were:

- Intelligent product and process design

- Production planning and scheduling

- Robotics, sensors, and control

- Systems engineering, learning, and architectures

This list also illustrates the potential overlap with the Robotic Planning and the Dynamic Scheduling TCUs. The distinction to be made here is again one of the specific versus the general – this document will only encompass robotic planning and dynamic scheduling as part of the manufacturing process rather than as general problems.

First we will discuss the state-of-the-art, both in Intelligent Manufacturing and in AI Planning and Scheduling. Next we will examine the requirements of the manufacturing domain for AI Planning and Scheduling. These requirements will then be mapped onto the current state-of-the-art in the technologies in order to assess what applications can currently be targeted and which require further research developments. These research developments will then be discussed and the time scales and resources required to achieve them will be analyzed. We will also consider the organizational and social obstacles to these developments and how such obstacles might be tackled. This document concludes with overall recommendations.

2.1.1.   The Intelligent Manufacturing Systems TCU

It is noticeable that the PIMS TCU was perhaps the smallest group of those interest groups in PLANET. It consisted largely of a small number of university members who had interests in the application of AI P&S technology to manufacturing applications.

- *University of Salford*: applying planning to the generation of plant operating procedures for chemical process plant.

- *University of Granada*: using planning to generate control sequences for manufacturing plant

- *Hungarian Academy of Sciences*: the Computer and Automation Research Institute is active in manufacturing research and also a member of the IMS NoE

- *Charles University Prague*: the constraint logic group has research interests in integrating scheduling with planning in manufacturing applications

Close links were also maintained with the Carlos III University of Madrid who coordinated the Workflow TCU, for reasons discussed above.

2.1.2.   AI P&S and intelligent manufacturing systems

In this section we will consider the state-of-the-art both in the field of Intelligent Manufacturing and in Planning and Scheduling. It should be said at the outset that one of the issues is that these are two almost completely separate communities consisting of different individuals and focusing on different sets of conferences, workshops and other activity as well as mostly publishing in different journals. Intelligent Manufacturing can be considered as a subfield of those interested in applying IT in general to manufacturing, while AI Planning and Scheduling is related to Computer Science. In other words, PIMS must work in a multi-disciplinary fashion in order to create dialogue and synergy between the two communities, making the work understandable and appealing enough for both (see [NGR95]). Work already exists showing that researchers can find a way forward [ASPC98, KJB98].

A further separation arises from the fact that similar terms are used to mean rather different things – thus *planning* in the AI sense does not correspond to planning in the manufacturing sense. Consider the example of production planning. Where AI Planning is an abstract approach to generating sequences of actions to meet goals from a defined initial state, production planning often starts with templates containing sequences of actions and allocates resources to them, making it closer to what in AI would be called scheduling. Some production planning is at least as much concerned with computational geometry, as for example, generating the machining sequences needed to produce particular parts.

There is a closer match in the use of the term *scheduling*, but the term is typically used in the AI community to cover a wider range of activities than those covered by the term in the manufacturing community, as for example rostering and timetabling as well as much of logistics.

Intelligent manufacturing

Themes   There appear to be three major themes in this area reflecting the overall changes in manufacturing: integration, flexibility and ability to make rapid changes.

Integration –of processes within an organization belonging to different stages of the production lifecycle, and between organizations, as in supply chain management (SCM)– is a major theme. Rather than managing one factory or one organization's set of factories, the idea of supply chain management is to cover all the factories supplying inputs to a particular production process. This has emerged as a consequence of ideas such as Just-In-Time –the elimination of stock holding by producing inputs at the time when they are needed– which can only work properly with this type of integration. Supply chain management is probably best thought if as operating at a strategic level of ordering and delivery processes and may involve inter-company processes such as service-level agreements. Software packages for supply chain management are available from companies like Manugistics and i2, but they are currently very expensive and up to now only affordable by very large companies.

ERP (Enterprise Resource Planning) is a similar extension of MRP (Materials Resource Planning) from immediate factory floor requirements into a longer view of resources within the whole enterprise. Supply chain management and ERP while superficially covering similar ground in fact appear to operate at different levels in the organization, to involve different groups of people and to be only loosely integrated with each other and with what actually happens on the factory-floor operations level. Being large systems with wide scope over different groups of people and processes they are difficult to implement and even harder to change once they have been implemented.

The latest buzz word in this domain appears to be Advanced Planning and Scheduling (APS). This seems to be an attempt to incorporate better technology as stand-alone 'point' solutions. APS is aimed on the one hand at smaller companies, who have not been able to afford an all-encompassing package, or on the other at integratable add-ons to existing ERP or SCM packages for larger organizations who feel their big systems need updating but do not want to reimplement the whole thing. While the introduction of AI Planning and Scheduling into the large integrated packages is problematic, influencing organizations producing APS solutions appears more feasible.

Note that integration supposes bringing together people doing different tasks within the organization as well as automation using big software packages. For example, one might supply more information

about operational constraints to designers of processes, or of design rationale and alternative possible designs to operations people in order to integrate design and operations more closely [NAF$^+$98].

A second manufacturing theme is *flexibility* in the face of customer requirements. In the classic manufacturing life-cycle, planning in the AI sense of producing an abstract sequence of actions is kept to a minimum, as it requires skilled personnel to carry it out and has a big impact on what happens on the factory floor. A factory that produces large runs of identical components, or a process plant that produces bulk supplies of the same chemical, are essentially pursuing one plan for long periods of time (maybe selected from a stable library of plans for a known set of products), with the scheduling of resources to support it on a day-to-day basis.

In this environment, while scheduling is carried out very often, and therefore may be seen as an activity that should be supported by computer-based tools, planning is carried out less often. In the past it was often linked to the introduction of a new product, new manufacturing process or new equipment. The engineers carrying out this planning often saw it as a one-off containing a large element of judgement and skill, and they did not see the need for extensive computer support.

However this environment is changing very rapidly. The basic idea of supply-chain management is that production should be determined by 'pull' from the market, not by 'push' from the production process. This supports minimal stock holding along the supply chain since manufacturing is then always responding to current demand rather than producing for stock. However, a consequence of the 'pull' approach to manufacturing is that variation in customer requirements is fed back into production to a far higher degree. No longer can you only have a car in any colour as long as it is black, or the standard widget a factory produces. Large numbers of goods can now be configured on demand, leading to a much greater planning as well as scheduling requirement. The Advanced Planning and Scheduling (APS) developments referred to above can be seen as a response to this need for flexibility.

A third theme, related to flexibility, is *proactive responsiveness to change*. The idea here is that a company ought to be 'nimble', that is when it spots a new opportunity, it should be able to organize itself to meet it as quickly as possible. This means that, for example, it should be possible to have an idea for a new product and turn that idea into manufacture and marketing much more quickly as well as much more frequently than was the case in the past. The requirement for computer support for more skilled and creative activities springs from the idea that these can then be carried out more rapidly, as well as from the idea that more alternatives can –and must– be explored. Planning and scheduling support form an important component of exploring the consequences of a new idea as well as of putting the new idea into practice much more rapidly.

AI/KBS technology in manufacturing applications   If one consults the proceedings of relevant conferences (for example [Lug98, ea98, DL99, oIMS99]) one finds a variety of problems tackled with a variety of techniques. Among these are some planning and scheduling applications related to different manufacturing problems, but these are approached very much from the viewpoint of simple heuristic search problems, and thus, seen as very simplistic AI problems. This reflects a general lack of knowledge by workers in the Intelligent Manufacturing community of generative planning technology and a tendency therefore to 'reinvent the wheel' for these problems. For example, the history of work in the generation of operating procedures for chemical plant, discussed in [ASPC98], shows that chemical engineering researchers attacked it with state-space search and other simple techniques,

finally arriving at a linear planner only in 1991. Because of the size and complexity of some of the search spaces considered, some workers in intelligent manufacturing have applied techniques such as genetic algorithms or constraint propagation, but have not drawn on the contributions of generative planning to the search of such spaces.

In the same way, much of the work on the machining of complex artefacts, which has been tackled by a large number of groups, uses other approaches than generative planning, such as case-based planning [MB98], constraint-based reasoning [HG97] or selection from plan libraries.

In the recent period, multi-agent approaches to manufacturing problems have become more popular, especially as they map well onto distributed control systems. A particular intelligent manufacturing slant on multi-agents is provided by the concept of "holons" [vB96, vBWV$^+$98], a term coined by Arthur Koestler while analyzing hierarchies and stable intermediate forms in living organisms and social organizations. He observed that although it is easy to identify sub-wholes or parts, "wholes" and "parts" in an absolute sense do not exist. He therefore proposed the word holon to describe the hybrid nature of sub-wholes/parts in real-life systems; holons simultaneously are self-contained wholes to their subordinated parts, and dependent parts when seen from the inverse direction.

One might think of holons as agents in the AI sense, though closer to Brooksian behavioural agents than the logic based agents, say, of BDI architectures, since reactiveness and adaption are often emphasized. They are inherently hierarchical but the hierarchy is usually not a fixed one. A substantial amount of work is being carried out in applying this concept to a variety of applications such as scheduling [BVvBP97], machine controllers [TDK95] and Computer-Aided Process Planning (CAPP) [KPDK94]. It is a concept onto which one could map distributed planning and scheduling in the PLANET sense and represents an area in which the two communities could make contact.

State of the art in planning and scheduling

In this section we will consider the state of the art in planning and scheduling with the emphasis on what technology is being used for applications rather than on theoretical developments which are as yet to be applied. We begin by an informal definition of what is meant by the terms AI Planning and AI Scheduling, since as, noted above, they do not always correspond to the meanings accepted in the manufacturing community.

Planning: the automatic or semi-automatic construction of a sequence of actions such that executing the actions is intended to move the state of the real world from some initial state to a final state in which certain goals have been achieved.

This sequence is typically produced in *partial order*, that is with only essential ordering relations between the actions, so that actions not so ordered appear in pseudo-parallel and can be executed in any order while still achieving the desired goals. However some models do explicitly represent true parallelism between actions.

Scheduling: in the pure case, the organization of a known sequence of actions or set of sequences along a time-line such that execution is carried out efficiently or possibly optimally. By extension, the allocation of a set of resources to such sequences of actions so that a set of efficiency or optimality conditions are met.

Scheduling can therefore be seen as selecting among the various action sequences implicit in a partial-order plan in order to find the one that meets efficiency or optimality conditions and filling in all the resourcing detail to the point at which each action can be executed.

The two definitions here reflect the division of the community itself into those concerned with planning and those concerned with scheduling – however as we shall see below this division is to some extent an artificial one.

What applications have been attempted and with what success

It has to be said immediately that there are few example generative planning applications in manufacturing, though quite a number of scheduling ones. The domain with the greatest number of "live" planning applications is Space: the European Space Agency have Optimum AIV [AAP+94], and an Astronaut training application, NASA have the Remote Agent, RAX, mounted on the Deep Space Probe May 1999 [MNPW98], VICAR [Chi94] –planning vision processing for scientists– and control of Deep Space Telecomms facilities [CHW+96]. While as seen below in Section 2.1.3 NASA researchers have drawn some interesting general lessons from these applications, the applications themselves are not something one could use as examples with a manufacturing audience.

Both SIPE-2 and OPLAN have been applied to a number of large-scale military logistics problems in the US, and AI Planning and Scheduling technology (mainly the latter) was famously applied in the DART [CW94] crisis action planning system used in the Gulf War which was said to have paid back all of 30 years funding of AI in the US in a matter of months. Again, these are not in general relevant to the needs of manufacturing though work using AI scheduling technology and constraint logic in particular has been applied to commercial logistics planning. SIPE-2 has also been applied to production planning for a brewery in Australia [Wil90]. It generated a daily plan (master schedule) for two eight-hour shifts on each of six production lines, while the plan scheduled dozens of orders (for possibly hundreds of products) with approximately 20 separate product runs (with their corresponding needs for different raw materials).

Some promising work has been carried out in the last few years on plant control applications, whether at the level of plant operating procedures (OPS-5) or at the lower level of generating control sequences (MACHINE-6). This work is at the proof-of-concept stage but it is clear that it could be taken through to industrial demonstrators and eventual systems if it were attractive to end-users in manufacturing. Planning has also been incorporated into a project investigating Wastewater plant control [SMCL+96], though here reactive planning technology has been used as a variation on the idea of a plan library.

An interesting use of a planner to support diagnosis rather than plan generation was demonstrated in the TIGER project [TMM97, MNG+94] in which the temporal-logic based planner IxTeT worked out expected sequences of actions for gas turbines in order to catch errors if they did not in fact occur. However it is noticeably that this has not as yet been incorporated into the commercial version of the TIGER software, possibly because it was seen as a complex and high-risk component.

As indicated above, a lot of work has been carried out into the matching of complex parts, but almost entirely using case-based planning or plan libraries. an exception to this however is the work on manufacturability being carried out in the US IMACS project [GRN96] which incorporates planning technology alongside feature extraction and other intelligent subsystems. The same group

at Maryland have applied some of these ideas to process planning for the production of microwave modules [MBB$^+$98].

One should note that the Agents Community are involved in a parallel activity to that of PLANET in assessing the fit of their technology to manufacturing [SN99]. While their preoccupation is at this stage with mapping problems onto a set of distributed agents and with negotiation and communication protocols, it is hard to avoid incorporating some form of planning into agents with any substantial functionality, and the possible convergence of interests should certainly be explored.

2.1.3. Requirements

One of the issues in examining current work is that it does not always consider the general lessons of the particular application being attempted. We present a tentative list therefore, with discussion following it indicating where each point was raised so far as this has been established.

a) Life-cycle costs must compete with traditional solutions

b) Knowledge must be acquired painlessly and quickly

c) Currently, configuring a planner to a new domain can be very time-consuming and require a planning expert

d) Most users want interactivity and mixed initiative. They like to retain the user in the decision-making loop and are very reluctant to allow planning software to automate decision-making.

e) Users want validation and ability to assess correctness of plans up front – hence the popularity of plan libraries

f) Users want results in the formalisms they are used to, rather than having to learn new ones

g) Integration with conventional software may be required to provide a complete solution

h) Users often express a wish for optimization of plans in terms of overall resource allocation and for the ability to assess plan quality numerically or via an actually scheduled plan.

Issues a)-c) are very closely linked. The issue a) of competing with existing life-cycle costs has been raised very strongly by Steve Chien of NASA [CHW$^+$96]. This point links in very closely with the work of the PLANET Knowledge Acquisition TCU, since the areas identified as potentially costly in comparison with conventional solutions were precisely those of modeling the domain –issue b)– and validating and verifying the knowledge acquired. The NASA work involved the development of tools for validation and verification, but such tools are not widely available for AI Planning systems. The same point has been noted by other, later work, see for example [ASPC98] where substantial effort was put into developing tools for acquiring planning knowledge in the domain of chemical plant operation in order to combat issue c).

Issues d) and e) have both been raised by workers who have gone for plan libraries, and more recently case-based planning [CLL98, MB98], rather than the generative planning approaches of PLANET. The advantage of both these approaches is that it is easy to involve the user in the selection process –

though note that plan adaption is more problematic unless plan representations are used that prevent users from producing incoherent or invalid plans. A library of plans or of planning cases can be developed off-line and validated and verified, giving the user more confidence than they necessarily feel in a generative planning mechanism [LKS98]. Similar points have been made with respect to users outside manufacturing, for example in military logistics [TDD96], where point e) has also been raised.

Points f) and g) are also closely linked, since the formalisms that users are used to are often those produced by conventional software packages. A key part of the acceptability of the live application discussed in [AAP+94] –OPTIMUM AIV, used for organizing the layout of rocket bays by the European Space Agency– was its integration with the Artemis scheduling application. It is argued strongly in [CFOG99] that the MACHINE application for generating plant control sequences will only be acceptable to control engineers if its output can be represented as the Petri nets or Grafcet diagrams they are accustomed to.

Point h) is a tricky one since optimization in real-world domains is often difficult in practice with "satisficing" –producing an acceptable outcome– frequently more realistic. However if resources are not allocated to a plan it is often hard for users to see how its quality might actually be measured.

2.2. Roadmap themes

2.2.1. What the end users would like

In this section we consider briefly what the requirements of users in manufacturing might actually be in relation to the PLANET technologies. It is in terms of these requirements that workers in AI technologies may well have to justify their approach to potential partners in manufacturing.

Bottom-line gain   To provide a measurable benefit in terms of efficiency, effectiveness, flexibility or responsiveness.

This is the key requirement for any change in a manufacturing environment. However while efficiency –the relationship between outputs and the inputs needed to obtain them– remains a basic requirement, flexibility w.r.t. contingencies throughout the whole production process, from design to control, continues to increase in importance. Real life, production and manufacturing are driven by the demands of the market. This directly implies the need for manufacturing organizations to re-adapt themselves to the continuous changes of that market.

Integration   With existing (legacy) systems and also with organizational structures. Unless the bottom-line gain is absolutely overwhelming, the costs associated with large-scale change are prohibitive. It is usually important not to replace everything at once but to develop incrementally.

Interactivity   As a minimum, the ability for a human user to explore different options easily and to contribute their judgement and preferences. More desirably, allowing an engineer to participate in resolution process in such a way that some decisions are made by humans and others by the

system, combining the strengths of each in a mixed initiative system. This draws on *ease of use* and *functional clarity* below.

Ease of use   The ability to use a system without having to understand in detail how it works and to provide any knowledge it requires quickly and painlessly. This has a bearing on the cost of a system as well as its effectiveness.

Functional clarity, correctness and efficiency   To know what problem a system is solving, for that problem to be important and for it to solve that problem reliably and understandably. Thus while knowing the detailed working of a system should not be essential, the external model of what it does and why at some higher level should be explicit and easy to understand. It must be possible to validate the system's results, given the mission criticality of planning and scheduling activities.

Technology transfer path   To know how systems can be introduced into the organization and maintained and evolved. This draws on many of the previous points, especially *integration*, *ease of use*, and *functional clarity*.

2.2.2.   Open issues

Here we will consider what the PLANET technologies have to offer, what new development and research is needed to increase the number of manufacturing applications, and what sort of applications look promising.

What does planning/scheduling technology have to offer

Any business activity that involves deciding what actions to carry out and in what sequence can be thought of as one to which AI Planning can be applied. AI Planning has a number of potential advantages.

Flexibility and responsiveness   Once the necessary knowledge base has been created for a particular domain, an AI planner can typically produce new plans very quickly and with very little user effort. This makes it useful for exploring "what if" scenarios and contingencies which are often too expensive to consider manually. It also makes it easier to re-plan if execution hits errors.

An intelligent interface   A hierarchical planner has the effect of increasing the level of abstraction at which a human user can operate since it can allow problems to be posed as high-level goals and work out the detailed implications. This makes planning quicker and also less prone to error as low-level interactions between actions are dealt with automatically.

Ability to maintain correctness   Plan representations demonstrate how all the causal links match up to produce a correct plan. Such representations therefore provide a basis for user modification of plans while preventing the production of incoherent or invalid plans as a result.

Making assumptions explicit   In common with other KBS technologies, the explicit representation of knowledge in a planner can make it clear what is implicitly assumed in a problem that is solved manually. Thus it has a role to play in knowledge management and knowledge-sharing.

2.2.3.   What new developments are needed

Manufacturing as a domain has a number of important characteristics from the point of view of AI Planning:

- Execution is key – planning and scheduling are only relevant insofar as they improve execution

- Thus the changeableness of the real-world must always be considered. All decisions may need to be revised

- There are often many execution agents

- Getting accurate current information about the state of the whole system is often hard

- Sensor data may be important

- Costs and efficient use of resources are always important

This suggests some of the areas in which further research effort is needed.

Execution and replanning

There is scattered work in this area, but no real consensus on how planning and execution are related and how replanning should be tackled if a plan fails during execution. Much planning research ignores the issue of how actions in the plan will be executed and does not consider whether execution resources should be represented or reasoned about during planning. In manufacturing, scheduling is required before a plan can be executed, so that ways of bringing planning and scheduling together are a vital research topic for the domain. Planners such as SIPE-2 and OPLAN have combined reasoning about resources with classic planning, as has much of the NASA work (see for example [Mus94] on combining planning and scheduling), but this technology is not perceived as being generally available to the research community.

Work in the Robot Planning TCU is of great relevance here as planning for robots naturally does have to confront these issues. Research in robot planning has considered the interleaving of planning and execution for example, as well as the issue of planning sensor actions as a way of gaining information for planning. A major theme in robot planning is also the integration of predictive and reactive planning as a way of making activity more resilient in the face of a changing environment.

User-friendly tools

There is so far very little work in this area. While NASA developed some in-house tools for validating and verifying domain models, such tools are not generally available to the research community, never mind to potential users in manufacturing. This is in contrast to the situation in Knowledge-Based Systems in general, where such tools have been extensively developed for many years.

Here, work in the Knowledge Acquisition TCU is of great relevance, since the costs of adapting a planner to a particular domain are currently very high and an expert in both planning and the domain are usually needed. This is a major barrier to the use of the technology outside the home community.

An easy entry-level

Expert systems became very widely applied at least partly because of the existence of small, low-function expert system shells, initially arising from the wide dissemination of the MYCIN story and the production of the shell E-MYCIN. Shells enabled enthusiasts in industrial domains to experiment at low cost and risk and to gain an appreciation of what the technology was good for. AI Planning has no such equivalent, since although AI planners are now available over the web they are non-trivial to use. Research is needed to establish whether a simple planning shell is feasible and what it should look like.

Integration

There is little work investigating integration issues either with existing conventional tools or with other AI and computer science techniques. For example, planning machining sequences requires interaction between a planner and computational geometry system. Planning repair actions after a failure requires interaction between a planner and a diagnosis system. Little investigation seems to have been carried out into general mechanisms for supporting this type of interaction, leaving groups who need it to work out ad hoc techniques for particular domains.

It is clear that just as the APS systems mentioned above have been built as bolt-ons to existing manufacturing software, so the PLANET technologies must be integrated with the conventional software already in use if they are to be taken up. This is another area in which there is common ground with the Agents Community, since integration with legacy software is an issue researched there.

The planning community also appears to have split into separate camps concerned with generative and with case-based planning. Yet a generative planner is an obvious way of producing material for a case base, while planner representations, as argued above, provide necessary constraints and safe-guards where users are allowed to adapt plans from a case-base.

Mixed initiative systems

Little work is carried out in the AI Planning community into interactive or mixed initiative planning. While individual projects have seen the need to support interaction (e.g. [ASPC98, LKS98, TDL98]), the only well-known project to concentrate on this issue was TRAINS [FAM96] which in recent years has seemed to focus more on natural language aspects than interactive planning ones. Research is

needed to establish what the general choices are for interaction with a planner and how a true mixed-initiative system can be built.

Common ontologies and libraries

Work on ontologies has been carried out in the ARPA Planning Initiative, and to some extent in the development of the domain description language PDDL. However much more work has been carried out by groups in other areas of KBSs and some of this has been used to inform the development of knowledge acquisition tools for particular domains. There is as yet nothing in AI Planning that corresponds for example to the KADS components developed for diagnosis systems.

2.2.4. What applications should we tackle

Much industrial activity can be characterized by a very abstract sequence. First a problem to be solved is formulated – the business opportunity. An analysis and design activity results, in which an artefact –a process, a factory, a product– is designed to meet certain criteria. The end result of this activity can be thought of as largely declarative in the knowledge-engineering sense, consisting of a CAD model or other specification. Design is a much investigated activity, as discussed above, but even where it is routine rather than heavily creative has more in common with configuration or constraint satisfaction than with planning or scheduling, though the constraint technologies often used in scheduling can of course also be applied to it.

However a design must then be decomposed into the sequence of operations which will allow artefacts to be constructed which meet it. In planning terminology, a design defines the end state to be achieved by a sequence of actions. Thus a plant has to be built, a process has to be operated, an artefact has to be manufactured and assembled. All these activities require the right actions to be executed in the right order.

In the same way, when a problem occurs in a process, diagnosis establishes what fault exists. In the same way as a design, a diagnosis is a declarative statement, in this case of the start state which is to be changed by a sequence of actions back into the desired repaired state. Again, the right actions must be carried out in the right order.

Finally, we should add the need to plan/schedule for contingencies. Since the real world is not always predictable, plans must be built taking this into account. This may involve the production of extra or alternative plans, or it may involve plans adapting their execution to possible contingencies or changes during such execution using information from sensors, or from the interaction with human controllers.

We briefly consider some of the manufacturing life-cycle and consider where planning applications could be located.

Design

Design must take into account basic aspects of functionality and engineering constraints. However increasingly, it is being realized that good design should also take into account operational constraints. For a component this is manufacturability, for a process plant operability, in construction,

buildability. In all cases the designer needs to know that the design can be executed, while the organization as a whole needs to know something about the life-cycle costs of doing so. Designers often lack the specialized knowledge needed to make this assessment unaided – clearly there are a whole number of planning applications which could make this knowledge available and assist the designer in producing a more effective and practical design. This very much fits into the integration theme discussed above.

Process planning

In component manufacture, this can be defined as the act of preparing detailed operating instructions that transform an engineering design to a finished part. This is clearly a planning problem, but must also draw on knowledge of computational geometry as well as the real-world characteristics of tools and ,materials. Many specific applications have been developed in this area using a KBS approach, what planning technology can offer is a much more generic system, which, supported by the appropriate libraries and knowledge sources, could be easily adapted to new components and new production processes. In this way production of new components could be carried out much more rapidly and flexibly.

Process planning may also involve organizing the work cell for production. Here configuration is a primary concern, but planning is then required to produce the desired configuration – in miniature yet another example of the link of planning to a design activity.

Production planning

We here define production planning as the more day-to-day activity of meeting the requirements of producing the right goods at the right time and price, with the right quality. While this is largely a scheduling activity, there is a clear interaction between planning production, and the basic process plan on which it is based. Investigation carried out for this document suggested that just as there is a gap between the concerns of the designer and the process planner, so there is a gap between those of the process planner and the production engineer. Process engineers normally propose a single process plan in which characteristics such as robustness and scope are implicit rather than explicit. A planning system is capable of producing alternative plans, in which the characteristics can be much more explicit, allowing production engineers to tailor the actions taken to the particular situation on the factory floor.

Operations and process control

There is a very clear relationship between planning and operations and control. In operations, humans carry out defined procedures, while in control, machine controllers do the same thing at a much lower level. In both cases, considerable effort up front goes into making sure that procedures and control sequences are correct and comprehensive. AI planning clearly has a potentially major role to play in this area [ASPC98, CFOG99] which has so far been little explored.

The other major operational area is that of maintenance and problem-solving. Maintenance is changing quite rapidly in most industries from a routinely scheduled activity to one carried out when a

problem is detected, increasing the role of automatic diagnosis systems. As argued above, where one has diagnosis, one also has repair, and this is essentially a planning task. For many maintenance activities, it is just a case of making sure a suitably comprehensive set of procedures have been developed in advance, but in some cases– process plant for example– this is non-trivial to achieve and coverage is far from complete. Here an AI planner has an obvious role to play.

This is also the case where problems may be signaled by complex signals of alarms. A diagnostic system is used to filter these for the root problem, but having done this, again, a course of action is required. It is impossible in complex plant to plan for every combination of faults in advance, while allowing human operators who may be under considerable stress to react without support may be potentially quite dangerous. A planning system could be sued here to generate good advice in dealing with a problem, with a particular strength in showing the causal links in the actions proposed and the assumptions made in proposing them.

2.3. Summary and conclusions

Research direction actions

There is some perceived bias in the AI Planning Community towards rewarding the development of new planning algorithms rather than some of the areas raised above. The International Planning Competition in its present form ,for example, encourages the development of fast algorithms but not the production of interactive planners or of replanning capabilities. The most attractive areas for research, judging from the volume of papers submitted in the recent period, is Graphplan and DecisonTheoretic Planning, both very much development of algorithms. If we agree that other areas should also be developed, then ways of encouraging researchers to do so should be sought.

Some possible actions include the development of some standard problems which require the desired research developments (these need not be in manufacturing domains at all); the formulation of outline projects for national or European funding bodies which researchers can develop according to their particular interests and concerted attempts to publicize what has been done in these areas via journal special issues or conference workshops.

Community bridge-building activities

It is clear that linking the AI Planning Community to those working in intelligent manufacturing is vital if the PLANET technologies are to be applied to manufacturing problems. As argued above, there is very little awareness of these technologies outside their home community, and, it should be added, there is generally very little appreciation of manufacturing issues in the PLANET community. Pushing the technology directly to end-users is a possibility, but working more closely with people who are already familiar with manufacturing seems a quicker and more efficient way of educating both communities.

Links have already been made with the ICIMS EU network, and their members have been circulated with a view to joining PLANET. They have also agreed to fund members who wish to attend PLANET activities. PLANET therefore has every reason to approach members of the ICIMS network in order to

try to involve them. This contact ought also to work in the opposite direction though, so that PLANET should encourage its members to participate in ICIMS activities. An obvious way of increasing contact and the exchange of knowledge would be to explore joint events – workshops, tutorials, etc.

The other community with which more contact would benefit both sides is that of Agents. Initial contact has been made with the Product Design and Manufacturing subgroup of FIPA, whose discussions on applying their technology to manufacturing parallel very closely those of PIMS. As can be seen in [SN99], many of the manufacturing domains of interest to the Agent community are the same as those to which PLANET technology can be applied. Discussion on the formulation of some common benchmark applications would help to bring researchers together and prevent each community having to absolutely master the technology of the other.

Unifying applications and problems

It is not possible in this document to arrive at a set of unifying problems or applications in manufacturing since this really requires some of the community bridge-building activities of the previous section. It might well be posed as a possible outcome of a workshop or series of workshops . What is required is a set of problems and backing materials which are reasonably realistic, in the sense of not excluding key features of the domain, but are of a tractable scale for researchers to practice on. Some of the areas discussed in Section 2.2.4 above could be examined for such unifying applications.

Conclusions

This Roadmap has attempted to evaluate the obstacles which have impeded the penetration of PLANET technologies into manufacturing applications as well as to examine the ways in which those obstacles might be eroded. PIMS has always been an embryonic group, but this document suggests that there is much promise in the manufacturing area for AI Planning and Scheduling, and that with the right approach, a much more thriving community of interest can be built to the benefit of all.

Bibliography

[AAP+94]   M. Aarup, M.M. Arentoft, Y. Parrod, I. Stokes, H. Vadon, and J. Stager. Optimum-AIV: A knowledge-based planning and scheduling system for spacecraft AIV. In M. Zweben and M.S. Fox, editors, *Intelligent Scheduling*, pages 451–469. Morgan Kaufmann, 1994.

[ASPC98]   R. Aylett, J. Soutter, G. Petley, and P. Chung. AI planning in a chemical plant domain. In *European Conference on Artificial Intelligence*, 1998.

[BVvBP97] L. Bongaerts, P. Valckenaers, H. van Brussel, and P. Peeters. Schedule execution in holonic manufacturing systems. In *Proceedings of the 29th CIRP International Seminar on Manufacturing Systems*, pages 209–215, Osaka University, Japan, May 11-13 1997.

[CFOG99]   L. Castillo, J. Fdez-Olivares, and A. Gonzalez. Automatic generation of control sequences for manufacturing systems based on nonlinear planning techniques. *Artificial Intelligence in Engineering*, 1999.

[Chi94]   S. Chien. Using AI planning techniques to automatically generate image processing procedures: A preliminary report. In *Proceedings of the Second International Conference on Al Planning Systems (AIPS-94)*, pages 219–224, 1994.

[CHW+96]   S.A. Chien, R.W. Hill, X. Wang, T. Estlin, K.V. Fayyad, and H.B. Mortenson. Why real-world planning is difficult: a tale of two applications. In M. Ghallab and A. Milani, editors, *New Directions in AI Planning*, pages 287–98. IOS Press, Washington DC, 1996.

[CLL81]   H. Chang, W. Lu, and X. Liu. GARI: A problem solver that plans how to machine mechanical parts. In *International Joint Conference on Artificial Intelligence (IJCAI 1981)*, pages 766–772, 1981.

[CLL98]   H. Chang, W. Lu, and X. Liu. Intelligent case retrieval and modification for machining process planning of axisymetric parts. In Luger [Lug98], pages 55–62.

[CW94]   S.E. Cross and E. Walker. DART: Applying knowledge-based planning and scheduling to crisis action planning. In M. Zweben and M.S. Fox, editors, *Intelligent Scheduling*. Morgan Kaufmann, 1994.

[DEW01]   M. desJardins D. E. Wilkins. A call for knowledge-based planning. *AI Magazine*, 22(1):99–115, 2001.

[DL99]   Y. Descotte and J.C. Latombe, editors. *7th IEEE International Conference on Emerging Technologies and Factory Automation*. IEEE Press, 1999.

[ea98]   J. Amat et. al, editor. *Workshop on European Scientific and Industrial Collaboration on Promoting Advances Technologies in Manufacturing - WESIC98*, University of Girona, 1998.

[FAM96]   G. Ferguson, J. Allen, and B. Miller. TRAINS-95: Towards a mixed-initiative planning assistant. In *Proceedings of the 3rd Conference on Artificial Intelligence Planning Systems (AIPS-96)*, pages 70–77, Edinburgh, Scotland, May 29-31 1996.

[GRN96]   Satyandra K. Gupta, William C. Regli, and Dana S. Nau. IMACS: A system for computer-aided manufacturability analysis. `http://www.cs.umd.edu/projects/imacs/imacs.html`, January 1996. Institute for Systems Research, University of Maryland.

[Hay96]   C. Hayes. P3: A process planner for manufacturability analysis. *IEEE Transactions on Robotics and Automation*, 12(2):220–234, April 1996. Special issue on Assembly and Task Planning.

[HG97]   C. Hayes and D. Gaines. Constraint-based algorithm for reasoning about the shape producing capabilities of cutting tools. In *Machined Parts, ASME Design Technical Conference*, Sacramento, CA, September 1997.

[KJB98]   I. Klein, P. Jonsson, and C. Backstrom. Efficient planning for a miniature assembly line. *Artificial Intelligence in Engineering*, 13(1):69–81, 1998.

[Kno96]   C. Knoblock. AI planning systems in the real world. *IEEE Expert: Intelligent Systems and Their Applications*, 11(6), December 1996.

[KPDK94]   J. Kempenaers, J. Pinte, J. Detand, and J.P. Kruth. A collaborative process planning and scheduling system. In *Proceedings of the 1994 ASME - International Computers in Engineering Conference*, volume 1, pages 221–226, Minneapolis, 1994.

[LKS98]   X. Li, S. Kambhampati, and J. Shah. An interactive approach for satisfying process plan generation. In Luger [Lug98], pages 97–103.

[Lug98]   George F. Luger, editor. *AI and Manufacturing Research Planning Workshop: State of the Art and State of the Practice*. AAAI Press, 1998.

[MB98]   M. Marefat and J. Britanik. Case-based planning for three-dimesional machined components. In J.M. Usher, J. Roy, and H. Parsaei, editors, *Integrated Product and Process Development: Methods, Tools and Technologies*, chapter 7, pages 185–225. John Wiley & Sons, 1998.

[MBB⁺98]   J. Meyer, M. Ball, J. Baras, A. Chowdhury, E. Lin, D. Nau, R. Rajamani, and V. Trichur. Process planning in microwave module production. In Luger [Lug98], pages 120–126.

[MNG⁺94]   R. Milne, C. Nicol, M. Ghallab, L. Trave-Massuyes, K. Buosson, C. Dousson, J. Quevedo, J. Aguilar Martin, and A. Guasch. Tiger: Real-time situation assessment of dynamic systems. *Intelligent Systems Engineering*, 3(3):103–124, 1994.

[MNPW98]   Nicola Muscettola, P. Pandurang Nayak, Barney Pell, and Brian C. Williams. Remote agent: to go boldly where no AI system has gone before. *Artificial Intelligence*, 103(1–2):5–47, 1998.

[Mus94]   N. Muscettola. HSTS: Integrating planning and scheduling. In M. Zweben and M.S. Fox, editors, *Intelligent Scheduling*, pages 169–212. Morgan Kaufmann, 1994.

[NAF⁺98]   W. Nederbragt, R. Allen, S. Feng, S. Kaing, R. Sriram, and Y. Zhang. The NIST design/process planning integration project. In Luger [Lug98], pages 135–139.

[NGR95]   D. Nau, S. K. Gupta, and W. C. Regli. AI planning versus manufacturing-operation planning: A case study. In *International Joint Conference on Artificial Intelligence (IJCAI-95)*, pages 1670–1676, 1995.

[oIMS99]   Working Group on Intelligent Manufacturing Systems, editor. *Intelligent Manufacturing Systems Workshop*, Leuven, September 22-24 1999.

[SMCL⁺96]   M. Sanchez-Marre, U. Cortes, J. Lafuente, I. R-Roda, and M. Pcha. DAI-DEPUR: An integrated distributed architecture for wastewater treatment plants supervision. *AI in Engineering*, 10:275–285, 1996.

[SN99]   Weiming Shen and Douglas H. Norrie. Agent-based systems for intelligent manufacturing: A state-of-the-art survey. `http://sern.ucalgary.ca/CAG/publications/survey-abm.htm`, 1999.

[TDD96]   A. Tate, B. Drabble, and J. Dalton. O-Plan: A knowledge-based planner and its application to logistics. In A. Tate, editor, *In Advanced Planning Technology, The Technological Achievements of the ARPA/Rome Laboratory Planning Initiative*. AIAI Press, Menlo Park, California, 1996.

[TDK95]   P.I. Tanaya, J. Detand, and J.P. Kruth. Holonic machine controller - a study and implementation of holonic behaviour to current NC controller. In *Co-operation in Manufacturing : CIM at work*, pages 375–396, Kaatsheuvel, The Netherlands, 1995.

[TDL98]   Austin Tate, Jeff Dalton, and John Levine. Generation of multiple qualitatively different plan options. In *Proceedings of the 4th International Conference on Artificial Intelligence Planning Systems (AIPS-98)*, Pittsburgh, 1998.

[TMM97]   L. Trave-Massuyes and R. Milne. Gas-turbine condition monitoring using qualitative model-based diagnosis. *IEEE Expert: Intelligent Systems and Their Applications*, pages 22–31, May/June 1997.

[vB96]   H. van Brussel. Holonic manufacturing systems, the vision matching the problem. In *First European Conf. on Holonic Manufacturing Systems*, Hannover, 1996.

[vBWV⁺98]   H. van Brussel, J. Wyns, P. Valckenaers, L. Bongaerts, and P. Peeters. Reference architecture for holonic manufacturing systems: PROSA. *Computers In Industry*, 37(3):255–276, 1998. Special Issue on Intelligent Manufacturing Systems.

[Wil90]   D.E. Wilkins. Can ai planners solve practical problems? *Computational Intelligence*, 6(4):232–246, 1990.

[WS99]   D. H. Norrie W. Shen. Agent-based systems for intelligent manufacturing: A state-of-the-art survey. *Knowledge and Information Systems*, 1(2):129–156, 1999.

3. Workflow Management

3.1. Introduction

This chapter presents an R&D Roadmap for AI Planning and Scheduling (AI P&S) applied to business process management (BPM) produced by the Workflow Management Technical Coordination Unit (TCU) of PLANET. The purpose of a Roadmap is to coordinate R&D by establishing end-user requirements on short medium and long time scales and proposing research and technology transfer goals and activities that will enable the requirements to be satisfied. The current version is only a first step towards such a Roadmap, which in any case should be a living document updated regularly. BPM and AI P&S are two disciplines with many parallels, but which have largely been pursued by disjoint communities. A necessary precursor to producing a Roadmap is to align the two disciplines so that specialists in each can understand each other. One of the main achievements to date has been to develop an understanding of how the "world view", vocabulary, challenges, etc. of Business Process / Workflow Management relate to AI Planning and Scheduling. This has been possible because of the active participation of a number of workflow and process management experts from end-user organizations and consultancy companies.

Currently, there is a growing interest in the application of Artificial Intelligence (AI) Planning and Scheduling (P&S) techniques to real world problems. We have recently seen impressive applications of AI P&S in space, robot, elevator control, military missions planning, etc. However, there are still many open tasks that can be (semi-)automated using AI P&S technology. One of such tasks is the production, and execution of models of organizations (workflow management). Here, we will describe what AI P&S techniques and models could be applied to workflow management and what actions can be carried out in order to achieve it.

3.1.1. The Workflow Management TCU

The main purpose of the TCU on Workflow Management is to promote the effective application of AI Planning and Scheduling (AI P&S) techniques to Workflow Management. We have tried to bring together researchers, practitioners, and software vendors in the fields of AI P&S and workflow management. During phase I of the PLANET research network, this TCU produced a Roadmap in which the commonalities were identified and described in some detail. The main conclusion is that they deal with common problems, so that AI P&S can greatly help on the automation of processes within organizations. This is an updated version of that document.

A popular way to model how organizations work is to focus on their internal processes (the ways they do business and the activities and business rules they follow). However, this is not a simple task;

organizations operating in the current economy, especially those doing business via the Internet, have processes which are constantly changing in response to the needs of their customers and the business environment. This raises two distinct requirements. Firstly, organizations need powerful tools to automatically model, simulate and optimize their processes in such a way that the generated models comply with business rules. Secondly, organizations need computational tools, usually called workflow tools, for executing (enacting), monitoring, and dynamically adapting to changes those processes. So, the output of the first type of tools, the process models, become the input to the second type of tools (see Figure 3.1). From the point of view of the organization, it can be seen as a set of agents (human and machine resources) executing processes (in parallel or in sequence) where each process is composed of a set of activities linked by constraints and dependencies.
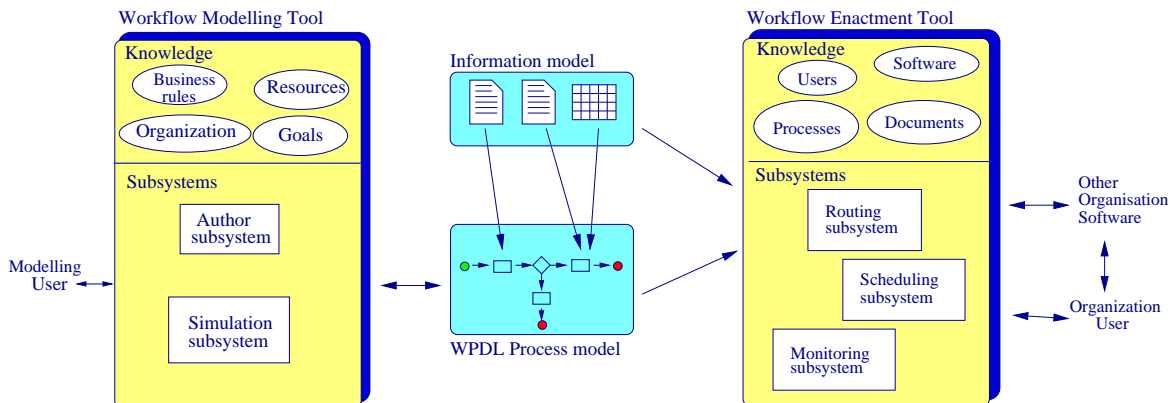
Figure 3.1.: High level view of an architecture for workflow applications.

From an AI perspective, P&S tools can be effectively applied for the first type of tools, since they provide a declarative representation of the knowledge within the activities of processes, as well as means to generate only valid process models. This is crucial for people to understand how the organization are really behaving. Also, P&S tools are able to obtain good process models according to one or several criteria (time to enact process, user satisfaction of the process, cost of the process, . . .). In relation to the second type of required tool (enacting processes), AI P&S allow for efficient monitoring (through the explicit reasoning about predicted situations after applying plans or parts of plans), as well as re-planning when problems arise during enactment of processes.

Workflow management is a field in which few applications exist of these techniques. Workflow management has two interesting properties: a potential big impact in modern organizations (specially in the e-business context); and a close connection of the way AI P&S techniques work and how problem solving occurs in workflow management. The current state of the art in workflow shows that very few commercial tools incorporate AI P&S techniques into them. Current Workflow Management systems (WfMS) essentially automate the routing of documents between workers or teams according to pre-defined processes definitions. At the same time, they also handle the sets of tasks to be performed by the workers. WfMS and AI P&S are two disciplines with many parallels, which have largely been pursued by disjoint communities.

3.1.2.   AI P&S and workflow management

In this section we will first describe workflow and process management. Then, we elaborate on some aspects related to the common issues between AI P&S and Workflow management. The two communities, Workflow Management and AI, are separated by a common language, with terms such as planning being used in both with different meanings. A necessary precursor to producing a Roadmap is to align the two disciplines so that specialists in each can understand each other. One of the major successes of the TCU to date has been to bring AI P&S researchers into contact with BPM specialists and so further mutual understanding.

Workflow management and business process management

A business process is the chain of activities involved in delivering a product or service to a customer (within or outside the organization). In fact, since the customer's satisfaction with one service influences requirements for future services, a business process is best seen as a closed loop. In addition to core business processes, there are management processes (including processes concerned with designing the core processes) and support processes that facilitate the other types of processes. The set of business processes for an organization comprises the organization's working practices. Organizations differ in how explicitly the processes are defined, and in the form they are represented. In some cases the processes are implicit, in others they are recorded in textual codes of practice, in others they are documented in (semi-) formal representations and/or software modeling tools. A set of business processes is highly analogous to a set of stored plan templates or a hierarchical task network (HTN) and could readily be represented in this way (this will be explained further later).

Designing business processes is a knowledge-intensive human activity supported by software modeling and simulation tools, and is closely tied in with matters such as business policy and enterprise organization and culture. An instance of a business process created, for example, to deliver a particular service to a particular customer is analogous to a plan in AI. In BPM terminology, however, a plan also includes allocation of resources (e.g. workers) and target start and completion times. In terms of AI, this would be the equivalent of generating a plan with resource and temporal information, that is, the integration of planning and scheduling techniques [Dra99].

In some application domains, for example military logistics, generating a plan and instantiating it with appropriate resources and time windows is complicated, and AI Planning techniques are being applied successfully in such areas [TLJD00]. However, following the way in which business processes are handled currently, planning only involves selecting from a set of pre-defined templates. The main technical challenges in this setup arise because an organization is a distributed system that executes many process instances concurrently in an uncertain environment. Furthermore, failures and other exceptions occur frequently, and re-planning must be integrated with execution. In the next future, automated planning tools should not only instantiate processes templates, but also be able to generate dynamically the executable processes templates.

A workflow management system (WfMS) automates the coordination of activities and transfer of documents within a business process [GHS95]. It delivers the work to the "in-tray" of the appropriate software component or human worker or team according to pre-defined rules (a process or workflow definition). Current WfMS do not (generally) perform planning, scheduling or resource allocation.

Any such considerations must be built in to the process definition or else handled by the productive resources owning the in-trays. Specifying this low level process or workflow definition is again primarily a human design activity performed with the assistance of software tools (often specific to the WfMS).

Business process management can be presented as having the following aspects (as also shown in Figure 3.2):
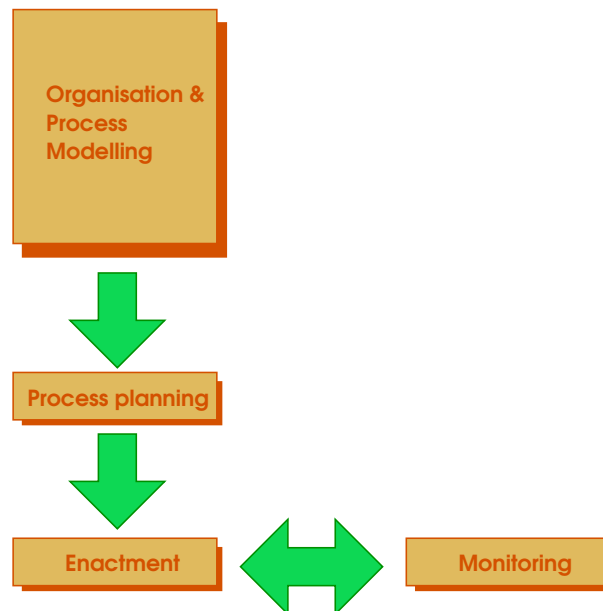
Figure 3.2.: Business process management steps.

Process modeling   This involves designing, modeling, evaluating (simulating), modifying, optimizing, etc. the organization's processes. For each basic product or service the organization offers to its customers, the activities involved, the relationships between them, resource requirements, etc. must be defined. It is basically a human activity, though supported by computer-based tools to record and display the process model, run simulations, etc. Design decisions are made based on experience and analogy to previous designs. Choices are tied closely to other aspects of the enterprise and business environment such as: the nature of the business, business goals, organizations standards or norms, organizational structure (of the enterprise), enterprise culture, legacy infrastructure, etc. Although process design is often presented as happening top-down, the practical constraints imposed by the current state of the enterprise mean that there is a strong bottom-up behavior. Design of the processes and activities typically go hand-in-hand, so that although the analogy between process and AI plan is strong, the analogy between the activities of process design and classical AI planning is much weaker.

Process planning (elaboration, resourcing and scheduling)   A process definition is basically a template. This phase involves identifying the appropriate template to use, elaborating and filling in an

instance of that template in sufficient detail for it to be executed. The first step is normally to gather information from the customer on the product or service required. This allows the tree of possible processes to be pruned considerably, but a number of alternative branches may still remain. The next step is to produce a schedule based on a target end date required by the customer, dependencies between tasks, and knowledge about how long tasks take (e.g. typical and minimum times). If it is not possible to achieve the target end date, negotiation with the customer takes place. Then, the people and other resources required for each task are identified and "reserved" for the appropriate time slots ("resourcing" or "provisioning").

The resourcing and scheduling problems are coupled by virtue of finite capacity and/or non-sharable resources. If the required resources are not available, then the earlier steps must be revisited. In process management the result is referred to as the plan, and the process of producing it as planning, which is a source of confusion as the usage is different from that in AI. Note that further detail will often be decided at execution time, and the balance between design time and execution time decisions varies considerably. Again, these activities are often performed by people assisted by relatively dumb software tools. The nature of the tools and the form of the output depend on context. For example, MS Project (or the equivalent from another supplier) could be used to create a "production plan" to be carried out by a human organization. Alternatively a proprietary tool could be used to generate a process description for enactment by a workflow engine. In each case, the tool and the representation is often different from those used earlier in the modeling phase.

Enactment   The production plan is carried out, with detail being elaborated during enactment. The boundary between planning and enactment is context dependent. Process planning is essentially the first part of enactment of a core process. Furthermore, at the start of enactment, the plan may still contain alternative branches that are pruned as information is gathered and decisions made during enactment. Almost always, execution is distributed, with different production resources, computer programs, or people carrying out the constituent activities. The activities have to be coordinated to ensure correct sequencing and that compatible variants of the activities are performed. Coordination takes place via mechanisms such as: events, transfer of documents, existence checks on documents, etc.

A workflow management system uses information contained in a low-level process plan definition to route work items to the appropriate production resource and provide the necessary coordination signals. Note that workflow systems (generally) do not plan work, and workflow also assumes resources will be available. Production resources will be involved in enacting multiple processes and instances of the same process in a time-sharing manner. A production resource (or rather a component encapsulating one or more resources) sees the processes in which it participates as a queue of work items (or tasks) waiting to be acted upon. Depending on how the system is organized it may simply work on the next task whose pre-conditions are satisfied, or it may have rules for prioritizing tasks. Either way, different processes can interfere with each other due to the finite capacity of a shared production resource.

Monitoring   As execution proceeds, information on progress (e.g. notification of completion of tasks, delays and other problems) is fed up to a management function. This compares actual progress with the production plan. Minor differences between the plan and actual progress may simply require

updating of the plan (for example with slightly different commencement times for tasks). These changes need to be propagated to the resources executing the plan. More significant differences may require the planned activities to be altered during execution. This may include some back-tracking, for example to remove some item of equipment that was installed following the earlier plan, but is now no longer required. More drastic problems may require all the effects of the plan to be undone and a new plan created. The monitoring function may try to anticipate future problems and modify the plan in advance to avoid the problems. This is sometimes known as jeopardy management.

AI Planning and Scheduling

AI Planning and Scheduling (AI P&S) is concerned with determining an ordered set of actions that when executed by one or more agents with the world in some initial state satisfying given conditions, results in world state satisfying given goal conditions. A process is a description of an ordered set of activities. A plan is a description of activity for a given objective; it is an instantiated process. AI Planning provides many different techniques to generate plans, but there are two main ways of specifying the domain. On one hand, in STRIPS style planning, the operators consist of individual activities [FN71]. A planner combines instantiation of these for a given objective to form a plan. On the other hand, HTN planning domain descriptions are essentially process descriptions. They let you specify parameterized descriptions of processes that can be automatically assembled and instantiated to form a plan for a given objective. Given the similarities between the HTN representations and some plan templates used in the workflow world, the HTN approach seems a well suited one for this domain. Classically, this overall problem is divided into a number of stages:

Modeling (or knowledge engineering)   This concerns finding the right way to represent the world and the problem so that planning and scheduling may be performed. Classically, this representation consists of a definition of some space of states that the world and its constituents may be in, and a set of primitive operators that can be applied to cause (constituents of) the world to change states.

Planning   This concerns finding one or more ordered sets of actions that should cause the world to change from the initial state to a state satisfying the goal conditions. The ordering of these actions is not necessarily completely determined. Planning is concerned with logical dependency of actions in the set, e.g. that if action A is necessary to bring about the pre-conditions for action B, then A is performed before B. Planning may be performed bottom up by chaining together actions until the gap between initial and final states is spanned (e.g. STRIPS model). Alternatively, it may be performed top-down by recursively refining generic plans until they are expressed entirely in terms of executable actions (e.g. HTN model).

Scheduling   A plan expresses the orderings of actions that should be able to bring about the goal. Scheduling determines which of the orderings of actions consistent with the plan will actually be used and on what time frame each activity will be executed. Often, this choice is based on some form of efficiency measure, for example overall time taken to execute (make-span). Also, scheduling handles the assignment of resources to individual actions (activities) so that resources are not over-allocated.

Traditionally, planning and scheduling have been separated fields. However, recently there is a strong interest on performing them in an integrated way. Either by following a scheduling step after a planning step, by providing the adequate interfaces, or by developing integrated tools [GL94, Mus94], a growing number of researchers are focusing on this issue, as it can be seen from the latest workshops and conferences papers [DKR02, GHT02, Ces01].

Execution   Execution as such has not been a major concern of AI P&S except for some domains, such as robotics, or space missions. However, particular branches of AI P&S are concerned with execution-time issues. For example, it is recognized that an action does not always achieve its intended result. Thus monitoring must take place to compare anticipated events with actual ones, and if deviation is significant plan repair is initiated. If deviation is excessive or repair impossible, then the plan is abandoned and a new plan generated. At the extreme end of the scale are so-called reactive planners in which planning and scheduling take place at execution time, with planning, scheduling and monitoring actions interleaved with the goal-achieving actions. There are also hybrid approaches that lie between the two extremes and allow to efficiently go from one extreme to the other.

A comparison

It is clear from the above descriptions that process management and AI P&S address similar issues, and there are many parallels between the two disciplines. Figure 3.3 compares the two at a coarse level, aligning phases that are roughly equivalent. However, there is no direct equivalent to AI Planning on the process management side - although a process (model) is approximately equivalent to an AI plan, it is generated by people supported by software drawing and modeling tools rather than by an analog of AI Planning. In the Figure, it is explicitly shown one of the main advantages of using AI technology in the framework of process management: the "continuous" flow of control and data among the different levels as represented by the discontinuous lines in the right of the figure. Process management usually sees the whole process as a unique path from process definition to enactment, without very few (automatic) feedback. AI can greatly help on incorporating automatic feedback among all levels.

Usually, in the case of BPM, processes have very few knowledge describing each activity. They usually have information on issues such as who is responsible of the activity, or the time and cost of the activity. Very rarely, one has to provide information on their pre- and post-conditions, as it is the case for AI P&S. However, from a knowledge-rich perspective of an organization,[1] those conditions should be specified, so that reasoning about itself can take place as was studied during the SHAMASH EU-funded project [ABCSA02].

Also, both business processes and AI plans can occur on multiple levels. Thus process management can itself be seen as an enactment of a meta process; enactment of strategic processes may involve definition of tactical processes and so on. Similarly execution of strategic AI plans may involve planning at a tactical level. In consequence, it is important to consider applications of AI P&S within the activities taking place during enactment a business process. For example, an early step

---

[1]There is clearly now a trend towards an explicit and declarative representation of organization knowledge through knowledge management, competencies modeling, ontologies, etc.

Workflow Management   **AI Planning & Scheduling**
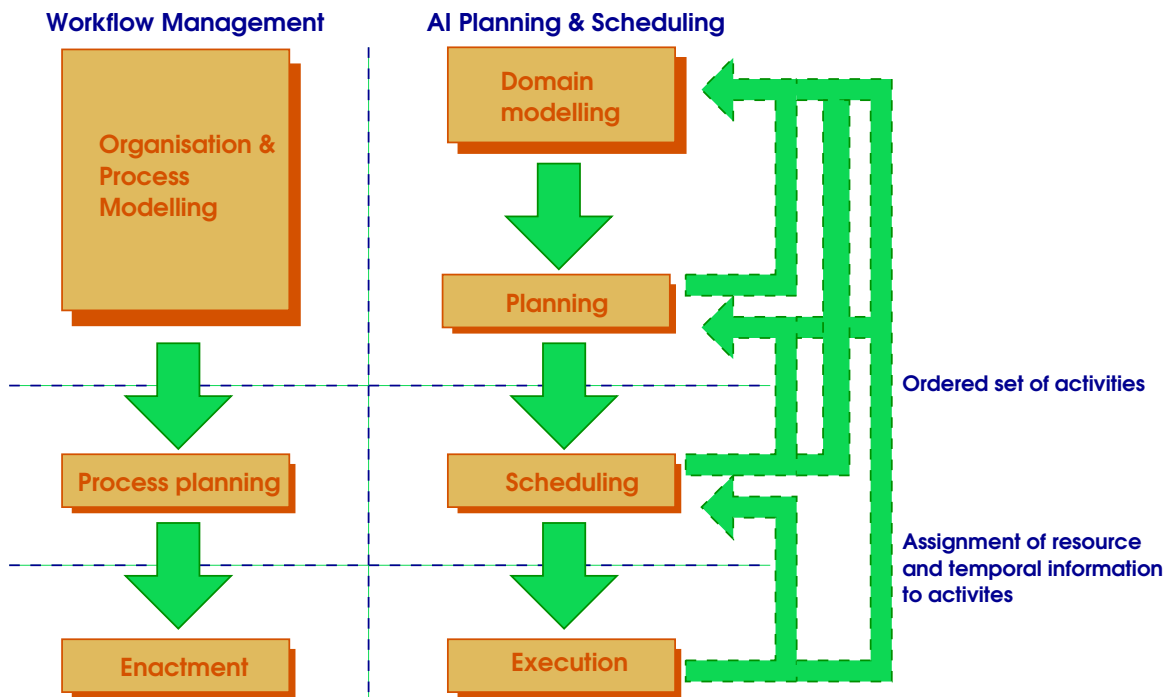
Figure 3.3.: A comparison of process management and AI P&S.

in a process may involve detailed planning or scheduling of activities occurring later in the process. Despite the similarities, there are also significant differences:

- Terminology - the word "plan" itself has a different meaning in the two disciplines as it has been specified before;

- Most of the design-time (as opposed to execution-time) activities in process management are performed by people assisted by relatively simple software tools. In contrast, the emphasis in AI P&S is on producing intelligent software that can perform planning and scheduling largely automatically, with occasional assistance from a person;

- AI P&S representations tend to be mathematically formal and semantically precise, though this often means they are difficult (for a non-expert) to understand. In process management, the opposite is true: the representations are domain-oriented and easy to understand, though the semantics are often somewhat vague.

- Languages for defining processes as input to workflow engines are basically scripting languages for coordination of activities and are at a lower level than AI plan languages.

- Classical AI Planning techniques focus on difficult combinatorial problems - many different combinations of operators and states are possible, only a few of which constitute viable plans. In process management, activities are fairly specific to processes, and there is much less scope for combining them in different ways to form different processes. These differences present opportunities for synergy as well as barriers to be overcome.

Current state of the art in workflow and process management

The idea of process management is still fairly new. In the past, organization processes were implicit in each organization structure and culture. Departmental procedures and practices would be known within the department, but no individual had a clear end-end view of a process. A similar statement could be made about the software systems that support the enterprise operations. These were, and still are, often large monolithic applications in which the business processes are implicit. Consequently they are difficult to change and tend to tie the organization into the processes encoded in the software. However, the importance of the day to day operation of the organization and the expense and disruption involved in replacing them mean that many of these so-called legacy systems are still in active use. The current trend in both organizations and their operational support software is to represent the business processes in an explicit and distinct manner. As a result, it is easier to study how to improve a process and also easier to implement the improvement. In the case of the software, the need for modifiability and software re-use has led to a component-based philosophy. Instead of monolithic applications, functionality is encapsulated in re-usable modules that can be combined in different ways to construct new "virtual" applications rapidly. One way to view workflow management systems is as the architectural glue that links the components together to form the application. At least in theory, the process definition can be changed independently of the components, and functionally equivalent components substituted without changing the process definition. Often these components do not replace the legacy applications. Rather the components use them as servers in providing their functionality.

Of course, much of the work in a business process is performed by people. A workflow management system treats people in much the same way as the computation components. Typically, an interface is provided that presents the user with an in-tray and out-tray of work items. This interface encapsulates the user in a similar manner to that in which the component interface encapsulates the software functionality. This approach is suitable for partial automation of well-understood routine processes.

Sometimes group-ware software systems (such as Lotus Notes and Microsoft Exchange) are described as workflow systems. These systems primarily provide a messaging and information sharing environment that can be used by participants in business processes. However facilities such as document routing scripts and forms can be used to define workflows to some degree. Industry is currently in transition from the old-style monolithic support applications and paper based office processes to workflow-based systems. Legacy applications certainly will not disappear overnight. Rather, components and workflow systems will gradually diminish their role. The legacy software may never disappear entirely, however, especially where the products, services and associated processes are relatively mature. AI scheduling (and to a much lesser extent planning) techniques have certainly been used in special purpose business support applications. However there has been little or no influence by AI P&S on process management as a discipline or on the methods and tools through which it is applied. Similarities (for example between plan and process description languages) are due more to convergent evolution than to direct influence.

3.1.3. Requirements

Within this document, requirements have been classified as short, medium and long term as follows:

short term address short-comings in current-generation process management software. The most important items in this category are: integration of temporal reasoning and resource allocation/management algorithms into workflow management software; and incorporation of a planning capability to enable a WfMS to modify the process instance automatically during execution, to cope with failure, changed objectives, and other exceptions.

medium term current generation workflow software handles high volume routine processes, typically involving low-skill workers. The medium term requirements concern extending this support to high-skill knowledge workers. This may involve, for example, building process knowledge awareness into software tools.

long term more radical (e.g. adaptive self-organizing) approaches addressing the need for organizations to function in a business environment that is increasingly uncertain and subject to change.

These requirements describe research and development goals that could be fulfilled with integrated projects within FP6.

3.2. Roadmap themes

The remainder of the document expands requirements and looks at ways in which they could be met by existing and future results from AI P&S and related disciplines. The discussion is divided into a number of themes (see Figure 3.4).
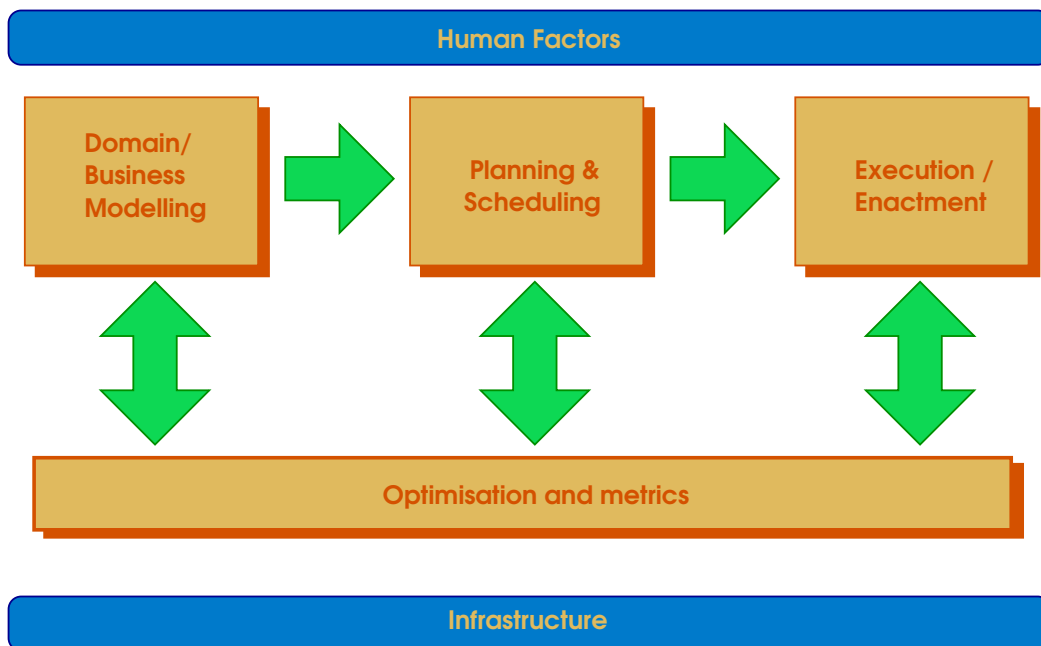
Figure 3.4.: Themes that are discussed in this document.

Each theme section includes subsections on the state of the art (including trends and current projects), research goals and open issues, and recommended actions. The first theme deals with human issues. A feature of business processes is that much of actual work is performed by people. There is a tendency in BPM to pursue automation and to treat human actors in the process as if they were machines. This is often counter-productive resulting in de-motivation and a failure to utilize human qualities. This section explores these issues and examines how they might be addressed. The following section looks at software infrastructure. It is important to appreciate that for AI P&S techniques to be applied in practice they need to be integrated with / interfaced to commercial software packages. This section looks at issues such as reference architectures and interface standards. A common understanding of architecture would also facilitate collaborative research and demonstrations. Sections covering life-cycle oriented technical themes then follow:

- business / process modeling and knowledge engineering: generating a computer usable representation of processes is the first task to be solved in both fields. Here, common problems arise such as how much the process representation corresponds to reality, or what language to use to represent the model.

- planning, scheduling and resourcing: both fields require the generation of ordered sets of activities to be executed (enacted) either in sequence or in parallel. These activities need also information with respect to resources, as well as time frames implied in their execution.

- enactment/execution and monitoring: less studied in the field of AI P&S and with more software tools available in the field of BPM, the execution (enactment) of the plans is a key component of the cycle. Specially important in the enactment of business process are failures (or predicting them) and how to handle them.

- adaptation, optimization and metrics: usually, processes (or plans) have been generated without optimization goals in mind. Organizations in current very restricted and competitive markets need gradually more emphasis on metrics and finding better processes. Also, a related aspect is how to dynamically change the processes to adapt to those markets.

The boxes in the middle of Figure 3.4 correspond to the main divisions shown in Figure 3.3. Some further explanation is needed where the two halves of Figure 3.3 do not align, however. Basically, activities in which people design or elicit models (possibly supported by software tools) are included in the modeling theme. Planning (as understood in AI P&S) is included with scheduling and resourcing.[2] Planning and scheduling can be performed as part of the enactment of processes as well as off-line. The optimization and metrics theme recognizes the need to measure attributes of the various models and provide feedback to improve desirable qualities. The feedback can be to the same box, e.g. measurements of the attributes of a process model can be used to optimism the process model. Larger-scale optimization is also important, however. For example, measurements during execution of a process can be used both to adjust the theoretical production plan to better reflect reality, and to provide information to help improve it. These improvements can be fed back into the executing process, and so on.

---

[2]We will use indistinctly the terms "scheduling" and "scheduling and resourcing". Therefore, we will consider that scheduling techniques can also handle resources, and not only time information.

The remaining two boxes represent "orthogonal issues": infrastructure and human factors. An important infrastructure issue is the establishment of a reference architecture for a highly-modular "AI-enabled" process management system, covering both design-time (off-line) and execution-time systems. The human factors box reflects the need to take into account (throughout process management) of the special characteristics of the people involved in performing the processes, and to form a symbiotic relationship between "man and machine". These two "orthogonal themes" are covered first.

3.2.1. Human issues

In this section, we will focus on the state of the art problems, requirements, and future actions that relate to the fact that humans are in the loop with respect to workflow applications. This will condition how AI P&S techniques will/can be applied to this domain. Perhaps, the biggest difference between the type of domain requirements from the area of Workflow Management and those from the Intelligent Manufacturing area is precisely related to human issues (see report on the common workshop organized by the two related PLANET TCUs in [AB02]).

Introduction

In relation to workflow management, human issues can be mainly divided into three main categories depending on the role of the user:

- Users involved in process design. This issue will be covered in more detail in Section 3.2.3. Also, for a deeper understanding of the knowledge engineering aspect of developing AI P&S tools, the reader can consult the Roadmap of the PLANET TCU on Knowledge Engineering.

- Users involved in process management tasks. Once the process has been modeled, it can be enacted. There is a human role (one can think of a finer division of this role into several) for monitoring the execution of activities, adjusting the workflow tool, or drawing conclusions from the observed behavior of the enacted process. This issue strongly relates to monitoring tools, which are auxiliary to the main task of the workflow tools (routing documents among people). We will not discuss here the issues that this type of users enforce into the requirements of the tools to be developed, but they will certainly be important in the future to consider, once workflow tools incorporate many of the features that are being described in this document.

- Users of the workflow tool. Given that processes are enacted by people from the organizations, it is crucial to study the impact of fielding a workflow system within the organization. As it usually happens with computer applications being used in organizations, there has to be a consensus of the utility of using such tools. In this section, we focus on this aspect.

Current state of the art

Most business processes require people to perform at least some of the tasks. Workflow management systems tend to view people and software resources in the same way: as means of carrying out process steps. At the present state of the art, they can be applied only to well defined, routine

processes, introducing even more regimentation into dull, boring jobs. All too often, software is seen as a means of decreasing costs through automation and standardization rather than as a means of enhancing value and quality of he product/service and hence customer satisfaction. It is also used to monitor productivity, thereby increasing pressure on workers further. Unfortunately the quantities measured tend to be those that are easy to measure (number of calls handled in a call center, number of studied proposals, etc.) rather than true measures of value contributed.

AI Planning, too, has traditionally been concerned with automation of processes. The goal has been to build intelligent machines, i.e. to enable machines to perform activities that currently only people/animals can do. Little attention has been paid to amplifying human abilities, though there is a body of work in the area of mixed-initiative planning [ASF$^+$95, TDL98, VMC97, dDOW99].

Computer-supported cooperative work [BL98] was a very active field in the early nineties, though activity seems to have died down recently, re-focusing on work within virtual teams. A stream of work within CSCW focuses on Tailorable Workflow systems [KSWM00], where workers can modify run-time functionality of workflow via preferences. Industrial psychologists work in the area of job design, which focuses on maximizing the motivational characteristics that people experience in their jobs [Old96], and have formulated factors that improve job satisfaction, for example autonomy, variety and responsibility. Technology such as workflow has the potential to both simplify and enrich the nature of work. Control over various aspects of work such as timing and method is thus considered crucial in workflow-type environments, with higher control leading to better productivity and work attitudes [JWMD93].

Research issues

The main objective with respect to human issues would be the understanding of how to achieve a synergistic, symbiotic relationship between human workers, managers and software systems within business processes. In order to achieve this goal, some research issues arise, such as:

- How to involve users in controlling their coordination support and workflow planning systems? A decision to empower users so that they can control workflow systems requires appropriate interfaces and methods to make this user control possible. These interfaces and methods have to be carefully designed to take into account the expected variety of user backgrounds and programming skills. The academic fields of End User Development [Nar93] and Visual Programming [BM95] have researched these issues in the general case of software programming and control. An interesting research direction would be to see how the general findings in these areas map onto the specific domain of workflow planning.

- How to take into account human issues when at the process modeling and definition stage of workflow planning systems? Two approaches that are used in mainstream software development are Participatory Design [KM93], which aims to involve user representatives in the design of new software, and input from industrial psychology such as job satisfaction factors.

- What is the optimal balance between users and software during the different stages of workflow planning and scheduling? Here again the work on mixed-initiative planning could help on finding automatic ways of balancing the control between the user and the software tool.

- If planning techniques are to be employed successfully, new means of visualization and explanation need to be developed to reflect the combinatorial nature of AI Planning. This is specially true when new very fast planners use search techniques that are difficult to explain to a human in case they have to collaborate at the search level.

Recommended actions

In order to carry on this research, a set of actions are recommended here:

- To run a set of trans-disciplinary workshops, which discuss the relationship between systems providing user-control of workflow and contributions from the areas of Participatory Design, Industrial Psychology, Visual Programming and End User Development. A report could summarize the workshop findings and possible new research directions can arise from it. A meeting as such would have the problem of first trying to use a common vocabulary, as we also suffer in the first meetings of this TCU. Typically, this type of interdisciplinary workshops tend to have a biased audience (more planning people than psychologist, for instance). Removing such bias should be strongly had in mind when organizing it.

- To develop a prototype demonstrator to test the feasibility of user control at different stages of the planning-driven workflow development and enactment. In order to do so, a prototype planning system would have to be built exhibiting different types of interaction, and then test with a set of people each setting. A simpler, more cost-effective approach, usually employed by the user interface community, would be to perform a "Wizard of Oz" study. Here, a human performs the role of the user interface, answering user queries as if a complete interface and system were on its place.

3.2.2. Infrastructure

This section focuses on the issue of the languages, interfaces, standards, and software tools that have to be used in order to integrate the P&S systems within the information system of any organization. The infrastructure can be divided into:

- an open, layered architecture and interface definitions that allow independently-developed modules to be combined in a flexible way; and

- the software modules that work within this architecture. PLANET's role should not be to favor one technology over another, or to enter the debate over free software, open-source software and/or proprietary software. Rather, it should aim to ensure that research, transfer and exploitation can be conducted effectively. Wherever possible, compliance with existing official and "de facto" standards and interoperability with existing solutions should be encouraged.

Current state of the art

The issue of the infrastructure related to a software tool to be used within an organization has many different views, given that this type of software has to interact with most software of the organization.

Therefore, it would be very difficult to describe all possible software interfaces that would have to be studied for integration. Here, we only mention some of the standards that are available and would most probably be the closest ones to consider.

The WfMC[3] has established a reference architecture based on five interfaces between workflow engines and other classes of associated software. Now is working on detailed definitions of these interfaces. Conformance to the WfMC standards by the major software developers is mixed, however. The interfaces they have defined are:

- Process model definition tools: defines the interface between modeling tools and workflow management tools. Basically, it uses WPDL (Workflow Process Definition Language), which is being re-designed towards XPDL (XML based). Since WPDL is stable, people willing to interface current planning and scheduling techniques or tools with workflow tools can already use this language for specifying the process models.

- users: defines how to provide information to users of the workflow tool with respect to the work (tasks, activities) that they have to accomplish

- automated systems: defines the interface with the other type of agent that can accomplish tasks, other software tools

- other workflow systems: given that currently many organizations are trying to interface their processes with suppliers and clients, it is needed to define the interface among the workflow management tools of the other organizations

- system administrator: any complex organization should have a workflow management administrator that is in charge of monitoring, and controlling the overall performance of the system.

The planning community on its side has defined a standard for domain specifications within planning tools, called PDDL [FL02]. This is based on the work of a committee of planning experts. The definition of this standard has helped on carrying on a planning competition [McD00, Bac01]. A second standard in the planning community has been during some time ADL (Action Description Language) [Ped89].

A three-four layer organization architecture has now become standard practice in industry. The layers consist of user client software, back-end applications, and one or two layers of so-called middle-ware. More effort will have to be devoted to understanding what level(s) are affected by introducing P&S tools in the loop, and how they are interfaced with tools in other layers.

There are various competing standards for middle-ware components including: CORBA (OMG), COM (Microsoft), java-based solutions (Sun Microsystems and others), etc. At some point it might be useful for AI P&S based tools to interface other systems using any of several of these components.

There are also a number of emerging standards for inter-operation of software agents from FIPA, DARPA (KQML/KIF), OMG, etc. Since the agent-based paradigm for programming of complex systems is gaining acceptance, it will be more important in the future to understand how to introduce P&S systems into agents, how several agents with planning and scheduling capabilities are able to

---

[3]http://www.wfmc.org

share their (partial) plans/schedules, or collaborate to generate plans that are going to be executed by several agents. There is already some preliminary work done with this respect [PSS$^+$99, CBMA01, DMAN01].

Requirements

A reference architecture and interface (language/API) covering modeling, build-time and execution is required to enable highly modular approach to research and application systems. A modular approach is necessary to facilitate:

- re-use of software and avoid wasteful duplication of effort;

- synergy between the work of research groups developing complementary technologies;

- exploitation of research results as add-on modules to "standard" software; and

- a steady flow of incremental enhancements from research into application.

The architecture must be compatible with the WfMC reference architecture and API and other standards.

Current research trends and active projects

It may be argued that software technology has developed by reducing the amount of information that a computational entity needs to know at runtime in order to be able to interoperate with other entities. Components have a more tightly defined interface than objects and provide inter-operation primitives in the form of events that pass highly informative objects to receiving entities. Software agents that use communication languages based on speech acts are sometimes presented as being the logical next extension of this trend. A number of collaborative and individual research projects have been and continue to be conducted in the application of agents to workflow management. Some of these collaborative projects include: ADEPT,[4] ENTERPRISE, TBPM (UK collaborative projects), EURESCOM Project P815, or SWIM.

Open issues

As it has already been said, it is important for PLANET to remain agnostic over middle-ware technology (CORBA vs. COM vs. Java Beans, .NET, Jini, etc.) until matters resolve themselves in the real world. However, to facilitate research collaboration, it would be better to pick one of these technologies on which to base a common research platform architecture. So, an open issue would be which one is closer to the demands of workflow tools and applications. Other related issues would be what type of structures have to be exchanged through different applications in relation to workflow and/or P&S tools.

---

[4] http://www.informatik.uni-ulm.de/dbis/f&l/forschung/workflow/ftext-adept_e.html

Research goals

The issue of infrastructure is strongly connected to development of tools. From the point of view of a research in AI, one potentially important goal would be the development of planning and scheduling servers that can be accessed by software components in the same way as back-end application software. Given that the P&S community already have a standard language, such as PDDL, it can facilitate the effort to have such a server application.

Recommended actions

Actions to be carried out with respect to infrastructure, relate to the previously discussed issues:

- draw up a reference architecture that can help foster the development of different modules that can be integrated in many different ways;

- agree on interface standards for research collaborations within the network, using the reference architecture as the baseline; or

- setting up a working group to establish interface standards for research cooperation with this respect.

3.2.3. Domain and business modeling

This section is concerned with methods, tools, languages, etc. used to model businesses and other application domains. Therefore, it has a strong connection to the Roadmap on Knowledge Engineering being developed within the corresponding PLANET TCU. Here, we will focus more on issues related to modeling languages, rather than on tools for knowledge acquisition and modeling. We direct interested readers to the Knowledge Engineering Roadmap, the DARPA initiative on Rapid Knowledge Formulation, or the recently created International Conference on Knowledge Capture.

Introduction

In business process management, the purpose is to design, improve or define more precisely the organization and its processes. In AI P&S, the purpose is to find a way of modeling the domain (and problems) that enables planning and scheduling techniques to be performed effectively. These purposes are entirely compatible, and there are many similarities between the representation languages used. In business process management the objective consists normally on defining a set of processes (divided in activities). Instead, in AI Planning, a space of plans/processes is defined (in terms of operators and/or task networks). That is, while in AI many different possible plans can be generated from the domain description, in BPM usually they only handle a small set of processes. With the growing requirement for flexibility and adaptability, modeling for process management is likely to move closer to the "planning" model in the future. There is also a difference, between business modeling and knowledge engineering, in the stage of development at which they are applied. Business modeling is performed when establishing requirements, whereas knowledge engineering is performed during early stages of software development.

Where workflow management systems are used, there are two distinct stages of modeling. The first is elicitation/documentation or design of the organization model (including business processes, resources and organizational structure). This may be documented, analyzed, and simulated using high level modeling tools such as ARIS[5] or the ones in `http://dmsweb.badm.sc.edu/bpr/aa-5.htm`. New work proposes the use of machine learning tools to elicit the models [vdAvDH+03]. In this setting, representation formalisms and storage formats tend to be proprietary. The second step is to produce a lower level model suitable for execution by a workflow engine. At this level, there is work by the WfMC towards an official vendor-neutral process definition language, though still different workflow products differ considerably in the style and syntax of input required. It is generally oriented to the requirements of automated execution of the flow of documents and control between task-performing resources. It says little or nothing about the nature and semantics of the tasks that are linked in this way. Although some high-level modeling tools do claim to generate workflow definitions for specific workflow engines, this capability is generally felt to be inadequate at present. Often engineers must write the workflow definition using a modeling tool associated with the workflow engine, using the high level model as a reference. It is hoped that tools from AI Planning may help bridge this gap.

If planning is to be used for BPR problems, the first step would be to think at a high level of what inputs of a planner correspond to the knowledge that BPR tools use, as well as what output of the planner corresponds to what knowledge on BPR tools. At a high level, one could establish the following relation:

Inputs of a planner:

Domain theory: usually composed of a set of operators in STRIPS-like (PDDL) language (described in terms of pre- and post-conditions). Each BPR domain (e.g. the accounting domain in an organization) can be defined in terms of a set of activities (here, the terminology can vary and use other words as tasks, or, even, processes) that are performed by organization agents (either human or software). Therefore, there is a strong relation between operators in planning and activities in BPR, but it is not clear yet how to go from an activity based representation (agents responsible of a task, resources to be used, time that it takes to perform it) to an operator based representation (pre- and post-conditions, and, in some cases, other issues such as time constraints).

Problem: in planning, problems are described in terms of an initial state and a set of goals. They represent particular instances of situations for which one would like to have a solution. For BPR, a problem might be described as a process that has to be designed (modeled) for a particular task to be performed within the organization. For instance, modeling the purchasing of an organization, or the process of installing a new telephone line at a given address.

Initial state: in planning, one has to specify the starting situation of the posed problem. In the case of the BPR domain one would have to represent all knowledge that the organization has about itself and can be used for the modeling of a specific process within the organization. For instance, the

---

[5]`http://www.ids-scheer.com`

hierarchical and/or functional representation of the organization, the resources that it can use in its processes, or the documents that are generated within the organization and travel around, being filled in, or filed, etc.

Goals: they describe in planning what one would like to be true at the end of the solution of the problem; that is, a set of assertions that have be true in a final state. In the case of BPR, this might be represented by the business goal of the organization with respect to that process. For instance, a purchase has to be done, having in mind a set of time or cost constraints.

Output of the planners: usually AI planners generate a plan or set of plans. A plan can be seen as an ordered set of operator applications that can lead from the initial state to a state in which the goals are reached. In the case of BPR, most processes are ordered sets of activities, adding conditional branches. Therefore, one would have to work on the generation of conditional plans, if a "typical" BPR model wants to be built.

There has been some preliminary work on automatically generating a planning domain theory, plus a problem description, from a representation of an organization set of activities, such as the one reported in [RMBM01].

Current state of the art

The state of the art is characterized by the large number and diversity of representation languages and modeling tools and techniques available. This may be indicative of the importance of the topic and that a definitive solution is still a long way off. Only a subset of what is available is presented, and some other references are provided. The issues of knowledge modeling in this field have many different perspectives. Here, some of this points of view are presented.

Process management Process management tools and languages focus on how to represent knowledge about how things are performed. There are many different modeling tools in the market already. They are often expensive and need skilled personnel, though there is considerable variation. A common criticism is that the modeling tools do not produce output in a format that is acceptable to process definition tools. Examples include: iThink (relatively simple to use, systems-oriented),[6] ARIS[7] and ARENA (both complex, include simulation tools), or ProSim/ProCap. For a more detailed list of tools, refer to `http://dmsweb.badm.sc.edu/bpr/aa-5.htm`. In [Ade03], one can find a comparative study of most of these tools.

Also, there is a variety of languages for describing such knowledge. Examples of standard process description languages include: Petri Nets, IDEFn,[8] PIF,[9] EPIF, PSL,[10] WPDL,[11] CPR (Core

---

[6]`http://www.hps-inc.com/bus\_solu/ithink/ithink.htm`

[7]`http://www.iwi.uni-sb.de/teaching/ARIS/aris-i/aris-e-i/index.htm`

[8]`http://www.idef.com/`

[9]`http://ccs.mit.edu/pif1.html`

[10]`http://www.mel.nist.gov/psl/`

[11]`http://www.wfmc.org`

Plan Representation),[12] and SPAR.[13] Apart from that, there are many other proprietary languages associated with particular tools.

Knowledge management   Knowledge management deals with the task of explicitly representing what an organization knows, not only about its environment, but also, and more importantly, about the organization itself. What the organization knows can be thought in terms of what the current people in the organization know, or even knowledge that people that have worked in the past in the organization and do not longer work there had. Perhaps, this has been one of AI's more important contributions to the field of computer science: what you know about the world should be *declarative and explicitly* represented in an inspectable format, so that you can later reason about that. Most current approaches to knowledge management rely on the creation and maintenance of an intranet with information about organization projects (past or present). In case the organization would have a well defined structure with such knowledge about the organization and its people, then we could reason about this knowledge when modeling organization processes. So, we could assign activities to the most appropriate people with respect to many different parameters.

AI planning   As we have already said, plan description/modeling languages include: ADL (Action Description Language), PDDL (Planning Description Domain Language), TF (O-Plan language),[14] the domain description language of IxTeT, HSTS-DDL,[15] the ones in[16] and theory of action formalisms. The knowledge engineering TCU agreed that PDDL was seen to be deficient in that it is not equipped with a methodology or language structure that helped the planning domain modeler. STRIPS/PDDL was likened to a "low level" language - theoretically expressive but not pragmatically expressive enough. Also, the underlying STRIPS-assumptions were thought to restrict the usefulness of the language. Some effort is being devoted currently to developing modeling tools for use with AI planners. An example is the PLANFORM project that has generated the GIPO tool [SMZ+01], developed by people from Universities of Durham and Huddersfield at UK.[17]

Another alternative way of representing domain models in planning has been through the use of cases, and using a problem solving approach named Case Based Reasoning (CBR) [Ham86, Kam89, Kol93, Vel94]. Instead of (or in addition to) explicitly representing the domain actions, they dynamically re-use and adapt past solutions (process models in the workflow domain) for generating solutions to new problems. This is a potentially useful approach for process models as, in some companies, process models are represented as templates that are re-used for new processes generation.

Ontologies   Ontologies are a way of expressing what is known about a given domain by using a hierarchy of concepts, their relations, attributes, and a set of axioms [HH00, DFvH02]. Perhaps, the most well known ontology effort has been CYC, developed by Douglas Lenat and his group [LG90]. Work on organization ontologies has been conducted by several groups. We could highlight in relation to organizations modeling:

---

[12]http://projects.teknowledge.com/CPR2/

[13]http://www.aiai.ed.ac.uk/~arpi/spar/

[14]http://www.aiai.ed.ac.uk/~oplan/

[15]http://www.cs.cmu.edu/afs/cs/project/ozone/www/PCP/hsts.html

[16]http://www.informatik.uni-ulm.de/ki/Biundo/publications/publications.html

[17]http://scom.hud.ac.uk/planform/

- TOVE project at University of Toronto [FG98].[18]

- Enterprise [Sta97] and TBPM [SMC$^+$00] projects at AIAI, University of Edinburgh.[19]

- MIT's Process Handbook is an evolving repository of business process knowledge. It is also available in "shell" form for organizations to populate with their own knowledge [MCL$^+$99][20]

A lot of work on ontologies has been sponsored by DARPA, e.g. the knowledge sharing effort, though it is not clear whether organization modeling has been addressed specifically. See also the proceedings of the AAAI 94 workshop on AI and business process re-engineering for other approaches.

Software/knowledge engineering   Software engineering modeling languages and knowledge representation techniques also cover similar ground. The Unified Modeling Language (UML) has become dominant in object oriented software engineering and is increasingly being used in business modeling too. Recent extensions improve its usefulness in modeling processes. The CoRE (Controlled Requirements Expression) method originally developed by British Aerospace (Military Aircraft) and Systems Designers has been used by AIAI in conjunction with TF/O-Plan and is also the basis of the COGSYS EnCore tool for requirements engineering. The best-known knowledge engineering method is CommonKADS [SWB93].[21]

From the connection with new ways of making business, through the Web, new standards on languages should be cited here. In the future, more and more organizations will describe, at some level of detail, their processes using formalisms developed for Web-based information exchange. The most well known example now is XML (eXtended Markup Language), though new ways of representing things are appearing in combination to ontologies standards, such as RDF, DAIML, OIL, or DAIML-OIL.

Requirements

A long term requirement would be the integration of (most) languages and standards to improve interoperability; that is be able to generate tools that can handle several of these languages when using only one (by creating the appropriate interfaces, for instance). This would also allow to minimize requirements for re-training when changing to a new tool.

In relation to the usability aspect, a requirement would be to generate easy to use tools combining modeling and simulation capabilities. It seems like current tools are either very easy to use, but with a very poor representation mechanism underlying the tool, or have a powerful set of tools, but are too complex to be used without extended training.

Another requirement is the ability to integrate AI P&S domain models languages with the ones needed for process definition tools. This would allow to use their features, improving the usability ratio. Also, it would allow to re-use the already available processes that had been generated with those tools.

---

[18] http://www.eil.utoronto.ca/eil.html
[19] http://www.aiai.ed.ac.uk/
[20] http://ccs.mit.edu/ph/
[21] http://www.commonkads.uva.nl/

Current research trends and active projects

A lot of work is currently devoted to defining web-based languages, in the context of the Semantic Web. Perhaps, this is the more active point of view related to knowledge modeling, so future applications and tools will certainly used some form of language or standard arising from this field. These efforts have been described in the section on current state of the art. With respect to the workflow management world, as it has been mentioned before, the WfMC is working on a reference standard that would help on defining the interfaces of workflow tools with other systems, or among them.[22]

Open issues

Given that the generation of common interfaces, languages and standards is a very active field now due to the Web efforts, the open issues relate to how to integrate those efforts to workflow modeling and AI P&S modeling. So, it would be needed to answer to the following question: what is the best way to synthesize business process management, AI P&S and ontology modeling languages?

Also, domain experts have stated a requirement for new simulation tools. It is not clear why they believe this is needed. Can we clarify the requirement given that powerful simulation tools do exist? Why are existing tools not good enough? It is the price or the ease of use?

Research goals

Related to what has been said, a research goal would be to define a modeling language for organizations and their activities that has, at least, the following set of features:

- domain experts are comfortable with: they can understand it and write in it;

- has a rigorous semantics;

- has textual and graphical representations, so that there is consistency; between this two types of presenting information to the users of workflow tools

- is executable, for simulation purposes;

- can be easily (syntactically) translated into other standard formalisms;

- is mathematically formal so that the (static and dynamic) properties of models can be analyzed; and

- planning techniques can be applied to.

Another goal would be defining means of creating and managing a library of processes in the spirit of the field of CBR. This should include, among other features, means of verification and analysis of redundancy, fast and appropriate retrieval of related processes, or easy update and version control of stored processes.

---

[22]http://www.wfmc.org

Recommended actions

In order to generate the above mentioned modeling language and Case Based approach, some potential actions to be taken could be:

- define a taxonomy of description languages, which includes features from current "de facto" standard languages, plus features from related fields, such as Semantic Web languages, or XPDL from WfMC.

- define a list of requirements of representational mechanisms that are needed to support various workflow capabilities. This would be based on the work done by the WfMC with their various standardization approaches

- write a report documenting an agreed comparison and classification of Process Management, AI P&S and Ontology modeling languages

- write a comparative survey of existing software tools

- organization of a "hands-on" workshop on languages for modeling in relation to the fields of AI P&S, the Web, e-Commerce, or workflow management. A preliminary workshop has been organized held in conjunction with ECAI'02. See TCU Web page for a report on that workshop.

3.2.4. Planning and scheduling

In this section, we will only describe from a high level what interactions may have AI P&S with planning and scheduling from the point of view of workflow management.

Introduction

The field of AI Planning and Scheduling is a very active one currently, mainly due to the arrival of new techniques that have enormously increased the solvability horizon of planners. We will not overview here all that has been done in the field. A good overview of the field until its publication can be found in [AHe90]. For an updated account of research that is going on since then, one should consult recent conference proceedings in the field, such as ECP, AIPS, AAAI, or IJCAI. Also, relevant papers are published in journals such as *Artificial Intelligence Journal* or *Journal of Artificial Intelligence Research*. Finally, some authors are currently preparing text books on the subject, such as Ghallab, Nau and Traverso on one side, and Kambhampati on the other.

There are two aspects that are relevant in the relation of AI P&S to workflow management applications.

In many workflow applications, combinatorics are significant. It is usual to see, in a medium-big organization, many people involved (or potentially involved) in different processes. Generating all possible combinations of assignments of people to tasks and computing the optimal combination is an NP-hard problem. AI P&S due to its heuristic nature can help on reducing the complexity of this process, even if an optimal solution is not always found.

As explained in the section on Human issues, in many cases it is very important the support to users decisions, rather than just the automation of their tasks. In order not to help the users, it is sometimes better leaving them with options and showing them alternatives, than providing them with a fixed set of tasks in a fixed order. Given that one of the main claims of AI is the explicit representation of knowledge, this can help on providing explanations to users, or computing different alternative solutions with different quality criteria (as it will be discussed in Section 3.2.6).

Current state of the art

Even if, the field of AI P&S is a very active one, from an application point of view, its transfer to industry is rather slow. Some characteristics of the current state of the art are the following:

There are many planning techniques already and software freely available.[23] Every two years a planning competition is held with better and better techniques, whose implementations are made publicly available. However, these techniques and systems need to be applied by an expert on planning technology due to the lack of interfaces, and the strong knowledge on the technique that is needed to understand which technique to apply for each problem, and how to make better use of it (parameterization). That is, the application of current AI P&S technology has to be done on a case-by-case basis.

There are, though, some commercial scheduling tools (e.g. ILOG Schedule). These are not entirely general purpose, so they need to be customized for new applications. Also, they need to be used by experts in the tool. With respect to applications, there have been already some successful industrial applications of planning and scheduling, which have shown its profitability. From elevators, satellites, or robots control to the design of ships, there is a wide range of systems that are being used (or have been used) in the industry.

Requirements

From all that has already been said, it is obvious that the next step consists on the integration of scheduling into process management (and execution) tools. So, we have to understand what steps to follow in order to incorporate this AI techniques into currently used workflow tools. One of the roles of PLANET consists on attracting people from industry, so that they understand the usefulness of AI techniques, so industry and academia are able to generate such integration. On one side, workflow tools have to provide a high level description that AI P&S techniques can use to generate plans+schedules. On the other side, AI P&S techniques have to be able to handle in an integrated way plans, resources and time, which is crucial for most workflow applications.

Current research trends and active projects

There is a large list of current research trends in planning that would take long to detail. Some of the most well known techniques and approaches currently are:

---

[23]See PLANET repository at `http://scom.hud.ac.uk/planet/repository`

- Integrated planning and scheduling. Industrial applications require that planning and scheduling not be two separated processes, but they are integrated one way or another. Examples of known integrated systems can be found in [GL94, Mus94, BK98].

- Fully instantiated models of planning. They rely on some form of search for solutions in the space of instantiated operators and states. Examples are the SATplan [KS96], Graphplan [BF95], or HSP [BG01] approaches.

- Generating schedules that guarantee certain behaviors at run-time, i.e. that are robust against limited changes in the environment. This is very important for workflow, given that there is a lot of interaction with the environment through the execution of tasks that people carry out.

- Constraint based approach to scheduling. This is related to the PLANET TCU on Dynamic Scheduling, so we refer the reader for more information to this document.

- Mixed initiative planning. The goal of these approaches is to introduce the human in the modeling loop. This has been described on Section 3.2.1.

- Decision-theoretic planning. In many domains, the world is uncertain and/or non deterministic (execution of the same action does not always arrive to the same state). In those domains, it is needed to explicitly reason about probabilities of fluents to be true, as well as actions to cause certain effects. One way to approach this type of planning task is by representing the problem as a Markov Decision Problem (MDP) and solving it by using either Dynamic Programming [Bel57] or using Reinforcement learning [KLM96]. Some approaches to solve this type of problems can also be found in [Bly94, BDH99].

- Model checking. It is based on the idea that planning can be formulated as verifying some formal properties of a formula [GT99].

- Applications. We are currently witnessing many interesting real applications of AI P&S technology. Among them, we can cite the projects at Nasa-JPL (such as Mars missions, imaging sequencing of telescopes, or the Deep Space One experiment) [MNPW98, MS97], DARPA projects (such as MOABS, rescue projects, or scheduling of military operations), virtual agents projects (such as the travel planners, or the Electronic Elves [CGK$^+$02]), elevator control, or satellites control.

Open issues

In relation to planning and scheduling for workflow management, one of the main issue relates to how to best combine human capabilities with planning and scheduling. This is related to mixed-initiative approaches in the sense of deciding what is the type of work that can be automated, and which is the type of work that should not be automated. There might be different reasons for not implementing fully automated systems, such as humans like doing some planning and scheduling tasks, or they like to be in control of some issues. In those cases, AI people should get software to work out the combinatorics and present results for person to use or modify.

Research goals

The goals for AI P&S in this context are related to these open issues. Perhaps a crucial goal is creating the appropriate interfaces, so that planners are easy to use for domain experts. This could imply limited capability, though.

Recommended actions

One action that remains to be performed consists on the definition of graduated reference problems in the field of workflow planning and scheduling. This is one of the goals of this TCU for the next future, that we hope to achieve at least partially by the end of the NoE period.

3.2.5. Enactment / execution

This section describes the issues related to the application of a pre-defined process model (or plan) in the real environment.

Introduction

If we are dealing with workflow tools, enactment deals with the application of a process model in the target organization. In case we are dealing with planning, execution deals with the application of plans to achieve the proposed goals. There are many common aspects that emerge from the comparison between the enactment of workflow processes and the execution of plans, such as tasks like monitoring, control, exception handling, adaptation, or interaction with the environment.

Current state of the art

Most of the current work on execution of plans belongs to the field of robot planning or integrated manufacturing, given that most work on planning has been devoted to developing faster or more powerful planners, and there have been very little applications of planning techniques within organizations. Within AI P&S, there has been work done in the following issues related to execution:

Conditional planning: In many domains it is very hard to think "a priori" about all possible outcomes of the actions in the environment, or the agents need to gather information (sense the environment) while they are executing a plan in order to know what should be performed next. Conditional planning generates plans with branches for solving this problem [DHW94, KHW95, PS92]. When executing a plan, every time alternatives are found, the current state of execution is consulted and one branch is selected.

Decision-theoretic planning: This has been described in Section 3.2.4 on Planning and Scheduling.

Reactive planning: Most work in robotic tasks deals with two types of planning: deliberative and reactive [Bro86]. Deliberative planning is used to generate high level descriptions of ordered sets of actions to be applied, without consideration of the actual details of the plans. When

execution begins, control is assigned to a reactive component that decides what to do in the real environment. It selects the next action to be performed according to the current state of the system and the desired goals. In many cases, the reasoning is as simple as a pre-defined algorithm, and in other cases, it performs a very narrow local search to decide what to do next. Usually, the reactive behavior has been learned by using many different techniques, although in some systems it is an "ad-hoc" procedure built from scratch. The PLANET Roadmap of the TCU of Robot Planning provides an extensive description of this type of planning approach. Examples of pure reactive, or hybrid deliberative-reactive planning systems are [AB97, Fir96, GL87, Kae87].

Integrated architectures for continuous processing, by interleaving planning, scheduling, execution and repair. These architectures have been mostly studied in the robotics field, and they can be very relevant for the workflow domain. Examples are 3T [KSB98], Saphira [KMRS97], RAX/PS [BDF$^+$98], ASPEN [CRK$^+$00], Cypress [WMLW95], or TRIPS [ABK$^+$02].

Requirements

In order to apply planning-execution techniques to enactment of workflow models, there is a need to consider the following set of requirements:

Techniques for monitoring execution: given the complexity of many organization processes, it is very important to be able to continuously look at the enactment of the processes. Monitoring should report on tasks that are being delayed, aborted, or resumed. Sometimes the identification of any of these issues is very difficult. For instance, it is very common that people forget to notify the computer (monitoring software) the completion of tasks, or changes in tasks development. Also, in most situations, people tend to delay the execution of tasks as much as possible, allowing very few time for reaction.

Techniques for exception handling: related to the previous requirement, once a problem is encountered, there is a need of defining procedures for recovery of the flaw. There have been some work in the field of planning with respect to re-planning, and recovery from failures that could be of some help to workflow failures. From the point of view of workflow systems, exception handling is usually performed by "ad-hoc" procedures that applied given that a problem is detected. In contrast, by allowing a declarative representation of operators, the system might be able to reason about possible failures and how to solve them.

Open issues, research goals, and recommended actions

Analyzing the current research trends and the state of the art on the execution of plans, the following is a set of subjects that are pending to be solved:

How to combine user preferences Usually a process that is being enacted is composed of many tasks to be performed by people with different roles and different qualifications. Assigning a task to a human is usually performed having in mind the set of roles that s/he is able to perform. However, there are other aspects that are worth considering such as user competence (even if a person is able to perform a given task, in what type of tasks is s/he really good at?), or user preferences (what type of tasks does s/he really enjoy performing?). Also, given that some tasks have to be performed by a group, this arises issues such as how to arrange the most productive group. Some of this issues are strongly related to the currently very intensive field of knowledge management. This generates the goal of developing theories that define and reason about user models with respect to the assignment of tasks

Flexible working with overall plan In most cases, detailing all possible aspects of the tasks can cause users to loose interest on their work. It is a better policy to provide some level of freedom for the execution of tasks, allowing decisions to be made by the human when executing a task. This is analogous to the integration of deliberative and reactive planning in robot tasks. A given degree of reactivity, allows the robot to be better prepared to cope with uncertainty and non determinism. A given degree of freedom, also allows the user to be prepared for uncertainty and non determinism, but also influences his/her way of looking at the work. The issue is how to combine the overall plan with the specific interests of the humans that have to carry out the plan steps. The associated goal would be the definition of models that allow to generate processes at various levels of detail, and interleave the execution of a high level plan with a somewhat reactive component.

Can plan repair or re-planning techniques be helpful in exception handling / jeopardy management? As mentioned before, there has been some work done from the planning perspective with respect to handling plan failures during execution. It is not clear how this work should help and/or influence the workflow jeopardy or failures. It would be needed to study the sets of possible failures that can occur within the enactment of a process, and the set of repair procedures for those failures.

How to provide personalized views of processes (visualization of big picture) Another of the features that users find very important when performing a task of a process is knowing issues such as: why am I doing this?, where does this document come from?, or who should read this document afterwards? All of them deal with the problem of giving the users the ability to inspect at a certain level of detail the connections between the activity they are performing and the overall picture of the whole set of processes of the organization. A research goal in this respect would be the description of variable visualization techniques for parts of processes and the relationships among the processes of an organization, having in mind security issues.

How to combine and interleave plans for multiple humans (agents) If a distributed plan has been generated, the execution of that plan should monitor the interactions of the plans for each agent and combine the executions in the most effective way. Also, it should solve problems arising from the failure in an agent plan that has connection with other agents plans. The definition of a protocol of communication and negotiation between agents plans, execution of plans, failures, and repair methods would be needed.

3.2.6.   Adaptation, optimization and metrics

In this section, we will discuss an increasingly important aspect of workflow enhancement: how processes can be optimized/adapted according to design or enhancement problems. We will also discuss about the metrics considered for changing the processes.

Introduction

In general, there are two places in the application of workflow technology to organization processes in which changes to the processes are involved:

Design phase: When designing a given set of processes, the user might want to obtain an optimal process model according to a set of metrics and constraints. Usually, time and cost have been the only metrics considered for optimization. Also, optimization has been mainly a manual process, helped by the use of (sophisticated) simulation and analysis tools.

Enhancement phase: When a process is being enhanced, many mismatches (might) occur between the designed process and its actual implementation. The role of adaptive workflow would be to feed the design and/or enhancement with those mismatches in order to optimize/adapt the process to the real situations.

Following the analogy between the process of applying planning technology and workflow technology presented in previous sections, there are several aspects that workflow and planning have (or not) in common with respect to optimization:

Design phase: The goal of both tasks (planning and workflow enhancement) is to obtain a process (plan) to be enhanced (executed) in the "real" world. However, while workflow has always considered optimization (of time and cost) as a part of its design phase, it has not always been the case for planning. In the case of planning, the main emphasis has traditionally been on satisfying a goal, rather than on finding an optimal plan. This is mainly so, due to the already inherent complexity of finding "a" plan in many complex problems. When plan quality is considered, it has been mainly computed as "plan length", instead of using any user defined metric.

However, there is a growing interest in the planning community for solving problems searching for optimality, or at least for better solutions [DKR02, Nar01]. In some cases, planners try to find an optimal plan according to a predefined criteria, such as make-span (total time to execute a partially ordered plan) or number of steps in the solution (in case of a totally ordered plan) [HG00, WH94]. In other cases, they take a plan as input and try to improve it [AK01]. Others learn control knowledge to guide the planner towards "good" solutions [ABI02, BV97, EM96, Iwa94, PC94, RK92].

Enhancement phase: The second main goal of both tasks is to enhance (execute) the designed process (plan). Here, we also find some differences between workflow and planning. Workflow enhancement is currently very widely done, so most organizations that have been (re-)designing their processes are following them. However, very few applications of planning

systems have been built and used. Therefore, from the optimization/adaptation point of view, there are many more lessons to be learned from workflow applications than from planning applications. Since optimization/adaptation coming from the enhancement (execution) needs to know what types of failures can occur within the execution of a process (plan), we might have more information coming from workflow.

Listing all possible metrics is an infinite task. However, there are some that have been considered in many applications:

- Cost: measured by whatever means. Currently, ABC analysis is commonly carried out within business processes.

- Time: usually measured as time steps of the process, or the make-span.

- Quality: e.g. defect rate in a product, delays and dropped packets in a network.

- Value of the end-product: e.g. adding an extra processing stage may increase the value of the end product more than it increases the cost.

- Flexibility: the ability to change processes quickly is important. Processes that are highly optimized with respect to cost or time may well be inflexible.

- Robustness: the probability of success of the processes.

A related issue is the use metrics to motivate and assess the performance of people. Inappropriate metrics can have the opposite effect to that intended. For example if targets are perceived as impossible, then people will ignore them. Thus if a target is made more demanding it may in fact decrease performance. Similarly, taking a call center as an example, an "obvious" performance metric is number of calls handled per day. However, this encourages staff to keep calls short, which may mean that poor answers are given leading to more calls. This improves apparent productivity, but customer satisfaction goes down. The "correct" productivity metric must take into account whether the caller was satisfied, but this is more difficult to measure.

In the next sections, we discuss issues related to optimization with respect to: open questions; research results; barriers to technology transfer; and software and application requirements.

Current state of the art

The following is a set of results that might be used to approach the open questions of previous section:

- There are all types of mature optimization techniques coming from AI and operations research such as: heuristic search; genetic algorithms; or linear/dynamic programming.

- There have been some approaches on planning for better solutions and learning to plan for better solutions that have been mentioned in the introduction of this section.

- Also, recently there is an interest towards using multiple criteria and considering them for planning or scheduling [DKR02].

Requirements

Here, we discuss what the workflow tools and applications should have in order to allow optimization:

- Integration with process design and enhancement tools: optimization and adaptation procedures should be integrated on one hand with process modeling techniques (for obtaining good models), and, on the other, with process enactment tools (for adapting the models according to actual enactment of the processes)

- Interaction with the user: an important aspect of the tools consists on allowing the user to interact with the optimization and adaptation procedures so that s/he is able to direct towards process models that comply with user expectations

- User-definable metrics and optimization parameters: the user should be able to provide in a given language descriptions on how metrics should be computed, as well as parameters for controlling how optimization and adaptation should be performed

Open issues, research goals, and recommended actions

The set of open questions with respect to optimization/adaptation and metrics are:

- Do workflow applications really need metrics different than time and cost? If we are going to define tools for performing adaptation/optimization according to user defined metrics, we should first make sure that users will need different types of metrics. A possible recommended action would be to survey in some organizations about this aspect.

- What language should we use to provide those metrics to the system? We should study what are good languages for describing those metrics, so that potential users of the tools are able to easily define metrics by themselves. PDDL2.1 has advanced on defining such language, by allowing the user to specify them in the language. Other planner-specific approaches allow also defining quality-based criteria [BVV01].

- If multiple agents are used, how should their respective metrics be combined/negotiated? Should it be left to execution time or should it be worked out before execution starts?

- What is the set of possible failures of a process (plan)? Although this question also appears in the section on execution, within this section, it refers to the generation of plans that are optimized according to, for instance, less probability of failure

- How should workflow enhancement influence optimization/adaptation? This issue is related to the plan repair techniques in the execution section

- Where should design/enhancement optimization knowledge come from? There might be three different types of sources: experts on a given domain (they usually know what models are wrong and why, what resources should be assigned to what task, etc.); experts on BPR or

workflow enactment (usually they work on consultancy firms and provide advice on how different organizations implement their processes); learning from past executions of the workflow or from the history of the processes execution in the organization.

- Are there experts on resolving failures of execution, or anticipating problems? This issue is related to the previous one. Usually, in big organizations there is people in charge of this task that could be of great help

- Can the systems recognize a "good" solution? Or how do we define procedures for computing how good a model is?

- How should the interaction with the user be integrated when optimizing? Optimizing a process might result in a less intelligible process, so an analysis on what is preferred.

When trying to apply optimization to process design/enhancement, the following is a list of possible and actual problems:

- The user might not know/distinguish when s/he needs optimization.

- How does the user describe optimization and metrics knowledge?

3.3. Summary and conclusions

This document has presented the PLANET R&D Roadmap for AI Planning and Scheduling applied to Workflow Management. In an applied discipline such as this, a Roadmap must not only identify research challenges, but also match them to current and projected end-user requirements. It must also consider the process by which the results are incorporated into the tools of the trade of the end-users and application developers. Furthermore, necessary preconditions for successful application of the results must be taken into account. This Roadmap is an important step towards a coherent strategy, but is not itself the definitive answer. The Roadmap needs to be a living document that is developed and updated and regular intervals.

Main achievements

One of the main achievements to date has been to develop an understanding of how the "world view", vocabulary, challenges, etc. of Business Process / Workflow Management relate to AI Planning and Scheduling. This has been possible because of the active participation of a small number of workflow and process management experts from end-user organizations and consultancy companies. The site visit to BT to gather information on existing (non-AI) software applications was also extremely valuable in this regard. For planning techniques to be of practical use they must be integrated with, or must interface to, commercial workflow management systems (WfMS) and other related software.

Requirements have been classified as short, medium and long term as follows:

short term address short-comings in current-generation process management software. The most important items in this category are: integration of scheduling and resource allocation/management algorithms into workflow management software; and incorporation of a planning capability to enable a WfMS to modify the process instance automatically during execution, to cope with failure, changed objectives, and other exceptions.

medium term current generation workflow software handles high volume routine processes, typically involving low-skill workers. The medium term requirements concern extending this support to high-skill knowledge workers. This may involve, for example, building process awareness into software tools.

long term more radical (e.g. adaptive self-organizing) approaches addressing the need for organizations to function in a business environment that is increasingly uncertain and subject to change.

This document has also made a start on identifying planning techniques and research goals that address these requirements. In addition to the application of planning and scheduling algorithms we discussed: advantages to be gained from using AI plan representations for processes, ideas from plan execution (especially in uncertain environments), and work on adaptation optimization and metrics. Further work remains to be done, however, to identify specific research goals and projects. Two further topics are also discussed: human issues and infrastructure. It is important to remember that much of the work in a business process is performed by people. Often technology is seen primarily as a means of cutting costs through automation rather than enhancing value by enabling people to work more effectively. The result of treating people like machines is often de-motivation, high staff turnover, loss of productivity, etc. In addition, human qualities are under-utilized. There is a danger that must be guarded against that planning and scheduling techniques may make this situation worse rather than better. The discussion of infrastructure mainly focuses on the need for a reference architecture and interface standards to AI-based software tools to be integrated with each other, with conventional process management software, and with the general organization infrastructure.

Summary of requirements

There now follow descriptions of areas in which current business process management is recognized to be deficient. The list is not exhaustive, and we invite proposals for additions to the list. The requirements are into short, medium and long time scale categories. The short term requirements concern ways in which current practice and tools can be improved. The medium term requirements concern extension of workflow-related support into classes of processes and users that are not catered for by current workflow systems. The long term requirements concern the need for a more radical re-think of how organizations and their software support infrastructure are organized.

Short term

The following are seen as short-comings in current-generation process management software. They are presented in approximate order of importance, though the first two are of comparable ranking.

- Integration of scheduling and resource allocation/management algorithms into workflow management software. Current workflow management software automates the flow of work items between work queues according to pre-determined rules. It does not deal with allocation of resources to tasks or take resource availability into account in prioritizing or scheduling the work.

- Re-planning. There is a requirement for incorporating an ability to modify the process instance automatically during execution, to cope with failure, changed objectives, and other exceptions. This could be done by altering the process instance plan being executed (inserting and deleting steps) or by creating and executing an ancillary plan (conditional plan) containing the additional process steps.

- Generation of workflow definitions from high-level process models. Process modeling tools work with relatively high level process definitions, whereas workflow management systems require low level definitions. Current generation tools do not do a good job of bridging this gap. Tools are required that automatically generate low-level definitions that can be input directly to workflow management systems. The ability to do this in reverse is also desirable.

- The ability to feed data captured in the workflow engine back into the modeling and simulation tool to improve modeling at that level. Workflow engines capture a great deal of data in the course of enacting process instances. This contains useful information latent within it, but it is rare that data mining techniques are used on it.

Medium term

- Process support for intermediate level and knowledge workers: current workflow and groupware systems "pick the low-hanging fruit", that is they automate that which is easy to automate - enactment of routine processes and providing information-sharing and communication services. The tasks performed within these processes are routine also, and are performed by relatively low skill workers. There is very little process-related support available for high skill (professional) knowledge workers or workers at intermediate skill levels . "Process aware" and "knowledge aware" support to enhance the effectiveness of intermediate and high-value knowledge workers is required, but much more difficult to achieve.

- Empowerment of users: current workflow management systems are suitable for routine processes and demand uniformity from users, effectively expecting them to behave like machines. This makes poor use of human abilities, even in the case of low-skill workers, and can have a de-motivating effect. Informing people about the context of their work is a necessary short-term requirement, which should eventually lead to systems that encourage and support initiative, but would require workflow systems to be enriched with a semantic knowledge on the processes they enact. Future systems need to assist people in achieving their potential in their roles, which means encouraging initiative and adapting to human diversity rather than enforcing regimentation.

- Visual representations of the current status of a process instance, so that workers within a process can see how their activities fit into the "big picture" [Zub88]. This is actually intermediate

between the "short term" requirements, which might be satisfied by incremental additions to current generation workflow, and the "medium term" ones which require a change in philosophy.

Long term

The following are factors driving process management development in the long term. Mostly they concern the need for organizations to function in a business environment that is increasingly uncertain and subject to change.

- Flexibility: one of the main drivers in process management is the need to be able to get new products and services to market quickly. This means that an organization and its supporting infrastructure must be capable of enacting a wide variety of processes, with the actual set of processes active at a given time being easily changed.

- Evolvability: no matter how flexible an organization is in the short term it will have to change in the longer term in response to changing markets, technology, etc. Change takes time, however, and the organization must continue to operate. The organization needs to be capable of gradual evolutionary change to avoid the current problems with legacy systems recurring in the future.

- Adaptiveness: currently, organizations and their business processes are seen as basically static, but subject to occasional discrete changes such as re-organization or introduction of a new product and/or process. However, the frequency of change is increasing. In the future organizational models will be needed in which continuous change is the normal state of affairs. Such models most incorporate processes that "sense" the drivers for change (e.g. increasing demand for a product) and cause appropriate changes to the organization model. Organization software infrastructure will need to support such dynamic organizational models.

- Decentralized management: a management paradigm shift is currently under way motivated by the need for flexibility, evolvability and adaptiveness. This is variously described as a move from centralized to decentralized management, from management push to market pull, and from plan and build to sense and respond. This involves moving decision-making responsibility from central management to autonomous local units. The behavior of the organization as a whole is then the cumulative result of local decisions. The role of higher management is then one of defining performance metrics and other incentives by which local managers make decisions, and also of providing means by which the autonomous units can interact constructively. Workflow management systems are very much tied into a plan and build management style. A new approach to software infrastructure is required to support decentralized management. An agent-based approach seems well suited in this respect. Use of an agent-based approach does not in itself guarantee the benefits sought from decentralization, however. A better understanding of how to apply agents and agent-based approaches to achieve the benefits is still required.

Dynamically changing organizations of the future will involve forming opportunistic organizational structures and dynamic supply chains. Theoretical work in Virtual Organizations and "switching" is related to planning approaches and can be used to build the workflow systems of the future [Mow01].

Recommended actions

This section highlights a set of actions that will be able to attain some of the objectives (requirements) that were defined in the previous section.

- The potential for benefit from applying existing AI P&S techniques to short term requirements should be explored through case studies of large-scale real problems conducted jointly with domain experts and process management experts. It is apparent that process management problems can be posed as planning and scheduling problems. However, for the techniques to be adopted, it must be shown that they result in a significant benefit compared to current practices in the context of realistic business processes. The techniques must also integrate with other components of organization software, and be usable by the typical software or business process engineer. PLANET cannot perform such case studies, but it should facilitate and encourage them. It should also publicize the results.

- Generally, links with process management and software engineering communities need to be strengthened by means of interdisciplinary events and other measures. Within the second phase of PLANET, PLANET II, a set of workshops have been organized in conjunction with people from WfMC and other relevant communities in order to further understand the gap between the two fields.

- To encourage case studies and increase awareness of process management challenges within the planning community, PLANET should collect examples of process management problem domains.

Main recommendations

The TCU main objective was to play a useful role in closing the gap between industry and academic research. However further work is needed:

- to make researchers aware of the real challenges and constraints of the workflow domain;

- to make application and tool developers aware of what AI Planning and Scheduling research has to offer;

- to address practical issues of integrating planning and scheduling technology into suites of application software, and of making the techniques usable by typical software engineers, analysts, etc.

- to form a consensus on medium and long term research goals. The Roadmap should be seen as a living document and be extended and updated regularly.

Further reading and links

Perhaps, the current best text for an overview of the relation between AI P&S and workflow management is the paper by Myers and Berry [MB99]. With respect to the field of workflow management

and business process re-engineering, some overviews can be found in [vdAvH02, AAAM97, DG95, JEJ95, Mow94]. In the Web page of the TCU,[24] one can find more links to Workflow Management related Web pages and documents.

---

[24]http://scalab.uc3m.es/~dborrajo/planet/wm-tcu/

Bibliography

[AAAM97]   Gustavo Alonso, Divyakant Agrawal, Amr El Abbadi, and C. Mohan. Functionality and limitations of current workflow management systems. *IEEE Expert*, 12(5), 1997.

[AB97]   Ronald C. Arkin and Tucker R. Balch. AuRA: principles and practice in review. *Journal of Experimental and Theoretical Artificial Intelligence*, 9(2), 1997.

[AB02]   Ruth Aylett and Daniel Borrajo. Workflow and intelligent manufacturing: Common problems, common solutions? *Planet News*, 3:25–26, 2002.

[ABCSA02]   Ricardo Aler, Daniel Borrajo, David Camacho, and Almudena Sierra-Alonso. A knowledge-based approach for business process reengineering, SHAMASH. *Knowledge Based Systems*, 15(8):473–483, 2002.

[ABI02]   Ricardo Aler, Daniel Borrajo, and Pedro Isasi. Using genetic programming to learn and improve control knowledge. *Artificial Intelligence*, 141(1-2):29–56, October 2002.

[ABK⁺02]   José Luis Ambite, Greg Barish, Craig A. Knoblock, Maria Muslea, Jean Oh, and Steven Minton. Getting from here to there: Interactive planning and agent execution for optimizing travel. In *Fourteenth Innovative Applications of Artificial Intelligence Conference (IAAI)*, Edmonton, Alberta, Canada, 2002. AAAI, AAAI Press.

[Ade03]   Martin Ader. *Workflow & Business Process Management Comparative Study*. Workflow & Groupware Stratégies, 2003.

[AHe90]   James F. Allen, James Hendler, and Austin Tate (eds.). *Readings in Planning*. Morgan Kaufmann, 1990.

[AK01]   José Luis Ambite and Craig A. Knoblock. Planning by rewriting. *Journal of Artificial Intelligence Research*, 15:207–261, 2001.

[ASF⁺95]   James F. Allen, Lenhart K. Schubert, George Ferguson, Peter Heeman, Chung Hee Hwang, Tsuneaki Kato, Marc Light, Nathaniel G. Martin, Bradford W. Miller, Massimo Poesio, and David R. Traum. The TRAINS project: A case study in building a conversational planning agent. *Journal of Experimental and Theoretical AI (JETAI)*, 7:7–48, 1995.

[Bac01]   Fahiem Bacchus. AIPS'00 planning competition. the fifth international conference on artificial intelligence planning and scheduling systems. *AI Magazine*, 22(3):47–56, Fall 2001.

[BDF⁺98]   D. Bernard, G. Dorais, C. Fry, E. Jr, B. Kanefsky, J. Kurien, W. Millar, N. Muscettola, P. Nayak, B. Pell, K. Rajan, N. Rouquette, B. Smith, and B. Williams. Design of the remote agent experiment for spacecraft autonomy. In *Proceedings of the IEEE Aerospace Conference*, Snowmass, CO (USA), 1998.

[BDH99]   Criag Boutilier, Thomas Dean, and Steve Hanks. Decision-theoretic planning: Structural assumptions and computational leverage. *Journal of Artificial Intelligence Research*, 11:1–94, 1999.

[Bel57]   R. Bellman. *Dynamic Programming*. Princeton University Press, Princeton, NJ., 1957.

[BF95]   Avrim L. Blum and Merrick L. Furst. Fast planning through planning graph analysis. In Chris S. Mellish, editor, *Proceedings of the 14th International Joint Conference on Artificial Intelligence, IJCAI-95*, volume 2, pages 1636–1642, Montréal, Canada, August 1995. Morgan Kaufmann.

[BG01]   Blai Bonet and Hector Geffner. Planning as heuristic search. *Artificial Intelligence*, 129(1-2):5–33, 2001.

[BK98]   Ralph Bergmann and Alexander Kott, editors. *Proceedings of the AIPS-98 Workshop Integrating Planning, Scheduling and Execution in Dynamic and Uncertain Environments*, Pittsburgh, PA, June 1998. AAAI Technical Report WS-98-02.

[BL98]   Michel Beaudouin-Lafon. *Computer Supported Co-operative Work (CSCW)*. John Wiley and Sons Ltd, 1998.

[Bly94]   Jim Blythe. Planning with external events. In *Proceedings of the International Conference on Uncertainty in AI*, 1994.

[BM95]   M. Burnett and D. W. McIntyre. Guest editors' introduction: Visual programming. *IEEE Compute*, 28(3):14, March 1995.

[Bro86]   Rodney A. Brooks. A roboust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, RA-2(1):14–23, March 1986.

[BV97]   Daniel Borrajo and Manuela Veloso. Lazy incremental learning of control knowledge for efficiently obtaining quality plans. *AI Review Journal. Special Issue on Lazy Learning*, 11(1-5):371–405, February 1997. Also in the book "Lazy Learning", David Aha (ed.), Kluwer Academic Publishers, May 1997, ISBN 0-7923-4584-3.

[BVV01]   Daniel Borrajo, Sira Vegas, and Manuela Veloso. Quality-based learning for planning. In *Working notes of the IJCAI'01 Workshop on Planning with Resources*, pages 9–17, Seattle, WA (USA), August 2001. IJCAI Press.

[CBMA01]   David Camacho, Daniel Borrajo, José Manuel Molina, and Ricardo Aler. Flexible integration of planning and information gathering. In Amedeo Cesta, editor, *Sixth European Conference on Planning (ECP'01)*, Lecture Notes in Artificial Intelligence, Toledo (Spain), September 2001. Springer Verlag.

[Ces01]   Amedeo Cesta, editor. *Proceedings of the Sixth European Conference on Planning (ECP'01)*, Lecture Notes in Artificial Intelligence, Toledo (Spain), September 2001. Springer Verlag.

[CGK+02]   Hans Chalupsky, Yolanda Gil, Craig A. Knoblock, Kristina Lerman, Jean Oh, David V. Pynadath, Thomas A. Russ, and Milind Tambe. Electric elves: Agent technology for supporting human organizations. *AI Magazine*, 23(2):11–24, Summer 2002.

[CRK+00]   S. Chien, G. Rabideau, R. Knight, R. Sherwood, B. Engelhardt, D. Mutz, T. Estlin, B. Smith, F. Fisher, T. Barrett, G. Stebbins, and D. Tran. ASPEN - Automating space mission operations using automated planning and scheduling. In *SpaceOps 2000*, Toulouse, France, June 2000.

[dDOW99]   M. desJardins, E. Durfee, C. Ortiz, and M. Wolverton. A survey of research in distributed, continual planning. *AI Magazine*, 4:13–22, 1999.

[DFvH02]   J. Davis, D. Fensel, and F. van Harmelen. *Towards the Semantic Web: Ontology-Driven Knowledge Management*. John Wiley, 2002.

[DG95]   Amit Sheth Diimitrios Georgakopoulos, Mark Hornick. An overview of workflow management: From process modeling to workflow automation infrastructure. *Distributed and Parallel Databases*, 3:119–153, 1995.

[DHW94]   Denise Draper, Steve Hanks, and Daniel Weld. Probabilistic planning with information gathering and contingent execution. In *Proceedings of the Second International Conference on AI Planning Systems, AIPS-94*, pages 31–36, Chicago, IL, June 1994. AAAI Press, CA.

[DKR02]   Brian Drabble, Jana Koehler, and Ioannis Refanidis, editors. *Proceedings of the AIPS-02 Workshop on Planning and Scheduling with Multiple Criteria*, Toulouse (France), April 2002.

[DMAN01]   J. Dix, Héctor Muñoz-Avila, and Dana S. Nau. Impacting shop: Putting an AI planner into a multi-agent environment. *Annals of Mathematics and AI*, 2001.

[Dra99]   Brian Drabble. Task decomposition support to reactive scheduling. In Susanne Biundo and Maria Fox, editors, *Proceedings of the 5th European Conference on Planning (ECP-99)*, pages 200–212, Durham (UK), 1999.

[EM96]   Tara A. Estlin and Raymond J. Mooney. Multi-strategy learning of search control for partial-order planning. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, volume I, pages 843–848, Portland, Oregon, August 1996. AAAI Press/MIT Press.

[FG98]   M. Fox and M. Gruninger. Enterprise modelling. *AI Magazine*, pages 109–121, Fall 1998.

[Fir96]   James R. Firby. Modularity issues in reactive planning. In *Proceedings of the Third International Conference on AI Planning Systems*, pages 78–85, Edinburgh (UK), May 1996.

[FL02]   Maria Fox and Derek Long. *PDDL2.1: An Extension to PDDL for Expressing Temporal Planning Domains*. University of Durham, Durham (UK), February 2002.

[FN71]   Richard E. Fikes and Nils J. Nilsson. Strips: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2:189–208, 1971.

[GHS95]   Dimitrios Georgakopoulos, Mark F. Hornick, and Amit P. Sheth. An overview of workflow management: From process modeling to workflow automation infrastructure. *Distributed and Parallel Databases*, 3(2):119–153, 1995.

[GHT02]   Malik Ghallab, Joachim Hertzberg, and Paolo Traverso, editors. *Proceedings of the Sixth International Conference on Artificial Intelligence Planning Systems (AIPS-02)*, Toulouse (France), April 2002. AAAI Press.

[GL87]   Michael P. Georgeff and Amy L. Lansky. Reactive reasoning and planning. In *Proceedings of AAAI-87 Sixth National Conference on Artificial Intelligence*, pages 677–68, Seattle, WA (USA), July 1987.

[GL94]   M. Ghallab and H. Laruelle. Representation and control in IxTeT, a temporal planner. In *Proceedings of the 2nd International Conference on AI Planning Systems*, 1994.

[GT99]   Fausto Giunchiglia and Paolo Traverso. Planning as model checking. In *Proceedings of the Fifth European Conference on Planning (ECP'99)*, pages 1–20, 1999.

[Ham86]   Kristian J. Hammond. *Case-based Planning: An Integrated Theory of Planning, Learning and Memory*. PhD thesis, Yale University, 1986.

[HG00]   Patrick Haslum and Hector Geffner. Admissible heuristics for optimal planning. In *Proceedings of the Fifth International Conference on AI Planning Systems (AIPS-2000)*, pages 70–82, 2000.

[HH00]   Jeff Heflin and James A. Hendler. Dynamic ontologies on the web. In *Proceedings of the AAAI-2000*, pages 443–449, 2000.

[Iwa94]   Masahiko Iwamoto. A planner with quality goal and its speed-up learning for optimization problem. In K. Hammond, editor, *Proceedings of the Second International Conference on Artificial Intelligence Planning Systems (AIPS94)*, pages 281–286, Chicago, Illinois, June 1994.

[JEJ95]   Ivar Jacobson, Maria Ericsson, and Agnet Jacobson. *Object Advantage: Business Process Reengineering With Object Technology*. Addison-Wesley, 1995.

[JWMD93]   P.R. Jackson, T.D. Wall, R. Martin, and K.P. Davids. New measures of job control, cognitive demand and production responsibility. *Journal of Applied Psychology*, 78:753–762, 1993.

[Kae87]   Leslie P. Kaelbling. An architecture for intelligent reactive systems. In M. Georgeff and A. Lansky, editors, *Reasoning about plans and actions*, pages 395–410. Morgan Kaufmann, San Mateo, California, 1987.

[Kam89]   Subbarao Kambhampati. *Flexible Reuse and Modification in Hierarchical Planning: A Validation Structure Based Approach*. PhD thesis, Computer Vision Laboratory, Center for Automation Research, University of Maryland, College Park, MD, 1989.

[KHW95]   Nicholas Kushmerick, Steve Hanks, and Daniel S. Weld. An algorithm for probabilistic planning. *Artificial Intelligence*, 76(1-2):239–286, 1995.

[KLM96]   Leslie P. Kaelbling, Michael L. Littman, and Andrew W. Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, pages 237–285, 1996.

[KM93]   S. Kuhn and M. Muller. Participatory design - introduction. *Communications of the ACM*, 36(4):24–28, June 1993.

[KMRS97]   Kurt Konolige, Karen L. Myers, Enrique H. Ruspini, and Alessandro Saffiotti. The Saphira architecture: A design for autonomy. *Journal of Experimental & Theoretical Artificial Intelligence*, 9(1):215–235, 1997.

[Kol93]   Janet Kolodner. *Case-Based Reasoning*. Morgan Kaufmann, 1993.

[KS96]   Henry Kautz and Bart Selman. Pushing the envelope: Planning, propositional logic, and stochastic search. In *Proceedings of AAAI-96*, pages 1194–1201, 1996.

[KSB98]   David Kortenkamp, Debra Schreckenghost, and R. Peter Bonasso. Three NASA application domains for integrated planning, scheduling and execution. In Ralph Bergmann and Alexander Kott, editors, *Proceedings of the AIPS'98 Workshop on Integrating Planning, Scheduling and Execution in Dynamic and Uncertain Environments*. AAAI, AAAI Press, 1998.

[KSWM00]   Helge Kahler, Oliver Stiemerling, Volker Wulf, and Anders Mørch. Computer supported cooperative work. *Journal of Collaborative Computing*, 9(1), 2000.

[LG90]   D. B. Lenat and R. V. Guha. *Building large knowledge-based systems. Representation and inference in the Cyc project*. Addison-Wesley, Reading, MA (USA), 1990.

[MB99]   Karen Myers and Pauline Berry. Workflow management systems: An AI perspective. Technical report, Artificial Intelligence Center, SRI International, Menlo Park, CA (USA), 1999.

[McD00]   John McDermott. The AIPS'98 planning competition. *AI Magazine*, 21(2), 2000.

[MCL+99]   Thomas W. Malone, Kevin Crowston, Jintae Lee, Brian Pentland, Chrysanthos Dellarocas, George Wyner, John Quimby, Charley Osborne, Avi Bernstein, George Herman, Mark Klein, and Elissa O'Donnell. Tools for inventing organizations: Toward a handbook of organizational processes. *Management Science*, 45(3):425–443, 1999.

[MNPW98]   Nicola Muscettola, P. Pandurang Nayak, Barney Pell, and Brian C. Williams. Remote agent: To boldly go where no AI system has gone before. *Artificial Intelligence*, 103(1-2):5–47, 1998.

[Mow94]   Abbe Mowshowitz. Virtual organization: A vision of management in the information age. *The Information Society*, 10(4):267–288, 1994.

[Mow01]   Abbe Mowshowitz. Virtual organization: The new feudalism. *IEEE Computer*, 34(4):110–112, 2001.

[MS97]   Nicola Muscettola and David Smith. On-board planning for new millenium deep
            space one autonomy. In *Proceedings of IEEE Aerospace Conference*, Snowmass,
            CO, 1997.

[Mus94]   Nicola Muscettola. HSTS: Integrating planning and scheduling. In M. Zweben and
            M.S. Fox, editors, *Intelligent Scheduling*, pages 169–212. Morgan Kaufmann, 1994.

[Nar93]   Bonnie A. Nardi. *A Small Matter of Programming*. MIT Pres, Cambridge, MA,
            1993.

[Nar01]   Alexander Nareyek, editor. *Working notes of the IJCAI'01 Workshop on Planning
            with Resources*, Seattle, WA (USA), August 2001. IJCAI Press.

[Old96]   G. R. Oldham. Job design. In C. L. Cooper and I. T. Robertson, editors, *International
            Review of Industrial and Organisational Psychology*. John Wiley, 1996.

[PC94]   M. Alicia Pérez and Jaime G. Carbonell. Control knowledge to improve plan qual-
            ity. In *Proceedings of the Second International Conference on AI Planning Systems*,
            pages 323–328, Chicago, IL, 1994. AAAI Press, CA.

[Ped89]   Edwin Pednault. ADL: Exploring the middle ground between Strips and the situa-
            tion calculus. In *Proceedings of the Conference on Knowledge Representation and
            Reasoning*, pages 324–332, 1989.

[PS92]   Mark A. Peot and David E. Smith. Conditional nonlinear planning. In *Proceedings of
            the First International Conference on AI Planning Systems*, pages 189–197, College
            Park, Maryland (USA), 1992.

[PSS+99]   Massimo Paolucci, Onn Shehory, Katia P. Sycara, Dirk Kalp, and Anandeep Pannu.
            A planning component for RETSINA agents. In *Proceedings of the Agent Theories,
            Architectures, and Languages Conference*, pages 147–161, 1999.

[RK92]   David Ruby and Dennis Kibler. Learning episodes for optimization. In *Proceed-
            ings of the Ninth International Conference on Machine Learning*, pages 379–384,
            Aberdeen, Scotland, 1992. Morgan Kaufmann.

[RMBM01]   María Dolores Rodríguez-Moreno, Daniel Borrajo, and Daniel Meziat. Planning-
            based generation of process models. In Lee McCluskey and Alfredo Milani, ed-
            itors, *Proceedings of the ECP-01/PLANET Workshop on Automated Planning and
            Scheduling Technologies in New Methods of Electronic, Mobile and Collaborative
            Work*, pages 22–34, Toledo (Spain), September 2001.

[SMC+00]   Jussi Stader, Jonathan Moore, Paul Chung, I. McBriar, M. Ravinranathan, and Ann
            Macintosh. Applying intelligent workflow management in the chemicals industries.
            In L. Fisher, editor, *The Workflow Handbook 2001*, pages 161–181. Workflow Man-
            agement Coalition (WfMC), October 2000.

[SMZ+01]   R. M. Simpson, T. L. McCluskey, W. Zhao, R. S. Aylett, and C. Doniat. An integrated
            graphical tool to support knowledge engineering in ai planning. In Amedeo Cesta

and Daniel Borrajo, editors, *Preprints of the Sixth European Conference on Planning (ECP'01)*, Toledo (Spain), September 2001.

[Sta97]   Jussi Stader. An intelligent system for bid management. *The International Journal of Project & Business Risk Management*, 1/3:299–314, 1997.

[SWB93]   A.T. Schreiber, B.J. Wielinga, and J.A. Breuker. *KADS: A Principled Approach to Knowledge Engineering*. Academic Press, London (UK), 1993.

[TDL98]   Austin Tate, Jeff Dalton, and John Levine. Generation of multiple qualitatively different plan options. In *Artificial Intelligence Planning Systems*, pages 27–35, 1998.

[TLJD00]   Austin Tate, John Levine, Peter Jarvis, and Jeff Dalton. Using AI planning technology for army small unit operations. In *Proceedings of the 5th International Conference on Artificial Intelligence Planning Systems (AIPS-00)*, pages 379–386, 2000.

[vdAvDH+03]   W. M. P. van der Aalst, B. F. van Dongen, J. Herbst, L. Maruster, G. Schimm, and A. J. M. M. Weijters. Workflow mining: A survey of issues and approaches. *Data and Knowledge Engineering*, 47(2):237–267, November 2003.

[vdAvH02]   Wil van der Aalst and Kees van Hee. *Workflow Management - Models, Methods and Systems*. MIT Press, Cambridge, MA/London, England, 2002.

[Vel94]   Manuela Veloso. *Planning and Learning by Analogical Reasoning*. Springer Verlag, December 1994.

[VMC97]   Manuela M. Veloso, Alice M. Mulvehill, and Michael T. Cox. Rationale-supported mixed-initiative case-based planning. In *Proceedings of the AAAI/IAAI-97*, pages 1072–1077, Providence, RI, 1997.

[WH94]   Mike Williamson and Steve Hanks. Optimal planning with a goal-directed utility model. In K. Hammond, editor, *Proceedings of the Second International Conference on Artificial Intelligence Planning Systems (AIPS94)*, pages 176–181, Chicago, Illinois, June 1994.

[WMLW95]   D. E. Wilkins, K. L. Myers, J. D. Lowrance, and L. P. Wesley. Planning and reacting in uncertain and dynamic environments. *Journal of Experimental and Theoretical AI*, 7(1):197–227, 1995.

[Zub88]   Shoshana Zuboff. *In the Age of the Smart Machine: The Future of Work and Power*. Basic Books, New York, NY, 1988.

4. Planning and Scheduling for the Web

4.1. Introduction

This chapter contains the Roadmap of the Technical Coordination Unit on Planning and Scheduling for the Web, [TCU03b] it summarizes results from Web TCU initiatives and events as well as discussions and contributions from TCU members.

Defining a roadmap for Planning and Scheduling for the Web represents a particular challenge because the target industry for technology transfer, i.e. the web industry, is not long established, not well defined and it is moving and reshaping itself very fast. On the other hand many sub-areas in this field can benefit from Artificial Intelligence Planning and Scheduling (AI P&S) technologies, and, in some cases, technology transfer has already started. For these reasons this document will consider and explore a set of topics, guidelines and objectives, which will need to be updated and deeply revised in short term as soon as new challenges, hot topics, requirements and issues emerge from the web and IT industry.

4.1.1. The Planning for the Web TCU

The TCU on Planning and Scheduling for the Web has the goal of promoting communications, exchanges and technology transfer inside and outside the PLANET II network in the field of the applications of Planning and Scheduling technologies to the Web.

The Web TCU has been established in the PLANET II network, since it has been recognized that AI P&S can play a significant role in view of the emerging electronic markets and the new associated business processes, the AI P&S technologies can fruitfully contribute to manage the complexity and the flexibility issues posed by Web applications and more generally by new Its application.

A number of initiative and events has been organized by the Web TCU in order to promote initiatives in the field and to elaborate a Technology Transfer Roadmap, TCU events have been frequently co-located with major events in order to maximize participation of nodes and potential audience, the more relevant events have been:

- TCU Kickoff meeting , Toledo, Spain September 2001,

- Workshop on "Automated Planning and Scheduling Technologies in New Methods of Electronic, Mobile and Collaborative Work" co-located with the European Conference on Planning ECP01, Toledo, Spain, September 2001,

- Workshop on "Web-based Resources and Knowledge Interchange Format for AI Planning and Scheduling" co-located with the European Conference on Artificial Intelligence ECAI 2002, Lyon (France), July 2002

- Panel on Planning Technologies for the Web, International Conference on Electronic Commerce ICEC02, Hong Kong, October 2002

- Web TCU Meeting "Roadmap Workshop", Fourth PLANET General Network Meeting, Ulm, Germany, February 2003

- Roadmap Presentation at ICT&P04, International Conference on Information and Communication Technologies and Programming, Varna, Bulgaria, June 2003

At the end of the PLANET II program the Web TCU sums up 24 members among which two eastern European nodes and one associated node from USA. The first release of the Web TCU Roadmap has been peer reviewed from Prof. Subbarao Kambhampati, Univ. of Arizona at Phoenix, USA, which is an internationally acknowledged authority in the field.

4.1.2. AI P&S and the web

TCU area and objectives

The Information Society (IS) is announced by a set of interrelated emerging technologies where the web it is certainly one of the most apparent and popular elements. These technologies envision new relationships of the individuals between his/her own tasks and the new tools.

Individuals are forced to develop new methods of work in order to exploit the ITs at their best, new tools and application should reflect and model this new methods in order to be effective.

Despite of the success of the keyword "web" and the prefix "e-lectronic", and the easy prevision of increasing relevance of "e-commerce", "e-business" etc., it is important to focus on a wider vision of the potential role of AI P&S technologies in the Information Society not limited to the web applications. An exponential number of Internet based services, low cost mobile devices over GSM and UMTS networks, tools which integrate traditional and knowledge based systems, represent promising application fields where AI P&S can have a primary or a supporting role.

AI P&S can convey the flexibility typical of AI technologies in order to answer to expectations and requirements of new methods of work and for the exploitation of the new ITs.

The role that we envision for planning technology in the web, is not intended to realize the development of general standalone AI P&S systems, but also the realization of a number of customized AI P&S services or AI P&S components of complex system, moreover we expect that new problems and requirements arising from the web industry will provide research challenges to be met by the AI P&S scientific community in the next future.

In the next paragraphs we will motivate positively the promising perspectives for a technology transfer (TT) between AI P&S and the web. Sample scenarios will be depicted to clarify the potential applications and limits of current planning technology, as well as new AI P&S research challenge issues require to meet more advanced objective application.

The relevant results and events in the industrial and scientific community which represent the supporting ground for TT will be also pointed out, drawing the conclusion that TT, in this area, has already started at some extent. Perspective and goals of the actual implementation of TT will be finally discussed with respect to short, medium and long term.

Why does the web represent an effective target for the application of AI P&S technologies?

The main objective of this document is to show that AI P&S can give significant contributions to web technologies, on the other hand, there are also important motivations for the web as an effective test-bed for AI P&S technologies which can foster the visibility of AI P&S itself:

Digital content The web is machine readable and it provides a large quantity of machine readable data and software entities to interact with; automatic knowledge acquisition can be potentially realized in the web environment;

Virtual is real The web offers the chance having "real" planning application bypassing the complex robot machinery needed in traditional planning application fields such as aerospace or robotics, for example the actions "browse a list of available books" and "buy-a-book" can be easily executed in the digitalized world thus producing a real and useful effect;

Scalable complexity web applications can have different degree of complexity, simple applications can exists for simple existing planning models. As an example it is worth citing machine translation as a case of a technology which is not mature (i.e. for literature translation) but it could be worthwhile in niches applications (i.e. web pages automated translation)

Planning and scheduling technologies represent, on the other hand, a key factor in the framework of the service technologies for the global new IS especially with respect to the current scenario of web applications, they provide:

Knowledge based flexibility AI P&S convey the flexibility typical of knowledge based technologies in support of new modalities of work, production, commerce, entertainment, education etc.;

Models of change and interaction AI P&S provides models of change and interaction, this research area has developed systems which provide automated generation of plans, tasks monitoring, plans checking etc.

Another key factor which motivates the application of AI P&S technologies to the web is its dimension: **web is large**. In other words, a massive audience of personal end-users and business end-users is developing the need of customized versions of services, and consequently the need for web industry of tools and technologies to support them. This scenario is made more complex by the increasing diffusion of personal mobile devices which fosters the development of new modalities of work and interaction between the individuals and the business organizations, and it tends also to modify the traditional modality of B2B interaction.

Relevant issue, scenarios, potential applications

The relevant issues to be supported and managed by services and applications over the infrastructure provided by the Web and the new ITs are in general: *autonomy*, *adaptation*, *distribution*, *mobility*, *agent interaction*, *automatic collaborative support*, etc. These issues are typical topics of AI P&S research which have produced relevant results (for example in fields such as robot planning, robot interaction, spacecraft on board autonomy, software generation), for this reason their applications to the Web and new ITs are likely to represent the first examples of the technology transfer to this area:

- automatic maintenance of web applications, web info and web services [SS01, MSZ01]; (the AI P&S knowledge based representation of change, constraint checking and failure repairing techniques has been applied to support automatic maintenance in non-web domains [CFM⁺98]);

- user personalization/support/automation of services; (AI P&S techniques for goal directed synthesis)

- automation and autonomy for machine to machine services, (AI P&S, soft-bot and agents technologies [EO94])

In the following paragraph we will focus on the definition of three sample scenarios for AI P&S on the web, in order to analyze in more detail potentialities and drawbacks of the current AI P&S technologies effectiveness. The limits and temporal horizon of the proposed technology transfer will be also pointed out.

Scenario 1: The web as a transparent media/infrastructure   A first level of application of AI P&S to the web can be understood by looking into the web as a transparent communication media. At this level the usual components of a planning system, (i.e. user interface, planner, sensors, and actuators), can be distributed over the web. AI P&S systems use the web as a communication media, *they act and sense via the web/Internet*.

A sample scenario is the following:

> *a user (at location U) interacts through a web interface with a planner (at location P), in order to set up a experiment with a satellite radio-telescope (at location S), the execution of the resulting schedule will also involve antenna operations and radio-telemetry (at location A); the results are processed by a software system (at location W) and presented to the end-user.*

Under the hypothesis of the web as a *transparent media*, the evolution to be expected for the AI P&S technology is then similar to what we have seen in the migration of business application and databases from traditional systems to the web/intranet, i.e. at least in the first phases technical issues are mainly a matter of designing web interface and distributing system resources.

The web is a world wide transmission media which can be exploited by a planner or by a scheduler to realize a user/system distribution. Examples of application of this type also include *robot remote control*, *remote assembly*, and tools for *remote medical diagnosis*.

Typical open problems in the web as a transparent media framework are related with robustness, interfaces and with managing the asynchronous nature of the communications over the web.

Scenario 2: The web as an environment   A more interesting view is to consider the *web as a planning environment*. In this view web entities and web tasks are the object of AI P&S systems.

Web entities exist (a simple *web planning domain* is made of elements such as web pages, emails, files etc.) on which typical *actions* can be executed by users (e.g. pay, subscribe, supply, order, browse, look for, find, download, etc.) or provided by service systems (e.g. web page servers, search engines, mail servers, messenger servers, etc.). Moreover user and systems are pursuing *web tasks*, i.e. tasks which involve web entities (e.g. find a book, buy it, and download), and in general user transactions and user activities over the web (e.g. informative tasks, educational tasks, distance work tasks, etc.). Sometimes these tasks are not given automatic support, i.e. they rely on the user decisions, or, if automatic support is given, then it uses a too rigid and procedural approach, which does not satisfy the wide variety of user/business needs.

Planners can develop plans to act on the web virtual world in order to reach goals on web entities,

A key points is that web entities do not necessarily have a "real" counterpart, and they are not necessarily designed as a part of a single distributed system. Consider for example the following completely virtual plan for the *virtual goal* of promoting a web site: *buy disk space*, *transfer your old site to a popular web portal* in order to *obtain a better click rate*.

The synthesis of this plan would require the ability of describing a model of change (e.g. defining the meaning of acting on the web), goals (defining concepts such "a better visibility"), and a model of the actions to be taken to execute it (i.e. pay, subscribe, download, etc.), i.e. web entities should be represented as part as part of a planning domain.

In addition there are some traditional planning phases that needs to be reconsidered in the web domain:

- domain knowledge acquisition can be realized by activities of information gathering, information discovery and comparison (about existing portals, rates and prices), with respect to more traditional planning domain in which domain knowledge has not frequent variations;

- action execution in the web could imply to take into account of a dynamical scenario, in which not only actions can have unpredictable failure, but the domain can change during execution.

Scenario 3: The web as part of the environment   In this view the web domain is only a part of the user planning domain. User tasks and goals are in general related with real world activities and should interact and be coordinated with actions and plans which act in the web domain.

A simple example is that of a personal planner assistant, which suggests the user to buy a textbook online as *part* of the plan "successfully preparing an exam", which contains some other *not web steps* such as *going to lessons* and *doing exercises*.

Despite of the simple example this latter scenario is certainly more general and realistically applies to those business activities which take place only partially on (or through) the web.

When more production related activities become available on the web (for example: suppliers chains, customers, markets, delivery, payments, etc.) the manufactures planning activity will need to model the web as part of the production plans.

Although this vision of the web as part of the planning environment is certainly more general and it offer interesting research challenges, it is worth to analyze the basic elements which allow to apply AI P&S to the web domain, and which constraints are posed on AI P&S theoretical models aimed at supporting it.

It is apparent that the first scenario is of minimal interest for our scope, since web as a transparent media mainly poses problems related to the features of a mere communication channel.

In the framework of the second scenario, web as "the" environment more specific applications of AI P&S technologies can be proposed. A first group is represented by applications which use AI P&S as components of tools and systems related with web technologies, such as

Components for web servers and tools (e.g. applications to goal guided synthesis and maintenance of web sites/pages)

Components for supporting online services (e.g. scheduling load distribution to online assistance operators, activation of assistance/emergency chains, online configuration of products)

Components of soft-bots (e.g. AI P&S components of soft-bots for automatic auctioning, stock-bots or in general for modeling agents which acts on the web)

Components in systems for online supply chains management (issues: logistics, resources management, workflow management, tasks scheduling and assignment)

Moreover, somewhat more *web intensive* applications of AI P&S technologies can be proposed under the hypothesis that planning activity which takes place and operates on the web domain:

Planning online services/web services (e.g. realizing goal oriented dynamical integration, and execution monitoring of services already available on the web; automatic composition and integration of web services)

Supporting user web tasks (e.g. cooperative recognition, automation, monitoring and repairing of user web tasks, consider, for example, a planning based systems which support the user in task such as "organize a vacation" or "organize a meeting");

User adaptive services (e.g. automatic synthesis/monitoring of online courses and educational plans based on the personal skills/advancements; automatic synthesis of personalized newspaper based on user models);

Online AI P&S services general or specialized tools which plan or schedule on demand (e.g. for checking/rescheduling activities) and are available on the web

The third scenario web as *"part of the"* environment offers a furthermore complete vision of the contribution of AI P&S technologies, where planning and scheduling components can monitor and support the user activities throughout a process characterized by new ITs in which the web can have only a partial role, two application example can be:

- Mobile user assistants, which help and assist the user in his/her personal/work daily goals by adapting and reacting to the environment and with the devices present in the environment, i.e. mobile phone, web connections, sensors, etc. are acting as mediator among the user and his/her plans. The assistants continuously reconfigure the execution of the plan depending on the dynamically available resources.

- Global supply chain management, an AI P&S component is able.

The wide range of potential applications listed above support the vision underlying this roadmap that AI P&S technologies represent an important opportunity to improve flexibility and adaptability of web systems.

4.1.3. Requirements

The main requirements for a Technology Transfer Roadmap in the area of web are the ability of rapidly identifying the area in which is worth to use AI P&S technologies as soon as these area dynamically emerge from the web scenario; and the ability of promoting among the industry the use of the technology, that is a matter of building a technological culture among analysts, designers and developers where AI P&S should be perceived as an area to take into account of.

In order to realize an effective roadmap start-up it is primarily required the

- identification of existing AI P&S models and components which can already support specific web applications, i.e. models for plan synthesis, plan checking, scheduling, etc. Also small degree of automation and reasoning provided by AI P&S components can be of a great added value in a web framework where services are massive services.

In the immediately following phase it is required to pursue in parallel the developments of advanced planning models and web standards:

- development and focusing of the AI P&S research on models which are able to adequately represents the web domains, examples of the features required in AI P&S models for the web are the ability of managing incomplete information and inconsistencies, dynamical acquisition of domain models, events, and more generally, richer knowledge representations;

- development of planning standards for the web, promoting the integration of planning standard with other widely accepted standards (e.g. workflow management, semantic web, etc.), developing inter-standard tools (e.g. compilers among PPDL/workflow, etc.). It is apparent that developing according to a technological standard and providing interoperability tools, contributes to foster the industrial acceptance of the technology.

A final general goal is to

- establish the AI P&S technology in end user technological culture, i.e. many application problems which have been the object of planning research are often "re-discovered" or "re-invented" by application developers simply because they ignore that the problem can be and have been formalized in a planning framework (this is the case for example of automation of web service composition).

A Technology Transfer Roadmap, from one point of view represent a stimulus from the research (i.e. the discovery of new theoretical models which can manage the new issues coming from the target area), nevertheless it should not be forgot that the target of TT is the world of industry, of real, innovative, marketable applications. In particular a quality objective for a roadmap is to identify what can be done in a short term and what will be realistically possible to do in a medium/long term with the support of applied research, avoiding too ambitious goals.

The first objectives that are expected to be reached in a short term are mainly in the area of

- support/automation of personalized services on the web (educational, e-learning, recreation, information, etc.);

- supporting mobility and distribution in organization: scheduling activities and information workflow in distributed organization through the web and new devices/communication media, supply chain management;

- automatic integration of Services/Web Services, goals directed synthesis and maintenance in some specific domain.

On a medium/long term horizon, we expect increasingly to met higher requirements on tools and techniques for automation of Web Services, and increasing requests for personal support services, either on the web or on-board of personal assistant devices, such as:

- robust automatic integration of Web Services, in broader application domains, support for automatic discovery of services and failure recovery;

- autonomous web agents, goals directed software agents which operate over the web on user's behalf;

- support to massive adaptive services on small personal devices, planning and scheduling support for daily activities;

- co-operative and distributed planning which integrates personal devices/web services/web agents for global user goals.

4.2. Roadmap themes

The only roadmap theme which has been identified in this section regards the general TCU topic of applying AI P&S to the web domain.

4.2.1. The web domain

Introduction

A number of applications of AI P&S, which could identify possible roadmap themes, already exist (in the area of adaptive web sites, information retrieval and query optimization, web services composition, personal assistants and in other areas) and these applications represent the first steps of a

breeding among web and planning and scheduling technologies. Nevertheless the area of AI P&S applied to the web is still too young, and the number of results are not enough to characterized a theme as emerging over the others, finally the emerging theme of web issues are not yet completely identified from the initial focus on information retrieval to the current one on .

For these reasons we prefer to focus on open issues and perspectives giving a global picture of interrelations among AI P&S current and potential themes for a technology transfer roadmap.

Current state of the art and research trends

The scientific and technological ground which can allow AI P&S to be applied to web domains (and more generally to play a role in new ITs) has been made possible by a series of advancements in the area of planning models, systems and applications, knowledge based systems, and in the field of web technical assessment and web standardization.

In the following paragraphs a brief chronological path is given along these lines, there will be indexed works or applications considered relevant to the specific objective of AI P&S for the web.

Scientific works and applications in the area of automated planning start to pay attention to web related issues with [Kno95], which introduces the idea of using a planner in order to gather information; although proposing a new way to look at planning, it reflects the vision of web applications, as a matter of information retrieval, where planning can mainly support flexibility and reconfiguration of info gathering steps. The approach to the web from a planning research point of view remains basically the same in the following years, nevertheless interesting contributions are produced to issues like *managing incomplete information* [GK96], *sensing* [NA97], *managing contingencies* [FM97] for info gathering.

In the following years the increasing diffusion of interactive services available on the web has suggested the possibility of Travel Assistant [Amb01], i.e. AI P&S based dynamic integration of web sources for travel organization (managing and combining multiple sources for air fares, parking fares, timetables, weather conditions, etc.); relevant contributions of this application work were the use of planning technique for user support [RLRG02], and the use of wrappers and mediators techniques [SA99] in order to abstract equivalent web sources from their presentation interface details.

A further step has been possible with the introduction of the technology of web services, in this vision not only web source are available but also web services, i.e. web operators which can produce state modifications, provided that input conditions are met. Various works [MS02b, MS02a, TKAS02] propose to use a planning model to give a semantics to the behavior of web services, and to use planners in order to generate compositions of them (i.e. plans) for reaching complex goals on the web.

On the field of the web infrastructure a number of elements has positively contributed to built the common ground for AI P&S mentioned in the introduction:

- an increasingly wide diffusion of XML standards,

- the introduction of web services as a uniform model for system to system interaction over the web (WSDL), and

- availability of models for service discovery (UDDI).

Another important element has been the proposal of Semantic Web [BLT01], which has the ambitious goals of allow reasoning on a web of knowledge and meanings, with respect to the initial web made of presentation tags (i.e. HTML) and a web of syntactic structures and terms (i.e. XML). The data on the semantic web are defined and linked in a way that digital resources can be used by machines not just for display purposes, but for automation, integration and reuse of data across various applications. Semantic Web is of great important in the future of knowledge bases applications for the web, because of its success among the research community, the growing number of available tools and application and the standardization factor, since it is supported and coordinated by W3C Consortium. Despite of the initial focus on ontologies and relationships, research started under the DARPA Agent Markup Language (the DAML program [M.00] has the goal of developing languages and tools to facilitate the concept of the Semantic Web) has moved to model more dynamical aspect of the web. For example, the proposal of DAML-S is oriented to model "semantic web services", i.e. to develop DAML-based Web Service Ontologies in order to facilitate the automation of Web service tasks including automated Web service discovery, execution, interoperation. In this scenario it is very important to point out the work of McDermott and his group which developed bi-directional DAML-PDDL translator [PDD, MDD] focusing on ontologies translation, and the work of McIlraith and her group [MS02b] which developed a tool for translating from DAML-S to PDDL.

Finally we mention, the great interest of planning community for the area workflow management which led to establish a special purpose Technical Coordination Unit on Workflow Management inside the PLANET II network [TCU03c]. In this area a relevant initiative is the Workflow Markup Language (WFML), an XML derived standard language defined by the Workflow Management Coalition. WFML basically provides representation for tasks and data flow in complex activities, although is more oriented to description purposes than to synthesis of task. WFML shares with AI P&S the domain of modeling tasks and activities, moreover it includes some important points: it is a W3C standard, and it is supported by major world-wide software companies.

In order to give a complete picture of the state-of-the-art, it is worth mentioning a number of events, which have taken place in the last three years among the scientific and industrial community.

These events testify the growth of a favorable environment and an increasing interest for the first phases of technology transfer between AI P&S and web, which is not limited to single sporadic works or pioneer research.

- PLANET/ECP Workshop on "Automated AI P&S in New Methods of Electronic, Mobile, and Collaborative Work", Toledo, Spain, September 2001

  This workshop took place as a side event of the European Conference on Planning (now ICAPS) and it was co-organized by the PLANET II TCU on P&S for the Web and the TCU on Knowledge Engineering. This event probably represented the first relevant international meeting which directly focused on the application of AI P&S to new ITs, and namely to the web [Amb01, B.01].

- PLANET/ECAI Workshop on "Web-based Resources and Knowledge Interchange Formats for AI Planning and Scheduling", Lyon, France, July 2002

This workshop was co-located with the European Conference on Artificial Intelligence and it has been organized by the three PLANET II TCUs on Knowledge Engineering, P&S for the Web, Workflow Management. The focus of the workshop was specifically aimed at investigating the possibility of developing a Knowledge Interchange Format for AI P&S, by comparing the experience and evaluating the possible contributions from W3C standards such as WFML, the Semantic Web proposal and the planning de-facto standard PDDL.

- AAAI Tutorial: Information Integration on the Web, July 2002

  This tutorial was organized by the American Association for Artificial Intelligence, on the issue of integrating information sources on the web.

- PLANET II 2nd International Summer School on AI Planning, Tutorial on "Planning on the Web", Halkidiki, Greece, September 2002

  The planning summer school organized and sponsored by PLANET II has included a specific tutorial about planning on the web.

- PLANSERVE Enabling Technologies for Intelligent Planning Services, Expression of Interest VI Framework Programme 2002-2003

  PLANSERVE is an integrated project proposal within the Sixth Framework Programme which aims at make available a number of intelligent planning and scheduling services under the framework of a web based problem solving grid,

- ICAPS Workshop on Planning for Web Services, Trento, Italy, June 2003

- ICAPS Workshop on PDDL, Trento, Italy, June 2003

- PLANET II 3rd International Summer School on AI Planning, Tutorial on "Planning on the Web" by Dan Weld, Madonna di Campiglio, Italy, June 2003

These last two events are co-organized by the International Conference on AI Planning and Scheduling and confirms the growing interest of the community in the topic of AI P&S for the web, the Workshop on PDDL opens the call for paper to submission of proposals of extensions of PDDL specifically oriented to web applications.

Open issues and AI P&S perspectives

In order to give a complete picture of the open issues involved in the application of AI P&S to the Web and new ITs we will discuss possible mapping between typical planning concepts (such as initial states, operators, goals, plans, etc.) onto the corresponding web entities. The purpose of the discussion is to clarify the perspective applications of AI P&S technologies, and, at the same time, to point out open issues and limits of the current planning models when operating on the web domain.

Mapping state, initial state, goal state   The notion of state is central to most planning models. The current state of the world is usually represented in planning by a set of facts and domain laws assumed to be true and characterizing the state of the world at a certain time.

In the web the concept of state is represented by the sum of the states of the various component of the web domain:

User internal state The set of features and facts describing the user state (e.g. maintenance conditions, general constraints, preferences, but also info about users such as identities and passwords);

Web-state facts The state of info as available on the web, in the simplest form web-state is represented by the set of current web pages, existing files etc., more likely the state description would address the content and the semantic of available information (e.g. stock quotations, available item on e-markets, etc.);

Web actors state This is represented by the internal state of interactive services, consider, for example, activities which require multiple steps in order be carried out, such as online reservation of a flight (it is necessary to be able to represent the current step of a given transaction), or consider services which require access authorization(it is necessary to represent that certain existing resources are/are not available for use).

The concept of state, that is used in planning to model initial states, goals state, and to model change as modification of state, will need reformulation in the framework of web applications, because of the inadequacy of current planning model to fully describe the state of a web domain in all its aspects.

Open issues   The main drawback of the current planning models of states derive from the fact that the web is a *vast and dynamical environment of active resources*, on the other hand planning domains are usually characterized by domain states which are fully knowledgeable and somewhat static, i.e. the planner is the only one (or one of the few) agent in the domain.

Summarizing, the main AI P&S research issues that should be addressed in order to fully capture the main features of a *web state* are:

Managing incompleteness Web states are vast and not completely knowledgeable, the web domain is inherently incomplete. There is no hope to model inside the planner a *complete* description of all the web, so mechanisms and strategies are required in order to do knowledge acquisition and to *circumscribe* the domain description for the planning problem at hand.

Managing events Web states change independently of the planning actor. This is usually addressed in planning models by the concept of *events*, i.e. state changes which are out of the control of the planner [M.98]. In the web domain the amount of *planning time* appear to be a crucial factor, since dynamical changes can occur during planning.

Managing inconsistency Web states can contain contradictory information coming from different sources. Consider for example info about the weather conditions used by a travel planner – concepts such as trust and believes and should be modeled [AK98].

Managing richer knowledge representations AI P&S models usually focus more on actions and plans than on the adequacy of the knowledge representation formalism to capture the *static* aspect of the domain. Since the web offers information in different but equivalent forms, the planners should be aware of ontologies and mechanisms for relating information from different sources. Consider for example the problem of representing the concept of *price*, in the sense of amount of money to be payed for buying a good, an effective representation requires that *stock exchange market quotations*, *monthly renting rates*, *price of a book* and *currency exchange rates* would represent instances of the same concept of price [BLT01].

Mapping actions and operators   The other fundamental concept in planning models is the notion of *action* and *operators*, which represents the basic elements to model change in the domain.

Actions (i.e. instances of operators) represent the state transition which occurs during the execution. Operator and action are usually modeled in term of precondition/effects, that is by describing in the preconditions the partial state and constraints in which the action can be executed, and describing the effects which are produced in the input state.

In a web domain actions are represented by the available services, which change the user/web state according to a preconditions/effects model. For example available services on the web such as *buying a book*, *reserving a meeting room*, *downloading a satellite image of a town*, *moving a remote web cam* , *sending an email*, are actions in the sense that are allowed to take place only under certain preconditions on the web state (e.g. *having enough money on the account*, *availability of the town satellite picture*, etc.) and produce effects on it (e.g. *changing book ownership*, *meeting room reservation state*, *a copy of the picture locally available*, etc.).

Open issues   It is worth noticing that an effective model of *web actions* should take into account the elements formerly pointed out about web states (incompleteness, inconsistency in particular), but it requires, in addition, to consider some *web specific* AI P&S research issues about the actions/operators model:

Operators can change, appear and disappear New services become available and change over time, while old service disappear. It is not possible to assume that this is under the control of the planner/executors, but it is likely to expect the planner to be able of discovery, monitoring, and maintaining its domain model.

Actions execution time Most planning models assume atomic instantaneous execution of actions and take not much into account of execution time, often the planner delegates to the execution monitoring phase the problem of actions failure. On the other hand, actions on the web take time (depending on bandwidth and servers overhead factors) and are likely to fail. These factors should promote research on those planning models which include timing constraints and failure recovery since plan generation phase.

Mapping plans   "A *plan* is a set of ordered actions, that, if executed in the initial state will transform it in the goal state." This definition of solution plan is common to most planning models since [FN71], and it is easy to see that the definition easily applies also to the web domain. Currently

most web tasks and goal oriented activities which take place on the web can be described basically as sequences of actions, i.e. on the web, plans are *sequences of interrelated services requests made on web entities*, (consider for example a typical user driven web-plan such as: *look for items sellers on search engine*, *browse and compare prices*, *order one and pay for it*).

Open issues　Although plan as sequences or partially ordered sequences are a formalism that is sufficient for planning in the web at a basic level, it is worth to investigate on planning models which:

Provide expressive and flexible models of plans　Few planning models have a satisfactory management of actions with duration, loops, and conditionals [LSH95].

Combine plan knowledge with task oriented languages　Planning models based on a hierarchical approach [KE94], as well as of workflow management models [TCU03c] offer examples of task oriented formalisms which should be included into generative planning models.

The objective should be to reflect also in the structure of plans, some typical element of the web domain, we have already pointed out, such as contingency, non-determinism, and dynamical aspects of the domain.

Mapping plan monitoring, execution, sensing, re-planning　Planning terms such as *plan monitoring*, *execution*, *sensing*, and *re-planning* regard the phase that follow plan synthesis and aimed at actually reaching the goals by executing the plan in the real world.

In conventional planning models, the execution of the solution plan is monitored during this phase, *actual actions* to be executed have to be chosen (for example if the plan gives only a partial order), the appropriate *actuators* are triggered and *sensors* are activated (note: actuators and sensors are typical terms from plan execution in robotics domains) in order to monitor the real world on the expected effects of actions, if the information coming from sensors detects action failure then a re-planning mechanism is usually invoked.

Once again the web domain (where *virtual* and *real* are somewhat overlapping) can characterized in a "web specific" way traditional planning concepts like execution and sensing:

Execution　Since the web is *large*, execution can involve complex decisions about choosing the service to invoke among a plurality of available and equivalent ones. Moreover, *execution takes time* also for processing and data transfer, so criteria are needed to combine the issue of execution time with the issue of bandwidth. Finally, note as the implementation of *actuators* would be greatly favored by the diffusion of web services, but the use of *wrappers* or similar mechanism should also be considered as an intermediate solution for interacting with services initially designed for human users.

Sensing　In the web environment sensing mainly means: *actively* looking for info, i.e. gathering info [Kno95, K.98, NA97] and results from web sources (for example *web pages*, *searching databases*, *streaming video*, *results from called services*). The main open problems in this area are related with the integration of information from different non-homogeneous sources, and

with decisions to be taken about *what to sense* (consider for example criteria like: *time for sensing and processing vs. bandwidth vs. timing goals and execution constraints*).

It is also worth noticing that the dynamical nature of the web suggests the investigation of theoretical models based continuous planning and planning during execution [HK96, FM97], where the plan synthesis phase is interleaved with execution and sensing.

Research goals

The AI P&S research community had increasing interest on topics such as modeling dynamic domains, time management, interleaving planning/execution and others topics which can be exploited in modeling the web environment.

Nevertheless the many open research issues raised in the previous paragraphs can be summarized in the research goals which are prerequisites for supporting the initial (short term) and next (medium-long term) steps of the technology transfer roadmap.

Short term research goals   The research goals which should be investigated in a short term include

- modeling domain discovery during planning,

- contingency and sensing in web environment,

- time duration, failure recognition, and repair,

- expressive and robust model of execution plans, (loop iterations),

- incorporating web oriented extension in PDDL-like language (see [TCU03a] and [TCU03c]), and

- wrappers and planning operator wrappers.

Medium term research goals   In a second phase advancements in the previous topics are needed, as well as an increasing the research efforts toward more technology oriented topics such as *portability*, *interoperability*, *scalability*, and advanced planning models, these objectives include:

- mixed initiative planning models, task support models,

- portability for planning algorithms, strategies, heuristics, preprocessing,

- interoperability: mapping between planning models,

- models and measures of planning scalability (small devices/ small domains),

- models for cooperative distributed planning, grids of planning and scheduling services, reconfiguration and extension of the grid, and

- domains/problem partitioning/integration/assignment problem-solver.

Technology transfer goals

The path proposed for the Technology Transfer Roadmap has the main purpose of full-filling the end-user expectations and establishing the planning technology among developers web and IT industries.

The Technology Transfer Roadmap is characterized by three strategic general objectives in the short, medium, long term. The objectives are strategic in the sense that are seen as milestone for the roadmap.

The initial, short term objectives will be:

- realizing applications which meets the end-user requirements and demonstrate the technology.

Furthermore, medium term technology transfer objectives will be

- consolidation of the results in widely accepted technology standards, development of standards and development of tools.

And, finally,

- including the analysis of planning and scheduling components as standard steps shortening the time required to incorporate and transform into technology the research advancements.

The following is a list of indicators in approximately increasing time order, which will represent positive indicators for the advancement in the technology transfer path:

- specialized/customized applications which test and demonstrate the technology as *part of* tools or services delivered through the web (e.g. automatic goal directed generation/maintenance of web sites; personalized educational plans; planning modules in web software agents; online supply chain management; distributed activities scheduling and workflow management; etc.);

- tools for planning operators wrapper definition and maintenance;

- tools for online services/web services integration;

- online planning and scheduling services for remote invocation (see PLANSERVE initiative);

- standard language for planning on the web or on the semantic web (expected some XPDDL, i.e. PPDL-XML extensions and/or integration with DAML-S, and/or integration with WFML and/or scheduling/time issues);

- accepted standardization portability of algorithms, strategies, heuristics, preprocessing, domain dimensions (let, for instance, XPDDL+);

- advanced systems for automatic discovery of web operators/wrappers;

- planning components on small devices for delivering support services on small/specialised/local domains (java tools, planning and scheduling web services);

- systems for robust and dynamic planning with web entities.

Recommended actions

It should be noted once again that a roadmap is "a living document that should be continuously updated", this is particularly true for the AI P&S for the Web Roadmap.

Since the target transfer area is the web, unpredictable developments in the web technology, or in general in IT technology, could produce acceleration or change of perspectives in a very short time horizon, and the development of the field could take unknown directions.

Nevertheless the transfer of AI P&S results seems to be an effective means for improving the quality of the systems/services offered by the web and ITs.

As noted in the previous sections there is a favorable ground for technology transfer to take place, and, in some cases, it already started, at least at the level of prototype applications. This autonomous transfer could benefit and it should be accompanied by measures such as:

- direct actions to promote the visibility of applications which demonstrate the effectiveness of AI P&S for the web, (e.g. promotion of spin-off enterprises on the matter);

- specific targeted projects for developments of tools;

- educational initiative (targeted to postdoctoral level, industrial and applied research departments of private companies and public institutions);

- actions for promoting standardization efforts, both in the form of initiatives internal to the planning community, or by joint initiatives (see DAML [M.00], or WFML [TCU03c]) which could multiply the community of developers and the visibility of results (esp. W3C joint initiative);

- promoting projects under the 6th Framework Programme Initiative (see PLANSERVE initiative).

4.3.   Summary and conclusions

The interrelated emerging technologies, which announce and characterized the Information Society, force individuals to develop new methods of work in order to exploit the ITs at their best, new tools and application should reflect and model this new methods in order to be effective.

The vision of the Web as large, evolutionary, massive environment where thousands of activities and users are pursuing their own interrelated-goals seems to find a natural framework into models and results from planning and scheduling research.

AI P&S can convey to the Web, and in a more wider vision to ITs, the flexibility typical of knowledge based technologies and provides models of goal directed change and interaction, which can be exploited in application frameworks such as the web domain.

AI P&S is crucial to govern dynamical aspect of the web such as personalized services, integration and reconfiguration of resources, and adaptation to changes.

The currently growing number of AI P&S applications to the web (e.g. in the area of web resources planning, web service composition, info gathering, etc.) confirms the great potentiality of the field

and show that the technology is already mature. On the other hand a great cultural and informative barrier exists since AI P&S is not yet perceived as a standard area: the industries of the sector are often rediscovering or redeveloping problems and solutions (e.g. in the area of workflow management, web service integration, etc.) already formalized and investigated from the AI P&S research community.

The main milestones of the Roadmap have been identified and they can be summarized by the following actions:

- promoting the identification and use of AI P&S components inside web applications, promoting visibility of successful applications;

- focusing on theoretical issues of AI P&S for the web in order to support further advanced applications (e.g. managing topics as *incompleteness*, *inconsistencies*, and *dynamical domains*);

- developing web based planning standards or extensions to existing standards to foster the technology;

- supporting educational initiative which allow to build a culture of planning and scheduling technology as accepted components in project design and system development;

- promoting applied research projects in the area (e.g. EU projects).

The proposed technology transfer represents a peculiar challenge because the target industry, i.e. the web industry, and more generally the ITs, is not long established, not well defined and it is moving and reshaping itself very fast. In this area "future emerging trends" are often unpredictable and expressions like "long term scenario", should likely refer to a few years term, or sometimes are likely to mean "months". A technology transfer roadmap in such a dynamical area can be successful only if a timed reaction is provided to the emerging themes, which could likely lead to a deep revision of the Roadmap itself. Nevertheless we believe that the proposed actions could foster and drive a technology transfer, which, at some extent, has already started.

Bibliography

[AK98]   Jose Luis Ambite and Craig A. Knoblock. Flexible and scalable query planning in distributed and heterogeneous environments. In R. Simmons, M. Veloso, and S. Smith, editors, *Proceedings of the 4th International Conference on Artificial Intelligence Planning Systems (AIPS-98)*, pages 3–10. AAAI Press, Menlo Park, California, USA, 1998.

[Amb01]   J.L. Ambite. Heracles: Building Planning and Information Assistants. In *Proc. of the ECP/PLANET Workshop on Automated P&S in New Methods of Electronic, Mobile, and Collaborative Work*, September 2001.

[B.01]   Laithwaite B. BritishTelecom Workforce Management System. In *Proc. of the ECP/PLANET Workshop on Automated P&S in New Methods of Electronic, Mobile, and Collaborative Work*, September 2001.

[BLT01]   Lassila O. Berners-Lee T., Hendler J. The Semantic Web. *Scientific American*, May 2001.

[CFM+98]   S. Chien, F. Fisher, H. Mortensen, E. Lo, R. Greeley, A. Govindjee, T. Estlin, and X. Wang. Using Artificial Intelligence Planning Techniques to Automatically Reconfigure Software Modules. In Angel P. Del Pobil, José Mira, and Moonis Ali, editors, *Tasks and Methods in Applied Artificial Intelligence, 11th International Conference on Industrial and Engineering Applications of Artificial In telligence and Expert Systems, IEA/AIE-98 (Vol. 2)*, volume 1416 of *Lecture Notes in Computer Science*, pages 415–426, Castellón, Spain, June 1998. Springer Verlag. ISBN 3-540-64574-8.

[EO94]   Weld D. Etzioni O. A Softbot-Based Interface to the Internet. *Communications of the ACM*, 1994.

[FM97]   Weld D.S. Friedman M. Efficiently Executing Information Gathering Plans. In *Proc. of the Fifteenth International Joint Conference on Artificial Intelligence (IJCAI-97)*, August 1997.

[FN71]   R. Fikes and N. Nillson. STRIPS: A new approach to the Application of Theorem Proving to Problem Solving. *Artificial Intelligence*, 2:189–208, 1971.

[GK96]   Weld D.S. Golden K., Etzioni O. Planning with Execution and Incomplete Information. Technical Report UW-CSE-96-01-09, University of Washington, 1996.

[HK96]   Veloso M. Haigh K.Z. Interleaving Planning and Robot Execution for Asynchronous User Requests. In *Proc. of the AAAI Spring Symposium on Planning with Incomplete Information for Robot Problems*, March 1996.

[K.98]   Golden K. Leap Before You Look: Information Gathering in the PUCCINI Planner. In *Proc. of the Fourth International Conference on Artificial Intelligence Planning Systems (AIPS-98)*, pages 70–77. AAAI Press, 1998.

[KE94]   J. Hendler K. Erol, D. Nau. HTN Planning: Complexity and Expressivity. In *Proc. of the Twelfth National Conference on Artificial Intelligence (AAAI-94)*, July 1994.

[Kno95]   G.A. Knoblock. Planning Executing, Sensing and Replanning for Information Gathering. In *Proc. of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95)*, August 1995.

[LSH95]   Dean T. Lin S.-H. Generating Optimal Policies for High-level Plans with Conditional Branches and Loops. In *Proc. of the Third European Workshop on Planning*, pages 205–218, 1995.

[M.98]   Ghallab M. Chronicles as a practical representation for dealing with time, events and actions. In *Proc. of the Sixth Italian Conference on Artificial Intelligence (AIIA'98)*, Lecture Notes in Artificial Intelligence, pages 6–10. Springer Verlag, September 1998.

[M.00]   Burke M. The Darpha Agent Markup Language Project DAML. `http://www.daml.org`, 2000.

[MDD]   Qi P. Mc Dermott D., Dou D. PDDAML:An Automatic Translator Between PDDL and DAML. `http://www.cs.yale.edu/homes/dvm/daml/pddl_daml_translator1.html`.

[MS02a]   Milani A. Marcugini S. Automated Planning of Web Services. In *Proc. of the Third International Conference on Electronic Commerce, (ICEC-2002)*, October 2002.

[MS02b]   Fadel R. McIlraith S. Planning with Complex Actions. In *Proceedings of AIPS 2002 Workshop on Exploring Real World Planning*, April 2002.

[MSZ01]   S. McIlraith, T.C. Son, and H. Zeng. Semantic Web Services. *IEEE Intelligent Systems*, Special Issue on the Semantic Web:46–53, March/April 2001.

[NA97]   Levy A.Y. Naveen A., Knoblock G. A. Information Gathering Plans with Sensing Actions. In *Proc. of the Fourth European Conference on Planning (ECP-97)*. Springer Verlag, 1997.

[PDD]   PDDL 2.1: Planning Domain Description Language. `http://www.dur.ac.uk/d.p.long/IPC/pddl.html`.

[RLRG02]   C. Rich, N. Lesh, J. Rickel, and A. Garland. A Plug-in Architecture for Generating Collaborative Agent Response. Technical report TR-2002-10, Mitsubishi Electronic Laboratory, 2002.

[SA99]   Azavant.F. Sahuguet A. WysiWyg Web Wrapper Factory. In *Proc. of the Eight International World Wide Web Conference (WWW8)*, May 1999.

[SS01]   Crim D. Smith S.F., Hildum D. Toward the Design of Web-based Planning and Scheduling Services. In *Proc. of the ECP/PLANET Workshop on Automated P&S in New Methods of Electronic, Mobile and Collaborative Work*, September 2001.

[TCU03a]   Knowledge Engineering for Planning Roadmap. `http://www.planet-noe.org/TCUs`, 2003.

[TCU03b] Website of the PLANET-II Technical Coordination Unit on Planning and Scheduling for the Web. `http://www.planet-noe.org/TCUs`, 2003.

[TCU03c] Planet Workflow Management R&D Roadmap, PLANET-II Technical Coordination Unit. `http://www.planet-noe.org/TCUs`, 2003.

[TKAS02] S. Thakkar, G.A. Knoblock, J.L. Ambite, and C. Shahabi. Dynamically Composing Web Services from On-line Source. In *Workshop on Intelligent Service Integration, The Eighteenth National Conference on Artificial Intelligence (AAAI-02)*, 2002.

5. Robot Planning

> We used to know what *(robot)* planning was. It was the automatic generation of an action sequence to bring about a desired state of affairs. ...Nowadays nobody works on this problem any more. As stated, the problem turned out to be too hard and too easy. It was too hard because it was intractable. It was too easy because action sequences are not adequate as a representation of a real robot s program.
>
> [McD92]

5.1. Introduction

The next generations of autonomous robots will be characterized by being more general than the current one – in at least three respects. First, the robots will be able to successfully carry out multiple, diverse, and possibly interfering tasks in changing and partly unknown environments. Second, they will be able to improve their performance by autonomously adapting their control software for the kinds of tasks they are given and the environments they are to operate in. Third, they will be able to perform novel tasks without learning the achievement of these tasks in long and tedious learning sessions.

As a research community, we believe that these aspects of generality cannot be achieved without the robots being capable of planning their course of action based on foresight and without them being able to autonomously learn better control routines. To achieve more generality in these respects, parts of the control programs, called the robot plans, are to be represented explicitly, such that the robot can reason about them and revise them. This approach to autonomous robot control is called robot planning or –more generally– plan-based control. The objective is to use plans and the respective reasoning mechanisms for them as resources for increasing the generality and the performance of robot control programs.

In this Research Roadmap we show and analyze the potential for impact of robot planning on autonomous robot and agent control. Our goal is to show appropriate ways of how this potential can be realized, and lay down what the research community, application developers, and funding agencies can do to stimulate and accelerate scientific and technological breakthroughs. We propose a set of challenge application scenarios and a technological milestones for near-term, middle-term, and long-term research and technology development projects that must be achieved to control au-

tonomous robots over extended periods of time, in natural and unmodified human environments, and to perform complex jobs with them.

We begin the Roadmap by defining the domain of robot planning and its relation to the related fields of control theory, autonomous agents, and artificial intelligence. We then propose a framework of computational mechanisms that are necessary and make successful plan-based robot control possible. Thereafter, we illustrate the state of the art by sketching several autonomous robots that employ plan-based control mechanisms and briefly review current technological developments of the respective computational mechanisms. Subsequently, we examine a variety of challenge demonstration scenarios that illustrate the potential technologies that can be realized through the technological advances that we project. We then discuss the several fundamental research problems that define a research agenda that is intended to realize the full potential of planning-based autonomous robot control.

This Research Roadmap has been made possible by the European Network of Excellence in AI Planning (PLANET). The network has provided both funding and a public forum for the development of these idea. This document has been composed from contributions from many authors. The Roadmap would also not have been possible without the the Dagstuhl curatorium giving us the opportunity to organize two Dagstuhl seminars on plan-based control of robotic agents. In particular, the roadmap at hand is a result of many discussions and sessions we had at the second seminar on Plan-based Control of Robotic Agents.

5.1.1.   The Robot Planning TCU

The Technical Coordination Unit "Robot Planning" has been a very successful and important platform for research on plan-based robot control in Europe and has fueled the cooperation and exchange of ideas of Europe's main research laboratories in plan-based robot control. Among many other activities the TCU has organized two Dagstuhl seminars on "Plan-based Control of Robotic Agents", which attracted not only European researchers but also the leading international researchers. In particular, the second of these seminars has provided valuable and substantial input to this roadmap document. Besides the seminars the TCU was actively involved in two of the PLANET summer schools and produced courses for these summer schools.

5.1.2.   AI P&S in robotics and its requirements

In recent years, autonomous robots, including WITAS [DGK+00], MARTHA [AFH+98], RHINO [BCF+00], MINERVA [BAB+01], and REMOTE AGENT [MNPW98b], GRACE [SGG+03], and XAVIER [SGH+97b]. have shown impressive performance in longterm demonstrations. In NASA's Deep Space program, for example, an autonomous spacecraft controller, called the *Remote Agent* [MNPW98a], has autonomously performed a scientific experiment in space. At Carnegie Mellon University XAVIER [SGH+97b], another autonomous mobile robot, has navigated through an office environment for more than a year, allowing people to issue navigation commands and monitor their execution via the Internet. In 1998, MINERVA [TBB+00] acted for thirteen days as a museum tour guide in the Smithsonian Museum, and led several thousand people through an exhibition.

These autonomous robots have in common that they perform plan-based control in order to achieve better problem-solving competence. In the plan-based approach robots generate control actions by

maintaining and executing a plan that is effective and has a high expected utility with respect to the robots' current goals and beliefs. Plans are robot control programs that a robot cannot only execute but also reason about and manipulate [McD92]. Thus a plan-based controller is able to manage and adapt the robot's intended course of action –the plan– while executing it and can thereby better achieve complex and changing tasks [PH99]. The plans used for autonomous robot control are often reactive plans, that is they specify how the robots are to respond in terms of low-level control actions to continually arriving sensory data in order to accomplish their objectives [LA89]. The use of plans enables these robots to flexibly interleave complex and interacting tasks, exploit opportunities, quickly plan their courses of action, and, if necessary, revise their intended activities.

To be reliable and efficient, autonomous robots must flexibly interleave their tasks and quickly adapt their courses of action to changing circumstances [Bee02]. Recomputing the best possible course of action whenever some aspect of the robot's situation changes is not feasible but can often be made so if the robots' controllers explicitly manage the robots' beliefs and current goals and revise their plans accordingly [BIP88]. The use of plans helps to mitigate this situation in at least two ways. First, it decouples computationally intensive control decisions from the time pressure that dominates the feedback loops. Precomputed control decisions need to be reconsidered only if the conditions that justify the decisions change. Second, plans can be used to focus the search for appropriate control decisions. They can neglect control decisions that are incompatible with its intended plan of action.

The present Roadmap describes a broadly supported view of where our field is going, both in terms of research as well as in terms of technology development. The Roadmap should enable researchers, developers, and funders to identify the main challenges in order to concentrate our efforts, set intermediate milestones, and measure the progress of the field. It also identifies important research directions that have to be followed in order to speed up the advance of our field.

What is robot planning and plan-based robot control?

Before we discuss the current research efforts, challenges, and future developments, we will first introduce the most important concepts of the field. We borrow a number of terms and their definitions from McDermott's [McD92] survey article on robot planning.

Let us start by stating what autonomous **robot control** is. It is the computational task of specifying how the robot is to respond to sensory data in order to accomplish a specified set of jobs. The response to the sensory data is given in terms of low-level commands for the robot's effectors including its drive, arms, cameras, etc. Key characteristics of the robot control are that sensor data typically arrive asynchronously as data streams. Also sensor data are often incomplete, inaccurate, and sometimes even incorrect. Therefore, sensor data are often interpreted probabilistically to compute a belief state with respect to the current state of the robot and its environment. Robots are equipped with several sensors and effectors which operate in parallel. As a consequence controlling a robot requires concurrent threads of control. Finally, the robots' effectors are not capable of operating perfectly. Actions might fail or actions, like movement actions, might have inaccurate effects.

A key preliminary of plan-based robot control is that we require the robots' behavior to be specified by **(control) programs**. A program is a formally representable object with a specified semantics that produces the robot's behavior typically in response to the data received by the robots' sensors.

Restricting ourselves to autonomous robots controlled by programs is not a serious limitation because almost all autonomous robots solving even only somewhat sophisticated tasks are controlled by programs. Arguably robots that are completely controlled by artificial neural networks might be an exception.

After we have agreed upon looking at robots controlled by programs we might ask ourselves what features the control programs must exhibit in order to produce good behavior. [LA89] state a number of requirements for robot control programs. First, control programs must react to the environment under real time conditions (cf. [Ark98]). Second, control programs must specify concurrent behavior because the robots' sensors and effectors operate in parallel. In addition, the robots' tasks might comprise concurrent subtasks such as monitoring the environment while navigating to a particular destination. Third, robots, such as robot couriers, manipulate objects in the environment. Because the control routines, such as grasping, cannot be called with the objects themselves but only with object descriptions acquired by sensing routines we have to be careful about how to "ground" the object descriptions in the real world [CS00, McD90]. Fourth, because sensing as well as acting is fundamentally failure prone robot control programs must continually monitor the execution of programs and provide mechanisms for failure detection, analysis, and recovery, and provide mechanisms for exception handling.

Many of these issues are also raised in the context of behavior-based programming [Bro91, Ark98] and in the context of control theory, in particular in the area of feedback control of dynamical systems [DW91, PA89]. The discussion of the characteristics of robot programming implies that the robot control programs are very complex software systems that contain components for control program interpretation, process management, software development tools, libraries of effector control routines such as grasping and holding, etc.

So why do robots need planning? Intuitively, planning is deciding on the course of action based on foresight rather than immediate physical experiences [BHGP02]. In the context of robot control, the need for planning arises as soon as the robot has to perform multiple possibly interacting tasks or as the set of tasks is dynamically changing. In these situations, it is most of the time not possible to preprogram behavior specifications for all conceivable sets of tasks. Thus, to be reliable and efficient, autonomous robots must flexibly interleave their tasks and quickly adapt their courses of action to changing circumstances [Bee00]. Recomputing the best possible course of action whenever some aspect of the robot's situation changes is not feasible but can often be made so if the robots' controllers explicitly manage the robots' beliefs and current goals and revise their plans accordingly. Another condition under which the need for planning arises is when robots are required to solve novel tasks or optimize their courses of action without relying on an extensive body of physical experience in accomplishing the respective tasks. Under these circumstances planning mechanisms will construct these special purpose controllers based on their predicted performance.

To allow for planning part of the control program has to be represented as a **plan**. We define according to [McD92] the plan to be that part of the robot's program whose future execution the robot reasons about explicitly. The use of plans helps to mitigate the problem of continually recomputing the best course of action in at least two ways. First, it decouples computationally intensive control decisions from the time pressure that dominates the feedback loops. Precomputed control decisions need to be reconsidered only if the conditions that justify the decisions change. Second, plans can be used to focus the search for appropriate control decisions. They can neglect control decisions that

are incompatible with its intended plan of action.

Plans in plan-based control have two roles. They are both executable prescriptions that can be interpreted by the robot to accomplish its jobs and syntactic objects that can be synthesized and revised by the robot to meet the robot's criterion of utility. Thus plans are resources for the robot controller that enable the controller to exhibit better behavior and better problem solving performance. Besides having means for representing plans, plan-based controllers must also be equipped with tools that enable planning processes to (1) project what might happen when a robot controller gets executed and return the result as an execution scenario; (2) infer what might be wrong with a robot controller given an execution scenario; and (3) perform complex revisions on robot controllers.

We consider **robot planning** to be the automatic generation, refinement, revision, and optimization of robot plans. As a computational problem it can be formulated as follows: Given an abstract plan P and a description of the current situation, find an executable realization Q that maximizes some objective function V. In this problem formulation an abstract plan might be go shopping and clean up the apartment, win a robot soccer game, monitor the traffic in a particular area using an autonomous helicopter, or, for a museum tour guide robot, inform and entertain the visitors of the museum. These abstract plans include achievement goals, maintenance goals, and information gathering goals.

In recent years, a number of researchers have argued that robot planning should not be studied in isolation but rather in conjunction with other computational tasks that are to be performed on plans [PH99, BHGP02]. These tasks include plan execution, monitoring, diagnosis, learning, time management, and several others. In the plan-based approach robots generate control actions by maintaining and executing a plan that is effective and has a high expected utility with respect to the robots' current goals and beliefs. Thus a **plan-based controller** is able to manage and adapt the robot's intended course of action –the plan– while executing it and can thereby better achieve complex and changing tasks. The use of plans enables these robots to flexibly interleave complex and interacting tasks, exploit opportunities, quickly plan their courses of action, and, if necessary, revise their intended activities.

The domain of plan-based robot control

In this section we will characterize the relationship of the research field of robot planning and other closely related research fields. These research fields are robotics, planning, reasoning about action, autonomous agents, and machine learning.

Plan-based robot control and robotics/control theory   The difference between plan-based robot control and robotics/control theory is mainly one of emphasis. Robotics and control theory mainly focus on the complexity of the physical system that is to be controlled and assume control tasks that are fairly simple [DW91, PA89]. Consequently, the controlled system is often modeled very accurately whereas the control task is often specified as a trajectory of state variable values that is to be achieved. The design of control system focuses on controllers that are and can be shown to be stable and controllable.

In contrast to control theory and robotics, robot planning often uses fairly abstract models of the controlled system, at least by standards of control theory. Instead, robot planning focuses on issues

of task complexity. Consequently, robot planning has developed the techniques to manage multiple, dynamically changing, interacting, and interfering tasks.

Because of their complementary strengths and weaknesses many of the successful plan-based control systems use hybrid approaches using control-theoretic as well as plan-based control mechanisms. Typically, control-theoretic approaches are applied to fairly homogeneous control tasks such as grasping and motion control. Plan-based mechanisms than configure these controllers and reason about possible interactions between different low-level control routines (see, for example, [ACF+98]).

Plan-based robot control and planning Enabling robots to plan their own actions is different from providing planning tools for logistics, production, or work flow management: the inputs for robot action planning are mostly generated from the robot's sensors and the outputs are used to control the robot's physical behavior. As a consequence, robot action planning poses several unique and characteristic challenges to planning research. In particular, the information provided by the robots' sensors is typically incomplete and corrupted by noise, robot actions are non-deterministic, plan execution is often required to react to unanticipated events, and robot action planners cannot fall back on humans to make difficult decisions for them.

In contrast to many other applications of AI planning [GNT04], plan-based control aims at optimizing the *behavior* that is produced by the plan. The plan itself is only a side effect. Therefore, additional issues such as the assessment of the behavior and the embedding of plans in the overall robot control program must be addressed. Unlike in many other AI planning applications robot planning posts strong requirements with respect to spatial, temporal, and spatio-temporal reasoning capabilities. In addition, plan-based robot control must thoroughly address the issues of producing plans under uncertain information and with limited computational resources. Finally, in plan-based robot control a number of special purpose planning mechanisms for specific problems such as navigation, grasping, and perception have been developed.

Plan-based robot control and reasoning about actions The area of reasoning about actions is one of the formal foundations of robot planning [AKPT90, AF94, McD85, McD78, LRL+97]. The idea is to come up with axiomatizations of the actions of robots, their effects and possible interferences, and of the semantics of control programs. These axiomatizations can then be used to better understand the planned behavior of robots and to formally prove important properties of plan-based robot control systems. To fully achieve these goals for plans used in autonomous robot control current approaches have to be substantially extended and fundamental problems like the interference of concurrent and continuous actions and the grounding of logical symbols in particular in object manipulation tasks have to be solved.

Plan-based robot control and autonomous agents As autonomous robots are specific forms of *physical* autonomous agents, plan-based robot control can also be viewed as a subfield of the autonomous agents research area. Several models of autonomous agency, in particular the belief-desire-intention model and its formalizations are extremely important and relevant to the plan-based control of robotic agents. At the same time, plan execution languages such as PRS [ICAR96], RPL

[McD91], and RAP [Fir89] have been shown to apply equally well to other agent applications such as the control of supply chains.

Other related disciplines   Other related disciplines include model-based reasoning [WICE03, MNPW98b, WN97, WN96], learning [Mah96, TM93, HV98, SPS98, BD95, AR01], decision-theoretic planning [Bly99, KCK96, RM01], in particular research on solving various variants of Markov Decision Problems [KCK96, SB98, KLM96].

Enabling technologies   It is also important to point out that a number of enabling technologies have made substantial progress over the last years. These technologies include the dissemination of entertainment robots such as the Sony AIBOS. More sophisticated robots with impressively rich sensory and actuator equipments, such as the humanoid *Sony SDR-4X* robots can be expected to enter the commercial market within the next years. In addition to the robots themselves, multimedia computers equipped with web cameras and other sensory and human computer interaction facilities are becoming very cheap. Finally, the advent of standard public-domain software libraries, including the robot simulators Stage and the vision software library OpenCV will also fuel the progress in our field.

A framework for robot planning and plan-based control

The building blocks of plan-based control are the representation of plans, the execution of plans, various forms of automatic learning, and reasoning about plans, including plan generation and transformation, and teleological, causal, and temporal reasoning.
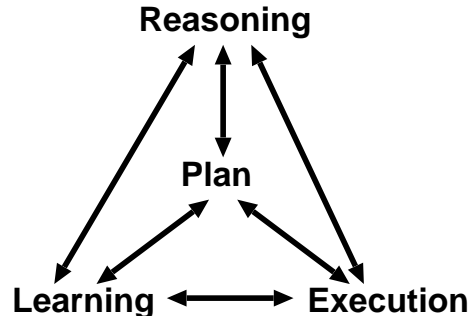
Figure 5.1.: The main components of plan-based control are plan representation, execution, learning, and reasoning and their interactions.

But before we dive in and discuss the building blocks of modern plan-based control models let us first get an intuition of how traditional robot planning techniques function. Most of these techniques are based on the problem-space hypothesis [New90]: they assume problems can be adequately stated using a state space and a set of discrete and atomic actions that transform states to successor states. A solution is an action sequence that transforms any situation satisfying a given initial state description

into another state that satisfies the given goal. Plan generation is the key inferential task in this problem-solving paradigm.

As a consequence, representational means are primarily designed to simplify plan generation from first principles. Problem space plans are typically used in layered architectures [BFG+97], which run planning and execution at different levels of abstraction and time scales. In these approaches planning processes use models that are too abstract for predicting all consequences of the decisions they make and planning processes cannot exploit the control structures provided by the lower layer. Therefore they lack appropriate means for specifying flexible and reliable behavior and plans can only provide guidelines for task achievement.

Contrary to the plan space approach, plan-based control of robotic agents takes the stand that there is a number of inference tasks necessary for the control of an autonomous robot that are equally important. These inference tasks include ones that enable the competent execution of given plans, ones that allow for learning plans and other aspects of plan-based control, and various reasoning tasks, which comprise the generation and assessment of alternative plans, monitoring the execution of a plan, and failure recovery.

These different inference tasks are performed on a common data structure: the plan. Consequently, the key design issues of plan-based control techniques are representational and inferential adequacy and inferential and acquisitional efficiency as key criteria for designing domain knowledge representations [RK91]. Transferring these notions to plan-based control, we consider the representational adequacy of plan representations to be their ability to specify the necessary control patterns and the intentions of the robots. Inferential adequacy is the ability to infer information necessary for dynamically managing, adjusting, and adapting the intended plan during its execution. Inferential efficiency is concerned with the time resources that are required for plan management. Finally, acquisitional efficiency systems is the degree to which they support the acquisition of new plan schemata and planning knowledge.

To perform the necessary reasoning tasks the plan management mechanisms must be equipped with inference techniques to infer the purpose of sub-plans, find sub-plans with a particular purpose, automatically generate a plan that can achieve some goal, determine flaws in the behavior that is caused by sub-plans, and estimate how good the behavior caused by a sub-plan is with respect to the robot's utility model. [PH99] stress the point that maintaining an appropriate and working plan requires the robot to perform various kinds of plan management operations including plan generation, plan elaboration, commitment management, environment monitoring, model- and diagnosis-based plan repair, and plan failure prediction.

It does not suffice that plan management mechanisms can merely perform these inference techniques but they have to perform them fast. The generation of effective goal-directed behavior in settings where the robots lack perfect knowledge about the environment and the outcomes of actions and environments are complex and dynamic, requires robots to maintain appropriate plans during their activity. They cannot afford to entirely re-plan their intended course of action every time their beliefs change.

To specify competent problem-solving behavior the plans that are reasoned about and manipulated must have the expressiveness of reactive plan languages. In addition to being capable of producing flexible and reliable behavior, the syntactic structure of plans should mirror the control patterns that cause the robot's behavior – they should be realistic models of how the robot achieves its intentions.

Plans cannot abstract away from the fact that they generate concurrent, event-driven control processes without the robot losing the capability to predict and forestall many kinds of plan execution failures. A representationally adequate plan representation for robotic agents must also support the control and proper use of the robot's different mechanisms for perception, deliberation, action, and communication. The full exploitation of the robot's different mechanisms requires mechanism-specific control patterns. Control patterns that allow for effective image processing differ from those needed for flexible communication, which in turn differ from those that enable reliable and fast navigation. To fully exploit the robot's different mechanisms, their control must be transparently and explicitly represented as part of the robot's plans. The explicit representation of mechanism control enables the robot to apply the same kinds of planning and learning techniques to all mechanisms and their interaction.

The defining characteristics of plan-based control is that these issues are considered together: plan representation and the different inference tasks are not studied in isolation but in conjunction with the other inference tasks. The advantage of this approach is that we can exploit synergies between the different aspects of plan-based control.

Plan management capabilities simplify the plan execution problem because programmers do not have to design plans that deal with all contingencies. Rather plans can be automatically adapted at execution time when the particular circumstances under which the plan has to work are known. Plan execution mechanisms can also employ reasoning mechanisms in order to get a broader coverage of problem-solving situations. The REMOTE AGENT, for example, employs propositional reasoning to derive the most appropriate actions to achieve the respective immediate goals [WN97, NW97]. On the other side, competent plan execution capabilities free the plan management mechanism from reasoning through all details. Reasoning techniques such as diagnostic and teleological reasoning are employed in transformational learning techniques in order to perform better informed learning decisions and thereby speed up the learning process [BB00]. Skill learning mechanisms have also been applied to the problem of learning effective plan revision methods [Sus77]. There is also a strong interaction between the learning and execution mechanisms in plan-based control. Learning mechanisms are used to adapt execution plans in order to increase their performance . Competent execution mechanisms enable the learning mechanisms to focus on strategical aspects of problem-solving tasks.

5.2. Roadmap themes

Let us now sketch the state of the art and recent research strands and directions in the field of robot planning.

5.2.1. Applications and deployment

State of the art and current research

In recent years, there have been a number of longterm real-world demonstrations of plan-based control, which have impressively shown the potential impact of this technology for future applications of autonomous service robots. In NASA's Deep Space program a plan-based robot controller, called the

Remote Agent, has autonomously controlled the performance of a scientific experiment in space. In the *Martha* project, fleets of robots have been effectively controlled and coordinated. *Xavier*, an autonomous mobile robot with a plan-based controller, has navigated through an office environment for more than a year, allowing people to issue navigation commands and monitor their execution via the Internet. In 1998, *Minerva*, another plan-based robot controller, acted for thirteen days as a museum tour guide in the Smithonian Museum, and led several thousand people through the exhibition.

Plan-based control of autonomous helicopters for traffic surveillance and assistance In the WITAS project [DGK+00] the control software of a small, unmanned helicopter carrying computers, video cameras, and other electronic equipment is developed. The task of the helicopter is to observe what goes on on the ground, in particular traffic situations on roads, and to make decisions on that basis. It is therefore required to "understand" what happens on those roads – conventional maneuvers of individual cars and other road vehicles, dangerous or otherwise exceptional maneuvers, structure of the traffic e.g. congestion. It is also required to perform tasks that are assigned by the operator or triggered by the observations it makes itself, for example to follow a certain car that flees from the scene of an apparent crime, or to assist a certain car so that it can make it through difficult traffic and get to a particular destination as quickly as possible, or to deliver a particular parcel to a particular point. The helicopter is instructed by a human in very abstract terms. The most important capabilities for the WITAS system are therefore (1) to form a model of ("understand") scenes and events that it observes on the ground, and (2) to predict, plan, and make decisions using that model.

Plan-based control of tour guide robots [BAB+01] implemented a plan-based high-level controller for an interactive museum tour guide robot. The tour guide robot, called MINERVA, has operated for a period of thirteen days in the Smithsonian's National Museum of American History [TBB+99]. In this period, it has been in service for more than ninety four hours, completed 620 tours, showed 2668 exhibits, and traveled over a distance of more than forty four kilometers. The plan-based controller controlled MINERVA's course of action in a feedback loop that was carried out more than three times a second. MINERVA used plan revision policies for the installment of new commands, the deletion of completed plans, and tour scheduling. MINERVA performed about 3200 plan adaptations. The MINERVA experiment demonstrates that plan-based controllers can (1) reliably control an autonomous robot over extended periods of time and (2) reliably revise plans during their execution.

Plan-based robot control for transshipment tasks The Martha project [AFH+98] has investigated the management of a fleet of autonomous mobile robots for transshipment tasks in harbors, airports and marshaling yards. In such contexts, the dynamics of the environment, the impossibility to correctly estimate the duration of actions (the robots may be slowed down due to obstacle avoidance or re-localization actions, and delays in load and unload operations, and so on) prevent a central system from elaborating long or medium term efficient and reliable detailed robot plans.

In the Martha project, a more flexible way is pursued in which the robots to determine incrementally the resources they need taking into account the execution context. As a consequence, [AIQ98] have developed advanced techniques for plan merging in order to improve multi robot cooperation.

Plan-based control for autonomous spacecrafts   The Remote Agent is a reusable control system that enables goal-based spacecraft commanding and robust fault recovery [MNPW98a]. The Remote Agent accepts high level goals from the operators and on-board software modules such as the autonomous navigator. The Remote Agent determines a plan of action that will achieve those goals and carries out that plan by issuing commands to the spacecraft. The Remote Agent considers the current state of the spacecraft both in planning how to achieve its goals and in executing that plan. This allows for robust responses to failures and other contingencies. The Remote Agent detects and responds to failures in real-time and if necessary generates a new plan for achieve its remaining goals. The REMOTE AGENT has successfully operated the Deep Space 1 spacecraft for two days in May of 1999. This experiment was the first time that an autonomous agent has controlled a deep space mission.

Other plan-based robot controllers   Other plan-based robot controllers include XAVIER, an autonomous mobile robot that has performed longterm navigation experiments in an office building [SGH+97a]. XAVIER has been in nearly daily use for more than one year and traveled in this period more than 75 kilometers in order to satisfy over 1800 navigation requests that were specified using a World Wide Web interface. [FPS+96] have developed CHIP, an autonomous mobile robot that is intended to serve as a general-purpose robotic assistant. The main application demonstration of CHIP was the Office Cleanup event of the 1995 Robot Competition and Exhibition. CHIP was to scan an entire area systematically and, as collectible objects were identified, pick them up and deposit them in the nearest appropriate receptacle. Their work is very interesting because they have developed a library of low-level plans that can be carried out in situation-specific ways.

Research goals and challenge application scenarios

The control of autonomous service robots can be made more challenging along several dimensions: the capabilities of the robot, the nature of the environment, and the complexity of the tasks to be performed.

In this section we present a selection of challenge application scenarios that have been developed during the second seminar "Plan-based Control of Robotic Agents". The scenarios are characterized by varying degrees of difficulty along these dimensions. The different application scenarios are: autonomous robots with sophisticated manipulation skills, the robot companion, autonomous spacecraft control, plan-based control in intelligent, sensor-equipped environments, and the autonomous household robot.

Autonomous robots with sophisticated manipulation skills   Our first challenge application scenario is to have a robot or a team of robots that are mobile and equipped with arms and hands. The robots are to assemble either pieces of Ikea furniture or assemble specified models using toy construction sets such as Lego, Fischer Technik, Baufix, etc. Thus, the objective is the development of mobile robots with manipulators that are taskable in natural language and are teachable by demonstration.

The reason that we propose this challenge scenario is that it exemplifies a number of the hard problems in robot planning and plan-based robot control. First, adding manipulation will increase the

complexity of the tasks the robots can do, and so will require more high-level planning and learning than what is currently needed. Second, communication in natural language requires advances in user interaction and the sensor-based grounding of symbolic descriptions in visual scenes. In particular, the robot needs to disambiguate between sensed objects that might satisfy a given symbolic description. Also, planning mechanisms need to be extended to deal with symbolic object descriptions that might be incorrect, inaccurate, incomplete, or ambiguous. Another research challenge is the application of planning techniques for speeding up learning by demonstration and/or being told. A fourth issue is that such open assembly tasks require new planning mechanisms that properly integrate symbolic task planning with motion, grasp, and roadmap planning. Finally, the robots need much more realistic models of construction plans and designs of pieces and what can be done with them as it is typical for many of the recent planning application tasks.

In the case where the robots are to assemble furniture from instructions additional challenges such as the manipulation of large objects and the cooperative manipulation with multiple robots are posed. In addition, the tasks have to be dynamically distributed among the robots in the team.

Robot companion   Our second application challenge scenario is the development of a robot that serve elderly people as a companion in their daily life. Such robot companions must be able to evolve and grow their capacities in close interaction with humans in an open-ended fashion. Quite intuitively the notion of companion includes a variety of issues. The robot must provide continual operation over long periods of time; provide sufficient coverage and monitoring without being intrusive; maintain the privacy of the users; interact with users and guarantee their safety. The robot is not only considered as a ready-made device but as an artificial creature, which improves its capabilities in a continuous process of acquiring new knowledge and skills. Besides the necessary functions for sensing, moving and acting, such a robot will exhibit the cognitive capacities enabling it to focus its attention, to understand the spatial and dynamic structure of its environment and to interact with it, to exhibit a social behavior and communicate with other agents and with humans at the appropriate level of abstraction according to the context.

From the point of view of plan-based robot control a robot companion needs planning mechanisms that are very different from those needed for construction set assembly. The companion challenge emphasizes mechanisms for integrating additional plans into existing commitments, elaborating and expanding them as needed, and monitoring their execution.

Besides the technological challenges the application challenge scenario also holds the potential for making very substantial contributions to the ever more important societal problem of aging populations. We believe that because of privacy considerations autonomous robots are expected to achieve better acceptance than intelligent sensor-equipped rooms.

A variant of the robot companion for elderly people is the deployment of plan-based robots that can navigate in a crowd in large environments (Disneyland, large shopping malls), take people who are lost to their destination, guide tours, carry suitcases, etc. Technical challenges posed by this scenario include the navigation and mapping of large areas requires advances in mapping, cooperation of multiple robots and sensors; the navigation in a crowd requires fast vision processing, detecting people intentions, using social conventions, etc.; and finding people who need help requires detecting user distress level, interacting with them, and adapting to their reactions.

Spacecraft control Plan-based robot control already plays an important role in NASA's research agenda for the exploration of space, the investigation of dangerous places on earth, including volcanos, and the probing of other planets and orbs. The deployment of intelligent autonomous robots and spacecraft enables NASA to perform more science missions at lower cost, perform science experiments that are much deeper in space, to take advantage of unexpected science opportunities, and run unmanned spacecraft missions of longer durations. The autonomy and plan-based control techniques therefore allow for running a greater variety of missions at lower cost.

Planned NASA missions include sending teams of planetary rovers to the Mars in order to explore the surface more thoroughly, orbiter missions around the Jupiter moon Europa, and various deep space missions. The Jupiter moon Europa is believed by a number of astrobiologists to provide conditions that might enable the evolution of a biosphere. Unfortunately this biosphere lies in a hypothesized ocean under a very thick crust of ice. Eventually astrobiologists intend to send an autonomous hydrobot to Europa that drills itself through the ice, explores the ocean, and sends back the results from the scientific experiments. Obviously, such a mission would require sophisticated planning capabilities on the robot.

Other example applications of robot planning include: spacecraft commanding and payload operations; planning and scheduling for process control; planning and scheduling for robotic space activities; operations of air, space and ground-based scientific observatories; scheduling of critical resources whether on the ground or on-board; science data analysis; design and analysis of spacecraft systems; planning and scheduling of scientific experiments; and planning and scheduling of crew activities.

Space applications of robot planning entail a number of important research questions. First, they imply extremely high demands on reliability, safety, and robustness. They also have require sophisticated task planning and scheduling techniques, true autonomy, and reasoning about detailed models of the controlled system.

Household robot challenge Realizing an autonomous household robot with sophisticated manipulation skills is an important longterm challenge application for the domain of plan-based robot control. In the last Dagstuhl seminar on "Plan-based Control of Robotic Agents" it was selected as a hallmark problem for guiding the further development of plan-based robot control techniques.

We consider a humanoid robot, such as the Sony SDR-3 or the Honda Asimov, with additional manipulation skills that is to do household chores as an interesting challenge for the field of plan-based control of autonomous robots. The challenge is to develop a plan-based controller for such a robot that enables the robot to be put in another household, to operate in this household for some months, and do a substantial part of the household chores satisfactorily. To meet this challenge, the robot must acquire models of its environments not only a map of the apartment but also models of the daily rhythm of the household, the time the dishwasher takes to clean the dishes, etc. and it must use these models to better manage its activities. It should also acquire or generate plans to perform its tasks, such as cleaning the living room and it should build up models of these activities that include information such as how long it typically takes to clean up the rooms. The robot is to do several things at a time for example, cleaning while baking a cake. These activities have to be interruptible: the phone might ring in the midst of cooking. Many of the activities require interactions between the robot and the environment and considering actions, such as cleaning the kitchen, to be discrete

does not suffice in many situations. These are only some of the aspects that require much more sophistication in controlling such robots than is provided by current robot planning mechanisms. Develop a robot that acquires knowledge from ontologies.

Other scenarios   Other interesting application challenge scenarios include but are not limited to RoboCup Rescue, logistics scenarios, intelligent sensor-equipped living environments, autonomous robot soccer, and plan-based driver assistance systems.

The goal in RoboCup Rescue is to provide emergency decision support by integration of disaster information, prediction, planning, and human interface. A generic urban disaster simulation environment is provided. Heterogeneous intelligent agents such as fire fighters, commanders, victims, volunteers, etc. conduct search and rescue activities in this virtual disaster world. Real-world interfaces such as helicopter image synchronizes the virtuality and the reality by sensing data.

Thus, RoboCup Rescue studies disaster rescue after catastrophes like earth quakes as a testbed for research in controlling very large numbers of heterogeneous agents in hostile environments. Research issues include multi-agent team work coordination, physical robotic agents for search and rescue, information infrastructures, decision-theoretic planning techniques for decision support systems, rescue strategies and robotic systems that are all integrated into a comprehensive systems in future. Interesting planning challenges include multi-agent planning, real time/anytime planning, heterogeneity of agents, robust planning, mixed-initiative planning.

Another interesting scenario is the control and the management of a fleet of autonomous mobile robots for transshipment tasks in harbors, airports and marshaling yards and supply chain monitoring and control in general. Here, plan-based robot control and scheduling techniques can plan and execute supply chains that are faster, and can react more spontaneous to problems.

Autonomous robot soccer is another challenge domain. Here, the dynamics of the game situations prevent the application of sophisticated online plan generation mechanisms. Yet, plans, plan-based control, and playbook learning seem to be important for acquiring sophisticated playing skills. A last domain that we would like to mention here is the enhancement of intelligent driver assistants systems through plan-based control mechanisms.

5.2.2.   Representation for plan-based control

State of the art and current research

Planning domain description language   One important driving force in the area of AI planning research is PDDL (Planning Domain Description Language) [McD98]. PDDL was originally developed by Drew McDermott and the 1998 planning competition committee. It was inspired by the need to encourage the empirical comparison of planning systems and the exchange of planning benchmarks within the community. Its development improved the communication of research results and triggered an explosion in performance, expressivity and robustness of planning systems. Since 1998 a number of extensions for more realistic planning problems have been proposed. PDDL2.1, which was used for the first time in the 2002 AI planning system competition, handles time and durations of actions [FL03]. Further enrichments to the modeling of hybrid and real-time systems (PDDL+) were proposed, while another committee is currently investigating extensions to probabilistic planning.

PDDL has become a de facto standard language for describing planning domains, not only for the competition but more widely, as it offers an opportunity to carry out empirical evaluation of planning systems on a growing collection of generally adopted standard benchmark domains. The emergence of a language standard will have an impact on the entire field, influencing what is seen as central and what peripheral in the development of planning systems.

Whether there should be a variant for robot planning is an issue of debate but even if not the future PDDL versions that include probabilistic as well as parallel and continuous models of actions and exogenous events will have a profound impact on the future development of our research field.

Plan languages   In addition to representation of planning domain descriptions, the representation of plans is also an active area of research. Most recent autonomous robots cannot fully exploit plans as resources for better problem-solving performance because of imminent limitations of their plan representations. Most plan representations are designed with issues of plan generation from first principles in mind but ignore other issues such as support for execution and learning.

A more comprehensive view of representation is offered by Rich and Knight ([RK91]). They propose representational and inferential adequacy and inferential and acquisitional efficiency as key criteria for designing domain knowledge representations. Transferring these notions to plan representation, we consider the representational adequacy of plan representations to be their ability to specify the necessary control patterns and the intentions of the robots. Inferential adequacy is the ability to infer information necessary for dynamically managing, adjusting, and adapting the intended plan during its execution. Inferential efficiency is concerned with the time resources that are required for plan management operations and adaptation. Finally, acquisitional efficiency systems is the degree to which they support the acquisition of new plan schemata and planning knowledge.

Several researchers have started to design special purpose plan representations for robotic agents that aim at being representationally and inferentially adequate as well as inferentially and acquisitionally efficient. Such plan languages provide substantial improvements over existing plan representations in that they support the proper integration of different mechanisms for perception, deliberation, action, learning, and communication and that they provide fast built-in causal and teleological inference and manipulation mechanisms for such expressive plan languages.

Models of dynamical systems   Another active research strand is the investigation of better models for the physical behavior of robots and their environments. So far, many robot planning systems use crude models of the physics of the system they control and the environment they act in. Sometimes the physics is completely represented within the action models used for planning. What are the gains in developing model-based robot planners?

We have already pointed out that in the area of control theory much more detailed and accurate models of physical evolution of systems are developed. Also, the extensions of the Planning Domain Description Language, in particular PDDL+, allow for the modeling of dynamic systems in terms of hybrid automata.

While these are encouraging first steps a substantial amount of research needs to be done to enable planning systems to reason about exogenous events with continuous effects [McD03, BG00]. We will see new approaches in this direction within the next couple of years.

Hybrid representations A complementary strategy to finding one master representation that helps you do as much reasoning consistently as you need, is finding mechanisms that help you switch between different representations of the world with rel-atively low effort and transfer reasoning results accordingly. Besides being an old hat in AI, the issue of multiple representations is pervasive in robotics, where you have synchronous, multiple sensor data, streams of sensor data, control data, and data from user interaction anyway, all of which represent overlapping portions of the same part of the world. Layered control architectures add degrees of granularity to that.

5.2.3. Learning

State of the art and current research

Learning has been an important subtask within plan-based control ever since Shakey the robot [Nil84]. Learning is critical for plan-based robot control and robot planning because in complex and dynamic environments the robots cannot be equipped with complete and accurate models of their actions and the physics of their environments. Hence, it is critical for the robots to be able to first, automatically acquire and adapt the models they use for controlling the robot and second, the routines that can be expected to work well in the particular setting.

In recent years, learning in the context of plan-based robot control is mainly used for plan-execution, in particular the execution of low-level plans [BBC02]. The main reason is that action planning systems generate action sequences that achieve goals, but do in general not produce robot plans that are optimized for execution.

There are two main applications of learning in current plan-based robot control systems. The first one is the learning of action models, in particular expectations about their durations and probability distributions about their effects (cf. models for Semi Markov Decision Problems) [SPS98]. Another aspect of action models is the decomposition and the structuring of continuous actions into appropriate sub-actions (behavior segmentation) [BB00]. Finally, substantial advances have been made by representing low-level control routines as Markov Decision Problems and applying reinforcement learning methods in order to optimize control routines with respect to given objective or cost functions.

The second application of learning in plan-based robot control is the application of transformational planning techniques for learning better execution routines for given plan steps. Again these techniques have been applied in the context of navigation tasks.

Others have learned models of exogenous events, such as typical trajectories of people in a household and used these models for avoiding interferences with the motions of people.

A limitation of current work in learning is that the application of learning techniques is mostly limited to navigation tasks. The main reason for this limitation is that very few autonomous robots have sophisticated object recognition and manipulation mechanisms.

5.2.4. Plan management and reasoning

State of the art and current research

The demands of dynamic, uncertain environments mean that in addition to being able to form plans –even probabilistic, uncertain plans– agents must be able to effectively manage their plans [PH99]. That is, they need to be able to decide which planning problems to consider in the first place. They need to be able to form incomplete plans now, adding detail later, and they thus need to be able to decide how much detail to include now and when to add more detail. They need to be able to weigh alternative incomplete plans, and to decide among competing alternatives. They need to be able to integrate plans with one another, and to decide when to treat an existing plan as an inflexible commitment and when, instead, to consider modifications of it. And they need to be able to do all this in a way that comports with the inherent bounds on their computational resources.

It is obvious that temporal constraints matter in dynamic environments: the agent needs to remember how long the cake has been in the oven, for example, or if washes the floor, it needs to consider what other tasks it can do while the floor is drying. Temporal uncertainty may also be important: it may be impossible to tell in advance exactly how long it will take for the cake to bake (or the floor to dry), and the robot will have to reason with such uncertainty.

5.2.5. Plan execution

State of the art and current research

Over the years, researchers in the robot planning community have made substantial progress in plan execution and in the realization of systems in which planning and execution are continually interleaved and actively managed. Indeed, many fielded autonomous robots use sophisticated plan execution systems including RPL [McD91], Proprice [DI99], RMPL (Reactive Model-based Programming Language) [WICE03], RAP [Fir87], ESL [Gat96], and PRS [ICAR96]. These plan execution systems typically support, to a greater or lesser extent, concurrent execution, decomposition of tasks into sequences of subtasks, resource management, execution monitoring, failure detection and analysis, exception handling, and execution recovery strategies. In RPL, for example, the concept of plan failure is deeply integrated into the semantics of the plan language and control structures are provided that signal failures, respond to failures, and recover from them.

In early plan execution languages reasoning was almost completely eliminated from the execution systems. The main reason was that the programmers wanted to ensure that the feedback control loops ran fast enough to guarantee the safety of the robots and the other inhabitants of the environment.

Over the years, however, computational devices have become much faster and algorithms for certain reasoning tasks much more efficient. Thus, the shortness of computational resources has been mitigated. Therefore, recently researchers have begun to put lightweight reasoning capabilities into the plan execution languages. [WICE03], for example, have integrated a parsimonious form of model-based, propositional planning and others have developed mechanisms for the integration of execution time planning. It has been recognized that the incorporation of such light-weight reasoning capabilities into the plan execution languages results in low level control routines that are much more concise,

more robust, and more adaptive. The further development of plan execution languages is an active research focus.

Unfortunately, the development of general, multi-purpose, low-level plan libraries for autonomous service robots has received surprisingly little attention. The work of [FPS⁺96] present a notable exception. They have developed a library of robot skills and plans that can be used to achieve a variety of everyday tasks. Thus, while addressing specific robot control tasks, they were building up generic skills and low level plans that can be reused as building blocks in future service robot applications.

5.2.6. Integrated plan-based control systems

State of the art and current research

The field of robot planning has reached the point where, in order to make significant new progress we need to continue building complete autonomous robot control systems that are composed of sensing and state estimation modules, have object manipulation capabilities, ane employ planning and plan execution components [ACF⁺98].

Like [Hor96], a number of researchers in the field point out that the next substantial contributions to AI in the near future will come from integrating components into complete robot control systems and testing them on naturalistic tasks: tasks which, unlike playing chess, closely match what we're good at and what we do in everyday life. We believe that shifting toward integration and naturalism will help bring out the structures, assumptions, and constraints that make everyday activity tractable. By redefining the computational problems in such a way that they closer resemble natural tasks we hope to avoid problem specifications that yield computational intractability.

5.3.   Summary and conclusions

Research goals and technology transfer

In this section we will outline technological topics where substantial progress and breakthroughs can be expected in the near-term, medium-term, and long-term future. We will also sketch variants of the application challenge scenarios that we expect to be realizable within the respective time frames.

Short-term research goals

We see the next steps in the development of plan-based control mechanisms for autonomous robots along the following dimensions:

1. the development of plan languages with embedded reasoning capabilities;

2. tighter integration of planning/reasoning and execution;

3. development of temporal plan management capabilities;

4. hybrid plan representation and reasoning systems;

5. object recognition and manipulation tasks (beginning with simulations);

6. general purpose plan libraries for service robots; and

7. development richer interactions of robots with people.

Plan (execution) languages with reasoning capabilities   Explore the extent to which planning should be fully embedded in the execution layer of the robot control architecture. Mobile robots operate in very noisy and dynamic environments. Adding planning capabilities to the executive will improve the exibility/ability to adapt and react. Limiting planning to high-level planning reduces the potential applicability of the planning technology.

Current work on extending the reasoning capabilities of plan execution languages comprises the integration of model-based reasoning and state estimation mechanisms and the integration of learning mechanisms.

Richer control flow and interactions between planning and execution   Explore richer control flow/interactions between modules or layers in the control system of robots. Develop methods to increase the ability to predict the outcomes of combining different behaviors. Traditional architectures can be too rigid and limit the exchange of information between modules. In general, it is not easy to predict how multiple behaviors will operate when combined together. Developing methods to model and predict how behaviors interacts will facilitate reuse of behaviors.

Time management for plans   Temporal constraints matter in dynamic environments: the agent needs to remember how long the cake has been in the oven, for example, or if washes the floor, it needs to consider what other tasks it can do while the floor is drying. Temporal uncertainty may also be important: it may be impossible to tell in advance exactly how long it will take for the cake to bake (or the floor to dry), and the robot will have to reason with such uncertainty.

Planning for human robot interaction   Different types of users require different types and modalities of interactions. Issues to be addressed include: detect the type of user (robot programmer, robot trainer, regular user, casual user, passer by) and act accordingly; develop or learn user models and adapt actions, behaviors, interface to the user; detect user intentions; determine levels of autonomy for different situations/operations; establish adequate levels of trust with users; develop ways of doing goal assessment, plan merging, and plan repair; develop methods of interaction based on natural language; solve symbol grounding problem to enable communication.

Heterogeneous representation and reasoning mechanisms for plan-based control   Many robot planning systems use crude models of the physics of the system they control and the environment they act in. Sometimes the physics is completely represented within the action models used for planning.

In the next years there will be substantial progress in representing models of the robot's capabilities and actions, the dynamics of the environment, and the behavior and preferences of the people in the robot's environment. This models will not be uniform. Rather, very different representational mechanisms will be used depending on the purpose of the model. Some representations will allow for capturing continuous motion, probabilistic behavior and effects, resource and time consumption, while others abstract away from these aspects in order to be more compact.

Coupled with these representations are a variety of reasoning mechanisms that can be applied to perform different reasoning tasks including plan generation, plan diagnosis and revision, plan assessment, plan monitoring, etc.

Object recognition and manipulation tasks   Another research topic will be the plan-based control of perception guided object manipulation. These tasks will raise issues that have not yet been satisfactorily solved. These issues include reasoning about which perceived objects should be manipulated for a given task (often called the symbol grounding or object designation problem. Also competent object manipulation requires robots to thoroughly integrate motion trajectory planning, grasp planning, and task-level planning. We expect to see fundamental progress towards such integrated planning mechanisms for object manipulation.

Challenge scenarios   Besides the advances in the methodological areas listed above we expect demonstrations of simpler variants of the challenge application scenarios. Here we expect to see autonomous robots with richer sets of capabilities. For example, the AAAI robot competition has posed a challenge task that asks autonomous robots to attend a conference. The challenge task includes online mapping of the conference site, finding the registration desk, stand in line to wait for registration, interacting with the registrar, recognizing researchers and doing small talk with them, finding the room of the respective technical session, and giving a talk. While conference participation is a large step forward from current state of the art tour guide robots, main aspects can still be solved with the currently available robots because the task does not require sophisticated manipulation skills.

Medium-term research goals

Topics for medium-term research include the focusing on more naturalistic tasks and everyday activity, on transformational planning techniques, and better physical models for robot planning.

Everyday activity   People can accomplish their daily jobs well even in situations they have not encountered yet for two reasons: they have general routines that work well in standard situations and they are, at the same time, able to recognize non-standard situations and adapt their routines to the specific situations they encounter. Making the appropriate adaptations often requires people to predict how their routines would work in non-standard situations and why they might fail.

The biggest progress in robot planning in the near future will come from integrating components into complete autonomous robots and testing them on naturalistic tasks: tasks which, unlike building blocks world towers, closely match what we re good at and what we do in everyday life.

Model-based robot planning systems   So far most robot planning systems employ fairly naive models of the physical world. Typically, one assumes that the physics can be represented through pre- and (discrete) effects of discrete actions. While many extensions of action representations have been proposed, the extensions are only little steps towards the expressiveness that is needed to predict what will happen when autonomous robots acting in populated environments execute plans that control multiple sensors and effectors. Clearly, the planners have to reason through concurrent continuous processes with interacting effects, such as what will the robot see when it turns it camera while moving down the hallway. Also, in populated environments there are other physical processes carried out by other agents that change the world in ways that are critical for the planned activity.

While there are approaches that provide some of the reasoning capabilities we see a substantial body of work in the area of developing comprehensive planning mechanisms, that comprise all these aspects, handle important aspects of uncertainty, and are effective in the context of autonomous robot control.

Computational models of plan debugging   Existing generative planner can handle only two kinds of plan failures: goals or preconditions of plan steps that have never been achieved and goals or preconditions of plan steps that have been achieved but also been clobbered before they are needed. Contrary to these assumptions plans of autonomous robots acting in natural environments can fail for many more reasons: the robot might overlook an object, the robot might have confused two objects that were similar, the world might have changed, an object might have been placed somewhere else, etc. It is difficult to imagine that we can develop plans that are robust against all these kinds of failures by generating them from first principles. A more plausible approach is to have planners that can produce plans that are robust against typical failures and then predict which other failures might be yielded by the particular situation at hand and then revise the plans to make them robust against those predicted failures.

Thus, in the mid-term time horizon we expect the development and investigation of comprehensive computational models of plan transformation that can handle substantial subsets of plan failures relevant for autonomous service robots acting in realistic environments. These computational models will include mechanisms for detecting such failures, for diagnosing them, and for patching the plans to avoid the failures.

Long-term research goals

One of the big challenges for the longterm research agenda is the development of plan-based autonomous robot controller that perform multiple jobs, in human environments, over long periods of time, and get better as they operate. Examples are autonomous spacecrafts and planetary rovers performing science experiments in deep space, autonomous robots that are companions for people, and robots performing complicated assembly tasks based on visual and verbal instruction.

In the second Dagstuhl Seminar "Plan-based Control of Robotic Agents" we have focused on one particular longterm challenge problem: the realization of an autonomous household robot.

We consider a humanoid robot, such as the Sony SDR-3 or the Honda Asimov, with additional manipulation skills that is to do household chores as an interesting challenge for the field of plan-based control of autonomous robots.

The challenge is to develop a plan-based controller for such a robot that enables the robot to be put in another household, to operate in this household for some months, and do a substantial part of the household chores satisfactorily. To meet this challenge, the robot must acquire models of its environments not only a map of the apartment but also models of the daily rhythm of the household, the time the dishwasher takes to clean the dishes, etc. and it must use these models to better manage its activities. It should also acquire or generate plans to perform its tasks, such as cleaning the living room and it should build up models of these activities that include information such as how long it typically takes to clean up the rooms. The robot is to do several things at a time for example, cleaning while baking a cake. These activities have to be interruptible: the phone might ring in the midst of cooking. Many of the activities require interactions between the robot and the environment and considering actions, such as cleaning the kitchen, to be discrete does not suffice in many situations. These are only some of the aspects that require much more sophistication in controlling such robots than is provided by current robot planning mechanisms. A more detailed description of the challenge task and some of the research issues implied by the household robot challenge are addressed in [BHGP03].

Recommended actions

There are a number of aspects that will help to speed up the progress in plan-based control of autonomous robots.

First, the field will get closer and closer to potential mass markets. The best examples are probably the entertainment robots that are developed by companies such as Sony. In particular, sophisticated humanoid robots such as the Sony SDR-4X will bring in their own need for planning mechanisms as the tasks they are to perform become more complex and the skill level higher. Clearly the closeness to the market will provide sophisticated sensors and effectors at cheaper prices and the opportunity to deploy plan-based control techniques in commercial applications quickly.

A second aspect are the funds available to research groups. Few robotics research groups have the resources to build, from scratch, every component of their robots. Having an autonomous robot laboratory requires currently expensive robots, equipment, and a research group. One consequence out of this situation is that the community needs to be able to share components allowing each research group to concentrate their efforts on pieces of the robotics puzzle. For this to happen several issues need to be addressed: researchers must have a consistent way of describing the problems and solutions; researchers must have away of measuring the effectiveness of various robotics components in different environment and situations; researchers must have away of exchanging components and a way of making those components work on their hardware in their environments; and researchers must have the support of funding agencies to do integration research and to distribute their results.

Third, in order to make substantial progress we need research projects that provide funding for three to five years. This is because the project teams have to build up the infrastructure and a robot control system that can be used for long-term experiments. Developing such research platforms takes additional time.

Fourth, the research community has to shift their bias because currently the investigation of integrated systems is undervalued compared to algorithm design and the study of specific and isolated

problems. The community has to encourage the publication and evaluation of integrated plan-based robot control systems because they are the precondition for the successful development of the field.

There are several other conditions should be established to accelerate the progress in the field. The research community should try, where possible to use common software architectures, at least at the middleware layer, should publish open source code repositories that are maintained by researchers of different groups, and they should cooperate with robot vendors. In particular, the AIBO league in RoboCup has shown that such code repositories for commonly usable robot behaviors are well accepted and used by other research groups.

Conclusions

Robot planning is the oldest and longest standing application of AI planning techniques. In recent years the field has gained more and more importance because of the appearance of different kinds of mobile robots in AI research groups. We have seen exciting demonstrations of the technology including the control of spacecrafts, autonomous helicopters, and tour guide robots. The importance of the field will continue to rise through the development of manipulation and interaction capabilities for these robots.

The defining characteristics of plan-based control is that different aspects of plan-based control, namely plan representation, reasoning, learning, and plan execution cannot be studied in isolation but must be considered in conjunction with the aspects of plan-based control. The advantage of this approach is that we can exploit synergies between the different aspects of plan-based control. Even more important is that plan-based control cannot be researched without contemplating the behavior of the robots that the plans produce. Consequently, research on plan-based robot control must be in the context of controlling autonomous robots.

These constraints for research on robot planning have important consequences for researchers in the field, robot vendors, and funding agencies: the focus must be the study of plan-based control as a resource for autonomous robot control. Therefore, we need autonomous robot laboratories, stronger cooperation between research groups, and the scientific valuation of integrated system research. Second, robot vendors should provide robot platforms with standardized low-level application programmers' interfaces. Third, funding agencies should provide financial support for longer term multi research lab projects. The combination of these activities will provide the basic ground for breakthroughs in the field and the technology transfer. These breakthroughs are needed to solve a number of crucial societal problems including the assistance of elderly people and the exploration of space and other planets, and the search for extra-terrestial life.

To sum up, in order to accelerate the progress in the area of robot planning and plan-based robot control:

Robot vendors should provide robot platforms for research, make parts of their code available to research groups. In return, the code developed in the research groups should be published as repositories that can then also be used by the robot vendors.

Funding agencies should support the building of research laboratories for autonomous robots, provide funding for longer term research projects (with a startup phase for building the infrastructure), and support multi university research initiatives with common code repositories.

Researchers should seek the collaboration with other research groups in order to tackle the challenge problems stated in this Roadmap. They should also encourage work on integrated robot control systems where plan-based robot control is employed and provide publication platforms for this kind of research. Finally, the research community should work on better experimental methods and testbeds that enable the researchers to better evaluate the claims of their work.

Bibliography

[ACF⁺98]   R. Alami, R. Chatila, S. Fleury, M. Ghallab, and F. Ingrand. An architecture for autonomy. *International Journal of Robotics Research*, 17(4), 1998.

[AF94]   J. Allen and G. Ferguson. Actions and events in interval temporal logic. *Journal of Logic and Computation*, 4(5):531–579, October 1994.

[AFH⁺98]   R. Alami, S. Fleury, M. Herb, F. Ingrand, and F. Robert. Multi robot cooperation in the Martha project. *IEEE Robotics and Automation Magazine*, 5(1), 1998.

[AIQ98]   R. Alami, F. Ingrand, and S. Qutub. A scheme for coordinating multi-robots planning activities and plans execution. In H. Prade, editor, *Proceedings of the 13th European Conference on Artificial Intelligence (ECAI-98)*, pages 617–621, Chichester, 1998. John Wiley & Sons.

[AKPT90]   J. Allen, H. Kautz, R. Pelavin, and J. Tenenberg, editors. *Reasoning about Plans*. Morgan Kaufmann, San Mateo, Cal, 1990.

[AR01]   D. Andre and S. Russell. Programmable reinforcement learning agents. In *Proceedings of the 13th Conference on Neural Information Processing Systems*, pages 1019–1025, Cambridge, MA, 2001. MIT Press.

[Ark98]   R. Arkin. *Behavior based Robotics*. MIT Press, Cambridge, Ma, 1998.

[BAB⁺01]   M. Beetz, T. Arbuckle, M. Bennewitz, W. Burgard, A. Cremers, D. Fox, H. Grosskreutz, D. Hähnel, and D. Schulz. Integrated plan-based control of autonomous service robots in human environments. *IEEE Intelligent Systems*, 16(5):56–65, 2001.

[BB00]   M. Beetz and T. Belker. Environment and task adaptation for robotic agents. In W. Horn, editor, *Procs. of the 14th European Conference on Artificial Intelligence (ECAI-2000)*, 2000.

[BBC02]   T. Belker, M. Beetz, and A. Cremers. Learning action models for the improved execution of navigation plans. *Robotics and Autonomous Systems*, 38(3-4):137–148, March 2002.

[BCF⁺00]   W. Burgard, A.B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. Experiences with an interactive museum tour-guide robot. *Artificial Intelligence*, 114(1-2), 2000.

[BD95]   Steven J. Bradtke and Michael O. Duff. Reinforcement learning methods for continuous-time Markov decision problems. In G. Tesauro, D. Touretzky, and T. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7, pages 393–400. The MIT Press, 1995.

[Bee00]   M. Beetz. *Concurrent Reactive Plans: Anticipating and Forestalling Execution Failures*, volume LNAI 1772 of *Lecture Notes in Artificial Intelligence*. Springer Publishers, 2000.

[Bee02]   M. Beetz. *Plan-based Control of Robotic Agents*, volume LNAI 2554 of *Lecture Notes in Artificial Intelligence*. Springer Publishers, 2002.

[BFG⁺97]   P. Bonasso, J. Firby, E. Gat, D. Kortenkamp, D. Miller, and M. Slack. Experiences with an architecture for intelligent, reactive agents. *Journal of Experimental and Theoretical Artificial Intelligence*, 9(1), 1997.

[BG00]   M. Beetz and H. Grosskreutz. Probabilistic hybrid action models for predicting concurrent percept-driven robot behavior. In *Proceedings of the Sixth International Conference on AI Planning Systems*, Toulouse, F, 2000. AAAI Press.

[BHGP02]   M. Beetz, J. Hertzberg, M. Ghallab, and M. Pollack. *Advances in Plan-based Control of Robotic Agents*, volume LNAI 2554 of *Lecture Notes in Artificial Intelligence*. Springer Publishers, 2002.

[BHGP03]   Michael Beetz, Joachim Hertzberg, Malik Ghallab, and Martha Pollack. The household robot challenge for plan-based robot controllers. `http://www9.in.tum.de/~beetz/challenge.pdf` (unpublished), 2003.

[BIP88]   M. Bratman, D. Israel, and M. Pollack. Plan and resource-bounded practical reasoning. *Computational Intelligence*, 4:349–355, 1988.

[Bly99]   J. Blythe. Decision-theoretic planning. *AI Magazine*, 20, 1999.

[Bro91]   R. Brooks. Intelligence without reason. In *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, pages 569–595, Sidney, Australia, 1991.

[CS00]   S. Coradeschi and A. Saffiotti. Anchoring symbols to sensor data: preliminary report. In *Proc. of the 17th AAAI Conf.*, pages 129–135, Menlo Park, CA, 2000. AAAI Press.

[DGK⁺00]   P. Doherty, G. Granlund, K. Kuchcinski, E. Sandewall, K. Nordberg, E. Skarman, and J. Wiklund. The witas unmanned aerial vehicle project. In *Proceedings of the 14th European Conference on Artificial Intelligence*, pages 747–755, 2000.

[DI99]   O. Despouys and F. Ingrand. Propice-plan: Toward a unified framework for planning and execution. In *Recent Advances in AI Planning. Proceedings of the 1999 European Conference on Planning*. Springer Publishers, 1999.

[DW91]   T. Dean and M. Wellmann. *Planning and Control*. Morgan Kaufmann Publishers, San Mateo, CA, 1991.

[Fir87]   J. Firby. An investigation into reactive planning in complex domains. In *Proceedings of the Sixth National Conference on Artificial Intelligence*, pages 202–206, Seattle, WA, 1987.

[Fir89]   J. Firby. *Adaptive Execution in Complex Dynamic Worlds*. Technical report 672, Yale University, Department of Computer Science, January 1989.

[FL03]   M. Fox and D. Long. Pddl2.1: An extension of pddl for expressing temporal planning domains. *Journal of Artificial Intelligence Research*, 2003. to appear.

[FPS⁺96]   J. Firby, P. Prokopowicz, M. Swain, R. Kahn, and F. Franklin. Programming CHIP for
            the IJCAI-95 robot competition. *AI Magazine*, 1996.

[Gat96]   E. Gat. Esl: A language for supporting robust plan execution in embedded autonomous
            agents. In *AAAI Fall Symposium: Issues in Plan Execution*, Cambridge, MA, 1996.

[GNT04]   M. Ghallab, D. Nau, and P. Traverso. *Automated Planning: Theory and Practice*.
            Morgan Kaufmann Publishers, 2004. to appear.

[Hor96]   I. Horswill. Integrated systems and naturalistic tasks. In: Strategic Directions in
            Computing Research AI Working Group, 1996.

[HV98]   K. Haigh and M. Veloso. Learning situation-dependent costs: Improving planning
            from probabilistic robot execution. In *Second Int. Conf. on Autonomous Agents 98*,
            1998.

[ICAR96]   F. Ingrand, R. Chatila, R. Alami, and F. Robert. PRS: A high level supervision and
            control language for autonomous mobile robots. In *Proc. of the IEEE Int. Conf. on
            Robotics and Automation*, pages 43–49, Minneapolis, 1996.

[KCK96]   L. Kaelbling, A. Cassandra, and J. Kurien. Acting under uncertainty: Discrete
            Bayesian models for mobile-robot navigation. In *Proceedings of the IEEE/RSJ In-
            ternational Conference on Intelligent Robots and Systems*, 1996.

[KLM96]   L. Kaelbling, M. Littman, and A. Moore. Reinforcement learning: A survey. *Journal
            of Artificial Intelligence Research*, 4, 1996.

[LA89]   D. Lyons and M. Arbib. A formal model of computation for sensory-based robotics.
            IEEE Journal of Robotics and Automation, 5(3):280–293, 1989.

[LRL⁺97]   H. Levesque, R. Reiter, Y. Lespérance, F. Lin, and R. Scherl. Golog: A logic pro-
            gramming language for dynamic domains. *Journal of Logic Programming*, 31:59–84,
            1997.

[Mah96]   S. Mahadevan. Machine learning for robots: A comparison of different paradigms. In
            *Workshop on Towards Real Autonomy , IEEE/RSJ International Conference on Intel-
            ligent Robots and Systems (IROS-96)*, Osaka, Japan, 1996.

[McD78]   D. McDermott. Planning and acting. *Cognitive Science*, 2(2):78–109, 1978.

[McD85]   D. McDermott. Reasoning about plans. In J. R. Hobbs and R. C. Moore, editors,
            Formal Theories of the Commonsense World, pages 269–317. Ablex, Norwood, NJ,
            1985.

[McD90]   D. McDermott. Planning reactive behavior: A progress report. In K. Sycara, edi-
            tor, *Innovative Approaches to Planning, Scheduling and Control*, pages 450–458, San
            Mateo, CA, 1990. Kaufmann.

[McD91]   D. McDermott. A Reactive Plan Language. Research Report YALEU/DCS/RR-864,
            Yale University, 1991.

[McD92]   D. McDermott. Robot planning. *AI Magazine*, 13(2):55–79, 1992.

[McD98]   McDermott. Pddl - the planning domain definition language. Technical Report CVC TR-98-003/DCS, Yale Center for Computational Vision and Control, 1998.

[McD03]   D. McDermott. Reasoning about autonomous processes in an estimated-regression planner. In *Proc. Int'l. Conf. on Automated Planning and Scheduling*, 2003.

[MNPW98a]   N. Muscettola, P. Nayak, B. Pell, and B. Williams. Remote Agent: to go boldly where no AI system has gone before. *Artificial Intelligence*, 103(1–2):5–47, 1998.

[MNPW98b]   Nicola Muscettola, P. Pandurang Nayak, Barney Pell, and Brian C. Williams. Remote agent: To boldly go where no AI system has gone before. *Artificial Intelligence*, 103(1-2):5–47, 1998.

[New90]   A. Newell. *Unified Theories of Cognition*. Harvard University Press, Cambridge, Massachusetts, 1990.

[Nil84]   Nils J. Nilsson. Shakey the robot. Technical Note 323, SRI International, Menlo Park, California, 1984.

[NW97]   P. Nayak and B. Williams. Fast context switching in real-time propositional reasoning. In *Proceedings of the 14th National Conference on Artificial Intelligence and 9th Innovative Applications of Artificial Intelligence Conference (AAAI-97/IAAI-97)*, pages 50–56, Menlo Park, 1997. AAAI Press.

[PA89]   K. Passino and P. Antsaklis. A system and control-theoretic perspective on artificial intelligence planning systems. *Applied Artificial Intelligence*, 3:1–32, 1989.

[PH99]   M. Pollack and J. Horty. There's more to life than making plans: Plan management in dynamic, multi-agent environments. *AI Magazine*, 20(4):71–84, 1999.

[RK91]   E. Rich and K. Knight. *Artificial Intelligence*. McGraw Hill, New York, 1991.

[RM01]   K. Rohanimanesh and S. Mahadevan. Decision-theoretic planning with concurrent temporally extended actions. In *17th Conference on Uncertainty in Artificial Intelligence (UAI)*, University of Washington, Seattle, WA, 2001.

[SB98]   R. Sutton and A. Barto. *Reinforcement Learning: an Introduction*. MIT Press, 1998.

[SGG+03]   R. Simmons, D. Goldberg, A. Goode, M. Montemerlo, N. Roy, B. Sellner, C. Urmson, A. Schultz, M. Abramson, W. Adams, A. Atrash, M. Bugajska, M. Coblenz, M. MacMahon, D. Perzanowski, I. Horswill, R. Zubek, D. Kortenkamp, B. Wolfe, T. Milam, and B. Maxwell. Grace: An autonomous robot for the aaai robot challenge. *AI Magazine*, 23(2), 2003.

[SGH+97a]   R. Simmons, R. Goodwin, K. Haigh, S. Koenig, and J. O'Sullivan. A modular architecture for office delivery robots. In *Proceedings of the First International Conference on Autonomous Agents*, pages 245–252, 1997.

[SGH⁺97b]   R. Simmons, R. Goodwin, K. Haigh, S. Koenig, J. O'Sullivan, and M. Veloso. Xavier: Experience with a layered robot architecture. *ACM magazine* Intelligence, 1997.

[SPS98]   R. Sutton, D. Precup, and S. Singh. Between mdps and semi-mdps: Learning, planning, and representing knowledge at multiple temporal scales. *Journal of AI Research*, 1998.

[Sus77]   G. Sussman. *A Computer Model of Skill Acquisition*, volume 1 of *Aritficial Intelligence Series*. American Elsevier, New York, NY, 1977.

[TBB⁺99]   S. Thrun, M. Bennewitz, W. Burgard, A.B. Cremers, F. Dellaert, D. Fox, D. Haehnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz. Minerva: A second generation mobile tour-guide robot. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'99)*, 1999.

[TBB⁺00]   S. Thrun, M. Beetz, M. Bennewitz, A.B. Cremers, F. Dellaert, D. Fox, D. Hähnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz. Probabilistic algorithms and the interactive museum tour-guide robot Minerva. *International Journal of Robotics Research*, 2000.

[TM93]   S. Thrun and T. Mitchell. Lifelong robot learning. Technical Report IAI-TR-93-7, University of Bonn, Department of Computer Science, 1993.

[WICE03]   B. Williams, M. Ingham, S. Chung, and P. Elliott. Model-based programming of intelligent embedded systems and robotic space explorers. *Proceedings of the IEEE: Special Issue on Modeling and Design of Embedded Software*, 9(1):212–237, 2003.

[WN96]   B. Williams and P. Nayak. A model-based approach to reactive self-configuring systems. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence and the Eighth Innovative Applications of Artificial Intelligence Conference*, pages 971–978, Menlo Park, 1996. AAAI Press / MIT Press.

[WN97]   B. Williams and P. Nayak. A reactive planner for a model-based executive. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI-97)*, pages 1178–1185, San Francisco, 1997. Morgan Kaufmann Publishers.

6. Aerospace Applications

6.1. Introduction

The purpose of this Aerospace Applications Roadmap is to identify the AI P&S research and technology transfer efforts needed in the coming three to ten years specifically to meet the needs of the aerospace domain.

The approach taken in this roadmap is to establish the current state-of-the-art and the desired future state of AI P&S technology to meet aerospace needs. The roadmap is then a feasible path from the current state to the desired future state. Feasibility is based on the contributors' judgement of how quickly AI P&S research and development could deliver technology mature enough to apply, assuming that current levels of R&D funding are maintained.

This chapter consists of three sections, a bibliography and an appendix. After this introductory section, aerospace domain and its sub-domains are defined and described. We then survey the planning and scheduling state-of-the-art in the aerospace sub-domains. We also identify the research needed over the coming three to ten years and propose a series of steps towards technology transfer. Section 6.3 draws conclusions and makes recommendations.

Appendix 6.4 is an object-oriented analysis of the entities, relationships and invariants in the aerospace domain, from which planning and scheduling applications can be systematically identified.

Intended readers of this PLANET II Aerospace Applications Roadmap are:

- Members of the AA TCU

- Other members of PLANET

- The European Commission, including reviewers

- The worldwide AI P&S R&D community interested in aerospace applications

- The worldwide aerospace community interested in solving planning and scheduling problems

- The suppliers of AI P&S-based products and services interested in aerospace applications

6.1.1. The Aerospace Applications TCU

The TCUs form the nucleus of PLANET's activities. The primary goal of each TCU is to prepare a roadmap for its specific area that:

- Defines and summarizes the state-of-the-art;

- Identifies focal points towards which research efforts should be directed;

- States what steps are needed for effective technology transfer.

The Aerospace Applications TCU (AA TCU) brings together PLANET members and other organizations interested in the application of Artificial Intelligence Planning and Scheduling (AI P&S) techniques to the solution of problems in the aerospace domain. The current AA TCU nodes and members are:

Node	Contact person	Remarks
Atos Origin Nederland (NL)	Tim Grant	Co-chairman
ISTC-CNR (IT)	Amedeo Cesta	Co-chairman
University of Ulm (GE)	Bernd Schattenberg	
MAP-CNRS (FR)	Camilla Schwind	
CERT (FR)	Claude Barrouil, Patrick Fabiani, Jean-Francois Gabard	
University Carlos 3 Madrid (ES)	Daniel Borrajo	
CNES (FR)	Denis Carbonne, Linda Tomasini	
UNIGE (IT)	Enrico Guinchiglia	
Universidad de Alcalá (ES)	Maria-Dolores R-Moreno	Contributor
ESA-ESTEC	Eric Bornschlegl	
ESA-ESOC	Alessandro Donati	
ONERA (FR)	Gerard Verfaillie	
NLR (NL)	Henk Hesselink, Ron Seljeé	Contributors
University of Huddersfield (UK)	Lee McCluskey	
Space Applications (BE)	Richard Aked, Leif Steinicke	
IRST-ITC (IT)	Paolo Traverso	
Thales Group (FR)	Patrick Taillibert	
University of Salford (UK)	Ruth Aylett	Contributor
BAE Systems (UK)	Andrew Burgess	
Anite Systems (GE)	Marc Niezette	

The AA TCU's activities began with a Kick-Off Meeting in Manchester, UK, on 18 June 2000. The surveys of the state-of-the-art and of needs took place over the summer of 2000, with the results being presented in the AA TCU workshop in Rome, Italy, on 18/19 September 2000.

Leading members of the AA TCU took part in the 17-19 October 2001 ESA/ESTEC workshop on On-Board Autonomy, resulting in ESA's invitation to PLANET to become ESA's center of excellence on AI P&S.

PLANET sponsored the 6 July 2002 workshop on AI Planning and Scheduling for Autonomy in Space Applications, held in Manchester, UK. At this workshop the AA TCU presented an overview of progress on the AA Roadmap, covering the state-of-the-art survey of AI P&S techniques relevant to aerospace and the object-oriented analysis of the aerospace domain (see Appendix 6.4 and PLANET workshop reports (cf. Section A, respectively).

Over the course of 2002 and 2003 the AA TCU co-chairmen gave advice on the organization of the fourth International Workshop on Planning and Scheduling for Space Applications, to be hosted by ESOC in Darmstadt, Germany, in Spring 2004.

In 2003 the final steps were taken with Prof. Aylett's contribution on a feasible path of R&D to meet aerospace needs, with probable timescales. This resulted in the presentation of a paper outlining the AA Roadmap [GCRM$^+$03] at the 2003 conference on Space Mission Challenges for IT, organized by the Jet Propulsion Laboratory and held in Pasadena, California, USA, on 14-16 July 2003.

This document is the roadmap produced by the Aerospace Applications TCU for inclusion in the PLANET deliverable D30 (Roadmaps update).

6.1.2.   AI P&S and aerospace applications

The AA TCU's approach is partly formal and partly market-driven. One of the authors with extensive domain experience (Tim Grant) performed a formal object-oriented analysis to identify planning and scheduling applications in the aerospace domain. Selected representatives of aerospace organizations (e.g. the potential AI P&S user community) were invited to review the results of the domain analysis, to confirm the applications, to provide the domain-specific names and characteristics of the confirmed applications, and to identify their existing applications and future needs for planning and scheduling technologies. The needs were matched with a formal survey of the current and foreseen developments in AI P&S technology to derive the Aerospace Applications Roadmap.

6.1.2.1.   Aerospace domain

This section defines the aerospace domain for the purposes of this roadmap, and then surveys the state-of-the-art and identifies trends in each aerospace sector.

What is aerospace?   In this roadmap, the aerospace domain is interpreted in the broadest sense, covering all aspects of civil and military aviation and space. Aerospace concerns controlled travel above the Earth's surface. Central to the analysis are the vehicles to be found in the aerospace domain, such as aircraft and spacecraft. Objects that move in ballistic, largely uncontrolled trajectories, like bullets and shells, are outside the scope of this roadmap.

Where the means of travel depends on the support of the atmosphere then it is regarded as aviation. For example, balloons, helicopters, and fixed-wing aircraft are all dependent on the support of the Earth's atmosphere.

Travel that is independent of the support of the atmosphere is regarded as space travel. Space launchers and satellites are (in principle) independent of the support of the atmosphere.

In terms of vehicle design, aviation and space travel are almost independent of one another. Those space vehicles that pass through an atmosphere must be appropriately designed to withstand aerodynamic pressures. Some space vehicles exploit aerodynamic effects for braking purposes or to alter their trajectory. The most advanced present-day example is the US Space Shuttle. Future Reusable Launch Vehicles (RLVs) are foreseen that will have to be licensed according to both air and space law. Plans are being made to explore the surface of Mars using unmanned air vehicles. Nevertheless,

given the ten-year timescale of this roadmap, aviation and space travel can be regarded as separate sub-domains.

Many of the technologies and organizations found in the aviation sub-domain are also to be found in the space sub-domain. Aircraft and spacecraft have in common that low weight and high safety are dominant concerns.

We identify a number of sectors in the aerospace domain, as follows:

Aerospace manufacturing The aerospace manufacturing sector includes the aircraft and spacecraft manufacturers and their suppliers of sub-systems, components and associated services. The sector also includes the authorities that regulate aerospace manufacturing activities, as well as national and international associations concerned with aerospace manufacturing. Example European organizations in the aerospace manufacturing sector include Airbus, EADS, Astrium, Alenia, Thales, Telespazio, Italian Space Agency (ASI), Netherlands Aerospace Agency (NIVR), Netherlands Industrial Space Organisation (NISO), Netherlands Aerospace Group (NAG), Vereniging Gasturbines (VGT).

Aircraft operations The aircraft operations sector includes the civil and military organizations that operate aircraft, i.e. airlines and air forces. It also includes the aircrew (pilots, flight engineers, load masters, cabin personnel) and ground crew (maintainers, technical services). The sector also includes the authorities that regulate aircraft operations, as well as national and international associations concerned with aircraft operations. Example European organizations in the civil aircraft operations sector include KLM, Alitalia, British Airways, Lufthansa, the UK's Civil Aviation Authority, GAPAN, ICAO, and IATA. Military aircraft operators include national air forces, air arms of navies and armies, and NATO. Research institutes concerned with aircraft operations include ONERA, the German Aerospace Research Institute (DLR) and the Dutch National Aerospace Laboratory (NLR).

Spacecraft operations The spacecraft operations sector, by analogy with the aircraft operations sector, includes the civil and military organizations that launch and operate spacecraft. It also includes the astronauts, cosmonauts, mission planners, flight dynamics experts, spacecraft operators and engineers. The sector also includes the authorities that regulate spacecraft operations, as well as national and international associations concerned with spacecraft operations. Example organizations in the civil spacecraft operations sector include the European Space Operations Center (ESOC), EUMETSAT, EUTELSAT, INMARSAT, the German Space Operations Center (GSOC), New Skies Satellites, and SES-Astra. Military spacecraft operators include the UK's Skynet organization and NATO. Research institutes in the spacecraft operations sector include the Center for Earth Observation at the EU's Joint Research Center (JRC CEO), the Dutch Space Research Organization (SRON), and the Center National d'Etudes Spatiales (CNES). National and international agencies concerned with spacecraft operations include the European Space Agency (ESA), the Italian Space Agency (ASI), and the Netherlands Agency for Aerospace Programs (NIVR).

Airport operations The airport operations sector includes the airports, ground handling services (aircraft towing, refueling, ground power, baggage handling), meteorological services, passenger check-in and handling, catering, duty-free shops, customs, airport police and security

organizations. The sector also includes the authorities that regulate airport operations, as well as national and international associations concerned with airport operations. Example organizations in the airport operations sector include Amsterdam Airport Schiphol, British Airports Authority, and Aeroporto di Roma.

Air Traffic Management The Air Traffic Management (ATM) sector includes the organizations that manage airborne civil and military air traffic, aircraft taxiing on the ground, and take-offs and landings. The sector also includes the authorities that regulate air traffic, as well as national and international associations concerned with air traffic. Example organizations in the ATM sector include the UK's National Air Traffic Service (NATS), the Dutch Lucht Verkeersleiding Nederland (LVLN) and Rijksluchtvaart Dienst (RLD) organizations, and EuroControl.

Maintenance, Repair & Overhaul The maintenance, repair and overhaul (MRO) sector includes the organizations involved in maintaining, repairing and overhauling aircraft, spacecraft, their engines, and other sub-systems and components. The sector also includes the authorities that regulate the maintenance, repair and overhaul of aircraft and spacecraft, as well as national and international associations concerned with aerospace MRO activities. Example organizations in the MRO sector include the technical departments of the airlines (e.g. KLM's Technische Dienst), maintenance departments of manufacturers (e.g. Rolls Royce, General Electric Services), independent repair and testing suppliers, and advisers on reliability, availability and maintainability (e.g. Sergem).

Factors affecting aerospace as a whole The whole aerospace sector is currently under severe economic pressure. Technological developments, particularly the growing importance of ICT, also influence aerospace strongly.

The overriding economic theme is the need to save costs. For civil aviation, this was already becoming apparent before the WTC attacks on 11th September 2001. In military aerospace, cost-saving has a longer history, stemming from the 1989 fall of the Berlin Wall, the end of the Cold War, and the drive to reap the "peace dividend".

The commercial sector normally follows the seven-year business cycle. Traditionally, the institutional (governmental) sector cushions the commercial cycle by moving anti-phase. However, this anti-cyclic behavior has been absent in the institutional sector over the past two to three years. There is a growing desire amongst European governments to reduce national debt and deficit budgeting. This leads to cost-cutting in the institutional sector as well.

ICT has continuing influence in every corner of the aerospace domain. Computer technologies are built into aircraft and spacecraft in critical applications such as fly-by-wire control, health-management systems and environmental controls. Less-critical applications include flight entertainment systems, satellite-based email and Internet access for passengers. Manufacturing processes are undergoing a revolutionary impact from web-enabled Computer-Aided Design/Computer-Aided Manufacturing (CADCAM), Production Data Management (PDM), and Supply Chain Management (SCM) systems. ICT enables suppliers throughout the world to collaborate in design and development. The Boeing 777 is probably the first major aircraft type to be designed predominantly using electronic means, speeding up design and development. Boeing says that the forthcoming 7E7 will be the "most e-enabled airplane in history".

In the military Joint Strike Fighter (JSF) project, the onward march of ICT extends into the MRO sector. The on-board Prognostics and Health Management (PHM) system will radio information on system performance and faults to the global, ground-based Autonomic Logistics (AutoLog) system. This will enable a maintainer with the right skills, tools and spare parts to meet the aircraft on landing.

Airport operations are characterized by the great number and diversity of stakeholders that interact with one another. These range from the airport organization, through the airlines that use the airport, to the ground-handling agents, catering companies, and duty-free shops. ICT provides the infrastructure that is only just starting to be used in supporting interaction and collaboration between the stakeholders.

Air Traffic Management (ATM) is absolutely dependent on ICT, from flow control to real-time conflict prediction and resolution. Eurocontrol and the national ATM agencies are currently engaged in an evolving, multi-year programme to apply a Collaborative Decision-Making (CDM) approach to increase the capacity of the European airspace. This will require large ICT investments to link pilots, airline operations control centers, and air traffic controllers.

Non-ICT technologies have an equally major influence on aerospace. Major drivers are technologies such as advanced materials, propulsion, and manufacturing processes. Combinations of ICT and non-ICT technologies are under active investigation, including smart sensors and intelligent structures. Spacecraft missions in which multiple satellites must fly in formation to nanometer accuracy are prominent in mission analyses. ESA's Aurora programme leading to the human exploration of Mars envisages swarms of autonomous mobile robots operating on the Martian surface and in the Martian atmosphere.

The complexity of aerospace systems is so great that their development can take nine to fifteen years. Many of the more futuristic aerospace developments mentioned above will take place outside the ten-year horizon of this roadmap. Nevertheless, the technological foundation will have to be laid in the coming ten years.

AI P&S technologies have a key role to play in aerospace. PLANET nodes and members have already participated in projects that have demonstrated major cost savings. Examples include the use of AI P&S techniques in the 1991 Gulf War to reduce the cost of the logistics build-up [CW94, ICGH96]. Replacement of conventional scheduling technology by AI P&S results in cost reductions of 20% to 30%, e.g. in reducing overtime, subcontracting, and aircraft assembly time [Dra03]. Moreover, the time needed to generate a schedule can be up to two orders of magnitude faster than conventional techniques. Initiatives such as the International Planning Competition (IPC) are leading to still faster AI P&S technologies. It is vital that PLANET spotlights documented case studies of these benefits of AI P&S, not only in the aerospace domain but also in other application domains.

The following sections in this chapter consider in more detail the trends affecting each aerospace sector. These trends have been extracted from sources such as the Aviation Week & Space Technology (AWST) 2003 Aerospace Source Book and Eurocontrol's documentation on Collaborative Decision-Making. It should be noted that the 2003 Aerospace Source Book was published in mid-January 2003, just prior to the Iraq war and the height of the SARS epidemic. Eurocontrol's CDM documentation typically dates from 1998 onwards.

Aerospace manufacturing　　The 2003 Aerospace Source Book emphasizes trends in the aerospace manufacturing industry. These are divided into articles on military fighter/attack, bomber, airlift, and

trainer aircraft, on unmanned air vehicles (UAVs), on civil transports, freighters, helicopters, and business aviation, and on MRO.

For fighter/attack aircraft, the 2003 Aerospace Source Book observes that the US market has grown, while the European market remains stagnant. Europe is in danger of losing its position and much of its industrial base in fighter/attack aircraft. The US fighter industry's prospects have been greatly enhanced by the "war against terror". Increasing US defense spending has been accompanied by a major divergence in foreign policy between the US and Europe. The US military services are using, improving, upgrading, and supporting their equipment. They are inventing new ways to channel data (e.g. data links), new doctrine to improve effectiveness (e.g. network-centric warfare (NCW) and effects-based operations (EBO)), and stockpiling spares. US fighter export success, particularly in the JSF programme, will have a halo effect for other systems. Client air forces are likely to stick with US training, US doctrine, US sensor data, and other US weapon systems. They are more likely to become interoperable with US command, control and communications (C^3) systems.

The only large bomber fleet is to be found in the US military services. All are legacy aircraft (B-1, B-2, and B-52), some up to 40 years old. Four possible ways of replacing the older aircraft are under consideration: a hypersonic platform with exo-atmospheric capabilities, a platform combining long-range supersonic capability with stealth, a low-risk derivative of the B-2, or a strategic version of the F-22 fighter with larger wings and upgraded engines. The case for a new bomber is being undercut by events and by technology. Recent wars show that substantial bombing capacity can be provided by multi-role strike fighters based on aircraft carriers or in nearby countries. The increasing use of precision-guided missiles (PGM) – 5% in the 1991 Gulf War, 35% in Yugoslavia in 1999 and over 60% in Afghanistan in 2001-2 – makes dedicated bomber platforms less necessary.

The military airlift market has been characterized for many years by "under-sizing and over-discussion". Lockheed Martin's C-130 Hercules is the worldwide workhorse, and together with the larger Boeing C-17 has 90% of the market by dollar value. Larger military air-lifters are to be found only in the US and Russian air forces. The only non-US airlift programmes are flying smaller aircraft, such as Alenia's C-27J and the EADS/CASA C-295/CN-235. The Airbus Military Company's A400M is supposed to provide the European solution to the requirement for an air-lifter between the C-130 and C-17 in size, together with an incentive for European governments to invest in a robust airlift capability. Unfortunately, the A400M programme has stagnated. The international neglect of military airlift seems entrenched in countries outside Europe too, as shown by Australia's recent cancellation of an air-lifter purchase. A recent breakthrough was the UK's decision in September 2000 to lease 4 C-17s until the A400M becomes operational. A substantial market is emerging in upgrading legacy air-lifters, such as the C-130 and C-5 fleets.

The world military aircraft trainer market remains stagnant. This is due to a unique combination of circumstances. First, there is an oversupply of trainers, which is the easiest type of military aircraft to design and build. There is a proliferation of indigenous trainer designs with small production runs, high unit costs, few export orders, and few industrial skills learned. Second, demand for trainers is low because defense forces have been cut worldwide. Pilot training requirements have changed, with information and systems management now taking precedence over flying skills. Such training can be best honed in ground-based simulators. Third, there is an increasing sharing of trainer fleets and outsourcing to the private sector. A key example is the NATO Flight Training in Canada (NFTC) programme, run by Bombadier on a "power-by-the-hour" basis. Customers include UK, Singapore,

Canada, Italy, Hungary and Denmark. Even the security-conscious Isreali forces are considering taking part.

The civil transport market is tough, with intense competition. The current economic environment is the most challenging that manufacturers have faced in several decades. Customer demand is down, airline traffic and profitability are down, the competitive landscape has been altered by the emergence of low-cost airlines, and there is a savage market-share war between the two major manufacturers (Boeing and Airbus). The airlines' financial performance gives the most cause for alarm. US airlines accumulated US$10 billion operating losses in 2001, despite massive US government subsidies. They made another US$8 billion loss in 2002. Jetliner orders have fallen from 1081 in 2000 to 546 a year later. Nevertheless, jetliner production remains at a historically high level. Manufacturers seem to think that 2003 is the low point with 575 jetliners being produced, but events are likely to prove them wrong. Previous lows were 379 jetliners in 1995 and just 244 in 1984. Jetliner pricing is extremely soft, which is unusual for an upturn. Then there is the problem of parked aircraft. The "Desert Air Force" consisted of 1813 transports (12% of the total fleet) in October 2002. Many of these unused aircraft are unlikely to re-enter service. Only 296 are in-production models, and another 202 may be economically viable. The result is a flood of planes, and this overcapacity will depress the market for some time. Recovery in 2004 is likely to prove optimistic.

Regional jets (RJs) in the 30- to 50-seat category are currently enjoying explosive popularity. This is because airlines are looking to save costs by outsourcing routes to subsidiaries that use RJs. American Airlines is developing rolling hubs that rely on frequency and a high level of aircraft utilization. This favors RJs. However, airlines are having difficulty in finding financial institutions that will lend them the money to buy RJs. Moreover, existing agreements with the aircrew unions limit the number and types of RJs that can be flown by the majors' subsidiaries, particularly in the larger 70- to 100-seat category.

In the civil air freight market, two factors have remained constant since 1970: the global market for air freight grew annually by twice the growth in global gross domestic product (GDP), and an annual increase in the freighter fleet in response to this traffic growth. This changed in 2001, with the US recession leading to a 5% decline in the first half of the year. This worsened after 11th September to give a 10% decline on many major routes. There was an improvement in 2002 but with regional variations: North America down 6%, Europe down 1.5%, and Asia-Pacific up 11%. The situation at the end of 2002 was comparable to 1999, giving three years of lost growth, with a substantial oversupply of freighter aircraft. However, long-term prospects are bright. The composition of the fleet has changed, with converted first-generation jetliners retiring. If global GDP reaches 3% per year, then airfreight should grow 6% annually. Asia and China in particular may reach 9% per year. Over 20 years at this rate traffic will triple. Given that about half of all freight travels in the bellies of passenger aircraft, the fleet should double. The average freighter is increasing in utilization and size, as more wide-bodies are converted. Conversions will dominate, aided by the lower value of passenger aircraft. Other issues are enhanced cargo security measures, the availability of a new group of turboprop freighters, and the possible advent of regional freighters now on the drawing board.

Aircraft operations   The severe financial problems faced by the major airlines has already been mentioned above in the context of the fall-off in civil aircraft manufacturing. The airlines have

trimmed their timetables to cut out flights with low yields, reduced seat pitch in economy class, and renewed their merger activity. The net result is that load factors have grown from around 70% to above 80%. Passengers are complaining about their restricted mobility from the reduction in seat-pitch increasing the danger of Deep Vascular Thrombosis on long flights. European airlines are making another round of attempting to merge to form larger groups. Lufthansa has effectively taken over the failing Swiss Airlines and is in discussion with Austrian Airlines. The latest development is KLM's announcement that it will be taken over by Air France, after breakdowns in repeated merger negotiations with British Airways and the withdrawal from a merger with Alitalia.

The low-cost airlines – "bus travel with wings" – are taking passengers away from the major airlines and slowing their recovery. Customers are increasingly sensitive to cost, aided by Internet booking and web sites offering cheap air fares. The cost-consciousness of business travelers threatens the biggest source of profit for the majors: full-fare tickets. 20% of the US market is now in the hands of low-cost airlines. Their share in Europe is 6% and growing fast.

Air forces worldwide are in a similar position to the major airlines. Procurement funding remains flat while the cost of aerospace technology continues to climb. Air forces are responding by outsourcing activities that do not contribute directly to the application of military force. Examples include aircraft maintenance and training. Contractors have always been responsible for 4th line maintenance. With the fall of the Berlin Wall, they took over 3rd line (or depot) maintenance. Now the trend is for contractors to move in on 2nd line maintenance. Even contracting out 1st line (or squadron-level or flight-line) maintenance is under discussion. Outsourcing of training, increasing use of simulation, and sharing of trainer fleets, such as the NATO Flight Training in Canada (NFTC) programme, have already been mentioned.

There are a number of new developments in military aircraft operations. The most important of these are joint and combined operations, network-centric warfare, and unmanned air vehicles. Traditionally, air forces have conducted their activities separately from the other military services. However, over the past decade there has been a strong move towards integrating the operations of the three military services: army, navy and air force. This is known as *joint* operations. Where several nations integrate their operations, e.g. under the auspices of NATO, then this is known as *combined operations*. The trend has progressed to the point where all military operations are joint operations, and most are also combined operations.

At the same time, there has been a shift in military thinking away from strictly hierarchical organizations where the focus is on the *platform* (i.e. an aircraft, ship or tank). Instead, the focus is increasingly on the *network* that links the platforms together. The new thinking is termed *network-centric warfare* (NCW). The key idea is that the platforms connected to the network all share data to build up a combined picture of the battlefield. An example of the advantages NCW gives is that a fighter aircraft that dare not use its radar without giving away its position to the enemy, can attack using a radar picture provided to it by (say) a navy destroyer cruising off the coast. A further step in military thinking made possible by NCW is "sensor-to-shooter" operations. In ICT terms, two platforms – one acting as the sensor and the other as the "shooter" – operate peer-to-peer. The time taken for key sensory information to reach the shooter can be reduced from two hours to ten minutes, because the delays from passing the information up the organizational hierarchy from the sensor to a control center and back down again to the shooter are eliminated. The military advantage is that mobile or fleeting targets (e.g. ballistic missile launchers or key enemy commanders) can be

attacked, as the 2003 Iraqi War demonstrated. This has major consequences for how military forces are organized and what infrastructure – especially ICT – they need in order to operate.

Worldwide interest in unmanned air vehicles (UAVs) continues to expand, prompted by their successful employment for reconnaissance in Kosovo, Afghanistan, Yemen and Iraq. UAVs generally have a longer endurance than equivalent manned aircraft, with flight times of 24 hours being common. Moreover, UAVs are seen as a natural component in NCW because they have to be remotely controlled via communications links. There is little doubt that the UAV market will continue to expand faster than other aerospace sectors. However, developments have been uneven in the US with strategic systems racing ahead while tactical systems are tangled in bureaucracy. Significant obstacles to UAV employment are the high attrition rate and the need for high bandwidth for control and information gathering. The pattern of UAV deployment in Europe is markedly different, with widespread tactical systems but few strategic ones. Israel continues its pioneer role in sales to smaller armed forces.

Novel areas are emerging, such as unmanned combat air vehicles (UCAVs), micro-UAVs for urban combat, UAV swarms, and mother ships. UCAVs are armed UAVs, and first saw improvised use in Afghanistan in October 2001. Two UCAV configurations are emerging. The hunter-killer exploits the scouting ability of slow and ungainly UAVs to provide real-time target acquisition, coupled with target attack. The strike UCAV is a sleek, stealthy, unmanned jet that is better for striking at high-value, defended targets. Where the hunter-killer is suited to low-intensity conflict, the strike UCAV is more tailored to medium- and high-intensity conflict.

Micro-UAVs are UAVs that are so small and light that they can be hand-launched. They are seen as having potential applications in urban warfare and hostage situations. As they fly slowly, they can be used to fly into a building to gather information about the people inside.

UAV swarms can be employed to cover a wider area, to assure a permanent presence over a particular area, or to overwhelm enemy defenses by their sheer numbers. In the 2003 Iraq War, UAV missions were co-ordinated with manned missions to attract the attention away from the manned aircraft, increasing the latter's survival rate. A further development of the concept would be to fly UAV swarms in which the UAVs would be functionally specialized. For example, a swarm could be used for bi-static radar applications, with a large radar, probably ground-based, transmitting a strong signal. The UAV swarm would be used like a stealthy phased-array antenna to receive the faint signals from the target (e.g. stealthy cruise missiles) and to integrate them into a high-resolution and accurate picture. Another obvious application for functional specialization in UAV swarms would be to combine reconnaissance UAVs ("sensors") with armed UCAVs ("shooters"), shortening the time to attack a target after it has been detected (the "sensor-to-shooter time").

UAV swarms and mother ships are linked concepts. Mother ships are larger aircraft, often manned, that would carry the UAVs to their operational area, launch them, and then co-ordinate their activities. American studies center on large manned mother ships, such as AWACS or J-STARS (both Boeing 707-class aircraft). European studies are based on fighter-size aircraft, such as the Eurofighter, usually based on their two-seater variant developed for the training role. Unmanned mother ships are also being considered, e.g. for delivering micro-UAVs over an urban area.

Spacecraft operations   The 2003 Aerospace Source Book divides spacecraft operations into expendable launch vehicles (ELVs), re-usable launch vehicles (RLVs), and satellites. ELVs are con-

ventional rocket launchers, such as Atlas, Delta, Ariane, etc. The only operational RLV is the US Space Shuttle.

Half of all launch vehicle programmes average one or two launches per year. Major launchers, such as Atlas, Cosmos, Delta, Long March, and (up to 1 February 2003) the Space Shuttle, average one every three to four months. Proton launches once every two months, and only Ariane achieves a monthly launch rate.

Competition in the ELV industry has suddenly become much stiffer. The total of 60 launches in 2001 was a drop of 29% from 2000, and as low as the 1960s. The total in 2002 – 65 launches – was not much better. Even the most successful programme, Ariane, is facing serious financial pressures from the current slowdown. The original business case for the two Extended ELVs (EELV), Lockheed Martin's Atlas 5 and Boeing's Delta 4, assumed as many as 19 launches per year, half for the US government and half for commercial clients, but that was always wildly unrealistic. Ironically, Boeing's loss of US government business following the recent discovery that Boeing gained access to 37000 pages of Lockheed Martin's proprietary technical and cost data during the USAF's EELV competition may leave Lockheed Martin with a viable EELV market. New low-cost competitors are emerging, such as Japan's H-2A and India's GSLV.

The dozen or so RLV programmes have faded away. Only the Space Shuttle is operational, although the loss of Columbia after the 2003 Aerospace Source Book was published has led to a suspension of operations. The Orbital Space Plane (OSP) and Kistler Aerospace's K-1 are the only other RLV programmes left, apart from a handful of technology efforts. The reason is that there is not enough investment capital available to complete a commercial RLV. RLVs are seen as high risk, because they depend on new and advanced technologies, and appear to lack a market. The lack of market is probably more perceived than real, but RLVs would have to bring launch costs from the current US$10000-30000 per kilogram down to US$2000 per kilogram to support a launch rate of one per week. Realistically, US government and industry must accept that the Space Shuttle will be around for another two decades after its return to flight status. It may be modernized to carry cargo to the International Space Station, with the crew riding the OSP.

There were 75 satellites launched to Earth orbit in 2001, of which just sixteen were for commercial operators. That was the lowest number in the past decade, and represented a 32% drop from the year before. 2002 was not much better, with just over 80 satellites launched, 33 of them commercial. This is half of 1998's total of 150 satellites launched.

The satellite market relies heavily on established systems such as Eutelsat, Intelsat, and PanAmSat, often in geostationary Earth orbit (GEO). More that two-thirds of satellites launched are for GEO constellations. Many of the low Earth orbit (LEO) mobile communications and broadband satellite constellations have been canceled. The three remaining constellations are starting their first replenishment cycles, but have a very different financial status. In February 2002 Globalstar filed for Chapter 11 protection from bankruptcy. The Globalstar constellation remains operational, but failed to attract enough customers. Fortunately for Globalstar, most satellites have surpassed their seven-and-a-half year design lifetime by four or five years. That means that replenishment does not need to begin until 2006 or even later. By contrast, the post-bankrupt Iridium is doing well, thanks to US Defense Department business. Several Iridium satellites failed earlier than their design lifetime, and replenishment of the constellation began in 2002. Orbcomm replenishment should begin in 2003.

The outlook for 2004-5 is around 100 satellite launches per year, of which at least 30 are commercial. To create a growth market new applications are needed. The Internet is clearly fueling the development of business and consumer applications, but it is not clear how this will translate into satellite launches. Many Internet companies are leasing capacity on in-orbit assets, and this will be a growing trend for the first half of the decade. There is a growing demand for Earth observation, but the number of satellites is small. Military opportunities are fewer, but do offer longer-term work. The military satellite business in other countries is not as robust as in the US, which has 90% of the military market. Small scientific satellites make up the rest of the launches, with NASA launching six to eight satellites annually (about one-third of the civil total). ESA flies even fewer missions.

A strong trend that is apparent in all types of space missions is the increasing tendency to design the mission in the form of a constellation of satellites. There are close parallels here with UAV swarms. Constellations are needed to assure coverage in applications such as navigation and telecommunications. The highly-successful Global Positioning System (GPS) was designed as a constellation from the outset. The European Galileo system will also be a constellation. Early telecommunications spacecraft were large satellites positioned in geostationary earth orbit (GEO) to link two continents or to serve a particular continent. Some civil spacecraft operators (e.g. Intelsat and Eumetsat) have gradually built up global coverage by progressively adding GEO satellites. Some military operators such as NATO have followed the same route. Others, such as the UK and other smaller nations, have not needed global coverage. Only the superpowers have flown military telecommunications constellations with global coverage, like the US MILSTAR system.

In the 1990s several would-be civil operators designed their telecommunications missions as constellations from the outset. Most have failed for commercial reasons, with only Iridium, Globalstar and Orbcomm becoming operational. While this is a setback, the logic for flying telecommunications constellations is inexorable, especially given the trends towards globalization and increasing demands for bandwidth.

Similar forces driving mission designers towards constellations are apparent in other types of mission. Astra-SES has built up a global constellation of TV broadcast satellites. European, US, Japanese and Indian meteorological satellites effectively form multi-national GEO and polar-orbiting constellations. Future Earth Observation (EO) missions will likely be constellations, with ESA's Envisat being the last large multi-instrument EO satellite[1]. Several other scientific missions are being designed as constellations, such as Darwin and the Next Generation Space Telescope.

Planetary space missions are also likely to see the increasing application of constellations and swarms of roving robots. The concepts under development in both America and Europe for Martian exploration depend on putting telecommunications constellations in orbit around Mars, with swarms of rovers exploring the surface and flying in the Martian atmosphere. Proposals for looking for life in the ocean believed to be present under the icy surface of Europa are based on swarms of robots that tunnel through the ice and then act as submarines.

Spacecraft operations are not limited to what happens in space (the "space segment"). The ground-based activities (the "ground segment") supporting operation of spacecraft in real time are vital to the success of space missions. The elements of the ground segment are the spacecraft users' facilities, the spacecraft mission control center(s), the ground stations that transmit commands to the spacecraft ("telecommands") and receive data back ("telemetry"), and the ground-based telecommunications

---

[1] Even Envisat can be regarded as part of an EO constellation because it flies in the same orbit as ESA's ERS-1 and ERS-2.

network that links it all together. Planning and scheduling processes occur in the users' facilities and in the mission control center.

Spacecraft mission control processes are SOP-based and include [Don03]:

- Planning and scheduling

- Monitoring, diagnosis and control

- Resource management and off-line analysis

- Simulation and training

There are typically three mission control teams, working shifts to give 24 hours per day, seven days per week coverage. The Flight Control Team is responsible for the space segment, the Ground Control Team is responsible for the ground segment, and the Flight Dynamics Team performs the off-line analysis of the spacecraft's attitude and orbit.

Donati notes that the effective introduction of innovation in the mission control processes must take into account the needs of the mission control teams. These teams are the end-users and intended beneficiaries of innovation. Because of the teams' overriding focus on safety, they are very sensitive to potential risks introduced by system changes. Their operational workload is such that they devote marginal time and effort to support innovation. Financial resources for innovation are limited. Therefore, Donati advocates an incremental, user-centered approach to introducing innovation.

Another strong trend is the demand for increasing autonomy in spacecraft operations. Making spacecraft more autonomous enables them to handle a degree of change in the environment or within the spacecraft itself without referring every event to mission control. Autonomy can be essential in deep space missions, where the transmission time is too long for timely response to events. For example, Martian rovers have to be able to make their own decisions if they are about to crash into a rock, because the 25-minute Earth-Mars transmission time would mean that a "stop" command from Earth would arrive too late. A second driver may be that the ground-to-space communications link is periodically interrupted. For example, the spacecraft may be in low Earth orbit (LEO) but served by just one ground station. Then the spacecraft would be out of sight of the ground station for two-thirds of the time. A third driver for spacecraft autonomy is when financial restrictions mean that the mission control center is unmanned for part of the time, e.g. normal working hours apply. A long-term aim is to go to "lights-out" operation, i.e. the control center is only activated when extreme non-nominal events occur. The control center personnel normally do other things, and the satellite calls them out when it has a problem that it cannot handle.

Autonomy introduces changes both on-board the spacecraft and on the ground. Autonomous spacecraft have to be designed to include additional on-board control functionality. The consensus list of autonomous functions includes autonomous guidance, navigation, and (attitude and orbit) control (GNC), fault detection, isolation and recovery (FDIR), and mission planning and scheduling. However, no consensus has emerged yet that integrates these autonomous functions with one another or with more traditional on-board data handling (OBDH) architectures. Autonomous planning and scheduling using AI P&S technology has been demonstrated in NASA's Deep Space One mission as the Remote Agent Experiment (RAX).

On the ground, the mission control teams must understand what the spacecraft is doing. The autonomous functions add complexity to the spacecraft's behavior and to the state as shown in telemetry data. In extreme cases it might be difficult for the mission control team to determine whether the spacecraft has made a valid autonomous decision to reject a telecommand sent from the ground or has failed.

Air traffic management European airspace is particularly confined, and European air traffic is dense. Eurocontrol expects that air traffic will double over the next twenty years (see Figure 6.1). Some parts of the European airspace are already congested, and cannot absorb even today's level of demand at peak times.
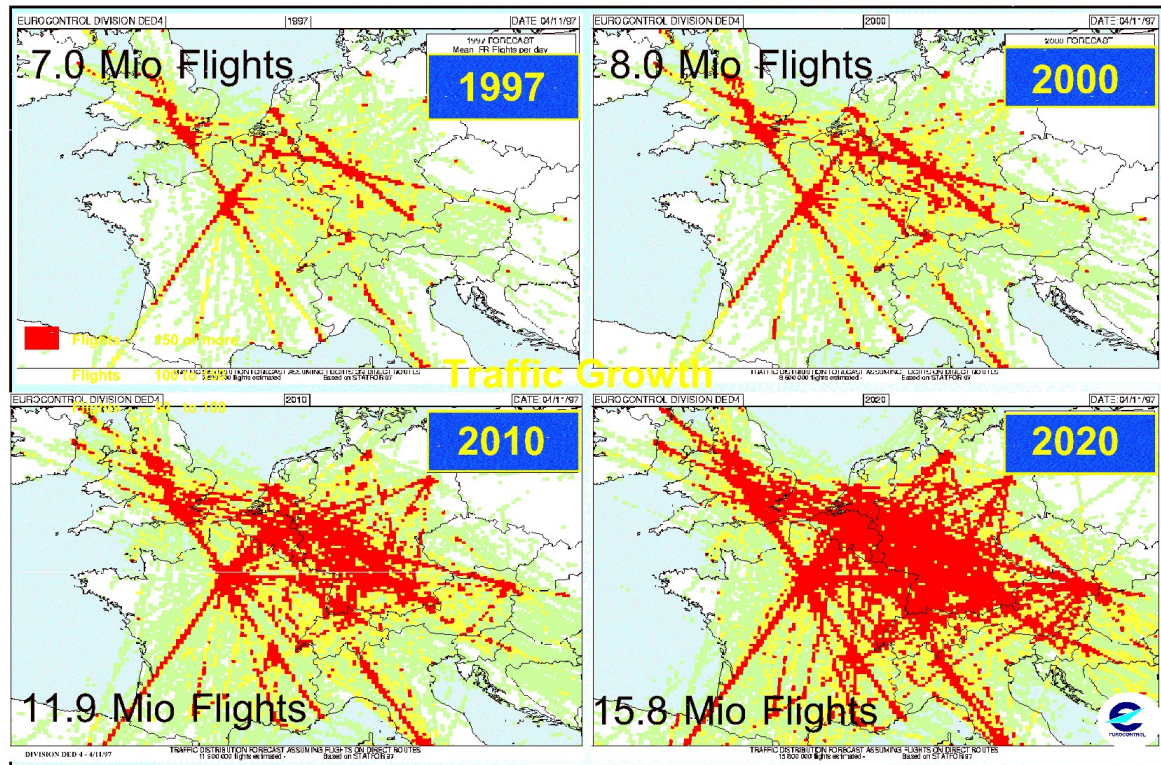
Figure 6.1.: Expected growth in European air traffic (source: Eurocontrol).

If nothing is done, both the number and severity of aircraft accidents will grow. The incidence of aircraft accidents is related to traffic density. As larger-capacity aircraft are introduced, like the Airbus A-380, then severity of accidents in terms of the number of passenger deaths can be expected to grow.

Air Traffic Management (ATM) is the process of managing and controlling airspace and its users. ATM is divided into four tasks:

- Runway arrival

- Runway departure

- Surface movement, i.e. the movement of aircraft and vehicles along taxiways

- En-route or flow control, i.e. the flight of aircraft from one airport to another

In the 1990s, all the various users of European airspace decided to study means by which additional capacity could be created while reducing direct and indirect ATM-related costs and increasing safety levels. The study was known as the European Air Traffic Control Harmonization and Integration Programme (EATCHIP). The study outcome was the "European ATM Strategy for 2000+".

The EATCHIP study showed that current ATM concepts and national infrastructures have inherent limitations and will become progressively less than adequate as traffic levels rise. Airport congestion, in particular, is likely to become a major concern and constraint on future aviation growth. Strong trends were identified in the way that air transport and ATM might develop, based on agreed policies and strategies. Nevertheless, there were uncertainties surrounding the feasibility of some potential concept options. There were a number of possible choices as to which change path to follow, each with its own balance in terms of costs and the capacity and efficiency gains that could be achieved.

The main operational concept options range between a managed ATM environment based on traffic structuring, greater traffic predictability, longer planning horizons, and extensive automated support, to a "free-flight" environment based on free routings and autonomous aircraft separation [EAT99]. In practice, the target concept will have to have elements of most of the available options to meet the varying requirements of the all the airspace users and the differing types of regional traffic conditions. The overriding need is to generate extra capacity in the busiest traffic areas while increasing safety levels.

The target concept is based on layered planning, based around a strategically-derived "daily airspace plan", with collaborative planning and decision-making between the parties involved. The emphasis must change from managing demand to managing resources. The concept incorporates a mix of route structuring, free routings, and autonomous aircraft operations to answer the needs of a diverse user community. This necessitates fundamental changes to current roles both in the air and on the ground. The responsibilities for assuring separation between aircraft must be distributed between the air and ground ATM elements, according to aircraft capabilities and the services provided. Greater use must be made of computer support tools to:

- Cope with increased levels of service;

- Keep cockpit and ATM workload within acceptable levels; and

- Enable more dynamic and flexible management of airspace.

Airport operations Many of the world's busiest airports are operating at their capacity limits. Airports are the bottleneck of the air transport system, resulting in flights that take longer than decades ago. Currently, around of 20% of the world's major airports are operating near saturation for most of the day. A further 50% is operating near saturation at peak hours.

An airport consists of many different kinds of resources, any one of which can become a bottleneck. There need to be sufficient runways long and strong enough to support the types of aircraft and numbers of flights arriving and departing at peak times. The taxiway network needs to be adequate

for the number of aircraft movements to and from the runways. There must be sufficient parking spaces, large enough to accommodate the number and sizes of aircraft using the airport. A sufficient fraction of these parking spaces needs to be provided with gates to the terminal buildings to load and unload the passengers. Another fraction of the parking spaces needs to be suitable for loading and unloading freight. The terminal buildings and their facilities (e.g. lounges, customs and immigration, baggage handling, check-in desks, security, catering, etc.) need to be sized adequately to cope with the peak throughput of passengers. A similar requirement applies to the cargo handling buildings and facilities. Sufficient technical accommodation and facilities are needed for the ground handling services, including refueling, aircraft power suppliers, aircraft servicing, towing, baggage trolleys, aircraft catering, etc.

There are many different stakeholders of many different kinds at an airport. There are several airlines, air traffic controllers, ground handling service providers, security staff, customs and immigration, fuel suppliers, caterers, bus and taxi companies, etc. Their activities have to be coordinated with one another to minimize conflicts.

Safety and security are overriding issues in airport operations. Aircraft safety is paramount. A major risk at airports is runway incursion. Runway incursion occurs when a runway is in use by an aircraft landing or taking off and a vehicle or another aircraft enters the runway. One of the world's worst air disasters occurred in Tenerife when one jumbo jet entered the runway from which another jumbo was taking off. Runway incursion has been one of the US National Transportation Safety Board's "most-wanted" items for the past 20 years [Boa03]. Loss of situation awareness is the main reason for nearly all fatal runway incursion accidents [Hug03].

The lengths to which airports will go to guarantee safety is illustrated by how Amsterdam's Schiphol airport designed its taxiway network when it was being extended with a sixth runway. To minimize the likelihood of runway incursion, the designers decided to route the new taxiways so that they did not cross any runway. This was done by designing the taxiways so that they ran around one or other end of every runway. In the resulting design, more land had to be purchased to accommodate the taxiways than if the taxiways had crossed the runways. Moreover, the taxiways were substantially longer. This means that aircraft take longer to reach the runway for take-off or to reach the gate after landing. More aircraft fuel is consumed in taxiing, and more noise and exhaust gases are generated, polluting the surrounding countryside and towns. The passengers have to sit longer in the aircraft.

Costs and the environment effects are important issues at airports. As the example of Schiphol's sixth runway shows, safety and security still override the issues of cost and environmental effects.

Maintenance, repair & overhaul (MRO) The current drivers in MRO are the delayed economic recovery and the shaky airline industry, according to the 2003 Aerospace Source Book. This will lead to suppressed MRO growth, weeding out the weak players, and spurring further consolidation, at least through to 2005. There is still too much MRO capacity, with fierce competition.

The MRO industry is re-thinking its short- and long-term plans. Dependent on the airlines, the industry is creating innovative packages offering reduced downtime and cost by turning to lean manufacturing methods. The shortage of skilled personnel is a problem, with most graduates from airframe and propulsion schools going to the higher-paying airlines, theme parks and public utilities. The key to survival is to provide a total service solution, including integrated information systems, processes and infrastructure, with closer co-operation between airlines, suppliers and the MRO industry.

A major challenge is managing the resulting extended enterprise, with different companies with their own procedures, work practices and information systems. Use and management of ICT will be crucial to success. Internet-based technologies are needed to support data exchange, with standards being an important issue. Historically, the MRO industry has been slow to embrace new technology, and, according to the 2003 AWST Source Book, is "far from the state-of-the-art in supply chain and shop floor management". Increasing collaboration means increasing demand for information systems specifically designed for the task. MRO operators that do not invest in this area will become extinct.

6.2.   Roadmap themes

This section surveys the planning and scheduling state-of-the-art and future needs in each of the sectors within the aerospace domain.

One factor common to all of the sectors is the overriding need for safety. This leads the whole aerospace domain to be conservative in applying new technologies. There is a strong emphasis on verification and validation of materials, processes, models, data, and plans prior to use. Operational knowledge is often documented in the form of Standard Operating Procedures (SOPs). SOPs can be carefully validated prior to use, again ensuring safety. In the aerospace domain, SOPs are very largely authored manually and validated using simulation. Automated SOP generation has been under development for 30 years in the petrochemical industry, but this has not yet migrated to the aerospace domain.

In our judgement, the desired future state in planning and scheduling in all sectors of the aerospace domain will include the following features:

Verification and validation in AI P&S Methods will have to be developed for verifying and validating domain and planning knowledge used in AI P&S.

SOP generation AI P&S technology needs to be developed for the automated and mixed-initiative generation of SOPs, drawing on the experience gained in the petrochemical industry.

6.2.1.   Aerospace manufacturing

The planning and scheduling state-of-the-art in aerospace manufacturing is much the same as in other manufacturing and production domains. The primary planning and scheduling applications are in process planning and in scheduling the manufacturing process to minimize the make-span for the assembly of aircraft and major systems (e.g. engines). The imperative to reduce costs ("lean manufacturing") has driven aerospace manufacturers to focus on maximizing the utilization of their workforce and other key resources. The conclusions of the PLANET Intelligent Manufacturing Systems (IMS) and Workforce Management (WfM) TCU roadmaps apply just as much to aerospace as they do to other manufacturing domains.

Aerospace manufacturing planning and scheduling is done off-line using large scheduling packages, such as ARTEMIS. These packages are based on the PERT and CPA algorithms from the 1950s. At the business level, aerospace manufacturers have adopted Enterprise Resource Planning (ERP)

products, such as SAP and Baan. Scheduling runs typically take longer than a shift to perform, making it impractical to perform "what-if" investigation of alternatives or to re-schedule in response to operational problems.

Aerospace manufacturing differs from other manufacturing domains in several respects. First, the aerospace industry is highly regulated. This drives aerospace manufacturers to define and standardize their processes by means of SOPs and similar mechanisms (e.g. "six-sigma production").

Second, the aerospace industry is vertically organized. Planning and scheduling of the supply chain is becoming increasingly important. For this reason, aerospace manufacturers are in the course of supplementing their ERP installations with Advanced Planning & Scheduling (APS) products that enable organizations in the supply chain to share plans and schedules.

Third, the aerospace industry is increasingly international in scope. Aerospace manufacturers demand that software tools interface with one another using international standards. CATIA has become the *de-facto* standard for designing aerospace products both within the American and the European industry. PGES/STEP has become the industry-wide standard for data exchange. Aerospace manufacturers have been among the leaders in adopting the Internet for purchasing products and services and for providing support services to their airline and air force customers.

Fourth, the aerospace industry is being faced with the need to incorporate planning and scheduling functionality into their products. As they become increasingly capable of autonomous action, UAVs, spacecraft, and planetary rovers will all require on-board planning and scheduling capabilities, together with the necessary on-ground domain modeling support. Suppliers of AI P&S products should now start to prepare for this new market.

In our judgement, the desired future state in planning and scheduling in the aerospace manufacturing sector will include the following features:

Further adoption of the Internet Web-based technologies must be introduced into all aspects of planning, scheduling, plan/schedule execution, execution monitoring, and re-planning and re-scheduling.

Faster scheduling The aim should be to reduce scheduling run-time to a fraction of a shift, making reactive re-planning and re-scheduling possible. The promise of AI P&S technology has already been demonstrated to Boeing by OnTime Systems' application of a constraint-based representation and squeaky-wheel optimization [Dra03].

The introduction of standards specific to planning and scheduling PDDL needs to migrate from the research world into commercial products. However, PDDL only provides standards for representing actions. Equivalent standards are needed for other kinds of knowledge used in planning and scheduling, e.g. plans, domain models, constraints, states, goals, and heuristics.

Application of standard AI P&S representations and algorithms throughout the supply chain APS enables plans and schedules to be shared, but this does not go far enough. Delivery of standard representations and algorithms throughout the supply chain is best achieved by the provision of standard web services for AI P&S, as well as for the associated domain modeling.

Recognition of the distributed, multi-disciplinary environment Given the geographical distribution and multi-disciplinary nature of the aerospace manufacturing industry, the development of distributed and collaborative AI P&S techniques is essential.

Integration into bigger systems AI P&S technologies need to be delivered in the form of components to enable their integration into bigger systems, particularly in applications on-board UAVs.

6.2.2. Aircraft operations

The planning and scheduling state-of-the-art in aircraft operations is characterized by the predominance of specialized products and services for planning and scheduling processes such as crew scheduling, timetabling, yield management, and flight or mission planning. Often these products and services will have been originally developed as stand-alone, batch applications by the airlines and air forces themselves.

Over the past decade or so, spin-off consultancy companies have been set up to exploit the commercial value of these products. The best of these companies have diversified into planning and scheduling in other domains, e.g. railway and bus operations, hospital operations, and so on. Nevertheless, there remains a close, almost symbiotic relationship between these planning and scheduling consultants and their airline customers.

Typically, operations knowledge is "hard-wired" into the products, making it difficult to sell the products to other airlines and impossible to apply to other domains. The product's market may also be limited by commercial and security considerations. When the users merge with other airlines, the knowledge embedded in the product may become outdated, making it necessary to replace the product.

AI P&S technologies are starting to make inroads into aircraft operations. For example, Parc Technologies Ltd in UK – a spin-off from Imperial College – market their AirPlannerTM product for airline timetabling and their Aircraft SwapperTM tool for short-term optimization of aircraft fleets. Both products are based on constraint technology in the ECLiPSe development environment. Additional examples are the visualization, business rules, and constraint-based optimization products of ILOG in France. Their products are widely used in crew, fleet and workforce scheduling in aircraft operations, as well as in domains other than aerospace.

In our judgement, the desired future state in planning and scheduling in the aircraft operations sector will comprise the following features:

Separation of domain-specific and planning and scheduling knowledge Operations knowledge should be separated from the planning and scheduling functionality, allowing the user to adapt their operations according to changes in the market. This can be achieved using AI P&S technology.

Usability issues To widen the market, the products must be made usable by airline and air force personnel. This requires the commercialization and application of mixed-initiative AI P&S techniques.

Real-time AI P&S The planning and scheduling products must escape from the off-line, batch processing approach, to provide re-planning and re-scheduling in real-time.

Collaborative AI P&S   Aircraft operations involves collaboration between multiple disciplines, such as pilots, cabin crew, dispatchers, aircraft engineers, ground services, passenger and freight handling, weather services, and air traffic control. The products should support collaborative planning and scheduling.

Distributed AI P&S   As airlines grow larger by merging, they may have several "hubs" or main operating bases. Their planning and scheduling processes will become distributed, making distributed AI P&S techniques a necessity.

Special AI P&S needs   In military aircraft operations, specialized AI P&S products and services are needed to support the joint/combined operations, NCW, and UAV operation.

6.2.3. Spacecraft operations

The planning and scheduling state-of-the-art in spacecraft operations is characterized by tailor-made applications for planning and scheduling processes. A recent study by the European Space Operations Center (ESOC) in Darmstadt, Germany, demonstrated that each space mission has its own planning and scheduling objectives, process and tool suite [Don02]. For example, the objectives for the Envisat mission planning system are to:

- Validate the end-user requests to ensure that these meet satellite and ground segment constraints.

- Provide the validated requests to the Payload Data Segment planning function. The Payload Data Segment schedules the ground stations and data acquisition facilities.

- Generate the telecommand schedule for the spacecraft.

- Generate time line event information for use by the Mission Control Team.

- Report on the success of actual operations against the plan.

Future developments and advanced concepts in spacecraft operations include coping with increased mission complexity, such as deep space missions, constellations, and autonomous spacecraft. Donati's analysis shows that the efficiency of mission planning and scheduling processes can be improved by:

- Automating the telemetry data importation process.

- Automating the multi-party planning cycle.

- Automating the management and resolution of activity or resource conflicts.

- Integrating the planning and scheduling processes.

- Designing core elements of the planning and scheduling processes so that they are re-usable from one mission to another.

- Implementing (autonomous) on-board dynamic planning and scheduling functions.

- Using goal-driven autonomous planning, i.e. to generate the telecommand schedule for the spacecraft on-board instead of on the ground.

- Implementing automatic or autonomous re-planning when a non-nominal situation arises. Automatic re-planning would be ground-based, and autonomous re-planning would be on-board.

From Donati's analysis, the desired future state in planning and scheduling in spacecraft operations will include the following features:

Real-time AI P&S Real-time AI P&S is needed to automate the planning cycle and the management and resolution of activity/resource conflicts.

Collaborative AI P&S Collaborative multi-disciplinary AI P&S is needed to support the planning process between the various parties involved in spacecraft mission control, i.e. the end-users and the mission control teams.

Distributed AI P&S Distributed AI P&S is needed for autonomous planning and scheduling in constellations and rover swarms.

AI P&S standardization Standards are needed for representing knowledge and algorithms used in planning and scheduling, to enable the re-use of planning and scheduling processes, products, and services across multiple missions.

Integration into bigger systems AI P&S technologies need to be delivered in the form of components to enable their integration into bigger systems, particularly in applications on-board autonomous spacecraft and planetary rovers. There is a lack of a consensus architecture, both on-board and on the ground.

Special needs In spacecraft operations, specialized AI P&S products and services are needed for autonomous on-board planning and scheduling in spacecraft and rovers, and for the ground-based mission control systems for autonomous spacecraft.

6.2.4. Air traffic management

The four ATM tasks – runway arrival, runway departure, surface movement, and en-route or flow control – involve different combinations of resources and have different characteristics. Runway arrival and departure involve the combination of aircraft, airspace, airport and runway. Of the two, arrival planning is the more difficult because it involves sequencing and timing of approaching aircraft so that a stream of aircraft is created such that the aircraft are optimally separated in time and space from one another. Arriving aircraft cannot stand still in mid-air to achieve the optimum sequence. By contrast, departing aircraft can be held at the start of the runway until the right moment is reached for them to begin their take-off.

Surface movement involves the combination of aircraft, the runway and taxiway network at an airport, parking spaces, gates, and other aircraft and vehicles. Although part of the ATM system, we have discussed this process under airport operations.

En-route or flow control involves the combination of multiple aircraft and multiple "chunks" of airspace. The most important goals in en-route control are to maintain adequate separation between the aircraft, as well as separation between the aircraft and terrain.

The "Target Concept Statement" of the EATMS Operational Concept [EAT99] is based on the idea of a collaborative and layered planning system, described as:

> "The exchange of current, relevant data between ATM, airports, Airline Operations Centers and aircraft, to enable the different system layers to support flexible decisions where needed, taking advantage of the availability of a common information pool, enhanced equipment, computer tools and operating procedures designed to increase capacity, efficiency and safety."

Increased involvement of the Airline Operations Center (AOC) and the airport in the ATM process are seen as priorities. At the operational level, the requirement for increased involvement is encapsulated in the concept of Collaborative Decision Making (CDM), identified as one of the corner stones of the EATMS Operational Concept. The ATM Strategy for 2000+ describes CDM as follows:

> "Both the collective requirements of all airspace users and the individual aircraft operator's preferences will be taken into account in determining solutions to events. The open systems environment and better information management will allow a permanent dialogue between the various parties (ATM, Aircraft Operators' Operations Centers, Pilots and Airport Operations) before departure, and as the flight progresses through the ATM system. This exchange of information will enable the various organizations to continuously update each other on relevant events in real-time and provide the basis for more efficient decision-making. Aircraft operators will have up-to-date and accurate information on which to base decisions about their flights, and will be able to apply factors which are not known to ATM, such as fleet management priorities, fuel consumption figures and other aircraft operating parameters, when determining solutions."

Besides CDM, the main characteristics of the EATMS Operational Concept are:

- Managing flights from gate to gate;

- Enhancing flexibility and efficiency;

- Responsive capacity management to meet demand;

- Managing airspace collaboratively; and

- Extending the level of automation in ATC.

Air Traffic Management and Control processes can be considered as a series of layers (see Figure 6.2). These layers include tactical ATC and a series of planning layers that aim to arrange the airspace and the traffic such that it is easier for tactical ATC. In time order, consider the following set of layers (or processes):

- The strategic management of airspace and traffic flows, including Airspace Management, strategic ATFM and the management of demand via Airport slots;

- Pre-tactical and tactical ATFM;

- ATC planning - the organizing of active traffic with the objective of simplifying the tactical control task (includes the short-term planning work of the Planning Controller and the medium-term planning tasks proposed in projects such as Arrivals Manager, Departures Manager and Multi-Sector Planner projects);

- Tactical ATC - the tasks carried out to achieve control and monitoring of flights to ensure safety of air navigation by users (includes radar and procedural control).

Similar planning layers exist in parallel for each Aircraft Operator (AO) (for crew, fleet, passenger management) and each Airport Authority (AA) (for ground management: gates, aprons, stands, etc.)
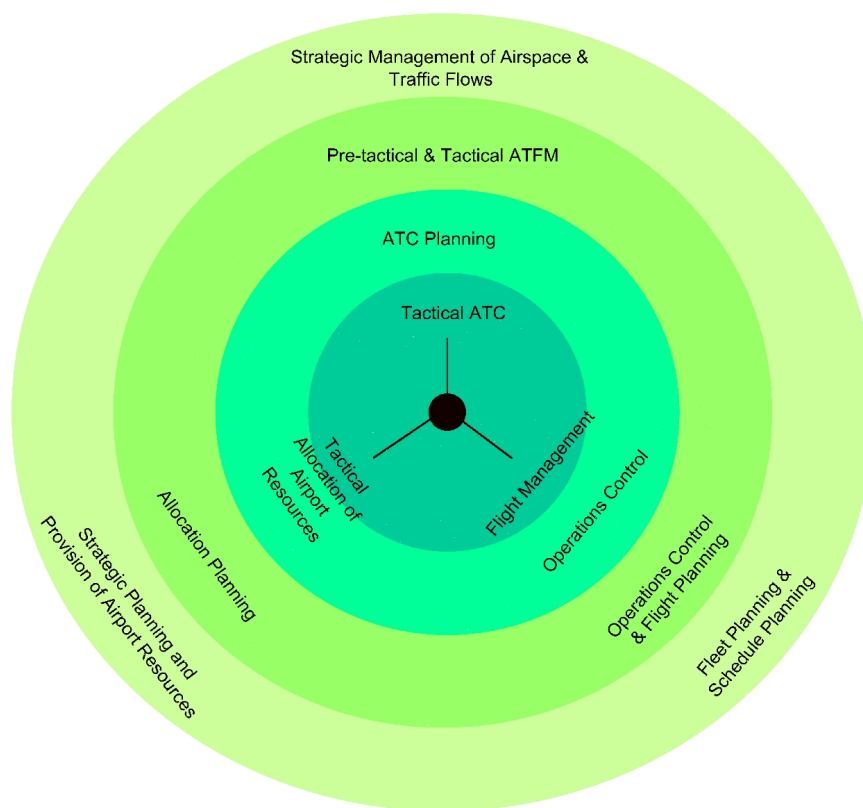
Figure 6.2.: Collaborative and layered ATM planning system [EAT98].

In order for a number of different parties to plan collaboratively, they must have access to consistent sets of information, including updates. (This is not to say that all actors will necessarily have the same set of information – each actor needs only the pieces of information that are relevant to him; they will in general hold overlapping subsets of the total available information.) Making available

that shared information is a necessary pre-requisite to the introduction of collaborative processes. Clearly, information management will be a key aspect of CDM.

[EAT98] envisages four possible levels of CDM in any area of the air traffic system:

- The first level is the distribution of information that already exists somewhere in the system to additional actors. In many cases, information held by one actor would be useful to other actors in their existing planning processes. In some cases, information is already distributed, but coverage is only partial, and therefore the information that is available cannot be used effectively. In some cases, it is simply a case of a useful presentation for the information being agreed between the suppliers and the users.

- The second level can be thought of as co-operation to improve the planning estimates that are available to all – where a number of actors each hold part of the picture, they can obtain a better prediction of what the overall outcome will be by pooling their information. (For example, in predicting a flight's take-off time considering the progress of ground preparation activities.)

- The third level is the modification of a planning process so that the current planner (usually ATM) takes into account the priorities of other actors, as communicated to him. In some cases, such collaborative planning processes could be introduced to enable the use of information that already exists in the system. In other cases, new information exchanges will be needed to support the process.

- The fourth level is the redistribution of decision-making to other actors, so that each decision is taken by the actor best placed to make that decision. This is only applicable when a decision can be separated from the overall "traffic picture". However, where it can be incorporated, it reduces the amount of information that has to be sent to the "central" planner, and therefore improves the overall efficiency of planning operations.

Note that it will not necessarily be appropriate to identify instances of collaborative applications at all of these levels in every aspect of Air Traffic Management. Also, the proposed levels should not be viewed as an implied sequence of implementation.

The applications identified in [EAT98] fit into the schema of planning layers and level of collaboration as shown in Table 6.1.

AI P&S technology is starting to appear in ATM applications. For example, the visualization, business rules, and constraint-based optimization products of ILOG are used for crew scheduling, although the emphasis appears to be on visualization.

In our judgement, the desired future state in planning and scheduling in ATM will comprise the following features:

Real-time AI P&S Real-time AI P&S is needed to automate the planning cycle and to manage and resolve conflicts between aircraft, terrain, parking spaces, gates, vehicles, and other resources relevant to ATM.

Collaborative AI P&S Collaborative multi-disciplinary AI P&S is needed to support the planning process between the various stakeholders involved in ATM, i.e. the pilots, airlines, AOCs, airports, and ATC authorities, particularly when CDM is adopted.

Planning layer: Level of collaboration:	Strategic management of airspace and traffic flows	Pre-tactical and tactical flow management	ATC planning	Tactical ATC
Level 1 - distribution of existing information	4. Information Distribution & Management	4. Information Distribution & Management 5.1 CDRs in Flight Planning	4. Information Distribution & Management 6.8 Airport Information for Flight Planning 7.1 Distribution of AO/Aircraft Flight Plan Information 7.3 Estimation of In-Block Time 8.2 Estimation of Departure Time 8.5 Information About Disruption	4. Information Distribution & Management
Level 2 - co-operation to improve planning estimates	4. Information Distribution & Management 6.2 Traffic Planning Model	4. Information Distribution & Management 6.2 Traffic Planning Model	4. Information Distribution & Management 8.3 Estimation of Off-Block Time	4. Information Distribution & Management 7.7 Meteo Information Exchange
Level 3 - additional actors' priorities in planning processes		6.3 Co-ordination between Airport Slot and ATFM 6.4 Re-routing 6.7 Slot Shifting	7.2 In-Flight Traffic Management 7.4 Optimization of Arrivals 7.5 Integrated Arr/Dep Mgt 8.4 Collaborative Departures Sequencing	
Level 4 - redistribution of decision making	6.1 Collaborative Flow & Capacity Management	6.1 Collaborative Flow and Capacity Management 6.5 Slot Swapping 6.6 Substitution on Cancellation 8.6 Disruption Recovery - Departures from Nearby Airfields	8.1 Collaborative Stand and Gate Management	7.6 Autonomous Separation in Free Flight Airspace

Table 6.1.: Potential applications by planning layer and collaboration level [EAT98].

Distributed AI P&S Distributed AI P&S is needed because the planning and scheduling processes in ATM are geographically distributed over airports and ATC sectors, and functionally distributed over the various stakeholders.

AI P&S standardization Standards are needed for representing knowledge and algorithms used in planning and scheduling, to enable the re-use of planning and scheduling processes, products, and services between the various stakeholders in ATM.

Integration into bigger systems AI P&S technologies need to be delivered in the form of components to enable their integration into bigger systems, particularly in CDM.

6.2.5. Airport operations

The planning and scheduling state-of-the-art in airport operations is characterized by each stakeholder having its own planning and scheduling processes, technologies and tools. There are specialized planning and scheduling products and services for runway slot planning, gate scheduling, and workforce management. Expert systems technology has been a feature of some products since the mid-1980s.

There is a clear need for the various stakeholders in airport operations to transition from stand-alone planning and scheduling processes to collaborative, airport-wide planning and scheduling processes. The stakeholders are aware of this need, and initial discussions have begun at a number of airports, including Schiphol.

AI P&S technology is starting to appear in airport operations applications. For example, the visualization, business rules, and constraint-based optimization products of ILOG are used in airport terminal management. A new generation of products based on AI P&S technology are needed for runway slot planning, gate scheduling, and workforce management, to replace the older, expert systems-based products.

In our judgement, the desired future state in planning and scheduling in airport operations sector will comprise the following features:

Real-time AI P&S Real-time AI P&S is needed to automate the planning cycle and the management and resolution of activity/resource conflicts.

Distributed AI P&S Distributed AI P&S is needed for airport operations applications. The various stakeholders are functionally diverse, and many of them also have many geographical locations.

Collaborative AI P&S Collaborative multi-disciplinary AI P&S is needed to support the planning process between the diversity of stakeholders involved in airport operations.

AI P&S standardization Standards are needed for representing knowledge and algorithms used in planning and scheduling, to enable the re-use of planning and scheduling processes, products, and services across the various types of stakeholders found in airport operations.

Integration into bigger systems AI P&S technologies need to be delivered in the form of components to enable their integration into bigger systems, particularly in a collaborative planning and scheduling processes for all stakeholders at a given airport.

6.2.6. MRO

The planning and scheduling state-of-the-art in MRO is very similar to the planning and scheduling state-of-the-art in aerospace manufacturing. Many MROs are spin-offs from manufacturing companies, although others are airline or air force spin-offs. As in aerospace manufacturing, lean manufacturing methods and collaboration are important. The shortage of skilled personnel is a problem. As in aircraft operations, ICT is being exploited to create innovative service packages.

In our judgement, the desired future state in planning and scheduling in the MRO sector is the same as for aerospace manufacturing.

6.3. Summary and conclusions

Desired future state of aerospace planning and scheduling

Having considered each sector in turn, we can now summarize the desired future state for aerospace planning and scheduling, as follows:

Separation of domain-specific and planning and scheduling knowledge In some sectors, operations knowledge is not yet separated from planning and scheduling functionality. AI P&S technology allows separation to be achieved, enabling the user to adapt their operations according to changes in user needs.

Faster, real-time planning and scheduling Planning and scheduling products must escape from the off-line, batch processing approach. The aim should be to reduce planning and scheduling run-time, making reactive re-planning and re-scheduling possible. Real-time AI P&S is needed to automate the planning cycle and to manage and resolve activity/resource conflicts.

Distributed AI P&S All sectors of the aerospace domain are geographically and/or functionally distributed. Distributed AI P&S need to be developed.

Collaborative AI P&S In most aerospace sectors, collaborative AI P&S is needed to support the planning process between the various disciplines represented among the planning and scheduling stakeholders.

Standardization in AI P&S Standards are needed for representing knowledge and algorithms used in planning and scheduling, to enable the re-use of planning and scheduling processes, products, and services. A start has made within the academic AI P&S community in the form of PDDL, but PDDL has yet to migrate into commercial products. Moreover, PDDL only provides standards for representing actions. Equivalent standards are needed for other kinds of knowledge used in planning and scheduling, e.g. domain models, constraints, states, goals, plans, and heuristics. Standard AI P&S representations and algorithms need to be delivered throughout the aerospace supply chain by the provision of standard web services, as proposed in PLANSERVE. APS enables plans and schedules to be shared, but this does not go far enough.

SOP generation AI P&S technology needs to be developed for the automated and mixed-initiative generation of SOPs, drawing on the experience in the petrochemical industry.

Verification and validation in AI P&S Verification and validation is essential in the safety-conscious aerospace domain. Methods will have to be developed for verifying and validating domain and planning knowledge used in AI P&S.

Usability issues in AI P&S To widen the market, the products must be made usable by end-users. This requires the commercialization and application of mixed-initiative AI P&S techniques.

Further adoption of the Internet Web-based technologies must be introduced into all aspects of planning, scheduling, plan/schedule execution, execution monitoring, and re-planning and re-scheduling.

Integration into bigger systems AI P&S technologies need to be delivered in the form of components or web services to enable their integration into bigger systems. To do this, there needs to be consensus on the functional architecture into which AI P&S components are placed.

Special AI P&S needs Specialized AI P&S products and services are needed to support autonomous vehicles (such as UAVs, autonomous spacecraft, and planetary rovers), and to enable network-centric or peer-to-peer systems.

Research needed

This section identifies the research needed to satisfy the desired future state of aerospace planning and scheduling. First, the current state of AI P&S research is established. Next, the differences between the current and desired future states are identified. Finally, the research needed is set against the likely time when it will become available for application, yielding a roadmap.

Current state of AI P&S research

The material which is suggested for further reading (cf. Section A) shows that a wealth of techniques have been developed for generating plans and schedules. Some of these have been applied to the aerospace domain. These are exclusively space applications. Among them, we find the following:

AI P&S system	Space mission	Time period
DEVISER	Voyager	1979-81
O-PLAN	Optimum AIV	Early 1990s
O-PLAN	ERS-1	Early 1990s
NMRA	Deep Space 1	1997
ASPEN	Mars Rover 1	2003

A number of topics can be identified that are currently an active area of research, as follows:

- Knowledge representation, including domain modeling, ontologies, protocols and formal planning languages.

- Reasoning methods, including classical planning, HTN planning, plan-graph planning, SAT-planning, and model-based planning.

- Integrating planning and scheduling techniques.

- Verification and validation in planning and scheduling.

- Knowledge acquisition and engineering, including techniques and tools.

- Relationships with the user, including automated vs. mixed-initiative vs. manual planning, and deliberative vs. hybrid vs. reactive on-line planning.

- Timeliness in planning, including one-shot vs. continuous planning, real-time planning, and anytime planning.

- Various planning tasks, including plan generation, plan validation and evaluation, and plan recognition.

- The uses of plans, including control, execution, interleaving planning and execution, execution monitoring, plan failure, and plan repair.

- Distributed planning and scheduling.

Differences between current and desired future states

The differences between the desired future state from the previous chapter and the current state of AI P&S research are shown in the following table:

Desired future state	Current research state	Research needed
Separation of knowledge	Standard practice in AI P&S	Continuation of fundamental research into knowledge representation for AI P&S and intensify research into knowledge engineering and acquisition for AI P&S
Faster, real-time planning and scheduling	International Planning Competitions have stimulated several orders of magnitude in speed-up, but real-time aspects (concurrency, deadlines, continuous planning, anytime algorithms) have received little attention.	Continuation of on-going improvements in planning time. Intensify research into real-time, concurrency, deadlines, continuous and anytime AI P&S. Covered by DCIRM FP6 proposal.
		continued on next page

Desired future state	Current research state	Research needed
Distributed AI P&S	Limited and dated (e.g. PGP) or at level of intelligent agents	Intensify research into distributed AI P&S. Part of research into architectures for AI P&S. Covered by DCIRM FP6 proposal.
Collaborative AI P&S	None	Initiate research into collaborative / multi-disciplinary AI P&S, perhaps by extending mixed-initiative planning (itself received only limited attention) to multiple human users. Real-time and distributed AI P&S is pre-requisite. Central to DCIRM FP6 proposal.
Standardization in AI P&S	Only action representation, yielding PDDL, and then only within academic community	Intensify standardization. Extend to all AI P&S-related representations for use by commercial and academic communities (e.g. NEN, OSI)
SOP generation	Extensive research in the petro-chemical industry, but not migrated to aerospace domain	Intensify research into SOP generation, and migrate to aerospace and other SOP-oriented domains (e.g. health, nuclear power, military)
Verification and validation in AI P&S	Some research using model-checking techniques	Continuation of research in model-checking and initiate research into other V&V techniques, drawing on software engineering
Usability issues in AI P&S	Islands of research	Intensify research into mixed-initiative planning and usability of AI P&S tools and products
Further adoption of Internet	Limited	Initiate development of web services for plan generation and knowledge engineering. Central to PLANSERVE FP6 proposal.
Integration with bigger systems	On-going research into integrating planning and scheduling (hot item – often known as temporal planning), plan generation and execution, and knowledge engineering and plan generation. Lack of consensus architecture, both at bigger level and at level of knowledge engineering, plan generation, scheduling, and execution	Initiate research into multi-level architectures for AI P&S, including defining interfaces. Research questions include "where do goals come from?" and specific interfaces to be defined include those between AI P&S and state estimation (e.g. autonomous GNC), and between AI P&S and FDIR.

Desired future state	Current research state	Research needed
Special needs – autonomy	Technology demonstrators flown, e.g. RAX	As specialization of research into multi-level architectures for AI P&S, initiate research into architectures for AI P&S in autonomous systems. Development needed to migrate AI P&S for autonomy to applications
Special needs – network-centric and peer-to-peer AI P&S	None	As specialization of research into collaborative, multi-disciplinary AI P&S, initiate research into network-centric and peer-to-peer (Grid) approaches to AI P&S. Real-time and distributed AI P&S is pre-requisite.

In summary, AI P&S research should be:

- Continued:

  - Speeding up planning time.

- Intensified:

  - Knowledge engineering for AI P&S.
  - Real-time AI P&S, including concurrent, continuous, and anytime AI P&S, re-planning and re-scheduling, and plan generation and scheduling with deadlines on planning time.
  - Distributed AI P&S.
  - SOP generation using AI P&S techniques.
  - Mixed-initiative AI P&S and usability.
  - Standardization in AI P&S.
  - Verification and validation by model-checking and other techniques.

- Initiated:

  - Collaborative AI P&S between multiple disciplines.
  - Development of web services in AI P&S.
  - Multi-level architectures for AI P&S, including network-centric and peer-to-peer architectures, architectures for autonomous systems, and defining interfaces.

Roadmap

The timescales for achieving maturity in each research need was judged on the basis of whether research was on-going. Where there was currently active research then it was assessed as being mature in about three years. Where research was on-going but it was limited or sporadic, then it was

Now	3 years	6 years	10 years
			Collaborative IP&S
			Standardisation in IP&S
		Real-time, concurrent IP&S	
		Verification & validation for IP&S	
		Web services for IP&S	
		Mixed-initiative and usability in IP&S	
	SOP generation		
	KE for IP&S		
Architectures for IP&S			
Distributed IP&S			
Classical & HTN planning			
Scheduling using constraint-based reasoning			
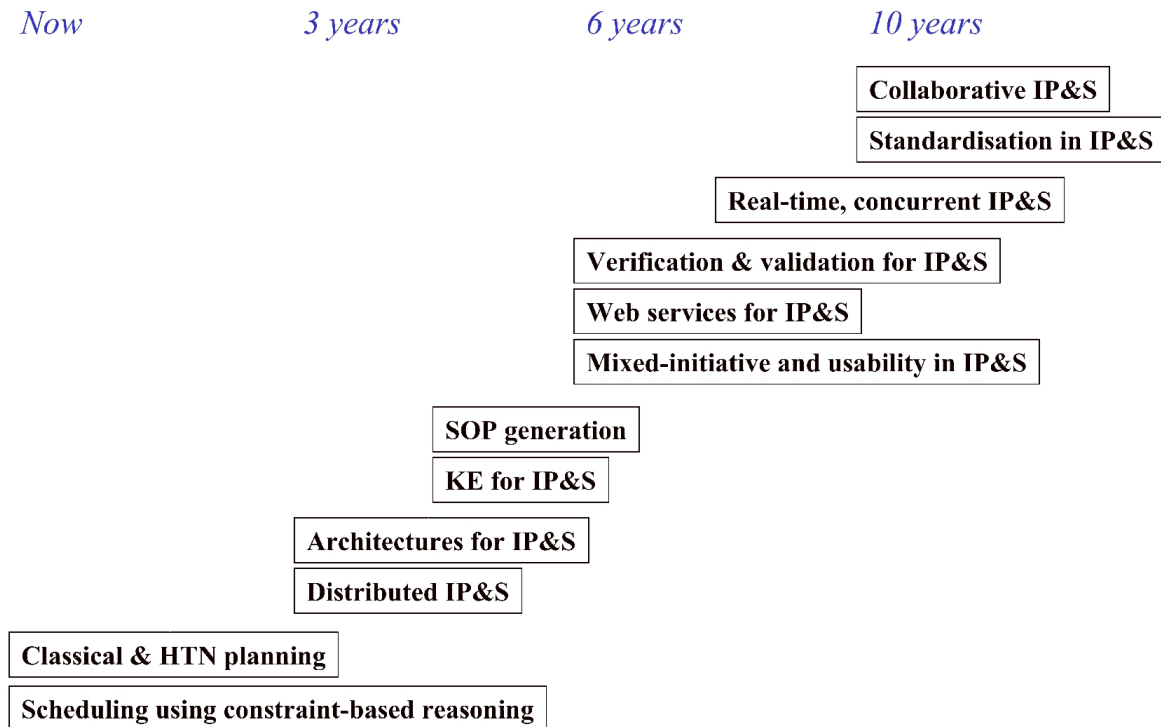
Figure 6.3.: AI P&S roadmap for aerospace applications.

assessed as needing around six years to achieve maturity. Research that has yet to be initiated was assessed as needing around ten years.

There are some dependencies between research needs. In particular, research into distributed AI P&S is an essential enabler for research into collaborative AI P&S. Research into architectures for AI P&S is needed for AI P&S web services, real-time and concurrent AI P&S, and for collaborative AI P&S. Architectures for AI P&S is closely associated with distributed AI P&S. Standardization can only be achieved fully when other research needs have been largely addressed.

The resulting roadmap for AI P&S in aerospace applications is shown in Figure 6.3.

It is worthwhile comparing this PLANET Aerospace Applications Roadmap with other roadmaps relating to AI P&S, particularly in the aerospace domain.

Polle proposed a technological roadmap [Pol01] for autonomous European spacecraft constellations, such as Galileo and Darwin. "On-line planning and scheduling" appears midway down their prioritized list, but is not further divided into topics. We regard this as equivalent to our real-time, concurrent, continuous and anytime AI P&S.

Kortenkamp proposes a more ambitious roadmap [Kor01] with chronological lines for a wide variety of technologies, one of which is planning and scheduling. Planning and scheduling is seen as progressing from "simple task planning for single subsystems", through "planning for different timescales", "mixed-initiative and crew activity planning", and "crop and menu planning", to "continuous planning and replanning". The first two of these are well within the capabilities of current

AI P&S technology, as is crew-activity planning. Crop and menu planning is a specific application. The three remaining areas of planning and scheduling research – mixed-initiative planning, continuous planning, and re-planning – are included in our roadmap.

The European Network of Excellence for Agent-Based Computing (AgentLink) has published its roadmap [LMPG02]. This calls for the development of multi-agent planning, but goes into little detail of the AI P&S needed. We regard this as equivalent to our distributed and collaborative AI P&S.

Steps towards technology transfer

It is difficult to assemble a set of steps towards technology transfer when the application domain consists of a wide diversity of sectors, as for aerospace. Moreover, AI P&S itself consists of a great variety of technologies. The PLANET network contains a diversity of organizations, including universities and research institutes as well as suppliers, each with their own strengths and needs.

Nevertheless, some broad-brush indications can be given. For example, an early step should be to build on the existing relationship with the European Space Agency. ESA's invitation to become their center of excellence for planning and scheduling has not been followed up, mainly because PLANET's modus operandi and limited lifetime does not lend itself to this. Even so, a group of PLANET nodes with shared interests in aerospace could decide to collaborate in meeting ESA's needs. In view of the increasingly close relationship between ESA and the EU, it would be valuable to seek support through EU channels.

Another possible initiative would be to make contact with Eurocontrol. They clearly have a need for expert advice on the relevance of AI P&S for their large-scale Collaborative Decision-Making research programme. Similarly, it would be worthwhile to team up with the members of the IMS TCU to approach Airbus. For both Eurocontrol and Airbus, it would be beneficial to exploit the channels that the EU will undoubtedly have with both organizations.

Apart from making contact with ESA, Eurocontrol and Airbus, some general guidelines can be given to PLANET member organizations, as follows:

- Gaining domain knowledge is essential. Since the aerospace domain is so diverse, focus on a small subset of aerospace sectors. This might be dictated by existing relationships your organization has with organizations in the aerospace domain. It might be determined by the proximity of particular aerospace organizations, or by personal preference or private contacts.

- Aim to introduce AI P&S innovations incrementally by small steps to reduce risk. Start with scheduling because this is repeated more often than plan generation. The benefits to be gained in scheduling, particularly the financial ones, are more obvious.

- Start with tactical planning and gradually shorten the time horizon. For example, start with plans that cover the next month or week's operations. Only when the user is convinced of the benefits of AI P&S at the tactical level, then progress to plans that cover the next day's operation. When that is assured, then progress to real-time re-planning and re-scheduling.

- Give adequate consideration to the non-functional features of AI P&S systems, i.e. those features that do not directly relate to AI P&S. For example, users may well gain more benefit

from a system that is fast-running and interfaces seamlessly to other software in the user organization (e.g. CADCAM, ERP, workflow management) than from the AI P&S technologies it employs. Architectures and standardization are key enablers for interoperability.

- To build confidence, be prepared to implement an AI P&S system so that it runs side-by-side with the existing planning and/or scheduling system for comparison purposes over several months. Equally well, be prepared to switch over quickly if the comparison shows that AI P&S does indeed offer major benefits.

A specific proposal can be made with respect to SOP generation. There has been 30 years of work on SOP generation in the petrochemical industry. PLANET member organizations that have relevant experience should transfer their knowledge to the aerospace domain, probably best starting with aerospace manufacturing.

Conclusions and recommendations

Conclusions

This Aerospace Applications Roadmap has surveyed the state-of-the-art in the aerospace domain, showing that it consists of the following sectors, each with their own characteristics and AI P&S-related needs:

- Aerospace manufacturing
- Aircraft operations
- Spacecraft operations
- Air Traffic Management
- Airport operations
- Maintenance, Repair and Overhaul (MRO)

Each sector has been considered in turn to identify its future needs for AI P&S. The overall desired future state of aerospace AI P&S includes the following features:

Separation of domain-specific and planning and scheduling knowledge This is enabled by AI P&S technology.

Faster, real-time planning and scheduling Real-time AI P&S is needed to automate the planning cycle and to manage and resolve activity/resource conflicts.

Distributed AI P&S All sectors of the aerospace domain are geographically and/or functionally distributed.

Collaborative AI P&S Collaborative AI P&S is needed to support the planning process between the various disciplines represented among the planning and scheduling stakeholders in most aerospace sectors.

Standardization in AI P&S Standards are needed for representing knowledge and algorithms used in planning and scheduling, to enable the re-use of planning and scheduling processes, products, and services. Standard AI P&S representations and algorithms could be delivered throughout the aerospace supply chain by the provision of standard web services.

SOP generation AI P&S technology needs to be developed for the automated and mixed-initiative generation of SOPs, drawing on the experience in the petrochemical industry.

Verification and validation in AI P&S Verification and validation is essential in the safety-conscious aerospace domain.

Usability issues in AI P&S To widen the market, the products must be made usable by end-users.

Further adoption of the Internet Web-based technologies must be introduced into all aspects of planning, scheduling, plan/schedule execution, execution monitoring, and re-planning and re-scheduling.

Integration into bigger systems AI P&S technologies need to be delivered in the form of components or web services to enable their integration into bigger systems. To do this, there needs to be consensus on the functional architecture into which AI P&S components are placed.

Special AI P&S needs Specialized AI P&S products and services are needed to support autonomous vehicles and to enable network-centric or peer-to-peer systems.

Comparing the desired future state of aerospace AI P&S with the current state of AI P&S research shows that AI P&S research should be:

- Continued:
    - Speeding up planning time.

- Intensified:
    - Knowledge engineering for AI P&S.
    - Real-time AI P&S, including concurrent, continuous, and anytime AI P&S, re-planning and re-scheduling, and plan generation and scheduling with deadlines on planning time.
    - Distributed AI P&S.
    - SOP generation using AI P&S techniques.
    - Mixed-initiative AI P&S and usability.
    - Standardization in AI P&S.
    - Verification and validation by model-checking and other techniques.

- Initiated:
    - Collaborative AI P&S between multiple disciplines.
    - Development of web services in AI P&S.

– Multi-level architectures for AI P&S, including network-centric and peer-to-peer architectures, architectures for autonomous systems, and defining interfaces.

Given the diversity of the aerospace sectors, the AI P&S technologies, and the PLANET organizations, it is difficult to define a single set of steps towards technology transfer. Broad-brush indications can be given. For example, it would be best to build on the existing relationships between ESA and PLANET. Eurocontrol should be approached next, with Airbus following. In each case, the maximum use should be made of the channels provided by the EU.

Some general guidelines can be given to PLANET member organizations, as follows:

- Gain domain knowledge by focusing on a small subset of aerospace sectors.

- Introduce AI P&S incrementally by small steps to reduce risk.

- Start with tactical planning and gradually shorten the time-horizon as successes are booked.

- Consider the non-functional features of AI P&S systems.

- Build confidence by running AI P&S systems side-by-side with the existing systems.

PLANET member organizations with experience in SOP generation in the petrochemical industry should migrate this experience to the aerospace domain.

Recommendations

We recommend that AI P&S research at the national and European levels should be initiated and intensified into:

- Multi-level architectures containing AI P&S components, including distributed, real-time and collaborative AI P&S.

- Knowledge engineering methods and tools for AI P&S.

- Delivery of AI P&S techniques in the form of web services.

- Mixed-initiative and usability techniques for AI P&S.

- Verification and validation of AI P&S.

- Standardization of AI P&S representations and algorithms.

We recommend that the PLANET Project Officer should support the PLANSERVE and DCIRM FP6 proposals, aimed at furthering research into delivering AI P&S web services and into an AI P&S-based, real-time, distributed and collaborative architecture for dynamic risk management.

We recommend that PLANET member organizations with experience of SOP generation techniques in the petrochemical industry should migrate the technology into the aerospace domain.

We recommend that interested PLANET member organizations should work together to build on the ESA invitation to become ESA's center of excellence in AI P&S, and to seek the EU's help in approaching Eurocontrol and Airbus.

We recommend that PLANET member organizations should take note of the guidance in transferring AI P&S technologies into aerospace applications.

Bibliography

[Boa03]   National Transportation Safety Board. Most wanted transportation safety improve-
             ments: Runway incursion/ground collision of aircraft. Web-site at `http://www.`
             `ntsb.gov/recs/mostwanted/runways.htm`, 2003.

[CW94]   Cross and Walker. DART: Applying knowledge-based planning and scheduling to
             crisis action planning. In Zweben and Fox, editors, *Intelligent Scheduling*, CA, USA,
             1994. Morgan Kaufmann.

[Don02]   A. Donati. Space mission operations planning and scheduling: Past, present and
             future, european space operations centre. invited presentation, PLANET 2nd Gap-
             Bridging Seminar, Delft University, Netherlands, November 2002.

[Don03]   A. Donati. Control technologies at ESOC: Current projects and future perspectives.
             PLANET Information Day in Ulm, Germany, 16 June 2003.

[Dra03]   B. Drabble. Intelligent planning & scheduling: Making a difference in the world.
             PLANET Information Day in Ulm, Germany, 16 June 2003.

[EAT98]   *EATCHIP, European Air Traffic Management Systems: State of the Technologies in
             CNS*, 1998. Reference FCO.ET3.ST01.

[EAT99]   Eurocontrol and European Organisation for the Safety of Air Navigation. *EATCHIP:
             Operational Concept Document*, January 1999. Reference FCO.ET1.ST07.DEL01,
             Edition 1.1.

[GCRM+03]   T.J. Grant, A. Cesta, M-D. R-Moreno, S. Biundo, and M. Ghallab. A european
             roadmap for applying intelligent planning and scheduling in aerospace domains. In
             Proceedings, SMC-IT'03, pages 131–139, Pasadena, CA, USA, July 2003. JPL Publi-
             cation 03-13A (paper) / 03-13B (CD-ROM).

[Hug03]   D. Hughes. A safety catch-22. *Aviation Week & Space Technology*, pages 60–61, April
             21 2003.

[ICGH96]   N. Fowler III, S.E. Cross, T.D. Garvey, and M. Hoffman. The ARPA-Rome labora-
             tory knowledge-based planning and scheduling initiative. In A. Tate, editor, *Advanced
             Planning Technology: Achievements of the ARPA/Rome Laboratory Planning Initia-
             tive*, Menlo Park, CA, USA, 1996. AAAI Press.

[Kor01]   D. Kortenkamp. Autonomy roadmap for long-term human space missions. In *ESA
             Workshop on On-Board Autonomy*, 2001.

[LMPG02]   M. Luck, P. McBurney, C. Priest, and C. Guilfoyle. The agentlink agent technology
             roadmap. *AgentLink II*, May 2002.

[Pol01]   B. Polle. Autonomy requirements for future constellations of spacecraft. In *On-Board
             Autonomy workshop, ESTEC*. ESA, 2001.

6.4. Appendix: Aerospace domain analysis

This Appendix presents an object-oriented analysis of the aerospace domain.

The analysis centers on the vehicles and related object-classes to be found in the aerospace domain. The key classes of vehicle are aircraft and spacecraft. With present-day aerospace technology, aircraft can only fly within the Earth's atmosphere. By contrast, spacecraft are designed to operate where there is essentially no atmosphere. Space launchers climb as quickly as possible to leave the atmosphere, where the satellites they carry are released into orbit. This mutually-exclusive distinction between aircraft and spacecraft makes it possible to partition the aerospace domain into aviation and space sub-domains.
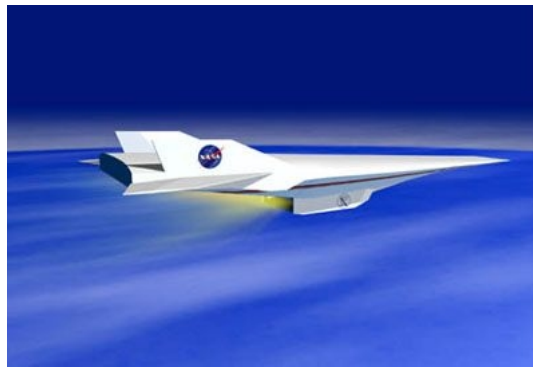
Figure 6.4.: Artist's impression of third-generation RLV.

The partitioning into aviation and space sub-domains will only remain valid for a number of years. Aerospace vehicles are now under development that will combine the features of aircraft and spacecraft. The only current example is the US Space Shuttle, but a second generation of so-called Reusable Launch Vehicles (RLVs) is under development. Second-generation RLVs will still be licensed according to space law and launched from dedicated sites like Cape Canaveral. Their operation will be kept separate from conventional aircraft. However, a third generation of RLVs is already foreseen (see Figure 6.4) that will take-off and land at airports, will fly in the same airspace as conventional aircraft, and will be licensed according to both space and aviation law. The domain analysis would then need to be refined, but this is unlikely to be necessary within the 10-year horizon.

Ontology for domain analysis

Planning and scheduling are particular forms of reasoning about actions. We adopt the traditional planning and scheduling view of actions, namely to regard them as changes in the state of domain objects, i.e. state-transitions. We model the state of an object as the set of binary instance relationships between it and other objects. States are constrained by domain invariants. In short, the domain analysis in this document uses an Object-Relationship-Invariant (ORI) ontology.

Object-classes and -instances

The domain object-classes and -instances form the first part of the ontology.

A mapping is made from the real objects found in the domain to modeled objects. Individual objects are modeled as object-instances. The object-instances are classified by similarity of their structure and behavior into object-classes. For example, the observation of real aviation activities[2] would lead to the sighting of many specific aircraft. Each aircraft observed could be modeled as an instance of the class of Aircraft.

Relationships

Inter-object relationships form the second part of the ontology.

Real objects interact with one another in a variety of ways. Some interactions are static and others are dynamic. These interactions are modeled as relationships. For example, someone observing activities at an airport would quickly see that aircraft take off from runways. This could be modeled as a (dynamic) relationship between the Aircraft and Runway classes.

There are three types of relationship that are relevant here:

Generalization-specialization The generalization-specialization relationship is a static relationship between two or more object-classes. One object-class (the *super-class*) is a generalization of the other object-classes (the *sub-class(es)*). The sub-classes inherit the characteristics of their super-class. Accordingly, generalization-specialization is often known as *inheritance*. Inheritance is transitive. We restrict sub-classes to having exactly one super-class, i.e. the generalization-specialization relationship structure forms a hierarchy, sometimes known as an *inheritance hierarchy*. In addition, we restrict super-classes to being *abstract* classes, i.e. they cannot have instances. In AI parlance, generalization-specialization is also termed *abstraction*, and object-classes can be termed *types* or *sorts*. An example of a generalization-specialization hierarchy might result from the observation that there are two types of aircraft: jet and propeller-driven. This could be modeled with the Aircraft object-class at the root of the inheritance hierarchy having two sub-classes: Jet Aircraft and Propeller Aircraft.

Whole-part The whole-part relationship is a static relationship between two or more object-classes. One object-class (the *whole*) is an aggregation of the other object-class(es) (the *part(s)*). The whole-part relationship is also known as *aggregation*, and the structure of whole-part relationships forms a hierarchy.[3] Continuing our observation of aircraft, we may see that all aircraft have wings, a fuselage, one or more motors, elevators, and a tail-fin. This could be modeled as a whole-part hierarchy with the Aircraft object-class at the whole, and the Wing, Fuselage, Motor, Elevator, and Tail-Fin object-classes as its parts.

Instance The instance relationship is a dynamic[4] relationship between objects, i.e. instances of object-classes. In this document we restrict ourselves to *binary* instance relationships, i.e.

---

[2]For example, by an aircraft spotter on the boundary of a major airport.

[3]Known in manufacturing applications as the Bill of Materials.

[4]That is, dynamic over the time-span of interest. For example, if the time-span is less than one year, then the number of runways at an airport can be regarded as fixed. This is because it takes longer than a year to build a runway. Planning

instance relationships between exactly two objects. The objects may be from the same object-class or from different object-classes. The potential for association between instances is shown as the instance relationship between the parent object-classes. The instance relationship is also known as *association*. The type of association is dependent on the classes of the objects involved. The relationship is true while the objects are associated to one another, and false otherwise. For example, the observation that aircraft take off from runways would lead to the `taking-off-from` relationship between the Aircraft and Runway object-classes. When a specific aircraft is taking off from a particular runway, then this could be modeled by saying that the `taking-off-from` instance relationship holds between the corresponding instances of the Aircraft and Runway object-classes. As soon as the aircraft has become safely airborne, then the `taking-off-from` instance relationship would no longer hold.

Domain invariants

Domain invariants form the third part of the ontology.

In the ontology, domain invariants are inter-relationship constraints. In particular, they constrain instance relationships. Since instance relationships are the elements in modeling object states, domain invariants enforce constraints in state-space.

There are two types of domain invariant that are relevant here:

Cardinality A cardinality invariant constrains an instance of a given type of instance relationship with another instance of the same type of instance relationship. The cardinality constraint may be one-to-one, one-to-many, many-to-one, or many-to-many. For example, an aircraft in flight is always in one or other piece of airspace, but it can only be in one airspace at a time. This cardinality constraint could be represented in the form of a rule such as:

IF `aircraft A flies in airspace X` is **TRUE**
THEN (for all airspaces Y) `aircraft A flies in airspace Y` is **FALSE**

Exclusion An exclusion invariant constrains an instance of a given type of instance relationship with an instance of a different type of instance relationship. For example, an aircraft in flight is always in one or other piece of airspace, but when it has landed at an airport then it cannot be in any piece of airspace. In other words, being on the ground excludes being in flight, and vice versa. This exclusion constraint could be represented along the lines of:

IF `aircraft A flies in airspace X` is **TRUE**
THEN (for all airports P) `aircraft A is landed at airport P` is **FALSE**

Identifying planning and scheduling applications

The procedure for identifying planning and scheduling applications depends on the definition of planning[5] as the construction of a sequence of actions. Since actions and states alternate and states are

---

and scheduling a runway-building programme is not of interest over this time-span. Accordingly, the relationship between runways and airports can be modeled as a whole-part relationship. If the time-span is (say) 10 years, then the instance relationship should be used.

[5]More strictly, plan generation.

modeled as sets of inter-object relationships, this is the equivalent to the construction of a sequence of sets of inter-object relationships.

We observe that states must be sequenced only where the combination of the corresponding sets of inter-object relationships would violate one or more domain invariants. For example, the above-mentioned cardinality constraint enforces an aircraft flying en-route to pass through a series of airspaces in sequence. In order to be (in the state of) flying in airspace X_2 the aircraft must leave airspace X_1. En-route flying would be represented as a sequence of states such as:

(from time t_1 to t_2) [aircraft A flies in airspace X_1]
(from time t_2 to t_3) [aircraft A flies in airspace X_2]

In the aerospace domain, generating such a sequence of states is known as *flight planning*.[6]

Similarly, the above-mentioned exclusion constraint enforces take-off to be represented as a sequence of states along the lines of:

(from time t_0 to t_1) [aircraft A is landed at airport P_1]
(from time t_1 to t_2) [aircraft A flies in airspace X_1]

and landing as the sequence:

(from time t_7 to t_8) [aircraft A flies in airspace X_7]
(from time t_8 to t_9) [aircraft A is landed at airport P_2]

Generating such a sequence of states from the airport's viewpoint is known as *slot planning*.[7]

The procedure for identifying planning applications is as follows:

1. Analyze the domain in terms of objects, relationships, and invariants.

2. Extract all pairs of instance relationships that are constrained by domain invariants.

3. Identify the planning and scheduling application corresponding to each constrained pair of instance relationships, together with the application's owner.

Notation

We illustrate the results of our analysis of the aerospace domain using diagrams in Coad and Yourdon's Object-Oriented Analysis (OOA) notation.

In OOA, object-classes are depicted as labeled rectangles. Rectangles without a dashed surround represent *abstract* object-classes. Rectangles with a dashed surround represent object-classes that can have instances, i.e. *concrete* object-classes. In Figure 6.5 the Airline, Airforce and Aircraft object-classes can have instances, but the Aircraft Operator object-class cannot.

Each rectangle is divided horizontally into three parts. The upper part contains the class name. The middle and lower parts would list the class attributes and methods, respectively. However, domain analysis does not go as far as identifying attributes and methods.

---

[6]From the ATC viewpoint, it is known as flow control.

[7]Strictly speaking, slot planning is concerned with runway slots, because the planner represents the airport in more detail, e.g. as an aggregation of runways, taxiways, parking positions, gates, etc.
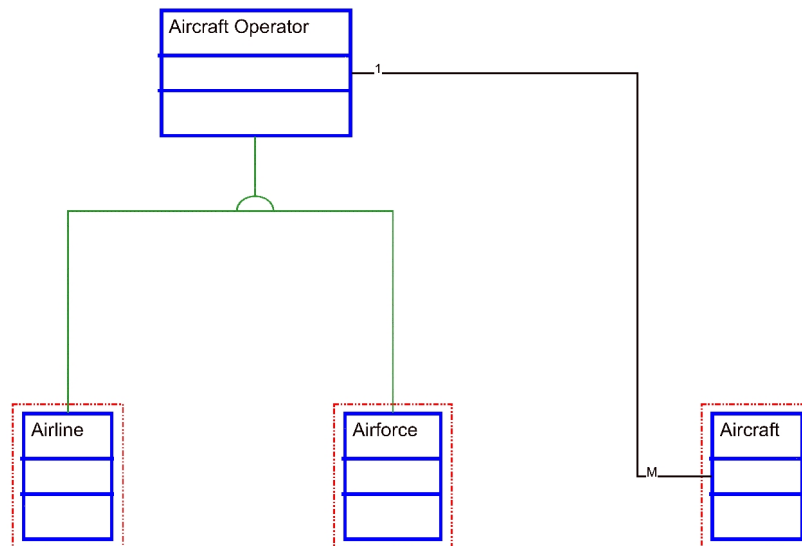
Figure 6.5.: Example of Generalization-Specialization and Instance relationships.

The generalization-specialization relationship is depicted in OOA as a set of lines joining the super- and sub-classes. The super-class has one line leading to a hemisphere, from where one or more additional lines lead to the sub-classes. In Figure 6.5 the Aircraft Operator object-class is a super-class with the Airline and Airforce object-classes as its sub-classes. We say that Airline and Airforce are specializations of Aircraft Operator, in that airlines operate aircraft for civilian purposes and air forces operate aircraft for military purposes.[8]

The instance relationship is depicted in OOA as a simple line joining two object-classes. The ends of the line may be labeled "1" or "M" to denote the "one" and "many" of the cardinality constraint associated with instance relationships. In Figure 6.5 the Aircraft Operator object-class has a one-to-many instance relationship with the Aircraft object-class. The relationship shows that an aircraft operator (i.e. an airline or an air force) can operate many aircraft at the same time, and that each aircraft is operated by a single aircraft operator.

The whole-part relationship is depicted in OOA as a line superimposed by a triangle joining two object-classes. The whole is the object-class with a line leading to the point of the triangle, below which a longer line leads to the part. The ends of the line may also be labeled to denote the cardinality constraint; this is invariably one-to-many with the "1" at the end of the line by the whole. In Figure 6.6 the Airport object-class has the Runway and Terminal object-classes as its parts. This can be read as stating that an Airport is an aggregation of one or more runways and one or more terminals. The Terminal object-class is itself a whole, consisting of an aggregation of one or more Gates.

The Coad and Yourdon OOA notation cannot represent exclusion constraints.

---

[8]They generally operate different types of aircraft, but this is not an essential difference because certain types of aircraft (e.g. transport aircraft) can be operated both by airlines and by air forces.
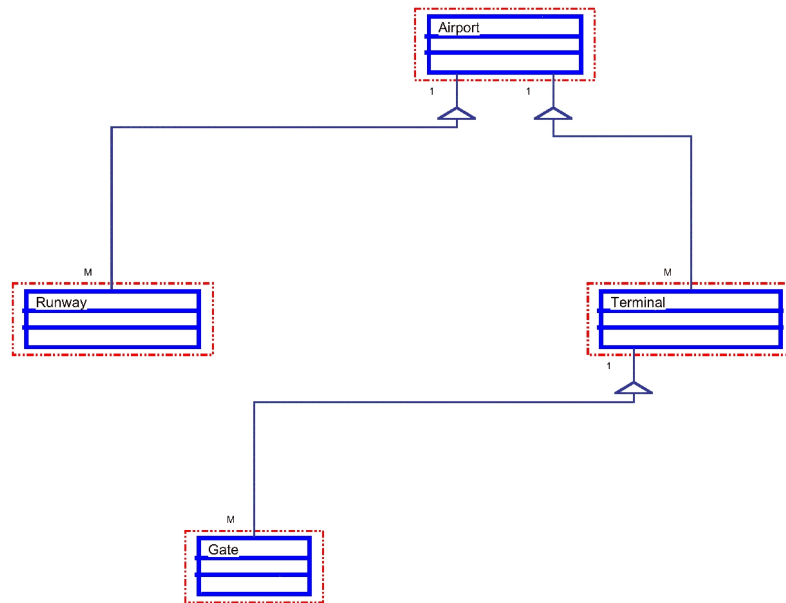
Figure 6.6.: Example of Whole-Part relationship.

Aviation sub-domain

Figure 6.7 depicts the aviation sub-domain objects and relationships we have identified for the purposes of this roadmap.

Object-classes in aviation sub-domain

The object-classes identified in the aviation sub-domain are:

Aircraft Aircraft is the abstract class of vehicles that fly within the Earth's atmosphere. The subclasses of Aircraft are:

> **Manned Aircraft** Manned Aircraft is the subclass of Aircraft that are flown by Aircrew, including unmanned air vehicles (UAVs) that are flown by ground-based Aircrew.

> **Unmanned Aircraft** Unmanned Aircraft is the subclass of Aircraft that are flown automatically.

Airspace Airspace is the class of defined volumes within the Earth's atmosphere fixed with respect to the Earth's surface. The analysis does not include the common distinction between controlled and uncontrolled airspace. This is because the introduction of satellite-based ATC will make all airspace controlled airspace within the time-horizon for this roadmap. Airspace could be subclassed into Airways, Control Zones, and Restricted Airspace, but, for simplicity, this has not been done for this roadmap.

Aircrew Aircrew is the class of persons that fly Manned Aircraft.

Figure 6.7.: Aviation sub-domain objects and relationships.

Aircraft Operator Aircraft Operator is the abstract class of organizations that operate Aircraft. The subclasses of Aircraft Operator are:

> **Airline** Airline is the subclass of Aircraft Operator that performs civil air operations.

> **Airforce** Airforce is the subclass of Aircraft Operator that performs military air operations.

Airport Airport is the class of locations on the Earth's surface where Aircraft can land, taxi, park, and take off. The parts of Airport are:

> **Runway** Runway is the class of locations at an Airport/Spaceport on which Aircraft/Spacecraft can land and take-off.

> **Terminal** Terminal is the class of buildings at an Airport that provide one or more Gates for parking Aircraft. The part of Terminal is:

>> **Gate** Gate is the class of Aircraft-parking locations associated with a Terminal.

Payload Payload is the abstract class of entities that may be carried by an Aircraft/Spacecraft. The subclasses of Payload are:

Pax   Pax ("passengers") is the subclass of Payload that represents passengers carried in Aircraft/Spacecraft.

Freight   Freight is the subclass of Payload that represents physical goods carried in Aircraft/Spacecraft.

Integrated Payload   Integrated Payload is an abstract subclass of Payload modeling complex payloads that must be integrated with Aircraft/Spacecraft systems (and serviced and repaired) by Industry. The subclasses of Integrated Payload are:

    Weapon   Weapon is the subclass of Integrated Payload that represents military weapons carried for the purposes of firing them from the Aircraft/Spacecraft. Aircraft/Spacecraft that carry Weapons are invariably military, and operated by Airforces/Spaceforces. Note that weapons that are merely transported from A to B in Aircraft/Spacecraft would be instances of Freight.

    Sensor   Sensor is the subclass of Integrated Payload that represents sensors (e.g. radar in AWACS, remote sensing payloads, etc.). Aircraft carrying Sensors are usually military and operated by Airforces. (N.B. Navigation-aid calibration aircraft and advertising/broadcasting balloons can be exceptions.) This rule-of-thumb does not apply to Sensor-carrying Spacecraft.

Authority   Authority is the abstract class of regulatory organizations. The subclasses of Authority are:

Aerospace Agency   Aerospace Agency is the subclass of Authority that is concerned with regulating the (aerospace) Industry. Currently, each nation has its own aerospace agency. In addition, there are international aerospace agencies such as the European Space Agency (ESA). In most nations, aerospace agencies cover both the aviation and space domains.

ATC   ATC ("Air Traffic Control") is the subclass of Authority that is concerned with controlling Airspace.

Operating Authority   Operating Authority is the subclass of Authority that regulates Aircraft Operators/Spacecraft Operators and their Aircrew/Astronauts.

Airports Authority   Airports Authority is the subclass of Authority that regulates Airports.

Industry   The (aerospace) Industry is the abstract class of organizations that construct, service and maintain Aircraft/Spacecraft. The subclasses of Industry are:

Manufacturer   Manufacturer is the subclass of Industry that builds Aircraft/Spacecraft and their parts.

Service   Service is the subclass of Industry that services Aircraft/Spacecraft, e.g. provides fuel, towing, baggage handling, electrical power, passenger and freight ground transport, etc.

MRO   MRO ("Maintenance and Repair Organization") is the subclass of Industry that maintains and repairs Aircraft/Spacecraft.

Relationships in aviation sub-domain

By means of the relationships, Figure 6.7 expresses the following statements about the aviation sub-domain:

- Aircraft can fly through Airspace, which is controlled by ATC centers.

- Manned Aircraft are flown by Aircrew, who are employed by Aircraft Operators.

- An Aircraft Operator is either an Airline or an Airforce.

- Aircraft Operators (and their Aircrew) are regulated by an Operating Authority.

- When not flying, Aircraft are on the ground at an Airport. They land on (or take-off from) a Runway, which is part of the Airport, and park at a Gate, which is part of one of the Airport's Terminals.

- Aircraft Operators are based at Airports.

- Airports are regulated by an Airports Authority.

- Aircraft can carry Payloads, which can be passengers (Pax), Freight, Weapons, or Sensors. Weapons and Sensors have to be integrated into the Aircraft.

- The aerospace Industry is regulated by an Aerospace Agency, and consists of Manufacturers, Service, and Maintenance and Repair Organizations (MROs).

- Loading and unloading of Payloads is done at a Terminal, with the handling being done by specialized Service industry, such as ground stewardesses, security staff, customs officials, fork-lift truck drivers, etc.

- Industry manufacturers and repairs Aircraft and their Integrated Payloads.

- Industry also services Aircraft, e.g. towing them, refueling them, replenishing oil and other fluids, providing ground transport, etc.

Space sub-domain

Figure 6.8 depicts the space sub-domain objects and relationships we have identified for the purposes of this roadmap.

Object-classes in space sub-domain

The object-classes identified in the space sub-domain are:

Spacecraft Spacecraft is the abstract class of vehicles that (are designed to) fly outside the Earth's atmosphere. The subclasses of Spacecraft are:

> **Launcher** Launcher is the subclass of Spacecraft that bring Satellites from a planetary surface (or one orbit) into (another) orbit.
>
> **Satellite** Satellite is the abstract subclass of Spacecraft that models spacecraft that are put into orbit by a Launcher. The subclasses of Satellite are:
>
>> **Manned Satellite** Manned Satellite is the subclass of Satellite that can carry Astronauts.
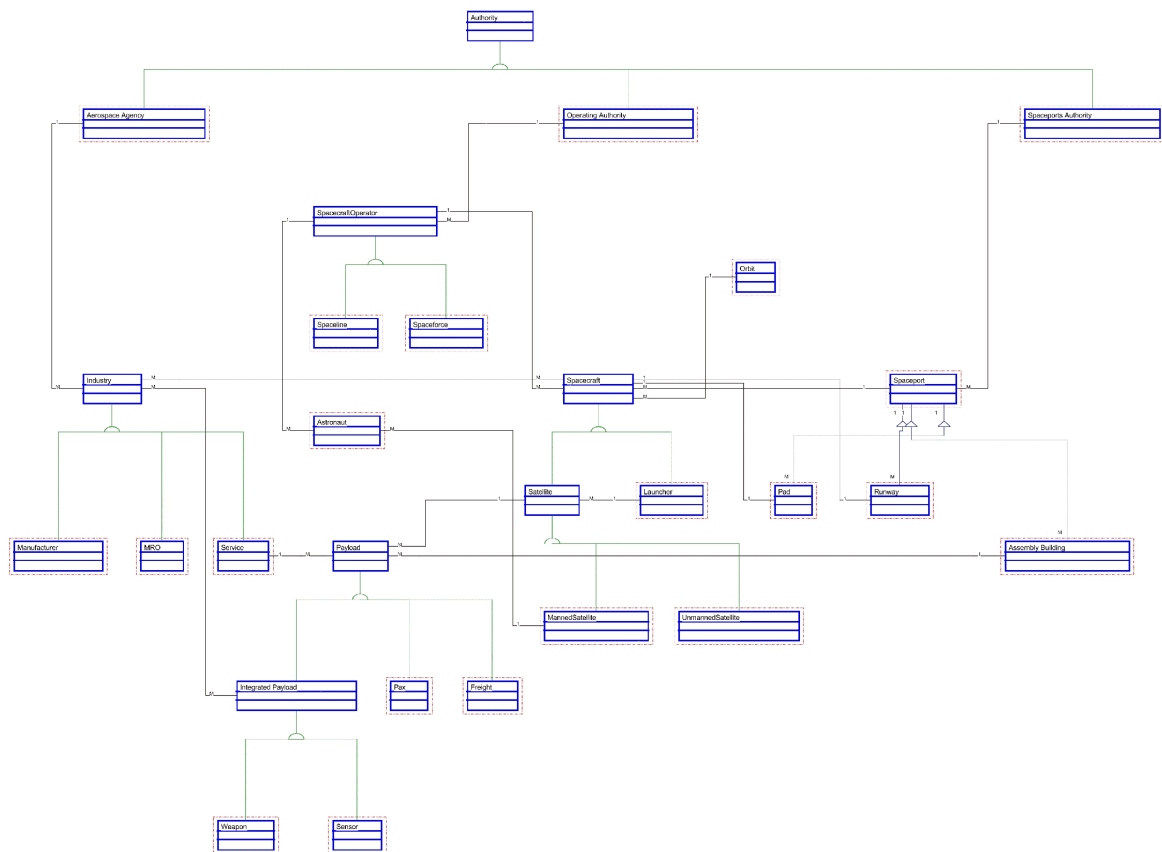
Figure 6.8.: Space sub-domain objects and relationships.

Unmanned Satellite Unmanned Satellite is the subclass of Satellite that does not carry Astronauts.

Orbit Orbit is the class of defined trajectories outside the Earth's atmosphere.

Astronaut Astronaut is the class of persons that fly in Manned Satellites. The class could be subclassed into Pilot and Payload Specialist, but this has not been done for simplicity.

Spacecraft Operator Spacecraft Operator is the abstract class of organizations that operate Spacecraft. The subclasses of Spacecraft Operator are:

Spaceline Spaceline is the subclass of Spacecraft Operator that performs civil space operations.

Spaceforce Spaceforce is the subclass of Spacecraft Operator that performs military space operations.

Spaceport Spaceport is the class of locations on planetary surfaces where Spacecraft are assembled, are launched from, and can land on. The parts of Spaceport are:

Runway Runway is the class of locations at an Airport/Spaceport on which Aircraft/Spacecraft can land and take-off.

Pad Pad is the class of Spacecraft-parking locations.

Assembly Building Assembly Building is the class of buildings at a Spaceport in which Payloads are loaded into Satellites or Satellites are loaded into Launchers.

Payload The subclasses of Payload are:

Pax

Freight

Integrated Payload The subclasses of Integrated Payload are:

Weapon

Sensor

Authority The subclasses of Authority are:

Aerospace Agency

Operating Authority

Spaceports Authority

Industry The subclasses of Industry are:

Manufacturer

Service

MRO

Relationships in space sub-domain

By means of the relationships, Figure 6.8 expresses the following statements about the space sub-domain:

- Spacecraft, which can be Launchers or Satellites, can fly in an Orbit.

- Satellites, which can be Manned or Unmanned Satellites, are launched into orbit by Launchers.

- Manned Satellites are flown by Astronauts.

- Astronauts are employed by Spacecraft Operators.

- A Spacecraft Operator is either a Spaceline or a Spaceforce.

- Spacecraft Operators (and their Astronauts) are regulated by an Operating Authority.

- When not flying, Spacecraft are on the ground at a Spaceport. They are assembled in an Assembly Building, launched from a Pad, and may be able to land on a Runway. Assembly Buildings, Pads and Runways are all parts of a Spaceport.

- Spaceports are regulated by a Spaceports Authority.

- Satellites can carry Payloads, which can be passengers (Pax), Freight, Weapons, or Sensors. Weapons and Sensors have to be integrated into the Spacecraft.

- The aerospace Industry is regulated by an Aerospace Agency, and consists of Manufacturers, Service, and Maintenance and Repair Organizations (MROs).

- Loading and unloading of Payloads into Satellites and of Satellites into Launchers is done in an Assembly Building, with the handling being done by specialized Service industry.

- Industry manufacturers and repairs Spacecraft and their Integrated Payloads.

- Industry also services Spacecraft, e.g. towing them, refueling them, replenishing fluids and other consumables, providing ground transport, etc.

Invariants in space sub-domain

The number of domain constraints is also too large to list in full. Example cardinality constraints are:

- A spacecraft can only be in one orbit at a time, but an orbit can contain several spacecraft.

- Only one launcher can be on a given launch pad at any time.

- A launcher can only be on one pad at a time.

- A runway cannot be occupied by more than one launcher.

- A launcher cannot be on several runways simultaneously.

- A launcher can carry one or more spacecraft (which can themselves be launchers (i.e. upper stages) and/or satellites).

- A spacecraft operator employs one or more astronauts.

- A satellite carries one or more payloads.

Example exclusion invariants are:

- A spacecraft cannot be both in orbit and at a spaceport.

- If a spacecraft is at a spaceport, it is either on a pad, on a runway, or in a bay in an assembly building.

Examples of space AI P&S applications identified from relationship-pairs subjected to 1-to-1 cardinality constraints are:

- Crew scheduling

- Scheduling launchers for take-off on pads

- Scheduling launchers for landing on runways

- Scheduling launchers in assembly bays

There are many examples of capacity-planning applications to be found.

Analogies and differences between sub-domains

The analogies between the classes found in the aviation and space sub-domains are as follows:

- Aircraft and Spacecraft are analogues.

- Airspace and Orbit are analogues.

- Aircrew and Astronaut are analogues.

- Aircraft Operator, Airline and Airforce have as analogues Spacecraft Operator, Spaceline and Spaceforce, respectively.

- Airport and Spaceport are analogues.

- Runway is identically the same in both sub-domains, except that (at present) Spacecraft can only land at a Runway while Aircraft can both land and take-off from a Runway.

- Payload and its subclasses are identically the same in both sub-domains.

- Authority and its Aerospace Agency subclass are identically the same in both sub-domains.

- Operating Authority in the two sub-domains are analogues.

- Airports Authority and Spaceports Authority are analogues.

- Industry and its subclasses are identically the same in both sub-domains.

The differences between the classes found in the aviation and space sub-domains are as follows:

- Aircraft and Spacecraft are analogues, but the inheritance sub-hierarchies below them are different. Both sub-hierarchies include a manned versus unmanned distinction. However, the Spacecraft sub-hierarchy includes the intervening distinction between launchers and satellites, which has no analogue in the aviation sub-domain.

- The space sub-domain has (as yet) no equivalent to the ATC class found in the aviation sub-domain.

- The parts of Airport and Spaceport differ. Although they both have a Runway part, there is no equivalence between the aviation sub-domain's Terminal and Gate classes and the space sub-domain's Assembly Building and Pad classes.

7. Knowledge Engineering

7.1. Introduction

We define Knowledge Engineering (KE) for AI Planning to be the process that involves

1. the acquisition, validation and verification, and maintenance of planning domain models;

2. the selection of appropriate planning machinery and its integration with the domain model to make up a planning application.

By 'domain model' we mean a body of knowledge that an agent can use to make rational deductions about the domain it represents. In particular to planning, it is commonly assumed that a model contains a declarative description of a domain and that the model's most important component is a set of action descriptions. Our first of three workshops on this area produced a set of issues relating to the nature of KE. This drove the initial form of our Roadmap, and determined the agenda for the remaining workshops.

The research communities in the related areas of knowledge acquisition for knowledge-based systems, and requirements engineering in software engineering, are very active and associated with a growing body of literature, methods and techniques. KE for planning may be seen as a special case of these general areas, although planning is separated by the fact that the knowledge elicited is largely knowledge about actions, and how objects are effected by actions. Also, the ultimate use of the planning domain model is to be part of a system involved in the synthetic task of plan construction (in contrast to classification or diagnosis tasks common in KBS). Despite this, we feel that the planning community needs to be better informed about and able to adopt tools and techniques from these general areas.

Our Roadmap examines several topics fundamental to knowledge acquisition and engineering, including:

Roles in the KE process: We have to take into account the differing kinds of stakeholders that have a share in the construction of the planning system. This group may include planning experts, domain engineers, domain experts, software and HCI experts, and eventual end-users.

KE Tools: These are necessary to support the knowledge acquisition method, and the management and validation of the domain model. Interface tools, translation tools and debugging tools are particularly important.

Machine Learning: The use of tools and techniques from the machine learning area to induce or refine parts of a domain model is of great interest. For example, complex action representations may be constructed more reliably by induction than by hand.

Formal Methods: Given a domain model is likely to be a statement in some formal language, experience in the engineering of formal specifications in software engineering is discussed. We consider the similarities between planning domain definitions and formal specification languages.

Although the KE process will involve the use of a variety of representations and models, it may be that at the heart of the model is a formal domain modeling language. In which case, it is important to identify the design criteria for the creation of such languages. In our second workshop, we concluded that such languages should have high level structuring devices, be integrated within a development method, be tool supported, be expressive and customizable. Also, they should to some extent support the operational needs in developing an efficient planning system.

Once we have acquired a domain model, a fundamental question is how to match the available planning technology to the domain at hand? For example, under what conditions would one consider using purely generative planning technology, and under what conditions would it be wise to use case-based technology and a plan library?

The Roadmap points out some general directions and problems to be overcome. These include:

1. there is a general lack of experience in the planning community (especially in Europe) of knowledge engineering concepts and methods, and we have much to learn and techniques to import from related knowledge engineering work.

2. the evaluation of KE tools and methods is problematic, and tends to be harder than the evaluation of, say, a planning algorithm. This is seen as a barrier to future research as researchers find it harder to publish in the planning literature.

3. we need to learn how to characterize domains, and hence build up knowledge about how to match up planning technology with domain models.

The body of this document consists of 8 themes. The first 2 – Domain model Representation Languages, and Knowledge Engineering Support Tools and Environments, are fundamental to the subject. The themes 3 – 8 are related areas, feeding in to the subject of this chapter.

7.1.1. The Knowledge Engineering TCU

Applying emerging software technologies to real problems often involves developing new methods which incorporate these technologies, and developing tools and platforms to support the methods. Whereas the area of "Software Engineering" originated from the need to provide a sound methodological base to the application of software, "Knowledge Engineering" has grown up as the area that provides methods to assist the implementation and maintenance of knowledge-based systems (KBS). Hence the field of knowledge engineering in AI Planning deals with activities involved in implementing and maintaining the knowledge-based aspects of planning systems. This includes the

acquisition, validation and verification, and maintenance of planning domain models (the latter are formalized, abstract models that an agent can use to make rational deductions about the application domain the model represents). It also includes the selection of appropriate planning machinery and its integration with the domain model to make up a planning application.

The Knowledge Engineering for Planning Technical Co-ordination Unit (KE TCU) of PLANET has been active for the past 5 years. It aims to identify and explore the current major problems involved in developing knowledge-based planning systems, and to examine the potential of future developments (such as the Semantic Web) to contribute to solutions of these problems. We also aim to synthesize related research and techniques from work in more general research areas into a relevant contribution to this area.

To achieve these aims the TCU is active in organizing workshops and sponsoring cross-site visits in the subject. Further, we have created and are maintaining the following Roadmap Document which summarizes past work, and includes a collection of activities and problems that need to be tackled for the future. The Roadmap distinguishes Planning from the more general field of KBS in that the knowledge elicited is largely knowledge about actions, and how objects are effected by actions. Also, the ultimate use of the planning domain model is to be part of a system involved in the synthetic task of plan construction (in contrast to classification or diagnosis tasks common in KBS). Our Roadmap examines topics inherited from these areas, and attempts to point out where we can learn from past work; it also contains other related research areas such as Machine Learning, KE Tools and Formal Methods.

Looking towards the future, we see the areas of knowledge sharing through the use of ontologies, and the development of the Semantic Net as very important to support KE in Planning. As more and more planning technology finds its way into applications, knowledge engineering issues are recognized as crucial to an application's success. The KE TCU therefore covers fundamental issues which are relevant to all of the application-oriented TCUs.

7.1.2. Knowledge engineering in AI P&S

By its very nature AI Planning is knowledge-based: planning involves the manipulation of knowledge of complex phenomena, such as actions themselves. The scope of knowledge associated with a planning application is called the *domain*, while the symbolic representation of this is called the *domain model*. Planning programs input goals or directives, and, using a domain model, find some ordering of actions to achieve them.

This chapter concerns Knowledge Engineering (KE) in AI Planning. It is the process that deals with the acquisition, validation and maintenance of planning domain models, and the selection and optimization of appropriate planning machinery to work on them. Hence, knowledge engineering processes support the planning process: they comprise all of the off-line, knowledge-based aspects of planning that are to do with the application being built, and any on-line processes that cause changes in the planner's domain model.

The main characteristic of a *domain model* is that it is possible for an agent to use one to make rational deductions about the domain it represents. In particular to planning, it is commonly assumed that a model contains a declarative description of domain dynamics and that the model's most important component is a set of action descriptions. The conventional wisdom is that a declarative model should

be developed to a certain extent independently of the planning engine and any other software that will form the rest of the application. This tends to ease the process of validating the domain model. It also gives the stakeholders and developers more flexibility in the use of their product and lessens the investment risk; an independent domain model may be used with a range of general planning engines, and may be used in many other ways not necessarily connected to planning.

Acquisition of a domain model may involve a prolonged analysis of the application domain by knowledge engineers, using structured interviews with stakeholders, system manuals, existing models, software documentation etc. Before acquisition can take place, however, the engineer must find out what kind of planning problem is to be solved by the system. This entails asking questions such as: Is the problem one of scheduling or planning, or a combination of the two? Can actions be modeled deterministically? Do they have to be modeled with duration? Do we have to model the resource they consume? Can we assume the planner has complete knowledge of the world? What are the kind of plans required? What is the relation between planning and execution? What are the the optimization criteria for plans? Hertzberg in section 3 of reference [Her96] declares a similar list of questions in his discussion of the characteristics of application domains.

The results of this early analysis is fundamental to and implicit in the kind of domain modeling language that will be used to capture the domain model, as well as critical in the choice of planning algorithm. Once the type of planning application has been established, and environmental assumptions are known, the process of domain modeling itself can begin. The knowledge acquired tends to fall into two categories:

Domain structure and dynamics: The identification of the relevant objects and object classes in the domain, their properties and relations (predicates), and constraints among these predicates. Dynamic knowledge is dependent on the environmental assumptions, but typically involves specifying how these predicates change truth value as actions are executed.

Domain heuristics: General approximate rules and the general applicability of techniques that have been found to help in plan generation, where their use leads to a speed-up in planning and/or an increase in the quality of the output plans.

From an engineering point of view, it is essential to be able to **validate** the results of domain acquisition. Validation of a domain model is the process that promotes its quality in terms of internal and external criteria by the identification and removal of errors in the model. Internal criteria include properties such as syntactic correctness and logical consistency; in general these properties can be proved formally and are not problematic. External criteria include properties such as accuracy, correctness and completeness. Given that the sources of the model will not often be a mathematical object, these properties can never be proved correct (in the same sense that a requirements specification can never be proved correct). Note the distinction between validation of a domain model and validation of a planning system. The former supports the latter, and occurs at a much earlier stage in system development.

The agents involved in the knowledge engineering process

There are several kinds of roles that have to be filled in the technical side of the knowledge engineering process: planning experts, domain engineers, domain experts, software and HCI experts, and

end users. Often a person may have more than one role; a researcher in academia may have to fill every role. Assuming there is a distinguished formal system within which the domain model is being encoded, it seems necessary to have several kinds of "interface languages" to this formalism to suit participants in different roles. You would not want, for example, a user to help in the static validation of a domain model by reading through a complex logic formalism.

The bigger picture

Knowledge acquisition for knowledge-based systems and requirements engineering in software engineering are very active areas in Computer Science, and are associated with a growing body of literature, methods and techniques. They are related areas in a number of respects. In particular, one can consider that the results of knowledge acquisition and requirements engineering both involve some kind of domain model [SG96]. Using our general definition above, KE for planning may be seen as a special case of these general areas. Hence it may prove useful to derive methods and adapt tools from these areas, although there are peculiarities of planning that clearly distinguish engineering planning knowledge from them:

- The knowledge elicited in planning is largely knowledge about actions and how objects are effected by actions. This knowledge has to be adequate in content (and ultimately in form) to allow efficient automated reasoning and plan construction. In contrast, knowledge elicited about processes/actions in traditional software engineering tends to be done with the purpose of helping in the analysis and understanding of a system, and to be used in the forming of a specification of a new system.

- The ultimate use of the planning domain model is to be part of a system involved in the "synthetic" task of plan construction. This makes it very specific in the world of Knowledge-based Systems (KBS), where many successful systems are, in contrast, aimed at solving diagnostic or classification problems.

Despite the difference of purpose, adopting tools and techniques from these general areas seems likely to be a good strategy. For example, the insights gained from the use and development of methods such as KADS [ABvH94], and the use of requirements modeling languages and methods in software engineering (e.g. [GMB94]) need to be used when developing KE methods for planning applications.

The number of planning projects which have exploited KE tools and techniques is growing rapidly. Some examples of planning projects that have addressed or are addressing the issues of KE and/or knowledge representation are as follows:

- O-Plan, SPAR and <I-N-OVA>: Edinburgh/ARPI projects that tackle plan representation and KB planning (e.g. see [Tat98, TPJ98])

- The Planform project, aimed to create a simple knowledge engineering tools environment (e.g. see [Pla])

- The SIPE-2 system, featuring advanced HCI tools (e.g. see [MW97])

- The multi-mission Vicar Planner applied to automated image processing

- The Remote Agent Experiment, the control of NASA's Deep Space 1 by an AI planner.

Experience in applied work in planning suggests that the outputs of the KE process, such as "activity descriptions", are a vital core concept for many types of reasoning beyond planning. KE tools, methods and languages need to be suited to the much wider role of capturing knowledge of activity whether or not planning is involved in guiding and directing those activities towards purposeful outcomes.

7.2. Roadmap themes

7.2.1. Domain model representation languages

State of the art and current research

The kinds of knowledge that need to be captured in order to build a planning application include knowledge about actions, goals, activities, time and resources. So called 'classical' planning has adopted a propositional state-based approach to knowledge representation, where states of the world are modeled by statements in logic, and actions are modeled by transformations between these states. Currently the research community's standard for communicating domain models is PDDL [GHK+98, FL03]. This was not, however, designed from a knowledge engineering perspective [McC03]. In the sections below we discuss what kind of requirements exist for practical representation languages, and introduce alternatives to traditional propositional encodings.

Requirements for a representation language

How can we provide better formalisms for capturing and expressing the required knowledge? Evidence from the KBS and Requirements Engineering Community suggests that a domain modeling language should:

- Be structured. It should provide mechanisms that allow complex actions, complex states and complex objects to be broken down into manageable and maintainable units. For example, the dynamic state of a planning application could be broken down into the dynamic state associated with each object. On this structure can then be hung ways of checking the model for internal consistency and completeness.

- Be associated with a workflow model, or method. This will give a set of ordered steps to be carried out in order to capture the domain model, thus guiding the knowledge engineer throughout the process. It could contain activities such as modeling of state changes and the discharging of proof obligations.

- Support the operational aspects of the model. The language's framework should include a set of properties and metrics which can be evaluated to assess a model's operationality and likely efficiency. It should be possible to predict whether the model can be translated to an efficient application.

- Be tool supported. These tools will support the steps in the method and, using the structure of the model language, be able to provide powerful support for statically validating, analyzing and operationalizing the model.

- Be expressive and customizable. The language needs to be generally applicable, yet customizable in some sense so that it "fits" well with applications. Since there is a whole range of assumptions involved in planning which may or may not hold in an application (e.g. to do with uncertainty, resources, closed world) it may be that the modeling language will have "variants" to deal with different assumptions.

- Have a clear syntax and semantics. As a basis for the other aspects above, and for the analysis of models encoded in the language, it should be possible to map models to a "meaning" within some well-known formal system such as a modal logic.

It seems that no modeling language fulfills all these criteria. Languages that have been developed from the point of view of knowledge acquisition include DDL.1 [CO96], TF [TPJ98] and OCL [Pla]. The most commonly used domain description language, PDDL [GHK$^+$98], appears to fare badly according to these criteria. PDDL, however, has the potential to be easily extensible and at the same time widely accepted and used within the planning community.

The domain representation: propositional vs. analogical

In addressing the question of how to provide better formalisms for capturing and expressing the required knowledge, the previous section proposed a list of generic requirements for a representation language. According to such requirements, an ideal domain-modeling language should be expressive and customizable, yet be able to produce an *efficient* encoding of the problem domain. This section takes a more critical and constructive approach, by (1) arguing that most of the current planning modeling formalisms produce rather inefficient encodings of realistically complex domains, and by (2) actually showing *how* more efficient formalisms can be developed.

In particular, in the field of knowledge representation two main types of modeling paradigms have been identified: the widely used *propositional* (or 'sentential' [LS87]) representation, and the so-called *analogical* (or model-based) representation [BF81]. This section illustrates how the main inefficiencies of current planning representations (namely, the frame and ramification problems) arise from the use of purely propositional representations, and claims that such problems can be overcome through the adoption of analogical planning domain descriptions.

Problems associated with propositional languages   For many years, researchers in AI have regarded the frame problem [MH69] as presenting a major difficulty for reasoning about action [Geo87, Sha97]. The frame problem (in essence, having to specify, for each action, which properties of the world remain *unaffected* by its execution) can be seen as the result of adopting a specific knowledge representation paradigm, and was addressed in STRIPS by requiring that the representation of an action (operator) explicitly listed only those propositions that *are* affected by the actual execution of the represented action, while those that are unaffected are simply assumed to persist [Lif86]. This assumption, still lying at the basis of modern planning domain description languages [FL03], allows

the frame problem to be avoided, but does not address a second, equally serious representational issue: the so-called *ramification* problem [Geo87]. John Pollock [Pol98] accurately describes this problem as one that

> "*arises from the observation that in realistically complex environments, we cannot formulate* axioms *that* completely specify *the effects of actions or events.* [...] [I]*n the real world, all actions have infinitely many ramifications stretching into the indefinite future. This is a problem for reasoning about change deductively* [...]" [Pol98, p.536]

To illustrate the significance and impact of this problem in planning, let us consider a variation of the familiar Blocks-World (BW) domain, in which the robot arm can lift towers of m blocks, with $m \leq h \in N$. In order to decide which blocks may be lifted in a given state, it should be possible to augment the problem description with the two following domain axioms:

$$\forall x, y : \mathtt{on}(x,y) \rightarrow \mathtt{above}(x,y,1) \tag{7.1}$$

$$\forall x, z \ (\exists y, n : \mathtt{on}(x,y) \wedge \mathtt{above}(y,z,n) \rightarrow \mathtt{above}(x,z,n+1)) \tag{7.2}$$

The condition "$\mathtt{clear}(x) \wedge \mathtt{above}(x,y,n) \wedge (n \leq h)$" would then be sufficient to guarantee that 'y' is the lowest block of a 'liftable' tower. Unfortunately, no one has yet provided a semantics for domain axioms containing numeric fluents, such as those shown above. Most importantly, given a certain world state (described using exclusively '$\mathtt{on}(\ )$' expressions), deducing the '$\mathtt{above}(x,y,n)$' relations for all the possible blocks requires a number of axiom applications that grows exponentially in the size (arity) of the axioms. More precisely, if 'o' is the number of objects (constant symbols) of the domain and 'r' is the arity of the axioms, the number of deductions required is in the order of $k \cdot o^r$, for a given constant k.

Because of this, extending planners to deal with domain axioms seems likely to involve significant additional computational costs. Consider, for instance, how a forward state-space planner could solve a problem in the above BW domain with axioms (7.1) and (7.2): whenever the truth value of any of the '$\mathtt{on}(\ )$' propositions changes, it will be necessary for the planner to re-calculate all of the '$\mathtt{above}(\ )$' relations, as the truth of some of them will have been affected by the change. A backward-search planner would incur in similar problems. Consider, for example, the process of determining how to achieve the goal (or precondition) '$\mathtt{above}(x,y,j)$', in which the variables x and y are still unbound (i.e., how to build a tower of $j+1$ blocks, for a given j, using any set of blocks). Transforming this expression into the equivalent formula "$\mathtt{on}(x,z_1) \wedge \mathtt{on}(z_1,z_2) \wedge \ldots \wedge \mathtt{on}(z_{j-1},y)$" (which can then be achieved using the standard BW operators) requires a rather complex process of reasoning, analogous to that used in the resolution procedure of a Prolog interpreter. In addition, in a backward-chaining search, the presence of axioms would significantly increase the branching factor.

In view of these difficulties, several researchers (e.g., [Gar00, GK97, DG02, THN03]) have been exploring the possibility of *preprocessing* planning problems and compiling axioms away in the domain description. However, recent theoretical results [THN03] show that this approach leads (in general) to exponentially (or polynomially, if the arity of the axioms is kept constant) longer domain descriptions and *plans*. Experimental evidence suggests that the resulting performances can be even worse than those obtained with planners that are able to deal with domain axioms internally [THN03].

To summarize, the ramification problem and the presence of domain axioms add a significant computational overhead to the planning process. This may become severe or even unacceptable if planning is to be carried out in realistically-complex scenarios, rather than in simple toy domains. In fact, when reasoning about (i.e., simulating) action in real-world domains, innumerable physical properties of the world should be taken into account, such as sound and light propagation, temperature, gravity, fluid and gas dynamics. Using Pollock's original example, among the effects of striking a match we must include such things as "displacing air around the match, marginally depleting the ozone layer, raising the temperature of the earth's atmosphere, marginally illuminating Alpha Centauri, making that match unavailable for future use, etc." [Pol98, p.537]. In realistic domains, the number and complexity of the axioms quickly become overwhelming, making planning very difficult a task.

It is contended here that the problem of domain axioms and that of frame axioms (requiring, respectively, the explicit specification of all properties that are - and that are not - affected by action) are, ultimately, two facets of the same *representational* problem, which lies in the use of purely "propositional" (or 'sentential', or 'Fregean' [Kul94]) domain description languages.

The term 'propositional' is used here as opposed to 'model-based' [HV91], or 'analogical' [Slo75, Dre81], or 'direct' which the current world state is described by a set of Well-Formed Formulæ (sentences) of a formal language (e.g., predicate logic). While a propositional representation can be said to *describe* the domain represented, an analogical representation rather *models* it [Kul94]. In particular, in model-based (or analogical) representations the world is modeled by data structures (and relations on them) which mirror the properties of and relations between the objects of the represented domain (in the next section a more detailed definition is provided).

Being purely descriptive, propositional planning languages (such as STRIPS, ADL and PDDL) are quite flexible and expressive (e.g., see [FL03]). Yet, because of their descriptive nature, they require all (physical) properties and constraints of the world (even the most trivial, such as the fact that an object cannot be moved on top of itself) to be represented 'extrinsically' [Pal78], i.e., to be explicitly imposed on the model via the addition of supplementary elements (in the form of formulæ and axioms). As seen earlier, this can quickly lead to a combinatorial explosion in the number of (trivial) inferences that must be explicitly represented and carried out – namely, to the frame and ramification problems. Hence, although propositional languages are widely applicable and may be used to model most aspects of the real world, they tend to produce representations that are *inefficient* for domains that involve a large number of distinct entities subject to (even simple) physical constraints.

In order to overcome these problems, more adequate planning domain description languages are needed. The author of this section (Max Garagnani) maintains that one of the ways in which new, more efficient (yet expressive) representations can be developed lies in the introduction of analogical formalisms, and in their use in conjuction with propositional descriptions.

Planning with analogical representations   Planning in realistic domains is closely related to the problem of common-sense reasoning [McC58]. In this context, several researchers (e.g., [Kul94, Lev92, LS87, KR96]) have argued for the need of formalisms that allow a more direct (or 'vivid') representation than current sentential descriptions. In particular, analogical and diagrammatic representations (briefly introduced earlier) have long been of interest to the knowledge representation community (e.g., [Slo75, Hay74, MK92, Lin95] – see [Kul94] for a useful account). In order to better characterize such representations, let us introduce a simple example of analogical representation,

which will be used throughout the rest of this section.

Consider a representation of the Blocks World domain in which the state is modeled as a set of lists of characters. Each list denotes a (possibly empty) stack of blocks, and each character represents a block. For instance, in a four-block problem, the state S = {[A,B,C],[D],[ ],[ ]} would correspond to the propositional description {on(C,B), on(B,A), clear(C), clear(D)}, in which predicates have their standard meaning (empty lists denote empty spaces on the table). The specification (using an appropriate syntax) of a generic operation for transforming legal states into legal states (consisting of removing the last character of a non-empty list and appending it to another - possibly empty - list) completes the description of the domain model.

Using this simple example of analogical representation for the (one-operator) BW domain it is already possible to identify the main features that differentiate such representations from propositional ones.

In a propositional representation the objects of the domain are represented as constant symbols (e.g., A, B, C,... etc.). The relevant relationships (like 'on' and 'above') between objects are described as sets of associations (relational instances) between such symbols, and are identified using other symbols (predicates) of the language (e.g., on(B,A), above(C,A)). The specific syntax chosen to build such formulæ, however, has no bearing to the properties of the represented world. In particular, the properties of the relations that exist between the objects of the domain and their interactions are *imposed* on the formal model through the specification of axioms and logical rules (e.g., axioms (7.1) and (7.2)).

In contrast, in an analogical representation the objects of the domain and the relationships that exist between them are not described by sets of relational instances, but modeled using appropriate data structures. The *syntax* of such data structures mirrors, for the relevant aspects, the semantics (relations and properties) of the problem domain [BF81]. In particular, the relations between elements of the representing structures of the model and the corresponding represented relations of the domain have the same algebraic structure (using Palmer's term, they are "naturally isomorphic" [Pal78]).

Consider the BW domain example. The analogical, list-based representation consists of a set of formulæ having the following syntax:

$$[arg_1, arg_2, \ldots, arg_n]$$

This syntax clearly reflects the semantics of BW: the first character of the list (arg_1) always represents the lowest block of a stack, while two consecutive characters arg_k, arg_{k+1} indicate that block arg_{k+1} is on block arg_k. The rightmost character of a list denotes a 'clear' block. In contrast, the formulæ used in the propositional representation adopt the following generic syntax:

$$predicate(arg_1, arg_2, \ldots, arg_n)$$

This syntax has no direct relation with the structure and properties of the BW domain.

In addition, consider the spatial relation "above", represented in the model by the relation "to the right of", defined on the (linearly ordered) characters of a list[1]. The *transitive* property of the relationship

---

[1]More precisely, the block denoted by character *x* is *above* the one denoted by *y* iff character *x* appears to the right of *y* (in the same list).

"above" (originally imposed on the model through the addition of axioms (7.1) and (7.2)) is an implicit property of the relation "to the right of", and does not need to be explicitly imposed or accounted for during the reasoning process. In other words, the latter relation and the represented spatial relation "above" are naturally isomorphic [Pal78].

Thanks to the above features, analogical descriptions can implicitly embody *constraints* that sentential representations would normally have to make explicit: the relevant properties of the relations existing between the objects of the domain are *inherent* to the structure of the representing relations, and do not need to be explicitly imposed on the model or included in the description.

Because of the implicit, unalterable structure of the relations which they employ, however, analogical representations tend to be less flexible and general than propositional ones. Nevertheless, the presence of such "inherent constraints" [Pal78] reduces the computational complexity of inference, avoiding the combinatorial explosion of trivial deductions that would have to be explicitly represented in sentential reasoning systems, and easing the frame and ramification problems (cf. [Lin95]). In the BW example, given a certain state description, the problem of deducing – using axioms (7.1) and (7.2) – whether a tower is 'liftable' becomes one of simply checking the current model, by counting the number of characters that follow a given one in a list. This check can be carried out in time *linear* in the number of blocks. In other words, thanks to the model-based nature of the representation, the computationally expensive theorem-proving aspects of the planning process can be replaced by simple model-checking (see also [HV91, KR96]).

By allowing simpler and more efficient methods of common-sense reasoning, analogical representations also make possible new, easier heuristic extraction, abstraction, and learning techniques. The data structures that form the model of the domain can be quickly and effectively checked for the presence of specific (syntactical) structures (or 'patterns') of objects; the recurrence of such patterns can then be used as a basis for macro-operator learning, abstraction, or evaluation metrics. For example, consider the difficulty of determining the existence of identical stacks of blocks in two different BW states: in a list-based representation, simple pattern-matching will quickly lead an answer, while a propositional encoding would involve a significantly more complex procedure.

Naturally, the simple list-based representation used for the BW domain example is not very expressive, and much more general representations are needed for modeling realistic problems. One way to obtain more expressive languages is to develop more 'abstract' and general data structures and formalisms for analogical representations, which can be used to model a wider variety of domains. This is the approach adopted by Garagnani in [Gar03], where a more expressive representation for physical domains is presented, based on the abstract structure of 'SetGraph'. This formalism can adequately and efficiently represent domains that involve object manipulations, movements and changes of state, and is at least as expressive as the classical STRIPS language. However, like STRIPS, this formalism would need to be further extended in order to be able to represent realistically complex domains, allowing such features as conditional effects, durative actions, numerical quantities and functions.

A second promising direction for developing new formalisms consists of integrating analogical and propositional representations within a single 'hybrid' paradigm. Early work on the integration of model-based and sentential representations can be found in [MK92, Ham93]. For example, Myers and Konolige [MK92] proposed a formal framework for combining analogical and deductive reasoning. In their system, the inference rules of a sentential module (based on a logical first-order

predicate language) were used to reason about and 'complement' the information contained in analogical (diagrammatic) structures, which represented the state of the world. However, the system did not allow the representation of actions for transforming the current world state.

An object-centered domain-description language (OCL) has been recently proposed by Liu and McCluskey in [LM00] (see also [Pla]), based on the idea of modeling a domain as a set of objects subject to various constraints. The idea of an object-based representation is clearly in line with a model-based approach. However, OCL is fully sentential; consequently, some of the constraints which would be implicit in an analogical representation still need to be explicitly declared (e.g., the fact that if an agent holds an object of a certain type, the location of the object must coincide with that of the agent). Nevertheless, OCL seems likely to represent a good candidate language to support the use of analogical representations in conjunction with propositional descriptions.

Open issues

The choice of which representation formalism to adopt in order to capture and express the required knowledge is clearly of vital importance for KE in planning. Research in AI has since long demonstrated that the type of representation adopted plays a fundamental role in determining the difficulty of problem solving (e.g., see [Ama68, GNC95, Sim81]).

Research goals

The community should try to produce 'standard' domain model representation languages for knowledge-based planning (short term goal).

Actions for the planning community

The introduction of model-based and analogical representations appears to be a promising approach for addressing the ramification problem, which arises when modeling realistically-complex domains. Hence, the recommended tasks for the research community are:

- investigate analogical, model-based and diagrammatic representation tools, techniques and methods that may be relevant to planning;

- attempt to implement more expressive and universally applicable analogical representations, providing a clear syntax and semantics of the language;

- build on existing work on the integration of analogical and propositional formalisms to realize *hybrid* planning-domain description languages (this may also be achieved by extending current propositional representations with analogical features, and vice versa).

7.2.2. Knowledge engineering support tools and environments

State of the art and current research

Tool support   To create a non-trivial domain model a team needs tool support. Naturally, the amount and coverage of tool support required depends on the size and nature of the application. Because AI Planning has been largely in the realm of research, most existing domain models have been built using nothing more than basic syntax checkers. Researchers have tended to "debug" their domain model through dynamic testing, using the failure of plan generation or plan execution to show the presence of, and identify, domain errors. For an application requiring a larger domain model, this approach appears at best inefficient.

For knowledge acquisition, tools are needed to support interviews with domain experts and encoding of their knowledge. Machine Learning tools that for example induce operator descriptions from traces of actions may also be used. A further possibility is to provide an interface to the model's formal language that allows a domain expert to directly encode planning knowledge. This could well be more efficient than having to employ a planning expert and knowledge engineer, and may preserve in the user the feeling of control and ownership of the problem.

Developing domain models in isolation from a planning engine means that we can split the process of knowledge acquisition into user inspection, static validation, and dynamic validation (animation). In user inspection, an appropriate tool would be one that maps back and forth between a user-friendly, diagrammatic language and the domain model language itself. This kind of tool would be similar to a graphical front-end to a formal specification language.

Tools for static validation essentially perform domain analysis: They reason with the model to check that it is self-consistent, to check that it is complete (in a restricted sense) and to output consequences of the model that might be useful for user inspection. For example, tools may check that an operator is consistent (it never inputs a valid state and outputs an invalid state); or reason with operators and output state invariants to be visually checked by a user; or output necessary goal orderings, to check for impossible goal combinations and help in dynamic testing.

Dynamic validation entails acquiring a set of test cases and associated (optimal) plans and using these to test the functioning of the planner as well as the validity of the model. Generated plans can be compared against expected plans in an attempt to identify errors in the domain model.

Both the O-Plan and SIPE projects have developed tools that help in the knowledge engineering process. With O-Plan we have the "Common Process Method" [Pol99, TPJ98]. With SIPE we have the Act Editor [MW97]. Both are essentially sophisticated directed graph presentation tools, i.e. they let the developer enter domain knowledge, and examine the nodes in a HTN operator and the temporal constraints between them. Deficiencies of the O-Plan/SIPE visual tools is that (1) they provide no visualization or checking for the state based model that the user builds around the operators. There is no definition of what properties the operators available for refining a task must hold nor the transitions that a domain object can pass through or the states that it can exist in. (2) There has been no evaluation of these tools to determine if they help domain model development and to what extent. (3) None of the tools are appropriate for use by domain experts without a computer background.

Given the importance of engineering methods in other disciplines, one can argue that the development of *knowledge engineering methods* for AI Planning, based around a set of tool support, is desirable. However, it is difficult to to evaluate and benchmark knowledge engineering methods. It appears very expensive in terms of time and effort to carry out case studies, and even then we only have the results of the case. It appears that the only other way is to compare a method to existing methods in similar domains, or base a KE for planning method on an existing one e.g. a planning-oriented form of KADS. It could be argued that the general problem of method evaluation tends to make researchers avoid looking into KE as it is harder to publish (given the problems of evaluation of the research).

Supporting large scale planning knowledge bases   Effective tools for supporting knowledge engineers in constructing large-scale planning knowledge bases are at last emerging from the research community. The goal of this section is to draw on the experience of an experienced knowledge engineer (Peter Jarvis of SRI) to encourage promising directions, dissuade a couple of approaches, and add an important class of knowledge engineering tasks to the research agenda.

Keep providing structure   "The knowledge bases that I work with contain of the order of 100 - 200 action schemata ranging over 5 abstraction levels and are described using a conventional hierarchical task network formalism. Using action schemata to structure a domain model results in an opaque description where domain properties are scattered, duplicated, and frequently implicit.

Emerging modeling approaches that encourage a central description of each object in a domain together with the state collections that each can occupy should be encouraged. It is encouraging to see new structural notions emerging that allow properties of action abstraction levels to be specified.

To exploit these developments in real domains I need them to work with existing applied planning engines. I currently have to balance the benefits of an improved modeling framework against the cost of loosing the broad inference range and efficiency of the engines I work with."

Please, no more cosmetic structured editors   "Simple structured editors have often been produced to answer client knowledge engineering concerns. They typically provide attractive interfaces that graph the components of action schemata to show their ordering constraints and provide tree views for navigating through action abstraction layers. While these tools provide useful viewing aids, they still force domain descriptions to be organized around action schemata. The next time you see one of these tools put forward as the solution to the planning knowledge engineering problem, I encourage you to ask about the automatic consistency checks that the tool provides and the classes of error that they uncover."

Planning knowledge bases need support over a long lifespan   "The succinct representation of domain properties and design decisions together with automated consistency checking tools become even more important as a knowledge base ages. I give three examples of the classes of knowledge base change that I frequently make to encourage the inclusion of model evolution on the research agenda.

The simplest class of change is the addition of a new method for refining a task (we have drive and fly, add take-train). Most of the errors I introduce when making changes of this class center on me neglecting to model for the new method some transformation obligation imposed by its parent.

A more complex (but almost as frequent) class is significant structural changes to a knowledge base. For example, a stand-alone knowledge base that covers military air operations might need to be moved to become a component of a larger knowledge bases that covers air and ground operations. These changes often demand considerable changes in the underlying conceptualization of the domain that ripples through all the action schemata in a knowledge base.

Finally, the representational devices used to describe a domain will evolve as the underlying planning engine is refined and enhanced. The domain descriptions that I support frequently have had to undergo significant structure changes and continue to run on several versions of a planning engine."

Domain analysis   With present planning technology, finding heuristics and domain control knowledge to improve planning efficiency and plan quality is an important aspect of the knowledge acquisition process. Planning systems receive such knowledge in two different ways: Either it is given as input along with the problem specification, in which case it is often called advice, or a planning system employs *domain analysis* (DA), *i.e.* automatic knowledge discovery tools.

The knowledge found by domain analysis, commonly called *domain knowledge*, lies in between specification and advice. Briefly, it consists of statements about a planning problem that are logically implied by the problem specification, but that are not part of the specification. Furthermore, domain knowledge should be "planner independent", *i.e.* not closely tied to the internal workings of any particular planning system, but such a requirement is difficult to formulate precisely.

The importance of domain analysis results from a rich structure commonly "hidden" in the domain description. Instead of letting a domain expert analyze this structure by hand and formulate this knowledge (or let a machine learning system discover it), it is often possible to find it automatically. In short, "if a machine can do something for you, don't do it yourself!".

DA techniques can aid planning and knowledge engineering for planning in several ways:

Planning speed-up and plan quality improvement: This has been the focus of research in domain analysis so far. It can be done either "off-line", i.e. only once for each domain, or "on-line", i.e. for every problem instance, depending on the DA technique.

Model validation: Static analysis can aid in the internal validation of a domain description, e.g. to find state invariants for user inspection, and to analyze the effects of applying hand-coded control knowledge.

Matching planning technology with domains: Domain analysis deals with structural features of planning domains and problems, and thus it can help in choosing the right planner and/or the right control knowledge for this planner in an automatic way.

There is a large variety of domain analysis techniques described in the literature, but most of them are integrated with a specific planning system and are not available as separate modules. A reason for this is that domain analysis does not directly produce control knowledge: It is only in combination with knowledge of the planning algorithm that domain knowledge can be effectively used for

control. Nevertheless, most DA techniques are useful for more than one planner and hence should be recognized a separate modules.

TIM [FL98, FL99] finds types and state invariants, as do DISCOPLAN [GS98] and others [Rin00]. RIFO [NDK97] removes irrelevant facts and operator instantiations, while RSA [Sch99] and RedOp [HJ00] find different types of constraints on what action sequences are necessary or relevant for solving a given problem. Detection of symmetry [FL99, CGLR96] and goal ordering [KH00] can also speed up planning.

Some DA techniques find knowledge which is useful in combination with particular planning algorithms, e.g. STATIC [Etz93] or Alpine [Kno94], others are helpful for a class of them, e.g. TOP [Sch99] for total-order planners. Several planners have some analysis preprocessing step, e.g. the graph construction with mutexes in Graphplan [BF97] or precomputation of heuristics [RV99, BG99].

The use of domain analysis tools as separate modules requires a language to state the resulting knowledge. One attempt in this direction is the domain knowledge expression language DKEL [HS03]. The use of DKEL is demonstrated by the testbed for planning systems [VSD02], which supports the quick construction and evaluation of planning systems from modules.

The extraction of properties such as types and invariants can be a way to automatically characterize planning domains and problems [LF99]. It can also be used to detect subproblems, e.g. a TSP or shortest-path problem, embedded in a planning problem [LF00]. Another use of domain analysis is reasoning about the planning process. An example of this is the planner HAP [VTV02], whose planning strategy is adjusted according to the existence and characteristic of domain properties, which are in part found by domain analysis.

Planning ontologies   There are several attempts to create planning ontologies traditional applications. One of them is the Shared Planning and Activity Representation (SPAR) project,[2] which is part of the DARPA/Air Force Research Laboratory (Rome) Planning Initiative (ARPI).[3] SPAR [Tat98] is based on the assumption that it is important that information about processes, plans and activities is able to be shared within and across organizations. Cooperation and coordination of the planning, monitoring and workflows of the organizations can be assisted by having a clear shared model of what comprises plans, processes and activities. SPAR is intended to contribute to a range of purposes including domain modeling, plan generation, plan analysis, plan case capture, plan communication and behavior modeling. By having a shared model of what constitutes a plan, process or activity, organizational knowledge can be harnessed and used effectively

SPAR has been a contributing source towards the development of the Core Plan Representation (CPR).[4] CPR is a model that expresses information common to many plan, process, and activity models. The goal is to leverage common functionality and facilitate the reuse and sharing of information between a variety of planning and control systems. The CPR embodies a standard that is general enough to cover a spectrum of domains from planning and process management to workflow and activity models. The representation supports complex, hierarchical plan structures. The initial application of the CPR is in addressing plan interchange requirements of several military planning

---

[2]http://www.aiai.ed.ac.uk/~arpi/spar/
[3]http://arpi.isx.com/
[4]http://www.teknowledge.com/CPR2/

systems, but the model goes beyond military planning and presents a more general plan representation.

Another effort for defining a planning ontology is is the Process Specification Language (PSL), a neutral representation for manufacturing processes developed by NIST (US National Institute of Standards and Technology).[5] Process data is used throughout the life cycle of a product, from early indications of manufacturing process flagged during design, through process planning, validation, production scheduling and control. In addition, the notion of process also underlies the entire manufacturing cycle, coordinating the workflow within engineering and shop floor manufacturing. The goal of PSL is to create a process interchange language that is common to all manufacturing applications, generic enough to be decoupled from any given application, and robust enough to be able to represent the necessary process information for any given application. This representation would facilitate communication among the various applications because they would all have a common understanding of concepts to be shared.

PLANET is an ontology developed at Information Sciences Institute (ISI) by Yolanda Gil and Jim Blythe, within the DARPA High Performance Knowledge Bases (HPKB) program [GB00]. PLANET is a reusable ontology for representing plans that is designed to accommodate a diverse range of real-world plans, both manually and automatically created. PLANET makes the following representational commitments to provide broad coverage: First, planning contexts that refer to domain information and constraints that form the background of a planning problem are represented explicitly. The same happens for planning problems and relevant alternative plans for each problem. Second, PLANET maintains an explicit distinction between external constraints (e.g. user advice or preferences) and commitments, which the planning agent elects to add as a partial specification of a plan.

The Enterprise Ontology [UKSZ98] has been developed by the Artificial Intelligence Applications Institute at the University of Edinburgh, within the Enterprise Project, the UK government's major initiative to promote the use of knowledge-based systems in enterprise modeling, aiming to support organizations effectively in the Management of Change.[6] The Enterprise Ontology is a collection of terms and definitions relevant to business enterprises, which are based on the definition of Activities, Organizations, Strategies, Marketing and Time. There are also some additional efforts to define planning ontologies, as for example the Planning Ontology Construction Group (POCG) of the ARPA/Rome Laboratory Planning Initiative (ARPI) or the Planning Ontology Project by John Doyle;[7] however additional information for these works is difficult to find.

The numerous independent efforts to develop ontologies for planning applications reveal the importance and the difficulty of this ambitious goal. PDDL has established a widely accepted syntactic framework for defining planning domains and problems, however its lack of semantic annotation makes it difficult for creating/merging/using ontologies. PDDL extensions like the Web-PDDL or the DAML family seem to constitute the natural next step towards a clear, unambiguous, expressive and decidable planning language.

---

[5]http://ats.nist.gov/psl/

[6]Refer to http://www.aiai.ed.ac.uk/project/enterprise/
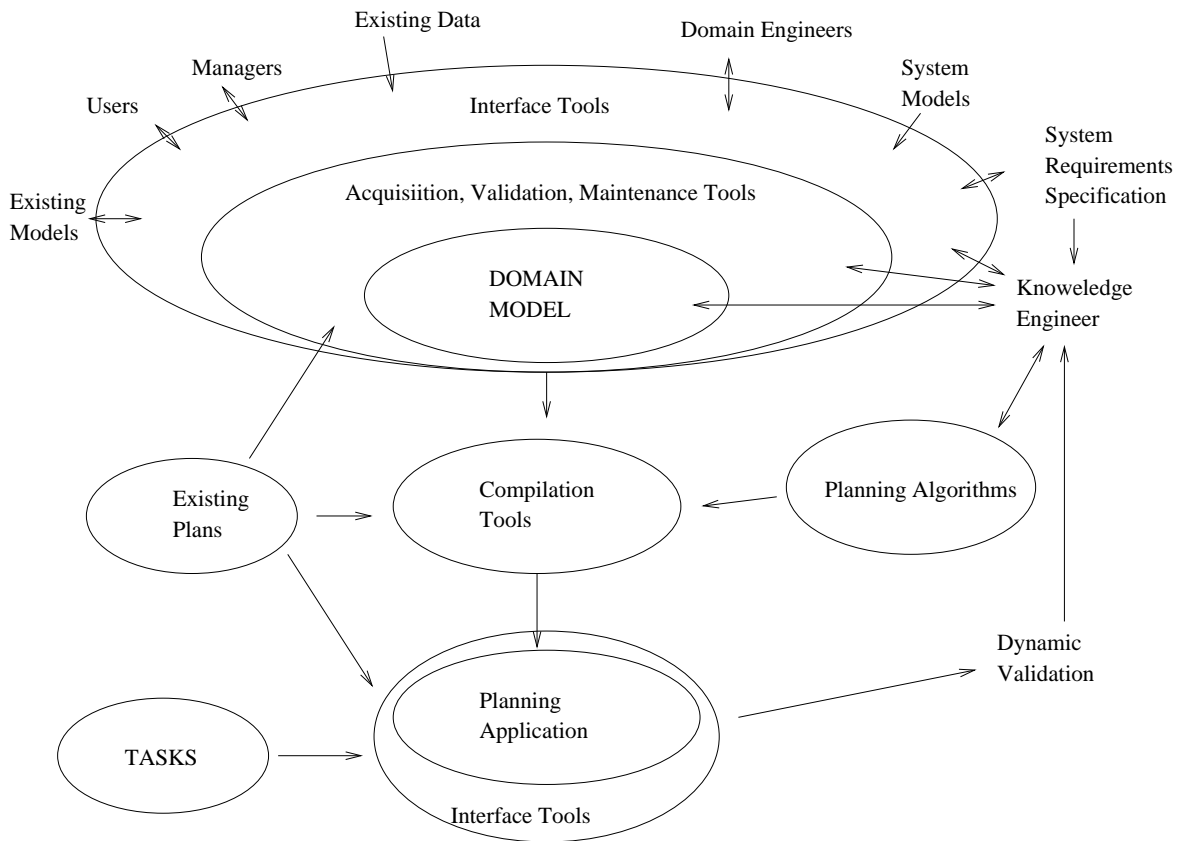
[7]http://www.kr.org/doyle/

Figure 7.1.: An Idealized Planning Knowledge Engineering Environment

A KE support environment   After having discussed the nature of KE, and the kinds of tools likely to be found in a KE environment, we now use Figure 7.1 to show the kind of architecture that integrates these artifacts. This "idealized planning KE environment" was inspired by the Planform project proposal [Pla], although of course it could be changed to other topologies. It may be, for example, that the interface tools to the domain model could be assumed to be interfacing to all aspects of the system, in which case "Planning Application" would be at the core of the environment.

Note that we concentrate here on the knowledge-based aspects of the environment; as the application of planning technology will generally be a complex task, it seems necessary that issues of configuration management, version control etc must also be addressed. These and other vital concerns relating specifically to software engineering and project management will not be considered further, as they lie outside the "knowledge-based" concerns.

Users, Managers and Domain Engineers may want to add and adjust knowledge, apply measurements to the model, inspect the model, animate the model (using a simple planner) or explore the truth of properties of the model.

Existing data and models may form a substantial part of the planning system, and will need a customized acquisition tool to convert it into the internal format.

The results of previous analysis and feasibility studies may prove useful - hence information from

known system models and from the project's requirements specification will influence the development of the planner.

Existing plans are a database of the kind of plans expected from the final planner. Hence they can be used in may ways: For initial knowledge acquisition, as a source of heuristics for the planner, and as validation of the final planner.

So that knowledge is not "hand coded" into the planning architecture, and to avoid problems in maintenance of the system, it seems sensible to keep separate algorithmic, heuristic and fundamental domain knowledge until these can be fused into a planning application. Ideally, this fusion will be performed by a compilation tool. This tool would, using aspects of the domain, evaluate to find out which planning technology was most appropriate and from this build up a final planning application.

After an initial knowledge acquisition and static validation phase, the compiler can be used to produce an application on which traditional dynamic testing can begin.

Actions for the planning community

Tool support

- Support for maintenance: Carry out research into the kinds of tools that would be useful in maintaining large Planning knowledge bases.

- Structured representation languages: Promote the inclusion of *structural* features of planning description languages as a necessary precondition for their use as domain modeling languages.

- Visualization tools: Carry out research studies to evaluate the effectiveness of currently available visualization tools, and derive lessons that can be used in the design of future tools. Aim to design visual tools that draw on the whole of the domain model (objects, object hierarchies, invariants, states, as well as actions) and help in checking self consistency, accuracy and error removal.

- Evaluation of knowledge engineering methods: Develop models with which to to evaluate and benchmark knowledge engineering methods.

Domain analysis   According to the points above, the planning community should

- Decouple domain analysis techniques from planning engines, so as to enable arbitrarily combining different DA techniques and planning engines. For this to be possible, the domain description formalism that is input to the planning engine must support expression of domain knowledge.

- Analyze hand-coded control knowledge in use and design automatic tools to find as much of that knowledge as possible.

Domain ontologies   We identify the following actions that the planning community should consider:

- Active participation in the standardization process of semantically rich languages, in order to ensure that they will support all the planning-related necessary information, as e.g. invariants, control knowledge etc. Existing languages, such as PDDL, should either be extended, or replaced by the new standards.

- Development and standardization of ontologies, both for the Semantic-Web domain and for the traditional ones.

- Support to semantically rich representation languages by the modern planning systems.

- Focus to the application of the planning technology to the Semantic web domain, either for information extraction, or for task accomplishment.

7.2.3.   Knowledge-based systems

State of the art and current research directions

In the last 20 years or so there has been a paradigm shift in the field of KBS. This was the transformation from the old idea of 'knowledge transfer', where constructing a KBS amounted to extracting the knowledge from experts and encoding it within an expert system 'shell'. This shift re-focused the field onto 'domain modeling' - with the consequent emphasis on the building of a deep causal model prior to an operational system. The domain model has to embody not just the procedural expert knowledge but the environment in which this knowledge is to be utilized. Several 'modeling frameworks' have been developed (e.g. CommonKads [WSB92] or for more recent work see reference [BKRG01]). These support the process of model acquisition and validation, and are underpinned by an overall method of development. CommonKads in particular advocates the use of a series of models during domain capture, each dealing with different aspects of the domain.

An underlying aspiration of recent KBS research is that of *knowledge sharing and re-use*, built on the assumption that it is neither effective nor efficient to consider building a KBS from scratch. The 'knowledge level' view has become prevalent in the literature [New82] and emphasizes the idea of implementation-independent, and hence more re-usable knowledge bases. Many aspects of KBS are common, in particular common sense knowledge and common strategies for problem solving. KBS researchers have concentrated on several aspects of this need:

Reusing/sharing of procedural knowledge: The formulation and re-use of generic problem solving strategies and knowledge bases lies at the heart of the solution to the 'knowledge acquisition bottleneck' in KBS. Problem Solving methods (PSMs) were developed, encouraging the growth of catalogs of 'generic methods'.

Reusing/sharing of declarative knowledge: The emergence of standard interfaces and language conventions for KBS is leading to the possibility of standardization and interoperability.

Another recent development has been the movement in the KBS community to make domain modeling more of a rigorous exercise [vHF95]. Capturing a domain in a model is not just about capturing the 'dynamics': the validity of descriptions (properties/relations) of objects is of crucial significance. This equates to validating that the 'possible worlds' in a model co-incides with actual possible worlds in the domain, as far as possible. This is a very complex KE issue often overlooked in the development of knowledge-sparse models. Attempting to address the validation issue requires separating out the validation process using the *structure* of a domain modeling language. The accepted wisdom from the field of Formal Methods is to capture the structure, and in particular the *invariants* of a domain in a formal language. A state that satisfies these structural restriction is called 'valid'. Operations are then captured (often using a similar notation to planning). A key verification task is then to check that the initial state observes the invariants, and that if any operation is executing in a valid state the output of the operation is valid. This precipitates an inductive proof that all generated states are valid. It has the benefits that (a) it provides documentation for the developers (b) it helps eliminate errors early development (c) it helps developers analyze the model in greater depth, and helps to reflect the domain being modeled.

Actions for the planning community

In planning, the development of application ontologies and planning ontologies is leading to the possibility of planning component standardization and interoperability. Efforts to develop an ontology for plans can facilitate the interoperability of plan manipulation tools (e.g. using the PLANET ontology [GB00] or SPAR [Tat98]). What is required is a review of the KBS knowledge engineering literature, and the creation of a catalog of tools and techniques that may be relevant to the AI Planning area. Important questions that such a review should answer include: Are these tools actually in use outside the laboratory? How have these tools been evaluated? What lessons can the Planning community learn from the methodology for research and development in related areas?

7.2.4. Semantic web and AI planning

State of the art and current research directions

Traditionally, AI Planning has focused mainly on specialized, non every-day life applications, such as robot navigation, aerospace applications, logistics management, workflow management etc. Although the benefits from the application of AI Planning technology in these areas were of great importance, the situation has great prospects to change in the next few years, due mainly to two reasons:

- the wider acceptance of e-commerce as the main way to purchase goods, and

- the gradual transformation of the traditional Web to the richer Semantic Web.

W3C, the World Wide Web Consortium[8] defines the Semantic Web as: *". . . the abstract representation of data on the World Wide Web, based on the RDF standards and other standards to be de-*

---

[8] http://www.w3.org/

fined."[9] A parallel definition is given by Lee, Hendler and Lassila [BLHL01]: *"The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation."*

The Semantic Web constitutes a new application domain for planning technology, having the dynamics of both to advance this technology and simultaneously to increase its recognizability and acceptability by the general public. This will lead to an increased demand for high quality planning and scheduling services, which will lead to increased financing of the planning and scheduling research, thus recurrently boosting the progress in this discipline.

The planning problem in the context of the Semantic Web usually has the following form: Given:

- a set of Web Services, i.e. agents on the Web that provide services (usually information gathering or on-line transactions) to other agents, by exposing an interface containing actions definitions, and

- a goal to be achieved, like e.g. get specific information or achieving a specific arrangement,

find a partial ordered set of actions, i.e. interactions with web services, which ensures the achievement of the imposed goal.

There are several technical problems that arise in the context of Semantic Web. These concern both knowledge representation and planning algorithms. Concerning knowledge representation, there is an imperative need for a widely-accepted web-services description language, in order to be feasible for a planning agent to exploit the full range of available resources in the web. Having the experience of the boost that the lisp-like Planning Domain Definition Language (PDDL, [GHK+98]), as well as the international planning competitions (in the conferences AIPS-98, AIPS-00 and AIPS-02) gave to the planning research in traditional domains during the last years, it is easy to predict how valuable a widely accepted language for describing Semantic Web domains would be.

However, there is a major difference between web-enabled languages and PDDL. PDDL is a syntactic language, which only defines the way operators and states have to be written. There is no special care for the atomic terms that are used to name predicates and actions. Actually, PDDL lacks its semantic counterpart. However, this was not a problem, since in most domains where PDDL has been used, the closed world assumption is supposed. Even in cases where an open world is assumed, this only concerns the number of the objects.

On the other hand, the Semantic Web is an open world. It is not the number of objects (i.e. agents) that is practically infinite, but also the number of concepts which continuously increases. New web services arise, providing for new actions that have to be absorbed and used by the planning agents. So, in order for a planning agent to absorb the new actions, understand their meaning, their prerequisites and their effects, a common vocabulary has to be defined, giving conceptual meaning in each atomic term and establishing conceptual relations among them. Usually, this is achieved with the use of ontologies, i.e. concept hierarchies, which are attached to Web documents thus giving to them semantic meaning. Of course, we are still too far from the adoption of the One Global Ontology, which will cover every aspect of our knowledge, however numerous ontologies have established,

---

[9] http://www.w3.org/2001/sw/

accepted and used by specific target communities, whereas preliminary attempts for automatic or semiautomatic correspondence or merging of ontologies have been undertaken.

The open world assumption also affects most of the modern planners. Actually, most of the planners that work by exhaustively enumerating all the ground facts and actions (e.g. heuristic and Graph-plan/Satplan planners) have to adapt to the Semantic Web domain. Moreover, the need for parallel plans, optimized with respect to several criteria (including response time and overall cost) is of great importance, in order for the planning technology to be adopted by the general public.

In the following we review the current status in defining languages for describing web services and the attempts to construct planners that work on the web.

Web languages   The Semantic Web differs from the traditional Web in that data are accompanied not only by formating annotation (as in the case of HTML) but also by conceptual characterizations. So, web-based agents have the ability to understand, reason and finally plan for their own goals. Semantic Web languages include, among others, XML, RDF, RDFS (information on these is available at `http://www.w3.org/`), SHOE[10] and DAML+OIL [MFHS02].

The XML (Extensible Markup Language) was originally designed for large-scale electronic publishing; however it plays an important role in data exchange over the Web. XML aims mainly at storing, carrying and exchanging data. It is characterized by a very simple but strict syntax, which facilitates the creation of software that manipulate XML documents. Moreover, XML Namespaces provide a means of semantic characterization of the information stored in XML documents, by identifying element and attribute names and solving name conflicts.

The RDF (Resource Description Framework) complements XML in that it provides a means for cataloging data in the web for effective and efficient retrieval. RDF files use triplets of the form ¡Resource, Property, Value¿ to characterize web resources, usually XML documents, with specific properties. RDF properties may be thought of as attributes of resources and in this sense correspond to traditional attribute-value pairs. RDF properties may also represent relationships between resources. RDF however, provides no mechanisms for describing these properties, nor does it provide any mechanisms for describing the relationships between these properties and other resources. That is the role of the RDF vocabulary description language, RDF Schema (RDFS). RDF Schema defines classes and properties that may be used to describe classes, properties and other resources. So, RDFS is recognizable as an ontology language. The combination of RDF/RDFS allows applications to be programmed with a standard method to process meta data and exchange information without worrying about interoperability.

However, RDFS is not a suitable foundation for the Semantic Web, since it is too weak to describe resources in sufficient detail. Three inefficiencies of the RDF/RDFS combination are the following ones:

- they are not formally specified

- they do not have adequate expressive power

- they do not provide automated reasoning support

---

[10]`http://www.cs.umd.edu/projects/plus/SHOE/`

The above inefficiencies restrict the chances of automated planning systems to fully exploit the opportunities of the Semantic Web. Several languages have developed to meet the above requirements. Some of them are mentioned in the following paragraphs.

SHOE[11] is a simple HTML Ontology extension, which allows web page authors to annotate their web documents with machine-readable knowledge. SHOE claims to make real intelligent agent software on the web possible. SHOE is no longer extended, since SHOE researchers have been targeted into OWL and DAML+OIL web ontology languages.

OIL (Ontology Interface Layer)[12] is a joint standard for specifying and exchanging ontologies, developed within the OntoKnowledge Project.[13] It is a proposal for a web-based representation and inference layer for ontologies, which combines the widely used modeling primitives from frame-based languages with the formal semantics and reasoning services provided by description logics. It is compatible with RDF Schema (RDFS), and includes a precise semantics for describing term meanings (and thus also for describing implied information).

The DAML language (DARPA Agent Markup Language)[14] is being developed as an extension to XML and the Resource Description Framework (RDF), as an effort to focus on the eventual creation of a web logic language. DAML-ONT,[15] released in 2000, was the initial ontology core of DAML, roughly corresponding to a frame-based / description logic starting place, allowing the definition of classes and subclasses, their properties, and a set of restrictions thereon.

DAML and OIL efforts were merged to produce DAML+OIL language[16] a semantic markup language for Web resources. It builds on earlier W3C standards such as RDF and RDF Schema, and extends these languages with richer modeling primitives. DAML+OIL provides modeling primitives commonly found in frame-based languages and values from XML Schema datatypes. The language has a clean and well defined semantics. DAML+OIL is going to become a W3C standard, under the new name OWL (Web Ontology Language).[17]

Of special interest for the planning technology is the description of web-services within the ontologies. Such services, like e.g. booking a hotel room or ordering a meal, constitute executable actions, the counterpart of the information gathering actions. DAML-S (S stands for Services)[18] is a markup language for describing Web services, jointly developed by ISI, BBN, CMU, Nokia, SRI and Yale. DAML-S builds on top of DAML and supplies Web service providers with a core set of markup language constructs for describing the properties and capabilities of their Web services in unambiguous, computer-interpretable form. DAML-S markup of Web services intends to facilitate the automation of Web service tasks including automated Web service discovery, execution, interoperation, composition and execution monitoring. DAML-S is supported by an ontology editor,[19] with a graphical and form-entry user interface, which allows for defining composite services based on simpler ones. Finally, a prototype tool for automated synthesis of composite web services has been implemented.

---

[11]http://www.cs.umd.edu/projects/plus/SHOE/
[12]http://www.ontoknowledge.org/oil/
[13]http://www.ontoknowledge.org/
[14]http://www.daml.org/
[15]http://www.daml.org/2000/10/daml-ont.html
[16]http://www.daml.org/2001/03/daml+oil-index.html
[17]http://www.w3.org/TR/webont-req/
[18]http://www.daml.org/services/
[19]http://www.ksl.stanford.edu/projects/daml/

A parallel approach for defining a web service ontology is WSDL,[20] submitted to W3C in 2001. WSDL is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information. The operations and messages are described abstractly, and then bound to a concrete network protocol and message format to define an endpoint. Related concrete endpoints are combined into abstract endpoints (services). WSDL is extensible to allow description of endpoints and their messages regardless of what message formats or network protocols are used to communicate.

A very interesting effort, aiming to provide a test-bed for experiments with intelligent agents and web-services is AgentCities.[21] It is about a worldwide initiative designed to help realize the commercial and research potential of agent based applications by constructing a worldwide, open network of platforms hosting diverse agent based services. The ultimate aim is to enable the dynamic, intelligent and autonomous composition of services to achieve user and business goals, thereby creating compound services to address changing needs. The initiative aims to build on several innovative technologies including agent technology, Semantic Web technologies, UDDI discovery services, eBusiness standards and Grid Computing. Potential application areas range from eHealth and eLearning to manufacturing control, digital libraries, travel and entertainment services.

Actions for the planning community

The Semantic Web domain adds another agent in the KE team: that of the web site developer. Now the development of a site needs to take into account not only the human-computer interface, but also the agent-to-agent one. In other words, the web site developer has to create two (usually identical) interfaces, the second one of them describing the services exposed by the site, using one of the languages mentioned in the previous paragraphs.

The questions that arise from the above presentation, with respect to the standard AI planning domain languages such as PDDL, are obvious: Is there any need for another language describing web information and web services? Is it worth extending languages like PDDL in order to cover web applications? Or are existing languages, like DAML, OWL or WSDL, sufficient for planning applications? The answer seems to be somewhere in the middle.

Drew McDermott introduced Web-PDDL, a Web version of PDDL which supports XML namespaces and flexible notations for axioms. Web-PDDL uses Lisp-like syntax and is a strongly typed first-order logic language. Moreover, a translator between Web-PDDL and DAML, named PDDAML, is provided.[22] Similar translators between DAML-S and PDDL are provided.[23] These works show that there is no great semantic difference between traditional AI planning languages like PDDL and new web-service languages, so it is possible for these two areas to converge.

---

[20]http://www.w3.org/TR/wsdl
[21]http://www.agentcities.org/
[22]http://cs-www.cs.yale.edu/homes/dvm/daml/
[23]http://www.ksl.stanford.edu/projects/daml/

7.2.5.   Formal methods in software engineering

State of the art and current research directions

Formal Methods in software engineering covers (1) the capture and analysis of a (formal) specification of software within a structured formal language, and (2) the refinement of a formal specification into an efficient implementation. The formal specification language has to be appropriate to the application at hand, be sufficiently abstract to allow formal reasoning, and be well supported with a tools environment. The primary concerns in formal methods is to show that the initial specification is internally consistent and externally valid, and to prove that the derived implementation is correct with respect to the specification. These processes are meant to improve the quality of the software process and product, as they are aimed at the early identification and removal of bugs. Superficially at least, there is a strong similarity between formal specification languages and planning languages. Both kinds of languages are designed to allow engineers represent actions precisely and declaratively. Take VDM-SL (the Vienna Development Method's Specification Language [Jon90]) and a STRIPS-language. Both are based around the notion of a state, allow the developer to create operators, and in both cases those operators are defined using pre- and postconditions. Further, they are both based on the assumptions of closed world, default persistence and instantaneous operator execution. VDM encourages the creation of state invariants for validity and documentation purposes; state invariants are also used in some planning languages, but the main rationale here seems to be plan generation speed-up.

The difference between those using formal specification to describe systems and those using a planning language to model a planning domain, is that in the former case the specification is used as a blueprint for design, whereas in the latter case the specification is used as input to a planner to be reasoned with in order to construct plans to achieve goals. States in languages such as VDM-SL and Z [PST91] are built up from mathematical data types such as sets, mappings, sequences etc. With the exception of work in deductive planning [BS96], much of the work carried out in planning research assumes little or no structure to types (predicates are often assumed to be "function free").

For an introduction to the algebraic and model-based formal specification one can consult reference [TM94]. This reference also describes how a Tweak-like planner can be be represented in VDM-SL, how its plan-space operations can be specified using VDM operations, and how the whole specification can be rigorously translated into a logic program.

Actions for the planning community

We can learn from the development of formal methods. One particularly successful use of a formal system in AI Planning and Scheduling is in the use of Petri Nets (e.g. [MYK98]). It is agreed that, in general, the take up of Formal Methods in mainstream software development has been problematic, mainly for the reason that the formalisms are not understandable to most stakeholders in the system being developed. In the same way, we would not want to let our users look at pieces of ADL or scrutinize HTN operators! One of the approaches being pursued in formal methods is the use of "methods integration", that is using an informal, graphical front-end method to allow non-mathematicians to develop the specification. Languages such as Z and methods such as OMT have been fused this way [AS94]. After initial informal development, tools translate the diagrammatic

language to a formal specification language. A formalist then fills in more details of the specification (the informal language invariably leads to under-specified action schemata) and the result is (ideally) mapped back to the friendly front end for further validation. This kind of methods integration may well help open up the use of formal planning domain descriptions languages such as PDDL to more widespread use and acceptance.

7.2.6. Machine learning

State of the art and current research directions

Introduction   In the AI research literature, there are many systems that use "learning" tools in the development of a planner, or within the planning product itself. This is an attractive area from the symbolic machine learning (ML) point of view, as planning involves the acquisition of knowledge and high level cognitive skills. Planning programs can then be used as the performance components in evaluating learning techniques. ML techniques, on the other hand, can help planning in several ways. Here, we will only give a short summary. A comprehensive state of the art can be found in [ZK03].

Knowledge acquisition: ML techniques could be used to remove the bottleneck of eliciting knowledge in much the same way as ML techniques have been used in front-ends for expert systems. For example, it may be more useful to induce the symbolic specification of complex actions than trying to get an expert to describe them in formal terms. For instance, in reference [CG90] experimentation is used in the domain to extract domain knowledge, while in [Wan94] a domain expert agent uses observation with the same purpose, and in [GMB98] domain actions are learned from an autonomous agent moving around in a robotic domain. More recently, similar techniques have been applied to learning methods for HTN planners and tested in a simplified NEO domain [IDNA02]. A recent trend is to combine ML and interaction with the user/expert, within a KE graphical environment [MRS02, GTR01].

Model validation: Complex models can be incorrect or incomplete. If classified training data is available then, for example, a theory revision tool could be used to help identify and remove errors [MM01a, MRS02, MM01b].

Improving the quality of plans: Finding plans of good or optimal quality is very important and it is known that this is more complex than just finding valid plans. Here "quality" is any metric applied to a plan: number of operators in the plan, economic cost of executing the plan, time required to execute it, , etc. Examples of tools that learn knowledge to augment particular planners so that they output quality plans are QUALITY [Pér95], Hamlet [BV97], and Scope [EM97]. Instead of learning control knowledge, other researchers learn rewriting rules to transform low quality plans into high quality ones [AKM00].

Improving planning efficiency: It is well known that solving even medium sized planning problems is a hard task for domain independent planning. A well researched way of improving planner efficiency is by using ML to learn control knowledge. This will be considered in the next section.

Exploiting ML for Planning Speed-up   Perhaps the most popular application of ML to planning is in plan generation speed-up, where learning concerns itself with improving planning efficiency (decreasing resources in time and space). Acquiring heuristics for any knowledge-based task is as difficult as acquiring the model's dynamics and structure. In the case of planning, to hand-code heuristics the user would need to know how the planning engine works to be able to define correctly those heuristics. Therefore, ML techniques have the potential to overcome this knowledge acquisition bottleneck by automatically learning those heuristics, with the possibility of presenting them to the user to refine or validate them. There have been many approaches, for example:

Macro-operator learning: Macro-operators are sequences of planning operators that have been compiled into a form that can be easily retrieved to form a partial solution to a planning problem. Early work concentrated on macros to help in general problem solving [Kor85, PK86]. In the FM system [MP97] macros were used to make problem solving in STRIPS-robots domains highly efficient.

Analogy/Case Based Reasoning: This approach uses planning problems that have already been solved to guide the solution of similar problems e.g. Chef [Ham86], Analogy [Vel94].

Learning sequences of subgoals: It is also possible to learn sequences of subgoals as stepping stones in the planning process. For example SteppingStone [RK89], EAS [RK92] (which is also CBR based).

Learning heuristics: Planning is usually seen as a search process, therefore it is possible to use heuristics to guide this process. Techniques belong to three main categories:

- **Trace based "deductive" systems** are similar to DA techniques in that they use the domain description, but in addition rely on a few traces of the planner after solving planning problems to create control knowledge for that domain. Most of them use EBL Prodigy-EBL [Min88], ULS [CZP$^+$89], for total order planners [MB87], and SNLP+EBL [KK94] for partial order planners. There are also hybrid systems, like DERSNLP+EBL [IK96] that combines CBR and EBL. FM [T.L88] used a deductive technique to create the hypothesis space for the conditions of a heuristic that was further refined using induction as more traces became available.

- **Trace-based systems can be purely inductive**, relying mostly on planning traces to build control knowledge. They generalize from the specific traces to build planning control knowledge, examples are given in references [Min88, Lec93, EM92].

- **Multi-strategy systems:**. Inductive systems require many examples and if they are incremental, depend on the ordering of learning instances [Ket94]. On the other hand, deductive systems need completely correct theories, usually produce knowledge that is too specific, and are prone to the utility problem. Multi-strategy systems combine deductive and inductive approaches to overcome these limitations (FM [McC87], AxA-EBL [Coh90], Hamlet [BV97], SCOPE [EM97], EvoCK [ABI02]).

Learning policies: Instead of learning control knowledge for a planner, some researchers learn policies, that determine which operator should be applied at any given planning situation (this is reminiscent of universal planning) [Kha99, MG00].

Learning constraints: nowadays, a new generation of very fast planning systems based on propositional representations/constraint satisfaction have come to the fore. It is more difficult to improve the efficiency of these planners, but there are already some results, like [HSK00], that learn declarative rules which are added to the planning problem as additional constraints.

Actions for the Planning Community

ML is as relevant for KE in planning as it is for KE in other fields: ML-based tools have the potential to reduce the knowledge acquisition bottleneck. Taking into account what has already been achieved, we believe that the focus of future research should encompass the following areas:

Real-world domains: it could be said that current research on ML applied to planning has shown that ML techniques are useful, but they have only been tested in simple domains. Although some studies show that the knowledge learned scales to hard instances in those domains (e.g. 20 blocks in the blocks world), future studies should consider medium-sized instances in real-world domains.

Improving the quality of plans: although this is a very important issue, not much work has been done in this area. Plan quality improvement should become a focus for research in the immediate future. This is likely to be a tough task, because finding a solution in some domains is "easy", whereas finding good or optimal solutions is very hard. Further complications will arise when multi-objective quality measures (like time and cost) are considered.

User interaction: and yet, it has been shown that the learned knowledge still contains errors that could be corrected by a human [AB02]. Therefore, future studies should consider putting the human back into the learning cycle, possibly assisted by graphical tools, to validate, correct, and improve learned knowledge.

Integration of ML into KE environments: A lot has to be done in studying the interaction between a ML system and a planning expert/user/knowledge engineer (for instance, to validate and correct knowledge automatically acquired). Integrating ML into KE graphical tools should be of primary importance if ML is to be useful to Planning KE community.

ML for modern planners: Nowadays, there is a new generation of fast planners based on propositionalization, constraint satisfaction, and heuristic search. However, most of the work done in speed-up planning, and learning in general concerns older planning systems. New work should be done to apply previous learning methods and to develop new ones for these planners. Although they are already very fast, domain independent planning will always be a NP task, and domain dependent knowledge will be needed to solve hard instances in tough domains. Some care will be required so as not to damage performance in small and medium-sized problems because of the utility problem. In any case, planners becoming more and more efficient should make improving plan quality occupy the center stage in future research. Also, other features that go beyond STRIPS, like duration, numerical quantities, scheduling, uncertainty, hierarchical task network planning (HTN), ... should also be addressed from the control knowledge perspective.

Techniques and representation of control knowledge: It is difficult to determine which techniques should be studied in the future, but we believe that multi-strategy learning has good chances of improving results further, by combining the appropriate standalone ML techniques. Also, researchers should study whether current languages for representing heuristics are powerful enough to achieve the desired goals (i.e. achieving the maximum efficiency or quality possible in a particular domain). Probably, more powerful languages will be necessary, including the ability to carry out complex computations and handling numeric quantities. Of course, the more powerful, the more difficult it will be to automatically learn heuristics for those languages.

7.3.   Summary and conclusions

Summary of problems

The Roadmap points out some general problems to be overcome. These include:

1. Little knowledge of the creation, validation and maintenance of large domain models exists in the planning community. There is a general lack of experience (especially in Europe) of knowledge engineering concepts and methods in this area, and we have much to learn and techniques to import from related knowledge engineering work.

2. The evaluation of KE tools and methods is problematic, and tends to be harder than the evaluation of, say, a planning algorithm. This is seen as a barrier to future research as researchers find it harder to publish in the planning literature.

3. We need to learn how to characterize domains, and hence build up knowledge about how to match up planning technology with domain models. An example of work in this direction, based on statistical analysis, may be found in [HDH$^+$99].

4. KE involves a group of stakeholders with differing backgrounds and knowledge of computing. In particular, it includes consideration of Human Factors.

5. Planning research has a history of association with toy problems where KE issues are not relevant. There has been a focus in the planning community on theoretical aspects of techniques that do not scale up to real-world planning problems.

Summary of actions

The problems lead to some general actions that the community should carry out. These are summarized in Figure 7.2.

1. Encourage the planning research community to develop greater appreciation for the applied end of the research continuum, as argued in reference [Wd00].
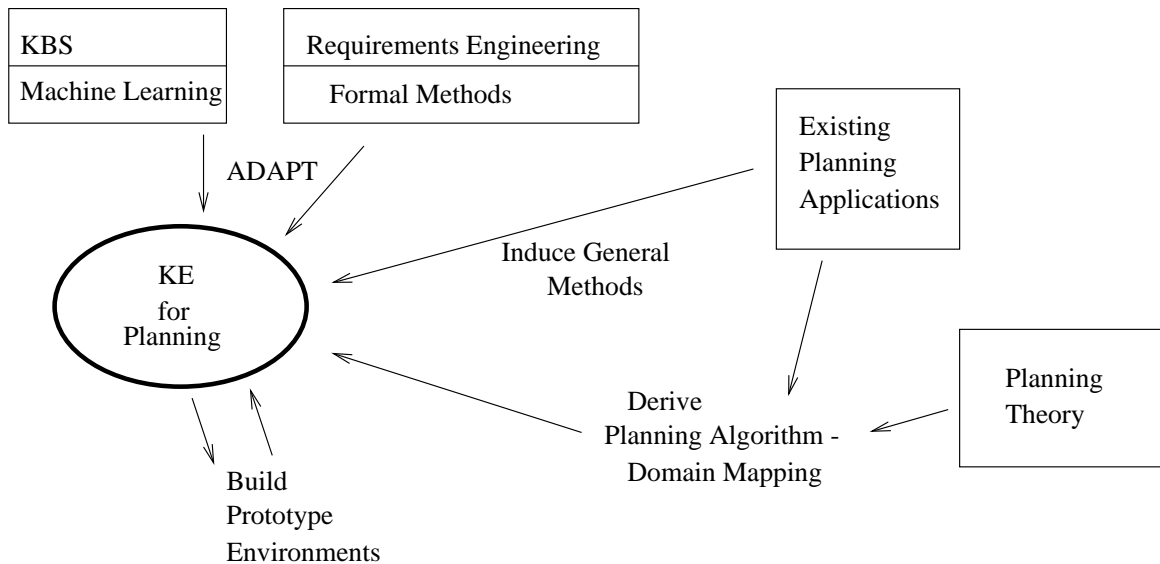
Figure 7.2.: Roadmap Summary

2. Survey the areas of knowledge acquisition, machine learning, requirements engineering and formal methods in software engineering for tools, techniques, and methods that may be relevant to planning.

3. Distill experience and induce general methods from the experience of previous applications of planning technology.

4. Attempt to build basic, integrated engineering environments for building planning applications. This may also involve promoting standards and/or formalisms that enable integration of planning engines and support tools developed separately.

5. Find a research methodology that allows efficient evaluation of work in KE to allow a route to publication and hence attract more research effort.

6. Build on previous work to create a planner taxonomy usable by knowledge engineers.

7. Develop a classification system and/or vocabulary for describing the characteristics of domains (as advised in reference [Her96])

Bibliography

[AB02]   Ricardo Aler and Daniel Borrajo. On control knowledge acquisition by exploiting human-computer interaction. In *Proceedings of the Sixth International Conference on Artificial Intelligence Planning Systems (AIPS-02)*, pages 141–149, Toulouse (France), April 2002. AAAI Press.

[ABI02]   R. Aler, D. Borrajo, and P. Isasi. Using genetic programming to learn and improve control knowledge. *Artificial Intelligence Journal*, 141(1-2):29–56, October 2002.

[ABvH94]   M. Aben, J. Balder, and F. van Harmelen. Support for the formalisation and validation of KADS expertise models. Technical Report KADS-II/M2/UvA/DM2.6a/1.0, ESPRIT, 1994.

[AKM00]   J.L. Ambite, C.A. Knoblock, and S. Minton. Learning plan rewriting rules. In *Proceedings of the Fifth Intenational Conference on Artificial Intelligence Planning and Schedulling*, Breckinridge. CO (USA), April 2000.

[Ama68]   S. Amarel. On representations of problems of reasoning about actions. In D. Michie, editor, *Machine Intelligence 3*, pages 131–171. Edinburgh University Press, Edinburgh, 1968.

[AS94]   P. M. Allen and L. T. Semmens. Integration of structured methods and formal notations. In *International Conference on Information Systems Development*, Bled, Slovenia, September 1994.

[BF81]   A. Barr and E. A. Feigenbaum, editors. *The Handbook of Artificial Intelligence. Vol. 1*. HeurisTek Press, Stanford, CA and William Kaufmann, Los Altos, CA, 1981.

[BF97]   Avrim L. Blum and Merrick L. Furst. Fast planning through planning graph analysis. *Artificial Intelligence*, 90:281–300, 1997.

[BF99]   Susanne Biundo and Maria Fox, editors. *Proceedings of the European Conference on Planning*. Springer, September 1999.

[BG99]   Blai Bonet and Hector Geffner. Planning as heuristic search: New results. In Biundo and Fox [BF99], pages 359–371.

[BKRG01]   J. Blythe, J. Kim, S. Ramachandran, and Y. Gil. An Integrated Environment for Knowledge Acquisition. In *Proceedings of the Interenational Conference on User Interfaces*, 2001.

[BLHL01]   Tim Berners-Lee, James Hendler, and Ora Lassila. The Semantic Web. *Scientific American*, 2001.

[BS96]   S. Biundo and W. Stephan. Modeling planning domains systematically. In *Proceedings of the European Conference on Artificial Intelligence (ECAI)*, 1996.

[BV97]   Daniel Borrajo and Manuela Veloso. Lazy incremental learning of control knowledge for efficiently obtaining quality plans. *AI Review Journal. Special Issue on Lazy Learning*, 11(1-5):371–405, February 1997.

[CG90]   Jaime Carbonell and Yolanda Gil. Learning by experimentation: The operator refinement method. In R. Michalski and Y. Kodratoff, editors, *Machine Learning: An Artificial Intelligence Approach*, volume III, pages 191–213. Morgan Kaufmann, Palo Alto, CA, 1990.

[CGLR96]   J. M. Crawford, M. Ginsberg, E. Luks, and A. Roy. Symmetry breaking predicates for search problems. In *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR)*, 1996.

[CO96]   A. Cesta and A. Oddi. DDL.1: A formal description of a constraint representation language for physical domains. In Ghallab and Milani [GM96]. Proceedings of the 3rd European Workshop on Planning (EWSP95), Assisi, Italy, September 27-29, 1995).

[Coh90]   W. W. Cohen. Learning approximate control rules of high utility. In *International Conference on Machine Learning*, pages 268–276, Austin, TX, 1990. Morgan Kaufmann.

[CZP$^+$89] M. P. Chase, M. Zweben, R. L. Piazza, J.D . Burger, P. P. Maglio, and H. Hirsh. Approximating learned search control knowledge. In *International Workshop on Machine Learning*, pages 218–220, 1989.

[DG02]   M. Davidson and M. Garagnani. Preprocessing planning domains containing Language Axioms. In *Proceedings of the Twentyfirst Workshop of the UK Planning and Scheduling SIG (PlanSIG-02)*, pages 23–34, Delft, NL, 2002.

[Dre81]   F.L. Dretske. *Knowledge and the Flow of Information*. MIT Press, Cambridge, MA, 1981.

[EM92]   Oren Etzioni and Steven Minton. Why EBL produces overly-specific knowledge: A critique of the Prodigy approaches. In *International Conference on Machine Learning*, pages 137–143, Aberdeen, Scotland, 1992. Morgan Kaufmann.

[EM97]   Tara A. Estlin and Raymond J. Mooney. Learning to improve both efficiency and quality of planning. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1227–1233, San Francisco, August 1997. Morgan Kaufmann.

[Etz93]   Oren Etzioni. Acquiring search-control knowledge via static analysis. *Artificial Intelligence*, 62(2):265–301, 1993.

[FL98]   Maria Fox and Derek Long. The automatic inference of state invariants in TIM. *Journal of AI Research*, 9:367–421, 1998.

[FL99]   Maria Fox and Derek Long. The detection and exploitation of symmetry in planning domains. Technical Report tr 1/99, Durham University, 1999.

[FL03]   M. Fox and D. Long. PDDL2.1: An extension to PDDL for expressing temporal plan-
            ning domains. *Journal of Artificial Intelligence Research – Special issue on the 3rd
            International Planning Competition*, 2003. (forthcoming).

[Gar00]   M. Garagnani. A correct algorithm for efficient planning with preprocessed domain
            axioms. In M. Bramer, A. Preece, and F. Coenen, editors, *Research and Development in
            Intelligent Systems XVII: Proceedings of ES2000*, pages 363–374, Cambridge, England,
            2000. Springer-Verlag.

[Gar03]   M. Garagnani. Model-Based Planning in Physical domains using SetGraphs. In
            M. Bramer, A. Preece, and F. Coenen, editors, *Proceedings of AI-2003*, Cambridge,
            England, December 2003. Springer-Verlag. (to appear).

[GB00]   Y. Gil and J. Blythe. PLANET: a shareable and reusable ontology for representing pla
            ns. In *Proceedings 17th International Conference on AI*, 2000.

[Geo87]   M.P. Georgeff. Planning. *Annual Review of Computer Science*, 2:359–400, 1987.

[GHK+98]   Malik Ghallab, Adele Howe, Craig A. Knoblock, Drew McDermott, Ashwin Ram,
            Manuela Veloso, Daniel Weld, and David Wikins. PDDL - the planning domain def-
            inition language. Technical Report CVC TR-98-003/DCS TR-1165, Yale Center for
            Computitational Vision and Control, 1998.

[GK97]   B.C. Gazen and C.A. Knoblock. Combining the expressivity of UCPOP with the Effi-
            ciency of Graphplan. In *Proceedings of the Fourth European COnference on Planning
            (ECP-97)*, pages 221–233, Toulouse, France, 1997.

[GM96]   M. Ghallab and A. Milani, editors. *New Directions in AI Planning*. IOS Press (Amster-
            dam), 1996. Proceedings of the 3rd European Workshop on Planning (EWSP95), Assisi,
            Italy, September 27-29, 1995).

[GMB94]   S. Greenspan, J. Mylopoulos, and A. Borgida. On formal requirements modeling lan-
            guages: RML revisited. In *International Conference on Software Engineering*. IEEE
            Computer Science Press, 1994.

[GMB98]   Ramn Garca-Martnez and Daniel Borrajo. An integrated approach of learning, planning,
            and execution. *IEEE Transactions on Systems, Man and Cybernetics*, 1998.

[GNC95]   J. Glasgow, N.H. Narayanan, and B. Chandrasekaran, editors. *Diagrammatic Reasoning*.
            MIT Press, Cambridge, MA, 1995.

[GS98]   Alfonso Gerevini and Lenhard Schubert. Inferring state constraints for domain-
            independent planning. In *Proceedings of the AAAI National Conference on Artificial
            Intelligence*, pages 905–912, 1998.

[GTR01]   Garland, Tyall, and Rich. Learning hierarchical task models by defining and refining
            examples. In *Proceedings of the First International Conference on Knowledge Capture*,
            pages 44–51, 2001.

[Ham86]   Kristian J. Hammond. Chef: a model of case-based planning. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, 1986.

[Ham93]   E. Hammer. Reasoning with sentences and diagrams. In G. Allwein and J. Barwise, editors, *Working papers on Diagrams and Logic. Preprint IULG-93-24*, pages 120–143, Indiana Univ., Bloomington, IN, 1993.

[Hay74]   P.J. Hayes. Some problems and non-problems in representation theory. In *Proceedings of the AISB Summer Conference*, pages 63–79, University of Sussex, Brighton, England, 1974.

[HDH$^+$99] Adele E. Howe, Eric Dahlman, Christopher Hansen, Michael Scheetz, and Annelise von Mayrhauser. Exploiting competitive planner performance. In Biundo and Fox [BF99], pages 60–72.

[Her96]   J. Hertzberg. On building a planning toolbox. In Ghallab and Milani [GM96]. Proceedings of the 3rd European Workshop on Planning (EWSP95), Assisi, Italy, September 27-29, 1995).

[HJ00]   Patrik Haslum and Peter Jonsson. Planning with reduced operator sets. In Steve Chien, Subbarao Kambhampati, and Craig A. Knoblock, editors, *Proceedings of the Conference on Artificial Intelligence Planning & Scheduling*, pages 150–158, April 2000.

[HS03]   Patrik Haslum and Ulrich Scholz. Domain knowledge in planning: Representation and use. In Derek Long, Drew McDermott, and Sylvie Thiébaux, editors, *ICAPS'03 Workshop on PDDL*, pages 69–78, May 2003.

[HSK00]   Yi-Cheng Huang, Bart Selman, and Henry Kautz. Learning declarative control rules for constraint-based planning. In *Proceedings of the 17th International Conference on Machine Learning*, pages 415–422, Stanford (USA), 2000.

[HV91]   Joseph Y. Halpern and Moshe Y. Vardi. Model Checking vs. Theorem Proving: A Manifesto. In J. A. Allen, R. Fikes, and E. Sandewall, editors, *Principles of Knoweldge representation and Reasoning: Proceedings of the 2nd International Conference (KR91)*, pages 325–332, 1991.

[IDNA02]   O. Ilghami, H. Muñoz-Avila D. Nau, and D. Aha. Camel: Learning methods for htn planning. In *Proceedings of the Conference on AI Planning and Scheduling (AIPS'02)*, Toulouse (France), 2002.

[IK96]   L. Ihrig and S. Kambhampati. An explanation-based approach to improve retrieval in case-based planning. In Ghallab and Milani [GM96], pages 395–406. Proceedings of the 3rd European Workshop on Planning (EWSP95), Assisi, Italy, September 27-29, 1995).

[Jon90]   C. B. Jones. *Systematic Software Development using VDM*. Prentice-Hall, 2 edition, 1990.

[Ket94]   Vera Kettnaker. Training of a case based reasoning planner with dynamically generated subproblems. Master's thesis, University of Kaiserslautern, January 1994.

[KH00]   Jana Koehler and Jörg Hoffmann. On reasonable and forced goal orderings and their use in an agenda-driven planning algorithm. *Journal of Artificial Intelligence Research*, 12:338–386, 2000.

[Kha99]   R. Khardon. Learning action strategies for planning domains. *Artificial Intelligence Journal*, 113:125–148, 1999.

[KK94]   Suresh Katukam and Subbarao Kambhampati. Learning explanation-based search control rules for partial order planning. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, volume 1, pages 582–587, Menlo Park, CA, USA, July 31–August 4 1994. AAAI Press.

[Kno94]   C. A. Knoblock. Automatically generating abstractions for planning. *Artificial Intelligence*, 68:243 – 302, 1994.

[Kor85]   Richard E. Korf. Macro-operators: A weak method for learning. *Artificial Intelligence*, 26:35–77, 1985.

[KR96]   R. Khardon and D. Roth. Reasoning with Models. *Artificial Intelligence*, 87:187–213, 1996.

[Kul94]   Z. Kulpa. Diagrammatic representation and reasoning. *Machine GRAPHICS & VISION*, 3(1/2):77–103, 1994.

[Lec93]   Christopher Leckie. Being correct is not enough - the role of domain knowledge in the utility problem. In *Australian Joint Conference on AI*, November 1993.

[Lev92]   H. Levesque. Is reasoning too hard? In *Proceedings of the Third NEC Research Symposium*, 1992.

[LF99]   Derek Long and Maria Fox. Automatic synthesis and use of generic types in planning. tr 3/99, Durham University, 1999.

[LF00]   Derek Long and Maria Fox. Extracting route-planning: First steps in automatic problem decomposition. In *AIPS Workshop on Analysing and Exploiting Domain Knowledge for Efficient Planning*, 2000.

[Lif86]   V. Lifschitz. On the semantics of STRIPS. In M.P. Georgeff and Lansky, editors, *Proceedigns of 1986 Workshop: Reasoning about Actions and Plans*, 1986.

[Lin95]   R.K. Lindsay. Images and Inference. In J. Glasgow, N.H. Narayanan, and B. Chandrasekaran, editors, *Diagrammatic Reasoning*, chapter 4, pages 111–135. MIT Press, Cambridge, MA, 1995.

[LM00]   D. Liu and T.L. McCluskey. The OCL Language Manual, Version 1.2. Technical report, Department of Computing and Mathematical Sciences, University of Huddersfield (UK), 2000.

[LS87]   J.H. Larkin and H.A. Simon. Why a diagram is (sometimes) worth ten thousands words. *Cognitive Science*, 11:65–99, 1987.

[MB87]   Jack Mostow and Neeraj Bhatnagar. 21 – a floor planner that uses EBG to learn from its
           failures. In John McDermott, editor, *Proceedings of the International Joint Conference
           on Artificial Intelligence (IJCAI)*, pages 249–265, Milan, Italy, August 1987. Morgan
           Kaufmann.

[McC58]   J. McCarthy. Programs with common sense. In *Proceedings of the Symposium on the
           Mechanization of Thought Processes, Vol. 1*, pages 77–84, 1958.

[McC87]   T. L. McCluskey. Combining weak learning heuristics in general problem solvers. In
           Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI).
           Morgan Kaufman, August 1987.

[McC03]   T. L. McCluskey. PDDL: A Language with a Purpose? In *Proceedings of the
           ICAPS-2003 Workshop on PDDL, International Conference on Automated Planning and
           Scheduling, Trento, Italy*, 2003.

[MFHS02]   Deborah L. McGuinness, Richard Fikes, James Hendler, and Lynn Andrea Stein.
           DAML+OIL: An ontology language for the Semantic Web. *IEEE Intelligent Systems*,
           17(5), 2002.

[MG00]   M. Martin and H. Geffner. Learning generalized policies in planning using concept
           languages. In *Proceedings of the 7th Conference on Knowledge Representation and
           Reasoning*, Colorado (USA), 2000.

[MH69]   J. McCarthy and P.J. Hayes. Some Philosophical Problems from the Standpoint of Arti-
           ficial Intelligence. In B. Meltzer and D. Michie, editors, *Machine Intelligence*, volume 4.
           Edinburgh University Press, 1969.

[Min88]   Steven Minton. *Learning Effective Search Control Knowledge: An Explanation-Based
           Approach*. Kluwer Academic Publishers, Boston, MA, 1988.

[MK92]   K. Myers and K. Konolige. Reasoning with analogical representations. In B. Nebel,
           C. Rich, and W. Swartout, editors, *Principles of Knowledge Representation and Rea-
           soning: Proceedings of the Third International Conference (KR92)*, pages 189–200.
           Morgan Kaufmann Publishers Inc., San Mateo, CA, 1992.

[MM01a]   T. L. McCluskey and M.M.West. The automated refinement of a requirements domain
           theory. *Journal of Automated Software Enginnering. Special Issue on Inductive Pro-
           gramming*, 6(2):195–218, May 2001.

[MM01b]   M. M.West and T. L. McCluskey. The application of machine learning tools to the
           validation of an air traffic control domain theory. *International Journal on Artificial
           Intelligence Tools*, 10(4):613–637, December 2001.

[MP97]   T. L. McCluskey and J. M. Porteous. Engineering and compiling planning domain mod-
           els to promote validity and efficiency. *Artificial Intelligence*, 95:1–65, 1997.

[MRS02]   T. L. McCluskey, N. Richardson, and R. Simpson. An interactive method for inducing
           operator descriptions. In *Proceedings of the 6th International Conference on AI Plan-
           ning and Scheduling (AIPS-2002)*, Toulouse (France), April 2002.

[MW97]   K.L. Myers and D.E. Wilkins. *The Act-Editor User's Guide: A Manual for Version 2.2*. SRI International Artificial Intelligence Center, Menlo Park, CA, September 1997.

[MYK98]   A. R. Moro, H. Yu, and G. Kelleher. Applying new search methodologies for scheduling FMS using petri nets. In *Proceedings of the 17th UK PlanSIG*, University of Huddersfield, UK, September 1998.

[NDK97]   Bernhard Nebel, Yannis Dimopoulos, and Jana Koehler. Ignoring irrelevant facts and operators in plan generation. In Biundo and Fox [BF99], pages 338–350.

[New82]   A. Newell. The Knowledge Level. *Artificial Intelligence*, 18:87 – 127, 1982.

[Pal78]   S.E. Palmer. Fundamental Aspects of Cognitice Representation. In E. Rosch and B.B. Lloyed, editors, *Computing and Categorization*, chapter 9, pages 259–303. Earlbaum, Hillsdale, NJ, 1978.

[Pér95]   M. Alicia Pérez. *Learning Search Control Knowledge to Improve Plan Quality*. PhD thesis, School of Computer Science, Carnegie Mellon University, July 1995. Also available as Technical Report CMU-CS-95-175.

[PK86]   Bruce W. Porter and Dennis F. Kibler. Experimental goal regression: A method for learning problem-solving hueristics. *Machine Learning*, 1(3):249–285, 1986.

[Pla]   Planform: An open environment for building planners. `http://scom.hud.ac.uk/planform`.

[Pol98]   J.L. Pollock. Perceiving and Reasoning about a Changing World. *Computational Intelligence*, 14(4):498–562, 1998.

[Pol99]   S. Polyak. A common process methodology for engineering process domains. In *Proceedings of the Systems Engineering for Business Process Change (SEBPC) workshop, SMBPI: Systems Modelling for Business Process Improvement*, University of Ulster, March 1999.

[PST91]   B. Potter, J. Sinclair, and D. Till. *An Introduction to Formal Specification and Z*. Prentice-Hall, 1991.

[Rin00]   Jussi Rintanen. An iterative algorithm for synthesizing invariants. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, 2000.

[RK89]   David Ruby and Dennis Kibler. Learning subgoal sequences in planning. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 609–614, San Mateo, CA, 1989. Morgan Kaufmann.

[RK92]   David Ruby and Dennis Kibler. Learning episodes for op5ization. In *International Conference on Machine Learning*, pages 379–384, Aberdeen, Scotland, 1992. Morgan Kaufmann.

[RV99]   Ioannis Refanidis and Ioannis Vlahavas. GRT: A domain independent heuristic for STRIPS worlds based on greedy regression tables. In Biundo and Fox [BF99], pages 346–358.

[Sch99]   Ulrich Scholz. Action constraints for planning. In Biundo and Fox [BF99], pages 148–160.

[SG96]   L. G. Shaw and B. R. Gaines. Requirements acquisition. *Software Engineering Journal*, 11:149–165, 1996.

[Sha97]   M. Shanahan. *Solving the Frame Problem*. MIT Press, Cambridge, MA, 1997.

[Sim81]   H.A. Simon. *The Sciences of the Artificial*. MIT Press, Cambridge, MA, 1981.

[Slo75]   A. Sloman. Afterthoughts on analogical representations. In *Proceedings of the First Workshop on Theoretical Issues in Natural Language Processing (TINLAP-1)*, pages 164–171, Cambridge, MA, 1975.

[Tat98]   A. Tate. Roots of SPAR - shared planning and activity representation. *Knowledge Engieering Review*, 13:121 – 128, March 1998.

[THN03]   Sylvie Thiébeaux, Jörg Hoffmann, and Bernhard Nebel. In Defense of PDDL Axioms. In *Proceedings of ICAPS'03 Workshop on the Competition: Impact, Organization, Evaluation, Benchmarks*, Trento, Italy, June 2003.

[T.L88]   T.L.McCluskey. Experience-driven heuristic acquisition in general problem solvers. Technical report, Department of Computer Science, The City University, London, 1988. PhD thesis.

[TM94]   J. G. Turner and T. L. McCluskey. *The Construction of Formal Specifications: An Introduction to the Model-Based and Algebraic Approaches*. McGraw-Hill series in Software Engineering. McGraw-Hill, 1994.

[TPJ98]   A. Tate, S. Polyak, and P. Jarvis. TF method: An initial framework for modelling and analysing planning domains. In *AIPS-98 workshop on "Knowledge Engineering and Acquisition for Planning: Bridging Theory and Practice"*, Carnegie-Mellon University, June 1998. AAAI Technical Report WS-98-03.

[UKSZ98]   M. Uschold, M. King, S.Moralee, and Y. Zorgios. The Enterprise Ontology. *Knowledge Engieering Review*, 13, March 1998.

[Vel94]   Manuela M. Veloso. Flexible strategy learning: Analogical replay of problem solving episodes. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, pages 595–600, Seattle, WA, August 1994.

[vHF95]   F. van Harmelen and D. Fensel. Formal Methods in Knowledge Engineering. Technical report, The University of Amsterdam, 1995.

[VSD02]   Klaus Varrentrapp, Ulrich Scholz, and Patrick Duchstein. Design of a testbed for planning systems. In Lee McCluskey, editor, *AIPS'02 Workshop on Knowledge Engeneering Tools and Techniques for AI Planning*, pages 51–58, April 2002.

[VTV02]   Dimitris Vrakas, Grigorios Tsoumakas, and Ioannis Vlahavas. Towards adaptive heuristic planning through machine learning. In Tim Grant and Cees Witteveen, editors, *UK Planning and Scheduling SIG Workshop*, pages 12–21, November 2002.

[Wan94]   Xuemei Wang. Learning planning operators by observation and practice. In *Proceedings of the Conference on Artificial Intelligence Planning Systems*, pages 335–340, Chicago, IL, June 1994. AAAI Press, CA.

[Wd00]   D. Wilkins and M. desJardins. A Call for Knowledge-based Planning. In *Procceedings of the 2nd NASA International Workshop on Plan ning adn Scheduling for Space*, page 187, 2000.

[WSB92]   B. J. Wielinga, A. Th. Schrieber, and J. Breuker. KADS - a modelling approach to knowledge engineering. *Knowledge Acquisition*, 4(1):5 – 53, 1992.

[ZK03]   Terry Zimmerman and Subbarao Kambhampati. Learning-assisted automated planning: Looking back, taking stock, going forward. *AI Magazine*, 2003.

8. Online Planning and Scheduling

8.1. Introduction

This Roadmap results from exchanges between people involved in the PLANET II TCU about *Online Planning & Scheduling*. These exchanges took place in various workshops which have been organized by the *Planning & Scheduling* and *Constraint-based Reasoning* communities:

- AIPS 2002 workshop on *Online Planning & Scheduling*, in April 2002, in Toulouse, France (see `http://www.laas.fr/aips/`; this workshop has been sponsored by the PLANET II network);

- CP 2001 workshop on *Online Combinatorial Problem Solving and Constraint Programming*, in November 2001, in Paphos, Cyprus (see `http://www2.cs.ucy.ac.cy/~iclpcp01/CP01home.html`);

- CP 2001 workshop on *Constraints and Uncertainty*, in November 2001, in Paphos, Cyprus (see `http://www2.cs.ucy.ac.cy/~iclpcp01/CP01home.html`);

- ESA workshop on *On-board Autonomy*, in October 2001, in Noordwijk, The Netherlands (see `http://www.estec.esa.nl/conferences/01C06/`);

- CP 2003 workshop on *Online Constraint Solving: Handling Change and Uncertainty*, in September 2003, in Cork, Ireland (see `http://www.cs.ucc.ie/cp2003/`).

8.1.1. The Online Planning and Scheduling TCU

This PLANET II TCU goes on with the PLANET I TCU on *Dynamic Scheduling*, which has been extended to AI Planning and Scheduling. Differently from other PLANET II TCUs which are dedicated to some specific application areas, like robot planning, intelligent manufacturing, or planning and scheduling for the web, this TCU is not application-oriented, but rather methodology-oriented. Its objective is to study all the consequences on AI Planning and Scheduling methods and tools of the online setting which is present in numerous applications.

It started from the observation that most of the AI Planning and Scheduling methods and tools have been developed for an use in an offline setting (planning and scheduling phase followed by the execution phase; see Figure 8.1), although most of the applications appear in a very different online setting (planning and scheduling phase and execution phase going off concurrently; see Figure 8.2).
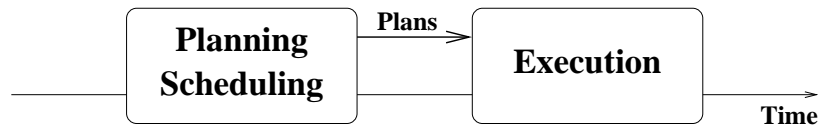
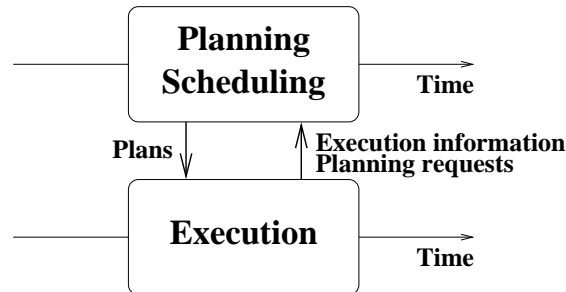Figure 8.1.: Offline AI Planning and Scheduling.

Figure 8.2.: Online AI Planning and Scheduling.

Following this observation, it appeared quickly that dealing with an online setting is not a simple question of adaptation of the existing methods and tools and that it implies at least to revisit, but sometimes to rethink and redesign these methods and tools, in order to be able to meet the requirements of a online setting.

8.1.2. AI P&S and online planning and scheduling

An online setting appears in most of the practical applications of the AI Planning and Scheduling methodology, in applications whose objective is to control correctly and optimally an autonomous engine (robot, unmanned air vehicle, satellite . . .), as well as in applications whose objective is to manage efficiently a large system (fleet of vehicles, communication network, production or distribution organization . . .).

In all cases, changes may occur at any moment from the environment (changes in the weather conditions, in the market indicators . . .), from the system users (changes in the current objectives, requests, orders . . .), from the controlled system itself (component failures, unexpected behaviors, unexpected activity durations or resource consumptions . . .), or from other controlled systems in a distributed system setting.

An illustrating example could be a travel management module embedded in a car and dedicated to the management of the route, of the tank refueling, of the car maintenance, of the reservations of hotels, restaurants or site visits, and of the meetings with other people. Changes may come from the environment (weather or traffic conditions), from the driver (request for a new visit or meeting), from the car (unexpected fuel consumption, suspicious behavior of a component), or from similar modules embedded in other cars (request for a meeting between several traveling people).

In most of the cases, these situations cannot be all offline anticipated, mainly because of the combinatorics which would result from all the possible combinations. The only practical solution is thus to be able to react online when such situations occur, according to the current situation to face.

But the controlled system cannot stop to run each time a change in the current situation occurs and each time the planning and scheduling module has to reason in order to build a new plan or to adapt the previous one. Consequently, planning and scheduling must be performed concurrently with the normal system life. Planning and scheduling, in the one hand, and execution, in the other hand, become concurrent activities.

8.1.3. Requirements

The main requirements which appear in such an online setting are mainly four:

1. it may be useful to limit as much as possible the successive calls to the planning and scheduling module; this can be achieved by producing robust solutions, which are plans which resist as much as possible the most likely changes;

2. when a call to the planning and scheduling module is necessary, because the current plan is no more consistent with the current situation, it may be useful to limit as much as possible changes in the current plan, mainly when people are involved in the execution of the plan, because people do not like too frequent changes in activity plans; this can be achieved by producing flexible solutions, which are plans which can be easily adapted to a specific situation;

3. always when a call to the planning and scheduling module is necessary, building a new plan or adapting the previous one must take as less time as possible, because the system goes on running and needs a plan to run correctly: for example, the driver of a car needs to know what to do at the next crossing;

4. finally, despite of these temporal deadlines on the reasoning task, one wants the produced plans remain consistent and their quality be as highest as possible, in order the system be correctly and optimally controlled.

8.2. Roadmap themes

The work of the TCU has been organized along three directions:

1. analysis of the planning and scheduling setting and of the associated requirements;

2. design of a generic control architecture;

3. analysis of the existing approaches.

As a fourth theme, the TCU investigated the perspective of current research trends to project promising future fields of research. These recommendations for potential research directions resulting from the discussion of the above themes will be presented in Section 8.3.

8.2.1. Online planning and scheduling setting and associated requirements

The first work of the TCU consisted in the analysis of the online setting of most of the applications of the planning and scheduling technology and of the resulting requirements.

This work had begun with the PLANET I TCU on *Dynamic Scheduling* and led us to the synthesis which can be found in sections 8.1.1, 8.1.2, and 8.1.3 above.

This synthesis needed a long work in order to overcome the particular points of view and the particular vocabularies which naturally result from the experience of each one when dealing with specific applications.

But we can consider that, despite some unavoidable shortcomings, we got a satisfying result, that is a common understanding of the setting and of its requirements.

8.2.2. A generic architecture

Another result of the TCU is an agreement about a generic architecture of the control system of either an engine or a large system, in which planning and scheduling take place. This architecture is roughly described in Figure 8.3. The control is divided into two main parts:

- a reactive part, which includes the supervision and the execution control modules;

- a deliberative part, which is the planning and scheduling module itself, which includes submodules dedicated to the off-line preprocessing, the online problem definition, the online problem solving, and the online problem solution.

In the reactive part, modules work in a real-time mode (requirement of a strictly limited reaction time). The supervision module is in charge of the interaction between the external world (physical system, environment, users, other controlled systems) and the control system and in charge of the global supervision of this control system (for example, activation and control of the planning and scheduling module). The execution control module is in charge of the execution and control of the decided actions on the physical system.

In the deliberative part, modules work in a deliberative mode (not strictly limited reaction time, more or less hard deadlines). The online problem definition module is in charge of the definition of the current planning and scheduling problem to solve (changes to take into account, length of the commitment, decision, and reasoning horizons, required levels of precision . . .). The online problem solving is in charge of the effective solving of the defined problem. It uses search algorithms and a search control submodule allows the temporal deadlines to be met. The online problem solution module is in charge of the management of the produced plans. Finally, an offline problem solving module may help the online problem solving, by using any form of preprocessing.

8.2.3. Analysis and synthesis of approaches

An important part of the work of the TCU consisted in the inventory, analysis and synthesis of the approaches, methods and tools that have been proposed so far to deal with online planning and scheduling problems. The synthesis led to a partition of the existing approaches into two main classes:
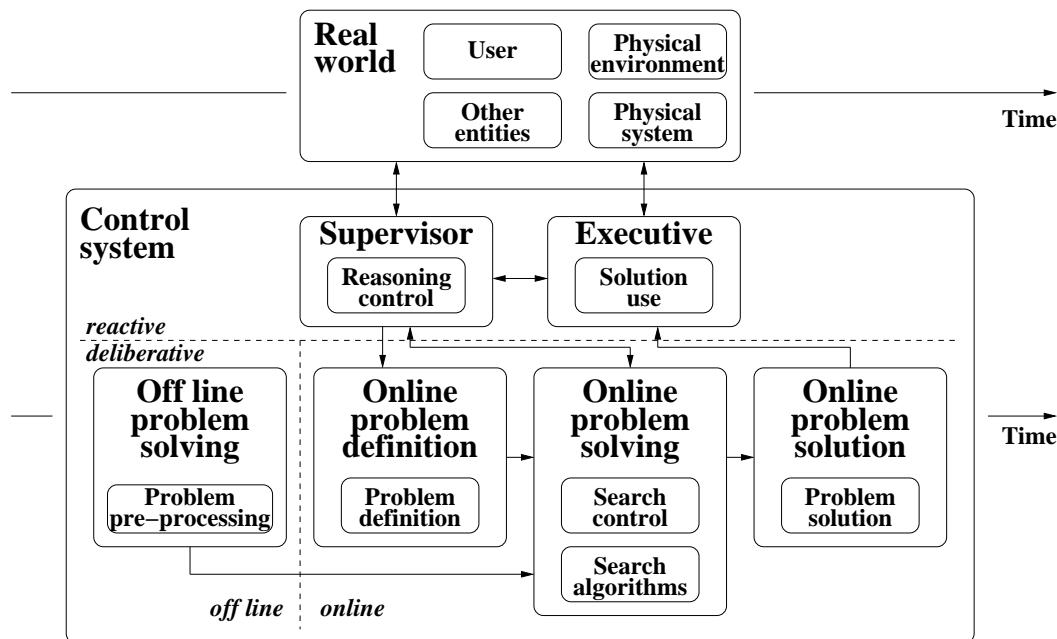
Figure 8.3.: Generic architecture of the control system of an engine or a large system.

Reactive approaches   They use no model of the possible changes and try to react as best as possible to these changes when they occur; in order to avoid planning and scheduling again from scratch each time a change occurs, these methods try to reuse either previous solutions or previous reasonings; for example, many methods, more or less inspired from the generic local search methods, have been proposed for the repair of a schedule when an event has invalidated it (see for example [MJPL92], [ZDDD94], [Smi95], or [CKS$^+$00]); other methods, inspired from generic methods in constraint satisfaction, have been proposed for the recording and reuse of justified deductions when reasoning on scheduling problems (see for example [EGJ02]); sometimes, the objective is not only to produce quickly a new consistent schedule, but moreover to find one which is as close as possible to the previous one, in order to favor solution stability (see for example [SW00]).

Proactive approaches   They use a more or less precise model of the possible changes and try to anticipate them when building a schedule; among them, we can distinguish methods which try to build robust solutions, that are schedules whose probability to remain valid is the highest (see for example [DGB01] or [BM02]) and methods which try to build flexible solutions, that are schedules which can be easily adapted in case of any event during execution (see for example [DBS94] or [PSCO03]); let us note that an interesting way of research, limited for the moment to the management of temporal constraints, tries to replace the usual notion of consistency (satisfaction of all the hard constraints) by a notion of controllability (existence of a decision which satisfies all the hard constraints whatever events which may occur during execution within specified limits; see for example [VF99]).

For a large survey of reactive and proactive methods in scheduling, see [DB]. Note however signif-

icant differences between these two classes of methods: the reactive ones do not need any model of the possible changes and are far easier to implement than the proactive ones; the former have been consequently far more developed and applied than the latter; but we can consider that most of the research effort should be now focused on the latter which are currently the most promising, as well as on a combination of both.

8.3.   Summary and conclusions

Potential directions of research

The most difficult work of the TCU consisted in the identification of the most relevant and promising directions of research which would allow online planning and scheduling applications to be correctly, optimally, and efficiently dealt with. Among all the directions we considered, we would like to enlighten the ones in the following paragraphs.

As it has been previously pointed out, research might be usefully focused on the study of proactive methods, which use models of the possible changes to produce robust and flexible solutions; among these methods, we think that a special focus should be done on the extension of the notion of controllability from temporal features to other features like resource levels or discrete system states (see for example [WF00] and [HLT02]).

Useful connections might be done between these studies and similar or concurrent studies in other communities like the ones on sequential decision making (Markov decision processes), uncertainty representation (Bayesian networks), constraint satisfaction (see for example [Wal02] and [BBCR01]) and boolean satisfiability (see for example [LMP01]).

Finally, each work on online planning and scheduling has considered one feature of the problem (reactivity, robustness, flexibility, or stability) and tried to improve existing methods with regard to this particular feature; it might be certainly interesting now to revisit the online planning and scheduling problem as a whole, with all its features, all its requirements, and all their interactions, in order to make progress with regard to:

1. the production of robust, flexible, and stable solutions; the tradeoffs between robustness, flexibility, stability, and optimality;

2. the redefinition of the utility of a solution as a function of its quality and of the time at which it is delivered to the execution module (decreasing function of this time); the tradeoffs between the complexity of the online reasoning and the utility of the resulting solutions; the online control of the reasoning, taking into account solution quality and utility; the use of the results of previous studies on resource and time-bounded reasoning (see for example [BD94] or [Zil96]);

3. the tradeoffs between offline preprocessing and online reasoning; the limits of an offline preprocessing in terms of time and space.

Conclusions

As a conclusion, we would like to plead for stronger connections between scientific communities and any initiative which would favor it. Indeed, we think that the AI Planning and Scheduling community, and more specifically all the people who work on online planning and scheduling, might take advantage of common studies with people involved in connected communities and particularly in the following ones:

- the operations research, constraint satisfaction, and boolean satisfiability communities: these connections have been existing from a long time and only need to be more systematically encouraged;

- the people who work on reasoning and decision under uncertainty (Bayesian networks, influence diagrams, Markov decision processes); this is justified by the fact that, on the one hand, Markov decision processes use basic results from decision theory, produce policies rather than plans, but use explicit representations of the states, actions and probabilities which make them unusable as soon as one wants to go out from small toy problems, and that, on the other hand, planning and scheduling produce plans rather than policies, use implicit representations which make them efficient, but do not correctly represent and reason about uncertainty;

- the people who work on real-time system control, discrete-event system control, and validation; this is justified by the fact that, on the one hand, these people are used to represent, simulate, and validate dynamic systems by using basic frameworks such as synchronous languages, automata, Petri nets, or others, but only consider very simple automatic systems, without any reasoning, planning, and decision capabilities, and that, on the other hand, planning and scheduling people try to build autonomous systems which involve reasoning, planning, and decision capabilities, but have some difficulties trying to deal with real-time constraints (see for example [FC97] for preliminary connections).

Bibliography

[BBCR01]   T. Benoist, E. Bourreau, Y. Caseau, and B. Rottembourg. Towards Stochastic Constraint Programming: A Study of Online Multi-choice Knapsack with Deadlines. In *Proc. of the 7th International Conference on Principles and Practice of Constraint Programming (CP-01)*, pages 61–76, Paphos, Cyprus, 2001.

[BD94]   M. Boddy and T. Dean.   Deliberation Scheduling for Problem Solving in Time-Constrained Environments. *Artificial Intelligence*, 67(2):245–285, 1994.

[BM02]   J. Branke and D. Mattfeld. Anticipatory Scheduling for Dynamic Job Shop Problems. In *Proc. of the AIPS-02 Workshop on "Online Planning and Scheduling"*, Toulouse, France, 2002.

[CKS⁺00]   S. Chien, R. Knight, A. Stechert, R.Sherwood, and G. Rabideau. Using Iterative Repair to Improve the Responsiveness of Planning and Scheduling. In *Proc. of the Fifth International Conference on Artificial Intelligence Planning and Scheduling (AIPS-00)*, Breckenridge, CO, USA, 2000.

[DB]   A. Davenport and C. Beck. A Survey of Techniques for Scheduling with Uncertainty. Unpublished. See http://4c.ucc.ie/j̃cb/publications.html.

[DBS94]   M. Drummond, J. Bresina, and K. Swanson. Just-In-Case scheduling. In *Proc. of the 12th National Conference on Artificial Intelligence (AAAI-94)*, pages 1098–1104, Seattle, WA, USA, 1994.

[DGB01]   A. Davenport, C. Gefflot, and C. Beck. Slack-based Techniques for Robust Schedules. In *Proc. of the Sixth European Conference on Planning (ECP-01)*, Toledo, Spain, 2001.

[EGJ02]   A. Elkhyari, C. Guéret, and N. Jussien. Conflict-based Repair Techniques for Solving Dynamic Scheduling Problems. In *Proc. of the 8th International Conference on Principles and Practice of Constraint Programming (CP-02)*, Ithaca, New York, USA, 2002.

[FC97]   M. Fromherz and J. Conley. Issues in Reactive Constraint Solving. In *Proc. of the CP-97 Workshop on "Concurrent Constraint Programming for Time-Critical Applications"*, Schloss Hagenberg, Austria, 1997.

[HLT02]   M.J. Huguet, P. Lopez, and T.Vidal. Dynamic Task Sequencing in Temporal Problems with Uncertainty. In *Proc. of the AIPS-02 Workshop on "Online Planning and Scheduling"*, Toulouse, France, 2002.

[LMP01]   M. Littman, S. Majercik, and T. Pitassi. Stochastic Boolean Satisfiability. *Journal of Automated Reasoning*, 27(3):251–296, 2001.

[MJPL92]   S. Minton, M. Johnston, A. Philips, and P. Laird. Minimizing Conflicts: a Heuristic Repair Method for Constraint Satisfaction and Scheduling Problems. *Artificial Intelligence*, 58:160–205, 1992.

[PSCO03]   N. Policella, S. Smith, A. Cesta, and A. Oddi. Steps toward Computing Flexible Sched-
           ules. In *Proc. of the CP-03 Workshop on "Handling Change and Uncertainty"*, Cork,
           Ireland, 2003.

[Smi95]   S. Smith. Reactive Scheduling Systems. In D. Brown and W. Sherer, editors, *Intelligent
           Scheduling Systems*. Kluwer, 1995.

[SW00]   H. El Sakkout and M. Wallace. Probe Backtrack Search for Minimal Perturbation in
           Dynamic Scheduling. *Constraints : An International Journal*, 5(4), 2000.

[VF99]   T. Vidal and H. Fargier. Handling Contingency in Temporal Constraint Networks: From
           Consistency to Controllabilities. *Journal of Experimental and Theoretical Artificial In-
           telligence*, 11(1):23–45, 1999.

[Wal02]   T. Walsh. Stochastic Constraint Programming. In *Proc. of the 15th European Conference
           on Artificial Intelligence (ECAI-02)*, Lyon, France, 2002.

[WF00]   R. Wallace and E. Freuder. Dispatchability Conditions for Schedules with Consumable
           Ressources. In *Proc. of the 14th European Conference on Artificial Intelligence (ECAI-
           00)*, pages 526–539, Berlin, Germany, 2000.

[ZDDD94]   M. Zweden, B. Daun, E. Davis, and M. Deale. Scheduling and Rescheduling with
           Iterative Repair. In M. Zweden and M. Fox, editors, *Intelligent Scheduling*, pages 241–
           256. Morgan Kaufmann, 1994.

[Zil96]   S. Zilberstein. Using Anytime Algorithms in Intelligent Systems. *AI Magazine*,
           17(3):73–83, 1996.

9. Summary of Recommended Actions and Conclusions

Intelligent Manufacturing Systems

This Roadmap has attempted to evaluate the obstacles which have impeded the penetration of PLANET technologies into manufacturing applications as well as to examine the ways in which those obstacles might be eroded. The Intelligent Manufacturing Systems TCU has always been an embryonic group, but this document suggests that there is much promise in the manufacturing area for AI Planning and Scheduling, and that with the right approach, a much more thriving community of interest can be built to the benefit of all.

There is some perceived bias in the AI Planning Community towards rewarding the development of new planning algorithms rather than some of the areas raised above. The International Planning Competition in its present form, for example, encourages the development of fast algorithms but not the production of interactive planners or of replanning capabilities. The most attractive areas for research, judging from the volume of papers submitted in the recent period, is Graphplan and DecisonTheoretic Planning, both very much development of algorithms. If we agree that other areas should also be developed, then ways of encouraging researchers to do so should be sought.

Some possible actions include the development of some standard problems which require the desired research developments (these need not be in manufacturing domains at all); the formulation of outline projects for national or European funding bodies which researchers can develop according to their particular interests and concerted attempts to publicize what has been done in these areas via journal special issues or conference workshops.

It is clear that linking the AI Planning Community to those working in intelligent manufacturing is vital if the PLANET technologies are to be applied to manufacturing problems. As argued above, there is very little awareness of these technologies outside their home community, and, it should be added, there is generally very little appreciation of manufacturing issues in the PLANET community. Pushing the technology directly to end-users is a possibility, but working more closely with people who are already familiar with manufacturing seems a quicker and more efficient way of educating both communities. An obvious way of increasing contact and the exchange of knowledge would be to explore joint events – workshops, tutorials, etc.

The other community with which more contact would benefit both sides is that of Agents. Many of the manufacturing domains of interest to the Agent community are the same as those to which PLANET technology can be applied. Discussion on the formulation of some common benchmark applications would help to bring researchers together and prevent each community having to absolutely

master the technology of the other.

Workflow Management

This document has presented the PLANET R&D Roadmap for AI Planning and Scheduling applied to Workflow Management. In an applied discipline such as this, a Roadmap must not only identify research challenges, but also match them to current and projected end-user requirements. It must also consider the process by which the results are incorporated into the tools of the trade of the end-users and application developers. Furthermore, necessary preconditions for successful application of the results must be taken into account. This Roadmap is an important step towards a coherent strategy, but is not itself the definitive answer. The Roadmap needs to be a living document that is developed and updated and regular intervals.

This document has made a start on identifying planning techniques and research goals that address the requirements of (intelligent) Workflow Management Systems. In addition to the application of planning and scheduling algorithms we discussed: advantages to be gained from using AI plan representations for processes, ideas from plan execution (especially in uncertain environments), and work on adaptation optimization and metrics. Further work remains to be done, however, to identify specific research goals and projects. Two further topics are also discussed: human issues and infrastructure. It is important to remember that much of the work in a business process is performed by people. Often technology is seen primarily as a means of cutting costs through automation rather than enhancing value by enabling people to work more effectively. The result of treating people like machines is often de-motivation, high staff turnover, loss of productivity, etc. In addition, human qualities are under-utilized. There is a danger that must be guarded against that planning and scheduling techniques may make this situation worse rather than better. The discussion of infrastructure mainly focuses on the need for a reference architecture and interface standards to AI-based software tools to be integrated with each other, with conventional process management software, and with the general organization infrastructure.

The following set of actions will be able to attain some of the objectives (requirements) that were defined in Chapter 3.

- The potential for benefit from applying existing AI P&S techniques to short term requirements should be explored through case studies of large-scale real problems conducted jointly with domain experts and process management experts.
- Generally, links with process management and software engineering communities need to be strengthened by means of interdisciplinary events and other measures.
- To encourage case studies and increase awareness of process management challenges within the planning community.

The TCU main objective was to play a useful role in closing the gap between industry and academic research. However further work is needed:

- to make researchers aware of the real challenges and constraints of the workflow domain;

- to make application and tool developers aware of what AI Planning and Scheduling research has to offer;

- to address practical issues of integrating planning and scheduling technology into suites of application software, and of making the techniques usable by typical software engineers, analysts, etc.

- to form a consensus on medium and long term research goals. The Roadmap should be seen as a living document and be extended and updated regularly.

Planning for the Web

It should be noted once again that a roadmap is "a living document that should be continuously updated", this is particularly true for the AI P&S for the Web Roadmap. Since the target transfer area is the Internet, unpredictable developments in the web technology, or in general in IT technology, could produce acceleration or change of perspectives in a very short time horizon, and the development of the field could take unknown directions. Nevertheless the transfer of AI P&S results seems to be an effective means for improving the quality of the systems/services offered by the web and ITs.

As noted in Chapter 4 there is a favorable ground for technology transfer to take place, and, in some cases, it already started, at least at the level of prototype applications. This autonomous transfer could benefit and it should be accompanied by measures such as:

- direct actions to promote the visibility of applications which demonstrate the effectiveness of AI P&S for the web, e.g. promotion of spin-off enterprises on the matter;

- specific targeted projects for developments of tools;

- educational initiative (targeted to postdoctoral level, industrial and applied research departments of private companies and public institutions);

- actions for promoting standardization efforts, both in the form of initiatives internal to the planning community, or by joint initiatives (e.g. DAML, WFML) which could multiply the community of developers and the visibility of results (esp. W3C joint initiative);

- promoting projects under the 6th Framework Programme Initiative (e.g. PLANSERVE initiative).

The interrelated emerging technologies, which announce and characterized the Information Society, force individuals to develop new methods of work in order to exploit the ITs at their best, new tools and application should reflect and model this new methods in order to be effective. The vision of the Web as large, evolutionary, massive environment where thousands of activities and users are pursuing their own interrelated-goals seems to find a natural framework into models and results from planning and scheduling research.

AI P&S can convey to the Web, and in a more wider vision to ITs, the flexibility typical of knowledge based technologies and provides models of goal directed change and interaction, which can be exploited in application frameworks such as the web domain. AI P&S is crucial to govern dynamical

aspect of the web such as personalized services, integration and reconfiguration of resources, and adaptation to changes.

The currently growing number of AI P&S applications to the web (e.g. in the area of web resources planning, web service composition, info gathering, etc.) confirms the great potentiality of the field and show that the technology is already mature. On the other hand a great cultural and informative barrier exists since AI P&S is not yet perceived as a standard area: the industries of the sector are often rediscovering or redeveloping problems and solutions (e.g. in the area of workflow management, web service integration, etc.) already formalized and investigated from the AI P&S research community.

The proposed technology transfer represents a peculiar challenge because the target industry, i.e. the web industry, and more generally the ITs, is not long established, not well defined and it is moving and reshaping itself very fast. In this area "future emerging trends" are often unpredictable and expressions like "long term scenario", should likely refer to a few years term, or sometimes are likely to mean "months". A technology transfer roadmap in such a dynamical area can be successful only if a timed reaction is provided to the emerging themes, which could likely lead to a deep revision of the Roadmap itself. Nevertheless we believe that the proposed actions could foster and drive a technology transfer, which, at some extent, has already started.

Robot Planning

Robot planning is the oldest and longest standing application of AI planning techniques. In recent years the field has gained more and more importance because of the appearance of different kinds of mobile robots in AI research groups. We have seen exciting demonstrations of the technology including the control of spacecrafts, autonomous helicopters, and tour guide robots. The importance of the field will continue to rise through the development of manipulation and interaction capabilities for these robots.

The defining characteristics of plan-based control is that different aspects of plan-based control, namely plan representation, reasoning, learning, and plan execution cannot be studied in isolation but must be considered in conjunction with the aspects of plan-based control. The advantage of this approach is that we can exploit synergies between the different aspects of plan-based control. Even more important is that plan-based control cannot be researched without contemplating the behavior of the robots that the plans produce. Consequently, research on plan-based robot control must be in the context of controlling autonomous robots.

These constraints for research on robot planning have important consequences for researchers in the field, robot vendors, and funding agencies: the focus must be the study of plan-based control as a resource for autonomous robot control. Therefore, we need autonomous robot laboratories, stronger cooperation between research groups, and the scientific valuation of integrated system research. Second, robot vendors should provide robot platforms with standardized low-level application programmers' interfaces. Third, funding agencies should provide financial support for longer term multi research lab projects. The combination of these activities will provide the basic ground for breakthroughs in the field and the technology transfer. These breakthroughs are needed to solve a number of crucial societal problems including the assistance of elderly people and the exploration of space and other planets, and the search for extra-terrestial life.

To sum up, in order to accelerate the progress in the area of robot planning and plan-based robot control:

Robot vendors should provide robot platforms for research, make parts of their code available to research groups. In return, the code developed in the research groups should be published as repositories that can then also be used by the robot vendors.

Funding agencies should support the building of research laboratories for autonomous robots, provide funding for longer term research projects (with a startup phase for building the infrastructure), and support multi university research initiatives with common code repositories.

Researchers should seek the collaboration with other research groups in order to tackle the challenge problems stated in this Roadmap. They should also encourage work on integrated robot control systems where plan-based robot control is employed and provide publication platforms for this kind of research. Finally, the research community should work on better experimental methods and testbeds that enable the researchers to better evaluate the claims of their work.

Aerospace Applications

This Aerospace Applications Roadmap has surveyed the state-of-the-art in the aerospace domain, showing that it consists of the following sectors, each with their own characteristics and AI P&S-related needs: Aerospace manufacturing, Aircraft operations, Spacecraft operations, Air Traffic Management, Airport operations, Maintenance, and Repair and Overhaul (MRO).

Given the diversity of the aerospace sectors, the AI P&S technologies, and the PLANET organizations, it is difficult to define a single set of steps towards technology transfer. Broad-brush indications can be given. For example, it would be best to build on the existing relationships between ESA and PLANET. Eurocontrol should be approached next, with Airbus following. In each case, the maximum use should be made of the channels provided by the EU.

Some general guidelines can be given to PLANET member organizations, as follows:

- Gain domain knowledge by focusing on a small subset of aerospace sectors.

- Introduce AI P&S incrementally by small steps to reduce risk.

- Start with tactical planning and gradually shorten the time-horizon as successes are booked.

- Consider the non-functional features of AI P&S systems.

- Build confidence by running AI P&S systems side-by-side with the existing systems.

PLANET member organizations with experience in SOP generation in the petrochemical industry should migrate this experience to the aerospace domain.

We recommend that AI P&S research at the national and European levels should be initiated and intensified into:

- Multi-level architectures containing AI P&S components, including distributed, real-time and collaborative AI P&S.

- Knowledge engineering methods and tools for AI P&S.

- Delivery of AI P&S techniques in the form of web services.

- Mixed-initiative and usability techniques for AI P&S.

- Verification and validation of AI P&S.

- Standardization of AI P&S representations and algorithms.

We recommend that the PLANET Project Officer should support the PLANSERVE and DCIRM FP6 proposals, aimed at furthering research into delivering AI P&S web services and into an AI P&S-based, real-time, distributed and collaborative architecture for dynamic risk management.

We recommend that PLANET member organizations with experience of SOP generation techniques in the petrochemical industry should migrate the technology into the aerospace domain.

We recommend that interested PLANET member organizations should work together to build on the ESA invitation to become ESA's center of excellence in AI P&S, and to seek the EU's help in approaching Eurocontrol and Airbus.

We recommend that PLANET member organizations should take note of the guidance in transferring AI P&S technologies into aerospace applications.

Knowledge Engineering

Chapter 7 of the Roadmap pointed out some general problems to be overcome. These include the following:

Little knowledge of the creation, validation and maintenance of large domain models exists in the planning community. There is a general lack of experience (especially in Europe) of knowledge engineering concepts and methods in this area, and we have much to learn and techniques to import from related knowledge engineering work.

The evaluation of KE tools and methods is problematic, and tends to be harder than the evaluation of, say, a planning algorithm. This is seen as a barrier to future research as researchers find it harder to publish in the planning literature.

We need to learn how to characterize domains, and hence build up knowledge about how to match up planning technology with domain models.

KE involves a group of stakeholders with differing backgrounds and knowledge of computing. In particular, it includes consideration of Human Factors.

Finally, planning research has a history of association with toy problems where KE issues are not relevant. There has been a focus in the planning community on theoretical aspects of techniques that do not scale up to real-world planning problems.

The problems lead to some general actions that the community should carry out. These are summarized in Figure 7.2 on page 201.

1. Encourage the planning research community to develop greater appreciation for the applied end of the research continuum.

2. Survey the areas of knowledge acquisition, machine learning, requirements engineering and formal methods in software engineering for tools, techniques, and methods that may be relevant to planning.

3. Distill experience and induce general methods from the experience of previous applications of planning technology.

4. Attempt to build basic, integrated engineering environments for building planning applications. This may also involve promoting standards and/or formalisms that enable integration of planning engines and support tools developed separately.

5. Find a research methodology that allows efficient evaluation of work in KE to allow a route to publication and hence attract more research effort.

6. Build on previous work to create a planner taxonomy usable by knowledge engineers.

7. Develop a classification system and/or vocabulary for describing the characteristics of domains)

Online Planning and Scheduling

The most difficult work of the TCU consisted in the identification of the most relevant and promising directions of research which would allow online planning and scheduling applications to be correctly, optimally, and efficiently dealt with. Among all the directions we considered, we would like to enlighten the ones in the following paragraphs.

As it has been previously pointed out, research might be usefully focused on the study of proactive methods, which use models of the possible changes to produce robust and flexible solutions; among these methods, we think that a special focus should be done on the extension of the notion of controllability from temporal features to other features like resource levels or discrete system states.

Useful connections might be done between these studies and similar or concurrent studies in other communities like the ones on sequential decision making (Markov decision processes), uncertainty representation (Bayesian networks), constraint satisfaction and boolean satisfiability.

Finally, each work on online planning and scheduling has considered one feature of the problem (reactivity, robustness, flexibility, or stability) and tried to improve existing methods with regard to this particular feature; it might be certainly interesting now to revisit the online planning and scheduling problem as a whole, with all its features, all its requirements, and all their interactions, in order to make progress with regard to:

1. the production of robust, flexible, and stable solutions; the tradeoffs between robustness, flexibility, stability, and optimality;

2. the redefinition of the utility of a solution as a function of its quality and of the time at which it is delivered to the execution module (decreasing function of this time); the tradeoffs between the complexity of the online reasoning and the utility of the resulting solutions; the online control of the reasoning, taking into account solution quality and utility; the use of the results of previous studies on resource and time-bounded reasoning;

3. the tradeoffs between offline preprocessing and online reasoning; the limits of an offline pre-processing in terms of time and space.

We would like to plead for stronger connections between scientific communities and any initiative which would favor it. Indeed, we think that the AI Planning and Scheduling community, and more specifically all the people who work on online planning and scheduling, might take advantage of common studies with people involved in connected communities and particularly in the following ones:

- the operations research, constraint satisfaction, and boolean satisfiability communities: these connections have been existing from a long time and only need to be more systematically encouraged;

- the people who work on reasoning and decision under uncertainty (Bayesian networks, influence diagrams, Markov decision processes); this is justified by the fact that, on the one hand, Markov decision processes use basic results from decision theory, produce policies rather than plans, but use explicit representations of the states, actions and probabilities which make them unusable as soon as one wants to go out from small toy problems, and that, on the other hand, planning and scheduling produce plans rather than policies, use implicit representations which make them efficient, but do not correctly represent and reason about uncertainty;

- the people who work on real-time system control, discrete-event system control, and validation; this is justified by the fact that, on the one hand, these people are used to represent, simulate, and validate dynamic systems by using basic frameworks such as synchronous languages, automata, Petri nets, or others, but only consider very simple automatic systems, without any reasoning, planning, and decision capabilities, and that, on the other hand, planning and scheduling people try to build autonomous systems which involve reasoning, planning, and decision capabilities, but have some difficulties trying to deal with real-time constraints.

A. Further Reading and Links

Conferences proceedings

- S. Steel and R. Alami, editors. *Recent Advances in AI Planning, Proceedings of the 4th European Conference on Planning (ECP-97)*, volume 1348 of *Lecture Notes in Computer Science*, Toulouse, France, Sept. 24–26, 1997. Springer Verlag, Berlin, Heidelberg, New York, etc.

- S. Biundo and M. Fox, editors. *Recent Advances in AI Planning, Proceedings of the 5th European Conference on Planning (ECP-99)*, volume 1809 of *Lecture Notes in Computer Science*, Durham, United Kingdom, Sept. 8–10, 2000. Springer Verlag, Berlin, Heidelberg, New York, etc.

- A. Cesta and D. Borrajo, editors. *Recent Advances in AI Planning, Proceedings of the 6th European Conference on Planning (ECP-01)*, *Lecture Notes in Computer Science*, Toledo, Spain, Sept. 12–14, 2001. Springer Verlag, Berlin, Heidelberg, New York, etc.

- R. Simmons, M. Veloso, and S. Smith, editors. *Proceedings of the 4th International Conference on Artificial Intelligence Planning Systems (AIPS-98)*, Pittsburgh, PA, USA, June 7–10, 1998. AAAI Press, Menlo Park, California, USA.

- S. Chien, R. Kambhampathi, and C. Knoblock, editors. *Proceedings of the 5th International Conference on Artificial Intelligence Planning Systems (AIPS-2000)*, Breckenridge, CO, USA, Apr. 14–17, 2000. AAAI Press, Menlo Park, California, USA.

- M. Ghallab, J. Hertzberg, P. Traverso, editors. *Proceedings of the 6th International Conference on Artificial Intelligence Planning Systems (AIPS-02)*, Toulouse, France, April 23–27, 2002. AAAI Press, Menlo Park, California, USA.

- E. Giunchiglia, N. Muscettola, and D. Nau, editors. *Proceedings of the Thirteenth International Conference on Automated Planning and Scheduling (ICAPS 2003)*, Trento, Italy, June 9–13, 2003. AAAI Press, Menlo Park, California, USA.

Books

- J. Allen, J. Hendler, and A. Tate, editors. *Readings in Planning*. Morgan Kaufmann, San Mateo, CA, 1990. ISBN 1-55860-130-9

- M. Zweben and M. Fox, editors. *Intelligent Scheduling*. Morgan Kaufmann, San Mateo, CA, 1994. ISBN 1-55860-260-7.

- M. Ghallab, D. Nau, and P. Traverso. *Automated Planning: Theory and Practice*. Morgan Kaufmann, 2003. To appear.

PLANET material

- S. Biundo, editor. *PLANET News: Newsletter of the European Network of Excellence in AI Planning*. Ulm, Germany. ISSN 1610-0204.

- The PLANET Websites at `http://www.planet-noe.org` [SERVICE] [REPOSITORIES & RESEARCH SITES]

- The PLANET Automated Planning and Scheduling Curriculum

Overview papers

- D. Weld. *Recent Advances in AI Planning*. Technical Report TR-98-10-01, University of Washington, Department of Computer Science and Engineering, Oct. 1998.

- D. Weld. An Introduction to Least Commitment Planning, *AI Magazine*,15(4):27–61,1994

B. Contributing Stakeholders

The following authors contributed to this Technological Roadmap on AI Planning and Scheduling:

Rachid Alami (LAAS-CNRS, Toulouse, France),

Ricardo Aler (University Carlos III de Madrid, Spain),

Ruth Aylett (Salford University, UK),

Michael Beetz (Technical University of Munich, Germany),

Susanne Biundo (University of Ulm, Germany),

Daniel Borrajo (University Carlos III de Madrid, Spain),

Luis Castillo (University of Granada, Spain),

Amedeo Cesta (ISTC-CNR, Rome, Italy),

Alexandra Coddington (University of Durham, UK),

Patrick Doherty (Linköping University, Sweden),

Brian Drabble (OnTarget Technologies, USA),

Patrick Fabiani (ONERA, Toulouse, France),

Maria Fox (University of Durham, UK),

Massimiliano Garagnani (The Open University),

Simon De Givry (Thales LCR, Orsay, France),

Miguel Angel Garcia (University Rovira i Virgili, Tarragona, Spain),

Malik Ghallab (LAAS-CNRS, Toulouse, France),

Maria Gini (University of Minnesota, USA),

Tim Grant (Atos Origin, Netherlands),

Patrik Haslum (Linköping University, Sweden),

Joachim Hertzberg (Fraunhofer, Germany),

Henk Hesselink (National Aerospace Laboratory, Netherlands),

Jim Hutton (British Telecom Laboratories, UK),

Felix Ingrand (LAAS-CNRS, Toulouse, France),

Peter Jarvis (SRI International, Menlo Park, USA),

Paul Kearney (British Telecom, Ipswich, UK),

Jonas Kvarnström (Linköping University, Sweden),

David Lesaint (British Telecom, Paris, France),

Vince Long (Troy Associates Ltd., UK),

Roger Mampey (ONERA, Toulouse, France),

Nicola Matino (Centro Ricerche Fiat, Turin, Italy),

Nikolay Mehandjiev (University of Manchester, UK),

Lee McCluskey (University of Huddersfield, UK),

Yannick Meiller (ONERA, Toulouse, France),

Alfredo Milani (University of Perugia, Italy),

José Manuel Morina (Universidad Carlos III de Madrid, Spain),

Martha Pollack (University of Michigan, USA),

Ioannis Refanidis (University of Macedonia),

Maria Dolores Rodrigues Moreno (University Carlos III of Madrid, Spain),

Alessandro Saffiotti (Örebro University, Sweden),

Araceli Sanchis (Universidad Carlos III de Madrid, Spain),

C. The Members of PLANET

Austria
- XIMES GmbH,
  Johannes Gärtner,
  `gaertner@ximes.com`

Belgium
- Robonetics NV,
  Filip Verhaeghe,
  `filip.verhaeghe@roboentics.com`
- Space Applications Services (SAS),
  Richard Aked,
  `ra@sas.be`

Bulgaria
- Institute for Information Theories and Applications (FOI ITHEA),
  Krassimir Markov,
  `foi@nlcv.net`

Cyprus
- University of Cyprus,
  Yannis Dimopoulos,
  `yannis@cs.ucy.ac.cy`

Czech Republic
- Charles University, Praha,
  Roman Barták,
  `bartak@kti.mff.cuni.cz`

France
- COSYTEC S.A.,
  Abderrahmane Aggoun,
  `abderrahmane.aggoun@cosytec.com`
- ILOG S.A.,
  Philippe Laborie,
  `laborie@ilog.fr`

- (LAAS-CNRS), Toulouse,
  Malik Ghallab,
  `malik@laas.fr`
- Laboratoire d' Informatique Marseille (LIM-CNRS),
  Camilla Schwind,
  `schwind@lim.univ-mrs.fr`
- MASA Group,
  Emmanuel Chiva,
  `emmanuel.chiva@masagroup.net`
- ONERA Systems Control and Flight Dynamics Department,
  TCU On-line Planning and Scheduling,
  Gérard Verfaillie,
  `Gerard.Verfaillie@cert.fr`
- THOMSON-CSF,
  Simon De Givry,
  `simon.degivry@thalesgroup.com`
- Université Technologique de Troyes,
  Christophe Doniat,
  `christophe.doniat@utt.fr`

Germany
- University of Ulm,
  Coordinating Node,
  Susanne Biundo,
  `biundo@informatik.uni-ulm.de`
- Anite Systems GmbH,
  Marc Niezette,
  `Marc.Niezette@anitesystems.de`
- PLASIM Planungs- und Simulationsbüro,
  Jens Klußmann,
  `klu@plasim.de`

- Aachen University of Technology,
  Gerhard Lakemeyer,
  `gerhard@cs.rwth-aachen.de`

- University of Bonn,
  Armin Cremers,
  `abc@informatik.uni-bonn.de`

- Bremer Institut für Betriebstechnik und ange-
  wandte Arbeitswissenschaft (BIBA),
  Hartmut Höhns,
  `hoe@biba.uni-bremen.de`

- Darmstadt University of Technology,
  Ulrich Scholz,
  `scholz@informatik.tu-darmstadt.de`

- German Research Center for Artificial Intelli-
  gence (DFKI),
  Mathias Bauer,
  `mathias.bauer@dfki.de`

- University of Freiburg,
  Bernhard Nebel,
  `nebel@informatik.uni-freiburg.de`

- Fraunhofer - Autonomous intelligent Systems
  (AiS),
  Joachim Hertzberg,
  `hertzberg@ais.fraunhofer.de`

- Technical University of Munich,
  TCU Robot Planning,
  Michael Beetz,
  `Michael.Beetz@informatik.tu-muenchen.de`

- Siemens AG,
  Wendelin Feiten,
  `wendelin.feiten@mchp.siemens.de`

Greece
- University of Macedonia, Thessaloniki,
  Ioannis Refanidis,
  `yrefanid@uom.gr`

- Aristotle University of Thessaloniki,
  Ioannis Vlahavas,
  `vlahavas@csd.auth.gr`

- Foundation for Research and Technology -
  Hellas (ICS-FORTH),
  Dimitrios Plexousakis,
  `dp@csi.forth.gr`

- National Centre for Scientific Research
  "Demokritos",
  Constantine Spyropoulos,
  `costass@iit.demokritos.gr`

- Technical University of Athens (ICCS),
  Spyros Tzafestas,
  `tzafesta@softlab.ece.ntua.gr`

- Technical University of Crete,
  Manolis Koubarakis,
  `manolis@ced.tuc.gr`

- University of Ioannina,
  Chrysostomos Stylios
  `stylios@cs.uoi.gr`

Hungary
- Computer and Automation Research Institute
  Hungarian Academy of Sciences (MTA SZ-
  TAKI),
  László Monostori,
  `laszlo.monostori@sztaki.hu`

Ireland
- University College Cork, National University
  of Ireland,
  Ken Brown,
  `k.brown@cs.ucc.ie`

Israel
- Ben Gurion University,
  Ronen Brafman,
  `brafman@cs.bgu.ac.il`

Italy
- DIEE - University of Cagliari,
  Giuliano Armano,
  `armano@diee.unica.it`

- DEIS - University of Bologna,
  Paola Mello,
  `pmello@deis.unibo.it`

- DIST - University of Genoa,
  Enrico Giunchiglia,
  `Enrico@dist.unige.it`

- Consiglio Nazionale delle Ricerche - Istituto di Psicologia (IP-CNR),
  TCU Aerospace Applications,
  Amedeo Cesta,
  `cesta@ip.rm.cnr.it`
- University of Perugia,
  TCU Planning & Scheduling for the Web,
  Alfredo Milani,
  `milani@dipmat.unipg.it`
- Istituto per la Ricerca Scientifica e Tecnologia (IRST), Paolo Traverso,
  `traverso@irst.itc.it`
- University of Parma,
  Agostino Poggi,
  `poggi@ce.unipr.it`
- University of Brescia,
  Alfonso Gerevini,
  `gerevini@ing.unibs.it`

The Netherlands
- Atos Origin Nederland B.V.,
  TCU Aerospace Applications,
  Tim Grant,
  `Tim.Grant@atosorigin.com`
- Delft University of Technology,
  Cees Witteveen,
  `witt@cs.tudelft.nl`
- NLR – National Aerospace Laboratory,
  Henk Hesselink,
  `hessel@nlr.nl`

Portugal
- Instituto Superior de Engenharia do Porto ISEP/IPP,
  João Rocha,
  `jrocha@ipp.pt`

Slovenia
- University of Maribor,
  Peter Kokol,
  `kokol@uni-mb.si`

Spain
- iSOCO S.A.,

Antonio Reyes Moro,
`toni@isoco.com`
- Technical University of Catalonia,
  Lluís Vila,
  `vila@lsi.upc.es`
- University of Granada,
  Luis Castillo,
  `L.Castillo@decsai.ugr.es`
- University Carlos III of Madrid,
  TCU Workflow Management,
  Daniel Borrajo,
  `dborrajo@ia.uc3m.es`
- Universitat Politécnica de Catalunya,
  Tom Creemers,
  `creemers@iri.upc.es`
- Universidad Politecnica de Valencia,
  Eva Onaindia,
  `onaindia@dsic.upv.es`
- Universitat Rovira i Virgili, Tarragona,
  Miguel Angel Garcia,
  `magarcia@etse.urv.es`

Sweden
- Linköping University,
  Patrick Doherty,
  `patdo@ida.liu.se`
- Örebro University,
  Alessandro Saffiotti,
  `alessandro.saffiotti@aass.oru.se`

United Kingdom
- BAE Systems Ltd.,
  Andrew Burgess,
  `andrew.burgess@baesystems.com`
- British Telecommunications,
  David Lesaint,
  `david.lesaint@bte.bt.com`
- University of Essex,
  Sam Steel,
  `sam@essex.ac.uk`
- University of Edinburgh,
  John Levine,
  `johnl@aiai.ed.ac.uk`

- University of Huddersfield,
  TCU Knowledge Engineering,
  Lee McCluskey,
  `t.l.mccluskey@zeus.hud.ac.uk`

- University of Manchester,
  Nikolay Mehandjiev,
  `Nikolay.Mehandjiev@co.umist.ac.uk`

- The Open University Walton Hall,
  Massimiliano Garagnani,
  `M.Garagnani@open.ac.uk`

- Salford University,
  TCU Intelligent Manufacturing,
  Ruth Aylett,
  `R.S.Aylett@iti.salford.ac.uk`

- Troy Associates Ltd.,
  Vince Long,
  `vlong@troyassoc.com`

- University of Durham,
  Julie Porteous,
  `j.m.porteous@durham.ac.uk`

Associated Members
- Norman Sadeh,
  `sadeh@cs.cmu.edu`,
  Carnegie Mellon University

- Peter Jarvis,
  `Jarvis@ai.sri.com`,
  SRI International

- Brian Drabble,
  `brian_drabble@yahoo.com`,
  On Target Technologies, Inc

- Sylvie Thiebaux,
  `Sylvie.Thiebaux@anu.edu.au`,
  The Australian National University