O-Plan: a Knowledge-Based Planner and its Application to Logistics

Austin Tate, Brian Drabble and Jeff Dalton
Artificial Intelligence Applications Institute
The University of Edinburgh
80 South Bridge, Edinburgh EH1 1HN, UK
{a.tate, b.drabble, j.dalton}@ed.ac.uk

Abstract

O-Plan is a command, planning and control architecture with an open modular structure intended to allow experimentation on, or replacement of, various components. The research is seeking to determine which functions are generally required in a number of application areas and across a number of different command, planning, scheduling and control systems.

O-Plan aims to demonstrate how a planner, situated in a task assignment and plan execution (command and control) environment, and using extensive domain knowledge, can allow for flexible, distributed, collaborative, and mixed-initiative planning. The research is seeking to verify this total systems approach by studying a simplified three-level model with separable task assignment, plan generation and plan execution agents.

O-Plan has been applied to logistics tasks that require flexible response in changing situations.

Summary

The O-Plan research and development project is seeking to identify re-usable modules and interfaces within planning systems which will enable such systems to be tailored or extended quickly to meet new requirements. A common framework for representing and reasoning about plans based on the manipulation of constraints underlies the model used by the architecture. Within this framework, rich models of an application domain can be provided to inform the planner when creating or adapting plans for actual use.

A number of important foundations have been laid for flexible planning work in the future. They are:

- A view of the planner as *situated* in the context of task assignment, plan execution and change.
- A simple abstract architecture based on an agenda of "issues" from which items can be selected for processing. The processing takes place on an available computational platform (human or machine), with the appropriate functional capabilities described as knowledge sources.

  This architecture allows for independent progress to be made in a number of important areas for successful planning systems, including search control and opportunism, planner capability description, and system resource scheduling.
- A structure that allows separate (often specialised) handlers for different types of constraint to be included, so that the results provide effective overall constraints on the operation of a planner.
- Ways to use domain knowledge, where possible, to constrain the search of a planner.
- The common model of activity, tasks and plans based on a set of constraints – the <I-N-OVA> constraint model. A common model can in turn support systems integration and open up collaboration and distribution opportunities.
- Symmetric interaction by system components and users. Both are seen as manipulating the same set of constraints.
- An approach to the user interface of a planner, based on Plan and World Views.

The O-Plan planner is general purpose and applies to a wide variety of important application areas. Its current application to military logistics planning tasks is described.

O-Plan – the Open Planning Architecture

The O-Plan Project at the Artificial Intelligence Applications Institute of the University of Edinburgh is exploring a practical computer-based environment that provides for the specification, generation and execution of activity plans, and for interaction with such plans. O-Plan is intended to be a domain-independent general planning and control framework with the ability to employ detailed knowledge of the domain. See [Allen et. al. 90] for background reading on AI planning systems. See [Currie & Tate] for details of the first version of the O-Plan planner which introduced an agenda-based architecture and the main system components. That paper also includes a chart showing how O-Plan relates to other planning systems. The second version of the O-Plan system adopted a multi-agent approach and situated the planner in a task requirement and plan execution setting [Drabble & Tate 95]. The multi-agent approach taken is described in greater detail in [Tate et. al. 94b].

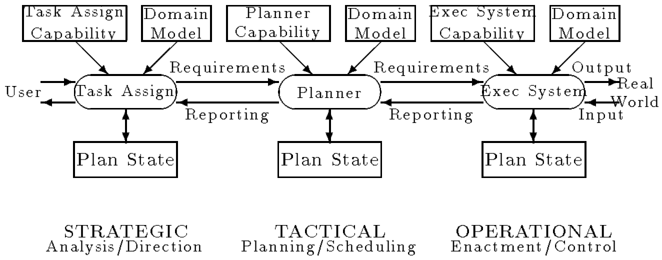

Figure 1: Communication between Strategic, Tactical and Operational Agents

Figure 1 shows the communications between the 3 agents in the O-Plan architecture[1]. A user specifies a task that is to be performed through some suitable interface. We call this process *task assignment*. A *planner* plans to perform the task specified. The *execution system* seeks to carry out the detailed actions specified by the planner while working with a more detailed model of the execution environment. The activities of the three agents may be more or less concurrent.

The O-Plan approach to command, planning, scheduling and control can be characterised as follows:

- successive refinement/repair of a complete plan or schedule which contains an agenda of outstanding issues;
- a least commitment approach;
- opportunistic selection of the focus of attention on each problem-solving cycle;
- incremental tightening of constraints on the plan, performed by "constraint managers", e.g.,
  - time point network manager,
  - object/variable manager,
  - effect/condition manager,
  - resource utilisation manager;
- localised search to explore alternatives where advisable;
- global alternative re-orientation where necessary.

The O-Plan project has sought to identify modular components within an AI command, planning and control system and to provide clearly defined interfaces to these components. The background to this work is provided in [Tate 93b]. The various components plug into "sockets" within the architectural framework. The sockets are specialised to ease the integration of particular types of component. See figure 2.

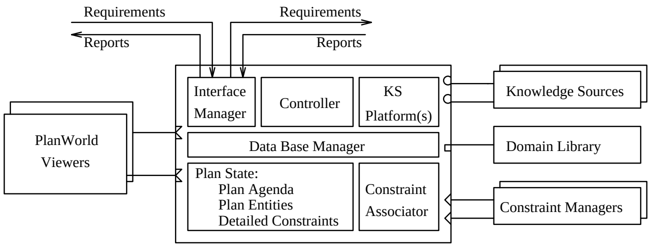

Figure 2: O-Plan Agent Architecture

The components that plug into the O-Plan agent architecture are:

PlanWorld Viewers – User interface, visualisation and presentation viewers for the plan – usually differentiated into technically oriented *plan* views (charts, structure diagrams, etc.) and domain oriented *world* views (simulations, animations, etc.).

Knowledge Sources – Functional components which can analyse, synthesise or modify plans. They provide the *capabilities* of the agent.

Domain Library – A model of the domain, including a library of possible actions. Different models or levels of detail of the model are possible within different agents.

Constraint Managers – Components which manage detailed constraints within a plan and seek to maintain as accurate a picture as possible of the feasibility of the current plan with respect to the domain model.

These plug-in components are orchestrated by an O-Plan agent kernel which carries out the tasks assigned to it via appropriate use of the Knowledge Sources and manages options being maintained within the agent's *Plan State*. The roles of the components are as follows:

Interface Manager – Handles external events (requirements or reports) and, if they can be processed by the agent, posts them on the agent *Agenda*.

Controller – Chooses Agenda entries for processing by suitable Knowledge Sources.

Knowledge Source Platform(s) – Chosen Knowledge Sources are run on an available and suitable Knowledge Source Platform.

Data Base Manager – Maintains the Plan State and provides services to the Interface Manager, Controller and Knowledge Sources.

[1] This simplified view of the environment within which a planner operates helps to clarify the O-Plan research objectives. It is sufficient to ensure that the tasking and execution environments are represented.

Constraint Associator Acts as a mediator between changes to the Plan State made by the Data Base Manager and the activities of the various Constraint Managers that are installed in the agent. It eases the management of interrelationships between the main plan entities and detailed constraints [Tate et. al. 94c].

A Situated Planner – Coordinating Task Assignment, Planning and Plan Execution

The O-Plan project has identified the need for AI planners to be viewed as situated in an environment where planning is one of a number of tasks involved in dealing with the whole problem of task assignment, planning, execution and control. While the planner deals with the plan generation aspect of the problem, other agents may deal with task elicitation, plan analysis, reactive execution, plan repair, etc. Each of these systems has its own perspective on the planning problem and each is capable of communicating in a way which allows other systems to assimilate new information into their perspective of the problem. This view of planners introduces a number of new issues: the role of authority, determining the quality of the plans being generated by other systems and controlling the execution of plans within other situated agents.

The activities of the various agents need to be coordinated, and authority management is viewed as one way in which this can be done [Tate 93a]. For example, in plan generation, it may be necessary to be given authority to work on certain options and to have direction on the level of detail to which a plan should be developed. In plan enactment, it is important to identify (and possibly name) which phases of the plans can be executed and which parts should be held back for further approval.

Current AI planners can generate a solution that satisfies the requirements they are given. Some planners provide facilities to control the quality of the solution to be returned, by using evaluation functions or search-control rules. However, they do not usually integrate plan quality considerations across several plans. In addition, their plan representations may not reflect the plan quality criteria that are necessary in practice. To date the O-Plan system is able to generate plans and communicate them to the EXPECT [Gil 94],[Gil et. al. 94] system for evaluation. Work is continuing to expand the interface between EXPECT and O-Plan to strengthen the support for users in specifying, comparing and refining the constraints on a range of different plan options, at the task assignment level of a planning support environment, and to allow this information to be used directly by O-Plan in guiding it in its search for a good solution.

The O-Plan architecture has been designed to support the creation of agents which are situated in an environment involving communication with other agents, and work to date has concentrated on building generative planning agents and execution agents, with links between them. The results of this research have been used in a number of systems that have drawn on the O-Plan work. For example, the Optimum-AIV [Aarup et. al. 94] system, developed for Assembly, Integration and Verification of spacecraft at the European Space Agency, and now in use for Ariane Launcher preparations, uses concepts from O-Plan's plan representation to support the repair of plans to deal with test failures. As part of the O-Plan research, an associated Ph.D student project explored the creation of a reactive execution agent within the O-Plan agent architecture [Reece 94]. This work also showed the value of using the plan intentions captured in Goal Structure to support effective reactive execution and re-planning [Reece & Tate 94].

Using Domain Knowledge in Planning

O-Plan provides the ability to use domain knowledge about time constraints, resource requirements and other features to restrict the range of plans being considered as feasible solutions to the tasks specified. The O-Plan research programme has studied a number of mechanisms for using such knowledge to prune or prioritise search. These include using temporal constraints [Bell & Tate 85],[Drabble & Kirby 91], resource constraints [Drabble & Tate 94], temporal coherence of conditions [Drummond & Currie 89], and Goal Structure condition type information [Tate 75],[Tate 77].

- **Temporal Constraints** – Each time point referred to in a plan is constrained to have an upper and lower bound on its temporal distance from other time points and from time "zero". The time points held in the Time Point Network (TPN) are indirectly linked to actions and events in a plan - which we refer to as the Associated Data Structure (ADS) [Drabble & Kirby 91]. This ensures that the TPN and entities represented in the ADS can both be independently changed. In addition, the functional interface to the TPN does not reveal the underlying representation, so that a different way of handling time constraints could be substituted.
- **Object/Variable Constraints** – O-Plan uses a rich model of constraints to handle the interactions and dependencies among the different objects and variables, including co-designation (equality), non-codesignation (inequality), scalar (set membership), and numeric range constraints.
- **Resource Constraints** – O-Plan uses a rich model to manage the detailed resource constraints within a plan. The Resource Utilisation Manager (RUM) [Drabble & Tate 94] can handle a number of different resource types and can reason about how resource levels change during the generation of a plan. There are two major resource types supported by the

RUM: consumable resources and reusable resources. Each of these can be further subdivided to model the resources of the domain.

- **Goal Structure and Condition Types** – One powerful means of using domain knowledge to restrict and guide search in a planner is to recognise explicit precondition types, as introduced into Interplan [Tate 75] and Nonlin [Tate 77] and subsequently used in other systems such as Deviser [Vere 81], SIPE–2 [Wilkins 88], and O-Plan [Currie & Tate],[Tate et. al. 94b]. O-Plan and Nonlin Task Formalism (TF) extends the notion of a precondition on an action and mates it with a "process-oriented" view of action descriptions. A TF schema description specifies a method by which some higher level action can be performed (or higher level goal achieved). A detailed description of the use of condition types to inform search in an AI planner is provided in [Tate et. al. 94a]. That paper also compares the use of condition types in O-Plan with a number of other planners.

<I-N-OVA> – Manipulating Plans as a Set of Constraints

The <I-N-OVA>[2] (*Issues – Nodes – Orderings/Variables/Auxiliary*) Model is a means to represent plans as a set of constraints [Tate 95],[Tate 96]. By having a clear description of the different components within a plan, the model allows for plans to be manipulated and used separately to the environments in which they are generated.

Our aim is to characterise the plan representation used within O-Plan [Currie & Tate],[Tate et. al. 94b] and to relate this work to emerging formal analyses of plans and planning. This synergy of practical and formal approaches can stretch the formal methods to cover realistic plan representations, as needed for real problem solving, and can improve the analysis that is possible for production planning systems.

A plan is represented as a set of constraints which together limit the behaviour that is desired when the plan is executed. Work on O-Plan and other practical planners has identified different entities in the plan which are conveniently grouped into three types of constraint. The set of constraints describes the possible plan elaborations that can be reached or generated as shown in figure 3.

The three types of constraint in a plan are:

1. Implied Constraints or "Issues" – the pending or future constraints that will be added to the plan as a result of handling unsatisfied requirements, dealing with aspects of plan analysis and critiquing, etc. The implied constraints are the issues to be addressed, i.e., the "to-do" list or agenda which can be used to decide what plan modifications should be made to a plan by a planner (user or system).

[2] <I-N-OVA> is pronounced as in "Innovate".

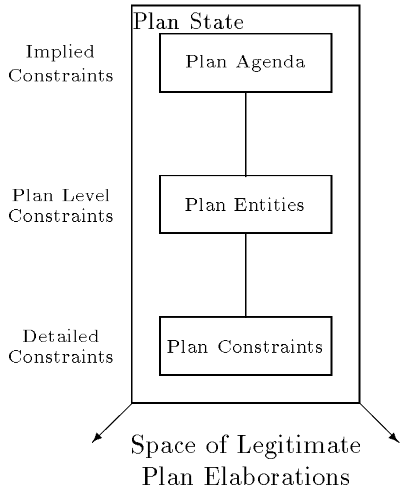

Figure 3: Plan Constraints Define Space of Plan Elaborations

2. Plan Entities or Plan Node Constraints – the main plan entities related to external communication of a plan. They describe a set of external names associated to time points. In an activity planner, the nodes are usually the actions in the plan associated with their begin and end time points. In a resource-centred scheduler, nodes may be the resource reservations made against the available resources with a begin and end time point for the reservation period.
3. Detailed Constraints – specialised constraints on the plan associated with plan entities. Empirical work on the O-Plan planner has identified the desirability of distinguishing two special types of detailed constraint: Ordering or Temporal Constraints (such as temporal relationships between the nodes or metric time properties); and Variable Constraints (co-designation and non-co-designation constraints on plan objects in particular). Other Detailed Constraints relate to input (pre-) and output (post-) and protection conditions, resources, authority requirements, spatial constraints, etc. These are referred to as *Auxiliary Constraints*.

Abstract View of the O-Plan Control Flow

O-Plan operates on a workflow principle, being driven by an agenda of "issues". It is useful to present a simple abstraction of the workflow within such systems.

O-Plan refines a "current state". It maintains one or more *options* within the state for alternative decisions about how to restrict the space of state elaborations which can be reached[3]. The system needs to know what outstanding processing requirements exist in the

[3] State constraint relaxation is also possible to increase the space of state elaborations in some systems.

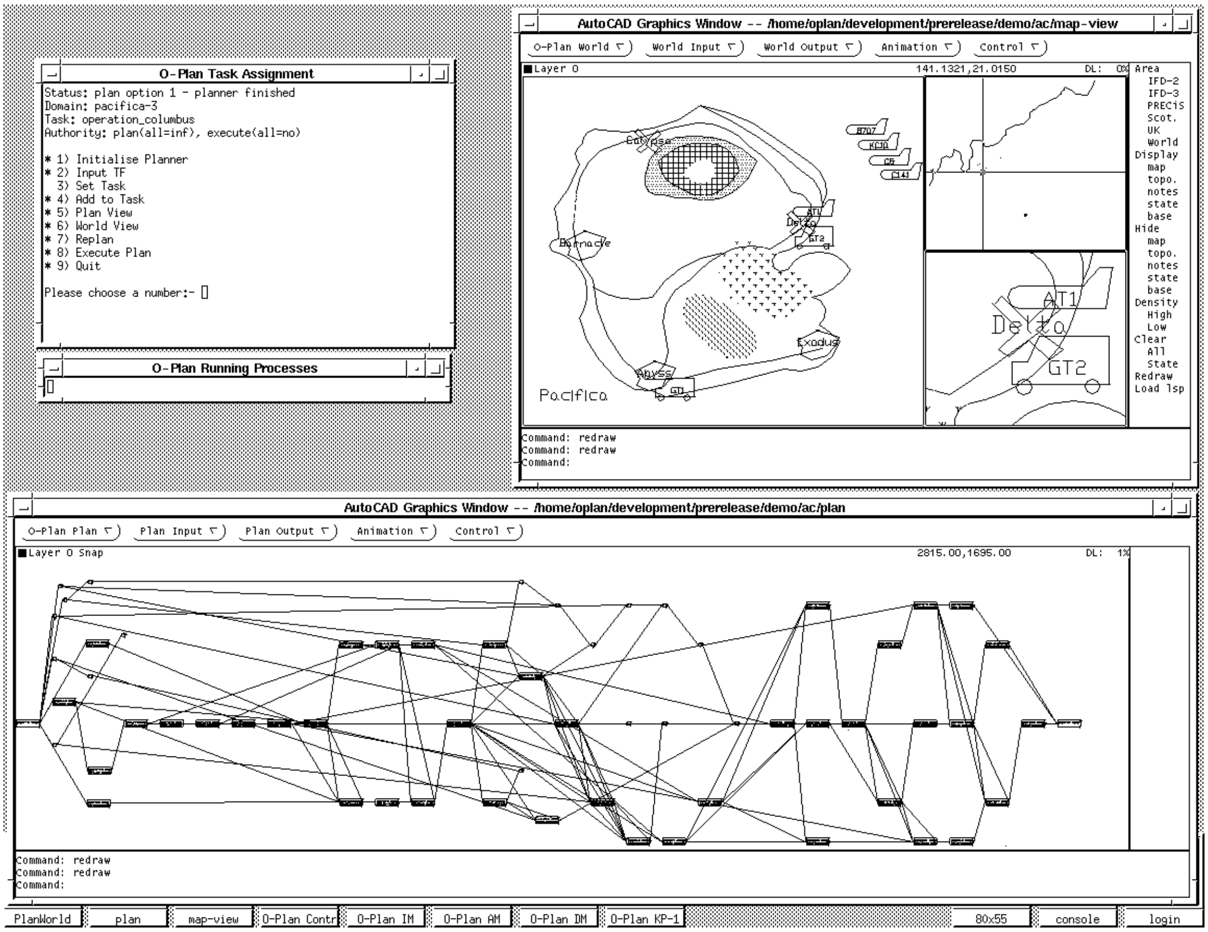

Figure 5: Example Output of the PlanWorld Viewer User Interface

state – the *Agenda of Issues*. These represent the implied constraints on valid future states. One (normally) of these outstanding processing requirements is chosen to be worked upon next (by the *Controller*). This calls up processing capabilities (*Knowledge Sources* or *Issue Handlers*) within the system which can make decisions and modify the State. The modifications can be in terms of definite changes to entities in the state or by noting further processing requirements (as a result of state analysis and critiquing, etc.) on the agenda.

We have found it useful to separate the entities representing the decisions already made during processing into a high level (representing the main entities shared across all planning system components and known to various parts of the system), and more detailed specialised entities (which form a specialised area of the representation of the plan state). These lower level, more compartmentalised, parts can represent specialised constraints within the plan state such as time, resource, spatial and other features. This separation can assist in the identification of opportunities for modularity within the system.

Working with the User

O-Plan is implemented in Common Lisp on Unix Workstations with an X-Windows interface. It is designed to be able to exploit distributed and multi-processor delivery systems in future. An interface to AutoCAD has been built to show the type of User Interface we envisage (see Figure 5). This is called the PlanWorld Viewer Interface [Tate & Drabble 95]. The window in the top left corner shows the Task Assignment menu and supports the management of authority [Tate 93a] to plan and execute plans for a given task. The lower window shows a *Plan View* (such as a graph or a gantt chart), and the upper right window shows a *World View* (for visualisation or simulations of the state of the world at points in the plan). The particular plan viewer and world viewer provided are declared to the system and the interfaces between these and the planner uses a defined interface to which various implementations can conform. O-Plan has been interfaced to a number of Plan and World Viewers including process modelling tools, map-based interfaces and tools that create animation sequences of possible plan execution. The developer interface to O-Plan is not shown to the normal planner user.

Recent work on O-Plan has focussed on the representation and management of constraints in planning, particularly in order to simplify some aspects of the architecture and to act as a mechanism for user/system mixed-initiative planning [Tate 94].

Target Applications for O-Plan

O-Plan is aimed at the following types of problems:

- project management, systems engineering, construction, process flow, integration and verification, etc.
- planning and control of supply and distribution logistics.
- mission sequencing and control of space probes and satellites such as VOYAGER, ERS-1, etc.

These applications fit midway between the large-scale manufacturing scheduling problems found in some industries (where there are often few inter-operation constraints) and the complex *puzzles* dealt with by very flexible logic-based tools. However, the problems of the target type represent an important class of industrial, scientific and engineering relevance.

The architecture itself has wider applicability. For example, it has been used as the basis for the design of the TOSCA manufacturing scheduler in a project for Hitachi [Beck 93].

Crisis Action Planning

The application emphasis of the O-Plan project has been to aid in the definition, generation and enactment of Courses of Action (COAs) within the military crisis action planning process. There are six phases identified in reponding to a crisis are shown in the table.

Phase 1	Situation Development
Phase 2	Crisis Assessment
Phase 3	COA Development: O-Plan provides support in the development of COAs and in estimating the feasibility of the generated COAs. This is the main contribution of the project.
Phase 4	COA Selection: O-Plan provides support in the refinement and presentation of COAs.
Phase 5	Execution Planning
Phase 6	Execution

The O-Plan research principally addresses phases three through six. AIAI has also worked with a number of groups on the representations of plans which can be used to communicate across the different phases and agents involved in the crisis planning process.

Crisis action planning has provided the focus for recent O-Plan applications with problems being tested in the PRECiS domain [Reece et. al. 93] and a simplified version of Integrated Feasibility Demonstration scenario number 2 (IFD-2) from the ARPA/Rome Laboratory Planning Initiative [Fowler et. al. 95]. These test domains allow for realistic, and military-relevant, scenarios and issues to be addressed in a setting suitable for research and development. Crisis action planning calls for plans to be developed which are flexible, robust and responsive to changing task requirements and changes in the operational situation. Current planning aids are too inflexible.

Current military planning systems usually allow only one COA to be fully thought through, and any alternatives are seen as poor relations. This is due to the fixed-step nature of the process: it is not viewed as an iterative process in which several sources of knowledge and techniques (e.g., tasking, planning, scheduling, resourcing and repairing) can be brought in as and when required. A more flexible planning framework may allow military planners to be freed from a step-by-step approach to consider more options and constraints where appropriate within the planning process.

PRECiS/Pacifica Domain

The principal development of O-Plan has been motivated by applications related to logistics, transportation planning/scheduling problems and Non-combatant Evacuation Operations (NEOs). The testbed is provided by the PRECiS (Planning, Reactive Execution and Constraint Satisfaction) environment. It defines the data and hypothetical background for logistics planning and reacting scenarios which can be used for demonstration and evaluation purposes.

The definition of the PRECiS environment has drawn on work by several people: Brown at Mitre Corporation to describe a realistic NEO scenario for the Planning Initiative's Integrated Feasibility Demonstration Number 3 (IFD-3); Reece and Tate to define an openly accessible fictional environment based on the island of Pacifica [Reece & Tate 93] suitable for enabling technology researchers interested in planning and reactive execution of plans; and Hoffman and Burnard at ISX Corporation to produce a cut-down demonstration scenario suitable for transportation scheduling research experiments within the ARPA/Rome Laboratory Planning and Scheduling Initiative. The results have been provided in a publicly available document [Reece et. al. 93] and other materials.

Four primary needs of the ARPA/Rome Laboratory Planning and Scheduling Initiative are met by the PRECiS environment.

1. Realistic scenarios can be explored from the data provided in the environment for COA generative planning, case based reasoning, transportation

scheduling and the reactive execution of plans.

2. Requirements of "tier-1" enabling researchers are sufficiently met by the data in order for them to pursue their individual research programmes.
3. Entities in the environment are hypothetical and do not reflect actual peoples and locations, yet are realistic in the types of data that would normally be available.
4. The scenario and domain descriptions are not confidential or military critical. They can be openly demonstrated and publications can be based upon them. This is important for enabling researchers.

Work on the PRECiS environment and the Pacifica island model has continued. Map viewers and simulators are now available for demonstration and evaluation purposes. O-Plan has been demonstrated developing Non-combatant Evacuation Operation (NEO) plans in this environment and a reactive execution agent (REA) based on the O-Plan architecture has been used to reactively modify plans to respond to operational demands in a simulation of the Pacifica island in the context of a NEO.

Acknowledgements

The O-Plan project is sponsored by the Advanced Research Projects Agency (ARPA) and Rome Laboratory, Air Force Materiel Command, USAF, under grant number F30602-95-1-0022. The O-Plan project is monitored by Dr. Northrup Fowler III at the USAF Rome Laboratory. The U.S. Government is authorised to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation hereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing official policies or endorsements, either express or implied, of ARPA, Rome Laboratory or the U.S. Government.

References

[Aarup et. al. 94] Aarup, M., Arentoft, M.M., Parrod, Y., Stokes. I., Vadon, H. and Stader, J. *Optimum-AIV: A Knowledge-Based Planning and Scheduling System for Spacecraft AIV*, in Intelligent Scheduling (eds. Zweben, M. and Fox, M.S.), Morgan Kaufmann, San Francisco, 1994.

[Allen et. al. 90] Allen, J., Hendler, J. and Tate, A., *Readings in Planning*, Morgan Kaufmann, Palo Alto, 1990.

[Beck 93] Beck, H., TOSCA: A Novel Approach to the Management of Job-shop Scheduling Constraints, Realising CIM's Industrial Potential: Proceedings of the Ninth CIM-Europe Annual Conference, pages 138-149, (eds. Kooij, C., MacConaill, P.A., and Bastos, J.), 1993.

[Bell & Tate 85] Bell, C.E. and Tate, A., Using Temporal Constraints to Restrict Search in a Planner, Paper presented to the Third UK Planning SIG Workshop, Sunningdale, Oxon, UK. Proceedings published by the Institution of Electrical Engineers, London, January 1985.

[Currie & Tate 91] Currie, K.W. and Tate, A., O-Plan: the Open Planning Architecture, *Artificial Intelligence* 52(1), pp. 49-86, North-Holland, 1991.

[Drabble & Kirby 91] Drabble, B. and Kirby, R., Associating A.I. Planner Entities with an Underlying Time Point Network, Proceedings of the First European Workshop on Planning (EWSP-91), Springer-Verlag Lecture Notes in Artificial Intelligence No 522, 1991.

[Drabble & Tate 94] Drabble, B. and Tate, A., The Use of Optimistic and Pessimistic Resource Profiles to Inform Search in an Activity Based Planner, Proceedings of the Second International Conference on AI Planning Systems (AIPS-94), AAAI Press, Chicago, USA, June 1994.

[Drabble & Tate 95] Drabble, B. and Tate, A., O-Plan: A Situated Planning Agent, Proceedings of the Third European Workshop on Planning (EWSP'95), Assisi, Italy, September, 1995. In *New Directions in Planning*, (eds. Ghallab. M. and Milani, A.), Frontiers in AI and Applications Series, No. 31, IOS Press, Amsterdam, 1995.

[Drummond & Currie 89] Drummond, M. and Currie, K. Exploiting Temporal Coherence in Nonlinear Plan Construction, in Proceedings of the International Joint Conference on Artificial Intelligence IJCAI-89, Detroit, USA, 1989.

[Fowler et. al 95] Fowler, N., Cross, S.E. and Owens, C. The ARPA-Rome Knowledge-Based Planning and Scheduling Initiative, IEEE Expert: Intelligent Systems and their Applications, Vol. 10, No. 1, pp. 4-9, February 1995, IEEE Computer Society.

[Gil 94] Gil, Y. *Knowledge Refinement in a Reflective Architecture*, in the proceedings of the Twelfth National Conference on Artificial Intelligence, Seattle, WA, USA. August 1994. Published by AAAI Press/ The MIT Press Menlo Park, CA, USA.

[Gil et. al. 94] Gil, Y., Tate, A. and Hoffman, M., Domain-Specific Criteria to Direct and Evaluate Planning Systems, Proceedings of the ARPA/Rome Laboratory Planning Initiative Workshop, (ed. Burstein, M.), Morgan Kaufmann, 1994.

[Reece 94] Reece, G.A., Characterization and Design of Competent Rational Execution Agents for Use in Dynamic Environments, Ph.D Thesis, Department of Artificial Intelligence, University of Edinburgh, November 1994.

[Reece & Tate 93] Reece, G.A. and Tate, A. The Pacifica NEO Scenario, Technical Paper ARPA-RL/O-

Plan/TP/3, March 1993.

[Reece et. al. 93] Reece, G.A., Tate, A., Brown, D. and Hoffman, M., The PRECiS Environment, Paper presented at the ARPA-RL Planning Initiative Workshop at AAAI-93, Washington D.C., July 1993. Also available as University of Edinburgh, Artificial Intelligence Applications Institute Technical Report AIAI-TR-140.

[Reece & Tate 94] Reece, G.A. and Tate, A., Synthesizing Protection Monitors from Causal Structure, Proceedings of the Second International Conference on AI Planning Systems (AIPS-94), AAAI Press, Chicago, USA, June 1994.

[Tate 75] Tate, A., Using Goal Structure to Direct Search in a Problem Solver. Ph.D. Thesis, University of Edinburgh, 1975.

[Tate 77] Tate, A., Generating Project Networks, Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI-77), Cambridge, Mass., USA, 1977.

[Tate 93a] Tate, A., Authority Management - Coordination between Planning, Scheduling and Control, Workshop on Knowledge-based Production Planning, Scheduling and Control at the International Joint Conference on Artificial Intelligence (IJCAI-93), Chambery, France, 1993.

[Tate 93b] Tate, A., The Emergence of "Standard" Planning and Scheduling System Components, in *Current Trends in AI Planning*, (eds. Backström, C. & Sandewall, E.), IOS Press, 1993.

[Tate 94] Tate, A., Mixed Initiative Planning in O-Plan2, Proceedings of the ARPA/Rome Laboratory Planning Initiative Workshop, (ed. Burstein, M.), Tucson, Arizona, USA, Morgan Kaufmann, 1994.

[Tate 95] Tate, A. Characterising Plans as a Set of Constraints – the <I-N-OVA> Model - a Framework for Comparative Analysis, Special Issue on "Evaluation of Plans, Planners, and Planning Agents", ACM SIGART Bulletin Vol. 6 No. 1, January 1995.

[Tate 96] Tate, A. Representing Plans as a Set of Constraints – the <I-N-OVA> Model, Proceedings of the Third International Conference on Artificial Intelligence Planning Systems (AIPS-96), Edinburgh, UK, AAAI Press, May 1996.

[Tate & Drabble 95] Tate, A. and Drabble, B., Plan-World Viewers, Proceedings of the 14th Workshop of the UK Planning and Scheduling Special Interest Group, Colchester, UK, November 1995.

[Tate et. al. 94a] Tate, A., Drabble, B. and Dalton, J., "The Use of Condition Types to Restrict Search in an AI Planner" Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI-94), Seattle, USA, August 1994.

[Tate et. al. 94b] Tate, A., Drabble, B. and Kirby, R.B., O-Plan2: an Open Architecture for Command, Planning and Control, in *Intelligent Scheduling* (eds. Zweben, M. and Fox, M.S.), Morgan Kaufmann, San Francisco, 1994.

[Tate et. al. 94c] Tate, A., Drabble, B. and Dalton, J. Reasoning with Constraints within O-Plan2, Proceedings of the ARPA/Rome Laboratory Planning Initiative Workshop, (ed. Burstein, M.), Tucson, Arizona, USA, Morgan Kaufmann, 1994.

[Vere 81] Vere, S. Planning in Time: Windows and Durations for Activities and Goals, *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 5, 1981.

[Wilkins 88] Wilkins, D. *Practical Planning*, Morgan Kaufmann, Palo Alto, 1988.