DYNAMIC PLANNING MODEL FOR AGENT'S PREFERENCES SATISFACTION: FIRST RESULTS

PAVLOS MORAÏTIS*

Decision Support Systems Laboratory, Technical University of Crete
University Campus, 73100 Chania, Crete, Greece
E-mail: moraitis@dias.ergasya.tuc.gr

ALEXIS TSOUKIÀS

**LAMSADE, University of Paris-Dauphine*
75775 Paris Cedex 16, France
E-mail:{moraitis, tsoukias}@lamsade.dauphine.fr

Dynamic planning means reasoning about planning and executing actions in a dynamic, real world environment, by taking into account changes generated by unpredicted events occurred during the execution of actions. Make an agent able of reasoning in dynamic situations is an important issue in the agency theory. In our approach we consider that agent have preferences among consequences of his possible actions performed to reach a fixed goal. Preferences are modeled as criteria in the multi-criteria planning problem we propose. Our objective is to present an approach on dynamic planning where environmental changes and their consequences, but also changes on agent's preferences and on his methods to evaluate them, are taken into account as revision of three specific structures called possible plans, efficient plans and best plans and modeled as a multi-objective dynamic programming problem.

Keywords: dynamic planning, decision theory, reasoning, action, preferences satisfaction

1 Introduction

The aim of our work is to propose a formal model of dynamic planning for agents having a set of preferences, while they realize fixed goals in a real world, where changes can come both from a dynamic environment and from the agent himself. Several works are proposed in the so-called "reactive planning" field in order to address planning in a dynamic environment under different approaches [3,4,5,6,13]. Such works propose different techniques in order to react to environmental changes, which may occur during the execution process. In this paper we adopt a more general approach since we consider that, in addition, any change may occur in agent's behavior (for any reason, i.e. according to a possible user suggestion) during the execution process, pushing him to change his preferences and consequently his actions or his method to evaluate these preferences. Changes on agent's preferences and on his evaluation methods, are taken into account as revision of three specific structures called *possible plans*, *efficient plans* and *best plans*. To model these structures we use graphs inspired by the ones described in [11]. Our formalism allows us to present a planning problem

as a multi-objective dynamic programming problem. Using dynamic programming in planning problems dates back to Bellman [1], but its use in agency theory has been limited in search algorithms, as in [13] or in the frame of "universal planning" algorithms (see [12]). Under such a perspective the model we propose allows an agent based on the set of possible actions to achieve a fixed goal, to express his preferences about the benefit he desires to take out (for example, profit, time, pleasure, etc.) by achieving this goal and consequently to define the efficient actions for this end. Further on by introducing some additional information concerning his preferences it is possible to define the best plan as the preferred compromise. During the execution of a single action the agent may modify his evaluations (a revision is necessary) or the world may be modified after an unanticipated event (an update is necessary). Such changes (how these are perceived is not considered in this paper) may invalid the plan under execution in the sense that it could be impossible to follow it or not any more convenient. So, the aim of our model of dynamic planning is to take in to account such changes and to decide what the agent should do. In the following, section 2 outlines our formalism and describes our multi-criteria planning model. Section 3 presents how dynamic planning can be pursued under our model. We conclude by situating our research compared to related work.

2 The Multi-Criteria Planning Model

Consider that each agent α_l has to accomplish a set T of tasks in order to accomplish a fixed *goal*. Each task t_i can be decomposed in subtasks necessary in order to achieve t_i. We can consider that an agent has to go through a set of "states of the world" from a state where no subtask and therefore no task is accomplished (the "nil" state of the world) to a state where all tasks are achieved and therefore the local goal is achieved (the "final" state of world). We can represent such a situation as an oriented graph. So, the agent has to execute some actions in order to accomplish his tasks. Each time an action is executed the agent perceives some consequences (for instance a resource is consumed, a distance is computed, a profit is reached etc.). Therefore each time a subtask is achieved the agent is able to register the level of associated consequences on a set of attributes on which he might be able to express his *preferences*.

Let us try now to formalize our model of planning for each agent. The available information consists in: a) set *T* of tasks t_i necessary for a fixed goal achievement b) a set *S* of possible states of accomplishment s_i^j for each task t_i (for each task at least two states are considered s_i^0, task not accomplished and s_i^f, task accomplished; if there exists intermediate states there exists possible subtasks) c) a set *A* of possible actions a_l d) a set *H* of partial orders $\geq$ k on the set A ($x \geq_k y$: the action *x* is at least as good as the action *y* on the partial order $\geq_k$) e) a set *P* of the possible sequences

of actions. Finally we consider that it is possible to define a set H of binary relations $\supseteq_k$ on the set P ($x \supseteq_k y$: the sequence of actions x is at least as good as the sequence of actions y on the relation $\supseteq_k$). For the moment we make the hypothesis that each such binary relation is reflexive ($\forall x \in P$, $x \supseteq x$). Our model considers that is possible to establish the relation $\supseteq_k$ on the set P from the partial order $\geq_k$ on the set A.

Let us introduce now the concept of "state of the world". A state w is a collection of propositions, predicates and/or functions $\langle \Sigma, \lambda, \pi^k_j \rangle$ where: Σ: is a set of propositions that specify what is true in that state of the world; $\lambda \subseteq T \; x \; S$: is a binary relation associating a task t_i to an accomplishment state s_j and $\pi^k_j : P \rightarrow R$ are functions mapping the set P of possible sequences of actions to the reals, representing the binary relations $\supseteq_k$. Of course such functions exist iff the corresponding relations are at least weak orders (complete and transitive). If some of the partial orders $\geq_k$ on the set A are at least weak orders then there exists real valued functions g_k, one for each such weak order. We represent with $g_k(a_l)$ the consequences of adopting action a_l under the preference g_k.

An Example: in this section we present an example which describes our model and the chosen context. Let's consider en empty room R, which has to be equipped by an agent α with a bookcase, A. Agent α has to assemble this bookcase, to move it inside of the room (the order of two actions execution has no importance) and to put the books on the bookcase. We make the assumption the door of the room leading inside is normally open. The situation is the following: initial state of the world: takedowned(A), OUT(bookcase), OUT(books), opened(door); the final state of the world to be attained is: (assembled(A), puton(B, A) IN(A)) ; possible actions of α: move(x, y, z, w): agent x moves object y from place z to place w; move(x, yk, z, w): agent x moves together objects y and k from place z to place w; puton(x, y, y'); agent x puts object y on object y', to-assemble(x, y): the agent x assembles the object y; putdown(x, y): agent x puts object y down, open(door); relations between actions: before(to-assemble(x, bookcase), puton(x, books, bookcase)); agents' preferences: (max-profit, p), (min-time, t). We assume that actions to-assemble(x, y) leave a profit of 2 units while they generate a loss of 1 time unit, actions move(x, y, z, w), puton(x, y, y') leave a profit of 1 unit while they generate a loss of 1 time unit and action putdown(x, y) generates a loss of 1 profit unit and a loss of 1 time unit. The action open(door) generates a loss of 1 time unit.

2.1 Possible, Efficient and Best paths.

Now we are able to write down a model of agent's behavior by modeling the agent's planning and/or reasoning problem as a multi-objective dynamic programming problem. We establish three graphs.

Definition 1-Possible Paths Graph: a possible paths graph contains a start node corresponding to a nil state (none subgoal is accomplished), an end-node corresponding to a given goal to achieve and a set of intermediate nodes corresponding to intermediate states of the world. Arcs correspond to the set of possible actions an agent can perform to achieve his goal through several subgoals achievement. We denote the possible paths graph as $\Gamma_P = \langle W_P, A_P \rangle$.

Definition 2-Efficient Paths Graph: an efficient paths graph represents the set of efficient paths among the possible paths, computed according to the agent's preferences. It represents all "efficient" (not dominated) ways to achieve the agent's goals (Fig 1). Generally it is impossible to find a path which will be the best for all the agent's preferences, (this is an elementary notion in multi-criteria decision aid, see [15]). It is clear however, that exist paths which are definitely dominated by other ones, in the sense that they are worse under all points of view (all preferences). Let's introduce a dominance relation >>. Given any two possible paths p, p': $p >> p' \Leftrightarrow \forall\ k\ p \supseteq_k p'$ and $\exists k^*: p \supset_{k^*} p'$. The set of efficient paths D will therefore be the set of paths which are not dominated: $D = \{p: \neg\exists p' \in P: p' >> p\}$. We denote the efficient paths graph as $\Gamma_E = \langle W_E, A_E \rangle$. Clearly $\Gamma_E \subseteq \Gamma_P$.

Definition 3-Best Paths Graph: a best paths graph represents the best compromise solution among the efficient paths according to some further additional information (as for instance an importance relation among his preferences). We will make the hypothesis that the agent has such kind of information and therefore he is able to identify a plan p^* such that $\forall p \in D\ \Delta(p^*, p)$, Δ representing a weak order on the set *D*. Under the hypotheses done in this paper there exists a lot of procedures to identify the "best" compromise solution among the efficient ones [7,8]. We denote the best paths graph as $\Gamma_B = \langle W_B, A_B \rangle$. Clearly $\Gamma_B \subseteq \Gamma_E$.

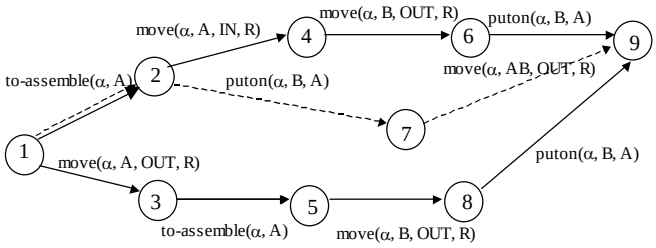

Figure 1: Efficient and best paths of α agent for min-time and max-profit preferences

Continuation of the example: Figures 1 represents efficient paths (in this example possible and efficient paths coincide) which correspond to the efficient plans which allow the agent to achieve his goal (assembled(A), puted-on(B, A) IN(A)). Practically it means to reach world state 9 taking into account two preferences (max-profit, min-time). If the agent prefers min-time he may choose the plan

presented by the dotted lines. If he prefers max-profit he may choose one of the two plans presented by the bold lines.

3 Dynamic Planning

3.1 Descriptive considerations

Suppose now that our agent α has done his choice and a path (plan) has been selected and is under execution. If nothing happens during the execution the agent will perform his tasks and the final state of the world will be reached with goal accomplished and preference(s) satisfied. Nevertheless, during the execution different events may occur such that the agent may modify his evaluations (a revision is necessary) or such that the world is modified (an update is necessary). It is possible that such revisions or updates (hereafter called changes) may invalid the plan under execution. So, what should the agent do? Let's try to classify the possible changes.

1. *Best paths revision* (c_1). A first change that may occur concerns the weak order Δ. For different reasons the agent may modify the weak order under which the specific best plan has been chosen among the efficient ones. Such a change will not affect any of the basis information available to agent. The possible graph Γ_P and the efficient graph Γ_E rest the same. What was considered as the best compromise is not any more such (for example the agent may have modified the priorities or importance of his preferences, i.e. choose min-time if before max-profit has been chosen, see §3.2, Fig. 2).

2. *Efficient paths revision*. A second change that can occur concerns the states of the world and particularly the functions π^k_j. Actually the way by which the agent evaluates the actions and therefore the plans, as far as his preferences are concerned, can change (for instance the agent may realize that some actions are "more expensive" from what has been considered at the beginning, (i.e. consider that the action move(α, AB, OUT, R) (Fig. 1) generates a loss of 2 time units while it has been before considered that it generates a loss of 1 time unit). Under this point of view the efficient graph Γ_E could be modified (although not strictly necessary) since a path considered efficient may not be any more, while a dominated path may become efficient. For example according to the assumption made just above, path 1-2-7-9 where ($\pi_9^p = 4$, $\pi_9^t = 4$) is not any more efficient compared to paths 1-2-4-6-9 and 1-3-5-8-9 where ($\pi_9^p = 5$, $\pi_9^t = 4$) considered for min-time and max-profit preferences. From the execution point of view the question is whether the intention plan is still the best compromise among the new efficient set of plans. The following possibilities may occur: 1) the present best path is not more efficient and therefore is not any more the best compromise (c_{21}) 2) the

present best path is still efficient and but is no more the best compromise (c_{22}) 3) the present best path is still efficient and the best compromise. Obviously only the first two cases may affect plan's execution.

3. *Possible paths revision 1*. A third change that may occur is the elimination of one or more possible actions from the set *A*. Such modification affects the possible paths graph Γ_E and therefore can affect the efficient paths graph Γ_E and the best paths graph Γ_B. The possible consequences of such a change are the following:

(c_{31}) Some states of the world are modified as far as the functions π^k_j are concerned (the sequences under which a state can be reached are now different; the values of some π^k_j can be modified). The considerations of point 2 apply here. For example, consider that agent α discover that he is unable to perform the action move(α, AB, OUT, R) (Fig. 1) because objects A and B are finally together to much heavy compared to his ability to move heavy objects. Under this possibility we include also the case where action(s) eliminated belong to the best plan. That means that some states of the world which have been foreseen to be reached under certain conditions remain reachable, but under new conditions.

(c_{32}) A state of the world becomes unreachable because all the actions leading to such a state are eliminated. If such a state belongs to the best plan then the agent has to reconsider the ongoing execution, otherwise the change will not affect his behavior. We call such a state as "infeasible state" and we denote it as $w^{\perp}$ (the state of node 9 (Fig. 1) if agent α is unable to perform the action move(α,AB, OUT,R)).

(c_{33}) A state of the world becomes a "cul-de-sac" in the sense that all actions (arcs) leaving this state (node) are eliminated. Again a problem will arise only if such a state belongs to the best plan. We call such a state an "infeasible state" and we denote it as $w^{\perp}$ (i.e. the state of node 7 (Fig. 1) if agent α is unable to perform the action move(α, AB, OUT, R)).

Possible paths revision 2. A fourth change that may occur is the availability or necessity of one or more actions, which before were impossible or unforeseen. Again such a modification affects Γ_P and therefore Γ_E and Γ_B. The possible consequences are the following:

(c_{41}) Some states of the world are modified as far as the functions π^k_j are concerned. A node which was reachable for a certain value of the function π^k_j is now reachable for new values (possibly better). Under such a perspective the new action will connect nodes which in the original possible paths graph were not adjacent. A problem will arise only if the modified states of the world belong to the efficient paths graph and can influence the best path graph.

(c_{42}) The new actions(s) may create a state of the world, which was not considered in the set *W* (for instance the new action may correspond to the necessity to accomplish a new task or subtask, which was not considered before). For example if an unpredicted event (the door is closed) occurs at the moment when the agent is in the node 2 during the execution of the path 1-2-4-6-9 (Fig. 1). The new state of the world not considered in the beginning is (assembled(A), OUT(A), OUT(B),

closed(door), see §3.2, Fig. 3). A problem will arise only if the new action(s) and the new state(s) of the world can belong to a path which confronted to the best path can be considered better.

3.2 Operational considerations

Different combinations of changes may occur simultaneously, leading to different necessities of re-planning reconsidering the plan under execution. In this paper we do not care how a change is perceived by the agent. Two basic problems are of our concern: 1) how to detect a modification and how to classify it according to the previous presentation? 2) how to react to the changes in order to adopt a new, possibly better, plan under the new information considering time constraints?
The algorithmic aspects of both the detection and the reaction (how the graph is modified) are not detailed in this paper (they are the subject of a future paper). However, generally the same dynamic programming approach applies on the modified graphs. When the execution of the plan is triggered a control program is also executed which may detect one of the following (at least): a) the weak order Δ is modified b) at least one of the relation $\supseteq_k$ is modified c) at least one arc (action) is eliminated d) at least one arc (action) is added. The agent can found himself in two situations: 1) he is able to interrupt the execution; this case is not frequent in real world dynamic situations if environment is the source of changes; however this can be possible if agent himself (or his eventually user) is the source of changes (i.e. evaluation changes, preferences changes), 2) he has to continue the execution trying to decide a reaction. Let us consider separately the two situations:
1. The agent is in the state w^0. In this case we consider as p^* the part of the best path not yet executed, in other terms the part of Γ_B going from w^0 to w^f. Moreover p^{**} is computed as Γ'_B using either Δ' or Γ'_E or Γ'_P (depending on the change occurred), considering as w^{nil} the w^0. In the case c_1, verifying if $\exists p^{**} \in \Gamma_E : \Delta'(p^{**}, p^*)$ is possible using a sensitivity analysis of the procedure exploiting Δ (and Δ'). Let us suppose, in our example, that agent α is in the node 2 having choose the max-profit and therefore to execute the path 1-2-4-6-9 in Figure 1. If in this moment he decides (for any reason) to change his preference (i.e. min-time) he will have to compute p^{**} which correspond to the path 2-7-9 (Fig 2) by using Δ' on Γ_B. Sensitivity analysis may also occur as far as the cases c_{21} and c_{22} are concerned, although in such cases the sensitivity analysis will apply on the construction of the dominance relation (and therefore the construction of the dominance relation and the construction of Γ_E) and of the weak order Δ.

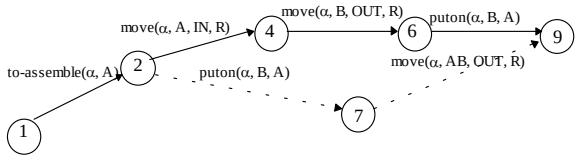

Figure 2. Dotted lines present the new best path for agent α after preference change (min-time)

In the case c_{31} the path $p^* \setminus a_{ij} \cup \gamma_{ij}$ (where γ_{ij} is a new sequence of actions) has to be compared to the set Γ'_E which has to be recomputed. In our example, let us suppose, like in §3.1 (c_{31}), that agent can discover that he is unable to perform the action move(α, AB, OUT, R) (Fig. 1) because objects A and B are finally together to much heavy compared to his ability to move heavy objects. In this case the sequences of actions γ_{ij} can be {putdown(α, B), move(α, A, OUT, R), move(α, B, OUT, R), puton(B, A)}. In the two cases c_{32} and c_{33} the path p^* becomes simply infeasible and therefore the agent has to make a new choice in Γ'_E. In all the previous three cases it is easy to observe that the graph Γ'_E can be obtained from Γ_E by simple modifications without recomputing all the efficient solutions.

In the case c_{41} the agent may compare all paths using the new arc a_{ij} to p^*. In case c_{42} if the new state is mandatory then the graph Γ_B has to be recomputed, otherwise it is sufficient to compare p^* to all paths going through the new state. In our case cited in (§3.1, c_{42}) the agent has to execute the action open(door) leading in a new node 2'. So it is obvious that it is not necessary to recompute Γ_B because p^* is still the best (if the preference is still the same) if we compare all paths started previously by node 2 and going now through the node 2' (Fig. 3).

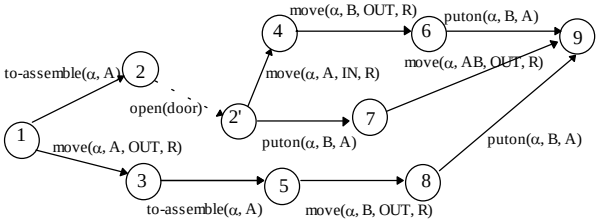

Figure 3. Dotted line present the new action in Γ'_P graph

2. The agent cannot interrupt the execution of the plan. He perceives the change while being in the state w^0 at time t_0. We have two possibilities:

2.1.The agent estimates that it is possible to compute a reaction in time t_r inferior to the time necessary to reach w^f or any infeasible state $w^\perp$. Considering that the agent is able to compute in which state will find himself after the time t_r (let's denote it w^r) it is possible to apply the reactions presented in the previous section as if the interruption occurred in state w^r. This situation can be possible because we consider that agent has information about his possible plans characteristics, like time

execution estimation, rate of success in the past, performed in similar situations. We consider that agent can use such information as criteria for plans choice (see [2]). In our example we can consider that agent has an estimation of the time he needs to assemble a bookcase.
2.2. The agent estimates that it is not possible to compute a reaction before the state w^f is reached. With the exception of the cases c_{32}, c_{33} and c_{42} (in the particular situation where the new state is mandatory) the agent will execute the intended plan although it may be no more the best. In the cases c_{32} and c_{33} the agent will necessarily interrupt the execution when will reach the infeasible state. He may therefore elaborate an alternative plan starting from any node, which is not infeasible and belongs to Γ_B. In the case c_{42} (and denoting the new mandatory state as w^m) the agent has to verify if it is feasible an action a_{fm} (from the previous final state w^f to the mandatory w^m). If yes he will just add such an action to the plan, whatever consequences may produce. If not he has to interrupt the execution to the least state from which it is possible to reach the mandatory state.

4 Relative Work and Conclusion

The principal difference of our work compared to other works in the field of so called "reactive planning" [3,4,5,6,13] where different techniques are proposed for react to environmental changes, is that we adopt a more general approach since we consider that, in addition, any change may occur in agent's behavior (for any reason, i.e. according to a possible user suggestion), pushing him to change his preferences and consequently his actions or his method to evaluate these preferences. Changes on agent's preferences and on his evaluation methods, are taken into account as revision of three specific structures called possible plans, efficient plans and best plans. These are generated using an original multi-criteria planning model taking into account agent's preferences. Several works (see [9,10]) have been proposed in literature where graph theory and dynamic programming is used for planning purposes. However, such approaches are based on the idea of a "search" on the space of possible states, thus operationally exploring a tree structure resulting from a branching procedure. Our approach is completely different both from a representational point of view (we have a real graph with a single source and sink) and from an algorithmic point of view due to the multi-objective nature of the problem we introduce. Veloso et al., [14] introduce the idea of rational-based monitoring of plan execution in a dynamic environment. Their approach is very similar to our classification of possible changes, but limited only to environmental ones. We claim that our model enables a more general characterization of the changes which may occur (i.e. agent preferences and evaluation methods) and how these may affect the computation of a new plan. Of course the problems open in this framework are more than the ones solved in this paper. However we believe that this paper highlights interesting issues by proposing dynamic planning as an useful

mean to reason about changes generated not only by the environment, but by the agent himself.

References

1. Bellman R., *Dynamic Programming*, Princeton University Press, Princeton, 1957.
2. Boussetta S., Henriet L., Tsoukiàs A., Decision-Theoretic Planning for Autonomous Agents: a multi-criteria analysis approach, submitted.
3. Firby R.J., Task networks for controlling continuous processes: issues in reactive planning, *Proc. AIPS-94*, 1994, pp. 49-54.
4. Gat E., Integrating planning and reacting in a heterogeneous asynchronous architecture for controlling real-world mobile robots, *Proc. AAAI-92*, 1992, pp. 802-815
5. Georgeff M.P., Ingrand F.F., Decision-Making in an embedded reasoning system, *Proc. IJCAI-89*, 1989, pp. 972-978.
6. Godefroid P., Kabanza F., An Efficient Reactive Planner for Synthesizing Reactive Plans, *Proc. AAAI-91*, 1991, pp. 640-645.
7. Hansen P., Bi-criterion path problems, G. Fandel, T. Gal, (eds.), *Multiple Criteria Decision Making: Theory and Applications*, LNEMS 177, Heidelberg: Springer-Verlag, 1980, pp. 109-127.
8. Henig M., Efficient Interactive Methods for a Class of Multi-attribute Shortest Path Problems, *Management Science*, 40, 1994, pp. 891 - 897.
9. Joslin D., Roach J., A theoretical analysis of conjunctive-goal problems, *Artif. Intell.*, 41, 1990, pp. 97-106.
10. Korf R.E., Planning as search: a quantitative approach, *Artif. Intell.*, 33 (1), 1987, pp. 65-88.
11. Moraïtis P., Tsoukiàs, A., A Multi-criteria Approach for Distributed Planning and Conflict Resolution for Multi-Agent Systems, *Proc.ICMAS-96*, 1996, pp. 212-219.
12. Schoppers M.J., Universal Plans for Reactive Robots in Unpredictable Environments, Proc. *IJCAI-87*, 1987, pp. 1039-1046.
13. Stentz A., The Focussed D* Algorithm for Real-Time Re-planning, *Proc. IJCAI-95*, 1995, pp. 1652-1659.
14. Veloso M.M., Pollack M.E. and Cox M.T., Rational-Based Monitoring for Planning in Dynamic Environment, *Proc. AIPS-98*, 1988.
15. Vincke P., *Multi-criteria Decision Aid*. New York: John Wiley, 1992.