Multi-Agent Planning and Scheduling, Execution Monitoring and Incremental Rescheduling: Application to Motorway Traffic

Pascal Mourou and Bernard Fade
IRIT, Institut de Recherche en Informatique de Toulouse
Université Paul Sabatier
118 route de Narbonne, 31062 Toulouse Cedex, France
Tel: +33 61 55 66 11 extension 72 63 or 63 30
Fax: +33 61 55 62 39
E-mail: mourou@irit.fr or fade@irit.fr

Abstract

This article describes a planning method applicable to agents with great perception and decision-making capabilities and the ability to communicate with other agents. Each agent has a task to fulfil allowing for the actions of other agents in its vicinity. Certain simultaneous actions may cause conflicts because they require the same resource. The agent plans each of its actions and simultaneously transmits these to its neighbours. In a similar way, it receives plans from the other agents and must take account of these plans. The planning method allows us to build a distributed scheduling system.

Here, these agents are robot vehicles on a highway communicating by radio. In this environment, conflicts between agents concern the allocation of space in time and are connected with the inertia of the vehicles. Each vehicle make a temporal, spatial and situated reasoning in order to drive without collision.

The flexibility and reactivity of the method presented here allows the agent to generate its plan based on assumptions concerning the other agents and then check these assumption progressively as plans are received from the other agents. A Multi-agent execution monitoring of these plans can be done, using data generated during planning and the multi-agent decision-making algorithm described here. A selective backtrack allows us to perform incremental rescheduling.

Keywords

Anytime Planning and Scheduling Algorithms, Execution Monitoring and Incremental Rescheduling, Managing limited computation time, Dependency Analysis and Plan Reuse, Autonomous Agents.

1 Multi-agent worlds

Monitoring a little structured multi-agent environment, such as a highway traffic, is an extension to the problem of monitoring robots in a factory. The agents are assumed to be "high-level" since they must have a great ability to perception and they must communicate with each other to cooperate, coordinate their actions and resolve any conflicts. The resolution of conflicts is the main point of interest. Logic schemata, attempting to model human thinking, have been developed to represent the wishes and beliefs [Bessiere, 84][Wilks and Ballim, 87] which are the mutual basic knowledge needed to resolve conflicts. Persuasion [Rosenschein, 82][Sycara, 89] is the aim of exchanging arguments. Most studies are simplified by assuming that agents cooperate (see [Cammarata et al., 83]). Rosenschein and Genesereth [Rosenschein and Genesereth, 85], on the contrary, attempt to allow for agents which are not necessary "benevolent".

2 The motorway

Unlike Wood [Wood, 83], we do not generate routes but consider the driving of the vehicle (acceleration, lane changes, etc.). We shall use a different approach to Fraichard and Demazeau [Fraichard and Demazeau, 89], who describe a centralized system to generate vehicle trajectories at cross-roads. We use a distributed system in which the number of central units increases as the number of agents increases. The multi-agent world was modelled on this basis (see [Mourou and Fade, 91a] and [Mourou, 90]).

Each vehicle has a co-pilot computer which may either be in an automatic mode, driving the vehicle, or in a supervision mode when it warns the driver or, if necessary, takes over control when an accident is imminent.

When all vehicles are in the "automatic driving" mode, it is simple: the vehicles are considered as autonomous robots which communicate with each other. The supervision mode requires a veritable "execution monitoring" which must be highly flexible and supervise drivers' acts by comparing them with the "ideal" plan generated in the automatic mode.

Co-pilots exchange data via a short-range communication network. The agents must cooperate to guarantee "efficient and safe traffic movement" and must respect the highway code, used as veritable "cooperative strategy"

[Cammarata et al., 83]. A number of objectives are also fixed for each agent, such as "to travel at the mean speed required by the driver". Unlike certain systems analyzed by Davis and Smith [Davis and Smith, 83], no tasks need be shared in the procedure since each agent knows what he must do. The negotiation therefore covers solely how its tasks can be accomplished.

The co-pilot in each vehicle is concerned solely by the N relations which affect the vehicle. The task of the co-pilot will therefore involve selecting the behaviour, which is satisfactory to the N influences to which it is exposed at each time. In considering highway traffic, the "common resource" is the space available on the road. The main task of each agent is to check that the space it needs will be free and, if not, to take appropriate action to reach a free space (acceleration, lane changes, etc.). Conventional problem resolution techniques are not capable of simultaneously managing the N conflicts possible at each instant in the future. Moreover, a "distributed scheduling" technique will be unsuitable since, although automatic control can be considered as a resource allocation problem, the inertia of the various vehicles will make it extremely difficult to break the road down into a series of "areas", each considered as a resource.

The method we describe is more "expert"-oriented, allowing the "rules" in the highway code to be expressed and used as they exist and high-level data exchanges to be used. For example "I'm going to move out and accelerate up to 110 km/h" is a kind of action generated by the planner and broadcasted through the network.

3 Time, influence of other plans and delay

The behaviour of each agent is represented by a linear, non-hierarchical plan. We make the assumption that the agents are synchronised by a common clock broadcast by radio for example. B's "time influence" on A covers all the B's actions and situations around T_j used to plan A's action at T_i. When some of them are missing, A must make assumptions on the actions planned by B and consequently progressively check these assumptions as the actual actions are received. If A's assumption is found to be correct, we shall have saved time. Otherwise, A must replan this action after B has transmitted its decision and no time will have been lost.

4 The "Is there an agent... ?" method

Knowing, or assuming, the actions of other agents, agent A must generate an action (the behaviour for a given step). It can repeatedly

pose this kind of question : "Is there an agent preventing me doing this ?". Each question determine whether there is a conflict which prevents one action (method M1).

An example of situation (Example 1) :
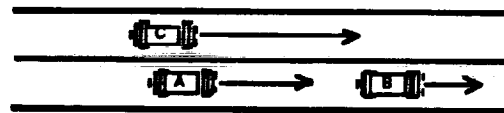
can be given by the possible conflicts A will detect :
- (C_b) : B is in front of A (Us), which is travelling faster and wants to accelerate even further.
- (C_c) : C is on the left of A and prevents A overtaking.

Questions which A could ask before deciding to "slow down" are:
- Can I accelerate ? (agent B imposes the reply "no")
- Can I move out to overtake B ? (agent C imposes the reply "no").

At a first view, it could be difficult to write directly an algorithm capable of taking a decision adapted to A's wishes when exposed to complex influences (see Figure 1).
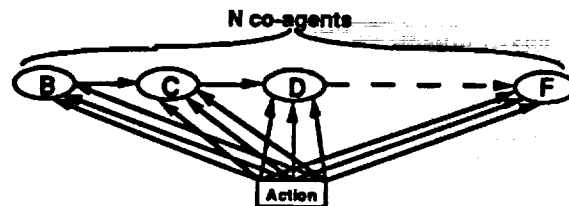

Figure 1. Method M1

5 The dual method: "Is there an action ... ?"

5.1 Definition

We could use tests of the type "Is there an action prevented by this agent ?". The agents would then be reviewed, one after the other, to collect all conflicts to which A is exposed into a "Results" structure (see Figure 2).
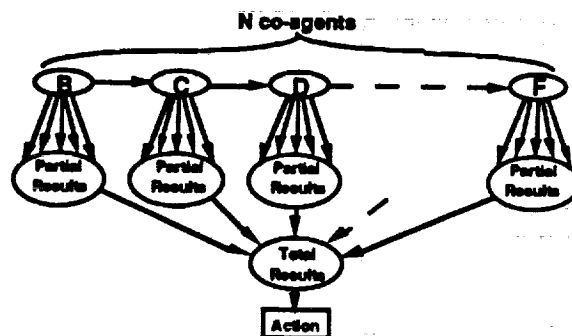

Figure 2. Method M2

- "Is there an action prevented by B ?" :
  "B prevents me not(slowing down or moving out)" = C_b
- "Is there an action prevented by C ?" :
  "C prevents me moving out" = C_c

A second phase allows the action to be determined :

- C_b and C_c --> "prevented from not slowing down" = "decelerate"

This second phase, used to find the best possible response in view of all the behaviours that are prevented and the requirements of A, occurs after determining all behaviours that are not possible (method M2). It allows K conflicts to be grouped and assessed simultaneously (K < P: maximum number of conflicts). It could be considered as simplified multi-agent planning which chooses an action in function of the prevented ones. The knowledge required for this reasoning is referred to as "N-agent knowledge".

On the other hand, each question allows assessment of a relationship between two agents. The term "bi-agent" refers to the process and knowledge used for each comparison. The result of a two-agent comparison is known as a "Partial Result".

A "mono-agent" phase may influence the Total Results in function of A's wishes before the series of bi-agent comparison.

5.2 Application to motorway traffic
The Total Results for the Example 1 would be:

Prevent-moving-out	t		
Move-out	t	Move-out-list	(B)
Slow-down	()	Slow-down-list	()

The decision-making rules for the bi-agent and then N-agent phases would be, for example:

• if A is in the right-hand lane and X is in front of A in the right-hand lane and at lower speed and if safety distance has been reached then Move-out(X) := t   Move-out-list(X) := (B)
• if Move-out and Prevent-moving-out then decelerate, choosing vehicles in Move-out-list
• if Move-out then move out
• if Prevent-moving-out then do nothing
• if true then accelerate

5.3 Selective backtrack
The use of Results and the separation of conflict recognition from their overall processing makes a selective backtrack

possible. Consequently, new information from agent B concerning an instant T_j, already planned, can be allowed for solely by comparison with B (see Figure 3).
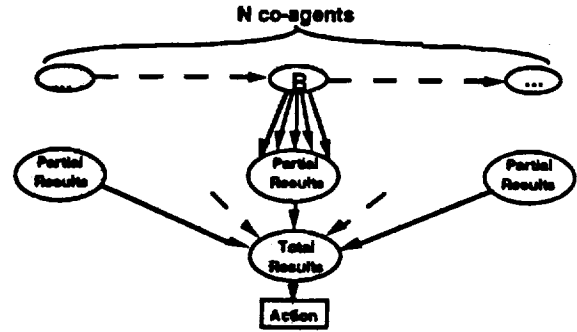

Figure 3. Selective backtrack for agent B

If the new Partial Results for P at instant T_j, designated "new-Partial-B-T_j" equal Partial-B-T_j (i.e. the response to the new influence is the same as that to the previous influence - see Example 2.1) or if "new-Partial-B-T_j" is already part of "Total-T_j" (i.e. the response to the new influence had already been requested by at least one of the agents - see Example 2.2), a total backtrack is pointless since the N-agent phase would produce the same conclusion. This selective backtrack is then sufficient for instant T_j. The same must then be repeated for each instant T_k between T_j and the current planned instant T_i.

Since case 2 covers case 1, there seems to be no point in memorizing the Partial Results but only the Total Results (method named M3*).

If, for any instant T_k between T_j and T_i, the results are not already included it can only be because a new response has been requested. The N-agent phase must, therefore, be triggered using a new Total Results which can be calculated in two ways:

new-Total-T_k := $(\bigcup$ Partial-X-T_k ; X $\neq$ B)
$\oplus$ new-Partial-B-T_k

new-Total-T_k := Total-T_k $\ominus$ Partial-B-T_k $\oplus$ new-Partial-B-T_k

In either case, it would be useful to know certain Partial Results.

The conflict recognition phase is, therefore, avoided for any agent other than B and the backtrack is still not total. If the resultant action is the same as that which would have been generated without the new information (see Example 2.3), we only need to continue selective backtracking on actions for instants after T_k.
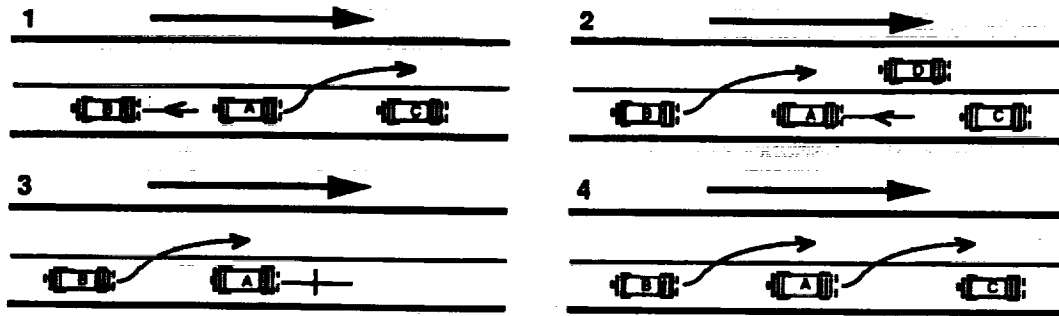
1. A's plan : overtake C ; B's plan : decelerate : the constraints B imposes on A are unchanged.
2. A's plan : slow down ; B's plan : overtake A : the new constraint B imposes on A, i.e. forbidden to move-out, was already imposed by D.
3. A's plan : do nothing ; B's plan : overtake A : the new constraint B imposes on A does not affect the action planned by A.
4. A's plan : overtake C ; B's plan : overtake A : the new constraint B imposes on A generates a new action, i.e. slow down. A must replan the following instants.

Example 2. Various selective backtrack levels

However, if the action generated is new (see Example 2.4), a total backtrack from T_{k+1} onwards is necessary since the new action could change the result of all the previous comparisons.

5.4 Execution monitoring
The execution monitoring of the plans generated can be done using the memorized Total Results and the selective backtracking possibilities to check that no agents, cause any infractions.

The real behaviour of human drivers could be monitored as follows:
• If man behaves approximately as the system expects then there will be no problem
• Otherwise:
  • If the man in question is driving our car, check whether the behaviour of the man is included in the prohibited behaviours memorized in the Total Results :
    • If there is infraction of one of these prohibitions, the driver could be warned (for a low-risk situation or a detected intention) or the system could take control to avoid an accident (for a dangerous situation).
    • Otherwise, complete replanning is required to adapt to this new behaviour (once the driver's intentions have been recognized...).
  • If the man is driving another vehicle, which possibly does not have the system, it is necessary to run a selective backtrack for each instant T_k between T_j and the current planned instant T_i to adapt our plan.

6 Theoretical efficiency of the methods
The theoretical costs of each of various methods (among 12 alternatives) were estimated making certain average assumptions about the multi-agent application and the way in which the databases or algorithms are designed (see [Mourou and Fade, 91b] and [Mourou, 92]). These costs are expressed as a mean number of influence tests in function of the number of other agents N, the maximum number of conflicts P and the mean number of influence tests Q used in a M1 conflict test. One result is :

M1	$Q \times N/2 \times P$
M3*	$Q \times N$

M3* requires all possible comparisons to be done while M1 only requires comparisons on request. However, it is more efficient since the influence tests are grouped.

7 Main experimental results
We simulated a highway with two lanes and carrying three vehicles fitted with a co-pilot and 10 other preprogrammed vehicles. The three equipped vehicles are associated with three different processes linked through pipes. 24 rules (10 bi-agent and 14 N-agent rules) are required with M3* to obtain an ideal response in "automatic mode" which respects safety distances and allows for the inertia of vehicles. The knowledge bases made it possible to write that for M1.

In this application, Q = 3 (relative position, relative speed, lane) and P = 3 (number of

booleans in the Results structure). N does not affect the relative performance.

M1 and M3* gave results which matched those from the above formulas : M3* is 30% faster than M1.

In the best case, but which is also the most frequent, when the Partial Results calculated are included in the Total Results, M3* performed its selective backtracks in only 20% of the time required by M1 to completely replan (with N = 10).

Conclusion

The multi-agent planning/scheduling methods described in this article make its possible achieve a more flexible, fast and reactive system. The co-pilot can anticipate the near future by using the available time and without be obliged to wait for its neighbours because it can easily check and integrate a new information.

In execution monitoring, a dangerous situation can be quickly detected. A backtrack of an agent and the selective backtrack of other agents allow to perform incremental rescheduling of the whole system.

References

[Bessiere, 84] Pierre Bessière, "Un Formalisme Simple Pour Représenter La Connaissance Dans un Contexte de Planification Multi-Agents", RFIA 84, Actes du 4ième Congrès de Reconnaissance des Formes et Intelligence Artificielle, 25-27 Janvier 1984, Paris, tome 2, pp. 345-354.

[Cammarata et al., 83] Stéphanie Cammarata, David McArthur and Randall Steeb, "Strategies of Cooperation in Distributed Problem Solving", IJCAI 83, pp. 768-770.

[Davis and Smith, 83] Randall Davis and Reid G. Smith, "Negotiation as a Metaphor for Distributed Problem Solving", Artificial Intelligence 20, pp. 63-109, 1983.

[Fraichard and Demazeau, 89] Th. Fraichard and Y. Demazeau, "Motion Planning in a Multi-Agent World", Workshop on Modeling Autonomous Agents in a Multi Agent World (MAAMAW 89), Cambridge, England, August 1989.

[Mourou, 90] Pascal Mourou, "Représentation et Simulation d'un Agent dans un Monde Multi-Agent: un Véhicule en Circulation Autoroutière", Rapport Interne N°IRIT/90-10, Février 1990, 41 p.

[Mourou and Fade, 91a] Pascal Mourou and Bernard Fade, "Vehicles controlling: Representation of knowledge and Algorithms of a Multi-Agent Decision", Sixth International Conference on Applications of Artificial Intelligence in

Enginneering (AIENG 91), 2-4 July 1991, University of Oxford, UK, pp. 683-697.

[Mourou and Fade, 91b] Pascal Mourou and Bernard Fade, "Co-Pilot for the Driver Monitoring", PROMETHEUS PRO-ART Workshop on Intelligent Co-Pilot, December 12-13, 1991, Grenoble, France, pp 183-193.

[Mourou, 92] Pascal Mourou, "Evaluation comparative de techniques de décision multi-agent", Rapport Interne IRIT, à paraître.

[Rosenschein, 82] Jeffrey S. Rosenschein, "Synchronization of Multi-Agent Plans", AAAI 82, pp. 115-119.

[Rosenschein and Genesereth, 85] Jeffrey S. Rosenschein and Michael R. Genesereth, "Deals Among Rational Agents", IJCAI 85, vol 1, pp. 91-99.

[Sycara, 89] Katia P. Sycara, "Argumentation: Planning Other Agents' Plans", IJCAI 89, vol 1, pp. 517-523.

[Wilks and Ballim, 87] Yorick Wilks and Afzal Ballim, "Multiple Agents and the Heuristic Ascription of Belief", IJCAI 87, vol 1, pp. 118-124.

[Wood, 83] Sharon Wood, "Dynamic World Simulation for Planning With Multiple Agents", IJCAI 83, vol 1, pp. 69-71.