# Adversarial Planning for Large Multi-agent Simulations

Viliam Lisý
Agent Technology Center, Dept. of Cybernetics, FEE, Czech Technical University
Technická 2, 16627 Prague 6, Czech Republic
viliam.lisy@agents.felk.cvut.cz

## ABSTRACT

We investigate planning for self-interested agents in large multi-agent simulations. We present two heuristic algorithms that exploit different domain-specific properties in order to find high quality plans with reduced amount of computations. We further suggest finding a formal framework for describing these properties and complementing our previous mostly experimental results by formal guarantees.

## Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search—*Heuristic methods*

## Keywords

Multi-player Game, Adversarial Search, Heuristic

## 1. INTRODUCTION

Multi-agent simulations are currently used for training of personnel for high risk situations, analysis and prediction of properties of complex systems, or developing and testing multi-agent technology. The algorithms and techniques for controlling intelligent entities in these simulations are also applicable in the multi-billion computer game industry and may be used for controlling autonomous military forces and support the decision-making of commanders in the future.

Most of these applications assume the presence of a large amount of autonomous agents that act in a shared environment towards their own, often mutually exclusive, goals. In our research, we focus primarily on the domains with a strong adversarial component, where successful fulfillment of goals importantly depends on positive or negative interactions with plans of other agents (e.g., wargaming).

The methods that are currently used for choosing agents' actions in complex simulations and games are, for the sake of efficiency, strongly based on human-designed scripted behaviors and rule based approaches for their switching [1]. These methods allow fast execution and high amount of control over the run of the simulation, but they make the agents time-consuming to develop and often strongly irrational in the situations, which were not foreseen by the designers of the scripts.

**Cite as:** Adversarial Planning for Large Multi-agent Simulations, Viliam Lisý, *Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, van der Hoek, Kaminka, Lespérance, Luck and Sen (eds.), May, 10–14, 2010, Toronto, Canada, pp. . 1665-1666


An alternative solution is a declarative description of the domain and goals of the agents and an algorithm that is able to find a plan suitable for each specific situation – i.e., multi-agent planning (MAP). However, current MAP techniques generally assume that all the agents in the system fully cooperate in reaching their goals. These agents avoid conflicts in their plans, remove redundancies and support each other [3]. In adversarial situations, some of the agents in the system seek conflicts and give their opponents many reasons for creating redundant (more robust) plans. This makes the MAP techniques inapplicable without substantial modifications.

Acting in adversarial situations has been studied in AI research on games from the very beginning of this field. The effort has brought many interesting techniques over the time. However, direct use of these techniques in large simulations is also not practical due to high computational complexity, especially if higher amount of agents is involved.

We focus on combining techniques from game playing and (multi-agent) planning, in order to create methods usable in larger adversarial domains. The general methods developed in both these fields are computationally very demanding; hence we focus on heuristic methods that use various properties of the domain structure for creating simplifying abstractions and decompositions of the problem. The next sections briefly introduce two of our methods together with the domain property that helps them work successfully.

## 2. BACKGROUND KNOWLEDGE

A heuristic often used in planning is restricting the space of all possible plans of an agent to a smaller set of the more reasonable plans using background knowledge about the domain (e.g., HTN or TLPlan).

In our paper [5], we introduce an adversarial planning algorithm based on the classical multiplayer game-tree search. The algorithm utilizes procedural knowledge capturing how individual players tend to achieve their sub-goals. The information is used to limit the search only to the part of the game tree that is consistent with pursuing the players' sub-goals. We impose no specific requirements on the format of the procedural knowledge; any programming language or agent specification paradigm can be employed. We evaluate the algorithm both theoretically and empirically, confirming that the proposed approach can lead to a substantial search reduction with only a minor negative impact on the quality of the produced solutions in our domain.

The domain property that allows this heuristic to work is that minor modification of the plan of an agent usually does not cause substantial change in the outcome of the game.

## 3. DECOMPOSITION

The second heuristic often used in many fields is decomposing a big problem to a set of smaller problems, solving them separately and combining the partial solutions.

In our paper [6], we have developed an adversarial search method that decomposes a game with higher number of agents into a large set of smaller sub-games featuring only small overlapping subsets of the agents. The sub-games are solved separately using a generic adversarial search algorithm. In order to produce high quality plans for an agent, its options should be evaluated in combination with each agent that can importantly interact with its plans within a fixed planning look-ahead. Moreover, consideration of interactions among other agents is also important to determine whether they would prefer to interact with the agent at all. An easy solution covering all these interactions to certain extent and still assures a favorable computational complexity is consideration of all agents' subsets of a small fixed size.

The separate adversarial search for each subset produces a plan for each agent in the subset and its additional characteristics in the sub-game, such as the parts of the domain they operate on. Based on this information, we suggest two strategies for creating the global solution. Both of them just select one of the plans, which were created for an agent in the searches for the sub-games. The first method is faster and decides on the final solution based on simple rules using the acquired plan characteristics. The other selects much better solutions, but requires much more computational effort. It is based on additional search in the space of all the agents strongly optimized using the information from the sub-games.

In the experimental evaluation, we have implemented this method on top of the adversarial search algorithm with background knowledge introduced in the previous section. We evaluate its performance on a game modeled after a humanitarian relief operation in conflict environment. The results show that the method often finds the same solutions as the complete search in a significantly shorter time.

The important domain property that allows this method to work is that most of the important events in the game can be assured/prevented by a limited number of agents.

## 4. FORMAL GROUNDING

One of the main problems with heuristic approaches is that it is often hard to exactly define the class of domains, where the heuristic can be useful and to provide any guarantees on the quality of the solution for a specific domain. One way how to validate and compare algorithms in this situation is creating a set of established benchmarks. A typical example could be the International Planning Competition. However, this way the assumptions about the domains that allow the heuristic methods to be successful are not stated explicitly. It may lead to over-fitting planning heuristics to this set of benchmarks, without a clear connection to their performance in the real world. This strongly motivates formal investigation of the domain properties that ensure the success of the heuristic methods [4].

An equivalent of IPC for adversarial domains is the General Game Playing (GGP) competition. The community forming around this competition is highly relevant to our research and it is a valuable source of inspiration. However, the requirement of full generality is too restrictive for our target domains that are simply too large for that.

In order to make our methods as reusable as possible, we suggests finding a suitable formalism that would allow describing various useful properties of the domains and the heuristic adversarial planning algorithms. These properties include various kinds of sparseness in interaction and reasonable plan spaces, relations between individual agent goals, or regularities in specific domains. We are interested in properties that could be exploited in creating efficient heuristics for game playing. In the ideal case, we would like to be able to create statements, such as: if your domain satisfies a property $\mathcal{P}$ you can use algorithm $\mathcal{A}$ with certain formally proven (at least probabilistic) guarantees on the quality of the produced solutions.

The formalisms that seem promising for this purpose are various extensions of alternating-time temporal logic (ATL), which has been shown to be suitable for describing properties of games in GGP [7] and extensible to describe less crisp properties that are necessary for our purpose [2].

## 5. CONCLUSION

We identify a challenging problem of planning actions for intelligent agents in large-scale multi-agent simulations with strong adversarial component. The complexity of this problem forces us to use highly heuristic methods. We have developed two such methods that substantially reduce the complexity of the adversarial planning problem. These methods work well in our domain, for which they were developed and we would like to explore their applicability to other domains. We will formally define the main properties of the domain that make the developed methods effective and will try to prove (probabilistic) guarantees on the quality of produced solutions for other domains that fulfill them.

## 6. ACKNOWLEDGMENTS

Effort sponsored by the AFOSR USAF grant number FA8655-07-1-3083 and by the Ministry of Education of the Czech Republic grants ME09053 and MSM6840770038.

## 7. REFERENCES

[1] D. Bourg and G. Seemann. *AI for game developers.* O'Reilly Media, Inc., 2004.

[2] N. Bulling and W. Jamroga. What agents can probably enforce. *Fundamenta Informaticae*, 93(1):81–96, 2009.

[3] M. de Weerdt, A. ter Mors, and C. Witteveen. Multi-agent planning: An introduction to planning and coordination. In *Handouts of the European Agent Summer School*, pages 1–32, 2005.

[4] J. Hoffmann, S. Edelkamp, S. Thíebaux, R. Englert, F. Liporace, and S. Trüg. Engineering benchmarks for planning: the domains used in the deterministic part of ipc-4. 26:453–541, 2006.

[5] V. Lisý, B. Bošanský, M. Jakob, and M. Pěchouček. Adversarial search with procedural knowledge heuristic. In *Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009)*, 2009.

[6] V. Lisý, B. Bošanský, R. Vaculín, and M. Pěchouček. Agent subset adversarial search for complex non-cooperative domains. Under review.

[7] J. Ruan, W. Van Der Hoek, and M. Wooldridge. Verification of Games in the Game Description Language. *Journal of Logic and Computation*, 2009.