Multiagent Planning by Distributed Heuristic Search (Dissertation Abstract)

Michal Štolba
(michal.stolba@agents.fel.cvut.cz)

Agent Technology Center,
Department of Computer Science, Faculty of Electrical Engineering,
Czech Technical University in Prague, Czech Republic

Multiagent planning is the problem of finding sequences of actions of multiple agents acting in a world such that the world is transformed in a desired way (some goal is achieved). Typically, each agent can observe and influence only a portion of the entire world. Similarly to classical planning, one of the most successful approaches to solve multiagent planning problems is heuristic search. An approach most prevalent among the multiagent planning techniques is to use a distributed search with a heuristic restricted to a single agent's view of the problem. As such restriction may significantly reduce the heuristic quality, we aim to find ways how to compute the heuristics distributively and how to incorporate such distributed heuristics in a search scheme.

The research in multiagent planning covers a wide variety of models, in our work, we focus on the MA-STRIPS (Brafman and Domshlak 2008) model introduced by Brafman & Domshlak in 2008. MA-STRIPS is a minimalistic extension of STRIPS and therefore retains most of its assumptions such as deterministic actions, propositional states and closed world. The model defines agents by the sets of actions they can perform and based on the actions the set of propositions the agent has access to is defined. All propositions shared among at least two agents are considered public, all other propositions private. This notion slightly redefines the assumption of full observability - each agent can fully observe all public and its private propositions. An action is considered public if it operates on any of the public propositions.

An obvious benefit of the MA-STRIPS model is its simplicity and similarity to classical planning, so that many techniques can be (more or less) straightforwardly adapted. Also the computational complexity of MA-STRIPS planning is reasonable in the context of planning as it depends exponentially only on the density of agent interactions, but on on the number of agents itself. This means, that domains with sparse interactions can be solved with similar results as in classical planning. The simplistic approach of MA-STRIPS has also its shortcomings. It is mainly the inability to express some features common in multiagent systems, such as self interested agents, uncertainty of action execution, reasoning about knowledge of other agents, etc. Nevertheless it is worth investing effort into research of multiagent planning using the MA-STRIPS model, as it can be further extended as was classical STRIPS to include more such features and the developed techniques, planners and theory can be reused and extended as well.

It is also interesting to notice, that for each MA-STRIPS problem there is an equal global STRIPS problem (obtained by joining all agent actions into one action set), solvable by a centralized algorithm. This can be used to theoretically and experimentally verify the distributed algorithms.

The first multiagent heuristic search planner based on the MA-STRIPS model was MAD-A* (Nissim and Brafman 2012), first introducing a multiagent distributed search scheme. The main principle of the search is that a state expanded using a public action is sent to all other agents which add the state to their open lists. The particular search was based on A* using admissible heuristics and thus being an optimal planner. The heuristics were computed on a problem projected to the set of propositions accessible to the particular agent (projected problem). Thus the projected problem consists of all public propositions, all agent's propositions, all agent's actions and public projections of all other agents' public actions (that is the precondition and effects of the actions are restricted only to public propositions). The search principle was extended from A* to greedy best-first search (GBFS) in (Štolba and Komenda 2014; Nissim and Brafman 2014).

Another state-of-the-art multiagent heuristic search planner is FMAP (Torreño, Onaindia, and Sapena 2014), which is a forward-chaining plan-space search planner which uses a distributed heuristic based on domain transition graphs. FMAP is not based on MA-STRIPS but a similar formalism. ADP (Crosby, Rovatsos, and Petrick 2013) is a centralized heuristic search planner, based on an automated agent decomposition. GPPP (Maliah, Shani, and Stern 2014) uses landmark detection to separate the global and local search phases and uses landmarks also as heuristic. GPPP is based on MA-STRIPS.

In the first direction of our work, we have focused on distributed heuristic computation approaches, namely in the context of the well known FF heuristic (Hoffmann and Nebel 2001). The first approach we explored was to distribute the construction of relaxed planning graph (RPG) and also the following relaxed plan (RP) extraction in such a way that the returned heuristic values would be provably equal to the values obtained from the centralized FF on the respective global problem. As presented in (Štolba and Komenda 2013), although with strong guarantees, this approach was not practically viable.

To improve the practical efficiency, we have relaxed the requirement on the equality of the heuristic values with values of the centralized heuristic on global problem. This *lazy* approach was based on an observation, that a RP constructed purely on the projected problem may contain projected actions of other agents which may have unsupported private preconditions. Only when such actions exist in the RP, the cost of support for private preconditions can be obtained from the respective agent owning the action. The requested agent can simply construct a projected RP from the current state to the preconditions of the action in question. If the resulting RP contains any projected actions, the procedure recursively continues. This approach was first introduced in (Štolba and Komenda 2013) as a practically feasible variant and its modification was present also in (Štolba and Komenda 2014).

In (Štolba and Komenda 2014) we introduced a general approach to effectively compute relaxation heuristics. This approach is based on the application of the distributed recursion principle of the lazy FF to an effective relaxed reachability analysis algorithm used in the Fast Downward planning system (Helmert 2006), sometimes denoted as exploration queue (EQ). The relaxed reachability is performed using a priority queue containing reachable propositions with their computed heuristic values (h_{add} or h_{max} (Bonet and Geffner 1999)). When a proposition is extracted from the queue, all actions for which it is a precondition has the number of unsatisfied preconditions decreased. Once an action has no unsatisfied preconditions, it is applied, that is its effects are added to the EQ with corresponding heuristic value. In the distributed version, the heuristic value of the projected actions is not known, therefore in the case a projected action of other agent should be applied, a request to the owner is sent instead. When the owner agent receives the request, it computes the heuristic estimate of the action preconditions (similarly to the *lazy* FF approach) and returns the value to the initiator. The EQ algorithm then continues as usual. Obviously the distributed recursion principle appears again.

A distributed FF heuristic in (Štolba and Komenda 2014) was computed using the previously described distributed reachability analysis, but the RP was then extracted only locally. Still, this approach outperformed the *lazy* FF updated to also use the EQ algorithm. Analysis of the *lazy* FF algorithm shows it weakness, which is over-counting of actions due to the distributed recursion. In our recent work (Štolba and Komenda 2015), we take inspiration to solve this problem from the set-additive variant of the FF heuristic (Keyder and Geffner 2008). To do so, the agents do not communicate only heuristic values, but the whole sections of the relaxed plans, where, for the sake of privacy, are private actions hidden by hash codes. The resulting heuristic was termed *set-additive lazy* FF heuristic. Nevertheless it is clear that the amount of privacy preserved is somewhat reduced. The privacy and the ways how it can be preserved or jeopardized is in multiagent planning still an open question.

In (Štolba, Fišer, and Komenda 2015) a state-of-the-art admissible LM-Cut heuristic (Helmert and Domshlak 2009) was treated analogously. Since in this case we aimed at admissible heuristic, we, again, give theoretical proofs of equality of the distributed algorithm on the MA-STRIPS problem and the original centralized algorithm on the respective global STRIPS problem. In the distributed computation of the LM-Cut heuristic, the initiator agent works on the projected problem (including private and public propositions and action projections), while the other agents work on a local projection of the problem, that is only the private propositions of the agent and the agent's actions restricted only to the private propositions. In this way, the necessary h_{max} values are computed by iteratively communicating updates between the initiator and other agents. Similarly, the search for the cuts is managed by the initiator agent which communicate with the other agents when necessary.

When experimentally evaluating each of the distributed heuristics we have observed a common pattern. To improve the search performance, a distributed heuristic has to provide enough additional information to outweigh its higher computational (and communicational) overhead. Typically in domains, where most of the information is public (and the private parts does not hide important information), the projected heuristics dominate, on the other hand, if a lot of the information (or some crucial information) is private, the search can leverage the additional information of the distributed heuristics.

In the second direction of our work, we are aiming to somehow combine the projected and distributed heuristics in order to amplify their benefits, that is the speed of the projected heuristic and estimate quality of the distributed one. A very first approach in this direction was presented in (Štolba and Komenda 2014), based on limiting the recursion depth of the distributed EQ algorithm. This means, that whenever the evaluation of an action would jump over more agents than some δ, no more requests will be sent and projected actions will be treated as ordinary actions. This approach allowed to transition between the projected and distributed heuristics to some extent, but was not able to bring a dominating result.

A property that allows such desirable combination seems to be the asynchronicity of the presented distributed variants of the FF heuristic. By asynchronicity we mean, that during the heuristic evaluation, other agents are evaluating parts of the heuristic and the initiator agent can meanwhile perform some other computation. In our recent work (Štolba and Komenda 2015), the other computation is a local search using the projected heuristic, combined together using a variant of multi-heuristic search. The MADLA (Multi-agent Distributed and Local Asynchronous) Planner based on this technique dominates in terms of coverage a distributed GBFS with distributed heuristic, nearly dominates (but still outperforms) a distributed GBFS with projected heuristic and outperforms all comparable state-of-the-art multiagent planners (most notably FMAP and GPPP).

It is not clear yet, whether the same technique would be applicable in the context of optimal search using the distributed LM-Cut heuristic. It may be more beneficial to experiment with incremental LM-Cut computation to be able to compute the heavy distributed heuristic only in some states and carry on the additional information (landmarks) to the states evaluated using only the projected heuristic. Both directions will be investigated in the (near) future work.

A more speculative future direction might be the research of the possibility of devising a heuristic which would have independent (that is additive) part for each agent and if such property could be used to somehow decouple the agents, so that the initiator would not have to wait for other agents to answer, but would continue the search and update the heuristic values upon the delivery of the answers. More fundamental question is whether some of the described techniques could be used to improve the classical planning.

References

Bonet, B., and Geffner, H. 1999. Planning as heuristic search: New results. In *ECP*, 360–372.

Brafman, R. I., and Domshlak, C. 2008. From one to many: Planning for loosely coupled multi-agent systems. In *Proceedings of the 18th International Conference on Automated Planning and Scheduling (ICAPS'08)*, 28–35.

Crosby, M.; Rovatsos, M.; and Petrick, R. 2013. Automated agent decomposition for classical planning. In *Proceedings of the 23rd International Conference on Automated Planning and Scheduling (ICAPS'13)*, 46–54.

Helmert, M., and Domshlak, C. 2009. Landmarks, critical paths and abstractions: What's the difference anyway? In *Proceedings of the 19th International Conference on Automated Planning and Scheduling (ICAPS'09)*, 162–169.

Helmert, M. 2006. The Fast Downward planning system. 26:191–246.

Hoffmann, J., and Nebel, B. 2001. The FF planning system: Fast plan generation through heuristic search. 14:253–302.

Keyder, E., and Geffner, H. 2008. Heuristics for planning with action costs revisited. In *Proceedings of the 18th European Conference on Artificial Intelligence (ECAI'08)*, 588–592.

Maliah, S.; Shani, G.; and Stern, R. 2014. Privacy preserving landmark detection. In *Proceedings of the 21st European Conference on Artificial Intelligence (ECAI'14)*, 597–602.

Nissim, R., and Brafman, R. I. 2012. Multi-agent A* for parallel and distributed systems. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'12)*, 1265–1266.

Nissim, R., and Brafman, R. 2014. Distributed heuristic forward search for multi-agent planning. 51:293–332.

Štolba, M., and Komenda, A. 2013. Fast-forward heuristic for multiagent planning. In *Proceedings of the 1st ICAPS Workshop on Distributed and Multi-Agent Planning (DMAP'13)*, 75–83.

Štolba, M., and Komenda, A. 2014. Relaxation heuristics for multiagent planning. In *Proceedings of the 24th International Conference on Automated Planning and Scheduling (ICAPS'14)*, 298–306.

Štolba, M., and Komenda, A. 2015. The MADLA planner: Multiagent planning by combination of distributed and local heuristic search. *in submission.*

Štolba, M.; Fišer, D.; and Komenda, A. 2015. Admissible landmark heuristic for multi-agent planning. In *to appear at ICAPS'15.*

Torreño, A.; Onaindia, E.; and Sapena, O. 2014. FMAP: Distributed cooperative multi-agent planning. *Applied Intelligence* 41(2):606–626.