Proceedings of the Twelfth International
Symposium on Combinatorial Search (SoCS 2019)

Multi-Agent Path Finding with Continuous Time and Geometric Agents Viewed through Satisfiability Modulo Theories (SMT)

Pavel Surynek [*]
Faculty of Information Technology
Czech Technical University in Prague
Thákurova 9, 160 00 Praha 6, Czechia
pavel.surynek@fit.cvut.cz

Abstract

We address a variant of *multi-agent path finding* (MAPF) with continuous time and geometric agents. The standard MAPF is a task of navigating agents in an undirected graph from given starting vertices to given goals so that agents do not collide. In the continuous version (MAPF$^\mathcal{R}$), agents move in the n-dimensional Euclidean space along straight lines that interconnect predefined positions. We present a new solving approach based on *satisfiability modulo theories* (SMT) to obtain makespan optimal solutions. Our SMT-based approach for MAPF$^\mathcal{R}$ called SMT-CBS$^\mathcal{R}$ reformulates the Conflict-based Search (CBS) algorithm in terms of SMT concepts.

Introduction and Background

In *multi-agent path finding* (MAPF) (Silver 2005) the task is to navigate agents from given starting positions to individual goals. The problem takes place in an undirected graph $G = (V, E)$ where agents are placed in its vertices with at most one agent per vertex. The task is to transform the initial configuration of agents $\alpha_0 : A \rightarrow V$ into the goal configuration $\alpha_+ : A \rightarrow V$.

We are dealing here with an extension denoted MAPF$^\mathcal{R}$ introduced only recently (Andreychuk et al. 2019; Walker, Sturtevant, and Felner 2018) that considers continuous time and space. Agents in MAPF$^\mathcal{R}$ move smoothly between predefined positions in the n-dimensional Euclidean space. In MAPF$^\mathcal{R}$ we assume geometric agents of various shapes that occupy certain volume in the space - circles in the 2D space, polygons, spheres in the 3D space etc. A collision is defined as an overlap between agents' bodies.

Our contribution consists in showing how to apply satisfiability modulo theory (SMT) reasoning (Bofill et al. 2012) in the makespan optimal solving of MAPF$^\mathcal{R}$. Makespan optimal solutions minimize the overall time needed to relocate all agents into their goals.

MAPF$^\mathcal{R}$ shares several components with the standard MAPF: the underlying undirected graph $G = (V, E)$, set of agents $A = \{a_1, a_2, ..., a_k\}$, and the initial and goal configuration of agents: $\alpha_0 : A \rightarrow V$ and $\alpha_+ : A \rightarrow V$; but it adds ρ, an extension defining continuous properties as follows:

- $\rho.x(v), \rho.y(v)$, ... for $v \in V$ represent the position of vertex v in the space
- $\rho.velocity(a)$ for $a \in A$ determines velocity of agent a
- $\rho.radius(a)$ for $a \in A$ determines the radius of agent a; we assume that agents are circular/spherical with omni-directional ability of movements

The major difference from the standard MAPF where agents move instantly between vertices is that in MAPF$^\mathcal{R}$ continuous movements of agents between a pair of vertices (positions) along the straight line takes place. Hence we need to be aware of the presence of agents at some point in the space at any time.

Collisions may occur between agents due to their size. In contrast to MAPF, collisions in MAPF$^\mathcal{R}$ occur not only in a single vertex or edge but also on pairs of edges if they are too close to each other w.r.t. to sizes of agents traversing them.

A solution to given MAPF$^\mathcal{R}$ $\Sigma^\mathcal{R}$ is a collection of temporal plans for individual agents $\pi = [\pi(a_1), \pi(a_2), ..., \pi(a_k)]$ that are mutually collision-free. An example of MAPF$^\mathcal{R}$ and its solution is shown in Figure 1.

Solving MAPF with Continuous Time

We observed that *conflict-based search* (CBS) for MAPF (Sharon et al. 2015) operates similarly as problem solving in

Figure 1: An example of MAPF$^\mathcal{R}$ instance with three agents and its makespan optimal solution.

[*]This work has been supported by the Czech Science Foundation - GAČR (grant registration number 19-17966S).

the SMT methodology. CBS uses the idea of resolving collisions lazily; that is, a solution of MAPF instance is searched against an incomplete set of movement constraints. Instead of forbidding all possible collisions between agents we start with initially empty set of collision forbidding constraints that gradually grows as new collisions appear. CBS can be modified to CBS$^{\mathcal{R}}$ a variant for MAPF$^{\mathcal{R}}$ as shown in (Andreychuk et al. 2019).

The basic problem solving in SMT divides the satisfiability problem in some complex theory T into an incomplete propositional part that keeps the Boolean structure and a simplified decision procedure $DECIDE_T$ that decides a conjunctive fragment of T. The standard SAT solver (Audemard and Simon 2009) solves the incomplete propositional part and $DECIDE_T$ then checks if the solution is consistent with axioms of T. If not, an incosistency elimination constraint c is reported back from $DECIDE_T$ and the incomplete model is extended with c for the next iteration.

We rephrased CBS$^{\mathcal{R}}$ in terms of SMT as it has been done with CBS (Surynek 2019). T will be represented by a theory describing movement rules of MAPF$^{\mathcal{R}}$. The plan validation procedure known from CBS$^{\mathcal{R}}$ will act as $DECIDE_T$ and will report back a set of collisions found in the current solution. The propositional part will be taken from existing propositional encodings of the standard MAPF such as the MDD-SAT (Surynek et al. 2016) provided that constraints forbidding collisions between agents will be omitted.

Decision Variable Generation

One important challenge in the SMT approach for MAPF$^{\mathcal{R}}$ is determining the set of decision variables. MDD-SAT introduces decision variables $\mathcal{X}_v^t(a_i)$ and $\mathcal{E}_{u,v}^t(a_i)$ for discrete time-steps $t \in \{0, 1, 2, ...\}$ describing occurrence of agent a_i in v or the traversal of edge $\{u, v\}$ by a_i at time-step t.

A significant difficulty in MAPF$^{\mathcal{R}}$ is that we need decision variables with respect to continuous time. Fortunately we do not need a variable for any possible time but only for important moments. Important moments are determined by conflicts generated by CBS. If for example the duration of a conflict in neighbor v of u is $[t_0, t_+)$ and agent a_i residing in u at $t \geq t_0$ wants to enter v then the earliest time a_i can do so is t_+ since before it would conflict in v. On the other hand if a_i does not want to waste time (let us note that we search for a makespan optimal solution), then waiting longer than t_+ is not desirable. Hence we only need to introduce decision variable $\mathcal{E}_{u,v}^{t_+}(a_i)$ to reflect the situation.

Algorithm 1 shows variable generation up to specified makespan bound μ_{max}. It is a breadth-first w.r.t. time search of the space-time. All possible relevant waitings in a vertex at hand are generated at lines 15-19.

Algorithm 1: Generation of decision variables in the SMT-based algorithm for MAPF$^{\mathcal{R}}$ solving

```
1  generate-Decisions ($\Sigma^{\mathcal{R}} = (G = (V, E), A, \alpha_0, \alpha_+, \rho)$, $conf$, $\mu_{max}$)
2      VAR ← $\emptyset$
3      for each $a \in A$ do
4          OPEN ← $\emptyset$
5          insert $(\alpha_0(a), 0)$ into OPEN
6          VAR ← VAR $\cup \{\mathcal{X}^0_{\alpha_0(a)}(a)\}$
7          while OPEN $\neq \emptyset$ do
8              $(u, t) \leftarrow \min_t$(OPEN)
9              remove-Min$_t$(OPEN)
10             if $t \leq \mu_{max}$ then
11                 for $v \mid \{u, v\} \in E$ do
12                     $\Delta t \leftarrow dist(u, v)/\rho.velocity(a)$
13                     insert $(v, t + \Delta t)$ into OPEN
14                     VAR ← VAR $\cup \{\mathcal{E}^t_{u,v}(a), \mathcal{X}^{t+\Delta t}_v(a)\}$
15                 for $v \mid \{u, v\} \in E \cup \{u, u\}$ do
16                     for $(a, \{u, v\}, [t_0, t_+)) \in conf$ do
17                         if $t_+ > t$ then
18                             insert $(u, t_+)$ into OPEN
19                             VAR ← VAR $\cup \{\mathcal{X}^{t+}_u(a)\}$
20     return VAR
```

Conclusion

We suggested a novel algorithm called SMT-CBS$^{\mathcal{R}}$ for the makespan optimal solving of the multi-agent path finding problem with continuous time and space based on satisfiability modulo theories (SMT).

For the future work we assume extending the concept with other cumulative objectives like the *sum-of-costs* (Sharon et al. 2013). We also plan to extend the node generation scheme to directional agents where we need to add a new dimension in addition to space and time: *direction* (angle).

References

Andreychuk, A.; Yakovlev, K.; Atzmon, D.; and Stern, R. 2019. Multi-agent pathfinding (MAPF) with continuous time. *CoRR* abs/1901.05506.

Audemard, G., and Simon, L. 2009. Predicting learnt clauses quality in modern SAT solvers. In *IJCAI*, 399–404.

Bofill, M.; Palahí, M.; Suy, J.; and Villaret, M. 2012. Solving constraint satisfaction problems with SAT modulo theories. *Constraints* 17(3):273–303.

Sharon, G.; Stern, R.; Goldenberg, M.; and Felner, A. 2013. The increasing cost tree search for optimal multi-agent pathfinding. *Artif. Intell.* 195:470–495.

Sharon, G.; Stern, R.; Felner, A.; and Sturtevant, N. 2015. Conflict-based search for optimal multi-agent pathfinding. *Artif. Intell.* 219:40–66.

Silver, D. 2005. Cooperative pathfinding. In *AIIDE*, 117–122.

Surynek, P.; Felner, A.; Stern, R.; and Boyarski, E. 2016. Efficient SAT approach to multi-agent path finding under the sum of costs objective. In *ECAI*, 810–818.

Surynek, P. 2019. Unifying search-based and compilation-based approaches to multi-agent path finding through satisfiability modulo theories. In *IJCAI*, in press.

Walker, T. T.; Sturtevant, N. R.; and Felner, A. 2018. Extended increasing cost tree search for non-unit cost domains. In *IJCAI*, 534–540.