Gliders2014: Dynamic Tactics with Voronoi Diagrams

Mikhail Prokopenko, Peter Wang, and Oliver Obst

CSIRO Computational Informatics, Statistical Learning
PO Box 76, Epping, NSW 1710, Australia
email: mikhail.prokopenko@csiro.au

Abstract. The RoboCup Soccer 2D Simulation League demands well-developed tactical skills. In this paper we describe a mechanism employed by our team, Gliders2014, during a tactic-dependent evaluation and selection of players' actions. The mechanism is based on Voronoi diagrams and is sufficiently generic to be applicable to other soccer Leagues. The proposed approach has been successfully applied to tactical multi-agent interactions, leading to an improved performance against benchmarks.

1 Introduction

The RoboCup Soccer 2D Simulation is the League which requires not only autonomous decision-making under constraints, but also quite sophisticated tactical plans and teamwork, specifically selected, evolved or adapted to different opponents [1,2,3,4,5,6,7]. Without a doubt, tactical proficiency required by and demonstrated in the 2D Simulation League is the highest of all RoboCup leagues. These days, low-level skills are quite similar across the participants, and so, arguably, the top Simulation 2D teams achieve their dominance by tactical flexibility, rather than some newly found excellence in basic behaviours.

In this short paper we describe a mechanism of action selection which depends on a dynamic tactical scheme. This mechanism is based on Voronoi diagrams, utilised by Gliders [8,9] — a simulated soccer team for the RoboCup soccer 2D simulator [10]. Gliders2012 and Gliders2013 reached semi-finals in RoboCup tournaments of 2012 and 2013. The team code is written in C++ using agent2d: the well-known base code developed by Akiyama et al. [11], and fragments of released source code of Marlik [12]. Other software packages are used as well:

- librcsc: a base library for The RoboCup Soccer Simulator (RCSS) with various utilities describing relevant geometrical constructs, world model, etc.;
- soccerwindow2: a viewer program for RCSS, working as a monitor client, a log player and a visual debugger;
- fedit2: a formation editor for agent2d, allowing to design a team formation;
- Gliders' in-browser basic soccermonitor (GIBBS): a log-player for viewing 2D Simulation League logs over web browser [13].

GIBBS is used to visualize matches in a browser window [13]. At this time, the main implementation uses HTML5 / javascript, with a pre-processing step to create simple-to-parse log files that can be conveniently transferred over the Internet. The current state of the visualization is basic but functional, and it is released to the RoboCup community in the hope that it will be improved over time, re-released, and be used for showing games to general public in future competitions. The GIBBS log player is available (along with the necessary converter) at the Gliders2012 web page www.oliverobst.eu/research/robotics-gliders2012-simulation-league-robocup-team.

Most of the Gliders source code has been released in 2013, and can be downloaded from the Official Data Repository of RoboCup 2D Soccer Simulation League: www.socsim.robocup.org/files/2D/binary/RoboCup2012/.

2 Motivation and approach

Better mobility and more comprehensive field control are critical to a successful performance in RoboCup Soccer. This idea can be traced to a generic framework describing abstract spatio-temporal relationships, suggested by Dylla et al. [14]. The latter work argued that a reachability relation is needed to express spatial relationships between the players and the ball. They suggested to use Voronoi diagrams: a Voronoi diagram is the partitioning of a plane with n points into n convex polygons such that each polygon contains exactly one point and every point in the given polygon is closer to its central point than any other [14]. This was further developed by Akiyama et al. who used a dual representation of Voronoi diagrams — the Delaunay triangulation [15,16]. Voronoi diagrams are also known to emerge as a result of parallel computation, providing an optimal spatial configuration for a robust resource distribution and quick transportation [17].

2.1 Action-dependent evaluation

Tactical schemes of Gliders2014 also make use of Voronoi diagrams, by following an action-dependent evaluation, introduced in Gliders2012 [8]. Specifically, at any given time, there is a number m of tactics represented by a set of desirable states: $\{S_1, \ldots, S_m\}$, and the feasible actions are partitioned into m sets: $A_1, \ldots, A_m$, so that for every action a_i, there is a set A_j such that $a_i \in A_j$. We denote the function mapping an action to its tactical state by

$$\mathit{tactics} : a \to S \ . \tag{1}$$

Then each feasible action is rated with respect to the corresponding desirable state:

$$r(a) = D(S = \mathit{result}(a), S^* = \mathit{tactics}(a)) \ , \tag{2}$$

where D is a simple distance metric, identical for all actions.

The action which is selected is simply the one that minimizes the distance between resultant and desirable states, i.e., minimizes this metric:

$$a^* = \arg\min_a r(a) \ . \tag{3}$$

Importantly, the desirable states that the player is trying to reach are not independent of actions, but rather *are* action-dependent, and this dependence is tactical. The definition (2) allows for different desirable states $S_1 = \mathit{tactics}(a_1)$ and $S_2 = \mathit{tactics}(a_2)$, where $a_1 \neq a_2$.

2.2 *Tactics* function

In this subsection we describe the mechanism implementing the function $S = tactics(a)$, that is, the process which assigns a desirable state dependent on the action in point. The first step of the process is construction of a *Voronoi diagram*, where the n points are given by $n = 11$ opponent players. For example, Figure 1 shows a Voronoi diagram, at time 105 in the game between agent2d (left team) and Gliders2014 (right team). The figure is produced using soccerwindow2, with light-blue lines showing resultant Voronoi segments, with yellow/green circles corresponding to agent2d players/goalkeeper. The dark-blue line, originating from Gliders player 9, indicates a possible action, a through-pass a, terminating at some possible resultant state $S(a)$, marked by a blue dot.

The second step of the process is construction of a set of possible *attacking nodes* $V(a)$, through which the team may develop an attack further. This set may, for example, include the Voronoi vertices (nodes), i.e. the points equidistant to three (or more) points, as well as nodes located at intersections between Voronoi segments and some lines, e.g., offside line, and/or line parallel to the offside line but set some distance deeper into the opponent field.

The next, third, step is setting *constraints* on the set $V(a)$, that is, constructing $V'(a) \subset V(a)$. In other words, only subset $V'(a)$ should be considered for the choice of the best attacking node: for example, only those nodes which are within a certain radius to the possible resultant state, and which have the x-coordinate larger than some value, dependent on the action's resultant state. Figure 1 illustrates these constraints by the dark-blue circle around the possible resultant state (radius equal to 34 meters) and the dark-blue line set 25 meters behind the x-coordinate of the resultant state. All elements of $V(a)$ (Voronoi vertices and intersection nodes) within the circle and to the left of the line are marked with red dots. All these nodes are possible nodes for promising continuation of the attack, being equidistant to opponent players.

The final step is the selection of one of these vertices as the most promising attacking node. At this step, the agent considering the action decides which of those vertices $v^* \in V(a)$ is the *best attacking node* trough which the team may develop an attack. This selection may be implemented in many ways, e.g. by maximising the difference between smallest distances from a vertex to (i) teammates and (ii) opponents. That is, for each vertex $v \in V'(a)$, the agent finds first the nearest opponent, at distance $d_o(v)$, then the nearest teammate, at distance $d_t(v)$, and determines the difference $d(v) = d_o(v) - d_t(v)$. Having determined these differences for all the nodes in $V'(a)$, the agent selects the vertex $v^* = \arg\max_{v \in V'(a)} d(v)$. In other words, the node which has a relatively close teammate and a relatively distant opponent is deemed to be the best attacking node. Figure 1 marks the best attacking node with a small red star — this is the Voronoi vertex closest to player 8. The field position of node v^*, selected in this way, is the final outcome: $S^*(a) = position(v^*) = tactics(a)$. We note that in agent2d, the best attacking node is always the centre of opponent's goal.

The distance between the action resultant state $S = result(a)$ and the best attacking node $S^* = tactics(a)$, captured by equation (2) evaluates the quality of the possible action under consideration.

This selection may appear to make sense only in context of a particular tactic. For example, the best attacking node v^*, marked with a small red star in Figure 1, may make sense when a chain of passes is considered from player 9, in possession of the ball, to player 8 near the the node v^*. It may not, at the first glance, appear sensible with respect to the

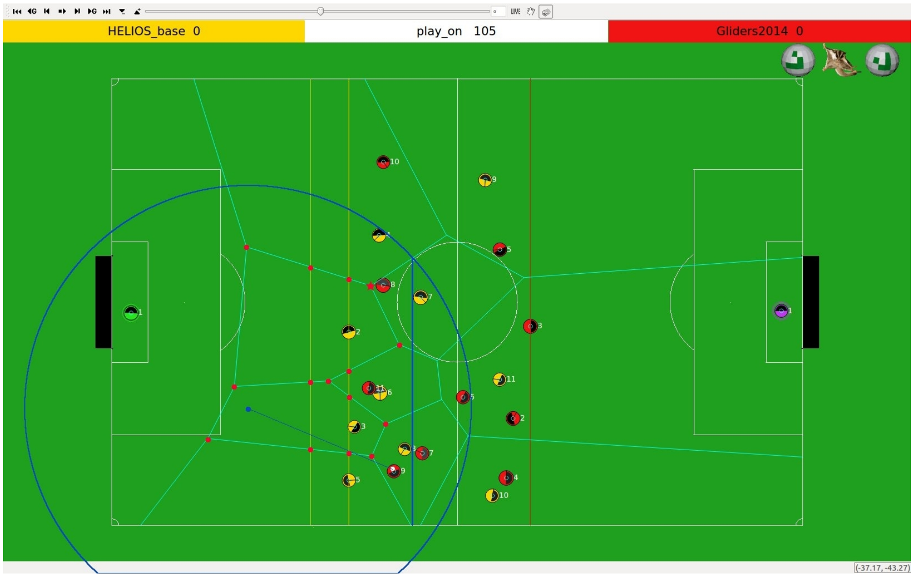

Fig. 1. The mechanism implementing $S = tactics(a)$. Voronoi diagram in the game between agent2d (left team) and Gliders2014 (right team) is shown at time = 105, with the n points given by $n = 11$ agent2d players (yellow/green circles). Light-blue lines show Voronoi segments. The dark-blue line, originating from Gliders player 9, indicates a possible action, a, terminating at possible resultant state $S(a)$, marked by a blue dot. A subset of possible attacking nodes $V'(a)$ is shown with red dots, within the constraints shown by the dark-blue circle around state $S(a)$, and the dark-blue line set behind the x-coordinate of $S(a)$. The best attacking node is marked with a small red star.

through pass a, illustrated in Figure 1. However, we argue that a significant subset of the actions can be judged with respect to the distance to this node, which represents a genuinely promising location for an attack's development, being relatively far from the opponents and close to a teammate. Figure 2 shows the state of play 19 cycles later than that of Figure 1, showing that in fact player 11 responded to the through pass, while the player 8 stayed deeper.

Of course, some other actions may be grouped with respect to another node. That is, for other actions, the node v^*, marked with the small red star, may not satisfy the constraint, being outside of the blue circle in Figure 1. This illustrates the concept of the action-dependence in Gliders' evaluation process.

3 Conclusion

We described a mechanism of action selection driven by a dynamic tactical scheme, based on Voronoi diagrams. The mechanism is utilised by Gliders, a simulated soccer team participating in the RoboCup 2D Simulation League. It is sufficiently generic and may be applicable to other soccer Leagues, as well as team sports. The approach has been successfully applied to tactical multi-agent interactions, improving performance against top Simulation League teams and other benchmarks, e.g. agent2d.

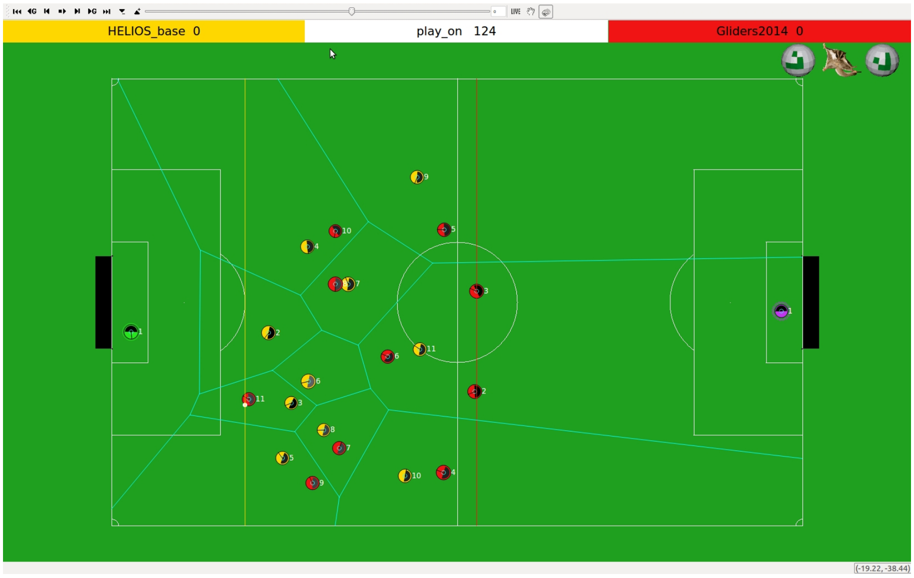

Fig. 2. The state of play 19 cycles after the state shown in Figure 1.

Acknowledgments We thank Jason Held and Saber Astronautics for their contribution to Gliders2012 [8], David Budden for developing new self-localisation method introduced in Gliders2013 [18], Oliver Cliff for developing tools for quantifying interaction networks and tactical analysis [19], and Brendon Duncan and Lincoln Flintoft for developing off-line trainer scripts.

Some of the Authors have been involved with RoboCup Simulation 2D in the past, however the code of their previous teams (Cyberoos and RoboLog, see, e.g., [7,20]) is not used in Gliders.

References

1. Noda, I., Stone, P.: The RoboCup Soccer Server and CMUnited Clients: Implemented Infrastructure for MAS Research. Autonomous Agents and Multi-Agent Systems **7**(1–2) (July–September 2003) 101–120
2. Riley, P., Stone, P., Veloso, M.: Layered disclosure: Revealing agents' internals. In Castelfranchi, C., Lesperance, Y., eds.: Intelligent Agents VII. Agent Theories, Architectures, and Languages — 7th. International Workshop, ATAL-2000, Boston, MA, USA, July 7–9, 2000, Proceedings. Lecture Notes in Artificial Intelligence. Springer, Berlin, Berlin (2001)
3. Stone, P., Riley, P., Veloso, M.: Defining and using ideal teammate and opponent models. In: Proceedings of the Twelfth Annual Conference on Innovative Applications of Artificial Intelligence. (2000)
4. Butler, M., Prokopenko, M., Howard, T.: Flexible synchronisation within RoboCup environment: A comparative analysis. In: RoboCup 2000: Robot Soccer World Cup IV, London, UK, Springer (2001) 119–128

5. Reis, L.P., Lau, N., Oliveira, E.: Situation based strategic positioning for coordinating a team of homogeneous agents. In: Balancing Reactivity and Social Deliberation in Multi-Agent Systems, From RoboCup to Real-World Applications (selected papers from the ECAI 2000 Workshop and additional contributions), London, UK, Springer (2001) 175–197
6. Prokopenko, M., Wang, P.: Relating the entropy of joint beliefs to multi-agent coordination. In Kaminka, G.A., Lima, P.U., Rojas, R., eds.: RoboCup 2002: Robot Soccer World Cup VI. Volume 2752 of Lecture Notes in Computer Science., Springer (2003) 367–374
7. Prokopenko, M., Wang, P.: Evaluating team performance at the edge of chaos. In Polani, D., Browning, B., Bonarini, A., Yoshida, K., eds.: RoboCup 2003: Robot Soccer World Cup VII. Volume 3020 of Lecture Notes in Computer Science., Springer (2003) 89–101
8. Prokopenko, M., Obst, O., Wang, P., Held, J.: Gliders2012: Tactics with action-dependent evaluation functions. In: RoboCup 2012 Symposium and Competitions: Team Description Papers, Mexico City, Mexico, June 2012. (2012)
9. Prokopenko, M., Obst, O., Wang, P., Budden, D., Cliff, O.: Gliders2013: Tactical analysis with information dynamics. In: RoboCup 2013 Symposium and Competitions: Team Description Papers, Eindhoven, The Netherlands, June 2013. (2013)
10. Chen, M., Dorer, K., Foroughi, E., Heintz, F., Huang, Z., Kapetanakis, S., Kostiadis, K., Kummeneje, J., Murray, J., Noda, I., Obst, O., Riley, P., Steffens, T., Wang, Y., Yin, X.: Users Manual: RoboCup Soccer Server — for Soccer Server Version 7.07 and Later. The RoboCup Federation. (February 2003)
11. Akiyama, H.: Agent2D Base Code. http://www.rctools.sourceforge.jp (2010)
12. Tavafi, A., Nozari, N., Vatani, R., Yousefi, M.R., Rahmatinia, S., Pirdir, P.: MarliK 2012 Soccer 2D Simulation Team Description Paper. In: RoboCup 2012 Symposium and Competitions: Team Description Papers, Mexico City, Mexico, June 2012. (2012)
13. Moore, E., Obst, O., Prokopenko, M., Wang, P., Held, J.: Gliders2012: Development and competition results. CoRR **abs/1211.3882** (2012)
14. Dylla, F., Ferrein, A., Lakemeyer, G., Murray, J., Obst, O., Röfer, T., Schiffer, S., Stolzenburg, F., Visser, U., Wagner, T.: Approaching a Formal Soccer Theory from the Behavior Specification in Robotic Soccer. Bioengineering. In: Computers in Sport. WIT Press (2008) 161–186
15. Akiyama, H., Noda, I.: Multi-agent positioning mechanism in the dynamic environment. In Visser, U., Ribeiro, F., Ohashi, T., Dellaert, F., eds.: RoboCup 2007: Robot Soccer World Cup XI. Springer, Berlin, Heidelberg (2008) 377–384
16. HELIOS2010 Team Description. In: RoboCup 2010: Robot Soccer World Cup XIV. Volume 6556 of Lecture Notes in Computer Science., Springer (2011)
17. Adamatzky, A., Prokopenko, M.: Slime mould evaluation of Australian motorways. IJPEDS **27**(4) (2012) 275–295
18. Budden, D., Prokopenko, M.: Improved particle filtering for pseudo-uniform belief distributions in robot localisation. In: RoboCup 2013: Robot Soccer World Cup XVII, Springer (2013)
19. Cliff, O., Lizier, J., Wang, R., Wang, P., Obst, O., Prokopenko, M.: Towards quantifying interaction networks in a football match. In: RoboCup 2013: Robot Soccer World Cup XVII, Springer (2013)
20. Obst, O., Boedecker, J.: Flexible coordination of multiagent team behavior using HTN planning. In Noda, I., Jacoff, A., Bredenfeld, A., Takahashi, Y., eds.: RoboCup 2005: Robot Soccer World Cup˜IX. Lecture Notes in Artificial Intelligence. Springer, Berlin, Heidelberg, New York (2006) 521–528