Real-Time Strategy Game Competitions

Michael Buro, David Churchill

■ *In recent years, real-time strategy (RTS) games have gained attention in the AI research community for their multitude of challenging and relevant real-time decision problems that have to be solved in order to win against human experts or to collaborate effectively with other players in team games. In this report we motivate research in this area, give an overview of past RTS game AI competitions, and discuss future directions.*

RTS games — such as StarCraft by Blizzard Entertainment and Command and Conquer by Electronic Arts — are popular video games that can be described as real-time war simulations in which players delegate units under their command to gather resources, build structures, combat and support units, scout opponent locations, and attack. The winner of an RTS game usually is the player or team that destroys the opponents' structures first.

Unlike abstract board games like chess and go, moves in RTS games are executed simultaneously at a rate of at least eight frames per second. In addition, individual moves in RTS games can consist of issuing simultaneous orders to hundreds of units at any given time. If this wasn't creating enough complexity already, RTS game maps are also usually large and states are only partially observable, with vision restricted to small areas around friendly units and structures. Complexity by itself, of course, is not a convincing motivation for studying RTS games and building AI systems for them. What makes them attractive research subjects is the fact that, despite the perceived complexity, humans are able to outplay machines by means of spatial and temporal reasoning, long-range adversarial planning and plan recognition, state inference, and opponent modeling, which — we hypothesize — are enabled by powerful hierarchical state and action abstractions.

RTS Game Competitions

An established way of improving AI systems for abstract games like chess has been running competitions. Applying this idea to commercial RTS games is tricky, because they usually don't provide programming interfaces. So, in 2002 we set out creating a free software RTS game system — Open Real-Time Strategy (ORTS),[1] which was the platform used in four annual game competitions held from 2006 to 2009. There were 4–5 game categories focusing on important subtasks — such as resource gathering (which is an exercise in multiagent path finding) and small-scale combat — and a full-featured RTS game including partial observability. The ORTS competitions mostly attracted academic entries and benefited from the requirement to release source code.

Since the advent of the Brood War Application Programming Interface (BWAPI) C++ library in 2009, the focus has quickly moved from ORTS to StarCraft. StarCraft by Blizzard Entertainment is one of the most popular commercial RTS games of all time. BWAPI not only allows programs to interact with the game engine directly to play autonomously, but also makes it easy to play games against strong human players, which is ideal for measuring progress.[2]

The first StarCraft AI competition was organized by Ben Weber at the University of California, Santa Cruz in 2010 as part of the AAAI Artificial Intelligence and Interactive Digital Entertainment (AIIDE) conference program.[3] There were four game categories, which ranged from skirmish to the full StarCraft game. There were 17 entries for the full game and 34 entries overall in all categories.

One year later, the second AIIDE StarCraft AI competition was held at the University of Alberta. An international field of 18 entries from seven different countries entered the competition, consisting of 13 academic and 5 independent teams. Team sizes varied from a single person to an entire AI class from Berkeley. Using newly developed tournament software, 2,360 full games were played distributed over 20 lab machines during four days in August. Two years in a row, the authors of the three best programs in the AIIDE competitions received StarCraft 2 collector's edition signed by the StarCraft team — a valuable prize donated by Blizzard Entertainment. To speed up development for future competitions, releasing the source code was made mandatory. Since then, a couple more annual StarCraft AI competitions have been initiated, such as CIG 2011 and SSCAI 2011. A complete competition list is available at the BWAPI website.[4]

State of the Art RTS Game AI Systems

The current state of the art in StarCraft bots comprises a mix of AI algorithms and scripted strategies. Because StarCraft is such a complex game, most bots are decomposed into components corresponding to strategic elements of the game, such as economic planning, high-level decision making, and unit micromanagement. Due to the high complexity of these strategic elements, most bots implement these components as a series of scripted rules based on observations from expert-level human play. However, over the past two years we have seen more intelligent AI solutions tackle these problems.

Each of the top-performing bots seems to have its own specialty AI solutions to one or more of these elements. For instance, the 2010 winner, Berkeley's Overmind (Huang 2011), used potential fields to guide the attack formations of its flying units, making their attacks much more effective. Skynet, the 2011 winner, used path finding to guide its units to safety to avoid being surrounded. UAlbertaBot, the 2011 second place finisher, used a heuristic search algorithm to plan its economic build orders (Churchill and Buro 2011), resulting in more efficient unit production. Other bots implemented particle models for state estimation (Weber, Mateas, and Jhala 2011), plan recognition (Synnaeve and Bessire 2011a), and Bayesian models for unit control (Synnaeve and Bessire 2011b). The source code of the 2011 competition entries can be downloaded from the competition website to serve as a starting point for subsequent competitions.

AI solutions have a large advantage over hard-coded scripts due to their ability to adapt to changes in the environment or opponent strategy. However, the most important aspect of any RTS game — high-level strategy decisions — still evades AI solutions. The 2011 Man Versus Machine Match saw former StarCraft pro Oriol Vinyals easily defeat Skynet, the winner of the computer competition. Despite strong opening play by Skynet, Vinyals was able to exploit the bot's inability to adapt to his midgame strategy changes. "Overall," commented Vinyals after the match, "Skynet had good macro, generally good micro, but bad strategic decisions and [lack of scouting/prediction]." Figure 1 shows crucial game situations Skynet mishandled.

2012 RTS Game AI Competitions

When humans play a game, they often take time

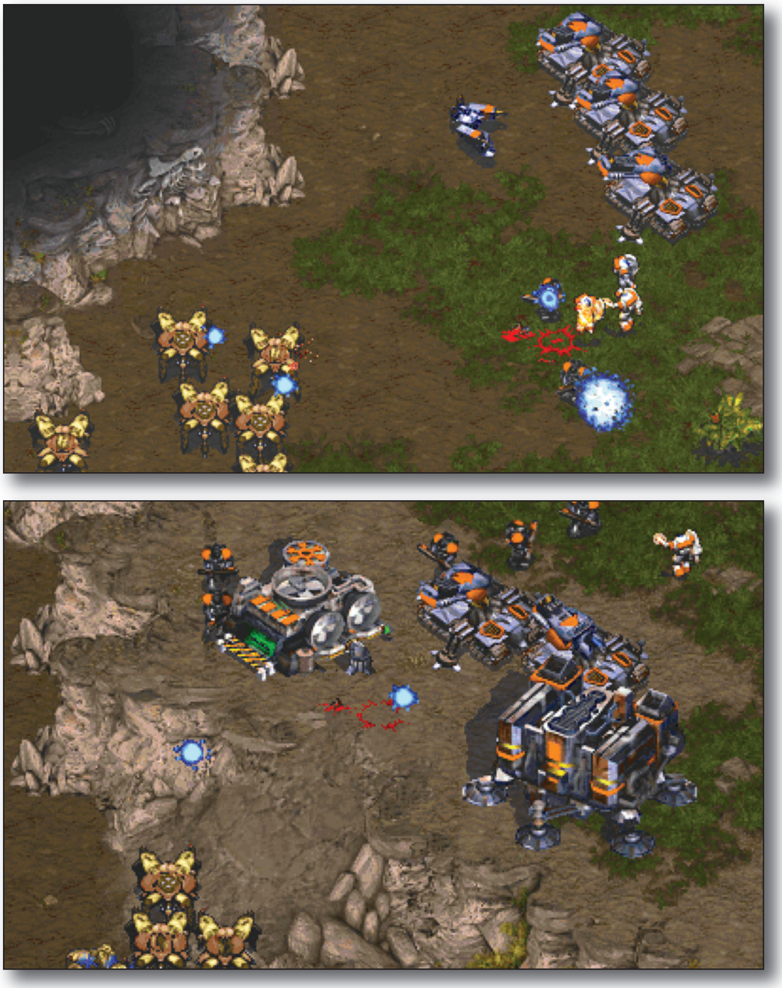

Figure 1. 2011 AIIDE Man Versus Machine Competition — Oriol Vinyals Versus Competition Winner Skynet.

Despite Skynet's strong openings a lack of adaptation and poor attack targeting cost it the match. In (a) Skynet playing as Protoss (bottom left) attempts an early attack on the human's base, which is defended by siege tanks. The bot doesn't realize that it can't succeed. In the final battle of the match (shown in b), Skynet focused on attacking Vinyals's medics, rather than attacking the tanks, allowing them to push through the defenses.

to study their opponents to learn their strategy, strengths, and weaknesses. Up until now, StarCraft competitions have not allowed bots to retain data from previously played games. In 2011 we saw many early aggression bots that were able to defeat their opponents in the same way every game. The 2012 AIIDE StarCraft AI competition, which was held in August 2012 again at the University of Alberta, allowed bots to store data about previous matches, thereby forcing bots to implement a wider variety of strategies. Also, until now software limitations did not allow bots to choose their race at random, which was fixed for the 2012 competition, adding an additional strategic element to the game. The results of the 2012 AIIDE StarCraft AI competition will be presented at the AIIDE conference in October, which also will host a workshop on RTS game AI. The CIG 2012 competition will be held in September, and the competition at SSCAI 2012 is still in the planning stage.

Notes

1. The ORTS website is located at www.cs.ualberta.ca/˜mburo/orts.

2. The C++ StarCraft BroodWar Interface (BWAPI) can be found at code.google.com/p/bwapi.

3. See AIIDE StarCraft AI Competition. The URLS are eis.ucsc.edu/StarCraftAICompetition (for 2010), www.cs.ualberta.ca/~mburo/sc2011 (for 2011) and starcraftaicompetition.com (for 2012). The Twitter account can be found at twitter.com/StarCraftAIComp.

4. See code.google.com/p/bwapi/wiki/Competitions.

References

Churchill, D., and Buro, M. 2011. Build Order Optimization in StarCraft. In *Proceedings, The Seventh AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 14–19. Menlo Park, CA: AAAI Press.

Huang, H. 2011. Skynet Meets the Swarm: How the Berkeley Over-Mind Won the 2010 StarCraft AI Competition. *Ars Technica* (18 January).

Synnaeve, G., and Bessire, P. 2011a. A Bayesian Model for Plan Recognition in RTS Games Applied to StarCraft. In *Proceedings, The Seventh AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*. Menlo Park, CA: AAAI Press.

Synnaeve, G., and Bessire, P. 2011b. A Bayesian Model for RTS Unit Control Applied to StarCraft. In *Proceedings of the 2011 IEEE Conference on Computational Intelligence and Games*, 190–196. Piscataway, NJ: Institute of Electrical and Electronics Engineers.

Weber, B. G.; Mateas, M.; and Jhala, A. 2011. A Particle Model for State Estimation in Real-Time Strategy Games. In *Proceedings, The Seventh AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment,* 103–108. Menlo Park, CA: AAAI Press.

Michael Buro is an associate professor in the Computing Science Department at the Unversity of Alberta.

David Churchill is a PhD student in the Computing Science Department at the University of Alberta.