AKBABA: The KBSG 2013 Team for the Angry Birds AI Competition

Stefan Schiffer and **Maxim Jourenko** and **Gerhard Lakemeyer**
Knowledge-based Systems Group
RWTH Aachen University, GERMANY
{schiffer,jourenko,gerhard}@kbsg.rwth-aachen.de
aibirds@kbsg.rwth-aachen.de

Abstract

This paper describes Akbaba, the KBSG's agent entry for the 2013 Angry Birds AI Competition (ABAIC) held at *IJCAI–13*. We describe our research background and present the design of our agent, thereby briefly showing how we differ from the sample agent delivered with the ABAIC framework.

Figure 1: Akbaba team logo

1 Introduction

The Knowledge-based Systems Group values (scientific) competition as a means to compare and benchmark research results. As an example, we have tested and deployed various AI methods on our domestic service robot [Schiffer *et al.*, 2012] which we competed with in RoboCup@home [Schiffer *et al.*, 2009]. We have also investigated how logic-based action languages can be used in agents for interactive computer games [Jacobs *et al.*, 2005a; 2005b].

Our research areas mainly focus on artificial intelligence and knowledge representation where, amongst others, cognitive robotics and interactive computer games serve as our application areas. To this end, we gladly enter the Angry Birds AI Competition to further develop and apply our research. In particular, we intend to adopt our previous research on qualitative spatial representations [Schiffer *et al.*, 2011] and logic-based agent control [Schiffer *et al.*, 2012] in domestic service robotics for the AI Birds domain.

While we plan to use a logic-based agent as a high-level controller eventually, for the first steps reported on in this paper we focus on the lower-level basis. In our approach, the agent uses hierarchical search to find optimized parameters for launching a bird in an instance of a game of Angry Birds. In this search, we use simulation as a means to weigh different alternatives.

2 Approach

As already mentioned, our approach is to use search to find an appropriate parameterization for launching a bird. That is, we search for the angle and the speed of a trajectory, to hit a specific target point in the scene. While the parameters to set in the game are the angle and the strength by which to pull the slingshot, we approach the set of alternatives from the other end. This means that in our search, we first consider different target positions to reach in the scene. Only then do we evaluate different possibilities to reach that target in terms of incoming angle and speed of the bird. We do so to guide our search more efficiently towards interesting targets which we will describe in the following.

Search

To navigate through the search space faster, we search for the parameters of a shot in a hierarchical fashion. That is, we split the search space in any dimension in parts and test a representative for each part, where we estimate the outcome for any such representative by means of simulation. Then, we only investigate those parts further and in more detail that yielded good results.

We do implement this hierarchical approach using qualitative (spatial) abstractions as follows. We use Quadtrees [Samet, 1984] in the subdivision of searching for a target point. That is, when searching for possible target positions we first generate possible hit points along the edges of objects. Then, we subdivide the scene with those points using quadtrees (cf. Figure 2). At any level of detail, we test the nodes at that level by running a simulation with the center of that node as a target point. We only investigate higher levels of detail for a certain percentage of nodes using the estimated score from the simulation as our ranking criterion. Similarly, we use abstractions in finding appropriate parameters for the trajectory to use in order to hit the target point selected in the step described above. For subdividing the trajectory in searching for the point to click in order to activate a special bird action as well as for different impact angles we started using qualitative spatial representations based on [Clementini *et al.*, 1997]. We make use of the so-called level of granularity in subdividing the angles of and the distances along the

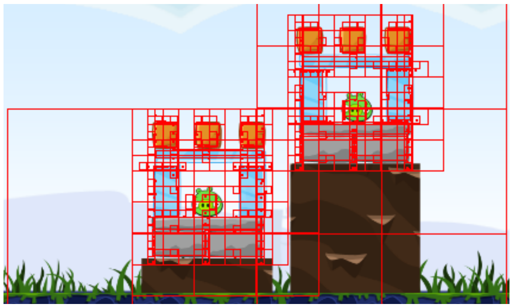

Figure 2: Quadtree of a scene, used to hierarchically structure the potential targets for simulation

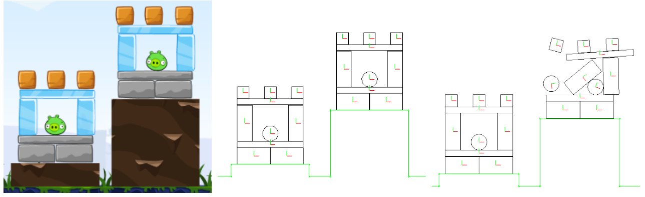

Figure 3: Original Angry Birds game scene and the same scene in simulation and in action after testing a shot

trajectory to build a hierarchy which we use to descend within our search.

Simulation

For the simulation, we use Box2D[1] which is allegedly also the physics engine that is being used by Angry Birds itself.[2] In the simulator we try different shots in order to find those shot parameters that maximize the estimated score. The estimation uses a heuristics that tries to replicate the score achieved in the actual Angry Birds game by the corresponding shot. An example of a scene and the replication in our simulation is given in Figure 3.

The quality of our solution cleary depends on the accuracy of our simulation. Therefore we need to mimic the physics of Angry Birds[3] as closely as possible and we intend to improve this over time by gathering more and more sample data to learn the parameters from (like humans do with repeatedly playing the game).

To further improve on the quality of our simulation we depend on as accurate input data for the scene as possible. This is why we also made some improvements to the vision system. As an example, we determine the rotation of objects and we correct bounding boxes that are often recognized too small by the stock vision.

Gameplay

As for the gameplay we proceed as follows. We first give each level one try. After that, we retry those levels that we failed in the first attempt. For the second attempt, however, we broaden some of the parameters that are used in our search to increase the likelihood of finding a solution.

3 Discussion & Future Work

We are still in an early stage of development and many of our higher-level components are yet to be developed and fine tuned. However, our participation in the 2013 Angry Birds AI Competition should reveal the general feasibility of our approach. With a solid ground work we feel to be in a good starting position for our future endeavours.

For future work we plan to try alternative and more sophisticated strategies for our abstraction mechanisms. One such alternative could be to use clustering in order to group objects for our target search. Once our search strategy has proven useful we will deploy ReadyLog [Ferrein and Lakemeyer, 2008] as a high-level controller as it was previously done in robotics [Schiffer *et al.*, 2012]. We will also further adapt fuzzy representations and control [Schiffer *et al.*, 2011] for Angry Birds domain and use it in level planning with the ReadyLog-based agent.

References

[Clementini *et al.*, 1997] Eliseo Clementini, Paolino Di Felice, and Daniel Hernandez. Qualitative representation of positional information. *Artificial Intelligence*, 95(2):317–356, 1997.

[Ferrein and Lakemeyer, 2008] Alexander Ferrein and Gerhard Lakemeyer. Logic-based robot control in highly dynamic domains. *Robotics and Autonomous Systems, Special Issue on Semantic Knowledge in Robotics*, 56(11):980–991, 2008.

[Jacobs *et al.*, 2005a] Stefan Jacobs, Alexander Ferrein, and Gerhard Lakemeyer. Controlling unreal tournament 2004 bots with the logic-based action language GOLOG. In *Proceedings of the First AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE)*, pages 151–152. AAAI Press, 2005.

[Jacobs *et al.*, 2005b] Stefan Jacobs, Alexander Ferrein, and Gerhard Lakemeyer. Unreal GOLOG bots. In *IJCAI-05 Workshop on Reasoning, Representation, and Learning in Computer Games*, pages 31–36, 2005.

[Samet, 1984] Hanan Samet. The quadtree and related hierarchical data structures. *ACM Computing Surveys*, 16(2):187–260, June 1984.

[Schiffer *et al.*, 2009] Stefan Schiffer, Tim Niemüller, Masrur Doostdar, and Gerhard Lakemeyer. AllemaniACs@Home 2009 Team Description. In *Proceedings CD RoboCup 2009*, Graz, Austria, 2009.

[Schiffer *et al.*, 2011] Stefan Schiffer, Alexander Ferrein, and Gerhard Lakemeyer. Reasoning with qualitative positional information for domestic domains in the situation calculus. *Journal of Intelligent & Robotic Systems*, 66:273–300, 2011.

[Schiffer *et al.*, 2012] Stefan Schiffer, Alexander Ferrein, and Gerhard Lakemeyer. Caesar – an intelligent domestic service robot. *Journal of Intelligent Service Robotics*, 5:259–273, 2012.

[1] http://www.box2d.org/

[2] http://blog.blprnt.com/blog/blprnt/angry-birds-box2d-an-opensource-holiday-wish

[3] http://www.wired.com/wiredscience/2010/10/physics-of-angry-birds/