Project Proposal

Hui Tu (htu@andrew.cmu.edu)
Chenkai Kuang(chenkaik@andrew.cmu.edu)
15-780: Graduate Artificial Intelligence March, 2013

Exploring Ms. Pac-Man Playing Agent

During the last two decades, games have always been used as a test platform for Artificial Intelligence(AI) algorithm. In the same period, game industry becomes one of the fastest-growing industries in the world with sophisticated technology in game consoles and development of mobile device and social gaming as well. Moreover, applying AI to games has achieved some very notable successes such as Deep Blue, a chess playing machine that defeated world-champion Garry Kasparov in 1997. As more fields, no matter industry or academia, become aware of the importance of AI research on game playing strategy, the variety of discussion increases.

Ms. Pac-Man is a popular maze game in recent AI research. While it is no longer the latest or most advanced example of game development, it still provide a classical platform that is complex and challenge enough to require intelligent strategies for superb gameplay. In recent years, IEEE Computational Intelligence in Games Conference(CIG) has hosted annual competitions that allow AI controller design for Ms. Pac-Man to complete for the highest score, which reflects its popularity.

Looking back in the history of AI competitions of Pac-Man and previous researches, most of these work can be divided into two groups: controllers with hand-coded approaches based on human-defined rules and agents that use Computational Intelligence techniques partially or completely. One of the most impressive hand-coded approach is ICE Pambush 2[1], which extracts the game objects from the screen capture and moves Pac-Man based on path costs. Several techniques from all Computational Intelligence paradigms have been applied to Pac-Man agents, such as Neural Network[2], Genetic Algorithms[3] and Reinforcement Learning[4]. According to overall competition statics, the final winner of every competition is the program that utilizes rule-based system, which has high stability and makes speedy decision. However, rule-based systems can't cover exceptional circumstances and are powerless in situations that are difficult or impossible to describe using conditional statements.

Our work will be based on the Ms. Pac-Man AI competition platform[5] and apply several different approaches, including using A* search to find the best path and one of computational intelligence paradigms such as genetic algorithm and neural network. The major challenge in A* search is to find a proper destination in real time and compute good admissible heuristics from data extracted from game screen capture. Building a genetic algorithm from scratch is also not a trivial task. We should also consider the tradeoff between speedy decision and high performance. Moreover, based on what we have implemented, it will a good try to combine the two approaches together, namely, use rule-based system as primary strategy and apply computational intelligence method when the situation is hard to describe using conditional statements.

Project Plan

We provide the following plan, with 75%, 100%, and 125% cutoffs corresponding to how much we will have done by the May poster session given an overestimation, correct estimation, or underestimation of the amount of work this project will require.

75% Plan
1. Become familiar with the Ms. Pac-Man vs. Ghost related research literature
2. Learn to use Ms. Pac-Man vs. Ghost Java start package.
3. Design and implement Best-Path algorithm based on A* search to find the best path for Ms. Pac-Man.
4. Test Best-Path algorithm in Ms. Pac-Man vs. Ghost Platform from most recent CIG competition.

100% Plan
5. Explore how Ms. Pac-Man can be modeled into genetic algorithm or evolutionary neural network algorithm. There are some previous researches on both algorithm, however, there is no clear conclusion which one is better. We will try to explore how well the two can be matched into Ms. Pac-Man and choose one to implement.
6. Build either evolutionary algorithm from scratch is not easy. It needs a lot of work, both on modeling the game environment into program and finding a way to determine if the implementation is correct. Meanwhile, as Ms. Pac-Man is a real-time game, the agent needs to make a decision in limited time(about 40ms), so we must compromise the algorithm performance when necessary.
7. Test our implementation in Ms. Pac-Man vs. Ghost Platform from most recent CIG competition.

125% Plan
8. Combine Best-Path algorithm with computational intelligence method. Utilize the former in common cases and latter in uncertain and hard-predictable circumstances. Compare its performance with using each method separately.
9. Any large scale computational intelligence paradigm is faced with the challenge of how to reduce time complexity. We want to explore parallel genetic algorithm or parallel neural network based on Hadoop MapReduce. This may not be applicable in developing Ms. Pac-Man agent, but is a very practical topic to get our hands on.

References

[1] R. T. Hiroshi Matsumoto, Chota Tokuyama, “Ice pambush 2” in http://cswww.essex.ac.uk/staff/sml/pacman/cec2009/ICEPambush2.pdf, 2008.

[2] S. M. Lucas, “Evolving a neural network location evaluator to play ms. pac-man,” IEEE Symposium on Computational Intelligence and Games, pp. 203–210, 2005.

[3] Atif M. Alhejali and Simon M. Lucas, “Evolving Diverse Ms. Pac-Man Playing Agents Using Genetic Programming”, 2010.

[4] J. S. D. Bonet and C. P. Stauffer, “Learning to play pac-man using incremental reinforcement learning,” Proceedings of the Congress on Evolutionary Computation, 1999.

[5] Ms Pac-Man vs. Ghosts League, http://www.pacman-vs-ghosts.net