Progress Report

Drew Wicke

March 26, 2012

The multi-agent learning simulator (MALSIM) is a simulator and benchmarking tool for multi-agent learning algorithms written in Java. The goal of MALSIM is to provide multi-agent learning researchers a common framework in which to compare and test various MAL algorithms. MALSIM will help to eliminate some of the empirical testing issues that are prevalent in the multi-agent learning community.

MALSIM is written in Java using the Netbeans 7.0.1 IDE. The following packages are incorporated to add specific features:

- Gamut - Game theory game generator [5]
- MPJ - MPI for Java [4]
- XStream - XML serialization of Java objects [3]
- JChart2D - real time graphing library [1, 2]

Since the Detailed Proposal, MALSIM now allows the user to interact with the GUI to:

- Start a multi-process *Batch* that uses MPJ
- Pause, resume and terminate *Tournaments safeley*
- View state of queued *Games* for a *Tournament*
- View state of the *Tournaments* in the *Batch*
- Set number of threads/processes to use in *Tournament*
- View a graph showing an agent's average reward per time step for a game
- Choose the Adaptive Dynamics Learner (ADL) algorithm as an agent

During the rest of the course I intend to add unit tests for the major classes and add minor features, such as saving the collected data to a file and improve the GUI. I also intend to research more multi-agent learning algorithms and if time permits implement and compare them to ADL using MALSIM. The source code is available online at https://github.com/dwicke/.

References

[1] Jchart2d. *http://jchart2d.sourceforge.net/*.

[2] Livegraph. *http://www.live-graph.org/*.

[3] Xstream. *http://xstream.codehaus.org/*.

[4] M. Bornemann, R. van Nieuwpoort, and T. Kielmann. Mpj/ibis: a flexible and efficient message passing platform for java. *Recent Advances in Parallel Virtual Machine and Message Passing Interface*, pages 217–224, 2005.

[5] E. Nudelman, J. Wortman, Y. Shoham, and K. Leyton-Brown. Run the gamut: A comprehensive approach to evaluating game-theoretic algorithms. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 880–887. IEEE Computer Society, 2004.