

# Artificial Intelligence

## REINFORCEMENT LEARNING IN AI GAME PLAY USING MicroRTS

BY CRAIG NELSON & ADAM SWIFT



DREXEL UNIVERSITY  
College of

Computing & Informatics

*Research Experience for  
Teachers in Engineering and  
Computer Science:*

*Machine Learning, Big Data and CS Principles*

**Description:**

AI is prevalent throughout society including, but not limited to, search engine algorithms, selection of cell towers to transmit phone calls, delivery patterns, interactive websites, GPS systems, etc. AI, artificial intelligence, is the foundation for a vast majority of technological products. Are you happy with where technology is today? Do you desire greater improvement, more speed, and/or specialization? As we demand more from our products/devices it is critical to understand what AI is, how it works, and ways in which we can improve it. The purpose of this lesson will be to introduce you to AI, explaining what it is and how it can be used to enhance technological products/devices. We will focus this lesson on the improvements of AI over the years and the direction it is heading. We will also delve further into the role of AI in video games, specifically microRTS, but it is important to keep in mind the vast services and products that rely on AI.

MicroRTS is a real time game in which both players act simultaneously with no time to stop and contemplate their next move, unlike a game like chess, in their attempt to win the game. It can be played by a user against the computer or the computer can play against itself using different/selected AI strategies. The objective in using microRTS in this lesson is to understand the basic game play and features of microRTS as well as evaluate the role of AI in the game's development. Students will understand the role of different algorithms in controlling built in AI's. As a group, they will discuss/evaluate the limitations of each algorithm (ie. An algorithm is a step by step process. In a real time video game can you prepare for constant changes and new scenarios ahead of time?).

## Resources:

Source files for microRTS (will be provided)

Game Background - <https://code.google.com/p/microrts/>

microRTS Handout – basic game description and features

Videos: Watson and Deep Blue (you tube)

Online Articles - Wired Magazine

AI Pre Lesson Questionnaire

mircoRTS Game Play Questionnaire

AI Post lesson Questionnaire

## Software

**Eclipse, NetBeans**, or comparable IDE – currently the only tested IDE's are Eclipse and NetBeans. NetBeans is an Integrated Development Environment (IDE) In June 2000, NetBeans was made open source by Sun Microsystems, which remained the project sponsor until January 2010 when Sun Microsystems became a subsidiary of Oracle. The Eclipse IDE is an Integrated Development Environment for Java Programmers. The Eclipse Foundation is a community for individuals and organizations who wish to collaborate on commercially-friendly open source software

Websites

Website: <https://www.eclipse.org>

Website: <https://netbeans.org>

**GIT Client** and **GIT Hub** with a **GIT Hub Account** - A GIT Hub account will be useful to anyone wanting to extend this research and continue contributing to the open source community. GIT is a Version Control System Client and Git Hub is a Version Control System Server. See the Installation and configuration document for further details

Websites

Website: <https://git-scm.com>

Website: <https://github.com>

TortoiseSVN - TortoiseSVN is an Apache Subversion (SVN) client, implemented as a Windows shell extension. See the installation and configuration document for further details.

Website: <http://tortoisesvn.net/>



Minimalist Real-Time Strategy Game designed for AI research

microRTS is a small implementation of an RTS game, designed by Santiago Ontañón in 2013 to perform AI research

Website: <https://code.google.com/p/microrts/>

The lesson contains an AI Pre-Questionnaire and an AI Post-Questionnaire. The purpose of the two questionnaires is to collect data on the success of using MicroRTS as engagement to increase the learning outcomes of disseminating knowledge about AI and Real Time Strategy Game Development. Below is a sample each of the quizzes.

## **PART 1**

### **AI Pre-Questionnaire**

Please take a minute to answer the following questions.

1. What is an algorithm?
2. Can you create an algorithm without knowing the end goal or scenario in which you are acting? Explain.  
(Consider – is the activity an isolated activity? **Example** Trying to make a basketball shot on your own vs. trying to make a shot in a game with four other teammates and a defense on the floor)
3. What is artificial intelligence?
4. What technological devices and products utilize artificial intelligence?
5. Are computers smarter than humans?
6. Which does better processing data?
7. Which does better making decisions in real time?

The first use of the term Artificial Intelligence: The founding father of the term artificial intelligence will be studied. Some of his accomplishments will be reviewed

“We propose that a 2 month, 10 man study of artificial intelligence be carried out during the summer of 1956 at Dartmouth College in [Hanover, New Hampshire](#). The study is to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it. An attempt will be made to find how to make machines use language, form abstractions and concepts, solve kinds of problems now reserved for humans, and improve themselves.” Computer Scientist - John McCarthy, August 31, 1955

### **Show Videos of More Recent AI's in Game Action**

Students will be guided to review some recent activity in game development involving AI. Some suggestions will be these videos on the Deep Blue Chess Challenge and the Watson Jeopardy Challenge.

Chess: Deep Blue - 1997

<https://www.youtube.com/watch?v=WGATtigzCNA>

Jeopardy: Watson - 2011

<https://www.youtube.com/watch?v=dr7IxQeXr7g>

Students will be separated into groups and provided varying articles on various areas in society where AI is used. Below are a few suggested sources to start from

**Articles** - to be read as a class for discussion or break students up into groups to read and bring back to share with overall class

<http://www.wired.com/2014/06/the-future-of-computer-intelligence-is-everything-but-artificial/>

<http://www.wired.com/2014/06/ai-healthcare/>

<http://www.bostonglobe.com/news/science/2013/10/06/mit-artificial-intelligence-center-backed-federal-grant-learning-from-infant-brain-research/MdPnWBnGv7KA1N3CVssKEO/story.html>

Videos & Articles should be used to spark introductory discussion on Artificial Intelligence. After basic discussions on AI class will be ready to start hand out activities and demo's to further their understanding of AI.

Below are links to a PDF file on an AI algorithm for a TIC TAC TOE game and a Video on AI and TIC TAC TOE. Student are given an assignment to read the article and view the video during the lesson.

**Classroom Demo 1** - Tic Tac Toe – see link for directions

<http://www.cs4fn.org/teachers/activities/intelligentpaper/intelligentpaper.pdf>

**Student Activity 1** – TIC TAC TOE – Creating an AI for when you go second

<http://www.cs4fn.org/programming/noughtscrosses/>

After completing Activity 1, the class will discuss thoughts about the experiment as well as their answers to the Pre AI Questionnaire.

**PART 2** – Students will be given to study algorithm development in terms of AI game development.

**Describe systems/games we have discussed so far:**

**Watson Video** – filter through all gathered data and map connections.

**Deep Blue** – Let opponent make move then re-evaluate best alternative before making its own move.

**Tic-Tac-Toe** – Strategy created prior to playing game and before knowing what your opponent will do. Does not re-evaluate strategy after each subsequent move by one's opponent.

Student will briefly discuss some advance concepts in AI game development. What about games or scenarios where multiple players are acting simultaneously and as a programmer you are unaware of what your opponent might do at any given moment? How can we improve our AI based on successes and failures?

The teacher will introduce students to microRTS. Students will be provided a handout outlining the basic features and instructions for playing microRTS. Within the context of playing microRTS, learning will be focused on how microRTS implements artificial intelligence. While learning about the game play and interacting with and against the different types of AI students will answer the In Game Evaluation Questions below.



## **In Game Evaluation**

Create a chart identifying/describing the different units on the plating area, their actions, and the role that they play in the game ?

Describe an algorithm (in words) that would explain how a Unit operates.

Describe the behavior of a Real Time Strategy Game

During the game play session, students will be guided to comment specific method calls and uncomment other method calls. During this point, students will be guided to observe Java code in the main method of the MicroRTS application. Through this, students can be shown working example of modular development, polymorphism, inheritance and composition in a professionally developed artificial intelligence game.

The commenting and uncommenting of code in the application will permit students to toggle on and toggle off different artificial intelligent strategies that have been hard coded into the MicroRTS game. This will permit the willing student to play a game against several different AI Strategies. They will be able to record which AI's agents that they were most successful and least successful against. Afterwards, they should be in a better position to try to explain the different algorithms controlling each of the AI Strategies.

## Part 3 Q Learning

### Classroom Demo 2 – simplistic demonstration of machine learning

<http://www.cs4fn.org/teachers/activities/sweetcomputer/sweetcomputer.pdf>

On its most basic level Q-Learning relies on two sets of parameters: an agent's state and the agent's corresponding actions. By simulating different agent states and subsequent actions over a large subset, a Q-Learning algorithm will gather data on the results from different actions by an actor in the same state. Each time the game is run, a state is created and initialized with values including the success or failure of that state. The state is stored in a table representing the history of successes and failures reached after each state. When a game is replayed and the AI agent is faced with a state that matches a state played in the past, and if the original action was successful, that is the action that will be played by the AI agent. If the action was not successful, a new arbitrary action will be played and new data is gathered on the success or failure of that action. This will continue until a reliable amount of runs provides the optimal action for each given state. Positive actions will be assigned a higher likelihood of being repeated while negative actions will see a decrease in their likelihood of being repeated. The correct action might not be made the first time through, but, over time the AI will have an ample data set to determine the best move given different situations. In this respect, the computer is learning. Obviously, the greater number of actions, the more time this learning will require. The fewer the choices the quicker this learning can be developed

(show clip from WarGames – end of video with “Joshua” learning how to play tic-tac-toe and realizing there is no winner in the end).

### AI in Gaming Post Evaluation

1. What are the limitations when using AI in real time game play (ie. microRTS)?
2. How does Q-Learning improve this limitation?
3. Does Q-Learning solve the issue of real time change in its entirety?
4. Are computers smarter than humans? Which does better processing data? Decision making in a real time? (**DID YOUR ANSWERS CHANGE**)
5. **Real World Application** – Provide a scenario in which AI could be thrown off by an unexpected interruption. How could Q-Learning improve the AI's ability to handle this unexpected interruption?