

The Game of Tic-Tac-Toe

Koen V. Hindriks

September 8, 2013

1 Introduction

The game of Tic-Tac-Toe, also known as Noughts (Os) and crosses (Xs), is a game between two players, the X player and the O player.

2 Playing Tic-Tac-Toe: An Example Agent Player

GOAL is distributed with one example agent player for the Tic-Tac-Toe game. This agent and its corresponding MAS file can be found in the `TicTacToe` folder within the `GOALagents` folder that is located in the directory where GOAL has been installed. If you did not start the IDE of GOAL yet, do so now, and locate and load the MAS file `tictactoe.mas2g`. Select the MAS file and launch it by pressing the run button in the IDE of GOAL. The game board depicted in Figure 1 should now be displayed.

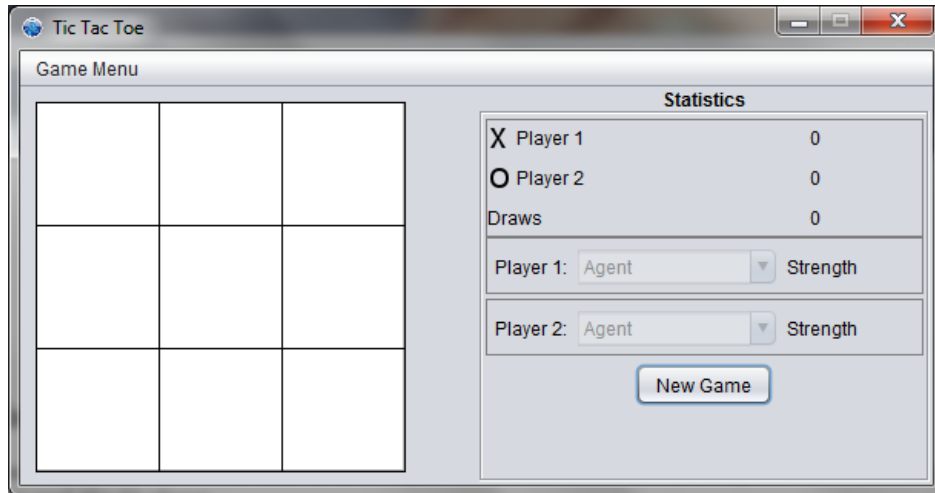


Figure 1: The Tic-Tac-Toe Game Board

2.1 Human Play

If a player is a human, that player can interact with the game board using the mouse. The player options in the GUI of the game board can be used to select that a player is a human player. Indicating that a player is human can also be done by means of the initialization parameters in a MAS file (See Section 4.1). Occupying a position on the board is as simple as clicking on that position.

Note that you can also start the game in a “stand alone” mode and you do not need to start the IDE of GOAL to play a game; see Section 4.

3 Actions and Percepts in the Tic-Tac-Toe Game

The Tic-Tac-Toe game only supports the action to occupy a position on the game board. The occupy action can be performed if the position on the board has not yet been occupied in a previous move and it is the player's turn. A position occupied by a player is marked with that player's symbol, i.e., an 'X' for the X player and an 'O' for the O player.

occupy(Position): The argument **Position** can be 1-9. The positions 1, 3, 7, and 9 denote corners. Position 5 denotes the center of the game board.

The game board configuration is fully observable during the game. This means that both players can see exactly which positions have and have not yet been occupied and, if a position has been occupied, who has occupied it. For each of the nine positions on the board the following percept is received each round:

pos(Position, Player): The **Position** argument corresponds to one of the 1-9 positions. The **player** refers to the x or o player.

At the end of the game an agent player also receives information about who won the game.

- **winner(Player):** The **Player** argument indicates who has won, either the x or o player. This percept only is received if there was a winner at the end of the game.
- **draw:** This percept is received at the end of the game if there was no winner.

4 Environment Package Description

We briefly summarize here some relevant information about the environment:

Name The environment file is named `tictactoe.jar` and can be found in the `environments` folder that is located in the directory where GOAL has been installed.

Pause The environment supports pausing.

Reset The environment supports reset.

Stand Alone The environment supports stand alone running of a Tic-Tac-Toe game. [**TODO:** How?]

4.1 Initialization

A game can be initialized by providing the types of players that will be playing the game. A player can be a human player, an agent player, or an AI player. A human player can play the game by using the game board and selecting a position the player wants to occupy by clicking on it with a mouse pointer. An agent player can occupy a position by performing the occupy action (see Section 3) and see the game board via percepts it receives from the game. An AI player is part of the environment itself. [**TODO:** can we set strenght of AI player?]

The following initialization parameters can be set in the `environment` section of a MAS file:

- **xplayer:** can be set to values "agent", "ai", "human". Selects the type of X player. See also explanation above. *Example:* `xplayer = "agent"`.
- **oplayer:** can be set to values "agent", "ai", "human". Selects the type of O player. See also explanation above. *Example:* `oplayer = "human"`.
- **gui:** can be set to "true" or "false". If set to true, the game board is displayed; otherwise it is not. *Example:* `gui = "false"`.