Game Theory Toolkit User Guide

Pranav Garg

January 24, 2024

This Guide will contain all valid cases of the parameters mentioned in the README.

About computational runtime, I will only consider examples that can be run reasonably fast on most machines. As the the scale increases, I will suggest faster algorithms with weaker guarantees. I am only considering games with a terminal state in this guide.

1   2 Player,
   Normal Form,
   Perfect Information,
   Zero Sum,
   Deterministic,
   Model based,
   Finite Discrete State Space,
   Finite Discrete Action Space

Normal form zero sum games are usually defined by one payoff matrix M such that player 2 pays player 1 the value in the matrix. Normal form games are perfect information by definition as the game lasts only one move, there is only one state. The aspect of Zero sum significantly reduces the complexity of finding an optimal strategy as we only have one payoff matrix to deal with. When it is deterministic, it means each matrix element can only be one value. For this case, even the stochastic case reduces to the deterministic one in expectation. Model-based simply means we have the payoff matrix available to learn a strategy. The state space is always finite discrete for normal form games as there is only one state. Here, since the action space is finite discrete the matrix is an m by n matrix where m and n are natural numbers.

Environments: The quintessial example of this type of game is Rock Paper Scissors. Other examples can include larger action spaces; weighted and non-symmetric variants of Rock Paper Scissors.

Optimality: Optimality for these games is defined as a Nash Equilibrium strategy. A Nash Equilibrium is vector of mixed strategies, one for each player, such that no player has an incentive to deviate from its mixed strategy given that the other does not deviate.

Algorithms: These games are fully solvable by a Linear Program with m constraints and n variables. LP solvers are reliable generally for $m, n < 10^8$. Larger values will be analogous to the Infinite Discrete action space case.

I have also included Fictitious Play [Robinson 1951] as an algorithm that works here. Although the Linear Program is a superior algorithm for this case, Fictitious Play's iterative process of calculating Best Response and updating strategies will be useful for other cases.