Paper title: "Robust Game Play Against Unknown Opponents"
Authors: Nathan Sturdevant, Michael Bowling
Subject: A modified *max*^{n} algorithm suitable for multiplayer games requiring minimal opponent modeling.

So far we have studied search algorithms in two player perfect information games. Minimax is provably the most effective algorithm in such cases; however, implicit in its structure is an opponent model that seeks to minimize the gains of the agent. Minimax generalizes to *max*^{n} in multiplayer games; there the implicit assumption is that all agents will seek to maximize their own gains at any particular moment. This is a sensible assumption in a two player game; however it does not generalize well to multiplayer games, where there can be many winning strategies and optimal play depends on accurate opponent modeling and forecasting.

Using 3-player perfect-information Spades (i.e. open-hand) as a testbed, this paper demonstrates 1) poor opponent modeling leads to detrimental performance, even when our agent adopts an optimal strategy; 2) by altering *max*^{n} to return sets of *max*^{n} values, a new algorithm "*soft-max*^{n} " can improve win performance without relying on accurate opponent models; and 3) *soft-max*^{n} can further improve its performance by updating its opponent model with version-space learning.

The authors generate two Spades player models, one "maximum trick" model that seeks to win all tricks, and one "minimum overtrick" model that seeks to stick close to its bet. The minimum overtrick model is the optimal strategy for Spades, but the authors generate data to demonstrate even this optimal strategy can have a low win rate if it makes the wrong assumption of its opponent model. They use this data to support the notion of an algorithm more robust to unknown opponents.

This algorithm is called "*soft-max*^{n} " and instead of back-propagating a single *max*^{n}-tuple up a game tree, it back-propagates sets of *max*^{n}-tuples. From the paper, the algorithm can be defined by the following three steps:

1. At leaf nodes in a game tree, the *max*^{n} set is a single tuple, evaluated by some evaluation function.
2. At internal nodes in a game tree, the *max*^{n} set is the union of all child sets that are not "strictly dominated" with respect to the acting player at that node.
3. At the root of the tree, the acting player can use any decision rule to select the best non-dominated set.

There is a notion of "domination" between *max*^{n} sets that is defined as follows:

- "Strict" dominance between sets occurs iff $\forall v_1 \in s_1 \, \forall v_2 \in s_2 \quad v_1[i] > v_2[i]$.
- "Weak" dominance between sets occurs iff
  - $\forall v_1 \in s_1 \, \forall v_2 \in s_2 \quad v_1[i] \geq v_2[i]$
  - $\exists v_1 \in s_1 \, \exists v_2 \in s_2 \quad v_1[i] > v_2[i]$
- Otherwise one set cannot be said to dominate another.

soft-max^n does not allow "strictly dominated" sets to back-propagate through the game tree.

There is data that demonstrates performance improvements using *soft-max*^{n} despite not having any specific opponent modeling. There is also some data presented that shows some performance improvements could be made by accurate opponent modeling, defined mathematically as the edge set of a digraph with nodes as *max*^{n}-tuples. The opponent model is effectively partially ordered and this ordering is refined with observations made throughout the course of play. For *max*^{n} sets,

- s_1 strictly dominates s_2 under opponent model O iff $\forall u_1 \in s_1 \forall u_2 \in s_2 (u_1, u_2) \in O$
- s_1 weakly dominates s_2 under opponent model O iff
  - $\forall u_1 \in s_1 \, \forall u_2 \in s_2 \quad (u_2, u_1) \notin O$
  - $\exists u_1 \in s_1 \, \forall u_2 \in s_2 \quad (u_1, u_2) \in O$

The data demonstrating the success of this approach is a little sparse. The authors claim that they make successful opponent inferences about every sixth hand, and that this results in 4-8% win rate improvement. However; this approach was not able to rule out various opponent models, both the "max trick" and "min overtrick" touched on earlier, as well as a third, more aggressive model. The success rates for the three models were 106, 103, and 51 out of 900 attempts – about 5-11%.

soft-max^n thus presents itself as a compelling start to a general multiplayer algorithm, but also needs much work to apply reliably in complex games. There is strength in its performance using a generic opponent model, rather than relying on some simplifying assumptions like the paranoia implementation (or regular *max*^{n} itself). Opponent modeling could be improved with an inspired set of heuristics and/or a more robust training set. *soft-max*^{n} is also conducive to applying Monte Carlo methods to move from perfect-information games to imperfect-information games. This will affect the inferences made by *soft-max*^{n} about its opponents, and is more in need of further work.