General Heuristic for Tic-Tac-Toe-Like Games

Martha Mercaldi, Katarzyna Wilamowska

November 10, 2004

Abstract: *We define and explore a general class of two-player board games. Members of this class are defined by the dimensionality and size of the board, the allowable moves and the length of a a run required to win. We evaluate and compare three heuristics which can be applied to any member of this class. We have implemented a general game-playing framework, referred to as Connect-Toe which can simulate all games in this class and which can be used for further investigation.*

1 Introduction

We have defined a general class of games to study. This class contains two board games which have already been studied extensively, Tic-Tac-Toe [2] and Connect Four. Researchers have developed specialized heuristics to evaluate incomplete boards for both of these games [7]. We believe that a general heuristic could be developed and applied to all games in this more general class of board games.

In games, where the search tree is potentially very large (both in depth and branching factor) a heuristic which can evaluate incomplete game states is useful. With such a heuristic a Minimax search algorith can be both targeted (to search directions that show the best potential first) and shortened (recursive search can return before reaching a leaf of the search tree).

We define three very simple heuristics, *Optimistic*, *Pessimistic* and *Undecided*, described in Section 3 and evaluate and compare their efficacy on a broad class of board games.

2 Class of Board Games

The class of games which we study for this project contains board games which are played by two players alternately dropping their own tokens into the board. When a token is dropped into some slot, it falls until it hits either another token or the bottom of the board. The first player to create a designated length string of his own tokens wins.

The games in this class vary according to several attributes:

dimensionality Dimensionality of the game board

size Size of board in each dimension (greater than or equal to 1)

winning run length Length of run (in any direction) required to win

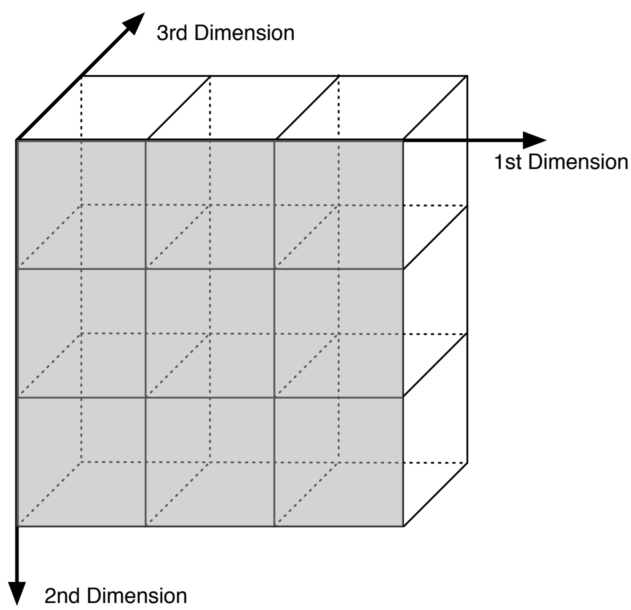

Figure 1: Classic Tic Tac Toe as a class member.

open faces An open face is a face through which pieces can be dropped into the board. Once inserted a piece will fall either to the bottom of the board, or until it is blocked by another piece already in the board. For example, in Connect Four, the grid is 7x6x1 with only one 7x1 face, 1, open. In Tic-Tac-Toe the grid is 3x3x1 with one 3x3 face, 2, open. See Figures 1 and 2 for illustrations. The shaded board faces are the open faces. Each face id is a signed integer. The magnitude indicates the dimension along which a piece will fall, and the sign indicates the direction, *upwards* or *downwards*. This attribute restricts the allowable moves.

We first focus our heuristic comparison on Tic Tac Toe and Connect Four. Because this class of games is defined by so many attributes, it makes sense to first study them one at a time, instead of taking them all on at once. After first looking at the special cases of Tic Tac Toe 5 and Connect Four 6, we focus our heuristic comparison on varying three dimensional board sizes 7. For the purposes of this paper, we have combined these two variants into testing the results from varying sizes of cubes. In Section 8 other potentially interesting and relevatory studies are discussed.

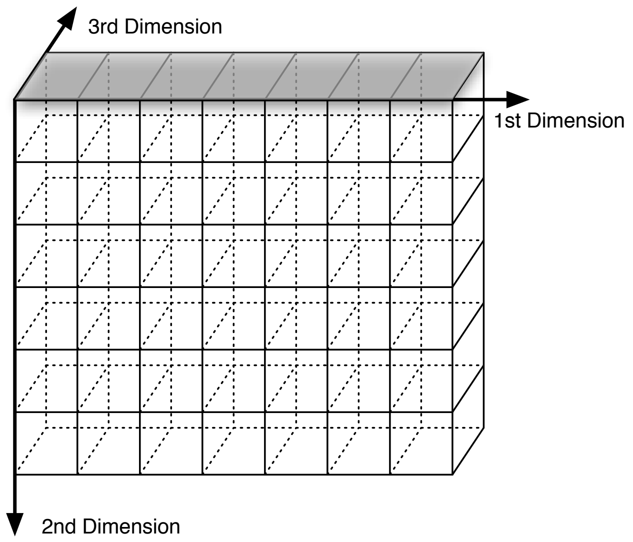

Figure 2: Connect Four as a class member.

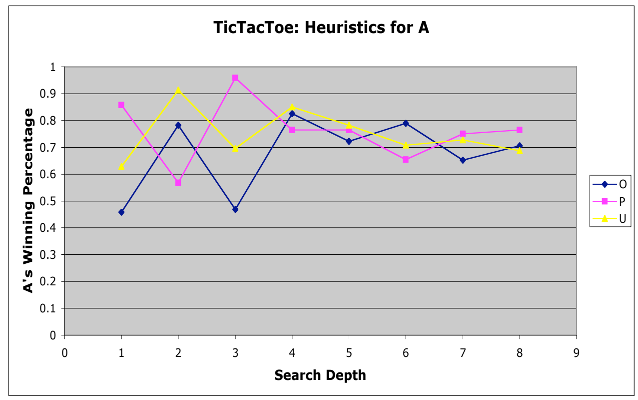

Figure 3: Player A's winning percentage at Tic Tac Toe broken down by heuristic.

3 Heuristics

The heuristics used in this study were designed to be simple. When presented with a complete board, determined by either the board being full, or a win for one of the players, all of the heuristics returned a +1 for a win, 0 for a draw and -1 for a loss. The variation between *Optimistic*, *Pessimistic* and *Undecided* heuristics developed from the game having to resolve the value of an unfinished/undecidable board resulting from a depth control added to decrease search time. *Optimistic*, when faced with an undecidable board, would choose to mark it as a possible winning move, +1. *Pessimistic* would choose a loss, -1, and *Undecided* would choose a draw, 0.

4 Methodology

For this study we implemented a general representation of all board games in our class. With this is a general Minimax search algorithm which can use each of the three heuristics. Our framework also supports a *random* game player, which makes all of its moves at random. For each game configuration we evaluate we simulated 10 games. A bigger sample set would have been ideal, but time concerns restricted us to just 10 runs of each game.

5 Special Case: Tic Tac Toe

First we examine the case of Tic Tac Toe. As we seek to evaluate the efficacy of our three simple heuristics, we look at how they perform as the search depth varies. We expect to see interesting trends since the search depth determines how early the heuristic is applied. The farther from the leaves the heuristic is applied the more of the search it influences. Figure 3 shows how the winning percentages of each heuristic are effected by search depth.

We see that relatively large (though inconsistent) differences between the heuristics with small search depths dimish as the search deepens. We explain this trend in the following way. When a board is empty, every possible game outcome is possible. As moves are made and the board fills, the set of possible outcomes shrinks. If the heuristic is applied when this set is small, it has less opportunity to affeect the game outcome.

The early part of this graph is difficult to decipher. There are not any clear trends. This could be due to our relatively small sample size, and that with more experiments the variation would settle and a more clear trend would emerge. It is also possible that looking at the data in more detail might reveal more. In Figure 4 we show the winning percentage for each of the heuristics when playing against each type of opponent. The 'R' opponent is a *Random* opponent.

Most noticable in this data is the "sweet spot" that the *Pessimistic* heuristic seems to have with searches 3 deep. However, we hesitate to trust the results. As shown in Figure 5 as the search deepens more and more games reach draws. Because the winning percentage is calculated as the number of games player A wins divided by the number of games either player won, the sample set is increasingly small. Starting with an already small sample set, the shrinking due to elimination of increasing numbers of draws only exacerbates the problem.

6 Special Case: Connect Four

Let us now examine our second special case, Connect Four. There the game player has a larger grid to contend with (6x7)

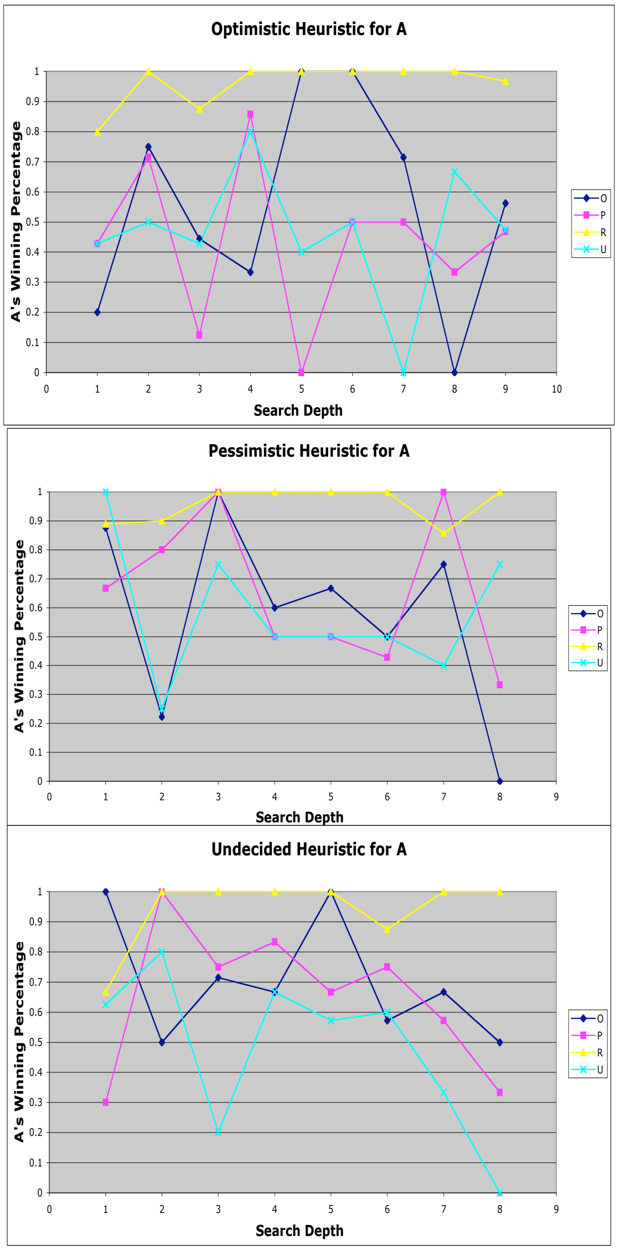

Figure 4: Breakdown of winning percentage at Tic Tac Toe for each strategy, by opponent.

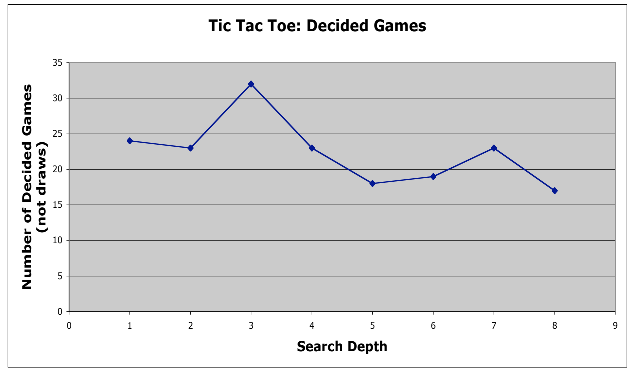

Figure 5: Number of Tic Tac Toe games that do not end in draws, by search depth.

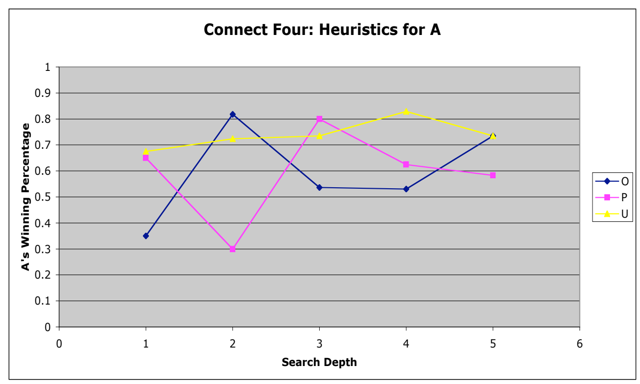

Figure 6: Player A's Connect Four winning percentage broken down by heuristic

yet there are fewer legal moves (seven compared to nine for Tic Tac Toe).

The experiments from 5 were repeated for the Connect Four. Just as before, we expected to see interesting trends dependent on the search depth. Figure 6 shows how the winning percentages of each heuristic is effected by search depth.

Where with Tic Tac Toe we had little evidence for one heuristic outperforming the others, in the Connect Four case, we are able to find a more likely winner. The best performance is given by *Undecided* with an average 74% winning percentage, followed by *Optimistic* with an average of 60% and *Pessimistic* at 58%. As further support, we can see in Figure 7 that *Undecided* tends to have at least 5% fewer losses than any other heuristic.

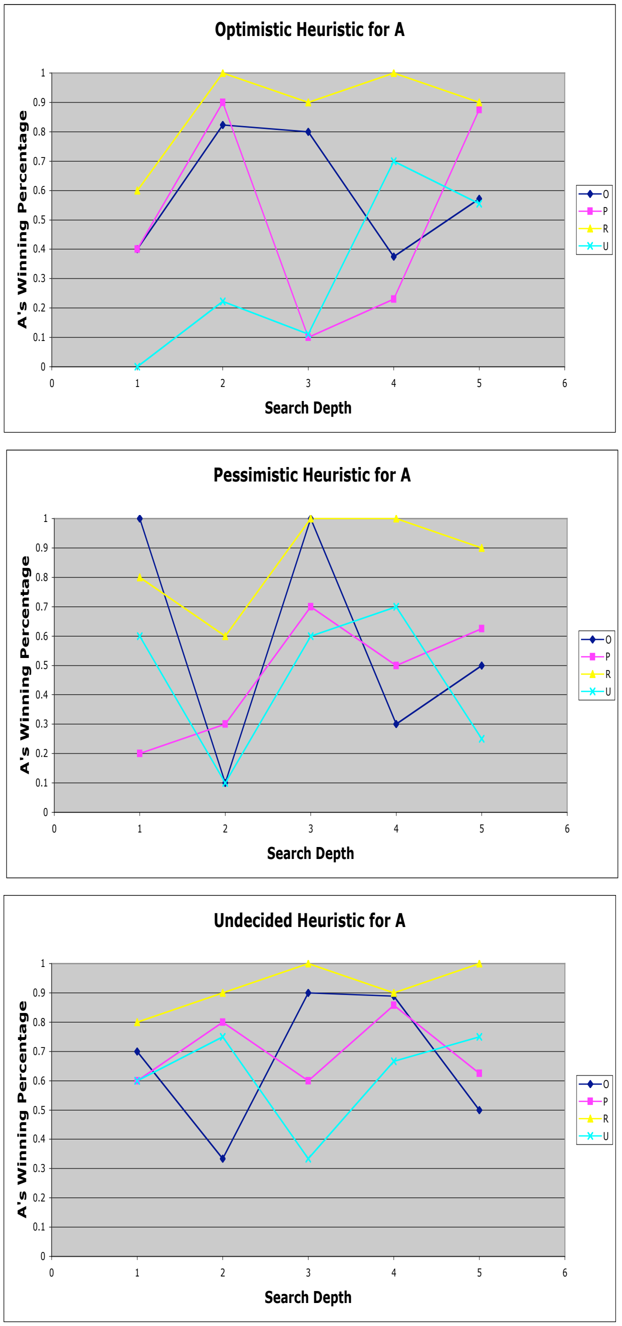

Figure 7: Breakdown of winning percentage for Connect Four for each strategy

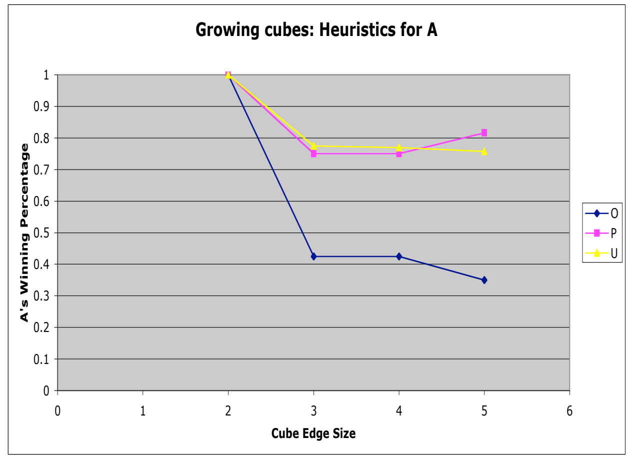

Figure 8: Player A's winning percentage at Cubic Tic Tac Toe broken down by heuristic, for increasing cube sizes.

7 Increasing Cube Sizes

As another test of our heuristics abilities, we experimented with 3D sized cubes while varying the size of the cube and matching the winning run length to the size of a cube side. We expect to see the diference of the cube size not to cause any change in performance of our heuristics. Although the search tree is much larger for each increasing size of cube, due to the increasing size of run length necessary to win, the problem is basically the same. As such, this experiment, in part checks whether our heuristics are scalable. Figure 8 shows how the winning precentages of each heuristic are effected by cube size.

We see first of all the inconsistency that all heuristics have a 100% winning ratio when applied to cubes of size 2x2x2. Such a result is expected, due to the size 2 3D cube being a generic case, since no matter what the opponent's next move may be, any next move that our player makes, forces an instant win for our player.

As we examine the sizes of cubes greater than 2, the hypothesis of scalability is realized. With slight variation, most likely a result of using a small sample, the winning percentage of any of our heuristics remains constant through the increases in cube size. We can observe that *Optimistic* has not performed nearly as well as either *Pessimistic* or *Undecided*. Additionally, we have further proof of *Undecided* outperforming the other two of our heuristics. Notice that the variation of *Undecidable* is less than 2%, Figure 9.

Let us now examine Figure 8. *Undecided* outperforms the other heuristics by losing the least percentage of games to its opponents, including itself, with *Pessimistic* as a close second. *Optimistic* plays poorly and barely outperforms *random* play.

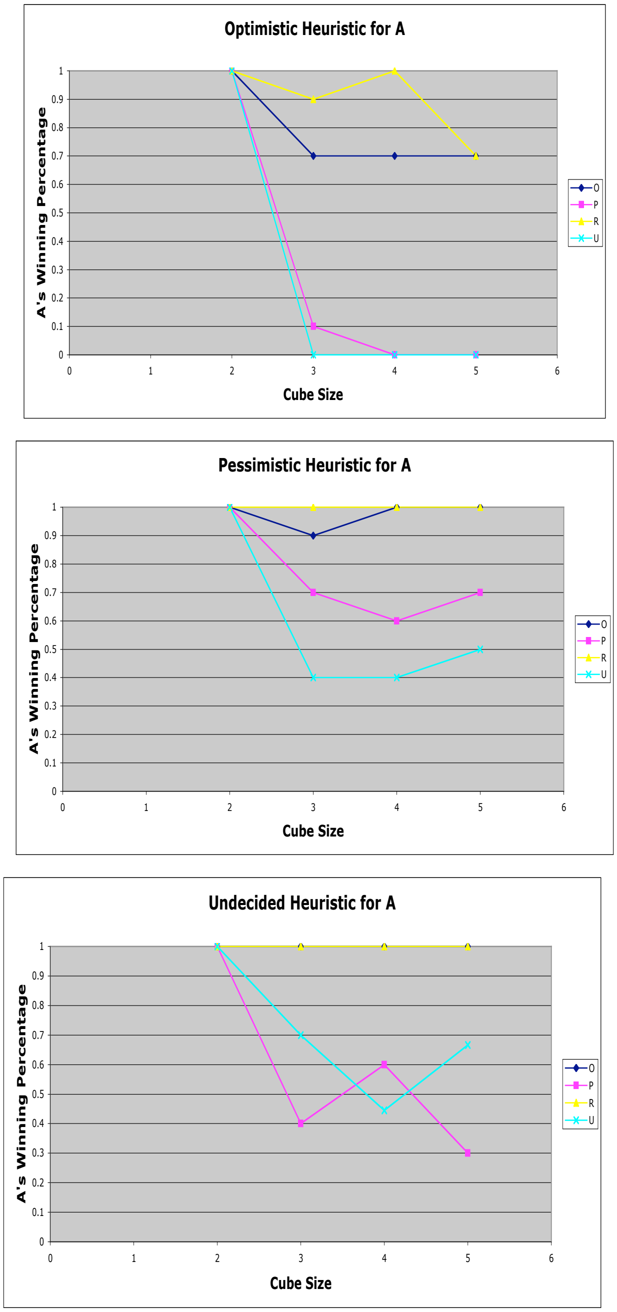

Figure 9: Breakdown of winning percentage at Cubic Tic Tac Toe for each strategy, by opponent, for increasing cube sizes.

8 Future Work

There are many future studies that could be done in this vein. The framework we have implmenented is extremely general, and in this work we have evaluated only a small sliver of the class of games it is capable of representing. As mentioned in Section 2 the class is highly multidimensional. That is to say game members can vary along many dimensions: board dimensionality, board size, the length of a winning run, the number and orientation of open faces. The basic heuristics we describe in this work could be evaluated along one or more of these dimensions.

We also believe that a more intelligent heuristic could be developed to more precisely evaluate an incomplete board state. Because the rules of the game are fairly regular, a player could find existing runs and depending on how long they are and if it is possible to extend them to winning length, make a more precise valuation of an incomplete board.

9 Conclusion

The Connect-Toe program designed to facilitate our experiments on the Tic-Tac-Toe and Connect Four class games has been built to play all games designed in Section 2. We examined how three simple heuristics perform against one another and against random play. With limited feedback, due to time constraints, we were able to discern that the *Undecided* heuristic tended to outperform the other heuristics. Our framework will support more thourough exploration both of the designated class of board games and the space of possible heuristics.

References

[1] Thuente, David J., and Rhys Price Jones. ”Beyond minimaxing for games with draws.” Proceedings of the 19th annual conference on Computer Science, April 1991: 551-557.

[2] Carroll, Maureen T., Steven T. Dougherty. ”Tic-Tac-Toe on a Finite Plane.” 15 Oct 2004 $<$ http://academic.scranton.edu/faculty/carrollm1/tictac.pdf $>$.

[3] Dyer, C.R. ”Game Playing (Chapter 6).” CS 540 Lecture Notes. 22 Oct 2004. University of Wisconsin, Madison, Wisconsin $<$ http://www.cs.wisc.edu/ dyer/cs540/notes/games.html $>$.

[4] Minimax and Alpha-Beta Template. 22 Oct 2004 $<$ http://www.cs.caltech.edu/ petrovic/games/archex/othellodir/node2.html $>$.

[5] ”Mnk-games” Wikipedia, the free encyclopedia 17 Oct 2004.

[6] ”MTD(f): A Minimax Algorithm faster than NegaScout.” Aske Plaat: MTD(f), a new chess algorithm. 15 Oct 2004 < http://www.cs.vu.nl/ aske/mtdf.html >.

[7] Pilgrim, Robert A. ”TIC-TAC-TOE: Introducing Expert Systems to Middle School Students.” SIGCSE Proceedings (1995):340-344.

[8] ”Topics: Minimax.” CS61B, Spring 2003.15 Apr 2003. 17 Oct 2004 < http://www.cs.berkeley.edu/ kamil/sp03/minimax.pdf >.

Data Tables

Tic Tac Toe

Search depth	O	P	U
1	0.458333333	0.857142857	0.628571429
2	0.782608696	0.567567568	0.913043478
3	0.46875	0.958333333	0.695652174
4	0.826086957	0.764705882	0.85
5	0.722222222	0.764705882	0.782608696
6	0.789473684	0.653846154	0.708333333
7	0.652173913	0.75	0.727272727
8	0.705882353	0.764705882	0.6875

Table 1: Source data for Figure 3.

Search depth	O	P	R	U
1	0.2	0.428571429	0.8	0.428571429
2	0.75	0.714285714	1	0.5
3	0.444444444	0.125	0.875	0.428571429
4	0.333333333	0.857142857	1	0.8
5	1	0	1	0.4
6	1	0.5	1	0.5
7	0.714285714	0.5	1	0
8	#DIV/0!	0.333333333	1	0.666666667

Table 2: Source data for Optimistic heuristic in Figure 4

Search depth	O	P	R	U
1	0.875	0.666666667	0.888888889	1
2	0.222222222	0.8	0.9	0.25
3	1	1	1	0.75
4	0.6	0.5	1	0.5
5	0.666666667	0.5	1	0.5
6	0.5	0.428571429	1	0.5
7	0.75	1	0.857142857	0.4
8	0	0.333333333	1	0.75

Table 3: Source data for Pessimistic heuristic in Figure 4

Search depth	O	P	R	U
1	1	0.3	0.666666667	0.625
2	0.5	1	1	0.8
3	0.714285714	0.75	1	0.2
4	0.666666667	0.833333333	1	0.666666667
5	1	0.666666667	1	0.571428571
6	0.571428571	0.75	0.875	0.6
7	0.666666667	0.571428571	1	0.333333333
8	0.5	0.333333333	1	0

Table 4: Source data for Undecided heuristic in Figure 4

Search depth	Decided Games
1	94
2	83
3	79
4	60
5	58
6	69
7	65
8	50

Table 5: Source data for Undecided heuristic in Figure 4

Connect Four

Search depth	O	P	U
1	0.35	0.65	0.675
2	0.818181818	0.3	0.724137931
3	0.536231884	0.8	0.735294118
4	0.530612245	0.625	0.828571429
5	0.735294118	0.583333333	0.735294118

Table 6: Source data for Figure 6.

Search depth	O	P	R	U
1	0.4	0.4	0.6	0
2	0.823529412	0.9	1	0.222222222
3	0.8	0.1	0.9	0.111111111
4	0.375	0.230769231	1	0.7
5	0.571428571	0.875	0.9	0.555555556

Table 7: Source data for Optimistic heuristic in Figure 7

Search depth	O	P	R	U
1	1	0.2	0.8	0.6
2	0.1	0.3	0.6	0.1
3	1	0.7	1	0.6
4	0.3	0.5	1	0.7
5	0.5	0.625	0.9	0.25

Table 8: Source data for Pessimistic heuristic in Figure 7

Search depth	O	P	R	U
1	0.7	0.6	0.8	0.6
2	0.333333333	0.8	0.9	0.75
3	0.9	0.6	1	0.333333333
4	0.888888889	0.857142857	0.9	0.666666667
5	0.5	0.625	1	0.75

Table 9: Source data for Undecided heuristic in Figure 7

Growing Cubes

Search depth	O	P	U
2	1	1	1
3	0.425	0.75	0.775
4	0.425	0.75	0.769230769
5	0.35	0.815789474	0.757575758

Table 10: Source data for Figure 8.

Search depth	O	P	R	U
2	1	1	1	1
3	0.7	0.1	0.9	0
4	0.7	0	1	0
5	0.7	0	0.7	0

Table 11: Source data for Optimistic heuristic in Figure 9

Search depth	O	P	R	U
2	1	1	1	1
3	0.9	0.7	1	0.4
4	1	0.6	1	0.4
5	1	0.7	1	0.5

Table 12: Source data for Pessimistic heuristic in Figure 9

Search depth	O	P	R	U
2	1	1	1	1
3	1	0.4	1	0.7
4	1	0.6	1	0.444444444
5	1	0.3	1	0.666666667

Table 13: Source data for Undecided heuristic in Figure 9