to appear soon in the MSRI text associated with the BIRS 2005 conference on combinatorial games
submitted 4 September 2006, revised 8 March 2007

A puzzling Hex primer*

Ryan B. Hayward

Dept. of Computing Science, Univ. of Alberta, Edmonton, AB, Canada, hayward@cs.ualberta.ca

Abstract. We explain some analytic methods that can be useful in solving Hex puzzles.

1 Introduction

Solving Hex puzzles can be both fun and challenging. In this paper — a puzzling companion to *Hex and Combinatorics* [5] and *Dead Cell Analysis in Hex and the Shannon Game* [2], both written in tribute to Claude Berge — we illustrate some theoretical concepts that can be useful in this regard.

We begin with a quick review of the rules, history, and classic results of Hex. For an in depth treatment of these topics, see [5].

The parallelogram-shaped board consists of an $m \times n$ array of hexagonal cells. The two players, say Black and White, are each assigned a set of coloured stones, say black and white respectively, and two opposing sides of the board, as indicated in our figures by the four stones placed off the board. In alternating turns, each player places a stone on an unoccupied cell. The first player to connect his or her two sides wins.

In the fall of 1942 Piet Hein introduced the game, then called Polygon, to the Copenhagen University student science club *Parenthesis*. Soon after, he penned an article on the game for the newspaper *Politiken* [6, 8, 9]. In 1948 John Nash independently re-invented the game in Princeton [4, 10], and in 1952 he wrote a classified document on it for the Rand Corporation [11]. In 1957 Martin Gardner introduced Hex to a wide audience via his *Mathematical Games* column [3], later reprinted with an addendum as a book chapter [4].

For Hex played on an $m \times n$ board, the game cannot end in a draw (Hein [6], Nash [11]); for $m = n$, there exists a winning strategy for the first player (Hein, Nash [11]; see also [3]); for $m < n$, there exists a winning strategy for the player whose sides are closer together, even if the other player moves first (Gardner/Shannon [4]); for arbitrary Hex positions, determining the winner is PSPACE-complete (Reisch [12]).

To start our discussion, consider Puzzle 1.

* The support of NSERC is gratefully acknowledged.

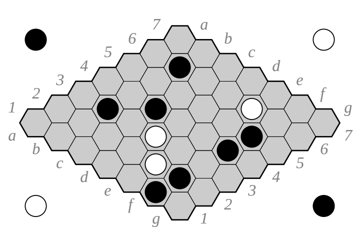
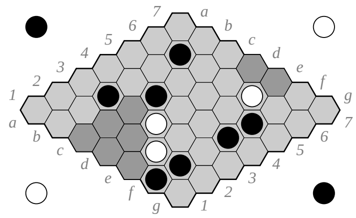


Fig. 1. Puzzle 1, an easy warm-up. White to play and win.



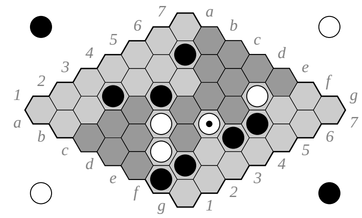


Fig. 2. Two white virtual connections (left) and, after a winning move, a side-to-side white virtual connection (right).

2 Virtual Connections

One useful Hex concept is that of a *virtual connection*, namely a subgame in which one player can establish a connection even if the opponent moves first. In Puzzle 1, as shown in the left diagram of in Fig. 2, the cell set $\{d7, e7\}$ forms a 'bridge' virtual connection between the white stone at $e6$ and the white border on the upper right side. If Black ever plays at one of these two bridge cells, White can then make the connection by playing at the other. Similarly, the white border on the lower left side is virtually connected to the two white stones at $\{d3, e2\}$ via the cell set $\{c1, c2, c3, d1, d2, e1\}$: if Black plays at any of $c1, c2, c3, d1, d2$ White can then play at $e1$, whereas if Black plays at $e1$ White can then play at $c2$ and subsequently make use of the resulting bridge cell sets $\{c1, d1\}$ and $\{c3, d2\}$.

As the left diagram in Fig. 2 suggests, the gap between the two white groups is an obvious place to look for a winning move; the right diagram shows such a move at $e4$. After this move, the new stone is virtually connected by the upper eight marked cells to the upper white side, and by a bridge to $\{d3, e2\}$, and so then by the lower six marked cells to the lower white side, yielding a virtual connection joining the two white sides. Thus $e4$ is a winning move for Puzzle 1.

3 Mustplay regions

Are there any other winning moves for Puzzle 1?

Hex is a game in which it is easy to blunder. Even from obviously won positions, there are usually many moves that lead to quick losses. Since there are no draws in Hex, one way to answer the above question is to first check whether any losing moves can be identified. A *weak connection* is a subgame in which one player can force a connection if allowed to play first. Does the opponent have any side-to-side, and so win-threatening, weak connections?

A virtual connection for a player is *winning* if it connects the player's two sides; a *win-set* is the set of cells of a winning virtual connection. Analogously, a weak connection for a player is *win-threatening* if it connects the player's two sides; a *weak win-set* is the cell set of a win-threatening weak connection. Fig. 3 shows three black weak win-sets for Puzzle 1.

Notice in Fig. 3 that, in order to prevent Black from winning, White's next move must intersect each of Black's weak win-sets, since any weak connection that is not intersected by White's move can be turned into a virtual connection on Black's subsequent move. More generally, at any point in a Hex game,

a move is winning if it intersects all of an opponent's weak win-sets.[1]

A *gamestate* specifies a *boardstate*, or board configuration, and whose turn it is to move. With respect to a player, a gamestate, and a collection of opponent weak win-sets, we call the combined intersection of these weak win-sets the *mustplay region*, since a player 'must play' there or lose the game.

[1] The converse of this statement holds as long as the opponent has at least one weak win-set; then a move is winning if and only if it intersects all of an opponent's weak win-sets. However, if the player about to move is so far ahead in the game that the opponent has no weak win-set, then the intersection of all of the opponent's weak win-sets is the empty set; thus the converse does not hold in such cases.

As shown in Fig. 3, the white mustplay region associated with the three weak connections is $\{e4\}$. We have already seen that $e4$ is a winning move for Puzzle 1; our mustplay analysis tells us that every other move loses. So, to answer the question from the start of this section, there are no other winning moves for Puzzle 1.

4 A Hex solver based on mustplay analysis

There is a straightforward way to solve any Hex puzzle: completely explore the search tree resulting from all possible continuations of the puzzle. This approach is usually impractical, as the number of different gamestates in the search tree is exponential in the number of unoccupied cells. Since solving Hex puzzles is PSPACE-complete, there is unlikely to be any 'fast', namely polynomial time, Hex-solving algorithm. Nonetheless, the search tree can often be pruned using various techniques. In particular, in this section we illustrate an algorithm that uses mustplay regions to prune the search tree.

To demonstrate, consider Puzzle 2. To start, we first look for a white win-set. Finding none, we next look for a black weak win-set. The reader may have already found one, for example using $d4$; Fig. 5 shows three such black weak win-sets. The associated white mustplay region, shown in the last diagram of Fig. 5, is the intersection of the black weak win-sets, namely $\{c4, c5, d4, e3, e4, f2, f4\}$. If White has a winning move, it is at one of these cells.

Fig. 6 shows what happens as, in no particular order, we next consider the moves of this mustplay region. In the first diagram we make the white move at $c5$; by continuing to recursively apply our algorithm, we eventually discover that Black wins the resulting gamestate with the black win-set as shown. At this point we undo the white move, so the black win-set becomes a black weak win-set. We next use this black weak win-set to update the white mustplay region; it becomes reduced to $\{c4, d4, e3, e4, f2, f4\}$. In similar fashion, we eventually discover that the next three white moves considered, namely $d4, e3, e4$, also lose for White; the resulting black weak win-sets are shown in Fig. 6. Notice that the last of these weak win-sets does not contain $f4$, so by this point the white mustplay region has been reduced to $\{c4, f2\}$.

Fig. 7 shows what happens as we consider these last two possible moves. The white move at $f2$ loses, but the white move at $c4$ wins. Thus $c4$ is the unique winning move for Puzzle 2.

We have omitted all the details from the recursive calls of this algorithm. We leave as exercises for the reader to verify that the five weak win-sets and the one win-set shown in Figs. 5-7 are correct.[2] As a guide, the reader might find it useful to follow Fig. 12, which gives a version of this algorithm due to Jack van Rijswijck [14].

Another exercise is to solve Puzzle 3, created by Claude Berge. There is more than one solution; running down the upper-left region is straightforward, while breaking through to the upper-right side is more difficult. Try to find a win-set with no unnecessary cells. One such win-set appears in the last section.

[2] The most challenging of these exercises is the last one, namely to show that *c4* wins for White. The strongest next moves for Black include *c3*, *c5*, *c6*, *d3*, and *e2*; respective winning replies for White include *d3*, *e4*, *e5*, *e4*, and *d3*. For other exercises on small boards, see the opening theory link on Jack van Rijswijck's Queenbee webpage [13].

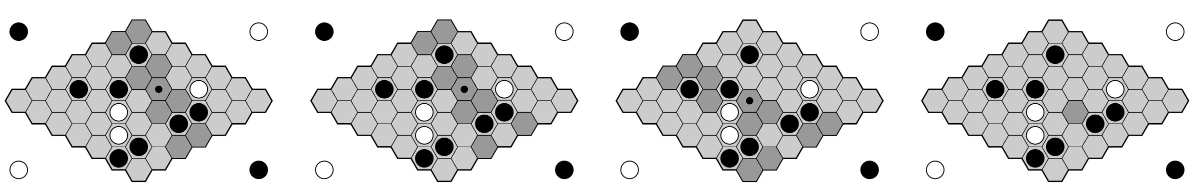


Fig. 3. For Puzzle 1, three black weak win-sets and the resulting white mustplay region. This region has only one cell, so White has only one possible winning move.

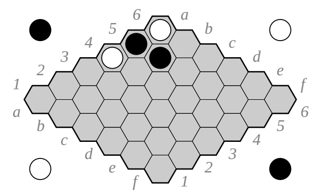


Fig. 4. Puzzle 2, a more challenging problem. White to play and win.

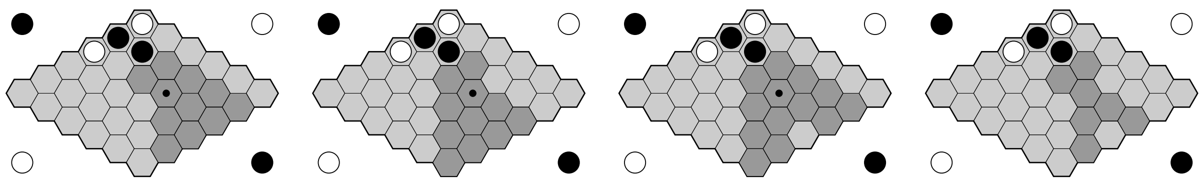


Fig. 5. For Puzzle 2, three black weak win-sets and the resulting white mustplay region.

5 Dead cell analysis

Mustplay analysis yields a set of cells that is critical to a gamestate's outcome. A different form of analysis is based on recognizing individual cells that are irrelevant. We illustrate this 'dead cell analysis' by working through Puzzle 4, created by Piet Hein.

A *completion* of a boardstate is any boardstate obtained by filling all vacant cells of the given boardstate with any combination of black and/or white stones. A cell of a boardstate is *dead* if, for every possible completion, changing the colour of the stone on the given cell does not alter the winner of the completion. A cell is *live* if it is not dead.

For example, the boardstate of Puzzle 4 has 25 vacant cells and so has 2^{25} completions. We leave it to the reader to consider a sample of these completions and verify that in each case, changing the colour of the stone at cell *d1* does not change the winner of the completion. Thus, in this boardstate *d1* is dead.

A gamestate is *undecided* if neither player has yet won. A useful feature of dead cells is that

placing or removing a stone of either colour at a dead cell does not alter the gamestate's winner, and so every undecided gamestate with a winning move has a winning move to a live cell.

Thus, dead cells can be safely pruned from the search tree of a gamestate.

Happily for Hex puzzlers, dead cells can be recognized without having to consider all of a boardstate's completions. The left diagram in Fig. 9 is the *white adjacency graph* for the Puzzle 4 boardstate. The nodes of the graph correspond to the vacant board cells; additionally, two terminal nodes represent the white borders. In the graph, a pair of nodes is joined by an edge if the corresponding cells touch or are joined by connecting white stones.

A path is *induced* if it has no 'shortcuts', namely if the only edges among vertices of the path are between pairs of vertices that are consecutive in the path. The following characterization is an easy consequence of the definition of dead.

A cell with a stone is live if and only if that cell is live after removing that stone. A vacant cell of a boardstate is live if and only if the cell is in some induced terminal-to-terminal path in each of the boardstate's adjacency graphs.

Notice that the white adjacency graph for Puzzle 4 has no induced terminal-to-terminal path that contains *d1*. Thus *d1* is dead in Puzzle 4, as are *a1* and *c1*.

The number of dead cells in a gamestate is often small. However, considering cells that can be 'killed' allows further possible moves to be ignored. In Puzzle 4 it would be pointless for White to play at the *white-vulnerable* cell *e2*, since a Black response at *d3* would kill a white stone at *f2*.

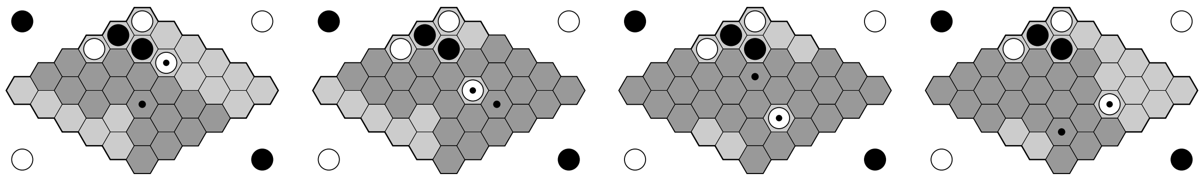


Fig. 6. Black weak win-sets after moves $c5, d4, e3, e4$ respectively.

Fig. 7. A black weak win-set after $f2$, and a white win-set after $c4$. Thus $c4$ wins for White.

This line of reasoning can be continued. Black has a 'second-player kill' strategy for $\{f2, f3\}$: if White ever plays at one of these cells, Black can reply at the other, leaving one cell black and the other dead. We say this set is *black-captured*, since assuming that these cells are already occupied by black stones does not change the theoretical outcome of the game. As an exercise, the reader should verify that $\{f1, e2, f2, f3\}$ is black-captured. It suffices to find, for the subgame played on these cells, a second-player strategy for Black that leaves every stone black or dead.

The notion of *dominated* is analogous to the notion of captured. In Puzzle 4 $\{a6, b5, b6\}$ is *white-dominated*, since White has a first-player strategy for the subgame on these cells that leaves every stone white or dead. The first move in this strategy is to *b5*, so for this strategy *b5* is *white-dominating* and the remaining cells are *white-dominated*. When White is searching for a winning move, it is sufficient to consider among the cells of a white-dominated set only the dominating cell since after moving there the remaining cells become white-captured.

To summarize these ideas, let us complete our analysis of Puzzle 4. It is White's turn to move. The cells in $\{a1, c1, d1\}$, $\{f1, e2, f2, f3\}$, and $\{a2, b2\}$ are respectively dead, black-captured, and white-captured. After white- and black-captured stones have been added to the board, the cells in $\{d3, f4, f5\}$ are white-vulnerable, as they would be killed by respective responses, and subsequent black-capturing, at $d4, e4, e5$. The sets $\{b4, a4, b3\}$, $\{b5, a6, b6\}$, $\{e5, d6, e6\}$, $\{f5, e6, f6\}$ are white-dominated by $b4, b5, e5, f5$ respectively.

This analysis is illustrated in the first diagram of Fig. 10, where dead cells are indicated with grey circles, captured stones are marked with dots, white-vulnerable cells are marked by 'v', and white-dominated cells are marked by 'x'. Any cell that is marked can be ignored in the search for a winning move, so there are only six cells left to consider.

As can be seen from the second diagram of Fig. 10, which shows a win-set found after the captured stones have been added, *a4* is a winning move for Puzzle 4. We leave it to the reader to check whether there are any other winning moves.

6 A win-set for Puzzle 3.

Berge designed Puzzle 3 to be a study rather than a puzzle, so there is more than one winning move. A solution that involves play in the upper right region of the board appears in [5].

Another solution is to start at *c11*, and use the threat of connecting the top white group of three stones with the white line ending at *e5* to force play towards the lower white border. A win-set for this solution, verified by a computer program written by Van Riswijck, is shown in Fig. 11. This win-set is minimal, in that it contains no unnecessary cells; if any cell of the win-set is removed and black stones are then placed at all vacant cells and the one removed cell of the win-set, then White can no longer win. As a final exercise, we

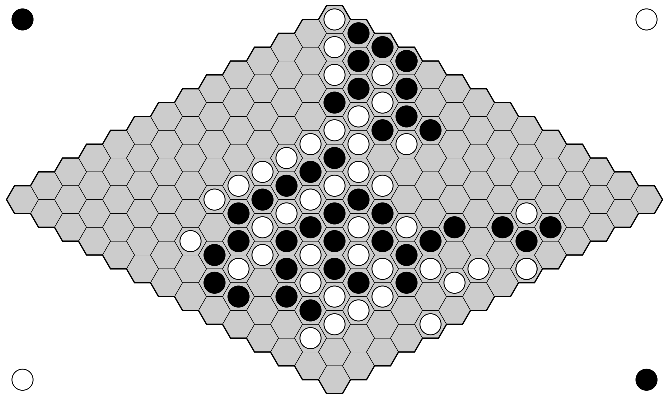


Fig. 8. Puzzle 3, by Claude Berge [1]. White to play and win.

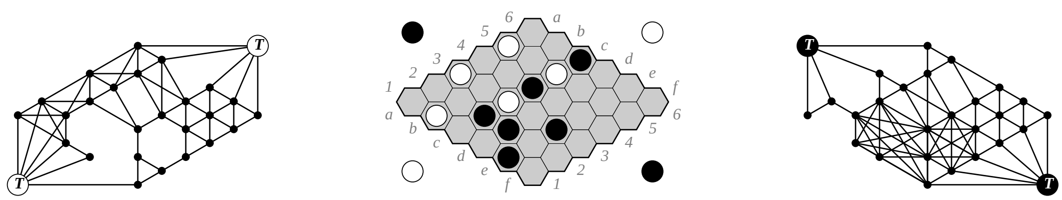


Fig. 9. Puzzle 4, by Piet Hein [7]. White to play and win. The puzzle is flanked by its white/black adjacency graphs.

leave it to the reader to find a winning strategy that uses only the cells of this win-set. An answer appears in Van Rijswijck's doctoral thesis [15].

Acknowledgements

I thank Jack van Rijswijck for his solution to the Berge puzzle, Cameron Browne and Jack for supplying Hex-drawing software, Mike Johanson, Morgan Kan, and Broderick Arneson for their programming support, Bjarne Toft and Thomas Maarup for providing Hein references and translations, Philip Henderson for critical feedback, and Richard Nowakowski for organizing the BIRS conference and encouraging me to finish this chapter.

References

1. Claude Berge. L'Art Subtil du Hex. Manuscript, 1977.
2. Yngvi Björnsson, Ryan Hayward, Michael Johanson, and Jack van Rijswijck. Dead Cell Analysis in Hex and the Shannon Game. In Adrian Bondy, Jean Fonlupt, Jean-Luc Fouquet, Jean-Claude Fournier, and Jorge L.R̃amirez Alfonsin, editors, *Graph Theory in Paris: Proceedings of a Conference in Memory of Claude Berge (GT04 Paris)*, pages 45–60. Birkhäuser, 2007.
3. Martin Gardner. Mathematical Games. *Scientific American*, 197, 1957. July pp. 145–150; August pp. 120–127; October pp. 130–138.
4. Martin Gardner. *The Scientific American Book of Mathematical Puzzles and Diversions*, chapter The game of Hex, pages 73–83. Simon and Schuster, New York, 1959.
5. Ryan Hayward and Jack van Rijswijck. Hex and Combinatorics (formerly Notes on Hex). *Discrete Mathematics*, 306:2515–2528, 2006.
6. Piet Hein. Vil de laere Polygon? Article in *Politiken* newspaper, 26 December 1942.
7. Piet Hein. Polygon. Article in *Politiken* newspaper, 3 February 1943.

Fig. 10. Dead, captured, white-dominated, and white-vulnerable cells of Puzzle 4 (left), and, after dead and captured stones are added, a black weak win-set (right).

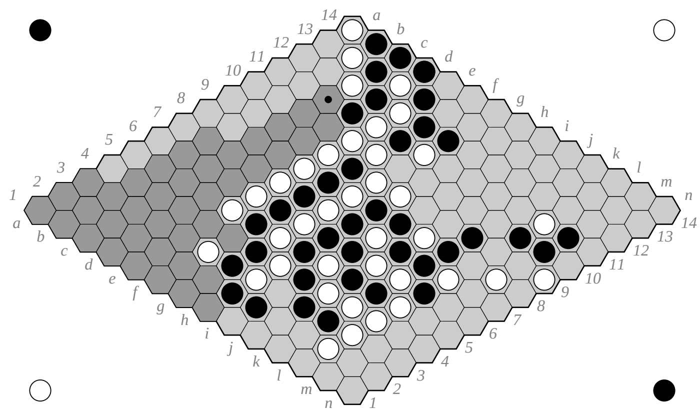


Fig. 11. A white win-set for Puzzle 3.

8. Thomas Maarup. Hex – everything you always wanted to know about Hex but were afraid to ask. Master's thesis, Department of Mathematics and Computer Science, University of Southern Denmark, Odense, Denmark, 2005.
9. Thomas Maarup. Hex webpage.
`http://maarup.net/thomas/hex/`, 2005.
10. Sylvia Nasar. *A Beautiful Mind*. Touchstone, New York, 1998.
11. John Nash. Some Games and Machines for Playing Them. Technical Report D-1164, Rand Corp., 1952.
12. Stefan Reisch. Hex ist PSPACE-vollständig. *Acta Informatica*, 15:167–191, 1981.
13. Jack van Rijswijck. Queenbee's home page.
`www.cs.ualberta.ca/~queenbee`, 2000.
14. Jack van Rijswijck. Search and Evaluation in Hex. Technical report, University of Alberta, 2002.
`www.cs.ualberta.ca/~javhar/research.html`.
15. Jack van Rijswijck. *Set Colouring Games*. PhD thesis, University of Alberta, Edmonton, Canada, 2006.

Algorithm WINVALUE
Input: (B, π), where B is a board configuration and π is the player to move
Output: (v, X), where v is 1/-1 if π wins/loses and X is a win-set

```
    if (B has a winning chain for π) then return (+1, ∅)
    if (B has a winning chain for opponent of π) then return (−1, ∅)
    W ← ∅        [W is the cell set of a winning virtual connection]
    M ← unoccupied cells of B        [M is the must-play]
    while (M ≠ ∅)
        m ← any cell in M
        B′ ← board configuration after adding to B at cell m a stone of π's
        π′ ← opponent of π
        (v, S) ← WINVALUE(B′, π′)
        if (v = −1) then return (+1, S ∪ {m})
        W ← W ∪ S;    M ← M ∩ S
    endwhile
    return (−1, W)
```

Fig. 12. A mustplay-based Hex solver due to Jack van Rijswijck.