IE 619 Ruleset Report

Expand and Conquer

Aditya Khambete[a,1]

[a] *IIT Bombay*

Professor Urban Larsson

Abstract—In this report we explore the combinatorial game 'Expand and Conquer'

Keywords—*Game Values, Bracket Form, All-Small*

1. Introduction

The game *Expand and Conquer* is a 2-player combinatorial game with alternate play. The objective of the game is to force the opponent out of moves and win. This could equivalently be said as 'Last move wins'

The game starts on an m*n board, with some pieces of each color on the board. To understand the

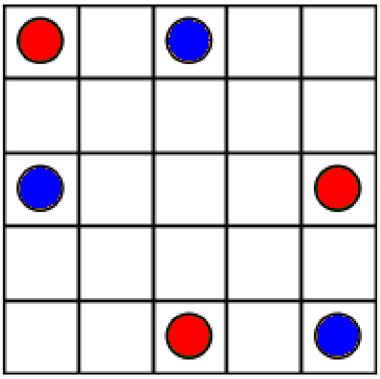


Figure 1. An example starting position

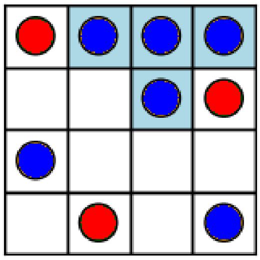
ruleset, let us define some terms we will be using

Definition 1.1. (Connected Group)[1] The set of orthogonally adjacent pieces of same color

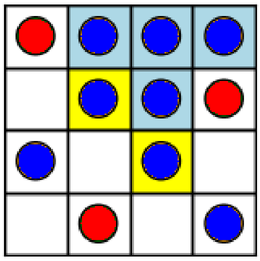
Definition 1.2. (Expanding a Group) Filling in all the orthogonally adjacent squares to your chosen connected group.

The players take turns alternately, on each turn, a player can point at one of his connected groups, and expand it, i.e. all the orthogonally adjacent squares to that group are filled, as shown in fig. 2. Note that squares highlighted by blue denote the chosen group, and those highlighted by yellow denote the expansion.

[1]The terms 'connected' and 'group' are not related to topology, or group theory in this context



(a) An example group



(b) blue expanding the top group

Figure 2. An example move

The game ends when the player has no moves left. The player making the last move wins, as in any standard combinatorial game, something like in section 1

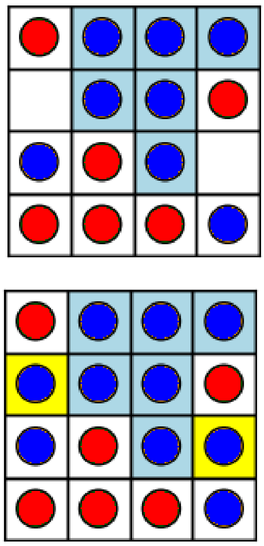


Figure 3. red has nothing to play after blue expands the highlighted group, so blue wins

2. Example Game

Say we have the following starting position-

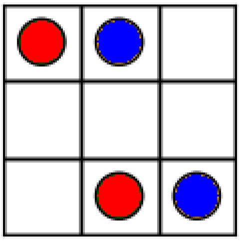


Figure 4. Starting Position

Assume, blue starts and plays the top group, then we get-

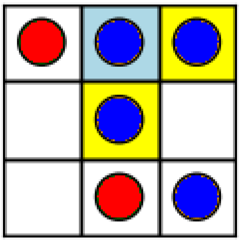


Figure 5. Move 1

Say red responds by selecting the corner group again, leading to-

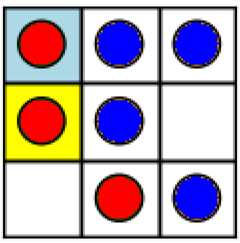


Figure 6

Notice now regardless of what blue will play, the resulting position will be the same, and he can't access the bottom-left corner anymore, which red can conquer by choosing any one of his groups (doesn't matter which one, the position would be the same)

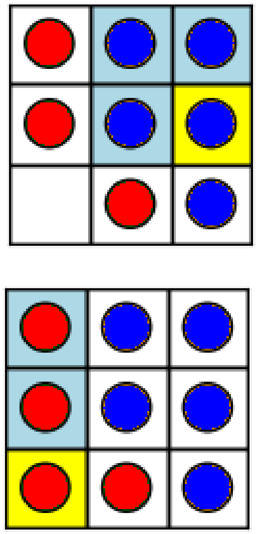


Figure 7. The forced end, Notice even if the other group was chosen (in both positions) the expanded cell would had been the same

So clearly, red wins this game.

On closer inspection it is not hard to see, even if blue had played the other opening move i.e. expanding the corner square in fig. 4 then red would have forced a win by expanding the bottom group. So we notice red could force a win if he moves second. By symmetry, it is very easy to see that blue could use the same strategy and win when red moves first, this means that this example position is a P position, or the game value is 0.

3. Some Game Values

Recall from the [2] the following recursive definition of integer game values.

Definition 3.1. (Integer Game). For all $n \in \mathbb{N}$, define $n = \{n-1|\}$ and $0 = \{|\}$

Before trying to achieve these values in our ruleset, we will need to set a convention of who is the left player and who is the right player, so from now 'blue' will be interchangeably referred to as the left player, likewise 'red' would be right.

So now, let us try to construct the integer positions, to make 0 is not hard, we can just have something like this, which is vacuously 0, as no player has any move So to reach 1 now, we need something where left option is 0, and right has no moves, so we have something like this where there is no red piece, and after 1 blue move, nobody has any moves i.e. the board is 0, so the game becomes $\{0|\}$ i.e. 1.

We can very easily extend this idea,

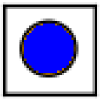


Figure 8. G=0

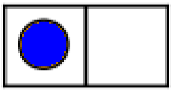


Figure 9. G=1

Theorem 3.2. *By keeping blue in the corner square, and putting in n blank squares next to it, the resulting game value is n.*

Figure 10. We have 5 blank squares next to blue, so the resulting game value is G=5

Proof. This is very easy by induction, we have already established the base case above, similarly when we have $k-1$ empty squares, say the game value is $k-1$ now, say a board with k empty squares, a blue moves result in $k-1$ empty squares, which by induction hypothesis is equal to $k-1$ so resulting game becomes $k = \{k-1|\}$ which proves the assertion. □

If we replace the blue squares in the above analysis by red squares, we get $-n$ instead of $+n$ by simple left-right symmetry.

Now let's see how we can achieve other numbers. Recall how we define Dyadic Rational Game-

Definition 3.3. (Dyadic Rational Game). For all $k \in \mathbb{Z}_{\geq 0}$, let the game $1/2^k = \{0 \mid 1/2^{k-1}\}$. The game $m/2^k$ is m copies of $1/2^k$ in a disjunctive sum.

Now, before trying to achieve this from ruleset, for convenience let us define the following notation

Definition 3.4. (Board String) A string made of $*, O, X$ denoting empty squares, blue and red respectively, where each row is separated by |.

Now we state the following lemma, about finding any game of form $1/2^k$

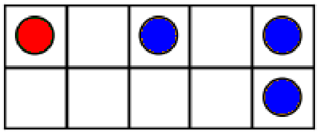


Figure 11. The visual form of board defined by string '$X * O * O| **** O$'

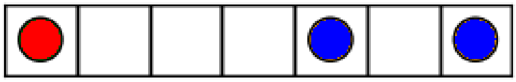


Figure 12. G=1/8

Theorem 3.5. *The game value of $X * \cdots * O * O$ is $1/2^k$ where k is the number of *s between first X and O*

Before proving this, we will need to know the value of positions of the type $X * \cdots * O$, since these will appear again in the proof of the theorem.

Lemma 3.6. *The game value of the above form is 0 if number of *s is even, $*= \{0|0\}$ if number of *s is odd.*[2]

Proof of the Lemma. Say the number of *s is m. Define $G_m = X ** \cdots * O$
For $m = 0$ the board is OX which is 0, since no player has any moves.
Now say our hypothesis is true for $m = k$, now if k is odd, the only move players have is to reduce number of *s by 1, which means to reach a G_{k-1} which is 0, so the game is $*= \{0|0\}$. Similarly if k is even, then we will reach $G = \{* \mid *\}$, which is 0. This can also be seen from seeing, the second moving player can force a win, as the game ultimately boils down to parity
This proves the lemma □

Now let's prove the theorem.

Proof of the theorem. Again, we will use induction. Notice how if the number of dots is 0, then the game board is $XO * O$ again, blue has only 1 move which ends the game immediately, while red has no moves, so the board is $\{0|\} = 1$

Now say the lemma holds for l, i.e. $G_{X*\cdots*O*O} = 1/2^l$, with l *s between X and O
Now lets look at $X * \cdots * O * O$, where there are $l+1$ *s between X and O Notice, the only X option is to reduce number of dots to l, which as we know

[2]By saying number of *s, we mean the empty cells *, and not the game *

is $1/2^l$ from induction hypothesis. Also, blue has 2 options, either expanding the corner group, which makes the game reach $G_{m=l}$ and the other group, which leads to $G_{m=l-1}$ in the lemma, which means one option is $*$ and other is 0 by the lemma.
So, the game reduces to $\{0, * \, |1/2^l\}$. By using the reversibility theorem from **1**, we get the game is $\{0|1/2^l\} = 1/2^{l+1}$
By induction, the theorem is proved. □

Now, to achieve any dyadic rational, all we need is disjunctive sums.

3.1. Achieving disjunctive sums

Recall the definition of disjunctive sum from [2]

Definition 3.7 (Disjunctive Sum)**.** Consider games G and H. Then the disjunctive sum $G + H$ is defined as

$$G + H = \{G^L + H, G + H^L \mid G^R + H, G + H^R\},$$

where $X + G = \{x + G : x \in X\}$ if X is a set of games.

Now, notice the following game position.

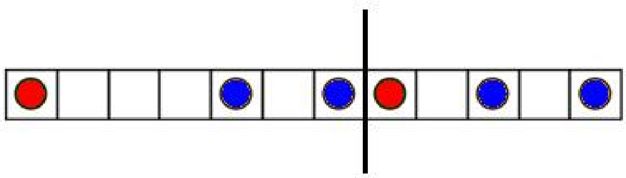


Figure 13. About the line, the game divides into 2 child games which are effectively added together

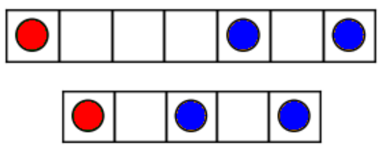


Figure 14. The child games

About the middle line, the game is divided into 2 separate games namely, $H_1 = X *** O * O$ and $H_2 = X * O * O$. Due to the $'OX'$ substring joining these together, these can not interact with each other, which means it is effectively a disjunctive sum of H_1 and H_2 And from theorem 3.5, it is easy to see $H_1 = 1/2^1 = 1/2$ and $H_2 = 1/2^3 = 1/8$, so $G = H_1 + H_2 = 1/2 + 1/8 = 5/8$. This technique makes it easy for us to achieve otherwise hard to achieve game values. In fact, since we have a disjunctive sum defined now, and all the dyadic rational, the following corrolary is immediate.

Corollary 3.8. *Any dyadic rational can be achieved only using 1*n board in this ruleset.*

This follows from the fact that we could get any diadic less than 1 (i.e. the fractional part) by the above technique, and for the integer part, adding the string $** \cdots * O$ will do the trick.

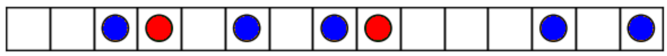


Figure 15. The game 21/8

Notice in the figure above, breaking the board into disjunctive sum yields us-

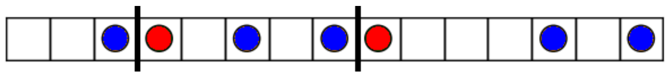


Figure 16. The disjunctive sum

Notice the first (leftmost) game is just 2, since its the exact same position as described in Theorem 3.2, and the other two sum up to 5/8 as we saw before, so the game value of G is $2 + 5/8 = 21/8$

4. All small variation

Recall the definition of all small ruleset.

Definition 4.1. From any game position, either both the left and right players have a move, or none of them has a move.

Now, notice this simple position in our normal ruleset. This anomaly is not only limited to the po-

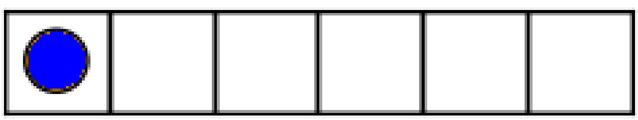


Figure 17. blue has a move, but red doesn't

sitions with piece of only 1 color, see the following position for example.

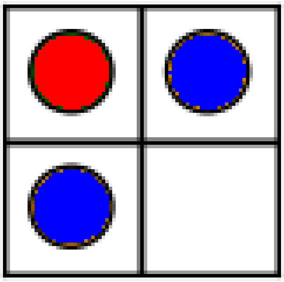


Figure 18. The only red piece is isolated by the blue pieces, so it can't expand

So, to make an all-small variation of our ruleset, we need to add the following simple, yet game-changing workaround. On each turn, a player can either

- *Normal Rule:* point at one of his connected group, and expand it, i.e. all the orthogonally adjacent squares to that group are filled.
  or
- *Added Rule:* place his piece at a square not accessible by any of his groups using Normal Rule, i.e. any square not orthogonally adjacent to any of his connected groups.

To see an example let's say we have the following position-

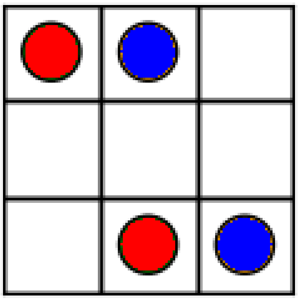


Figure 19

This position we will have the following possible moves (for player O)

1. Group at [(0, 1)] → Expand to [(1, 1), (0, 2)]
2. Group at [(2, 2)] → Expand to [(1, 2)]
   Isolated squares available:
3. Place a single O at (1, 0)
4. Place a single O at (2, 0)

It is easy to see, that 3,4 are the new options blue has due to the allsmall variation.

This simple addition, 'expands' the range of the game. Now the game can even be started on an empty board. On the simple positions also, this rule adds interesting dynamics. For example the game position in fig. 17 is equal to 5 in the normal variation, but in the all-small variation it is equal to $*$[3].

4.1. Atomic Weight Analysis[4]

One of the key parameters about all-small variations are atomic weights.

The easiest atomic game position is $\uparrow = \{0 | \ *\}$. This is easy to achieve by the following position- The

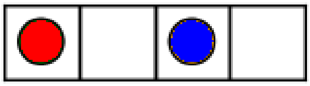


Figure 20. $\uparrow$ game (all-small)

reason is fairly straightforward, if red plays first, the remaining game has both player having 1 move each, which is the $*$ game, and if blue plays first, the game immediately ends, which is 0, so result is $\uparrow = \{0 | \ *\}$, but the problem with this is we can't generalize this and use this in a disjunctive sum, because we should have at both the ends separate colors, here one of the ends is empty, so it will interact with the other boards, which makes $\uparrow\uparrow$ hard to make using only 1*m boards.

However, some playing around initial positions from [1], I discovered the following initial position

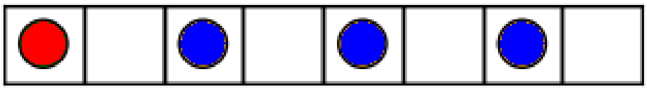


Figure 21. $\uparrow\uparrow$ game

The bracket form computed by the program for this was equal to $\uparrow\uparrow$. However, when the initial position $X * O * O *$, the result I got was $\downarrow$. Similarly on using $X * O * O * O * O *$ the result was 0, but when I computed $X * O * O * O * O * O *$, the result was $\uparrow\uparrow\uparrow$

So say we denote $g_n =' X * O * O \cdots O *'$ where number of O is n, then the game values of first few games are as follows.

[3]Calculated using python code [1] to compute bracket form from game position and CGSuite to simplify it into canonical form [3]

[4]All the game positions in this section are analysed by the all-small ruleset, unless stated otherwise.

Game	Canonical Form	Atomic Weight
g_1	$\uparrow$	1
g_2	$\downarrow$	-1
g_3	$\uparrow\uparrow$	2
g_4	0	0
g_5	$\uparrow\uparrow\uparrow$	3
g_6	$\uparrow\uparrow\uparrow *$	3
g_7	$\uparrow 4$	4

Table 1. Games of form g_n

While computing g_8, the code at [1] crashed. However, based on this observation, here is a conjecture

Conjecture 4.2. *$g_{2n-1} = \uparrow n$ where g_k is defined as above.*

If the above conjecture is true, in games of form 1*m, it is possible to create all integer atomic weights as well, since $\uparrow n.AtomicWeight = n$.

5. Final Remarks

While playing the game on big boards, both versions of the ruleset become very unpredictable and there doesn't seem to be a fixed strategy.

However, using some observation, we conjecture about the empty boards, for allsmall rulesets.

Conjecture 5.1. *For empty boards of size m*n, the game is 0 if m*n is even, else the game is $*$.*

A intuitive strategy for these is playing close to the center, and always expanding that grpup, however there doesn't seem to be a valid explanation on why this will always work, the boards I have used this on, this just seems to be better than the others.

To enjoy playing the game, using theoritical positions is not a very good idea, since most of them we discussed today are just 1*m positions. However, an interesting position to play the game in the normal variation would be some symmetric position like the one in the figure below.

This position also crashes the python code, however it is not hard to see, the position is either P or N, but which one of those is not easy to check, which makes this game interesting. The vast variety of starting positions, and add to that the all-small ruleset as well make the game a fun unpredictable game to play.

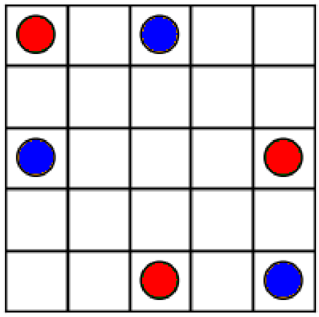


Figure 22. An interesting starting position

A. Further Possibilities

An interesting direction for further studies can be to try to find winning strategies in particular cases, and try to generalise it. This was not possible in this report.

Another direction could be to try and find more such patterns, such as Conjecture 4.2, and try to prove it.

More study on Conjecture 4.3 might also be interesting and lead to some interesting results.

B. Software Use

The main softwares used for making this report are

- To make the game playable, and compute the bracket forms
- Python libraries, for the above purpose, particularly matplotlib for visualising all the game positions
- CGSuite for computing the canonical form of long bracket forms
- Claude 3.7, to help with the coding part, and debugging the Python code in [1]

References

[1] AdityaK1729, *Expand and conquer, github repo*. [Online]. Available: https://github.com/AdityaK1729/Expand_Conquer.

[2] U. Larsson, *Lecture notes in combinatorial game theory*. [Online]. Available: https://www.ieor.iitb.ac.in/sites/default/files/LectureNotesCombGameIE619_Spring2025_OrganizedSections5April_0.pdf/.

[3] A. Siegel, *Cgsuite a computer algebra system for research in combinatorial game theory*. [Online]. Available: https://www.cgsuite.org/.