LUMINES Strategies

Greg Aloupis, Jean Cardinal, Sébastien Collette*, and Stefan Langerman**

Département d'Informatique, Université Libre de Bruxelles,
Boulevard du Triomphe CP212, 1050 Bruxelles, Belgium.
{greg.aloupis,jcardin,secollet,stefan.langerman}@ulb.ac.be

Abstract. We analyze a new popular video-game called LUMINES, which was developed by SONY for the PSP platform. It involves a sequence of bichromatic 2x2 blocks that fall in a grid and must be shifted or rotated by the player before they land. Patterns of monochromatic 2x2 blocks in the terrain are regularly deleted. The primary goal is to contain the terrain within a fixed height and, if possible, clear the grid.
We deal with questions such as the following: Can the game be played indefinitely? Can all terrains be eliminated? We examine how answers to these questions depend on the size of the grid and the rate at which blocks are deleted.

1 Introduction

LUMINES[1] is a popular puzzle video game for the *PSP*[2] platform, originally released in December 2004 in Japan. The original concept was proposed by Tetsuya Mizuguchi and half a million units of this game have been sold within the first year of its release. In this paper, we consider a model that follows the properties of the game as closely as possible, and which is described below.

In this two-dimensional grid game, a sequence of 2x2 blocks falls and the user may rotate and shift each block before it reaches the existing terrain. From this instant, the user may no longer shift or rotate that block, and if only one of the two columns of the block is supported by the terrain just below it, the other 2x1 column continues to fall without intervention until it is also supported. Once a new terrain has been formed, the next block falls.

Each of the four square cells that form a block is coloured either black or white, and any cell that belongs to a 2x2 monochromatic square in the terrain is marked. Note that some of those 2x2 monochromatic squares could overlap. At regular intervals, a vertical line sweeps across the terrain and deletes marked cells. If other cells existed above the deleted ones in the terrain, then they collapse so that a new terrain is formed. An example is illustrated in Figure 1.

The main goal of the game is to manipulate the falling blocks so that the terrain remains within the grid. A secondary goal is to clear the terrain entirely

* Aspirant du F.N.R.S.
** Chercheur qualifié du F.N.R.S.
[1] LUMINES is a trademark of Bandai.
[2] PSP is a trademark of Sony Computer Entertainment Inc.

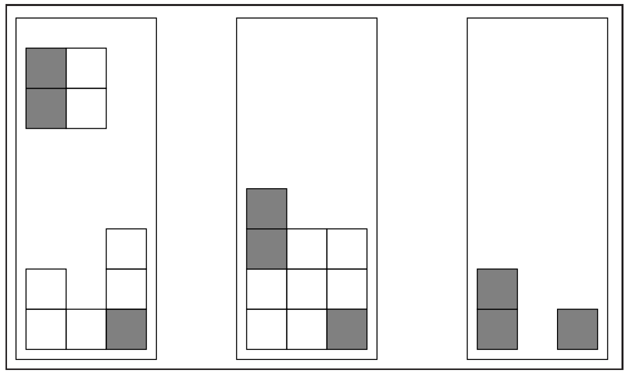

Fig. 1. Example of a block falling: both 2x1 columns fall until supported, the two overlapping squares are deleted, and the the remainder of the terrain collapses.

as much as possible, as this is worth many points. Variations that the game offers include trying to build a target terrain from a given one.

The actual game is played on a grid with 16 columns and 10 rows. While one block is falling, the player can see what the next three blocks will be. The rates at which the sweep-line appears and moves depend on the level of the game.

There are four kinds of blocks (see Figure 2), and six shapes overall:

- (1-2) monochromatic (black and white)
- (3) H-blocks, consisting of two black cells above two white cells
- (4-5) L-blocks, with 3 black and one white cell (and vice versa)
- (6) X-blocks, no two cells of the same color in the same row or column.

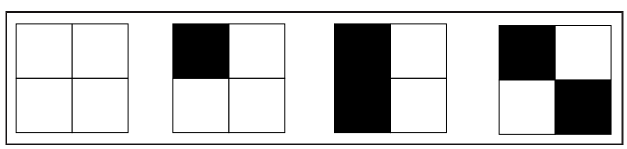

Fig. 2. The four block types in Lumines: monochromatic, L-block, H-block, X-block.

In this work we analyze the game and discuss how to play as best as possible.

Related works include analyses of similar board games, where blocks fall and must be arranged by the player. Tetris, the best known falling-block game invented by Alexey Pajitnov in 1985, has been extensively studied. Burgiel [4] proved that there exists a sequence of tetrominoes that will fill up the board no matter how the player places them. This improves a previous result of Brzutowski [3] who showed that there is no winning strategy if the computer uses the information of how the previous tetrominoes are placed. Similar results have also been proposed by Tucker [10]. Brzutowski however proved that there exist winning strategies for restricted versions of Tetris, in which for instance only one type of tetrominoes appear.

We are not aware of any previous scientific analysis of LUMINES. However, even though the game was released recently, many documents available on the Internet discuss tips and strategies. References and links to previous analyses of the game can be found in [11]. For instance, it is known that the game can be played continuously by compartmenting the board into different sections corresponding to each type of block [5, 6].

Other types of results involve the application of complexity theory to puzzles and games. A survey of algorithmic and complexity results for two-player games and puzzles was published recently by Demaine [7]. He also provided a complexity analysis of Tetris [2, 8], showing that for a given sequence of blocks it is NP-hard to find a corresponding sequence of moves that minimizes the final height of the stack or maximizes the number of cleared rows. Further complexity and decidability results for Tetris are given in the recent paper from Hoogeboom and Kosters [9]. Tetris also served as a model for packing problems (see e.g. [1]).

The known results on strategies for Tetris are to be compared with our result for LUMINES in Section 2. There we propose an optimal winning strategy for LUMINES, that uses as few steps, rows and columns as possible for each section of the board. We analyze the interferences which can occur between sections, and also the impact of the speed of the sweep-line on our strategy. We try to clear the board completely as often as possible.

Section 3 is dedicated to the question of whether any given terrain, made of previously fallen blocks, can be cleared, provided it does not contain any obviously indestructible structure. Although at first sight a positive answer seems within reach, we show that one must take care of traps.

2 Playing Forever

Unless mentioned otherwise, we assume that the sweepline moves quickly across the grid, with respect to the rate at which blocks fall. In effect, we assume that all marked blocks are deleted simultaneously when the sweepline appears.

Lemma 1. *On the original* LUMINES *grid, starting from the empty terrain, it is possible to continue playing indefinitely as long as the sweep-line appears after every block.*

Proof. We provide a simple repetitive strategy that can be used to play forever, and in which every type of block is placed in predefined columns.

First we examine the situation where the sweep-line appears at least four times after every block.

We allocate columns 1 and 2 for monochromatic and H-blocks, and maintain that the terrain will be identical in these two columns. Clearly in such conditions a monochromatic block is deleted as soon as it becomes part of the landscape. H-blocks are always rotated so that they form monochromatic rows, and so that the bottom row matches the top row in the terrain (which, by construction, is monochromatic). The two matching rows will be deleted immediately.

Since the top row of columns 1 and 2, as well as the bottom row of the falling monochromatic or H-block will always be "instantly" deleted, the height of the terrain in these two columns will be at most 2, in the case that an H-block falls on an empty terrain. Of course, the height can temporarily be 4, just before deletion.

Columns 4-6 are allocated for L-blocks that contain three white cells. We provide a sequence of operations so that every four of these blocks combine to yield an empty terrain. To avoid possible interference between monochromatic and L-blocks, we always keep column 3 clear. Columns 8-10 are reserved for the alternate type of L-block, and column 7 is kept clear. Our sequence of operations for L-blocks is shown in Figure 3. This sequence produces a maximum height of 6, and when the last L-block has been added the triplet of columns is cleared in three sweeps.

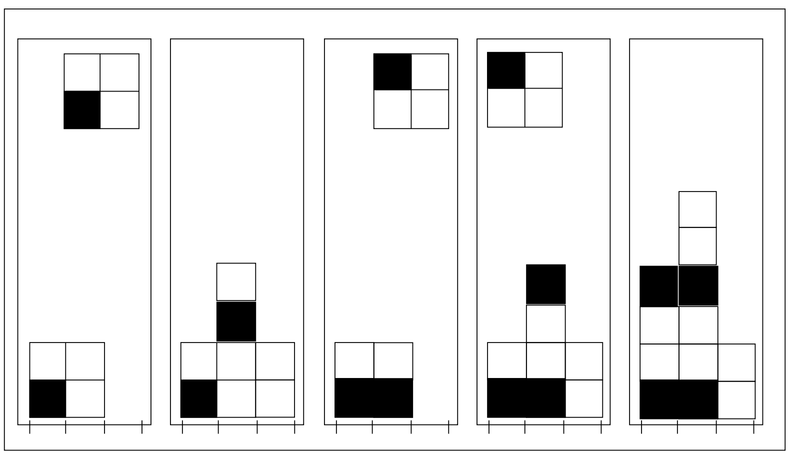

Fig. 3. Sequence of operations for L-blocks. Between the second and third frames, four white cells are deleted and part of the terrain collapses. After the last frame, three sweeps will clear the terrain.

Column 11 is kept clear and we allocate columns 12-14 for X-blocks. We demonstrate how to clear the terrain within these three columns after every four such blocks, in Figure 4. The maximum height in these columns is 8. The three columns are cleared in four sweeps after the last X-block falls.

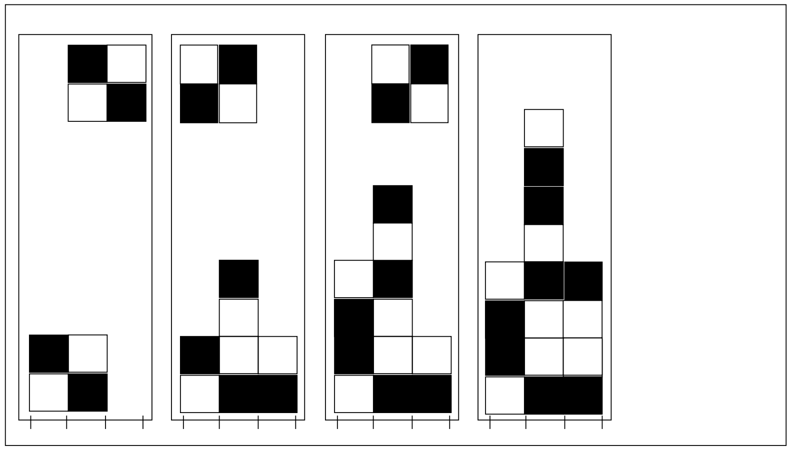

Fig. 4. Sequence of operations for X-blocks. No cells are deleted until after all four blocks have formed a terrain. Then four sweeps clear the terrain.

A closer examination of these separate groups shows that we can place monochromatic and H-blocks in columns 1-2, one type of L-blocks in columns 3-5, X-blocks in columns 6-8, and the alternate L-blocks in columns 9-11. No interference occurs.

For a sweepline that appears only once after each block, we do not modify our method of dealing with each block type. We continue dropping blocks in the same patterns, but now within each triplet of columns the terrain does not follow one simple cycle. Depending on the sequence of blocks and sweep-line appearances, a triplet of columns may go through several other cycles. Each have been verified and stay within a height of 8. For example, in Figure 3, if only one sweep-line appears after the last illustrated step, then just four white cells will be deleted instead of all three columns. Adding an L-block in the first two columns will just stack four cells onto the terrain, but then the four existing black cells will be deleted. In this way, proceeding with our normal pattern of dropped blocks, the sequence of deletions is not affected.

However, the new patterns that occur do cause interference between separate block-type regions. So if we wish to handle a slower sweep-line rate, we do need buffers and 14 columns overall. We have not yet examined whether this interference can be handled otherwise. Finally we just mention a similar strategy which allows indefinite play within 16 columns and 4 rows. However, the cycle of each block type does not include the empty state. □

The fact that one can play LUMINES forever is in sharp constrast with the opposite result known for Tetris. This can be interpreted as LUMINES being ”easier” than Tetris. Yet it seems, based on recent reviews, that the game is no less addictive.

3 Clearing Terrains

A terrain is defined to be a *stable* configuration of cells, in the sense that no cells would collapse if a sweep-line were to scan through. In other words, no four cells of the same colour forming a square exist in a terrain. We say that a terrain is *flat* if all columns have the same height. We make the observation that some terrains cannot be cleared, on a board of finite width. It suffices to construct a flat terrain where the top row consists of alternating colours. No cell in this top row can ever be part of a monochromatic square, so the row will never be deleted. A terrain with no alternating row is called a *legal terrain*.

The goal of this section is to determine whether all legal terrains can be cleared and how to clear them. In the algorithms that follow we assume that enough sweeps occur to delete the terrain.

The following Lemma is useful for clearing terrains.

Lemma 2. *Let (A, B, C) be three consecutive columns with heights $(0, 0, k)$ respectively, where $k \geq 2$. The height of all three columns can be reduced to zero using an appropriate sequence of blocks, whether adjacent columns exist or not.*

Proof. To simplify the main proof assume that no adjacent columns exist, so that no interference occurs with the blocks that we will use within (A, B, C). This will be dealt with at the end of the proof.

Our general strategy is to add two new cells to B so that they duplicate those opposite to them in C. At the same time we add two white cells to A. The idea is to create duplicate columns in B and C, which can be easily eliminated using monochromatic or H-blocks. Following this, the white cells in A can be eliminated using white blocks.

Some deletions may occur as we add blocks in (A, B), and in fact it is clear that the rows in (B, C) will end up having alternating colours. When deletions occur, B and C both lose the same number of cells. Any deletion that happens in A also happens in (B, C), so A will always be at least as high as the other two columns.

We have no method of eliminating a single cell or single row within three columns, and suspect that it is not possible. Therefore it is important to avoid constructing these patterns during our procedure. There are several end-configurations which can lead to the two forbidden patterns if care is not taken. These are not included here due to space constraints. We just mention that we avoid being left with a single row by maintaining A higher than the other two columns.

Interference due to a column adjacent to A can cause its height to decrease. However we can always re-fill it in this case. Interference with C is not really an issue. □

It is easy to take care of the two exceptional height sets (0,0,1) and (1,1,1) if a fourth column is used. It suffices to drop a block so that two cells land in C and the other two in its adjacent column.

Lemma 3. *Given four consecutive columns, including one column C of height zero, the height of a column adjacent to C can also be reduced to zero, with an appropriate sequence of blocks.*

Proof. Examine the case where C is the first column in the set (C, D, E, F):

It is easy to drop blocks in (E, F) so that the height of E reaches far above D, and the excess cells are all white. If E is initially higher than D, then we drop blocks in (D, E) to achieve the same effect. Then we can drop blocks in (C, D), so that cells in C match the original ones of D, while the new cells in D are white and instantly eliminated along with part of E. Thus D does not grow in height as we add cells to C. We can even add one cell to D with the aid of (E, F), to ensure that there is no parity difference between C and D. Thus the alternating rows in (C, D) can be eliminated with monochromatic blocks.

If C is the second column, in the set (B, C, D, E), we do the following:

Simply drop blocks in (C, D) so that the cells in C match those in B and then eliminate the alternating rows in (B, C) as mentioned above. We ommit the details of dealing with a parity difference. □

Theorem 1. *Any terrain of width at least four and containing a column of zero height can be cleared with an appropriate sequence of blocks.*

Proof. We use Lemma 3 to create two adjacent empty columns, and then repeated uses of Lemma 2 to clear the remaining columns. □

Corollary 1. *If a terrain has minimum height h, with an appropriate sequence of blocks the entire terrain can be flattened to height h, or the minimum height can be reduced.*

Proof. If the height cannot immediately be reduced, then use the column C which has height h to apply the technique of Lemma 3. □

The preceding result almost leads one to conclude that all legal terrains can be cleared with an appropriate sequence of blocks. However, in Figure 5 we illustrate a legal terrain that cannot be cleared.

The top row is indestructible, unless some cell is dropped into the "well". If it is white, it will complete an alternating row instantly. If it is black, it will trigger a series of deletions which result in the bottom row becoming alternating. This does not contradict Corollary 1, since the terrain can be reduced for a few steps, but when the minimum height is one it can only be flattened, not reduced further. Corollary 1 does not specify that once the terrain is reduced to height h, it will also be legal.

We also have an example of an indestructible terrain with a well of width two, shown in Figure 6. Again the top row is indestructible, and either of the

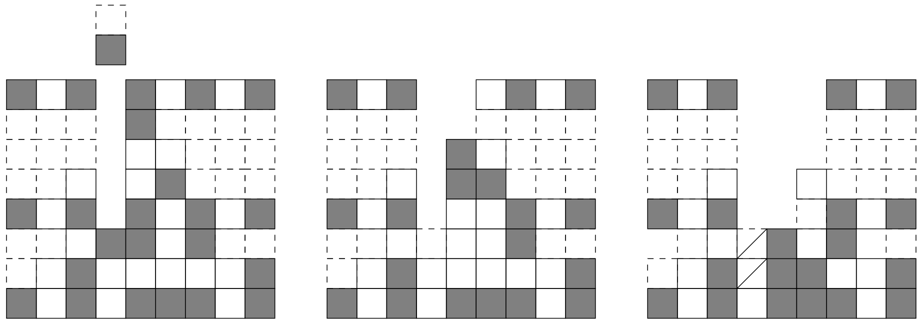

Fig. 5. Left: an indestructible terrain and a cell that triggers a series of deletions. Blocks with dashed sides have irrelevant colour; Middle: after the first deletion; Right: after the second deletion. The two blocks with a diagonal may or may not be present. The final deletion which causes the lowest row to become alternating is not shown.

two positions at the bottom awaits a cell which will trigger disaster or help to complete an alternating row at that level. For lack of space, an illustration of the triggered deletions is ommitted. We just mention the interesting construction in which two separate triggers form different cascading deletions which ultimately affect the exact same position at lower height.

Let a *feasible* terrain be one which has at least one column of height 0. We have already proved that feasible terrains can be cleared with an appropriate sequence of blocks. Let a *fast* sweep-line be one which appears at least four times after every block falls.

Theorem 2. *Any feasible terrain with at least 15 columns and arbitrary height can be cleared, assuming that all block types appear eventually after any given instant, and that the sweep-line is fast. If the sweep-line is not fast, then 3 more columns are needed.*

Proof. Since we allow arbitrary height, we can use two consecutive columns that are furthest from the column of height 0, in order to place blocks that are not useful to us at the moment that they appear. This allows us to assume that we have at our disposal exactly the sequence of blocks that we need. Since the two storage columns are at one extreme of the grid, there are 13 consecutive columns in which to apply the strategy of Theorem 1.

After clearing 13 columns, we can begin using the first 11 as a storage for useless blocks, according to Lemma 1. This allows us to clear the two initial storage columns, with the aid of the two columns nearest to them.

Of course, we will have to wait until a moment when each of our block types has "self-eliminated" within its allocated column space, for the terrain to actually be cleared. If blocks fall randomly, this will happen with probability one. There is simply no deterministic method of achieving this goal, due to adversarial arguments. □

To conclude, we claim that any feasible terrain with height h can be cleared without ever reaching a height greater than $2h$ (at least if the width of the grid is greater than 30), and we conjecture that this bound is tight. Terrains requiring $\frac{3h}{2}$ exist.

As we have seen, in LUMINES not all terrains can be cleared. In comparison, *any* Tetris terrain can be cleared if an appropriate sequence of tetrominoes is given. In this sense, LUMINES is harder.

Future work may involve giving a complete characterization of clearable legal terrains.

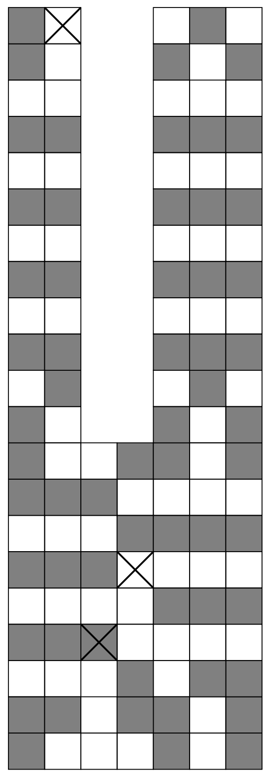

Fig. 6. An indestructible terrain. The black marked cell ultimately takes the place of the white cell below it in the lowest row. Depending on which side is triggered, one of the two marked white cells will also fall into the lowest row. The left trigger involves 9 deletions, and the right one involves 12.

Acknowledgments

This project was initiated during the algorithmic meetings at the ULB Computer Science department. The authors thank all participants, in particular Guy Louchard and Barry Balof, for helpful discussions.

References

1. Y. Azar and L. Epstein. On two dimensional packing. *Journal of Algorithms*, 25(2):290–310, 1997.
2. R. Breukelaar, E. D. Demaine, S. Hohenberger, H. J. Hoogeboom, W. A. Kosters, and D. Liben-Nowell. Tetris is hard, even to approximate. *International Journal of Computational Geometry and Applications*, 14:41–68, 2004.
3. J. Brzutowski. Can you win at tetris? Master's thesis, The University of British Columbia, 1992.
4. H. Burgiel. How to lose at Tetris. *Mathematical Gazette*, (81):194–200, 1997.
5. Casey. Lumines: Killing the fun. *http://caseyporn.com/blog/archives/000818.html*, 2005.
6. Casey. Lumines: Killing the fun, part two. *http://caseyporn.com/blog/archives/000819.html*, 2005.
7. E. D. Demaine. Playing games with algorithms: Algorithmic combinatorial game theory. In L. N. in Computer Science, editor, *Proceedings of the 26th Symposium on Mathematical Foundations in Computer Science (MFCS 2001)*, pages 18–32, 2001.
8. E. D. Demaine, S. Hohenberger, and D. Liben-Nowell. Tetris is hard, even to approximate. In *Proceedings of the 9th International Computing and Combinatorics Conference (COCOON 2003)*, pages 351–363, 2003.
9. H. J. Hoogeboom and W. A. Kosters. The theory of tetris. *Nieuwsbrief van de Nederlandse Vereniging voor Theoretische Informatica*, 9:14–21, 2005.
10. R. Tucker. Tetris. *Eureka Magazine*, (51):34–35, 1992.
11. Wikipedia contributors. Lumines. *Wikipedia, The Free Encyclopedia*, 2006.