

CAN YOU WIN AT TETRIS?

By

John Brzustowski

B. Math, University of Waterloo, 1988

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE

in

THE FACULTY OF GRADUATE STUDIES
DEPARTMENT OF MATHEMATICS
INSTITUTE OF APPLIED MATHEMATICS

We accept this thesis as conforming
to the required standard

.....
.....

THE UNIVERSITY OF BRITISH COLUMBIA

March 1992

© John Brzustowski, 1992

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

T

R

Department of Mathematics
Institute of Applied Mathematics
The University of British Columbia
2075 Wesbrook Place
Vancouver, Canada
V6T 1W5

Date:

23 March 1992

Abstract

TETRIS is a popular video game in which you try to fill rows in a rectangular well using a sequence of tetrominoes chosen by the machine. Each time you succeed in filling a row, it is deleted from the well. Your game ends when you have stacked pieces up to the top of the well. I build a model of TETRIS and analyze the worst-case scenario, in which the machine is treated as an adversary. I say you have a winning strategy when you can make your game last indefinitely. I construct winning strategies for some subsets of the TETRIS pieces, and prove that none exists for some others. Finally, I compare these analytic results to some empirical average-case data that I obtain from a passive survey of TETRIS players.

Table of Contents

Abstract	ii
List of Tables	v
List of Figures	vi
Acknowledgement	viii
1 Introduction	1
1.1 The Real Game	1
1.2 TETRIS meets Plato	4
2 Strategies	6
2.1 Doing well at tetris and TETRIS	6
2.2 Telling you how to play	7
2.3 Strategies for tetris	8
2.4 Meaningless numbers	9
3 Two at a time	17
3.1 Squares with bars or elbows	17
3.2 Elbow couples, single elbows, and the bar	22
4 Organic tetris and the cycle-band	26
4.1 Life history of flats	26
4.2 Killing flats with squares	27

4.3	A recipe for your defeat	30
4.4	Published recipes and two minor fallacies	33
4.5	The cycle-band	34
5	Kinky Sets	38
5.1	Playing in the cycle-band	38
5.2	The shield in the cycle-band	43
5.3	Defeating you with kinks	45
6	Conclusion	49
6.1	Pessimism vs. Realism	49
6.2	The player survey	50
6.3	Personal Conclusion	52
	Bibliography	56

List of Tables

2.1	A summary of winning single-piece strategies.	13
3.2	A summary of winning two-piece strategies for wells of even width. . . .	25
6.3	The average rank of TETRIS pieces.	49
6.4	Results of the TETRIS player survey.	54

List of Figures

1.1	A schematic diagram of the TETRIS machine.	2
1.2	Typical TETRIS screens.	3
1.3	The seven TETRIS pieces.	3
2.4	Four examples of arrows, and what their labels tell you to do.	8
2.5	A strategy for playing only squares.	9
2.6	A strategy for bars	10
2.7	A strategy for left elbows.	10
2.8	A strategy for left kinks.	11
2.9	A strategy for tees.	11
2.10	Strategies for left elbows in wells of widths 3 and 5.	14
2.11	Strategies for tees in wells of widths 3 and 5.	15
2.12	The proof that you can't win with kinks in odd wells.	16
3.13	A winning strategy for bars and squares.	17
3.14	A winning strategy for right elbows and squares.	18
3.15	Making and smoothing a bumpy lane.	19
3.16	Playing a square in the bumpy lane clears two rows.	21
3.17	Waiting for an elbow	23
3.18	Waiting for a bar.	24
4.19	Specimens of flats.	27
4.20	A changing population of flats.	28

4.21	\square_2 with some \square_2 -targets and \square_3 with some \square_3 -targets.	29
4.22	A sequence of plays you might make.	31
4.23	A cycle induced by left kinks.	35
5.24	Left- and right-immune flats.	38
5.25	What $P(x)$ means.	39
5.26	The only way to fill the empty cell of an $\square\square$ segment in flat f	40
5.27	Four ways you can play a left kink.	40
5.28	Lanes in a left kink cycle-band.	42
5.29	Finding the shield in the cycle-band.	44
5.30	Playing one more kink.	45
6.31	The TETRIS player survey form.	52
6.32	Instructions for TETRIS player survey participants.	53

Acknowledgement

These people deserve (and are hereby given) my thanks for their help and/or encouragement: my supervisor Richard Anstee, my second reader Tom McCormick, my parents Tom and Louise, my brothers Marc and Paul, and (in alphabetical order) Bob Bajwa, John Baker, David Balsler, Doug Brigham, Judy Burke, James Cash, Donald D. Cowan, Jack Edmonds, Kensi Gounden, Mike Ho, Kaz Hobbes, Bryce Kendrick, Leah Edelstein-Keshet, Paul McLelland, Gracia Murase, Sunmbola Oyawoye, Luigi Pavan, Anthony Poh, Eric Sather, Stephen M. Smith, Phil Taylor, Sébastien Trudel, Kazuko Tsurumi, and a couple of dozen anonymous TETRIS players at UBC. All of them have made positive contributions to the completion of this thesis, whether they realize it or not.

In addition, these people get my thanks for purchasing, sight unseen, the “education” which this thesis brings to a close: the citizens of Ontario, British Columbia, and Canada. I hope someday to compensate them.

... instead of a classical concerto, I chose one of my own. While I might not be able to compete successfully in performance of a classical concerto, there was a chance that my own might impress the examiners by the novelty of technique; they simply would not be able to judge whether I was playing it well or not! On the other hand, even if I did not win, the defeat would be less mortifying since no one would know whether I had lost because the concerto was bad or because my performance was faulty. [1]

Chapter 1

Introduction

If I tell you I'm doing "applied math", you might ask, with raised eyebrows, "applied to what?". If I respond with "plants", "people", or the name of any other complicated part of the real world, your interest might well turn into suspicion. To avoid this, I've chosen to study a simple object, one whose reflection in the Platonic world of mathematical existence is clear enough that you will find no cause for alarm in the analysis I perform on it.

1.1 The Real Game

TETRIS is a video game invented by Russian mathematician Alexey Pazhitnov and first programmed by Vadim Gerasimov. I'll describe first the arcade version (©1988 by Atari Corporation, TM and ©1987 by AcademySoft-Elorg) and then the generic version which I actually analyze. Figure 1.1 shows what you'll see if you venture into an arcade for a game. After absolving yourself of a quarter and selecting single player mode, you must choose whether to begin at the **Easy**, **Medium**, or **Hard** level.

Figure 1.2 is a sample of what appears on the screen in a typical TETRIS game. The large rectangular region is the **well**, and the shaded squares in it are coloured **tiles**. (On Easy level, you begin the game with an empty well.) The well has space for 20 rows of ten tiles each. At the top of the well, you can see the **current piece**, which is made up of four tiles. The **lookahead piece** is shown just outside the well. If you don't touch the controls, the current piece will drop straight down the well until it hits a tile, or the

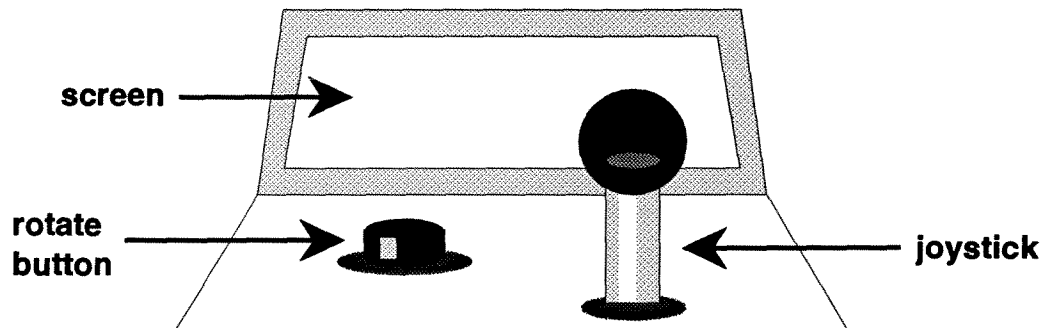


Figure 1.1: A schematic diagram of the TETRIS machine. The quarter slot is not shown.

bottom of the well, and stops. The lookahead piece then becomes the current piece at the top of the well, and a new lookahead piece is displayed outside. The game ends when tiles have accumulated to such a height that the machine can't place a new piece at the top of the well.

Most TETRIS players are not content merely to watch this (expensive) stacking, and instead, grab the controls. Hitting the **rotate button** rotates the current piece 90° clockwise as it falls (see Figure 1.3). Pushing the joystick left or right moves the current piece in that direction, again while it falls. (Pulling the joystick toward you causes the piece to fall more rapidly, and is a sign of impatience.) The point of all this is to place pieces so as to create full rows of tiles. After a piece stops falling, and before the next piece enters play, TETRIS deletes every full row of tiles from the well (you are said to have **cleared** these rows), and each tile in the well drops down one position for every row deleted below it. Clearing rows delays the end of your game, since it moves tiles away from the top of the well. Moreover, depending on how many rows (1, 2, 3, or 4) you clear with a piece, points (50, 150, 450, or 900) are added to your **score**, which is shown to the right of the well in Figure 1.2.

A game of TETRIS is divided into rounds on the basis of how many rows you've

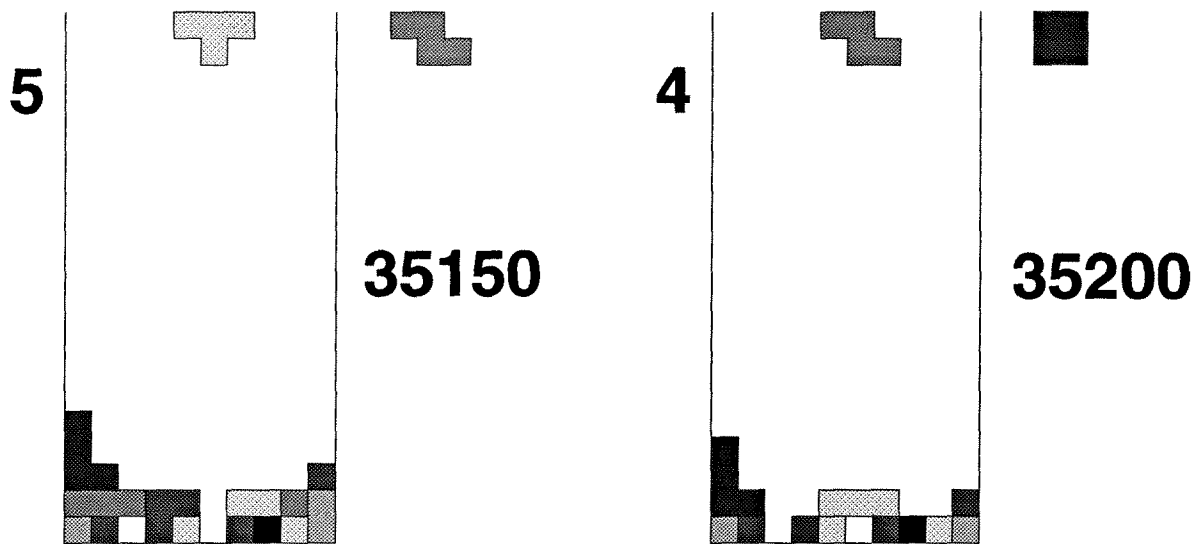


Figure 1.2: The typical TETRIS screen on the left can lead to the one on the right if you don't touch the controls. The tee has fallen straight down, filling and clearing the second row from the bottom. As a result, you are awarded 50 points, and have one less row left to clear.

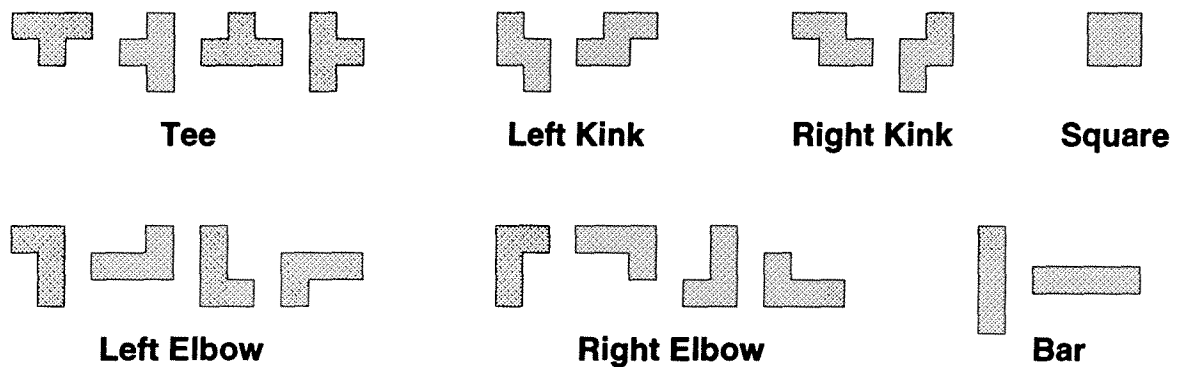


Figure 1.3: The seven TETRIS pieces are shown in all of their orientations. Hitting the rotate button gives the next orientation (in cyclic order from left to right) of a given piece.

cleared. The number still to be cleared in the current round is displayed outside the well (it's 5 in Figure 1.2). At the end of each round, you're awarded a number of bonus points that depends on the maximum height of tiles in the well at that time (the lower the tiles, the more points earned). The next round begins with a new well (which may already contain some tiles) and a new goal of rows to clear. The speed at which pieces fall generally increases from one round to the next, leaving you less and less time to decide where to place them. Moreover, some rounds include extra complications: the machine adds tiles to the well at random locations and times, or it inserts extra rows of tiles at the bottom of the well, pushing up any tiles already in the well.

In most implementations of TETRIS for microcomputers and home video game units, these complications don't exist. Rather, the entire game is just like the first round of the Easy level of arcade TETRIS: you begin with an empty well, and the game ends when you have filled cells up to its top. It's this **generic** version that I'll call "TETRIS" from now on.

1.2 TETRIS meets Plato

In constructing a mathematical model of the game, I'll make several simplifications and idealizations. You should recognize TETRIS behind this axiomatic façade, but I postpone a full discussion of its validity until Chapter 6, where the opinions of other TETRIS players are included. My abstraction of TETRIS, namely **tetris**, follows.

First, the **well** is a rectangular array of **cells**, each of which is either **full** or **empty** (so I've removed colour). **Rows** of cells are numbered from the bottom of the well, and **columns** from its left, both starting at 1. The well has no fixed depth, so the rows of empty cells extend to the heavens, but pieces enter into play just above the highest full cells. The pattern of empty and full cells in the well is called its **state**, and changes as

the game proceeds.

Next, the rules of movement are just as in TETRIS: you can move a piece down, left, and right, as well as rotate it in 90° increments, until you decide to let it drop. (Disputes over the validity of rotations in the vicinity of full cells can be resolved by a Euclidean referee: the well and the piece are copied to the plane, and a rotation in the original well is allowed only if it can be performed in the plane without ever having the piece intersect a full cell.) The piece then falls straight down, without further rotation, until its descent is stopped by either a full cell or the well floor. The status of the four cells it occupies then changes from empty to full, and if this fills any rows, they are deleted and the well compressed, just as in TETRIS. I let you move pieces left and right as far as you wish, but notice that gravity still acts once you drop them. More importantly, I've eliminated time as a factor: you can take as long as you want to play a piece.

Finally, the game ends when you fill a cell above row 20; this is essentially the same as in TETRIS. To summarize, a game of tetris is a sequence of **plays**, each of which proceeds as follows:

- the machine hands you a piece, while displaying the next piece you'll be given
- you put the piece into the well above all full cells, then slide and rotate it into any position, never moving it through a full cell
- you let go of the piece, and it falls straight down as far as possible
- the machine deletes each full row, moving all rows above it down one position
- if there is a full cell above row 20, the game ends

You'll notice that I've said nothing about tetris scores. This is because there won't be any! The reasons why not lead to the central concept of my thesis, and you'll find them in the next chapter.

Chapter 2

Strategies

Let me introduce you to the main concept of this thesis, but first, some motivation.

2.1 Doing well at tetris and TETRIS

Your high score is one measure of your TETRIS-playing ability, but perhaps you're so talented that you can actually beat the game: you can make your score as high as you like. But is this really possible? Can TETRIS be beaten? This is *almost* the question I'll answer, but first, a matter of equity.

With all due respect, is it fair that you must compete against a machine? Your reflexes are no doubt spectacular, but can they hold out indefinitely against an unwavering silicon opponent when you are forced to play faster and faster using a sluggish joystick? Surely not. For there to be any hope of beating TETRIS, its pieces cannot be allowed to fall so quickly that you lose track of the game: there must be a speed limit. But since a reasonable speed limit for you might be still too fast for other players, I'm going to set it *so* low that, effectively, all players can take whatever time they need in positioning a piece. That's what I've done in tetris: the speed and timing factors of TETRIS have been eliminated. So, the fairer question I ask is "Can *tetris* be beaten?"

"Wait a minute", you implore, "how can I beat tetris when I don't even get any points?" The answer is to use the length of your game, rather than a score, as the measure of your ability: if the game ends, you have lost, otherwise, you have beaten tetris. This might seem strange, but think about what you need to beat TETRIS: making

your score very high requires playing for a long time. If you can score as high as you want, you really must be able to make your game last indefinitely. Therefore, if you can beat TETRIS, you can also beat tetris. Conversely, suppose you can beat tetris. Each piece adds four new full cells to the well. If you don't keep clearing rows, then eventually, some of these full cells will be above row 20, and your game will end. Therefore, as long as you continue the game, you must clear, on average, four rows every ten plays, so that the number of full cells doesn't increase past a certain point. But if you can do this in tetris, then you can do it (with good reflexes) in TETRIS.

I could end here simply by stating "Yes, *I* (or someone else) can beat tetris", but you'd probably be less than convinced (especially if you had seen me play). Alternatively, I could state "*No one* can beat tetris", but you would doubt this even more: had I actually investigated every claim to the contrary ever made, and found each to be false? Surely what you demand is either a way of beating tetris, or a proof that it can't be done.

2.2 Telling you how to play

If there's a way to beat tetris, then I can describe it in pictures. Figure 2.4 shows what such pictures are composed of, namely state diagrams with arrows between them. In a state diagram, the well is a simple outline which I draw only as tall as I need to show you all the state's (shaded) full cells. If I need to refer to a state, I will label it with a letter just below its diagram. Between some states I draw arrows to represent a play that takes the well from the state at the arrow's tail to the one at its head. Each arrow is labelled with pieces: the current piece is the large one with a number below it, and any small ones are lookahead pieces. The best way to understand how to interpret these labels is to study Figure 2.4.

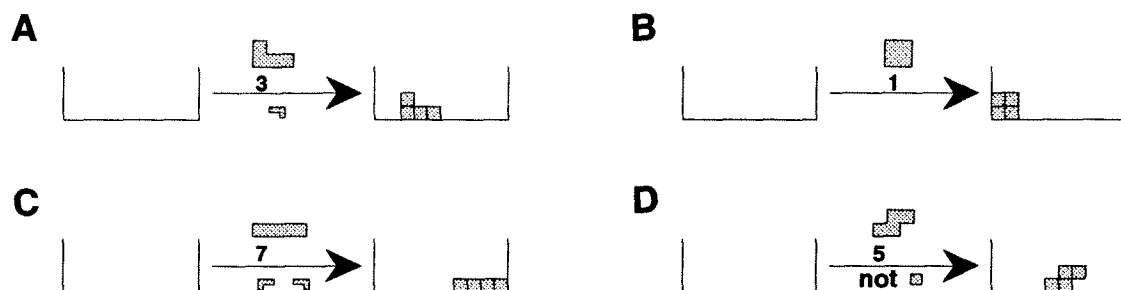


Figure 2.4: Four examples of arrows, and what their labels tell you to do. **A:** When the (big) current and (small) lookahead pieces are both right elbows, rotate the elbow into the same orientation as the big piece, and place it so that its leftmost cell goes in column 3. **B:** when the current piece is a square, and the lookahead piece is anything, play the square so that its leftmost cell goes in column 1. **C:** when the current piece is a bar and the lookahead piece is one of the elbows, play the bar horizontally, with its leftmost cell in column 7. **D:** when the current piece is a left kink and the lookahead piece is any piece other than a square, play the kink horizontally, starting in column 5.

2.3 Strategies for tetris

The pictures in Figure 2.4 are only fragments of instructions: if you try to use them, you will get stuck in the first state unless the machine gives you the same current and lookahead pieces as appear on the arrow. Even if it does, you'll be stuck after the first play, since there are no arrows leading from the second state. You can see that for a complete set of instructions, every state must have enough links leading from it that you won't get stuck, no matter what pieces the machine gives you. Such a complete set of states and links, whether or not it actually beats tetris, is what I call a **strategy**. My quest is to seek a **winning** strategy for tetris.

Any strategy must be able to deal with whatever sequence of pieces the machine sends you. I'll test my strategy building and your arrow interpretation by considering the simplest sequences, those in which all pieces are the same. Figure 2.5 shows a simple

way of playing such a sequence of squares. Notice that the highest row in which a full cell is found is 2. I'll call this number the **height** of the strategy. If the square were the only piece in tetris, this would be a winning strategy, since its never fills a cell higher than row 2, let alone row 20.

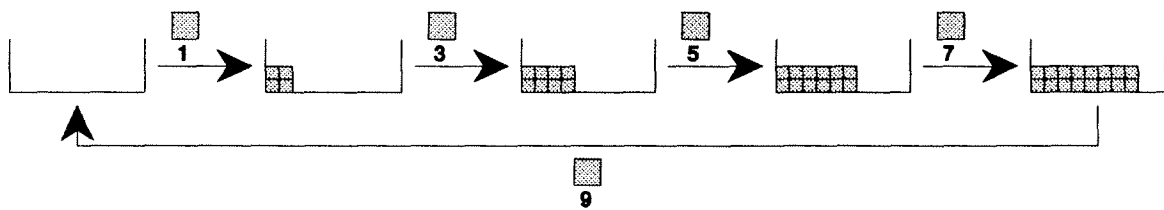


Figure 2.5: A strategy for playing only squares. You clear two rows when you play the square in column 9, and this leads you back to the empty well.

The square is not alone: if the machine sends you a sequence consisting entirely of any one of the pieces, you can make your game last indefinitely by using one of the single-piece strategies shown in Figures 2.6 to 2.9. For kinks and elbows, I've only shown a strategy for the left version of the piece. You can obtain a strategy for the right version by reversing the picture with a mirror, left to right. It's important that these strategies exist, for otherwise, the machine could quickly end your game by simply handing you the same piece over and over again.

2.4 Meaningless numbers

The size of the well in arcade TETRIS happens to be 10 by 20, but there's no obvious reason why it couldn't be different. In some versions of TETRIS, the well is wider, and so I might as well try to find winning strategies for all widths. Similarly, there seems to be nothing magical about 20: other well depths are possible (and likely). I will note

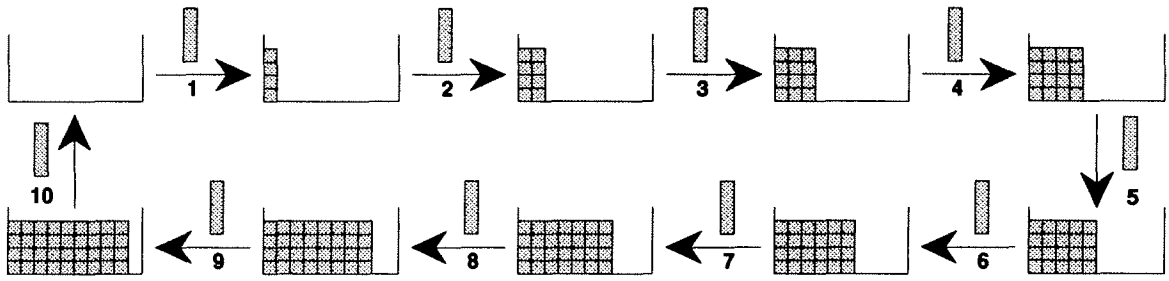


Figure 2.6: A strategy for bars with a height of 4 and 10 states.

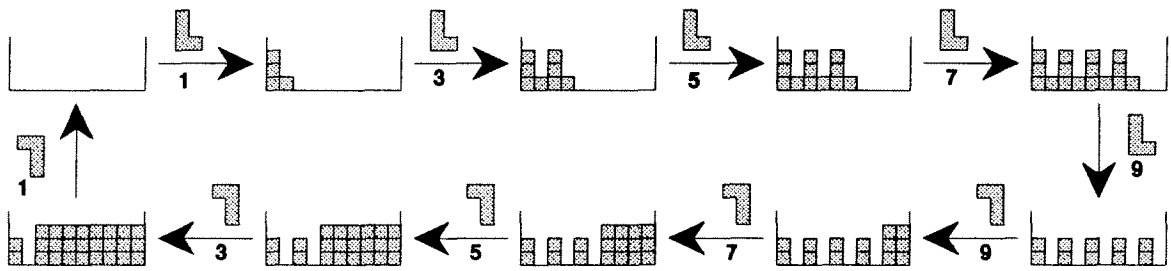


Figure 2.7: A strategy for left elbows with a height of 3 and 10 states.

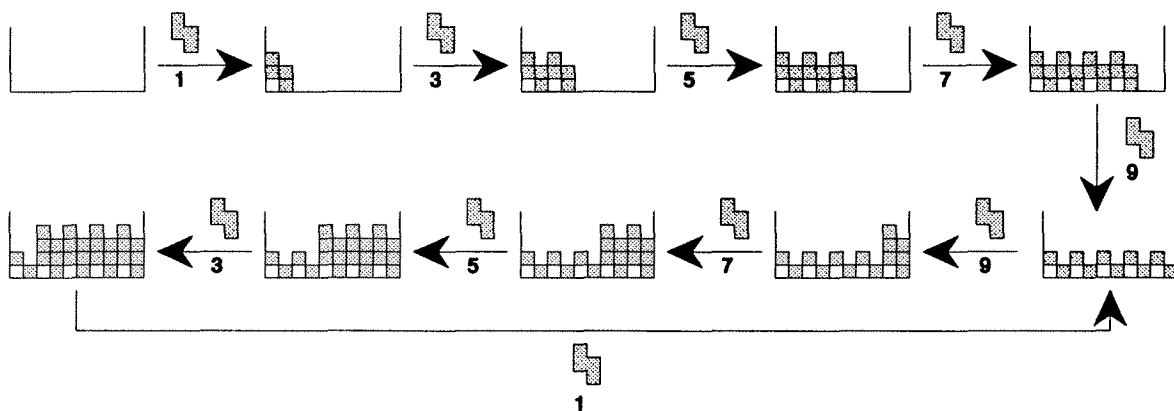


Figure 2.8: A strategy for left kinks with a height of 4 and 10 states.

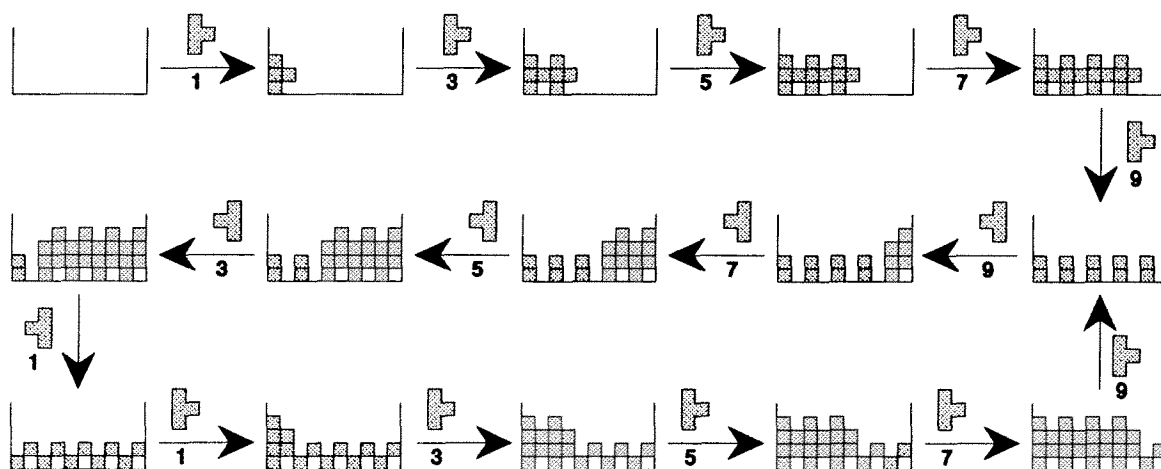


Figure 2.9: A strategy for tees with a height of 4 and 15 states.

the height of each strategy I find so that you will know the depth of well you require in order to use it. (It might be the case that I can find a winning strategy for seven-piece tetris in a well of depth 21, but that none exists for the real well of depth 20.) For single pieces, you'll see that the width of the well usually doesn't matter.

I begin by showing you that the single-piece strategies of Figures 2.5 to 2.9 easily lead to strategies for all wells of even width. You can think of the well as being partitioned into five pairs of adjacent columns which I'll call **lanes**, and will number from left to right, starting at 1. Thus lane 1 consists of columns 1 and 2, lane 2 consists of columns 3 and 4, and so on. Notice that in each strategy, pieces are placed entirely within lanes: no piece adds full cells to two different lanes. Moreover, in each strategy, the pattern of playing pieces is identical for every lane. This pattern can be repeated for any number of lanes, giving strategies for wells with any even number of columns. The height of these strategies doesn't depend on the width of the well, except for the simple case of width 2. There, the square has a zero-height strategy (it disappears upon being played), the bar has a height four strategy, and the other pieces all have strategies of height two. (Also, you might have noticed that there is a zero-height strategy for the bar in a well of width 4, and a height one strategy for the bar in wells of widths 8, 12, 16, and so on.)

What about wells of odd width? Squares immediately present a problem, since they are two cells wide, and so you can't even fill a single row with them. This settles the question: there can be no winning strategy for tetris in an odd width well, since the machine can simply send you a sequence of squares, and you'll never clear a row. (Perhaps that's why every version of TETRIS that I've ever seen has a well of even width.)

Are squares the only pieces which prevent you from winning in an odd well? The pattern of Figure 2.6 gives a strategy for bars in odd wells: just play each one vertically. Elbows and tees are more interesting, and in Figures 2.10 and 2.11, I've shown strategies for the left elbow and for the tee in wells of width 3 and 5 (as before, you get a strategy

for right elbows by reflecting the left elbow strategy in a mirror). The figure captions tell you how to construct strategies for these pieces in wells of any other odd width. On the other hand, Figure 2.12 explains why there is no winning strategy for either of the kinks in a well of odd width.

The winning strategies I've found for single pieces are summarized in Table 2.1. They let you beat tetris in wells of various widths only if the machine sends just one type of piece and the well is at least as deep as the height of the strategy.

Table 2.1: A summary of winning single-piece strategies.

Piece	Well Width=2		Well Width= $2n, n > 1$		Well Width= $2n + 3, n \geq 0$	
	Height	No. of States	Height	No. of States	Height	No. of States
Square	0	1	2	n	(no winning strategy)	
Kink	2	2	4	$2n$	(no winning strategy)	
Bar	4	2	4	$2n$	4	$2n + 3$
Elbow	2	2	3	$2n$	8	$4n + 6$
Tee	2	3	4	$3n$	7	$5n + 5$

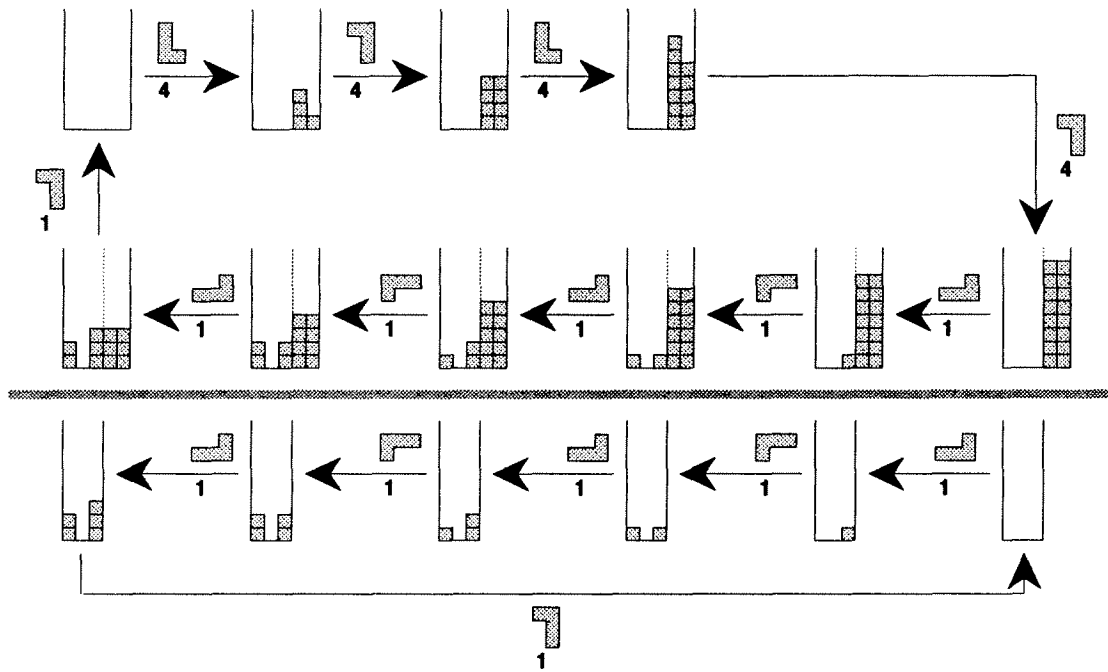


Figure 2.10: Strategies for left elbows in wells of widths 3 (below the heavy gray line) and 5 (above it). The first four plays in the width 5 strategy fill in the two rightmost columns of the well. This leaves three empty columns which can be treated just like a well of width 3. In fact, the next six plays in the width 5 strategy are exactly the same as the plays in the width 3 strategy, and to emphasize this, I've drawn them on top of each other, separated by the gray line (the dashed vertical lines inside the states of width 5 are to help you see this correspondence). For wells of greater odd width, you perform the same trick: fill pairs of columns just like you filled in columns 4 and 5 here, then use the width 3 strategy in the remaining three empty columns.

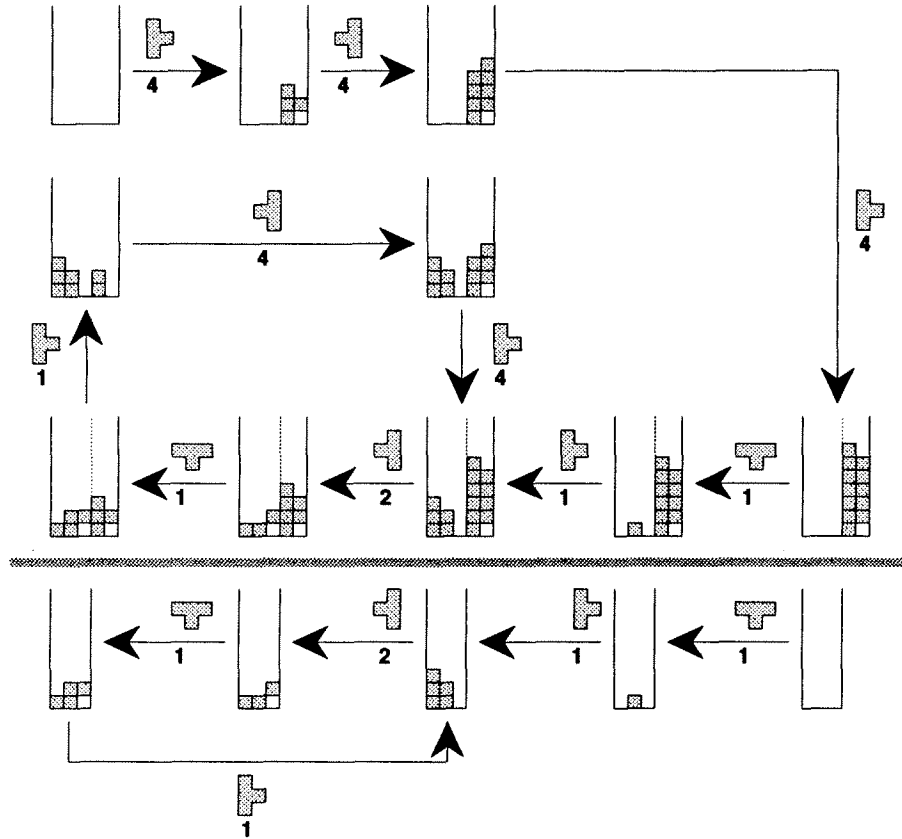


Figure 2.11: Strategies for tees in wells of widths 3 (below the heavy gray line) and 5 (above it). Just as for left elbows, the first part of the width 5 strategy serves to fill (almost) the rightmost two columns, leaving three empty columns which can be treated (almost) like a well of width 3. After the first three plays of the width 5 strategy, the next five mimic those in the width 3 strategy, and I've drawn them above each other, separated by the gray line. To construct strategies for wells of greater odd width, repeat the first three plays of the width 5 strategy in every additional pair of columns. You will then be left with three empty columns in which you can perform the width 3 strategy for the next five plays. Then, you must refill columns 4 and 5, 6 and 7, and so on, just as in the width 5 strategy.

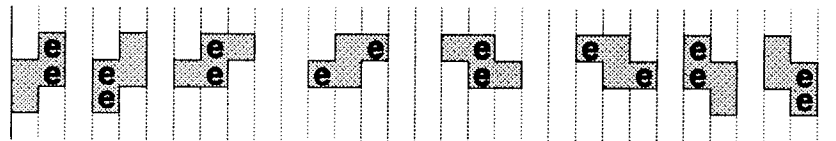


Figure 2.12: The proof that no winning strategy exists for a kink in wells of odd width is easy. Notice that no matter how you play either kink, you create the same number of full cells in even columns (“even cells” – labelled e) as in odd columns (“odd cells” – shaded but unlabelled). I call the number of even cells minus the number of odd cells the **skew**. The picture shows that when you play a kink, you don’t change the skew. A full row has one more odd cell than even cell, so when you clear a row (i.e. when you remove a row of full cells from the well), you increase the skew by one. As you make more plays, the only way to prevent the skew from increasing is to stop clearing rows, but this will soon end your game. If, on the other hand, you keep increasing the skew, then the number of even cells keeps increasing too, and that will eventually end your game — either way, you lose.

Chapter 3

Two at a time

In the last chapter, I gave you winning strategies for tetris played with only one piece. The logical next step is to show you how to play tetris with two pieces.

3.1 Squares with bars or elbows

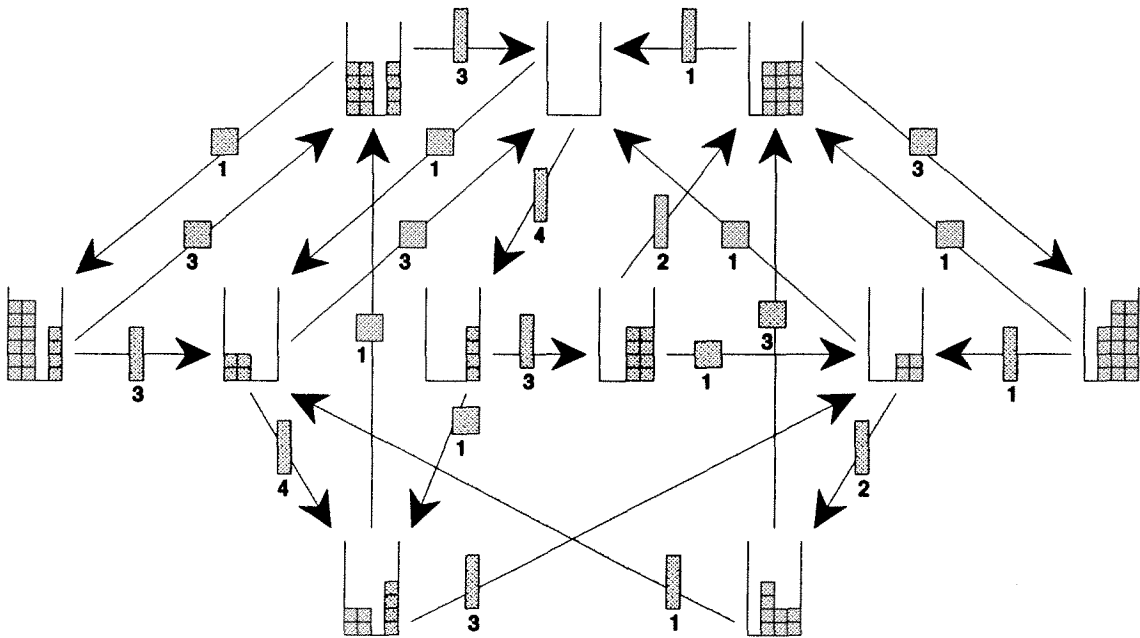


Figure 3.13: A winning strategy for bars and squares in a well of width 4.

In Figure 3.13, I've drawn a winning strategy for bars and squares in a well of width 4. A similar winning strategy for right elbows and squares appears in Figure 3.14 (for left

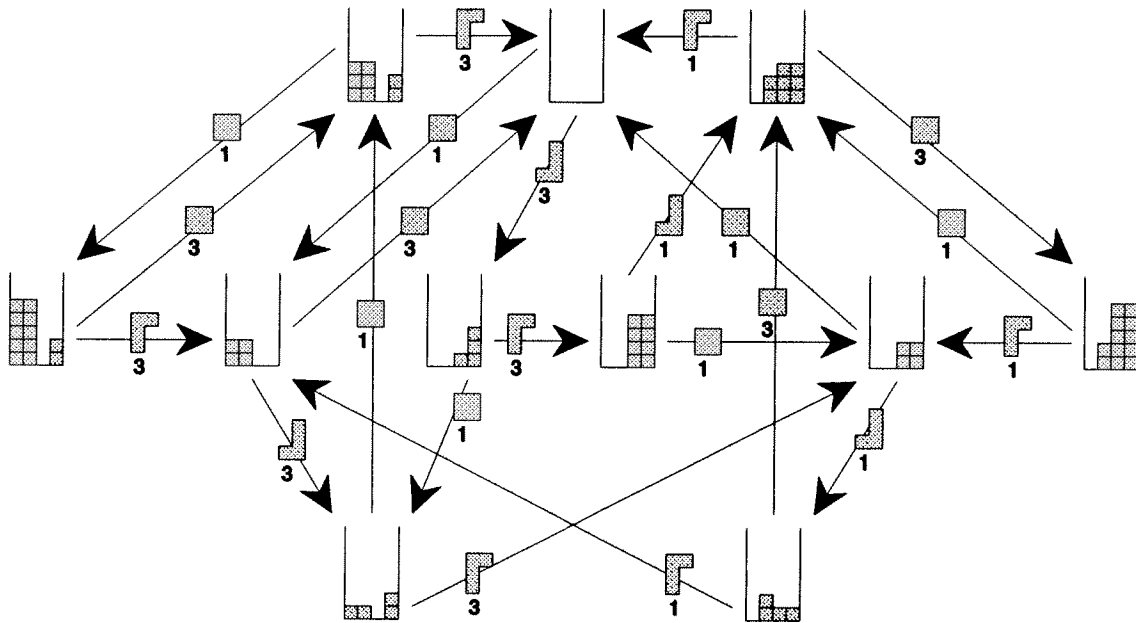


Figure 3.14: A winning strategy for right elbows and squares in a well of width 4.

elbows and squares, use a mirror). Neither strategy uses lookahead information. Similar strategies exist for wells of any even width:

Lemma 1 *If you play tetris using only the square and one of left elbow, right elbow, or bar, then for a well of any even width, you have a winning strategy whose height doesn't depend on the width of the well.*

Proof: You can probably discover the proof yourself, by staring at the diagrams. I'll call the elbow or bar the **non-square** piece. Just as for states and columns, the **height** of a lane is the height of the highest full cell in that lane. Notice that in every state shown in Figures 3.13 and 3.14, each lane, except possibly one, is **smooth**, consisting of a 2 column wide rectangle of full cells (of zero height, if the lane is empty). The possible exception is a **bumpy** lane which, in at least one row, has only a single full cell. You create a bumpy lane when you play the non-square piece in a smooth lane. You can restore that lane to smoothness by playing the same piece in it again — see Figure 3.15.

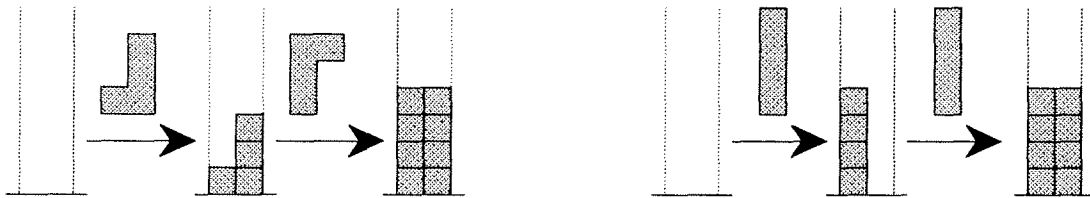


Figure 3.15: How to make a bumpy lane and how to smooth it. By playing a single elbow (left) or bar (right) in a smooth lane, you make it bumpy. By playing the same piece in a bumpy lane, you smooth it. (These plays don't necessarily occur one right after the other.)

For any even width of well, I describe the strategy by telling you how to play each piece:

Square: If there is a bumpy lane and if its height is at least two less than that of any other lane, then play the square in the bumpy lane. Otherwise, play the square in a lowest smooth lane.

Non-Square: If there is a bumpy lane, then play this piece to make it smooth. Otherwise, play this piece in a lowest smooth lane, making it bumpy.

Notice that if every lane in a state is smooth, then at least one lane is actually empty. Otherwise, the bottom row in the state would be full, which is impossible. Therefore, when you play the non-square piece in a lowest smooth lane, you are actually playing it in an empty lane. Also, notice that there is never more than one bumpy lane.

I claim that the strategy above will never result in a state whose height is more than 7. To see this, consider the following three statements:

Statement S1: There is at most one bumpy lane in the well.

Statement S2: If there is a bumpy lane, its height is at most 4.

Statement S3: The height of every smooth lane is at most 7.

Notice that if both S2 and S3 are true for some state of the well, then the height of that state is at most 7. Each of S1, S2, and S3 is certainly true of the empty initial well. I will show that the strategy has a height of at most 7 by showing that every type of play in the strategy preserves the truth of these three statements. That is, if S1, S2, and S3 are all true for a state, then playing either piece leads to another state for which they are also true. There are four cases:

- If you play a square in a lowest smooth lane, then either that lane is empty (if there is no bumpy lane in the well), or that lane is less than two rows higher than the bumpy lane. In the first case, the new height of the lane will be 2, and in the

second case, the new height will be at most 3 plus the height of the bumpy lane, which is at most 7, since S2 was true of the state you played on. In both cases, the new height of the lane you played in is at most 7, and so S3 is still true. S2 is still true since the bumpy lane is unaffected. Also, S1 is true since playing a square never makes a lane bumpy.

- If you play a square in the bumpy lane, it disappears, clearing two rows (see Figure 3.16). The height of the smooth lanes decreases by 2, and the height of the bumpy lane doesn't change. Therefore, S2 and S3 are both still true. Again, S1 remains true since you're playing a square.

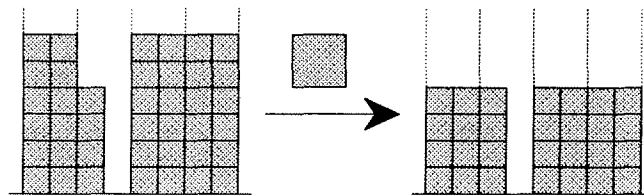


Figure 3.16: Playing a square in the bumpy lane clears two rows.

- If you play the non-square piece in a lowest smooth lane, then there is no bumpy lane and so any lowest smooth lane is actually empty. Thus, you create a bumpy lane whose height is at most 4, and you don't affect the heights of the other smooth lanes. This means S2 and S3 remain true. S1 is still true since you've just created the only bumpy lane in the well.
- Finally, if you play the non-square piece in the bumpy lane, then you make it smooth, but its height will still be at most 4, since a pair of bars or elbows form a 2 by 4 rectangle. The heights of the other smooth lanes only changes if this play

decreases them by clearing rows. Again, S2 and S3 will still be true. Moreover, S1 remains true since you've just smoothed the only bumpy lane.

More careful accounting can actually show that the smooth lanes never reach a height of more than 6 when you play with the bar, and 5 when you play with an elbow, but this is a minor improvement. ■

In this lemma, the description of the strategy was incomplete: I told you to play pieces in *a* lowest smooth lane, giving you the freedom to choose which one. This really means there are several strategies which satisfy the description. Using any of them, you can win at two-piece tetris in an even well if the two pieces are the square and either the bar or one of the elbows.

3.2 Elbow couples, single elbows, and the bar

If you play tetris with the two elbows, or with one elbow and the bar, you have winning strategies which are very similar to those in the previous section. I won't give proofs, but will just describe how these strategies differ from the previous ones. The idea is again to have as few bumpy lanes as possible, but now you might need two of them, one for each type of piece. If there is a bumpy lane of each type in the well, then whatever the current piece is, you can use it to smooth one of those lanes. There is a new problem you might run into, which I'll illustrate using bars and left elbows. Suppose that at some point, the well has a bumpy lane, say l , which requires a left elbow to smooth it. I'll call l an **elbow-deficient** lane. What happens if the machine sends a long sequence of bars? As before, you can play these bars in lanes other than l , taking them to quite a height. Eventually, these lanes will all be smooth, and at least four rows higher than lane l . At this point, you must make a decision (see Figure 3.17). If the current piece is an elbow, you can just play it to smooth lane l , clearing three rows. However, if the current piece

is a bar, playing it in lane l to clear rows would make lane l into an elbow-deficient lane, but of the opposite type. That is, lane l would now require a *right* elbow to smooth it. To avoid this problem, you examine the lookahead piece. If it's an elbow, you play the current bar on top of a smooth lane, and use the elbow to smooth lane l , thus clearing three rows. On the other hand, if the lookahead piece is a bar, then play the current and next pieces (both bars) in lane l , thus clearing four rows and leaving l left-elbow-deficient.

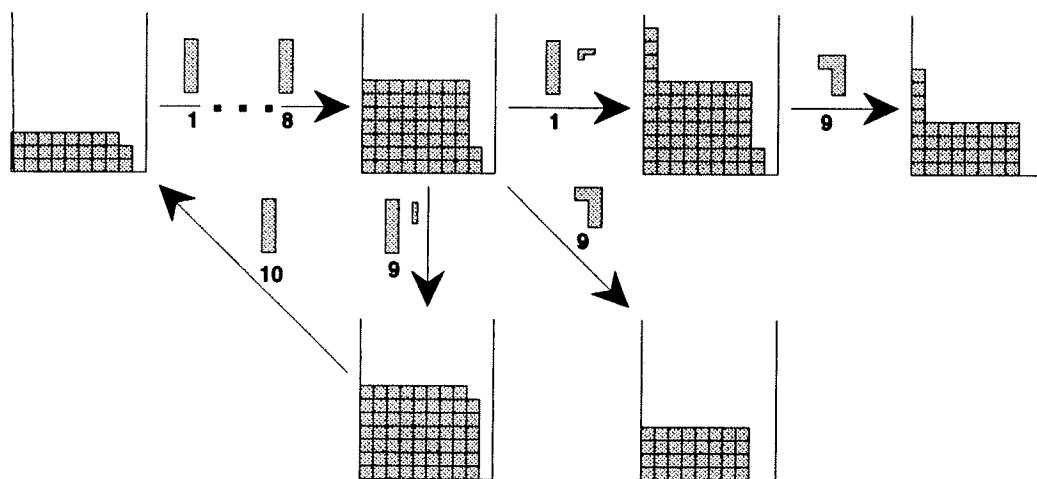


Figure 3.17: Where lookahead is used in playing bars and elbows. Above: if there is an elbow-deficient lane, and you get a long sequence of bars, play them as shown until you get to the second state. There, depending on the lookahead piece, you will either play the bar on a smooth lane, or play two bars in the elbow-deficient lane. Either way, you end up in a state with only one bumpy lane, and with lower smooth lanes.

The trick in the last paragraph also works when the well has a bar-deficient lane and the machine sends you a long sequence of elbows (see Figure 3.18), or when the well has an elbow-deficient lane of one type (left or right), and the machine sends you a long sequence of elbows of the opposite type. Notice that there are never more than two bumpy lanes, and the bumpy portion of those lanes (i.e. the rows in which those lanes are not completely full) each consist of at most four rows. Moreover, no pair of lanes will

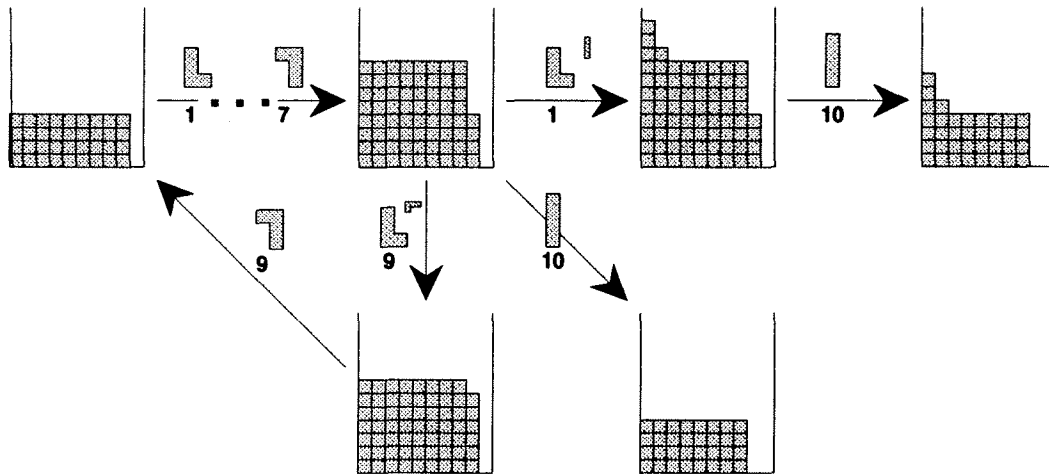


Figure 3.18: If there is a bar-deficient lane, and you get a long sequence of elbows, play them as shown until you get to the second state. There, you use lookahead to determine how to proceed. You end up in a state with only one bumpy lane, and with lower smooth lanes.

ever differ in height by more than a fixed amount. This is enough to show that these are winning strategies, at least for deep enough wells. In fact, careful counting can show that the heights of these strategies are 10 for the pair of elbows, and 11 for one elbow and the bar.

I've summarized the two-piece strategies in this chapter in Table 3.2. You might find it amusing to discover two-piece strategies for other pairs of pieces, say in a well of width 4. In that setting, I quickly and easily found strategies for the pairs of pieces listed in the table. However, for some other pairs of pieces, I didn't find strategies even though I spent a long time trying. There is a good reason for this, but I'll need the next two chapters to give it to you.

Table 3.2: A summary of winning two-piece strategies for wells of even width.

Pieces	Height of Strategy	Uses Lookahead?
Square, Bar	6	No
Square, Elbow	5	No
Elbow, Bar	11	Yes
Both Elbows	10	Yes

Chapter 4

Organic tetris and the cycle-band

I now examine what goes on in the tetris well from a different perspective. The ideas in this chapter will help settle the question of whether there exists a winning strategy for tetris. Moreover, they demonstrate how well-known concepts from biology can be used, as analogies, to discover and explain proofs in mathematics.

4.1 Life history of flats

Flats are multi-celled creatures that live in the bottom of the tetris well (see Figure 4.19). When a game of tetris begins, there are no flats in the empty well. New flats are born whenever you play a piece that fills cells in a previously empty row of the well. Flats grow when your plays add full cells to them, and die and disappear when they've grown to fill an entire row of the well. Flats are territorial (each non-empty row of the well is occupied by just one) but gregarious (there are never any empty rows between them — think about why not). Because they are diffuse (not all of their full cells are necessarily adjacent), you can often drop a piece right through a flat. Although they are not motile, flats do sink deeper in the well whenever a flat below them dies, and so they can occupy several different rows over the course of their lives. As you might expect, if new flats are born faster than existing flats die, then eventually, the well is overrun and your game ends. Therefore, if you want to win at tetris, you must keep killing flats.

In order to keep track of flats, I give them unique tags which they bear throughout their lives. These are simply numbers, and I assign them to flats on the play in which

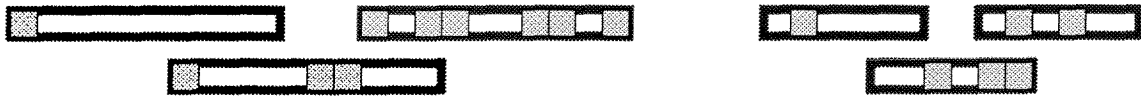


Figure 4.19: Specimens of flats. On the left, three flats taken from a well of width 10. On the right, three flats taken from a well of width 6. The thick-walled gray shell is not visible when flats are in their well.

they are born: the first flat born will be tagged “1”, the second, “2”, and so on. If several flats are born on the same play, I will tag them in order by position, starting with the lowest one. When a flat dies, its tag goes with it, never to be reused. If you look at the flats in a single state of the well, their tag numbers increase from the bottom of the well to its top. I will say that one flat is **older(younger)** than another when its tag is smaller (larger). This means that an older flat has been in the well for at least as many plays as a younger one. Don’t confuse a flat’s age with the number of full cells it has — they are not necessarily related. Figure 4.20 shows a few plays in the life of a population of flats.

4.2 Killing flats with squares

To demonstrate that my organic view of tetris is helpful, I will use it in proving that no winning strategy exists for a variant of the game in which no lookahead is provided and in which you use only two pieces: the usual two by two square, which I’ll denote \square_2 , and one new piece, a three by three square that I’ll call \square_3 . To keep the argument concrete, I’ll fix the width of the well at 6. Although it says nothing about standard tetris, I’m including this proof because it lets me introduce some concepts more easily than I can using only standard pieces.

Squares are simple pieces to analyze because they have only one orientation: hitting the rotate button does nothing. When you play a square, any flats you affect all grow

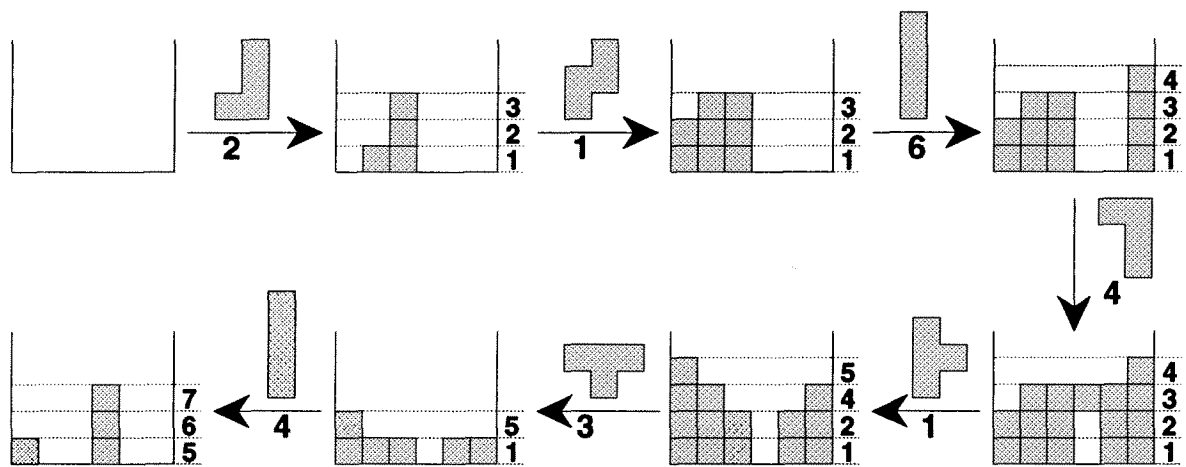


Figure 4.20: A changing population of flats in a well of width 6. For emphasis, dashed lines have been drawn between flats. Their tags are shown to their right, just outside the well. (As always, the number under a piece tells you in what column to play its leftmost cell.) Play by play: (i) flats 1, 2, and 3 are born. (ii) flats 1, 2, and 3 grow. (iii) flat 4 is born and flats 1, 2, and 3 grow. (iv) flats 1, 2, and 3 grow. (v) flat 5 is born, flat 3 dies, and flat 4 grows and sinks. (vi) flats 2 and 4 die, and flat 5 sinks (vii) flats 6 and 7 are born, flat 1 dies, and flat 5 grows and sinks.

by the same number of cells. For example, \square_3 always adds three cells to the flats you play it into. Therefore, if you want to fill and kill a flat using a single \square_2 (or \square_3), that flat must have exactly two (or three) adjacent empty cells. I call these flats \square_2 -targets (or \square_3 -targets), and have drawn some in Figure 4.21. So, \square_2 -targets are the only flats which you can kill with a single \square_2 , and \square_3 -targets are the only ones you can kill with a single \square_3 .

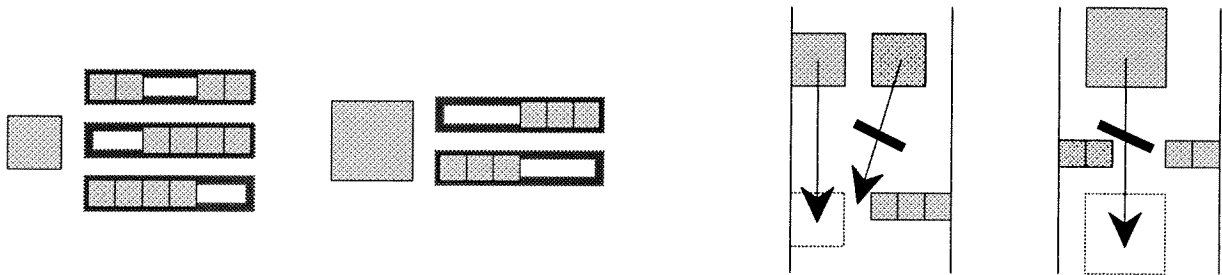


Figure 4.21: Left: \square_2 with some \square_2 -targets and \square_3 with some \square_3 -targets. Right: \square_2 can't kill a \square_3 -target and \square_3 can't move through a \square_2 -target. A bar through an arrow indicates an impossible play.

Because its empty space is only two cells wide, a \square_2 -target won't let a \square_3 through it. Therefore, a \square_2 -target **protects** all the \square_3 -targets below it. If you make a \square_3 -target grow by using a \square_2 , it will no longer be a \square_3 -target, but you still won't be able to kill it with \square_2 , since it will have only one empty cell. Therefore, once a flat has been a \square_3 -target, it can never become a \square_2 -target (see Figure 4.21 again).

How can I show that you have no winning strategy for (variant) tetris using only \square_2 and \square_3 in a well of width 6? One way is for me to play machine's advocate: each time you make a play, I'll look at the well and help the machine to select the next piece to give you (you aren't getting lookahead information, so we can wait until the end of your play before deciding what piece to send next). Since you must keep killing flats in order

to win, we will try to defeat you by handing you harmless pieces.

4.3 A recipe for your defeat

You start with the empty well, and we hand you a \square_3 . You might play it in columns 1, 2, and 3 (or 4, 5, and 6), thereby creating three \square_3 -targets. If so, it would be stupid for us to hand you another \square_3 right away, since you could just use it to kill those \square_3 -targets. On the other hand, if you played the \square_3 without creating any \square_3 -targets, then we may as well give you another \square_3 since you can't kill any flats with it. Thus, we'll keep giving you \square_3 to play until you create some \square_3 -targets. You must eventually do this, otherwise you'll never kill a single flat, they will overrun the well, and your game will end.

At this point, there is at least one \square_3 -target, which I'll call f , in the well, so we'll start sending you \square_2 , with which you can never kill it. (In fact, you'll soon see that you will never be able to kill it with \square_3 either — f is **immortal!**) In playing these \square_2 , you must eventually create some \square_2 -targets, or (again) you'll never kill another flat. We won't stop sending you \square_2 just yet, though. Instead, we'll wait until you create a \square_2 -target somewhere above flat f . (We know you'll do this because eventually, the \square_2 -targets which you are creating and killing must be younger than f , and so live higher than f in the well.) Notice that this \square_2 -target protects f , and any present or past \square_3 -targets below f , from \square_3 . So next, we'll hand you these harmless \square_3 until you create another \square_3 -target. Then we'll hand you \square_2 until that \square_3 -target is in turn protected by a higher \square_2 -target, and so on (see Figure 4.22).

As the machine and I continue this process indefinitely, the population of immortal \square_3 -targets will increase beyond any bound and overrun the well, no matter how deep it was to start with. This ends your game in defeat. We didn't have to start with the empty well: any initial population of flats would have led to the same outcome. Moreover, the

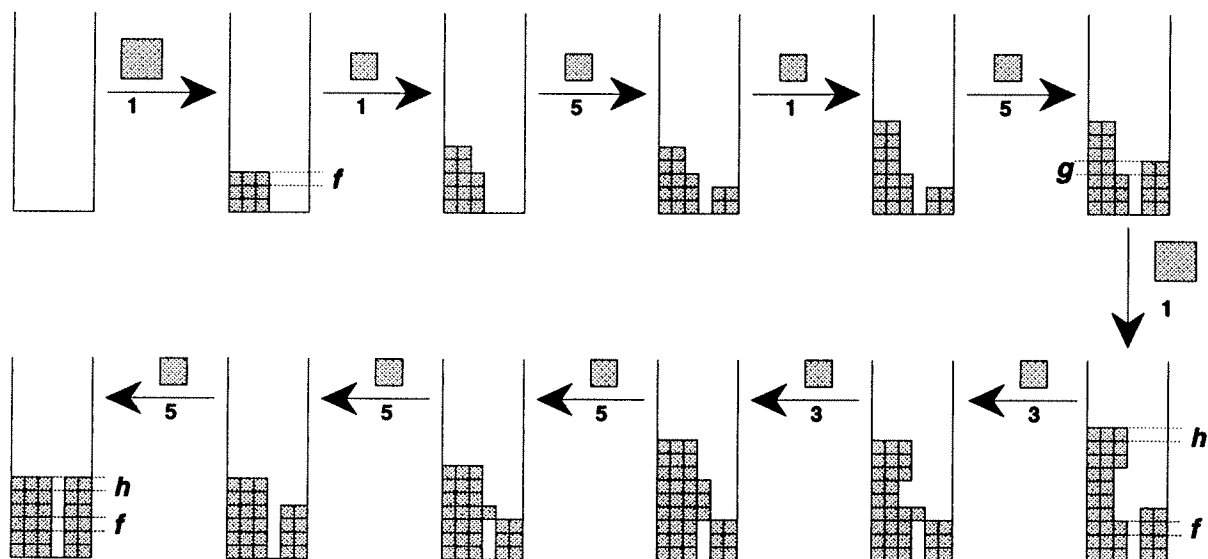


Figure 4.22: A sequence of plays you might make. You create flat f , a \square_3 -target, in the first play, so we then give you \square_2 until you create the \square_2 -target g above f (notice that g protects f from the subsequent \square_3). Next, we send \square_3 until you create more \square_3 -targets, one of which I've labelled h . In the subsequent sequence of \square_2 , you manage to kill several flats, but f and h still live.

choice of 2 and 3 for square sizes, and the choice of 6 for the well width are not essential to the proof. If we use squares with sizes a and b (with $a < b$), then we can defeat you as long as b is not a multiple of a .

“That’s not fair”, you complain, “you’re helping the machine!” True, but the help I’ve given can easily be mechanized to make what I’ll call a **defeating algorithm**: a recipe for your defeat at tetris (or one of its variants) which the machine can follow. The defeat I described above is summarized as:

Algorithm 1 *If you play (variant) tetris using only \square_2 and \square_3 as pieces and with no lookahead information, the machine can defeat you by following this recipe:*

1. Send \square_3 until there is an unprotected \square_3 -target in the well.
2. Send \square_2 until a \square_2 -target protects all \square_3 -targets in the well.
3. Go to step 1.

To summarize the previous discussion, this algorithm leads to your defeat because all the \square_3 -targets detected in step 1 are immortal: they can’t be killed by \square_2 , and while you play \square_3 , they are protected from above by a \square_2 -target. Notice that the machine can easily determine whether or not there are unprotected \square_3 -targets in the well: in order, starting with the highest flat in the well, it examines one flat at a time until it finds either a \square_3 -target or a flat (such as a \square_2 -target) whose largest hole is less than 3 cells wide. If the former, it has found an unprotected \square_3 -target, otherwise, any \square_3 -targets in the well are protected by the latter.

You might still complain that in real TETRIS, the machine seems to send pieces at random, and so this idea of a recipe is irrelevant. Your objection is a strong one, but for now, my answer to it is simply that even if the machine does send pieces at random, there is a chance that this random sequence will be exactly the same (finite) one that

the machine would have generated by following the recipe. I'll return to this matter in Chapter 6.

4.4 Published recipes and two minor fallacies

I can think of two ways in which you might (and in which I did) misinterpret the result of the last section. Both of them are based on the following observation: "I know the recipe the machine will use to defeat me, so I can work out in advance exactly what sequence of pieces I will be given." You might then conclude:

Fallacy 1 *Since I know what sequence of pieces I will get, I actually do have lookahead information, whereas your proof of my defeat assumed I had none. Therefore, you haven't proven that I will be defeated once I know the recipe.*

or

Fallacy 2 *Since I can always predict what piece I'll get next, you've actually proven that the machine can defeat me even if it gives me lookahead information, since that information doesn't tell me anything I don't already know.*

Both of these false conclusions result from a subtle misunderstanding of lookahead. To say that you have one piece lookahead (as you do in TETRIS) means not only that the machine must tell you in advance what piece you're getting next; it also means that the machine must decide what piece that will be *before* it sees how you play the current one. Lookahead is a promise on the part of the machine to hand you a certain piece next, regardless of how you play the current one. Since the defeating algorithm above doesn't decide what piece it will send you next until after you've played the current one, it doesn't actually provide lookahead and neither do your predictions — the pre-commitment just isn't there.

4.5 The cycle-band

When you use a winning strategy, you'll eventually find that you're seeing the same well states over and over. This is because there is only a finite number of well states with no full cells above row 20. Since each of the $10 \times 20 = 200$ cells in or below row 20 is either full or empty, there can be at most $2 \times 2 \times \dots \times 2 = 2^{200}$ different states in a winning strategy. (This is a slight over-estimate, since no full rows and no empty rows below non-empty rows are allowed.) For each of these possible states, the machine can be displaying as lookahead any of the seven pieces. Therefore, there is still only a finite number of different **situations** (i.e. combinations of well state and lookahead piece) which you can encounter when using a winning strategy. (The same is true for a non-winning strategy, but simply because your game ends after a finite number of plays). If you continue your game indefinitely, you will eventually have made more plays than there are situations in the strategy, and so some of those plays must have been made in the same situation. A **cycle** is a sequence of plays which starts and ends with the same situation. For example, all of the single-piece strategies in Chapter 2 eventually end up in a cycle. Notice that if a sequence of pieces induces a cycle, and if the machine repeats that sequence of pieces, you will, if you are following a strategy, be forced to repeat the cycle.

In a single pass through a cycle, some flats are born, some grow, some sink, and some die. I call this collection of flats the **cycle-band** corresponding to that cycle. Some flats are born into the cycle-band, others are there from the start of the cycle, and some die before the cycle ends, as I've shown in Figure 4.23. If a sequence of pieces leads you through a cycle, then the machine can force you to pass through that cycle endlessly, simply by repeating the sequence of pieces over and over again. Each of these passes has a cycle-band associated with it, and I call the collection of all the flats in these cycle-bands

the **infinite cycle-band**. In other words, the infinite cycle-band is the set of all flats which are born, grow, sink, or die when a cycle is repeated *ad infinitum*. I now prove some useful facts about these objects.

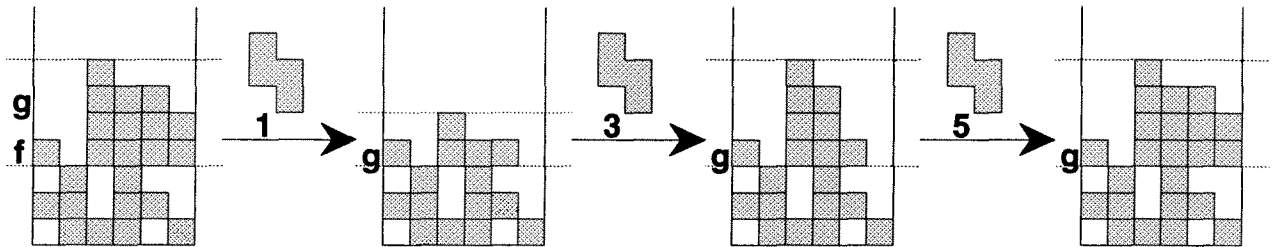


Figure 4.23: A cycle induced by left kinks. One pass through the cycle consists of three plays, and restores the well to its initial state. Even so, the identity of the flats changes: for example, flat f is the oldest flat in the cycle-band, and dies on the first play of the cycle. On the same play, the flat g grows and sinks into the row formerly occupied by f . On the last play of the cycle, g grows again so that its structure is the same as f 's was in the initial state. In each state, the flats in the cycle-band are those between the dashed lines.

Proposition 1 *If f is a flat in the cycle-band which doesn't sink in the cycle, then it must die there. Moreover, the oldest flat in the cycle-band dies during the cycle.*

Proof: Since f doesn't sink, it must occupy the same row (call it r) throughout the cycle, or until it dies. Suppose f is born or grows during the cycle (if neither, then f must die during the cycle in order to satisfy the criterion for membership in the cycle-band). On a play in which this happens, the number of full cells in row r increases. But this number must decrease before the end of the cycle so that the well can return to the cycle's initial state, in which r had fewer full cells. The only way this can happen is for f to die (and to be replaced by a flat with fewer full cells, or by an empty row). The oldest flat in the cycle-band is also the lowest there, and so it can't sink. (Otherwise, it

would have to sink into a row just vacated by another flat, and the latter would thus be in the cycle-band, and lower than the oldest such flat. This is a contradiction.) ■

Proposition 2 *If f is a flat in the infinite cycle-band, then f eventually dies.*

Proof: The last time f sinks, it moves into a row just vacated by another flat. (If f never sinks, then it must die.) That other flat must have died, rather than sunk, since otherwise f would sink again when this play was repeated in the next pass through the cycle. Thus, f itself will die on the next pass through the cycle. ■

Proposition 3 *If g is a flat which lives in a state of the cycle, and g is younger than the oldest flat in the cycle-band, then g is also in the cycle-band.*

Proof: Let f be the oldest flat in the cycle-band. If g is in the well when f dies, then g sinks (it's younger and so lives above f), and thus is in the cycle-band. On the other hand, if g is born in the cycle only after f dies, or if g dies during the cycle before f does, then, by definition, g is in the cycle-band. ■

You might wonder why I chose the term "cycle-band". This is explained by:

Proposition 4 *In any state of the cycle, those flats which are in the cycle-band form a contiguous band at the top of the state.*

Proof: Let g and h be flats in a state of the cycle. Suppose that g is in the cycle-band and that h lives above g . Since g is no older than f (the oldest flat in the cycle-band), and since h is younger than g , therefore h is younger than f , and so, by the previous proposition, h is in the cycle-band. ■

In summary, if you are using a winning strategy, you will eventually pass through a cycle. The flats which participate in that cycle will form a band at the top of each state of the cycle. You'll soon see why this matters.

Chapter 5

Kinky Sets

Now that I've told you what cycle-bands and defeating algorithms are, I'm ready to show you why there is no winning strategy for standard tetris. In fact, I'll prove that as long as you are playing with a set of pieces that includes both kinks, you have no winning strategy in any well. Before giving a recipe for your defeat, I'll use the next two sections to establish facts which I need in order to prove that the recipe actually works.

5.1 Playing in the cycle-band

You might recall that the well can be divided into pairs of adjacent columns that I call lanes. In each lane, a flat can have zero, one, or two full cells, and I'll use **segment** to refer to the portion of a flat that occupies one lane. Moreover, I'll use the symbols $\square\square$, $\square\blacksquare$, $\blacksquare\square$, and $\blacksquare\blacksquare$ to represent the four possible configurations of empty and full cells in a segment.

If a flat has an $\square\blacksquare$ as its leftmost segment, then you can't kill that flat with a left kink (see Figure 5.24). I'll call such a flat **left-immune**. Similarly, a flat with a $\blacksquare\square$ as its rightmost segment is **right-immune**. I now show that if the machine gives you a long enough sequence of left kinks, and you play them without ending your game, then the well will develop a certain kind of structure.

Lemma 2 *While following a strategy, if you pass through a cycle in which every piece is a left kink, then you must have played each one entirely within a single lane. Moreover,*

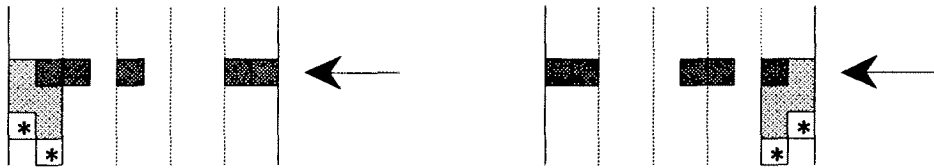


Figure 5.24: Arrows indicate left- and right-immune flats. The plays shown are impossible since they require the pieces to be moved upward. Even if I allow you upward moves, the \square^* cells will have to be empty for you to slide the pieces into place, and gravity will then pull the kinks down into these empty cells. You still can't kill the immune flats.

for every lane in every state of the cycle, the portion of that lane within the cycle-band will consist of a stack of zero or more $\blacksquare\blacksquare$ segments, topped with a single $\blacksquare\square$.

Proof: To begin, I will define a sequence of statements which might describe your plays during the cycle. Each of these statements tells how your plays affected one lane, and can be either true or false. The statements are called $P(1)$, $P(2)$, and so on, where the numbers represent which lane the statement is about. The statements have the following meaning:

For a lane x , $P(x)$ says that whenever you played a left kink so as to fill at least one cell to the left of lane x , you played that kink entirely within a single lane. This means, among other things, that during the cycle, you never played a left kink so that it filled cells in both lane x in the lane to the left of x (see Figure 5.25).

I repeat: each of $P(1)$, $P(2)$, and so on could be either a true or a false statement about how you played the kinks during the cycle. I want to show that, in fact, all of them are true.

What does $P(1)$ really mean? It means that during the cycle, any kink you played which filled some cells to the left of lane 1 was played entirely within a single lane. This

is true, in a trivial way, simply because there are no cells to the left of lane 1, and so you couldn't have filled any. More importantly, this means that you never played a kink which stuck partly into lane 1 from the left (again, this just isn't possible).

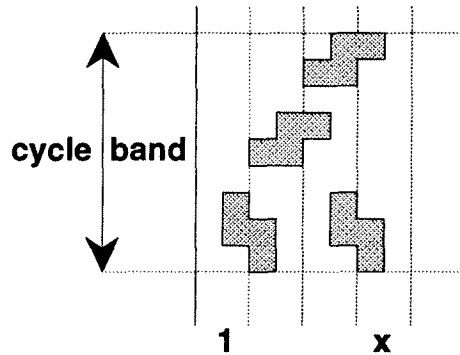


Figure 5.25: $P(x)$ is true if and only if plays of the type illustrated *don't* occur: lane boundaries to the left of lane x , including x 's own left boundary, are not crossed by pieces played in the cycle. $P(1)$ is true simply because no piece can cross the left wall of the well.

Now that you know $P(1)$ is true, I will use that fact to prove that $P(2)$ is true. Even better, I will prove that if for some lane n , $P(n)$ is true, then $P(n+1)$ (i.e. the statement for the next lane to the right) is also true. This will mean $P(2)$ is true, and thus that $P(3)$ is true, and so on, all the way across the well.

Let n be a lane for which $P(n)$ is true. I first prove that no flat in the cycle-band can have an $\square \blacksquare$ segment in lane n . Suppose, to the contrary, that f is the oldest flat (in the cycle-band) that has an $\square \blacksquare$ in lane n . If you fill the empty cell of that segment during the cycle, it can only be by making the play shown in Figure 5.26, because $P(n)$ implies no left kink can be played to fill the cell from the left. However, this play creates another $\square \blacksquare$ in lane n , but in a flat, say g , which is lower (and hence older) than f . Moreover, since this play changes g , it is also in the cycle-band, contradicting my choice of f (see

Figure 5.26). Therefore, you can't fill the empty cell in that segment of f , and so even if the cycle is endlessly repeated, you won't kill f . But this contradicts the fact that f is in the infinite cycle-band, since every flat therein must eventually die. Therefore, neither f nor any other flat in the cycle-band can have a $\square \blacksquare$ segment in lane n .

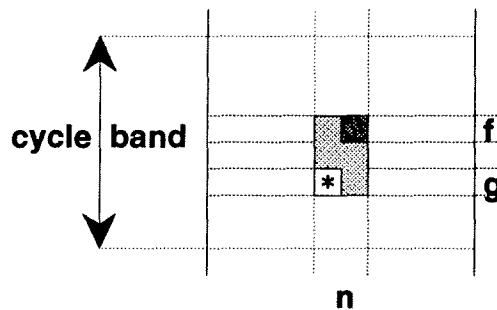


Figure 5.26: The only way you can fill the empty cell of an $\square \blacksquare$ segment in lane n of flat f , if $P(n)$ is true. Doing so requires that the $\square *$ cell be empty: think of where the kink could have been just before ending up as drawn. Thus, if you play the kink this way, you create a new $\square \blacksquare$ segment in a flat g that's below f .

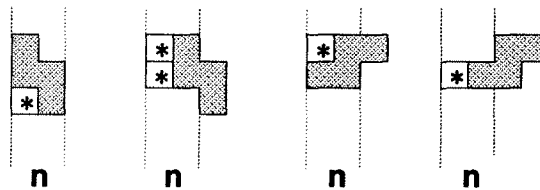


Figure 5.27: When $P(n)$ is true, there are at most four ways you can play a left kink so as to fill a cell in lane n . In each case, the $\square *$ cells must be full or else you will create an $\square \blacksquare$ in lane n .

In Figure 5.27 I've shown the four possible ways you could fill a cell in lane n (when $P(n)$ is true, as I've assumed). In order not to create the forbidden $\square \blacksquare$ in lane n , the

$\boxed{*}$ cells must already have been full when you played the kink. But that means the segment containing that cell must have been a $\boxed{\blacksquare}\boxed{}$ (consisting of the full $\boxed{*}$ cell and its empty right neighbour) before you played the kink. Therefore, if you filled a cell in lane n , there must already have been a $\boxed{\blacksquare}\boxed{}$ there. But notice that the last three ways of playing a left kink change at least one $\boxed{\blacksquare}\boxed{}$ into a $\boxed{\blacksquare}\boxed{\blacksquare}$ and so decrease the number of $\boxed{\blacksquare}\boxed{}$ in lane n by at least one. In contrast, suppose you played the left kink in the first way I've shown (i.e. entirely within lane n). You would have destroyed one $\boxed{\blacksquare}\boxed{}$ segment (the one with the $\boxed{*}$), but created another (the one at the top of the piece; its right cell is empty since otherwise, that segment would have been the forbidden $\boxed{}\boxed{\blacksquare}$ before you played the piece). Thus the number of $\boxed{\blacksquare}\boxed{}$ segments in lane n of the cycle-band could only have declined or remained constant each time you played a left kink. In fact, it must have remained constant since the initial and final states of the cycle are identical. Therefore, during the cycle, you could have used only the first method to play a left kink when filling a cell in lane n . Hence, during the cycle, you never played a left kink that filled cells in both lane n and lane $n + 1$. Together with $P(n)$ being true, this is enough to imply that $P(n + 1)$ is true. Therefore, all of the statements $P(1)$, $P(2)$, and so on are true: during the cycle, you played every left kink entirely within a single lane.

The second part of the lemma is a consequence of the first. Notice that during the cycle, you must have played each left kink at the *top* of a lane: playing it below the top would have required an $\boxed{}\boxed{}$ segment below the non-empty top segment of the lane. But how could this empty segment have arisen? During the cycle, gravity would prevent you from leaving any empty segments in a lane, and so in infinite repetition of the cycle, all such empty segments would eventually disappear. But then the play which required the empty segment could never be repeated. Therefore, such a play doesn't occur in the cycle.

Playing left kinks within and at the top of lanes can give rise to lane structures only of the type shown in Figure 5.28. ■

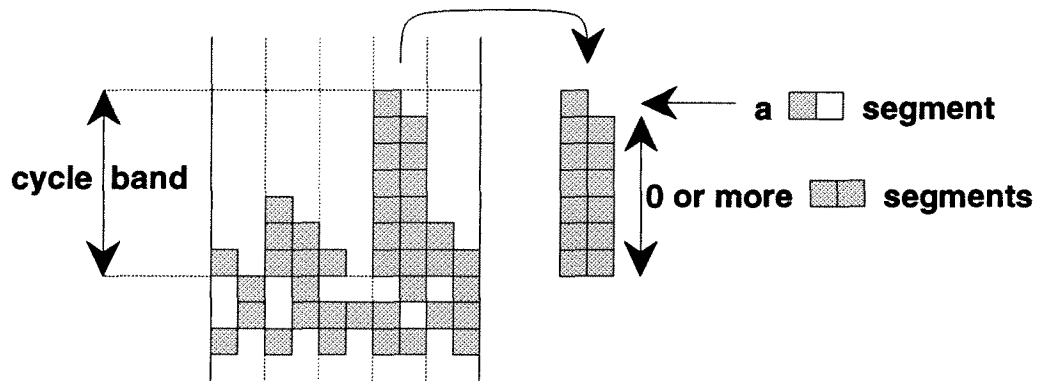


Figure 5.28: During a left kink cycle, the lanes in the cycle band all have a simple structure: a $\blacksquare\blacksquare$ segment on top of a (possibly empty) stack of $\blacksquare\blacksquare$.

5.2 The shield in the cycle-band

The importance of this kind of cycle-band structure is that it contains a collection of flats which together protect underlying flats from right kinks, and which no number of right kinks can kill. This is what I call a **shield** against right kinks.

Lemma 3 *In a cycle induced by a sequence of left kinks, the cycle-band contains a shield against right-kinks.*

Proof: Look at any state in the cycle. Let f be the highest flat in that state whose rightmost segment is not an $\blacksquare\blacksquare$. The previous lemma shows that this segment must be a $\blacksquare\blacksquare$, and so f is right-immune. This is the highest flat in the shield, and you can find the other ones by this procedure:

1. f is a flat in the cycle-band which can't be deleted by right kinks, and whose rightmost segment is not an $\square\square$.
2. If none of f 's segments is an $\square\square$ (for example, if f is the lowest flat in the cycle-band for this state) then each of f 's segments is either a $\blacksquare\square$ or a $\blacksquare\blacksquare$. But then f has no pair of consecutive empty cells at all, and so it prevents right kinks from reaching any rows below (kinks need a space of two empty cells to pass through a row). This flat is the lowest one in the shield, so you can stop.
3. **Otherwise** one of f 's segments is an $\square\square$. Let l be the lane with the rightmost such segment, and notice that f has a full cell in the left column of lane $l+1$ — see Figure 5.29. (Notice that l is not the rightmost lane in the well, since f 's rightmost segment isn't an $\square\square$.)
4. Put your hand on this full cell and descend lane l , dragging your hand down along the “wall” of full cells to your immediate right, until you reach a flat, say g , whose segment in lane l is not an $\square\square$. Flat g is still in the cycle-band since the bottom flat of the cycle band, having no $\square\square$ segments (by the previous lemma), would have stopped your descent. Thus, g must have a $\blacksquare\square$ segment in lane l (previous lemma again). Notice that the empty cell of this segment can't be filled by a right kink until all the full cells you touched are gone (Figure 5.29 again). That is, flat g can't be killed (using only right kinks) until you've killed all of the flats above g , up to and including f . Since f can't be killed by right kinks, neither can g .
5. Reuse the labels so that “ f ” now refers to this flat g .
6. Go to step 1.

Since you descend at least one row each time you perform step 4, following this procedure will eventually lead you to stop, successfully, in step 3. At that point, the set of all flats

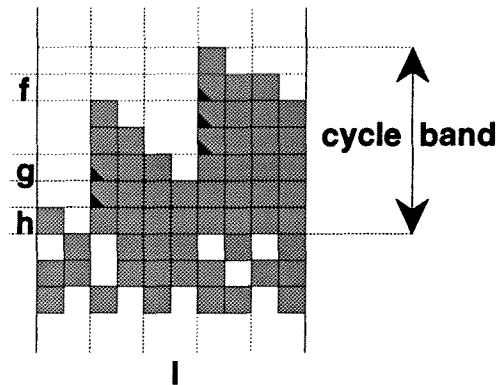


Figure 5.29: The flat f is right-immune, but has an $\square\square$ in lane l . The cells you touch while dropping down empty portions of lanes are shown notched. In this example, the shield you eventually find consists of flats f , g , and h .

you ever called f forms a shield against right kinks, since you can't kill any of these flats with right kinks, and since the lowest one actually prevents you from playing right kinks anywhere below. ■

5.3 Defeating you with kinks

When the machine performs the defeating algorithm for \square_2 and \square_3 which I gave in Chapter 4, it does not provide you with lookahead information. However, when it performs the following algorithm, it does. You can verify this by making sure that in every step, the piece being handed to you is the one displayed (as lookahead piece) in the previous step.

Algorithm 2 *The following procedure generates a sequence of kinks which defeats any strategy for tetris:*

1. Send left kink and display left kink until a cycle is detected.

2. Send one left kink and display right kink.
3. Send right kink and display right kink until a cycle is detected.
4. Send one right kink and display left kink.
5. Go to step 1.

Proof: For reasons I've already mentioned, if your game doesn't end in step 1 or 3, you must eventually pass through a cycle. The machine will detect this when you revisit a state within step 1 or 3. Lemmas 2 and 3 show that when a cycle is detected in step 1, the cycle-band will contain a shield against right kinks. Can you disable this shield when you play the left kink sent in step 2? No: the shield consists of flats having structures of the type illustrated in Figure 5.30. The only way you can fill the empty cell in one of those structures (and hence the only possible way you can disable the shield) with the single left kink of step 2 is to play that kink within a lane. But doing so preserves the structural form of the cycle-band: it will contain a (possibly new) shield against right kinks.

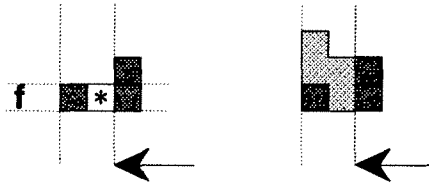


Figure 5.30: Every flat, say f , in the shield must have an empty cell, shown as a $\boxed{*}$, with a full cell in the position one up and one to the right. The only way you can fill this empty cell using a single left kink is shown. (In both cases, the lane boundary indicated with an arrow might actually be the right wall of the well.)

So, even after step 2, the well still has a shield against right kinks. Therefore, nothing you do with the right kinks you get in steps 3 and 4 will kill any of the flats in that shield.

Also, notice that the cycle-band of the cycle detected in step 3 must lie entirely above the shield from steps 1 and 2. This is simply because the top flat of that shield can't be killed by right kinks, and so isn't in the infinite cycle-band corresponding to repetition of the right-kink cycle. (Remember, every flat in the infinite cycle-band must die.) But now the mirror images of Lemmas 2 and 3 show that the cycle-band induced by right kinks will contain a shield against left kinks. Since this new cycle-band is above the old shield, so is the new shield. Thus, the second time the machine follows step 1, you will be playing left kinks above this new shield, without killing any of the flats in it. As the machine repeats steps 1 to 4, you will be forced to create a stack of shields of alternating type until eventually, one of the flats in those shields is above row 20, and your game ends. Obviously, you could have chosen any other maximum well depth and the machine would still have (eventually) defeated you. ■

When following Algorithm 2, how can the machine detect a cycle? One way (popular with the American recreational vehicle crowd) is for it to create a list of states visited. Each time you enter step 1 or 2, the machine empties the list. Every time you reach a state within that step, it checks to see if it's already in the list. If so, you have just passed through a cycle. If not, it adds the state to its list. It's quite possible for this list to get arbitrarily large, even if a cycle is always (eventually) detected, so a better method is for the machine to wait for the cycle-band structure itself. This structure can arise even if you haven't yet completed a cycle. Moreover, the machine requires no list for this, but only needs to scan each state you visit to see if it has the required structure.

"Now that I know how the machine is defeating me, can't I choose to avoid creating the shields?" No. Remember that with my concept of "strategy", the way you play a piece can depend only on the current state of the well and on which piece is shown as lookahead. Therefore, if you revisit a situation, you have no choice but to play the current piece in the same way you did when you first encountered that situation. This is

enough to guarantee that if you are given a long enough sequence of left kinks, you must eventually pass through a cycle (or end your game). Once you've done this, Lemma 3 guarantees that there will be a shield in the corresponding cycle-band.

I repeat: no matter what the width of the well, no matter how great you set its depth, no matter what state the well is initially in, and even if you are allowed to move pieces up before dropping them, you can't beat tetris. The machine can defeat you by using just the two kinks. You can't win at tetris, and so you can't win at TETRIS.

Chapter 6

Conclusion

I now return to the real world to see what, if anything, the analysis of the previous chapters has to say about it.

6.1 Pessimism vs. Realism

If I had given you a winning strategy for tetris, you could have used it to make a real game of TETRIS last as long as you wished, provided the game didn't get too fast for you. You would have been able to beat TETRIS no matter how cleverly it was programmed to defeat you. In fact, I've shown that you don't have a winning strategy for tetris, because if it were programmed to do so, the machine could send you a sequence of kinks which would quickly end your game. But what does this say about TETRIS?

I have no evidence to contradict the hypothesis that the TETRIS machine simply flips a seven-faced coin to decide which piece it will hand you — TETRIS doesn't appear programmed to act as an opponent. In a typical game, the random sequence of pieces you receive will probably allow you to play for much longer than you could if the pieces were carefully chosen. If I call this the **average case** scenario, then my concept of strategy assumes a **worst case** scenario: the machine tries its best to defeat you, turning the game into a contest between two players. (You can actually play tetris against a human opponent using a pencil and a piece of paper; delete rows by folding them out of existence. If the player sending the pieces has read this thesis, the game won't last long!)

I suspect it would be more difficult to come up with any exact results in the average

case scenario. However, I'd still like some indication of how my worst-case results relate to reality. One question I find interesting is whether the seven pieces can be ranked according to how difficult, on average, they are to play. Instead of analyzing the average case mathematically, I've decided to bring in some expert opinion.

6.2 The player survey

For two weeks in February, 1992, I conducted a passive survey of TETRIS players. I put a total of 100 survey forms (see Figure 6.31) into envelopes stuck to three TETRIS machines on the UBC campus (one in each of Gage and Vanier residences, and one in the SUB video arcade). Players were informed about the survey by the instruction sheet shown in Figure 6.32. Twenty-four forms were returned, and I have summarized the results in Table 6.4.

Table 6.3: The average rank of TETRIS pieces, where smaller means harder. The sample size is 24, and the standard error equals the standard deviation divided by $\sqrt{24}$.

	Right Kink	Left Kink	Right Elbow	Left Elbow	Tee	Square	Bar
Avg. Rank	0.75	0.96	2.13	2.54	3.92	3.96	4.67
Std. Dev.	1.01	1.40	1.13	1.22	2.02	1.65	1.95
Std. Err.	0.21	0.29	0.23	0.25	0.41	0.34	0.40

In Table 6.3, I've listed the "average" rank for each piece. Each player was asked to rank the pieces, with 1 for the hardest, higher numbers for easier pieces, and with equally difficult pieces getting the same rank. I counted, for each piece and player, how many pieces that player ranked as being less difficult to play than that piece. The averages and standard deviations in the table are for this transformed rank. For example, the average rank of 2.54 for left elbows indicates that, on average, players ranked 2.54 other pieces as being more difficult to play than left elbows.

The fact that the two kinks had the lowest average rank indicates they were generally perceived to be the hardest pieces to play. This is reminiscent of the results in Chapter 5. The square was ranked easier than all other pieces except the bar. This might remind you of the two-piece strategies in Chapter 3: if one of the pieces you are playing with is a square, then your winning strategy (if you have one) doesn't require lookahead. Elbows received an intermediate rank. This might reflect the fact that winning strategies for sets of pieces with an elbow require lookahead, unless the other piece is a square. I know of no strategy for the two elbows which doesn't use lookahead. That makes elbows "more difficult" to play than the square, but "less difficult" than the kinks. These comparisons are vague, but I find them interesting nonetheless.

I posed questions 8 and 9 of the survey to see what factors players found most important in ending their games. One of the assumptions I made in formulating tetris was that you could take as long as you wanted to play the pieces. This affects the relevance of tetris to TETRIS: the more important are time factors, the less relevant is this thesis. For the game just played, 14 out of 24 players chose a time-related factor (i.e. "b", "c", or "h") as most important in ending their game, while only 6 chose a piece-related factor (i.e. "a" or "g"). (Other responses deal with factors incidental to the game.) As for what usually ended their games, 18 of 24 players indicated a time-dependent factor, while only 6 chose a piece-related factor. It probably doesn't surprise you that speed and timing play a major role in influencing the length of a game of TETRIS, and thus the score obtained therein.

The answers to the lookahead questions indicate that most players make use of this information at least often (19 of 24 players), with a slight trend towards greater usage among players with higher scores. Of the 19 frequent lookahead users, 5 thought two piece lookahead would not be helpful, 8 thought it would be somewhat helpful, and 5 thought it would be very helpful. In Chapter 3, I gave you winning strategies for some

pairs of pieces in which you had to use lookahead, but not very often. I know of no set of pieces for which there is a winning strategy with two- (or more) piece lookahead, but none for one-piece lookahead.

The most common piece of advice selected was “Try not to leave any holes” (8 of 24 players). Players’ own advice included “Stay calm”, “Leave a space by the wall for a tetris [i.e. a simultaneous clearing of four rows]”, “Don’t wait for that perfect piece”, and “Pretend you are having sex” (the latter proffered by player 3, who, judging by his/her score, is adept at following this advice).

Unfortunately, when I posed question 10, I was thinking in terms of tetris. In TETRIS, the best way to play a piece seems usually to depend on how many rows you have left to clear in the current round. I didn’t indicate this number in the six situations, and so it’s not clear that I can conclude anything from the responses. Moreover, I had intended each situation to pose a simple tradeoff (clearing a row vs. not leaving a hole, for example), but, for example, players indicated five different plays in the first situation, and six in the second.

6.3 Personal Conclusion

Even with knowledge of the results in this thesis, I can’t get a TETRIS score higher than about 100000. However, 6 of the 24 people in the survey can obtain much higher scores, and the typical high score on TETRIS machines at UBC is roughly 900000. Therefore, I have no choice but to conclude that, at least for TETRIS,








We do not learn from inference and deduction and the application of mathematics, but by direct intercourse and sympathy. [2]

Tetris Player Survey (to be completed right after a game)

1. What was your score (roughly)? _____ 2. At what level did you begin *this* game? **Easy Medium Hard**

3. At what level do you *usually* begin? **Easy Medium Hard**

4. Rank the Tetris pieces according to how difficult you *usually* find it to play them. Use 1 to indicate the hardest piece(s) to play, and use 2, 3, and so on to indicate easier pieces. If you find two or more pieces roughly equally difficult to play, give them the same number. Please assign a number to every piece.

Piece:       

Your ranking:
(1=hardest) _____















5. While you are playing a piece, Tetris shows you the next piece you'll have to play. How often do you use this information?
a) Never b) Seldom c) Often d) Usually

6. At low game speeds, how helpful would it be to see the next two pieces, rather than just the next one?
a) Not helpful b) Somewhat helpful c) Very helpful


7. If you wanted to help a beginning Tetris player, and could give only one simple piece of advice to that player, what would it be? (If you have advice you prefer to the ones listed, please circle f and write it in the space provided.)
a) Play pieces as low as possible. b) Try not to leave any holes. c) If you can clear a row, do it.
d) Watch to see which piece will come next. e) Play pieces of the same type close to one another.
f) (your own): _____


For questions 8 and 9, choose the factor that you feel contributed most to ending *this* game of Tetris, and the one which *usually* does so when you play. If you think a factor not listed here was more important, please circle i and write it in the space provided.


8. This game: 9. Usually: (Please circle only one letter in each column.)

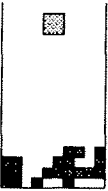
a)	a) I was waiting for a certain piece, and the machine didn't send it.
	please circle which one:       
b)	b) I was trying to play a piece in a certain spot, but it was dropping so fast I didn't have time to move it across to that spot.
c)	c) I accidentally played a piece where I didn't mean to.
d)	d) The joystick or rotate button malfunctioned.
e)	e) I got distracted by something outside of the game.
f)	f) I didn't feel like playing any longer.
g)	g) The machine was sending too many of a certain type of piece.
	please circle which one:       
h)	h) The game got so fast I couldn't keep track of what was happening.
i)	i) (your own): _____


10. Here are 6 situations you might face. Please draw the pieces where you would play it. (The next piece is shown in small.)














Thanks for your help. If you have comments/questions about Tetris or this survey, please put them on the back of this sheet.

Figure 6.31: The TETRIS player survey form.

The Tetris Player Survey: Why and How?

Why:

The purpose of this survey is to discover commonly held beliefs about Tetris. In particular, I'm interested in strategies people are using and in how different aspects of the game affect their scores.

This data is being collected as part of the research for my Master's thesis. (I'm a graduate student in the Institute of Applied Mathematics, here at UBC.) Survey results will appear in copies of the thesis at both Main and Math libraries.

The aim of the thesis is to investigate, mathematically, common knowledge about Tetris. The survey data will tell me both what beliefs (if any) people hold about the game, and what aspects of the game are essential and must be included in reasonable mathematical models of it. Such models will be used to explore questions about strategy like those on the other sign.

How

- * Play a game of Tetris
- * Fill out a survey form:
 - give the answers **you** feel are best
 - don't peek at other people's forms
 - if you're really unsure about what a question is asking for, leave it blank.
 - write any comments or questions you have about this survey, or Tetris, on the back of the page
- * Leave the completed form in the envelope provided.
- * Feel good about yourself for contributing to interesting research and for helping another student to graduate.

Thank-you for your help!

Figure 6.32: Instructions for TETRIS player survey participants.

Table 6.4: Results of the TETRIS player survey. Each row corresponds to answers provided by one player, in the same order as questions appear on the survey. Levels: easy, medium, hard. **Ranks**: Tee, Right Elbow, Left Kink, Square, Bar, Left Elbow, Right Kink. **Lookahead**: 1 is use of current lookahead, 2 is potential use of 2-piece lookahead. Reasons why game ends: **Now** is this game, **Most** is usually. The letters correspond to the choices on the survey form.

Player No.	Score	Levels	Ranks	Look		Advice	Game End	
				1	2		Now	Most
9		e,e	3,2,1,4,5,2,1	c	a	a	d	b
14		m,e	3,2,1,5,4,2,1	c	b	b	b	a
10		e,e	6,3,1,5,7,4,2	d	c	d	a	f
7		e,e	1,2,2,3,3,2,2	d	c	f	ah	ach
6			7,3,5,6,1,2,4	b	b	f		ac
2	16000	e,e	3,4,2,6,7,5,1	b	a	b	c	c
12	20000	e,e	3,1,2,4,5,1,2	d	b	b	b	b
22	20000	e,e	4,3,1,2,5,3,1	d	a	a	c	b
16	30171	e,e	6,4,1,3,7,5,2	b	b	c	eg	abcdfh
8	50000	e,e	7,5,2,6,3,4,1	c	b	c	bc	bcdh
4	59289	e,e	2,3,1,4,5,3,1	d	b	b	c	b
21	60000		6,2,1,4,5,3,2	cd	b	cf	c	c
18	75000	e,m	6,3,1,5,7,4,2	c	c	b	a	b
11	80978	e,e	5,2,1,3,4,2,1	c	a	d	b	b
13	93000	h,h	7,5,1,3,4,6,2	b	b	b	a	f
17	103979	h,h	4,3,2,1,5,3,2	d	b	c	c	cg
20	110000	e,e	7,2,5,6,3,4,1	b	c	a	h	h
24	122246	m,m	5,3,1,6,7,4,2	d	c	f	h	f
23	235000	e,e	3,4,2,6,7,5,1	d	a	b	c	c
1	287635	e,e	6,4,2,5,3,4,1	c	b	c	e	d
19	350000	e,e	1,1,1,1,1,1,1	d	c	f	d	h
5	400000	h,h	3,3,2,4,4,3,1	d	a	f	c	c
3	847088	e,e	1,3,2,4,5,3,2	d	b	f	a	gi
15	987000		4,1,6,3,7,2,5	c	b	b	e	c

Bibliography

- [1] David Gutman. *Prokofiev*. The Alderman Press, London, 1988.
- [2] Henry David Thoreau. *The Natural History Essays*. (Robert Sattelmeyer, ed.) Peregrine Smith Books, Salt Lake City, 1980.