Universiteit Leiden
The Netherlands

Opleiding Informatica

Aspects of Tectonic

Michiel W. F. van Spaendonck

Supervisors:
Walter A. Kosters and Jeannette de Graaf

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS)
www.liacs.leidenuniv.nl

30/06/2017

Abstract

In this paper we describe ways of solving the Japanese puzzle Tectonic. We look at different ways to accomplish this, and we want to find out which way of solving Tectonics is best for distinguishing difficulty levels for these Tectonics. To do this we look at different rules and translate the puzzles to the SAT-problem; with available SAT Solvers we try to solve Tectonics, while using a translation to 2-SAT for finding sub-solutions.

Contents

Chapter 1

Introduction

Tectonic is a little known puzzle among the Japanese puzzles, but it is getting more and more attention, with some puzzles in the newspapers and puzzle books with Tectonics being sold. Tectonic might very well be the next Sudoku, looking a bit like it, but having more possibilities and being more challenging.

Tectonics always have one solution, but they can have many different difficulty levels. For certain puzzles there has to be a minimum number of required starting numbers, which we call ”givens”, for it to have only one solution. These givens will be indicated by a small surrounding circle. The research question is: in which ways can we solve any Tectonics and can we use these ways to determine the difficulty level of any Tectonic?

In Chapter 2 we will define Tectonics and look at ways to solve a Tectonic, which will eventually help determining the desired difficulty levels. The ways we look at are the rule-based approach, rules extended with 2-SAT and the ”normal” SAT problem, which will be further explained in Chapter 6.
In Chapter 3 we will discuss the little related work that has been done and similar puzzles.
In Chapter 4 we will explain the rule-based strategy. Here we will define the rules and explain with an example how they work and how they can solve the puzzle.
In Chapter 5 we will explain 2-SAT, which will be used when the rule-based strategy does not suffice to solve puzzles. If 2-SAT is needed that means the rules cannot solve the puzzle. This means that the difficulty level is high, so 2-SAT will also be a way of determining the eventual difficulty level.
In Chapter 6 we explain SAT Solvers: we explain the way that they work, how we use them to find a solution and how these help to eventually determine a difficulty level.
In Chapter 7 we highlight some additional work we did, which did not get as much attention as the rest of the thesis subjects.
In Chapter 8 we look at the results from the created ways of solving Tectonics and we define some difficulty levels, based on these solving techniques.
In Chapter 9 we look back and draw a conclusion.

We did this research for a bachelor thesis at Leiden University, with the help of my supervisors Walter Kosters and Jeannette de Graaf. For this research we used LIACS computers and servers.

Chapter 2

Tectonic

The Japanese puzzle TECTONIC, also known as Suguru or Nanba Burokku, is a puzzle where the goal is simple: fill every cell in a diagram, divided into different regions, with a number. Of course there are rules that determine which cell can contain which number. The puzzle was invented by Naoki Inaba, and occurs in a wide variety of difficulties, of which there are usually nine.

2.1 Definition

Each TECTONIC is a rectangle divided into squares, called cells. This means that a TECTONIC is not necessarily a square itself, like a SUDOKU, although it might look like a SUDOKU. TECTONICS are also divided into groups of cells, which we call regions. But unlike a SUDOKU, these regions do not have a set place and their size is different, because in TECTONICS each region's size is one of the numbers 1 to 5, never more. Each region then also has to contain the numbers 1 to the region's size. So a region of size 3 contains the numbers 1, 2 and 3. This means that two cells in the same region cannot have the same value. This is also true for adjacent cells, which are what we call neighbours, either vertically, horizontally or diagonally. The TECTONIC also does not have a set width or height, which means a one by one TECTONIC, or a cell, is also a valid TECTONICS as well as a 9 by 11 TECTONIC, for instance.

For indicating the cells in any TECTONIC, we use coordinates (X,Y). Here X is zero at the left and increases when we move to the right and Y is zero at the top and increases when we move to the bottom. So (0, 0) is the top-left cell of any TECTONIC.

2.2 Rules

TECTONICS can be solved in different ways. One such way is to solve the TECTONIC by following certain rules. These rules can find many, or even all of the numbers in the solution. The first rules are the restrictions of the Tectonic: no adjacent cells or cells in the same region can have the same value and the numbers 1 to the region's size have to be given in a region. These rules form the basis for all the other rules. E.g., a rule which we can conceive is that when all the cells of a region with four cells are neighbours of another cell, that cell has to have the value 5, which means that its region's size also has to be 5. This is of course because all the cells in the firstly mentioned region contain the numbers 1 to 4 and because they are all neighbours of one cell. So that cell cannot have the values 1 to 4, thus only 5 remains. Another, simpler, rule is that a region of size 1 can only contain a 1. Aside from these simple rules there are also more complex rules conceivable, which will be discussed in the rules section. In the left puzzle in Figure 2.1 we give a 5 by 5 Tectonic, where each cell with a specific color belongs to the same region. in Figure 2.1 we see on the left the starting puzzle, in the middle an intermediate solution, where some rules are applied once, and on the right the final solution.

①			⑤	
①		②		④
	③			

①			⑤	
	5			
①		②		④
2				
	③			1

①	3	2	⑤	1
4	5	4	3	2
①	3	②	1	④
2	5	4	5	3
4	③	1	2	1

Figure 2.1: See [Bum]. Published by KrazyDad.com. ©2005-2017.

2.3 Satisfiability

Every Tectonic — with the appropriate givens — should have only one solution and is therefore satisfiable. This also means that there are puzzles without enough givens, which therefore have multiple solutions. For instance the Tectonic in Figure 2.2a has 2 solutions, which are shown in Figure 2.2b and Figure 2.2c. There are cases where the fact that the puzzle has only one solution can be used as an aid to find the values of cells which have to be a certain value, if the puzzle really only has one solution. However, this is dangerous when you are not sure of this fact, and on top of that it is usually a short-cut we don't have to take to come to the final solution. Therefore we will not make use of this assumption in any of our rules or other solution-finding techniques.

(a) A Tectonic with 2 solutions.

1	2

(b) The first solution.

2	1

(c) The second solution.

Figure 2.2

As mentioned before, it can happen that we come across a Tectonic which is not solvable by mere simple rules alone. When this happens we need to start looking towards other valid ways of finding the solution. To do this we used the rule-based technique to find what appears to be a sub-solution, and use 2-SAT to find the missing values. Another way is to use the satisfiability problem and find a solution by converting the puzzle to an instance of the SAT problem and use a SAT-solver to find a possible — and in most cases the only — solution.

Both 2-SAT and SAT will be further explained in Chapter 5 and Chapter 6 respectively.

Chapter 3

Related Work

Puzzles like TECTONIC are:

1. SUDOKU: Just like with the TECTONIC, the numbers in the same region cannot be the same. Each number also forbids its row and column of having the same number in one of its cells in SUDOKU, but not the diagonals. In TECTONICS this is not true for the same row or column, but only if the cells are neighbours, (or if the cells are in the same region,) which also includes the diagonals. For more information see [Sud].

2. RIPPLE EFFECT: This puzzle differs from TECTONIC only because numbers can be adjacent to each other diagonally, and if a number is used multiple times in a row or a column, the space between these two numbers must be equal to or larger than the value of the number. For more information see [Rip].

3. TECTONIC-/SUGURU-Lines: This puzzle differs from TECTONIC only by giving all the numbers, but not all the lines. For a SUGURU-Lines online application see [Web].

Aside from these similar looking puzzles there has not been done much research concerning TECTONIC, whereas many of the aspects of, for instance, SUDOKU have been researched. A good example of the extensive research that has been done can be seen in [Del06], [Dav] and a SAT-based SUDOKU Solver in [Web05].

I contrast to the amount of research about TECTONIC, the construction of a TECTONIC puzzle is done by many people, such as KrazyDad (see [Bum]).

Chapter 4

Rule-Based Strategy

With the rule-based approach we try to give as many restrictions to the puzzle as can be derived from looking at the form as well as the givens of the puzzle, and finally looking at the numbers we derive their form as well. Eventually we relate these rules to a certain difficulty level, which will indicate the difficulty of the puzzle. For instance, when we only have to use the fact that numbers are directly visible by being the only valid possibility in their region (by neighbourhood — and region — preclusion alone) the difficulty was not very high.

When we have a more difficult puzzle we do not want to apply each rule to each cell, but rather look at a rule and see if it changes the possibility of any cell having a certain value. For this reason we will keep a list for each cell of possible final values. We call this list "the possibilities" of that cell. A cell only gets a value if that is the only possibility left.

For each rule we can conceive, we need to implement the rule into the algorithm. This algorithm should eventually be able to solve any TECTONIC, up to a certain difficulty level. We will also define the rules in formulae, according to the following functions: $block(v)$ = the set of all cells in the same region as cell v, $value(v)$ = the value of cell v, $neighbours(v)$ = the set of all cells that are neighbours to cell v, $|block(v)|$ = the amount of cells in the region of cell v: the region's size, $possibilities(v)$ = the dynamic set of values; initially equal to $\{1, ..., |block(v)|\}$, which contains the possible final values of cell v in the current situation.

For each rule we will look at how this should be done:

1. Originally looking at a puzzle we can think of the puzzle as nothing more than a rectangle divided into cells. Each cell only has the possibility to have any of the values 1 to 5. However, because of the region sizes this is of course not true for all cells. That is why we only give each cell the possibilities which are possible if we take the region's size into account. If we then look whether there are any cells with only one possibility, and there is such a cell, then that cell needs to have that possibility as a value. This immediately leads to the rule: every region, whose size is 1, can only have the value 1 (see Figure 4.1a). For every cell w:

$$1 \leq value(w) \leq |block(w)|$$

2. Now we look whether a cell has only one possibility, not only because of the region's size, but because of the values of its neighbours and the cells in its region as well. We call this "to update the cell". Updating a cell together with the first rule can, therefore, also lead to new values of cells (see Figure 4.1b).
   For every cell w and cell v, with $v \neq w$:

$$(w \in neighbours(v)) \rightarrow possibilities(v) := possibilities(v) \setminus value(w), \text{ and}$$
$$block(w) = block(v) \rightarrow possibilities(v) := possibilities(v) \setminus value(w)$$

(a) The orange region has size 1, so its cell has value 1. **(b)** The cell at (0, 3) has 1 possibility left.

Figure 4.1

3. We can see that all cells now have their initial possibilities. If we start looking at the neighbours as well, there is a possibility that all cells in a region are precluded from having a certain value, except for one cell. That last cell with the possibility to have that value should then have that particular value (see Figure 4.2a). So here we see the case that only one cell in a region has a certain possibility.
   For every cell w, cell v and integer $i \in \{1, ..., |block(w)|\}$:

$$\forall i : (\forall v : (v \in block(w), \text{ with } v \neq w \text{ and } i \notin possibilities(v)) \rightarrow value(w) := i)$$

4. The next rule is a bit more difficult. When we see that there are cells in a region, which are the only cells in their region that have a certain value in their possibilities, then any cell which is a neighbour of all those cells cannot have those values in its possibilities. This entails that if a cell is the neighbour of all the cells in a region, then the value of that cell has to be higher than the size of the region of those neighbouring cells. That means that any cell which is a neighbour to all the cells of a region with size 4 needs to have value 5, even if not all values of that region are filled in yet, (see Figure 4.2b) and any cell which is a neighbour to all the cells of a region with size 3 needs to be either 4 or 5.
   For every cell w, cell v, where $v \neq w$, cell u, where $u \neq v$ and $u \neq w$, and integer $i \in \{1, ..., |block(w)|\}$:

$$\forall i : (\forall u : (u \in block(v) \text{ and } i \in possibilities(u) \text{ and } u \in neighbours(w)) \rightarrow possibilities(w) :=$$
$$possibilities(w) \setminus i)$$

(a) Only the cell at (3,3) can have the value 4 in its region. **(b)** The cell at (2, 3) is next to a whole region of size 4.

Figure 4.2

5. In the same way as rule 4, we can see that if two cells have the same two possibilities, these two cells are each others neighbours and are both neighbours to a third cell, then that third cell can not have the same value as any of those two possibilities of its two neighbours. The difference with rule 4, of course, is that the two cells need to have the same two possibilities, because they are not in the same region. We can see this rule applied later in the puzzle.
   For every cell w and cell v and cell u:

$$v \in neighbours(w) \text{ and } u \in neighbours(w) \text{ and } v \in neighbours(u) \text{ and } possibilities(w) = possibilities(v)$$
$$\text{and } |possibilities(w)| = 2 \rightarrow possibilities(u) := possibilities(u) \setminus possibilities(w)$$

6. If all four cells, except for one cell m — in a region of size five — (or all three cells in a region of size four, and so on) are neighbours to a cell in another region, that neighbour has the same possibilities as the cell m. This is a simple implementation of the connections we will use for 2-SAT in Chapter 5, but it is implemented because it makes use of rule 3 and rule 4 alone: The cell in the other region has the possibility to be one of five values, but then four of those values are no longer possible, because it is a neighbour of four cells in one region (rule 4) and m no longer has those possibilities, because it is in the same region as the same four cells (rule 3). See (0, 0) and (1, 2) in the final solution in Figure 4.3b.
   For every cell w, cell v and integer $i \in \{1, ..., |block(w)|\}$:

$$neighbours(v) \cap block(w) = block(w) \setminus \{w\} \text{ and } block(w) \neq block(v) \text{ and } |block(w)| \geq |block(v)|, \text{ and}$$
$$\forall\, i, \text{ where } (i \notin possibilities(w)) \,||\, i \notin possibilities(v)) \rightarrow possibilities(v) := possibilities(v) \setminus i, \text{ and}$$
$$possibilities(w) := possibilities(w) \setminus i)$$

The first five rules are the main ways to find out the value of any given cell, if the puzzle is not too difficult. The sixth rule, made by a combination of rules, is a rule to decrease the amount of possibilities, which makes it different from other rules and combinations of rules. The only other way of finding out values is by using combinations of rules which find values, and updating the possibilities of each cell, because they could be neighbours or in the same region. For instance, we can in this way see that both the cell at (1, 0) and the cell at (2, 1) do not have the possibility to be 1. This is because both cells have a duo of neighbours (the two empty yellow cells and the two neighbouring red cells, respectively), which are the only cells with the possibility to be 1 in their regions (see Figure 4.3a).

(a) Only the cell at (3, 0) has the possibility for a 1 left.

(b) The final solution.

Figure 4.3

The rest of the puzzle can be solved as follows by using the rules:

- With rule 2 we find the value 1 in the cell at (1, 2), because its neighbouring cells have all other values.
- With rule 2 and rule 3 we can see that the yellow region can be filled: the cell at (0, 0) is the only cell in its region with the possibility to be 1 and then the cell at (1, 1) can find all values except for the value 4 in its region.
- With rule 5 combined with rule 2 we find that the cell at (2, 1) needs to have the value 5. Because its neighbouring cells at (2, 2) and (3, 2) both only have the possibility to be 2 and 3, so the cell at (2, 1) cannot have a 2 or a 3 as its value and its neighbours have the values 1 and 4 (it has cells with 1 and 4, as their values, in its region as well), so the cell at (2, 1) needs to have 5 as its value.
- With rule 2 or 3 we can see that the cell at (1, 0) is the only cell in its region with the possibility to have the value 2 i.e., all other values are in its region, so the cell at (1, 0) has to have value 2.
- Now we only have 4 empty cells left and all of them need to have either value 2 or 3. With rules 2 and 3 we can see, as we did with the previously filled cells, which cell gets value 2 or value 3. This is because the cell at (1, 4) is a neighbour of the cell at (0, 3), which has value 3. This means that the cell at (1, 4) gets value 2 (rule 3), the cell at (3, 2) gets value 2 as well (rule 2) and the remaining cells get value 3 (rule 2 or rule 3). By using this last step we arrive at the final solution, as can be seen in Figure 4.3b.

We can see, by solving the puzzle this way, that this is a very easy puzzle, because although we used rule 4 and rule 5 as an example, we could also have solved the puzzle without using it: e.g., where we use rule 5 we could also have used rule 2 and rule 3 to fill the green region, as we did with the yellow region. We also saw that in the end rule 2 and rule 3 become interchangeable and that the difference between them is mainly useful in the beginning of the solving process. So, using these rules can lead to the solution of simpler puzzles. However, if this does not give the solution we need to use a different kind of puzzle solving and for this we need 2-SAT. In the case of the example puzzle we have been trying to solve, however, we saw that it is possible to solve the puzzle with just these rules.

Chapter 5

2-SAT

Given that the rule-based strategy is not able to solve all puzzles, we know that there are a certain amount of unfilled cells still left in the puzzle after applying the rule-based strategy. We try to solve the remaining part with 2-SAT (also called 2-Satisfiability). This is when we try to solve the satisfiability problem with two literals per clause. The general satisfiability problem will be further explained in Chapter 6.

5.1 The Problem

An example of a puzzle which has been partly solved by the rule-based technique is shown in Figure 5.1. We have applied the rule-based technique until no more changes were made to the puzzle. When we look at the figure, we see that there are indeed no more numbers we can add by simple rules, so we have to look at a different approach. The best way to do this is to find all the numbers in the puzzle which have to have the same value, because of different arguments. For instance, when we observe cell (0, 3) we see that, if we just look at the neighbours and region, that only the numbers 1 and 2 cannot be the value of the cell. But we can also see that, because of the form of the region of cell (0, 3), the cell has the same value as cell (1, 5). Since the cell (1, 5) can only have a 3 or 4 as its value, cell (0, 3) can only have the value 3 or 4 as well.

2	③	1		⑤	1	④		④
⑤	4	5		2	③	2		
1	2	1		1		1		
								1
		2	1					
		⑤			2	1	5	1
1	2	1	2	1	4	3	4	②
4	3	4	③	5	2	1	5	1
1	2	1	2	1	3	4	3	2
3	4	3	⑤	4	5	2	①	5
2	1	2	1	2	1	3	④	3

Figure 5.1: A rule-based, partly-solved puzzle from [Den17b]

So a possible way we can solve the rest of the puzzle is to implement this connection. The connection between cells is caused by cells which have the same value, because of the form of the regions, in addition to the numbers already filled in to some regions. For example, the cells (6, 4) and (8, 4) have the same value as well, because the cell at (6, 4) is next to cells with values 1, 2 and 5 and the cell at (8, 4) also only has the possibility to have 3 or 4 as its value, and they are both adjacent to cell (7, 4), which also has a 3 or a 4 as its value.

5.2 Representation

The way we use 2-SAT is to represent the puzzle as a graph, where there is an edge between two nodes if there is a connection between the values of cells and other values, both in its own cell as in other cells. So each edge is a connection and each node is a cell having a certain value. This way we have a direct connection between each possible value from each cell compared to every other possibility for each cell. These connections can then be followed from a value in a cell to the same value in other cells and so on. This can be seen as a sort of chain connection which is visually represented in Figure 5.2 by adding letters to the puzzle for connected cells. In this case we see that there are only a few regions which have only two empty cells. If we start looking at the regions with only two cells left empty, we know that if any of those has a link to other cells, then the cell it is linked to also has only two possible values left. Not all connections can be visualized this way, because there are connections which might not be used. For example, when we look at cell (7,0), we see that a 4 is not a possibility because the cell is adjacent to a cell with value 4. So if the variable A is a 3, the connection between the cell at (7, 2) and (7, 0) is used, but if there is no connection between those cells if A is a 4. This not only shows that not all connections can be made visible, but also how we need to look at which edges we need to add in the graph for this representation of 2-SAT to be a good representation.

2	③	1	A	⑤	1	④		④
⑤	4	5	B	2	③	2		
1	2	1	A	1		1	A	
B	A							1
		2	1		A	B	A	B
A	B	⑤	A		2	1	5	1
1	2	1	2	1	4	3	4	②
4	3	4	③	5	2	1	5	1
1	2	1	2	1	3	4	3	2
3	4	3	⑤	4	5	2	①	5
2	1	2	1	2	1	3	④	3

Figure 5.2: Representation of 2-SAT connections which lead to the final answer

Now, as is also clear from Figure 5.2, this does not yet provide a solution, because it is not clear whether A is a 3 or a 4, and of course the same is true for the value of B. But what we can see now is that both the cells at (2, 3) and (3, 3) and the cells at (4, 4) and (4, 5) have only the possibility left to be 5 and B. Now, because we also see that (3, 3) and (4, 4) are adjacent, we notice that neither 5 nor B can be the value of the cell who is adjacent to both (3, 3) and (4, 4): the cell at (4, 3), so the only possibility left there is a 2. Now that we know that, and we know that both the A and B in the cells at (5, 4) and (6, 4) are adjacent to the cell at (5, 3), we also know that only a 5 is the last possible value for the cell at (5, 3). So since the only empty cell left in this region is at (5, 2) and the cell at (3, 2) has the value of A, the value of (5, 2) is equal to the value of B. And since this B is next to a 3, we see that B has to be a 4. This leads to the final solution, as can be seen in Figure 5.3, by using the rule-based strategy again until the whole puzzle is filled.

2	③	1	3	⑤	1	④	3	④
⑤	4	5	4	2	③	2	5	1
1	2	1	3	1	4	1	3	2
4	3	4	5	2	5	2	5	1
5	1	2	1	4	3	4	3	4
3	4	⑤	3	5	2	1	5	1
1	2	1	2	1	4	3	4	②
4	3	4	③	5	2	1	5	1
1	2	1	2	1	3	4	3	2
3	4	3	⑤	4	5	2	①	5
2	1	2	1	2	1	3	④	3

Figure 5.3: Final solution with 2-SAT

5.3 2-SAT Solver

Now that we have a graph full of connections / edges, and we know which cells (nodes) already have a final value and which cells do not have a final value yet, we can build a 2-SAT solver, which will make use of this graph to find the final solution. First of all we go to the first possibility of the first cell in the puzzle, which we call v, of which we know from the graph, that it does not have a value yet and that the possibility actually occurs in the list of possibilities. We then look for connections and see if we would find a contradiction when we take this first possibility to be the final value of the cell. We do this by taking a copy of the graph and filling in the value of v to be true: the empty cell's first possibility is now the value of that first empty cell. Therefore, we then need to look if it is also possible for all the possibilities in the given cells, which the value of v has connections with, to be filled in as though they were true.

If that gives a contradiction, we know that the value of v is not true. We can then say that the value of v is false: the empty cell definitely does not have its first possibility as its value. We then proceed with the next possibility for that empty cell, or the first possibility for a next cell, and so on. If taking the value of v to be true does not give a contradiction, we do not know for certain if the value of v is the right solution, because it is possible that we just looked at a v which is not key in finding the final solution. That is why we then need to look at if — when we take the value of v to be true — we actually find a solution. To do this we first apply the rule-based technique again until no more changes can be made. If this does not lead to a solution, but we also did not find a contradiction, we do not make the value of v false. Instead we go to the first possibility of the next empty cell, which we now call v, and see if making the new value of v true, as well as the cells it has connections with, makes it possible for us to find a solution by applying the rule-based technique again. If we did find a solution — or do find a solution by using the rule-based technique and 2-SAT — we have found the final solution.

Chapter 6

SAT Solvers

Another more general way of solving Tectonics is to use the Boolean satisfiability problem, or SAT. SAT is the problem of trying to determine an answer for a given Boolean problem: if we consistently change the Boolean variables of the problem to either true or false, do we get a solution? SAT Solvers find solutions for these kinds of problems with a sophisticated algorithm. There are many different SAT Solvers and several of those are available online. For solving Tectonics we use the SAT Solver Lingeling (see [Lin]). The makers of Lingeling also wrote a paper about their entry to the 2015 SAT race (see [BBIS16]).

In order to use the SAT Solver we first have to translate the Tectonic into a logic formula in conjunctive normal form, or CNF, for Lingeling to be able to solve it. A formula as we need it is "a logic formula in CNF" when we have a conjunction of clauses, where the clauses are the disjunction of literals and each literal stands for a certain Boolean variable being either true or false.

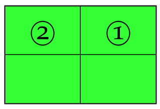

Figure 6.1: A small Tectonic for a SAT example with two solutions

To translate a Tectonic into a logic formula in CNF we have five values for each cell in the Tectonic again. Then we need to translate the basic restrictions into a logic formula in CNF. That way we can transform the Tectonic, with which the SAT Solver can try to solve the puzzle. So for every cell having a certain value, all adjacent cells and all cells in the same region cannot have the same value, but also there can only be one value per cell. So we have to make sure the SAT Solver knows that if a cell gets a value, all other values in that cell cannot be true and that a cell cannot have no value at all either. As an example we take the simple Tectonic in Figure 6.1: for this example the following CNF-formula is generated, but we should mention that xv is the Boolean variable, where x indicates that the number is a variable and $v = MaxSize * (X\text{-}coordinate + Y\text{-}coordinate * width) + value$, where $MaxSize$ stands for the maximum size of a region (which is normally 5), $value$ stands for the value that the cell would get if the variable were to be true and $width$ is the width of the puzzle:

A cell needs to have at least one value, but also not more than one value:

$$(x1 \lor x2 \lor x3 \lor x4 \lor x5) \land (\neg x1 \lor \neg x2) \land (\neg x1 \lor \neg x3) \land (\neg x1 \lor \neg x4) \land (\neg x1 \lor \neg x5) \land (\neg x2 \lor \neg x3) \land (\neg x2 \lor \neg x4) \land$$

$$(\neg x2 \lor \neg x5) \land (\neg x3 \lor \neg x4) \land (\neg x3 \lor \neg x5) \land (\neg x4 \lor \neg x5) \land$$

$$(x6 \lor x7 \lor x8 \lor x9 \lor x10) \land (\neg x6 \lor \neg x7) \land (\neg x6 \lor \neg x8) \land (\neg x6 \lor \neg x9) \land (\neg x6 \lor \neg x10) \land (\neg x7 \lor \neg x8) \land (\neg x7 \lor \neg x9) \land$$

$$(\neg x7 \lor \neg x10) \land (\neg x8 \lor \neg x9) \land (\neg x8 \lor \neg x10) \land (\neg x9 \lor \neg x10) \land$$

$$(x11 \lor x12 \lor x13 \lor x14 \lor x15) \land (\neg x11 \lor \neg x12) \land (\neg x11 \lor \neg x13) \land (\neg x11 \lor \neg x14) \land (\neg x11 \lor \neg x15) \land (\neg x12 \lor \neg x13) \land$$

$$(\neg x12 \lor \neg x14) \land (\neg x12 \lor \neg x15) \land (\neg x13 \lor \neg x14) \land (\neg x13 \lor \neg x15) \land (\neg x14 \lor \neg x15) \land$$

$$(x16 \lor x17 \lor x18 \lor x19 \lor x20) \land (\neg x16 \lor \neg x17) \land (\neg x16 \lor \neg x18) \land (\neg x16 \lor \neg x19) \land (\neg x16 \lor \neg x20) \land (\neg x17 \lor \neg x18) \land$$

$$(\neg x17 \lor \neg x19) \land (\neg x17 \lor \neg x20) \land (\neg x18 \lor \neg x19) \land (\neg x18 \lor \neg x20) \land (\neg x19 \lor \neg x20) \land$$

A cell cannot have the same value as its neighbour or the same value as any other cell in its region:

$$(x1 \lor x6 \lor x11 \lor x16) \land (\neg x1 \lor \neg x6) \land (\neg x1 \lor \neg x11) \land (\neg x1 \lor \neg x16) \land (\neg x6 \lor \neg x11) \land (\neg x6 \lor \neg x16) \land (\neg x11 \lor \neg x16) \land$$

$$(x2 \lor x7 \lor x12 \lor x17) \land (\neg x2 \lor \neg x7) \land (\neg x2 \lor \neg x12) \land (\neg x2 \lor \neg x17) \land (\neg x7 \lor \neg x12) \land (\neg x7 \lor \neg x17) \land (\neg x12 \lor \neg x17) \land$$

$$(x3 \lor x8 \lor x13 \lor x18) \land (\neg x3 \lor \neg x8) \land (\neg x3 \lor \neg x13) \land (\neg x3 \lor \neg x18) \land (\neg x8 \lor \neg x13) \land (\neg x8 \lor \neg x18) \land (\neg x13 \lor \neg x18) \land$$

$$(x4 \lor x9 \lor x14 \lor x19) \land (\neg x4 \lor \neg x9) \land (\neg x4 \lor \neg x14) \land (\neg x4 \lor \neg x19) \land (\neg x9 \lor \neg x14) \land (\neg x9 \lor \neg x19) \land (\neg x14 \lor \neg x19) \land$$

Each cell with a value gets that value and the possibilities of a cell are less than its region's size:

$$x2 \land x6 \land \neg x5 \land \neg x10 \land \neg x15 \land \neg x20$$

A case could be made for not adding the possibility for the cells here to be able to be 5 and then remove that possibility again. It is, nonetheless, added for the speed of the algorithm, such that it is able to easily jump through the puzzle when it is of a larger size. That is also why the possibility of a cell having the value 5 does not have to be added to the restriction that cells in the same region cannot be the same value. Moreover, there is no distinction between cells being neighbours and cells being in the same region, if both apply, such as in this puzzle. Therefore those relations are not added.

For more information about Satisfiability, we would recommend the Handbook of Satisfiability (see [BHvMW09]).

Chapter 7

Additional Work

During the process of creating this thesis, we have looked at some other aspects of Tectonic, which we did not address as intensely. In this chapter we will mention these aspects and what we could find out about them.

7.1 Making Tectonics

In order to be able to make Tectonics we can look at the rules, because there are situations which can never occur, because otherwise the Tectonic would have either a too large or a too small amount of answers for the puzzle to be solvable.

- The situation can never occur that any cell would be the neighbour of five cells in one region. This is because the one cell would then not be able to be any of the numbers 1 to 5, which are the only possible numbers in an original Tectonic.
- The situation can never occur that any cell in a region of size 1 would be a neighbour of all cells in any region. Otherwise this would mean that the number one can not occur in that neighbouring region.
- It could happen that the placing of regions is wrong in such a way that it does not leave a possible solution. For instance, if we take a region of size 2 and place a region of size 1 next to the first region, we have to make sure the region of size 1 is a neighbour to only one of the two cells of the region with size 2. If we then want to add a region with size 3, we need to make sure that the region of size 3 is not placed such that all cells of the other two regions are its neighbours (see Figure 7.1 for an example).

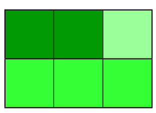

Figure 7.1: An example of a puzzle, where the regions are such that there is no solution

Apart from the regions' placing and shape, it could also be possible that there are either no solutions, or more than one solution because of (wrong) givens. This could be solved in multiple ways:

- Changing the given values, forcing the puzzler to another answer.
- Changing which numbers are given.
- Changing how many numbers are given.

7.2 Smallest (Solvable) Tectonics

Since a Tectonic is a rectangle of cells, one cell alone is also already a Tectonic. This of course is also the smallest Tectonic and the smallest solvable Tectonic, because it can only contain a one.

If we look at adding more rows or columns, we still get valid Tectonics. For instance, a normal region of size 2. We can say the same for any other row or column of one region with a size larger than one. We could also have a region of size 1 and a region of size 2 in the same row or column. This Tectonic also only has one solution and in this we can recognize a pattern: all Tectonics with only one row or column filled with one region of size 1 and n regions of size two is a solvable Tectonic with only one solution (where $n \geq 0$). Adding multiple regions of size 1 to this Tectonic would make it unsolvable.

If we do not only add rows or columns, but both rows and columns, we see that we are gonna need the numbers 3 and 4, because the smallest possible Tectonic with multiple rows and columns is a square with both height and size 2 (see Figure 6.1). These are the smallest solvable puzzles.

7.3 Empty Solvable Puzzles

We are now looking for empty puzzles with only one solution. To do this we can first look at the smallest solvable Tectonics. One empty cell is an empty solvable Tectonic, because it has only one solution: it can only contain a one. The same is true for a row or column with one region of size 1 and multiple regions of size 2, because the one cell precludes the neighbouring cells of containing a one, so the adjacent regions are solved and those then preclude the next cells of having a one as its value, so the whole empty puzzle has only one solution. However, when we have one row or column with regions with size larger than two, multiple answers can be given, and if we have multiple regions of size one the Tectonic would not be solvable.

Now we start looking at multiple rows and multiple columns and we come across an enormous amount of solvable puzzles, but most of them also have an enormous amount of possible solutions. For instance the smallest Tectonic with multiple rows and columns, as can be seen in Figure 6.1, already has many solutions without givens: $4 * 3 * 2 * 1 = 24$ possible solutions.

Chapter 8

Experiments

In this chapter we see which results we can get by experimenting with the solving techniques we made. To do this, we converted 452 puzzles (from [Den17a, Den16a, Den16b, Den16c]) to text, such that we can experiment on them. These puzzles consist of about 50 puzzles per difficulty level, where the books have 9 difficulty levels in total. Also: the puzzles are divided into 4 sizes: 5 by 4, 5 by 9, 11 by 4 and 11 by 9, where the first number is the height and the second number is the width. Only level 1 does not have any 11 by 9 puzzles.

Firstly we look at the givens per difficulty level and size of the puzzle (see Table 8.1). In the table we can see that the average givens, rounded to two decimal places, in the levels 1 to 3 decline, but any other averages of givens for higher level difficulties do not consistently decline.

Level	5 by 4	5 by 9	11 by 4	11 by 9
1	6.13	13.17	13.00	n/a
2	4.53	12.00	9.60	20.00
3	3.93	11.78	9.10	17.50
4	4.38	7.75	9.50	19.00
5	3.60	8.78	8.20	11.50
6	3.62	8.08	8.17	16.50
7	3.97	9.00	7.60	18.50
8	3.67	7.50	8.42	16.00
9	3.96	7.75	8.17	14.50

Table 8.1: Average of givens per level, height and width of the puzzle (height by width).

In Table 8.2 and Table 8.3 we can see the results of the rule-based technique on all the puzzles and how many get solved per level. To indicate the differences between the use of all combinations of rules, we remove rules which can be removed without leaving the puzzles unsolvable. We represent this removal of rules in the tables by use of $-\mathrm{R}x, u, v, w$. Here x represents the rule we remove and u, v and w can be added to represent the combination of rules which are also removed. Furthermore, 2-SAT stands for the use of 2-SAT after the rules have been used, if the rules could not solve the puzzle.

In the tables we see that all level 9 puzzles are only solvable by using 2-SAT after using the rules. Without 2-SAT we solve all puzzles up to level 5, but never all puzzles of a higher level. From the results we also see that every rule is not only added because they make sure we find more final answers to puzzles. They are also added because if we do not use some rules, the other rules can still solve some of the problems solved by the removed rules, which can lead to the final answer. The first two rules are vital to the puzzle solving, because they describe the basic restrictions of any puzzle.

All the puzzles of level 1 and 2 are solvable by using the first three rules, with the exception of two puzzles from level 2, among which is the only 11 by 9 puzzle from level 2. This is how we see that if a puzzle is solvable by rules 1 to 3 alone, we know it is a puzzle of level 1 or level 2. We could name this an altogether new level 1. If we make this new leveling system we should look at the principal differences between groups of puzzles. For instance, level 3 to 5 are all solvable by the normal set of rules, or without rules 6 and 5, but not solvable by rules 1 to 3 alone. From level 7 to 9 there is only a small amount solvable and none at all without rules 5 and 6. Since for level 6 only half of the puzzles are solvable by the normal rules and without rules 5 and 6, we can draw a line there for the second level. The remainder of the puzzles are solvable by using 2-SAT after using the normal rules, so we could call that the third rule.

Level	2-SAT	All Rules	−R6	−R5	−R4	−R3	−R6,5	−R6,4	−R6,3
1	48/48	48/48	48/48	48/48	48/48	46/48	48/48	48/48	46/48
2	51/51	51/51	51/51	51/51	49/51	50/51	51/51	49/51	50/51
3	51/51	51/51	51/51	51/51	38/51	48/51	51/51	36/51	48/51
4	50/50	50/50	50/50	50/50	16/50	40/50	50/50	8/50	40/50
5	51/51	51/51	51/51	51/51	12/51	43/51	51/51	2/51	43/51
6	50/50	42/50	37/50	30/50	6/50	33/50	26/50	1/50	30/50
7	51/51	15/51	0/51	14/51	1/51	6/51	0/51	0/51	0/51
8	50/50	3/50	0/50	2/5	0/50	1/50	0/50	0/50	0/50
9	50/50	0/50	0/50	0/50	0/50	0/50	0/50	0/50	0/50

Table 8.2: Solved / Solvable results for Rule-based strategies

Level	−R5,4	−R5,3	−R4,3	−R6,5,4	−R6,5,3	−R6,4,3	−R5,4,3	−R6,5,4,3
1	48/48	46/48	37/48	48/48	46/48	33/48	35/48	31/48
2	49/51	49/51	30/51	49/51	48/51	24/51	28/51	22/51
3	7/51	44/51	15/51	0/51	44/51	9/51	3/51	0/51
4	1/50	27/50	7/50	0/50	27/50	2/50	0/50	0/50
5	2/51	35/51	4/51	0/51	34/51	0/51	0/51	0/51
6	0/50	30/50	0/50	0/50	19/50	0/50	0/50	0/50
7	0/51	4/51	0/51	0/51	0/51	0/51	0/51	0/51
8	0/50	0/50	0/50	0/50	0/50	0/50	0/50	0/50
9	0/50	0/50	0/50	0/50	0/50	0/50	0/50	0/50

Table 8.3: Solved / Solvable results for Rule-based strategies continued

In Table 8.4 and Table 8.5 we can see the average time differences between all the results of the rule-based techniques. We represent the removal of rules in the same way as in Table 8.2 and Table 8.3, as explained before.

We can see from these results that the average time almost always increases with the increasing difficulty levels, but it does not do this consistently. For instance, all time results of level 6 puzzles seem to be the highest. The reason for this is most likely that the larger level 6 puzzles have many high numbers in the top of the puzzles, which is no more than a coincidence.

In some cases rule 4 can replace rule 5 and rule 3 and, on top of that, rule 4 also finds a lot of numbers, because it can be used on so many occasions, as can be seen in Chapter 4. Because rule 4 is such an important rule, it also makes the most changes, which makes the process take more time, which is why the difference in time can be seen best when rule 4 is not removed.

The use of 2-SAT after using the rule-based technique takes the most time on average and shows an increasingly steep incline of average time with the increasing difficulty levels, taking on average 56 ms to solve the level 9 puzzles. In this incline it is visible that there is a difference between each level. Not only is there a difference in the amount of solvable puzzles, but we also have to use a 2-SAT a lot more in the higher levels. However, these average times are of course very versatile, because we are speaking in terms of milliseconds and not a large amount of milliseconds either, which is an explanation for i.e., that using all rules and then 2-SAT is sometimes faster than just using all rules by a small margin.

Level	2-SAT	All Rules	−R6	−R5	−R4	−R3	−R6,5	−R6,4	−R6,3
1	4	4	4	3	3	5	2	4	3
2	6	8	5	5	4	10	5	4	8
3	7	8	6	5	5	11	6	3	8
4	9	11	10	9	4	15	9	2	12
5	10	11	10	9	5	12	8	2	11
6	13	13	13	12	4	16	12	2	12
7	32	11	10	10	3	12	9	2	9
8	38	13	11	10	3	13	10	2	11
9	56	13	11	11	4	12	11	2	12

Table 8.4: Average time in ms for Rule-based strategies

Level	−R5,4	−R5,3	−R4,3	−R6,5,4	−R6,5,3	−R6,4,3	−R5,4,3	−R6,5,4,3
1	2	4	3	3	3	2	1	2
2	3	6	4	3	6	3	2	2
3	2	7	3	2	7	2	2	1
4	2	11	3	2	11	1	1	1
5	2	9	2	1	10	1	1	1
6	2	11	2	1	11	1	1	1
7	2	10	2	1	8	1	1	1
8	2	11	2	2	11	1	1	1
9	3	11	2	2	11	2	1	1

Table 8.5: Average time in ms for Rule-based strategies continued

In Table 8.6 we see the amount of solved puzzles and average time for using SAT to find the final solution to all puzzles or first using rules and then SAT. From the information in the table we see that SAT is able to solve all puzzles and at virtually no time increase. However, we also see that the puzzles, which can get solved by the normal rules, take less time to be solved than solving the same puzzles with SAT. Because the rules do not solve all puzzles and take increasingly more time to find out if they can solve puzzles of higher levels, we can see that using SAT is even more useful. Using the rules first and then using SAT takes more time than using only SAT, but we do see that, if we compare the times of the rules added to the times of SAT, that rules followed by SAT is faster in all cases, although it is only in a small amount.

Because we use the SAT-solver Lingeling (see [Lin]), we also get access to some additional information. Not all this information is useful, since all puzzles get solved and all puzzles are solvable, most information gives no feedback. However, we did get the amount of propagations that were used by the SAT-solver as well as the amount of propagations per second, as can be seen in Table 8.7. From these tables we can see that there is again a slight inconsistent increase, the higher the level. This time the increase is in propagations. We call this inconsistent again, because the propagations take a slight dip in level 3, 5 and 7. This might have a cause in that these levels have more puzzles, but it is most likely coincidental, because we can find no real other link.

Level	SAT	Rules, followed by SAT
1	48/48	48/48
2	51/51	51/51
3	51/51	51/51
4	50/50	50/50
5	51/51	51/51
6	50/50	50/50
7	51/51	51/51
8	50/50	50/50
9	50/50	50/50

Level	SAT	Rules, followed by SAT
1	11	16
2	12	18
3	13	20
4	13	23
5	13	22
6	13	26
7	13	23
8	14	25
9	13	25

Table 8.6: Solved / Solvable results and average time in ms for SAT and rules followed by SAT

The propagations per second are even more inconsistent than the propagations, which is most likely because of the small amounts of time it takes for SAT to find the final solutions of the puzzles. For instance, when using the rules before giving the puzzles to the SAT-solver shows that the propagations per second look nothing like what they were without using the rules first. When we look at level 1 , which had virtually no propagations per second, because the puzzles were solved that quick, uses 1102 propagations per second when using the rules first. This same increase can be seen in the difficulty levels 2 and 3, but then level 4 suddenly gets an enormous drop, using only a quarter of the original amount of propagations per second.

Level	SAT	Rules, followed by SAT
1	122	122
2	148	154
3	139	161
4	159	174
5	149	161
6	163	171
7	148	149
8	164	160
9	163	160

Level	SAT	Rules, followed by SAT
1	0	1102
2	1294	3504
3	4867	5955
4	8887	2291
5	7485	5343
6	8950	6827
7	4583	4465
8	8385	8770
9	8975	8610

Table 8.7: Propagations and Propagations per Second for SAT and for rules followed by SAT

For all the puzzles, we have seen that they can be solved, but in most of the available information we only find clear lines of what could be differences between levels when we look at the rules, where we found three possible difficulty levels. Using SAT or rules and then SAT also solves all Tectonics, but does not provide clear lines to further split the difficulty levels.

Since we found that the rules give the clearest way of dividing the Tectonics into levels, we now want to look at how many times some of these rules are used when we are able to solve all puzzles, so we first use the rules and then 2-SAT. Here we say ”some of the rules”, because the first two rules are so important that they cannot be removed, which also means they will not give a good indication of how hard a puzzle is, because they indicate how the puzzle works. Since we also found out that rules 5 and 6 are useful for solving puzzles, but not for defining difficulty levels, we will only look at how many times rules 3 and 4 are used, rounded to two decimal places (see Table 8.8).

Level	Use of Rule 3	Use of Rule 4
1	12.31	5.35
2	15.53	9.31
3	16.53	11.92
4	19.24	16.12
5	17.10	16.67
6	18.38	20.24
7	28.29	20.41
8	34.88	28.42
9	54.26	41.86

Table 8.8: Average use of the rules 3 and 4 for Rules, extended with 2-SAT

In the table we can see that the use of rules increases. Like in most results we found, the increase of the use of rules 3 and 4 does not increase consistently, because level 5 and level 6 puzzles use rule 3 less times on average than rule 4. A possible reason for this could be that the use of rule 4 increases faster than the use of rule 3 in the lower level Tectonics. When the usage of the rules is almost the same, we see that the usage of rule 3 decreases. The explanation could be that this occurs because rule 3 and 4 are sometimes interchangeable, as can be seen when we use only rule 3 or only rule 4 in Table 8.3. This means that the amount of times rule 3 or rule 4 is used depends on the order in which the rules are used. In our case that means the simplest rule is used first. Another explanation could be that the average amount of givens for the level 5, 11 by 9 puzzles is way less than the average amount of givens in any other level (see Table 8.1), which leaves more cells to be filled.

So, if we look at Table 8.8, we can see that we could distinguish which level a puzzle is by the difference in the usage of rule 4, combined with the three levels we could already distinguish. Only the difference between level 4 and level 5, and level 6 and level 7 is not big enough to be able to define which level we are dealing with, so we look at the use of rule 3. This is very useful when we look at the difference between level 6 and 7, because the average amount of usage of rule 3 is quite large. However, as we said, the usage of rule 3 decreases between level 4 and 5, so if we would have to make a guess, we would probably say a level 4 puzzle is actually more difficult.

Now that we know the difference between the difficulty levels, we only need to make a clear line, such that every future Tectonic is assigned the right difficulty level. For instance, we could take the median of the average usage of the rules to define this line. That would mean, that the difference between level 1 and level 2 would be a usage of level 4 more or less than 7.33 times, the difference between level 3 and level 4 would be 14.02 and so on. In this way we are able to distinguish between level 1 and level 2, level 3 and level 4, level 5 and level 6, level 6 and level 7, level 7 and level 8, and level 8 and level 9. And since we were already able to distinguish between level 2 and level 3 and between level 5 and half of level 6 and the other half of level 6 and level 7, we can now distinguish 8 different difficulty levels.

Chapter 9

Conclusion and Future Work

We looked at different ways to solve any Tectonic. We found that we are able to achieve this goal by implementing a set of rules, and extending this with 2-SAT, as well as using SAT with the SAT Solver Lingeling (see [Lin]). Furthermore we wanted to distinguish between different levels of Tectonic. The books give nine levels of difficulty levels and we took many puzzles of each level, where we looked at the information we could take from the puzzle solving techniques. Now that we had this information we were only able to distinguish between 8 different difficulty levels, which included a set of puzzles of the "normal" difficulty levels. The levels only coincide with the original 9 difficulty levels in level 4 and level 5, because they can not be distinguished from each other.

We wanted to conclude if it is possible to solve any puzzle and after doings so see, if we could determine the difficulty level, and we can now conclude that for any puzzle in the book it is possible to solve it with certain solving techniques, after which we can indeed determine the difficulty level, although we end up with less levels.

There is still work we think is worth looking at. For instance, in Chapter 7 we looked at some additional work, which we did not look at as intensely as we did to the other work. Also, it might be possible that there is a different set of rules, which would solve higher level puzzles, or a way of solving Tectonics in such a manner that more difficulty levels can be distinguished. This work also awaits further research.

Bibliography

[BBIS16] T. Balyo, A. Biere, M. Iser, and C. Sinz. SAT race 2015. *Artif. Intell.*, 241:45–65, 2016.

[BHvMW09] A. Biere, M. Heule, H. van Maaren, and T. Walsh. *Handbook of Satisfiability: Volume 185 Frontiers in Artificial Intelligence and Applications*. IOS Press, Amsterdam, The Netherlands, 2009.

[Bum] Jim Bumgardner. KrazyDad. `https://krazydad.com/suguru/`. [Online; accessed 14-June-2017].

[Dav] Tom Davis. The mathematics of sudoku. `http://www.geometer.org/mathcircles/sudoku.pdf`. [Online; accessed 16-July-2017].

[Del06] Jean-Paul Delahaye. The Science Behind Sudoku. *Scientific American*, 294:80–87, June 2006.

[Den16a] Denksport. *Tectonic 5-7**. Aldipress BV, 6 edition, 11 2016. Puzzle 126, page 51.

[Den16b] Denksport. *Tectonic 5-7**. Aldipress BV, 7 edition, 12 2016.

[Den16c] Denksport. *Tectonic 7-9**. Aldipress BV, 2 edition, 11 2016.

[Den17a] Denksport. *Tectonic 3-5**. Aldipress BV, 13 edition, 12-1 2016-2017.

[Den17b] Denksport. *Tectonic 1-3**. Aldipress BV, 17 edition, 1 2017.

[Lin] Lingeling, Plingeling and Treengeling. `http://fmv.jku.at/lingeling/`. [Online; accessed 12-June-2017].

[Rip] Ripple Effect. `https://en.wikipedia.org/wiki/Ripple_Effect_(puzzle)`. [Online; accessed 14-June-2017].

[Sud] Sudoku. `https://en.wikipedia.org/wiki/Sudoku`. [Online; accessed 14-June-2017].

[Web] Digitown Webdesign. Suguru-Lines. `http://sugurupuzzles.com/suguru-puzzle-lines.html`. [Online; accessed 15-June-2017].

[Web05] Tjark Weber. A SAT-based Sudoku solver. In Geoff Sutcliffe and Andrei Voronkov, editors, *LPAR-12, The 12th International Conference on Logic for Programming, Artificial Intelligence, and Reasoning, Short Paper Proceedings*, pages 11–15, December 2005.