Office Hour for Z3

Practical Exercise Sheet 1

Eva Röper

May 17, 2022

General Information

Part1

Organization of the Office Hour:

- Short introduction to your task
- Live demo: Constraint network for Coloring Australia example
- Q&A

General Information

Part2

- Task: Modeling a Sudoku puzzle as CSP
- Tool: Z3 theorem prover[1]
- Solutions due Friday, May 27, 23:59
- Additional information: Z3-Tutorial[2]

[1] https://github.com/Z3Prover/z3

[2] https://www.philipzucker.com/z3-rise4fun/guide.html

Sudoku Constraints

Typical Sudoku constraints:

1. All cells must be filled with numbers between 1 and 9.
   (already implemented in sudoku.z3)
2. Numbers cannot be repeated in any row, column, or 3x3 square
3. Cells whose values are already specified must be assigned to the respective values.

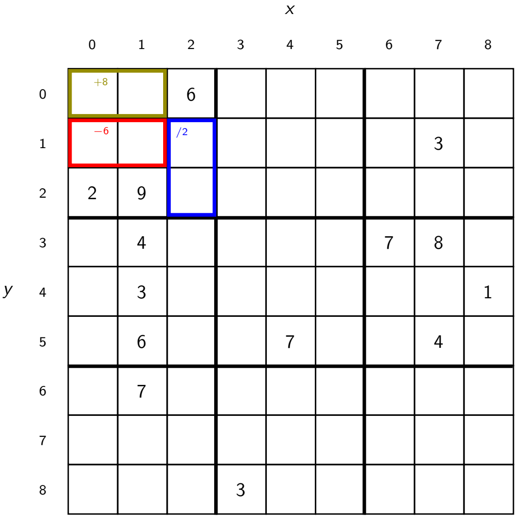
x

		0	1	2	3	4	5	6	7	8
	0			6						
	1								3	
	2	2	9							
	3		4					7	8	
y	4		3							1
	5		6			7			4	
	6		7							
	7									
	8				3					

Additional Constraints

Top left square:
The numbers must comply with the arithmetic expressions drawn in the figure:

1. $\langle 0,0\rangle + \langle 1,0\rangle = 8$
2. $\langle 0,1\rangle - \langle 1,1\rangle = 6$
3. $\langle 2,2\rangle / \langle 2,1\rangle = 2$

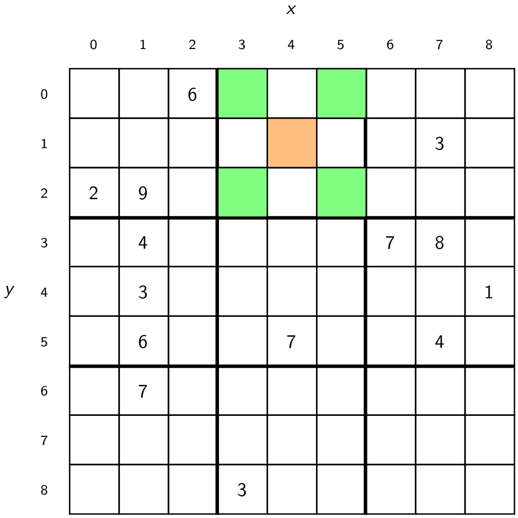


y \ x	0	1	2	3	4	5	6	7	8
0	+8		6						
1	−6		/2					3	
2	2	9							
3		4					7	8	
4		3							1
5		6			7			4	
6		7							
7									
8				3					

Additional Constraints

Top middle square:

- The values of the green cells must be either all odd or all even.
- Moreover, if the green cells contain odd numbers, then the orange cell must contain an even number.
- If the green cells contain even numbers, then the orange cell must contain an odd number.

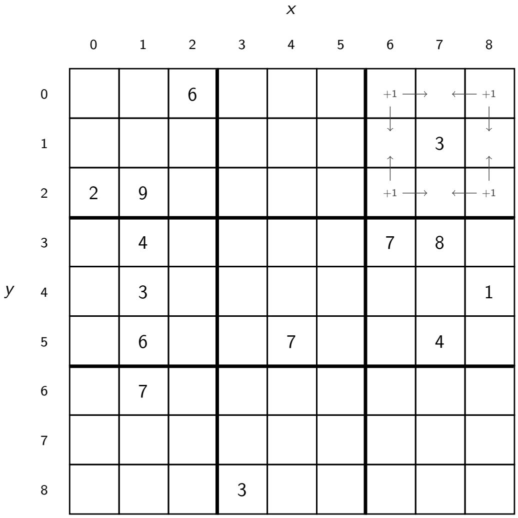


	0	1	2	3	4	5	6	7	8
0			6	🟩		🟩			
1					🟧			3	
2	2	9		🟩		🟩			
3		4					7	8	
4		3							1
5		6			7			4	
6		7							
7									
8				3					

Additional Constraints

Top right square:

- For every corner cell of this square, one of the horizontally or vertically adjacent cells must equal the value plus 1.
- When for example the cell $\langle 6, 0 \rangle = 4$, then either $\langle 7, 0 \rangle = 5$ or $\langle 6, 1 \rangle = 5$.



y \ x	0	1	2	3	4	5	6	7	8
0			6						
1								3	
2	2	9							
3		4					7	8	
4		3							1
5		6			7			4	
6		7							
7									
8				3					

Additional Constraints

Middle left square:
The sum of all rows in this square must be equal, i.e.,

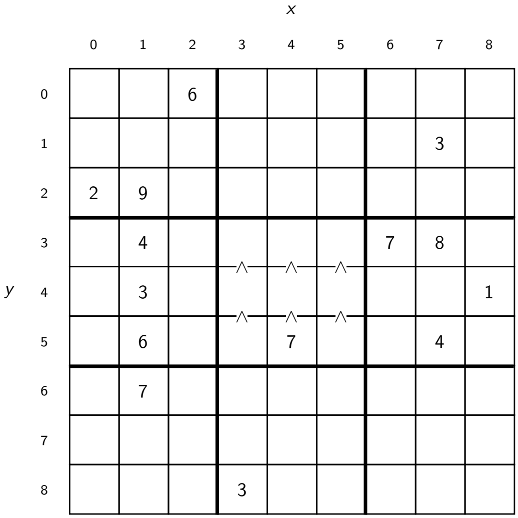
$$
\begin{aligned}
&\langle 0,3\rangle + \langle 1,3\rangle + \langle 2,3\rangle \\
= &\langle 0,4\rangle + \langle 1,4\rangle + \langle 2,4\rangle \\
= &\langle 0,5\rangle + \langle 1,5\rangle + \langle 2,5\rangle.
\end{aligned}
$$

						x				
		0	1	2	3	4	5	6	7	8
	0			6						
	1								3	
	2	2	9							
	3		4					7	8	
y	4		3							1
	5		6			7			4	
	6		7							
	7									
	8				3					

Additional Constraints

Center square:
Numbers must comply with the inequalities. More specifically:

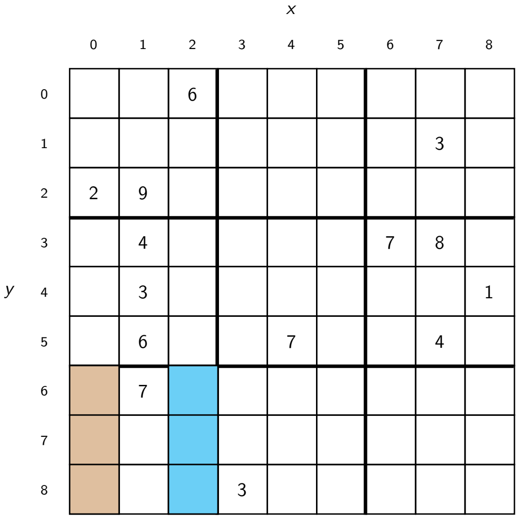
1. $\langle 3,3\rangle < \langle 3,4\rangle < \langle 3,5\rangle$
2. $\langle 4,3\rangle < \langle 4,4\rangle < \langle 4,5\rangle$
3. $\langle 0,5\rangle < \langle 1,5\rangle < \langle 2,5\rangle$



	0	1	2	3	4	5	6	7	8
0			6						
1								3	
2	2	9							
3		4					7	8	
4		3							1
5		6			7			4	
6		7							
7									
8				3					

Additional Constraints

Bottom left square:
Multiplying the sums of the two indicated columns gives an odd number:
$(\langle 0,6\rangle + \langle 0,7\rangle + \langle 0,8\rangle)$
$\times(\langle 2,6\rangle + \langle 2,7\rangle + \langle 2,8\rangle)$
must be odd.

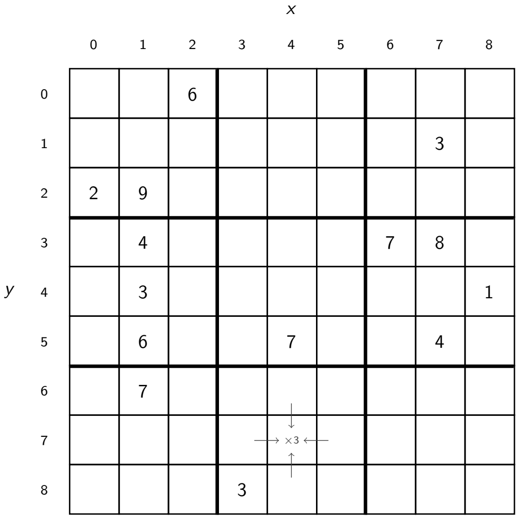


	0	1	2	3	4	5	6	7	8
0			6						
1								3	
2	2	9							
3		4					7	8	
4		3							1
5		6			7			4	
6	🟫	7	🟦						
7	🟫		🟦						
8	🟫		🟦	3					

Additional Constraints

Bottom middle square:

- The sum of the indicated cells must be equal to three times the value of the center cell.
- In other words: $\langle 4, 6\rangle + \langle 3, 7\rangle + \langle 4, 8\rangle + \langle 5, 7\rangle$ $= 3 \times \langle 4, 7\rangle$



	0	1	2	3	4	5	6	7	8
0			6						
1								3	
2	2	9							
3		4					7	8	
4		3							1
5		6			7			4	
6		7							
7					×3				
8				3					

Additional Constraints

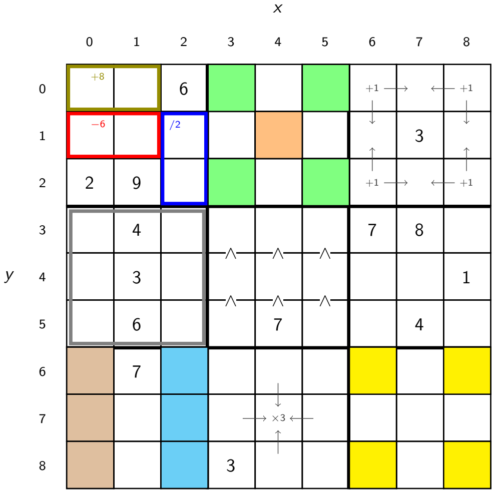
Bottom right square:
At most one of the yellow cells may contain a value larger than 4.

x

y	0	1	2	3	4	5	6	7	8
0			6						
1								3	
2	2	9							
3		4					7	8	
4		3							1
5		6			7			4	
6		7					🟨		🟨
7									
8				3			🟨		🟨

Additional Constraints

All Together



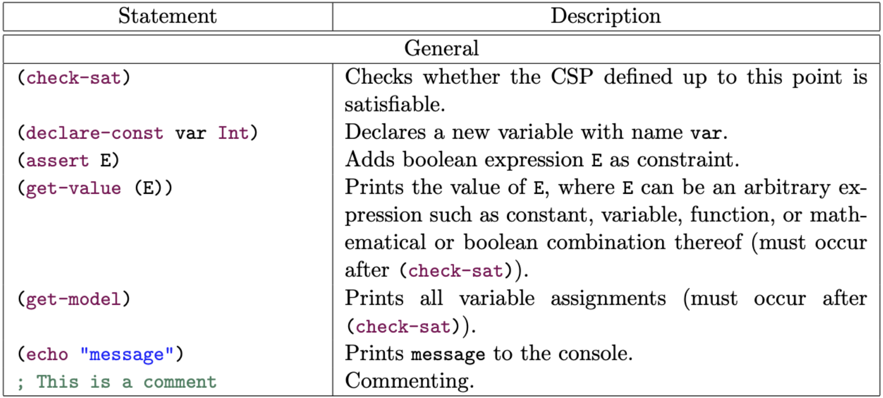
	0	1	2	3	4	5	6	7	8
0			6						
1								3	
2	2	9							
3		4					7	8	
4		3							1
5		6			7			4	
6		7							
7									
8				3					

Z3 Information

- Fill 9 constraints in the provided *sudoku.z3* file
- Only use statements shown in table 1 of the practical sheet (these will follow on the next slide as well)
- Run your implementation by passing *sudoku.z3* to your Z3 executable: **z3 path/to/sudoku.z3**
- Print additional information about the solving process by: **z3 -st path/to/sudoku.z3**
- Result of Z3 (**sat** or **unsat**) will be printed to the console

Allowed Z3 Statements

Part 1



Statement	Description
General	
`(check-sat)`	Checks whether the CSP defined up to this point is satisfiable.
`(declare-const var Int)`	Declares a new variable with name `var`.
`(assert E)`	Adds boolean expression `E` as constraint.
`(get-value (E))`	Prints the value of `E`, where `E` can be an arbitrary expression such as constant, variable, function, or mathematical or boolean combination thereof (must occur after `(check-sat)`).
`(get-model)`	Prints all variable assignments (must occur after `(check-sat)`).
`(echo "message")`	Prints `message` to the console.
`; This is a comment`	Commenting.

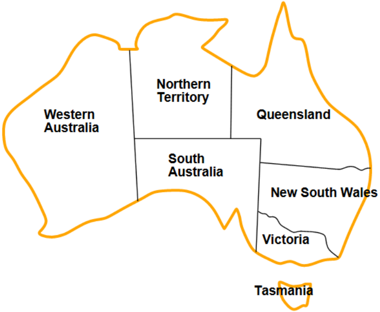
Allowed Z3 Statements

Part2

Mathematical Expressions	
c	Constants $c \in \mathbb{Z}$.
var	Evaluates to the value of variable var.
(Board x1 y2)	Evaluates to the value of the cell with coordinates $\langle 1, 2\rangle$
($\circ$ E_1 ... E_n)	Evaluates to $E_1 \circ E_2 \circ \cdots \circ E_n$, where $\circ$ can be any of +, -, and *.

Boolean Expressions	
true	Constant for true.
false	Constant for false.
(not E)	Negation of the boolean expression E.
(and E_1 ... E_n)	Conjunction over the boolean expressions E_1 to E_n.
(or E_1 ... E_n)	Disjunction over the boolean expressions E_1 to E_n.
($\circ$ E_1 ... E_n)	Is true iff for the evaluation of the expressions E_1 to E_n, it holds that $E_1 \circ E_2$ and $E_2 \circ E_3$, ..., and $E_{n-1} \circ E_n$, where $\circ$ can be any of <, <=, >, >=, and =.
(distinct E_1 ... E_n)	Is true iff every expression E_1 to E_n evaluates to a different value.

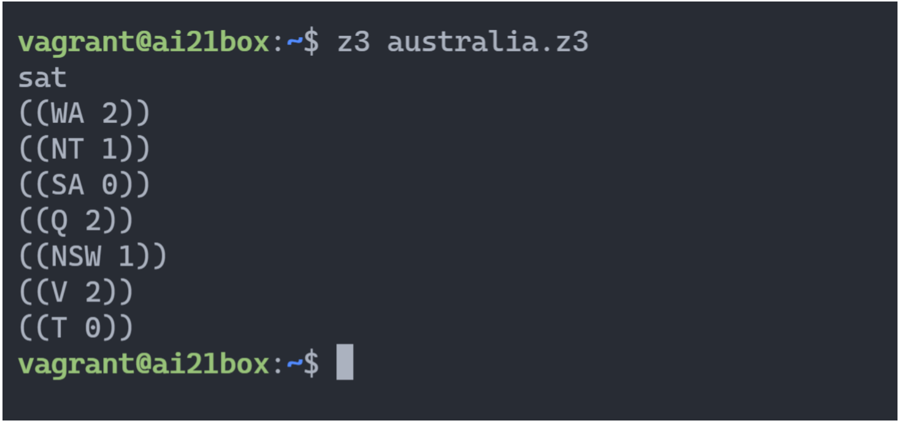
Z3 in Practice



- **Variables:** $V = \{WA, NT, SA, Q, NSW, V, T\}$.
- **Domains:** For all $v \in V$: $D_v = \{red, green, blue\} =: D$.

  $\rightarrow$ If all variables have the same domain, abusing notation we will write D to denote that "global" domain.
- **Constraints:** C_{uv} for adjacent states u and v, with $C_{uv} ="u \neq v"$, i.e., $C_{uv} = \{(d, d') \in D \times D \mid d \neq d'\}$.

Z3 in Practice



```
vagrant@ai21box:~$ z3 australia.z3
sat
((WA 2))
((NT 1))
((SA 0))
((Q 2))
((NSW 1))
((V 2))
((T 0))
vagrant@ai21box:~$ 
```