Additional practical examples: Formalization in Propositional Logic

Chiara Ghidini
ghidini@fbk.eu

12 March 2013

Traffic Light

Problem

Define a propositional language which allows to describe the state of a traffic light on different instants. With the language defined above provide a (set of) formulas which expresses the following facts:

- the traffic light is either green, or red or orange;
- the traffic light switches from green to orange, from orange to red, and from red to green;
- it can keep the same color over at most 3 successive states.

Traffic Light

Solution

- $g_k =$"*traffic light is green at instant k*", $r_k =$"*traffic light is red at instant k*" and $o_k =$"*traffic light is orange at instant k*".

Traffic Light

Solution

- g_k ="*traffic light is green at instant k*", r_k ="*traffic light is red at instant k*" and o_k ="*traffic light is orange at instant k*".
- Let's formalize the traffic light behavior:

Traffic Light

Solution

- g_k ="*traffic light is green at instant k*", r_k ="*traffic light is red at instant k*" and o_k ="*traffic light is orange at instant k*".
- Let's formalize the traffic light behavior:
  1. *"the traffic light is either green, or red or orange"*

Traffic Light

Solution

- g_k ="*traffic light is green at instant k*", r_k ="*traffic light is red at instant k*" and o_k ="*traffic light is orange at instant k*".
- Let's formalize the traffic light behavior:
  1. *"the traffic light is either green, or red or orange"*
     $(g_k \leftrightarrow (\neg r_k \wedge \neg o_k)) \wedge (r_k \leftrightarrow (\neg g_k \wedge \neg o_k)) \wedge (o_k \leftrightarrow (\neg r_k \wedge \neg g_k))$

Traffic Light

Solution

- g_k ="*traffic light is green at instant k*", r_k ="*traffic light is red at instant k*" and o_k ="*traffic light is orange at instant k*".
- Let's formalize the traffic light behavior:
  1. *"the traffic light is either green, or red or orange"*
     $(g_k \leftrightarrow (\neg r_k \wedge \neg o_k)) \wedge (r_k \leftrightarrow (\neg g_k \wedge \neg o_k)) \wedge (o_k \leftrightarrow (\neg r_k \wedge \neg g_k))$
  2. *"the traffic light switches from green to orange, from orange to red, and from red to green"*

Traffic Light

Solution

- g_k ="*traffic light is green at instant k*", r_k ="*traffic light is red at instant k*" and o_k ="*traffic light is orange at instant k*".
- Let's formalize the traffic light behavior:
  1. *"the traffic light is either green, or red or orange"*
  $(g_k \leftrightarrow (\neg r_k \wedge \neg o_k)) \wedge (r_k \leftrightarrow (\neg g_k \wedge \neg o_k)) \wedge (o_k \leftrightarrow (\neg r_k \wedge \neg g_k))$
  2. *"the traffic light switches from green to orange, from orange to red, and from red to green"*
  $(g_{k-1} \rightarrow (g_k \vee o_k)) \wedge (o_{k-1} \rightarrow (o_k \vee r_k)) \wedge (r_{k-1} \rightarrow (r_k \vee g_k))$

Traffic Light

Solution

- g_k ="*traffic light is green at instant k*", r_k ="*traffic light is red at instant k*" and o_k ="*traffic light is orange at instant k*".
- Let's formalize the traffic light behavior:

1. "*the traffic light is either green, or red or orange*"
   $(g_k \leftrightarrow (\neg r_k \wedge \neg o_k)) \wedge (r_k \leftrightarrow (\neg g_k \wedge \neg o_k)) \wedge (o_k \leftrightarrow (\neg r_k \wedge \neg g_k))$
2. "*the traffic light switches from green to orange, from orange to red, and from red to green*"
   $(g_{k-1} \rightarrow (g_k \vee o_k)) \wedge (o_{k-1} \rightarrow (o_k \vee r_k)) \wedge (r_{k-1} \rightarrow (r_k \vee g_k))$
3. "*it can keep the same color over at most 3 successive states*"

Traffic Light

Solution

- g_k ="*traffic light is green at instant k*", r_k ="*traffic light is red at instant k*" and o_k ="*traffic light is orange at instant k*".
- Let's formalize the traffic light behavior:
  1. "*the traffic light is either green, or red or orange*"
     $(g_k \leftrightarrow (\neg r_k \wedge \neg o_k)) \wedge (r_k \leftrightarrow (\neg g_k \wedge \neg o_k)) \wedge (o_k \leftrightarrow (\neg r_k \wedge \neg g_k))$
  2. "*the traffic light switches from green to orange, from orange to red, and from red to green*"
     $(g_{k-1} \rightarrow (g_k \vee o_k)) \wedge (o_{k-1} \rightarrow (o_k \vee r_k)) \wedge (r_{k-1} \rightarrow (r_k \vee g_k))$
  3. "*it can keep the same color over at most 3 successive states*"
     $(g_{k-3} \wedge g_{k-2} \wedge g_{k-1} \rightarrow \neg g_k) \wedge (r_{k-3} \wedge r_{k-2} \wedge r_{k-1} \rightarrow \neg r_k) \wedge (o_{k-3} \wedge o_{k-2} \wedge o_{k-1} \rightarrow \neg o_k)$

Graph Coloring Problem

Problem

Provide a propositional language and a set of axioms that formalize the graph coloring problem of a graph with at most n nodes, with connection degree $\leq m$, and with less then $k+1$ colors.

- node degree: number of adjacent nodes
- connection degree of a graph: max among all the degree of its nodes
- Graph coloring problem: given a non-oriented graph, associate a color to each of its nodes in such a way that no pair of adjacent nodes have the same color.

Graph Coloring: Propositional Formalization

Language

- For each $1 \leq i \leq n$ and $1 \leq c \leq k$, color_{ic} is a proposition, which intuitively means that *"the i-th node has the c color"*
- For each $1 \leq i \neq j \leq n$, edge_{ij} is a proposition, which intuitively means that *"the i-th node is connected with the j-th node"*.

Graph Coloring: Propositional Formalization

Language

- For each $1 \leq i \leq n$ and $1 \leq c \leq k$, color_{ic} is a proposition, which intuitively means that *"the i-th node has the c color"*
- For each $1 \leq i \neq j \leq n$, edge_{ij} is a proposition, which intuitively means that *"the i-th node is connected with the j-th node"*.

Axioms

- for each $1 \leq i \leq n$, $\bigvee_{c=1}^{k} \text{color}_{ic}$
  "each node has at least one color"

Graph Coloring: Propositional Formalization

Language

- For each $1 \leq i \leq n$ and $1 \leq c \leq k$, color_{ic} is a proposition, which intuitively means that *"the i-th node has the c color"*
- For each $1 \leq i \neq j \leq n$, edge_{ij} is a proposition, which intuitively means that *"the i-th node is connected with the j-th node"*.

Axioms

- for each $1 \leq i \leq n$, $\bigvee_{c=1}^{k} \text{color}_{ic}$
  "each node has at least one color"
- for each $1 \leq i \leq n$ and $1 \leq c, c' \leq k$, $\text{color}_{ic} \rightarrow \neg\text{color}_{ic'}$
  "every node has at most 1 color"

Graph Coloring: Propositional Formalization

Language

- For each $1 \leq i \leq n$ and $1 \leq c \leq k$, color_{ic} is a proposition, which intuitively means that *"the i-th node has the c color"*
- For each $1 \leq i \neq j \leq n$, edge_{ij} is a proposition, which intuitively means that *"the i-th node is connected with the j-th node"*.

Axioms

- for each $1 \leq i \leq n$, $\bigvee_{c=1}^{k} \text{color}_{ic}$
  "each node has at least one color"
- for each $1 \leq i \leq n$ and $1 \leq c, c' \leq k$, $\text{color}_{ic} \rightarrow \neg\text{color}_{ic'}$
  "every node has at most 1 color"
- for each $1 \leq i, j \leq n$ and $1 \leq c \leq k$, $\text{edge}_{ij} \rightarrow \neg(\text{color}_{ic} \wedge \text{color}_{jc})$
  "adjacent nodes do not have the same color"

Graph Coloring: Propositional Formalization

Language

- For each $1 \leq i \leq n$ and $1 \leq c \leq k$, color_{ic} is a proposition, which intuitively means that *"the i-th node has the c color"*
- For each $1 \leq i \neq j \leq n$, edge_{ij} is a proposition, which intuitively means that *"the i-th node is connected with the j-th node"*.

Axioms

- for each $1 \leq i \leq n$, $\bigvee_{c=1}^{k} \text{color}_{ic}$
  "each node has at least one color"
- for each $1 \leq i \leq n$ and $1 \leq c, c' \leq k$, $\text{color}_{ic} \rightarrow \neg\text{color}_{ic'}$
  "every node has at most 1 color"
- for each $1 \leq i, j \leq n$ and $1 \leq c \leq k$, $\text{edge}_{ij} \rightarrow \neg(\text{color}_{ic} \wedge \text{color}_{jc})$
  "adjacent nodes do not have the same color"
- for each $1 \leq i \leq n$, and each $J \subseteq \{1..n\}$, where $|J| = m$,
  $\bigwedge_{j \in J} \text{edge}_{ij} \rightarrow \bigwedge_{j \notin J} \neg\text{edge}_{ij}$
  "every node has at most m connected nodes"

Sudoku Example

Problem

Sudoku is a placement puzzle. The aim of the puzzle is to enter a numeral from 1 through 9 in each cell of a grid, most frequently a 9×9 grid made up of 3×3 subgrids (called "regions"), starting with various numerals given in some cells (the "givens"). Each row, column and region must contain only one instance of each numeral. Its grid layout is like the one shown in the following schema

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

Sudoku Example

Problem

Sudoku is a placement puzzle. The aim of the puzzle is to enter a numeral from 1 through 9 in each cell of a grid, most frequently a 9×9 grid made up of 3×3 subgrids (called "regions"), starting with various numerals given in some cells (the "givens"). Each row, column and region must contain only one instance of each numeral. Its grid layout is like the one shown in the following schema

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

Provide a formalization in propositional logic of the sudoku problem, so that any truth assignment to the propositional variables that satisfy the axioms is a solution for the puzzle.

Sudoku Example: Solution

Language

For $1 \leq n, r, c \leq 9$,define the proposition

$$in(n, r, c)$$

which means that the number n has been inserted in the cross between row r and column c.

Sudoku Example: Solution

Axioms

Sudoku Example: Solution

Axioms

1. *"A raw contains all numbers from 1 to 9 "*

Sudoku Example: Solution

Axioms

1. *"A raw contains all numbers from 1 to 9 "*

$$\bigwedge_{r=1}^{9}\left(\bigwedge_{n=1}^{9}\left(\bigvee_{c=1}^{9} in(n, r, c)\right)\right)$$

Sudoku Example: Solution

Axioms

1. *"A raw contains all numbers from 1 to 9 "*

$$\bigwedge_{r=1}^{9}\left(\bigwedge_{n=1}^{9}\left(\bigvee_{c=1}^{9} in(n,r,c)\right)\right)$$

2. *"A column contains all numbers from 1 to 9"*

Sudoku Example: Solution

Axioms

1. *"A raw contains all numbers from 1 to 9 "*

$$\bigwedge_{r=1}^{9}\left(\bigwedge_{n=1}^{9}\left(\bigvee_{c=1}^{9} in(n,r,c)\right)\right)$$

2. *"A column contains all numbers from 1 to 9"*

$$\bigwedge_{c=1}^{9}\left(\bigwedge_{n=1}^{9}\left(\bigvee_{r=1}^{9} in(n,r,c)\right)\right)$$

Sudoku Example: Solution

Axioms

1. *"A raw contains all numbers from 1 to 9 "*

$$\bigwedge_{r=1}^{9}\left(\bigwedge_{n=1}^{9}\left(\bigvee_{c=1}^{9} in(n,r,c)\right)\right)$$

2. *"A column contains all numbers from 1 to 9"*

$$\bigwedge_{c=1}^{9}\left(\bigwedge_{n=1}^{9}\left(\bigvee_{r=1}^{9} in(n,r,c)\right)\right)$$

3. *"A region (sub-grid) contains all numbers from 1 to 9"*

Sudoku Example: Solution

Axioms

1. *"A raw contains all numbers from 1 to 9 "*

$$\bigwedge_{r=1}^{9}\left(\bigwedge_{n=1}^{9}\left(\bigvee_{c=1}^{9} in(n,r,c)\right)\right)$$

2. *"A column contains all numbers from 1 to 9"*

$$\bigwedge_{c=1}^{9}\left(\bigwedge_{n=1}^{9}\left(\bigvee_{r=1}^{9} in(n,r,c)\right)\right)$$

3. *"A region (sub-grid) contains all numbers from 1 to 9"*

$$\text{for any} \quad 0 \leq k, h \leq 2 \qquad \bigwedge_{n=1}^{9}\left(\bigvee_{r=1}^{3}\left(\bigvee_{c=1}^{3} in(n, 3*k+r, 3*h+c)\right)\right)$$

Sudoku Example: Solution

Axioms

1. *"A raw contains all numbers from 1 to 9 "*

$$\bigwedge_{r=1}^{9}\left(\bigwedge_{n=1}^{9}\left(\bigvee_{c=1}^{9} in(n,r,c)\right)\right)$$

2. *"A column contains all numbers from 1 to 9"*

$$\bigwedge_{c=1}^{9}\left(\bigwedge_{n=1}^{9}\left(\bigvee_{r=1}^{9} in(n,r,c)\right)\right)$$

3. *"A region (sub-grid) contains all numbers from 1 to 9"*

for any $0 \leq k,h \leq 2$ $$\bigwedge_{n=1}^{9}\left(\bigvee_{r=1}^{3}\left(\bigvee_{c=1}^{3} in(n, 3*k+r, 3*h+c)\right)\right)$$

4. *"A cell cannot contain two numbers"*

Sudoku Example: Solution

Axioms

1. *"A raw contains all numbers from 1 to 9 "*

$$\bigwedge_{r=1}^{9}\left(\bigwedge_{n=1}^{9}\left(\bigvee_{c=1}^{9} in(n,r,c)\right)\right)$$

2. *"A column contains all numbers from 1 to 9"*

$$\bigwedge_{c=1}^{9}\left(\bigwedge_{n=1}^{9}\left(\bigvee_{r=1}^{9} in(n,r,c)\right)\right)$$

3. *"A region (sub-grid) contains all numbers from 1 to 9"*

for any $0 \leq k, h \leq 2$ $\quad \bigwedge_{n=1}^{9}\left(\bigvee_{r=1}^{3}\left(\bigvee_{c=1}^{3} in(n, 3*k+r, 3*h+c)\right)\right)$

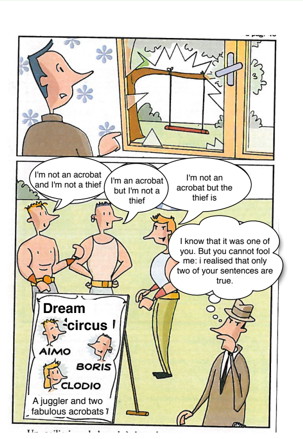
4. *"A cell cannot contain two numbers"*

for any $1 \leq n, n', c, r \leq 9$ and $n \neq n'$ $\quad in(n,r,c) \rightarrow \neg in(n',r,c)$

The circus puzzle

Problem

Consider the following puzzle



The circus puzzle: Solution

Language

- AA = "Aimo is an acrobat"
- AJ = "Aimo is a juggler"
- AT = "Aimo is a thief"
- BA = "Boris is an acrobat"
- BJ = "Boris is a juggler"
- BT = "Boris is a thief"
- CA = "Clodio is an acrobat"
- CJ = "Clodio is a juggler"
- CT = "Clodio is a thief"
- A = "I'm not an acrobat and I'm not a thief"
- B = "I'm an acrobat but I'm not a thief"
- C = "I'm not an acrobat but the thief is"

The circus puzzle: Solution (?)

Axioms

- A $\equiv \neg$ AA $\wedge \neg$ AT ("I'm not an acrobat and I'm not a thief")
- B $\equiv$ BA $\wedge \neg$ BT ("I'm an acrobat but I'm not a thief")
- C $\equiv \neg$ CA $\wedge$ (AT $\supset$ AA) $\wedge$ (BT $\supset$ BA) $\wedge$ (CT $\supset$ CA) ("I'm not an acrobat but the thief is")
- AT $\vee$ BT $\vee$ CT ("the thief is one among the three")
- (AJ $\wedge$ BA $\wedge$ CA) $\vee$ (AA $\wedge$ BJ $\wedge$ CA) $\vee$ (AA $\wedge$ BA $\wedge$ CJ) ("there are a juggler and two acrobats")
- (A $\wedge$ B $\wedge \neg$ C) $\vee$ (A $\wedge \neg B$ $\wedge$ C) $\vee$ ($\neg$ A $\wedge$ B $\wedge$ C) ("only two statements are true")
- AA $\equiv \neg$ AJ, BA $\equiv \neg$ BJ, CA $\equiv \neg$ CJ ("one cannot be juggler and acrobat at the same time")