Program Equivalence in SMT

- Prove that these two programs are equivalent.

```c
int power3(int in0){
    int i, out_a;
    out_a = in0;
    for (i= 0; i< 2; i++)
        out_a = out_a * in0
    return out_a;
}
```

```c
int power3_new(int in0){

    int out_b;
    out_b = (in0 * in0) * in0;
    return out_b;

}
```

$$\Phi_a \equiv (out0_a = in0_a) \wedge (out1_a = out0_a * in0_a) \wedge (out2_a = out1_a * in0_a)$$

$$\Phi_b \equiv out0_b = (in0_b * in0_b) * in0_b$$

Program Equivalence in SMT

```c
int power3(int in0){
     int i, out_a;
     out_a = in0;
     for (i= 0; i< 2; i++)
          out_a = out_a * in0
     return out_a;
 }
```

```c
int power3_new(int in0){

     int out_b;
     out_b = (in0 * in0) * in0;
     return out_b;

 }
```

$$\Phi_a \equiv (out0_a = in0_a) \wedge (out1_a = out0_a * in0_a) \wedge$$
$$(out2_a = out1_a * in0_a)$$
$$\Phi_b \equiv \; out0_b = (in0_b * in0_b) * in0_b$$

- Show that:

$$(in0_a = in0_b) \wedge \Phi_a \wedge \Phi_b \rightarrow (out2_a = out0_b)$$

$(in0_a = in0_b) \wedge \Phi_a \wedge \Phi_b \wedge (out2_a \neq out0_b)$ is UNSAT.

Program Equivalence in SMT

```
int fun1(int y){
    int x[2];
    x[0] = y;
    y = x[1];
    x[1] = x[0]
    return x[1]*x[1];
}
```

```
int fun2(int y){

    return y*y;

}
```

- Show that the following formula is UNSAT:

$$x_1 = store(x, 0, y) \;\wedge\; y_1 = select(x_1, 1) \wedge$$
$$x_2 = store\big(x_1, 1, select(x_1, 0)\big) \;\wedge$$
$$ret_1 = select(x_2, 1) * select(x_2, 1) \wedge$$
$$ret_2 = y * y \wedge$$
$$ret_1 \neq ret_2$$

Logic Puzzle in SMT

- Someone who lived in Dreadbury Mansion killed aunt Agatha. Agatha, the butler and Charles were the only people who lived in Dreadbury Mansion. A killer always hates his victim, and is never richer than his victim. Charles hates no one that aunt Agatha hates. Agatha hates everyone except the butler. The butler hates everyone not richer than aunt Agatha. The butler also hates everyone Agatha hates. No one hates everyone. Agatha is not the butler.
  - Who killed aunt Agatha?

Logic Puzzle in SMT

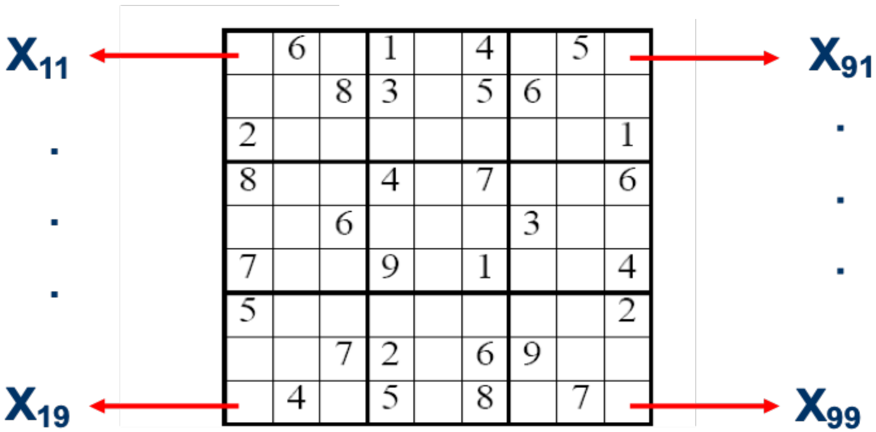
- Someone who lived in Dreadbury Mansion killed aunt Agatha. Agatha, the butler and Charles were the only people who lived in Dreadbury Mansion. A killer always hates his victim, and is never richer than his victim. Charles hates no one that aunt Agatha hates. Agatha hates everyone except the butler. The butler hates everyone not richer than aunt Agatha. The butler also hates everyone Agatha hates. No one hates everyone. Agatha is not the butler.
  - Who killed aunt Agatha?
  - Constants are blue.
  - Predicates are red.

Logic Puzzle in SMT

$killed/2, hates/2, richer/2, a/0, b/0, c/0$

$$\exists x \; killed(x, a)$$
$$\forall x \; \forall y \; killed(x, y) \rightarrow (hates(x, y) \land \neg richer(x, y))$$
$$\forall x \; hates(a, x) \rightarrow \neg hates(c, x)$$
$$hates(a, a) \land hates(a, c)$$
$$\forall x \; \neg richer(x, a) \rightarrow hates(b, x)$$
$$\forall x \; hates(a, x) \rightarrow hates(b, x)$$
$$\forall x \; \exists y \; \neg hates(x, y)$$
$$a \neq b$$

Sudoku



- Variables
  - 9x9 variables X_{ij} for each cell
- Constraints
  - Cell values: $1 \leq X_{ij} \wedge X_{ij} \leq 9$
  - Initial assignments. E.g., $X_{21} = 6$.
  - Difference constraints on all the rows, columns, and 3x3 boxes. E.g.,

    distinct($[X_{11}, X_{21}, X_{31}, \ldots, X_{91}]$) → expands to disequlaities $X_{ij} \neq X_{i'j'}$

    distinct($[X_{11}, X_{12}, X_{13}, \ldots, X_{19}]$)

    distinct($[X_{11}, X_{21}, X_{31}, X_{12}, X_{22}, X_{32}, X_{13}, X_{23}, X_{33}]$)

Disjunctive Scheduling in SMT

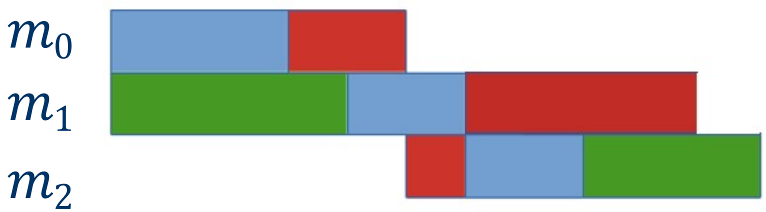
- Given:
  - n jobs where each job i is formed as a sequence of tasks t_{ij} to be performed in order;
  - m machines that can operate at most one task at a time;
  - machine requirements (m_{ij}) and durations (d_{ij}) of tasks t_{ij};
- decide:
  - when to execute each task so as to minimize the makespan, subject to temporal and resource constraints.

Disjunctive Scheduling in SMT

- Example
  - 3 jobs and 3 machines.
  - A task t_{ij} is described by (m_{ij}, d_{ij}):

$$j_0 = [(0,3),(1,2),(2,2)]$$
$$j_1 = [(0,2),(2,1),(1,4)]$$
$$j_2 = [(1,4),(2,3)]$$

  - Solution:



Total running time: 11

Disjunctive Scheduling in SMT

- Given

  $j_0 = [(0,3),(1,2),(2,2)],\ j_1 = [(0,2),(2,1),(1,4)],\ j_2 = [(1,4),(2,3)]$
- Variables
  - Task start time for each task t_{ij}: $t_{00}, t_{01}, t_{02}, t_{10}, t_{11}, t_{12}, t_{20}, t_{21}$
- Constraints
  - Start time values: $t_{ij} \geq 0$
  - Precedence among the successive tasks of each job, e.g.:
    - $(t_{00}+3 \leq t_{01} \wedge t_{01} + 2 \leq t_{02})$
  - Disjunction between each pair of tasks requiring the same machine, e.g.:
    - $(t_{00}+3 \leq t_{01}) \vee (t_{01}+2 \leq t_{02})$
- Objective
  - Minimize the makespan: $mimize(max(t_{02} + 2, t_{12} + 4, t_{21} + 3))$