IC3: Where Monolithic and Incremental Meet

Fabio Somenzi Aaron R. Bradley

Department of Electrical, Computer, and Energy Engineering
University of Colorado at Boulder

FMCAD, 30 October 2011

Outline

Outline

1. Proving Invariants by Induction
   - Induction for Transition Systems
   - Strengthening
   - Relative Induction

2. IC3
   - Basic Algorithm
   - Examples
   - Efficiency

Finite-State Transition Systems

IC3 works on a symbolic representation of a system:

$$S : (\bar{i},\ \bar{x},\ I(\bar{x}),\ T(\bar{i},\bar{x},\bar{x}'))$$

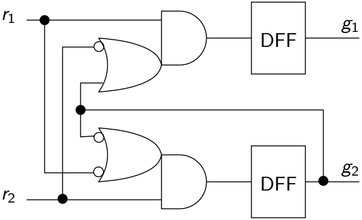
- $\bar{i}$: primary inputs
- $\bar{x}$: state variables
- $\bar{x}'$: next state variables
- $I(\bar{x})$: initial states
- $T(\bar{i},\bar{x},\bar{x}')$: transition relation

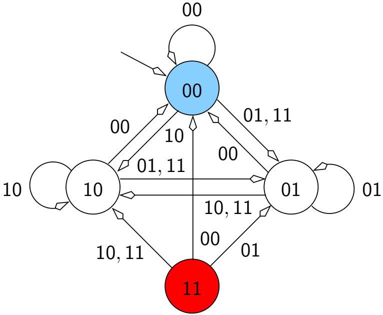
Invariance Properties

IC3 proves (or refutes) invariants

- Prove that every reachable state satisfies $P(\overline{x})$
  - P is a propositional formula
- Checking safety properties is reduced to checking invariance properties

Mutual Exclusion for a Simple Arbiter





$$I(\overline{g}) = \neg g_1 \wedge \neg g_2$$

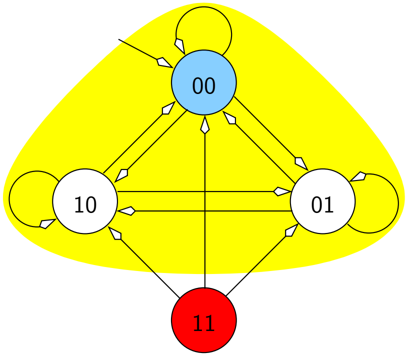
$$\exists r_1, r_2 \,.\, T(\overline{r}, \overline{g}, \overline{g}') = \neg g_1' \vee \neg g_2'$$

$$P(\overline{g}) = \neg g_1 \vee \neg g_2$$

Inductive Proofs for Transition Systems

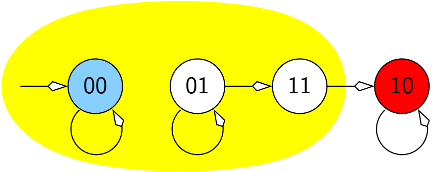
- Prove initiation (base case)
  - $I(\overline{x}) \Rightarrow P(\overline{x})$
  - All initial states satisfy P
  - $(\neg g_1 \wedge \neg g_2) \Rightarrow (\neg g_1 \vee \neg g_2)$
- Prove consecution (inductive step)
  - $P(\overline{x}) \wedge T(\overline{i}, \overline{x}, \overline{x}') \Rightarrow P(\overline{x}')$
  - All successors of states satisfying P satisfy P
  - $(\neg g_1 \vee \neg g_2) \wedge (\neg g_1' \vee \neg g_2') \Rightarrow (\neg g_1' \vee \neg g_2')$
- If both pass, all reachable states satisfy the property
  - $S \models P$

Visualizing Inductive Proofs

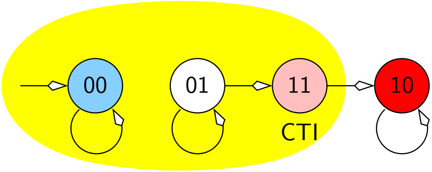


The inductive assertion (∼yellow) contains all initial (blue) states and no arrow leaves it (it is closed under the transition relation)

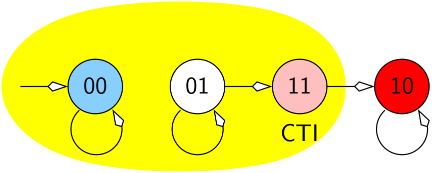
Counterexamples to Induction: The Troublemakers



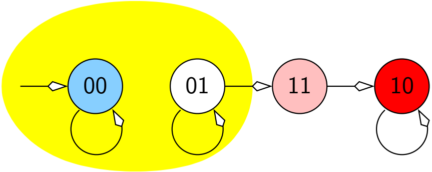
Counterexamples to Induction: The Troublemakers



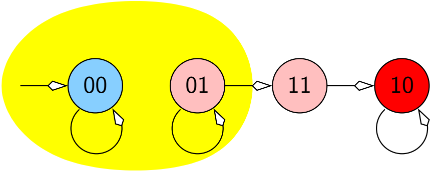
Invariant Strengthening



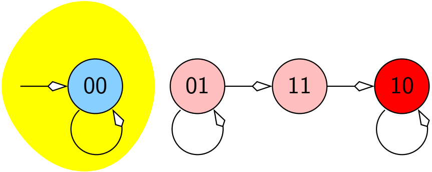
Invariant Strengthening



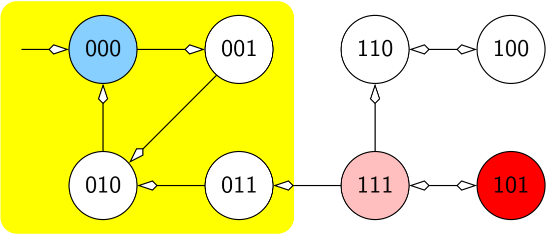
Invariant Strengthening



Invariant Strengthening



Strong and Weak Invariants



Induction is not restricted to:

- the strongest inductive invariant (forward-reachable states)
- . . . or the weakest inductive invariant (complement of the backward-reachable states)
- $\neg x_1$ is simpler than $\neg x_1 \wedge (\neg x_2 \vee \neg x_3)$ (strongest) and $(\neg x_1 \vee \neg x_3)$ (weakest)

Completeness for Finite-State Systems

- CTIs are effectively bad states
  - If a CTI is reachable so is at least one bad state
- Remove CTI from P and try again
- Eventually either:
  - An inductive strengthening of P results
  - An initial state is removed from P
- In the latter case, a counterexample is obtained

Examples of Strengthening Strategies

- Removing one CTI at a time is very inefficient!
    - Several strategies in use to avoid that
- Fixpoint-based invariant checking: if $\nu Z \,.\, p \wedge \mathrm{AX}\, Z$ converges in $n > 0$ iterations, then $\bigwedge_{0 \leq i < n} \mathrm{AX}^i\, p$ is an inductive invariant
    - In fact, the weakest inductive invariant
- k-induction: if all states on length-k paths from the initial states satisfy p, and k distinct consecutive states satisfying p are always followed by a state satisfying p, then all states reachable from the initial states satisfy p.
- fsis algorithm: try to extract an inductive clause from CTI to exclude multiple CTIs

Relative Induction

Suppose the assertion φ is a conjunction

$$\varphi = \bigwedge_{0 \leq j < n} \varphi_j$$

Suppose each φ_j is inductive relative to the previous assertions and P. That is, for every $0 \leq j < n$, $I \Rightarrow \varphi_j$ and

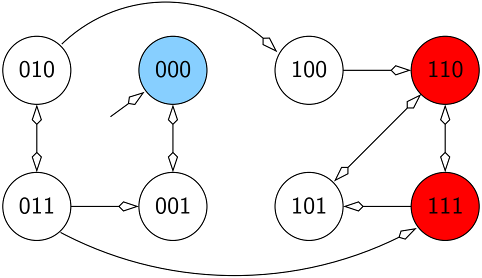
$$P \wedge \bigwedge_{0 \leq i \leq j} \varphi_i \wedge T \Rightarrow \varphi_j'$$

Finally, suppose P is inductive relative to φ; that is, $I \Rightarrow P$ and

$$P \wedge \bigwedge_{0 \leq i < n} \varphi_i \wedge T \Rightarrow P'$$

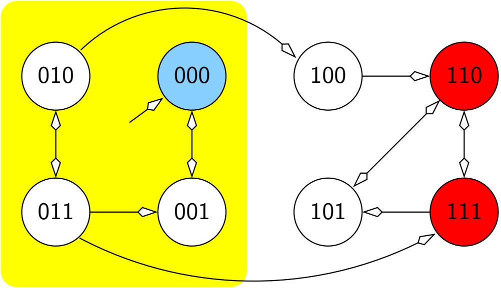
Then P is an invariant of S

Relative Induction



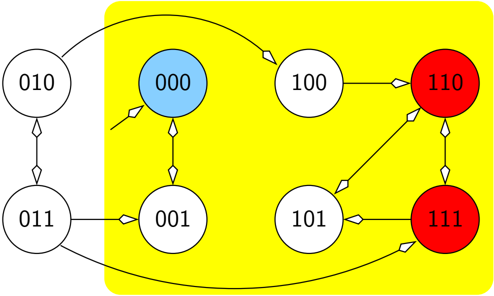
$$\varphi = \neg x_1 \wedge (x_1 \vee \neg x_2)$$

Relative Induction



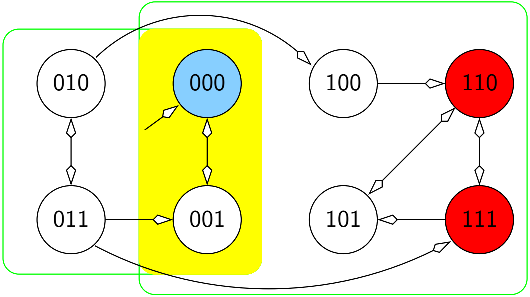
$\neg x_1$ is not inductive

Relative Induction



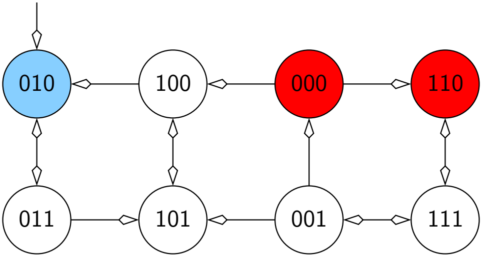
$x_1 \vee \neg x_2$ is inductive

Relative Induction



$\neg x_1$ is inductive relative to $x_1 \vee \neg x_2$

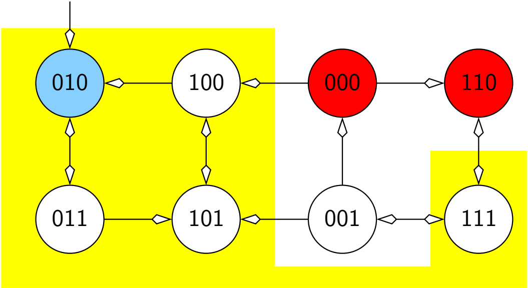
Shortcoming of Relative Induction



$$P = (x_1 \lor x_2 \lor x_3) \land (\neg x_1 \lor \neg x_2 \lor x_3)$$

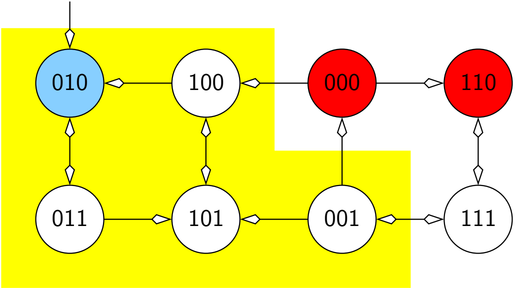
$$\varphi = (x_1 \lor x_2) \land (\neg x_1 \lor \neg x_2)$$

Shortcoming of Relative Induction



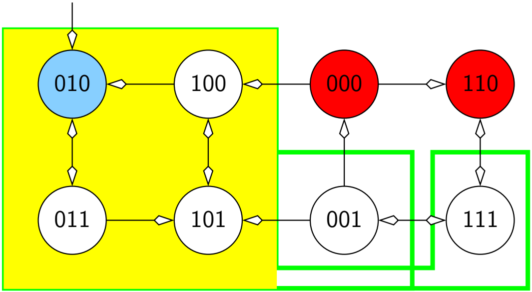
$$(x_1 \vee x_2) \wedge P \wedge T \not\Rightarrow (x_1' \vee x_2')$$

Shortcoming of Relative Induction



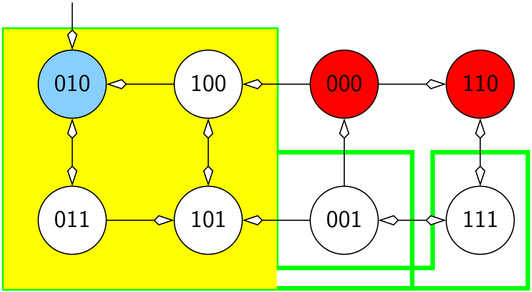
$$(\neg x_1 \vee \neg x_2) \wedge P \wedge T \not\Rightarrow (\neg x_1' \vee \neg x_2')$$

Shortcoming of Relative Induction



$$(x_1 \vee x_2) \wedge (\neg x_1 \vee \neg x_2) \wedge P \wedge T \Rightarrow (x_1' \vee x_2') \wedge (\neg x_1' \vee \neg x_2')$$

Shortcoming of Relative Induction



$(x_1 \vee x_2)$ and $(\neg x_1 \vee \neg x_2)$are mutually inductive

Outline

What Does IC3 Stand for?

- Incremental Construction of
- Inductive Clauses for
- Indubitable Correctness

Basic Tenets

- Approximate reachability assumptions
  - F_i: contains at least all the states reachable in i steps or less
  - If $S \models P$, F_i eventually becomes inductive for some i
  - Approximation is desirable: IC3 does not attempt to get the most precise F_i's
- Stepwise relative induction
  - Learn useful facts via induction relative to reachability assumptions
- Clausal representation
  - Learn clauses from CTIs
  - A form of abstract interpretation

IC3 Invariants

- The four main invariants of IC3.

$$
\begin{aligned}
I &\Rightarrow F_0 & \\
F_i &\Rightarrow F_{i+1} & 0 \le i < k \\
F_i &\Rightarrow P & 0 \le i \le k \\
F_i \wedge T &\Rightarrow F'_{i+1} & 0 \le i < k
\end{aligned}
$$

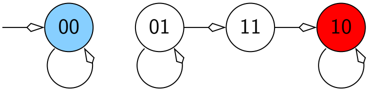
- Established if there are no counterexamples of length 0 or 1
- The implicit invariant of the outer loop: no counterexamples of length k.

Pseudo-Pseudocode

```
bool IC3 {
    if (I ⇏ P or I ∧ T ⇏ P')
        return ⊥;
    F_0 = I; F_1 = P; k = 1
    repeat {
        while (there are CTIs in F_k) {
            either find a counterexample and return ⊥
            or refine F_1, ..., F_k
        }
        k++;
        set F_k = P and propagate clauses
        if (F_i = F_{i+1} for some 0 < i < k)
            return ⊤
    }
}
```

Passing Property

No counterexamples of length 0 or 1

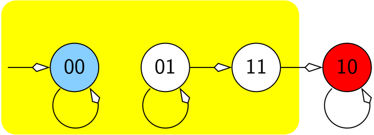


$$I = \neg x_1 \wedge \neg x_2$$
$$P = \neg x_1 \vee x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

Does $F_1 \wedge T \Rightarrow P'$?

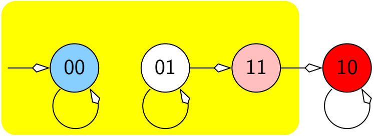


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = P = \neg x_1 \vee x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

Found CTI $s = x_1 \wedge x_2$

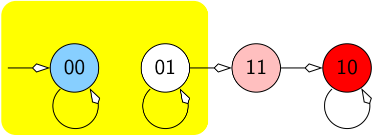


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = P = \neg x_1 \vee x_2$$

$$
\begin{aligned}
I &\Rightarrow F_0 & & \\
F_i &\Rightarrow F_{i+1} & & 0 \leq i < k \\
F_i &\Rightarrow P & & 0 \leq i \leq k \\
F_i \wedge T &\Rightarrow F'_{i+1} & & 0 \leq i < k
\end{aligned}
$$

Passing Property

Is $\neg s$ inductive relative to F_1?

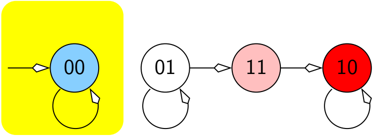


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = P = \neg x_1 \vee x_2$$

$$\begin{aligned} I &\Rightarrow F_0 & & \\ F_i &\Rightarrow F_{i+1} & & 0 \leq i < k \\ F_i &\Rightarrow P & & 0 \leq i \leq k \\ F_i \wedge T &\Rightarrow F'_{i+1} & & 0 \leq i < k \end{aligned}$$

Passing Property

No. Is $\neg s$ inductive relative to F_0?

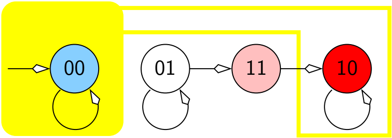


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = P = \neg x_1 \vee x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

Yes. Generalize $\neg s$ at level 0 (in one of the two possible ways)

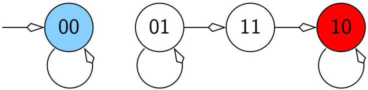


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = P = \neg x_1 \vee x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

Update F_1

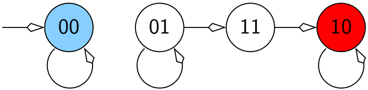


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = (\neg x_1 \vee x_2) \wedge \neg x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

No more CTIs in F_1. No counterexamples of length 2. Instantiate F_2

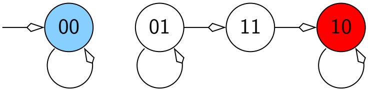


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = (\neg x_1 \vee x_2) \wedge \neg x_2$$
$$F_2 = P = \neg x_1 \vee x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

Propagate clauses from F_1 to F_2

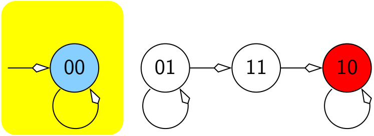


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = (\neg x_1 \vee x_2) \wedge \neg x_2$$
$$F_2 = (\neg x_1 \vee x_2) \wedge \neg x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

F_1 and F_2 are identical. Property proved

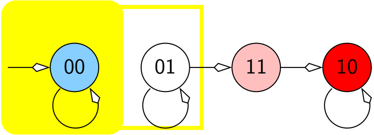


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = (\neg x_1 \vee x_2) \wedge \neg x_2$$
$$F_2 = (\neg x_1 \vee x_2) \wedge \neg x_2$$

$$
\begin{aligned}
I &\Rightarrow F_0 & & \\
F_i &\Rightarrow F_{i+1} & & 0 \le i < k \\
F_i &\Rightarrow P & & 0 \le i \le k \\
F_i \wedge T &\Rightarrow F'_{i+1} & & 0 \le i < k
\end{aligned}
$$

Passing Property

What happens if we generalize $\neg s$ at level 0 in the other way?

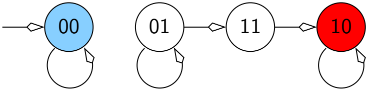


$F_0 = I = \neg x_1 \wedge \neg x_2$
$F_1 = \neg x_1 \vee x_2$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

Update F_1

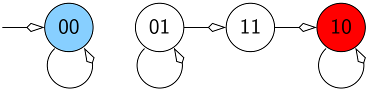


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = (\neg x_1 \vee x_2) \wedge \neg x_1$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \le i < k$$
$$F_i \Rightarrow P \qquad 0 \le i \le k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \le i < k$$

Passing Property

No more CTIs in F_1. No counterexamples of length 2. Instantiate F_2

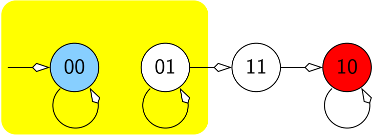


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = (\neg x_1 \vee x_2) \wedge \neg x_1$$
$$F_2 = P = \neg x_1 \vee x_2$$

$$
\begin{aligned}
I &\Rightarrow F_0 & & \\
F_i &\Rightarrow F_{i+1} & & 0 \leq i < k \\
F_i &\Rightarrow P & & 0 \leq i \leq k \\
F_i \wedge T &\Rightarrow F'_{i+1} & & 0 \leq i < k
\end{aligned}
$$

Passing Property

No clauses propagate from F_1 to F_2

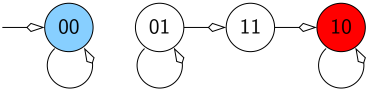


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = (\neg x_1 \vee x_2) \wedge \neg x_1$$
$$F_2 = P = \neg x_1 \vee x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

Remove subsumed clauses

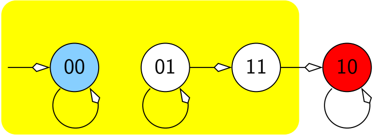


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = \neg x_1$$
$$F_2 = P = \neg x_1 \vee x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

Does $F_2 \wedge T \Rightarrow P'$?

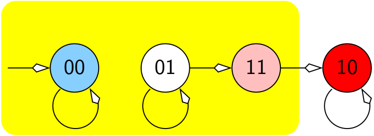


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = \neg x_1$$
$$F_2 = P = \neg x_1 \vee x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \le i < k$$
$$F_i \Rightarrow P \qquad 0 \le i \le k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \le i < k$$

Passing Property

Found CTI $s = x_1 \wedge x_2$ (same as before)

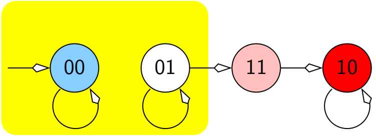


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = \neg x_1$$
$$F_2 = P = \neg x_1 \vee x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

Is $\neg s$ inductive relative to F_1?

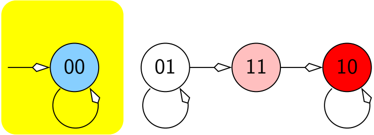


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = \neg x_1$$
$$F_2 = P = \neg x_1 \vee x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

No. We know it is inductive at level 0.

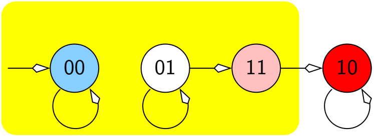


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = \neg x_1$$
$$F_2 = P = \neg x_1 \vee x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

If generalization produces $\neg x_1$ again, the CTI is not eliminated

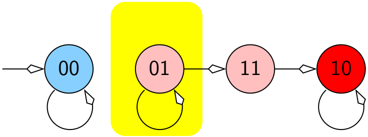


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = \neg x_1$$
$$F_2 = P = \neg x_1 \vee x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \le i < k$$
$$F_i \Rightarrow P \qquad 0 \le i \le k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \le i < k$$

Passing Property

Find predecessor t of CTI in $F_1 \setminus F_0$

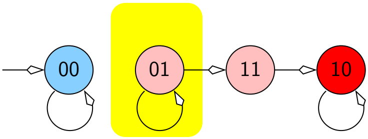


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = \neg x_1$$
$$F_2 = P = \neg x_1 \vee x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

Found $t = \neg x_1 \wedge x_2$

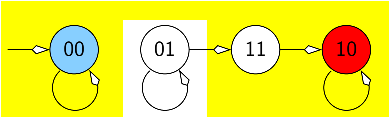


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = \neg x_1$$
$$F_2 = P = \neg x_1 \vee x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \quad 0 \leq i < k$$
$$F_i \Rightarrow P \quad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \quad 0 \leq i < k$$

Passing Property

The clause $\neg t$ is inductive at all levels

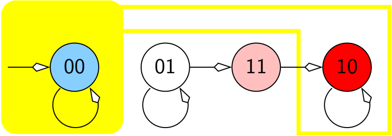


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = \neg x_1$$
$$F_2 = P = \neg x_1 \vee x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

Generalization of $\neg t$ produces $\neg x_2$

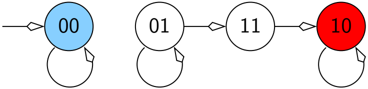


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = \neg x_1$$
$$F_2 = P = \neg x_1 \vee x_2$$

$$
\begin{array}{rcll}
I & \Rightarrow & F_0 & \\
F_i & \Rightarrow & F_{i+1} & 0 \leq i < k \\
F_i & \Rightarrow & P & 0 \leq i \leq k \\
F_i \wedge T & \Rightarrow & F'_{i+1} & 0 \leq i < k
\end{array}
$$

Passing Property

Update F_1 and F_2

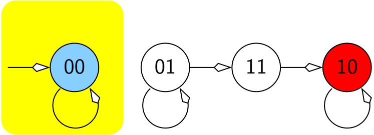


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = \neg x_1 \wedge \neg x_2$$
$$F_2 = (\neg x_1 \vee x_2) \wedge \neg x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Passing Property

F_1 and F_2 are equivalent. Property (almost) proved

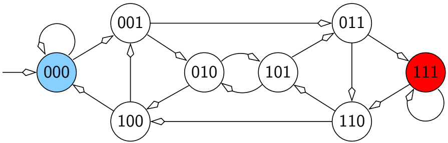


$$F_0 = I = \neg x_1 \wedge \neg x_2$$
$$F_1 = \neg x_1 \wedge \neg x_2$$
$$F_2 = (\neg x_1 \vee x_2) \wedge \neg x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Failing Property

No counterexamples of length 0 or 1

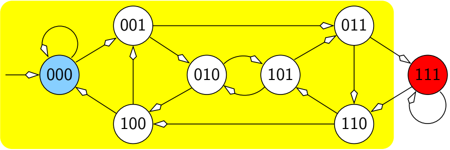


$$I = \neg x_1 \wedge \neg x_3 \wedge \neg x_3$$
$$P = \neg x_1 \vee \neg x_2 \vee \neg x_3$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Failing Property

Does $F_1 \wedge T \Rightarrow P'$?

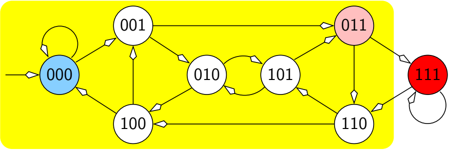


$$F_0 = I = \neg x_1 \wedge \neg x_3 \wedge \neg x_3$$
$$F_1 = P = \neg x_1 \vee \neg x_2 \vee \neg x_3$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Failing Property

Found CTI $s = \neg x_1 \wedge x_2 \wedge x_3$



$F_0 = I = \neg x_1 \wedge \neg x_3 \wedge \neg x_3$

$F_1 = P = \neg x_1 \vee \neg x_2 \vee \neg x_3$

$$I \Rightarrow F_0$$

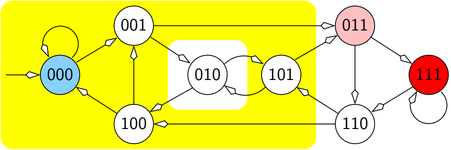
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$

$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$

$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Failing Property

The clause $\neg s$ generalizes to $\neg x_2$ at level 0

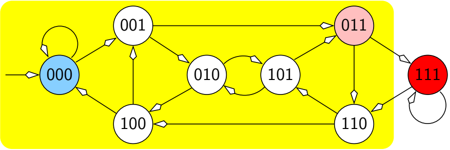


$$F_0 = I = \neg x_1 \wedge \neg x_3 \wedge \neg x_3$$
$$F_1 = (\neg x_1 \vee \neg x_2 \vee \neg x_3) \wedge \neg x_2$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Failing Property

No CTI left: no counterexample of length 2. F_2 instantiated, but no clause propagated

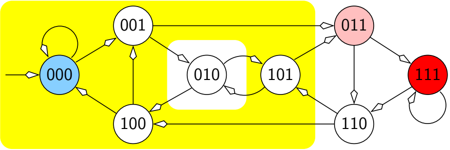


$$F_0 = I = \neg x_1 \wedge \neg x_3 \wedge \neg x_3$$
$$F_1 = \neg x_2$$
$$F_2 = P = \neg x_1 \vee \neg x_2 \vee \neg x_3$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Failing Property

The clause $\neg s$ generalizes again to $\neg x_2$ at level 0

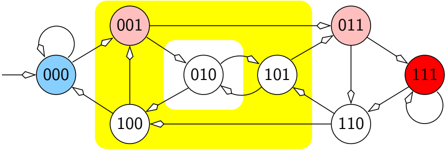


$$F_0 = I = \neg x_1 \wedge \neg x_3 \wedge \neg x_3$$
$$F_1 = \neg x_2$$
$$F_2 = P = \neg x_1 \vee \neg x_2 \vee \neg x_3$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Failing Property

Suppose IC3 recurs on $t = \neg x_1 \wedge \neg x_2 \wedge x_3$ in $F_1 \setminus F_0$

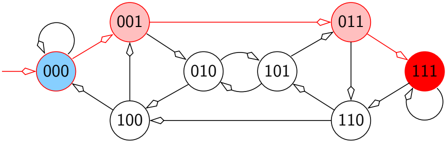


$$F_0 = I = \neg x_1 \wedge \neg x_3 \wedge \neg x_3$$
$$F_1 = \neg x_2$$
$$F_2 = P = \neg x_1 \vee \neg x_2 \vee \neg x_3$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Failing Property

Clause $\neg t$ is not inductive at level 0: the property fails

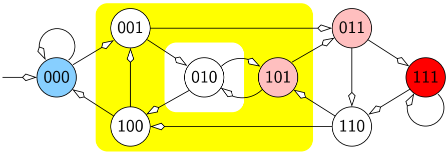


$$F_0 = I = \neg x_1 \wedge \neg x_3 \wedge \neg x_3$$
$$F_1 = \neg x_2$$
$$F_2 = P = \neg x_1 \vee \neg x_2 \vee \neg x_3$$

$$
\begin{aligned}
I &\Rightarrow F_0 & & \\
F_i &\Rightarrow F_{i+1} & & 0 \le i < k \\
F_i &\Rightarrow P & & 0 \le i \le k \\
F_i \wedge T &\Rightarrow F'_{i+1} & & 0 \le i < k
\end{aligned}
$$

Failing Property

Suppose now IC3 recurs on $t = x_1 \wedge \neg x_2 \wedge x_3$ in $F_1 \setminus F_0$

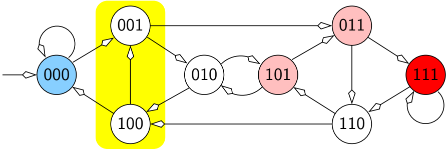


$$F_0 = I = \neg x_1 \wedge \neg x_3 \wedge \neg x_3$$
$$F_1 = \neg x_2$$
$$F_2 = P = \neg x_1 \vee \neg x_2 \vee \neg x_3$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Failing Property

Clause $\neg t$ is inductive at level 1

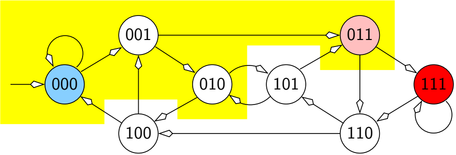


$$F_0 = I = \neg x_1 \wedge \neg x_3 \wedge \neg x_3$$
$$F_1 = \neg x_2$$
$$F_2 = P = \neg x_1 \vee \neg x_2 \vee \neg x_3$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Failing Property

Generalization of $\neg t$ adds $\neg x_1$ to F_1 and F_2

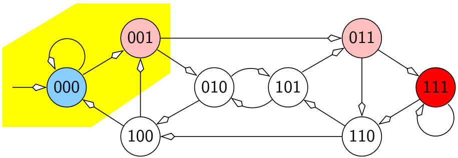


$$F_0 = I = \neg x_1 \wedge \neg x_3 \wedge \neg x_3$$
$$F_1 = \neg x_2 \wedge \neg x_1$$
$$F_2 = (\neg x_1 \vee \neg x_2 \vee \neg x_3) \wedge \neg x_1$$

$$I \Rightarrow F_0$$
$$F_i \Rightarrow F_{i+1} \qquad 0 \leq i < k$$
$$F_i \Rightarrow P \qquad 0 \leq i \leq k$$
$$F_i \wedge T \Rightarrow F'_{i+1} \qquad 0 \leq i < k$$

Failing Property

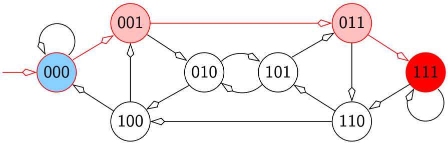
Only $t = \neg x_1 \wedge \neg x_2 \wedge x_3$ remains in $F_1 \setminus F_0$



$$
\begin{aligned}
I &\Rightarrow F_0 & \\
F_i &\Rightarrow F_{i+1} & 0 \leq i < k \\
F_i &\Rightarrow P & 0 \leq i \leq k \\
F_i \wedge T &\Rightarrow F'_{i+1} & 0 \leq i < k
\end{aligned}
$$

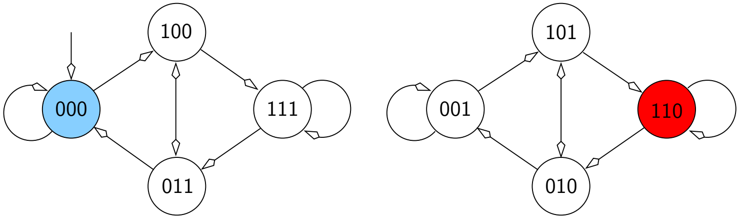
Failing Property

The same counterexample as before is found



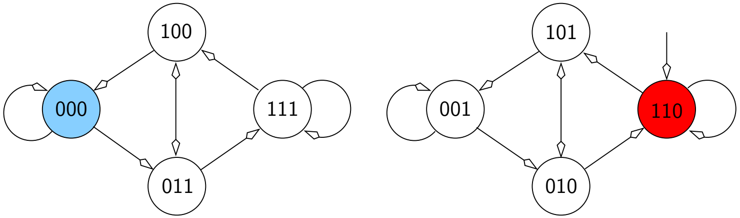
$$
\begin{aligned}
I &\Rightarrow F_0 & \\
F_i &\Rightarrow F_{i+1} & 0 \le i < k \\
F_i &\Rightarrow P & 0 \le i \le k \\
F_i \land T &\Rightarrow F'_{i+1} & 0 \le i < k
\end{aligned}
$$

Reverse IC3



Build reachability assumptions around the target

Reverse IC3



Equivalent to reversing all transitions

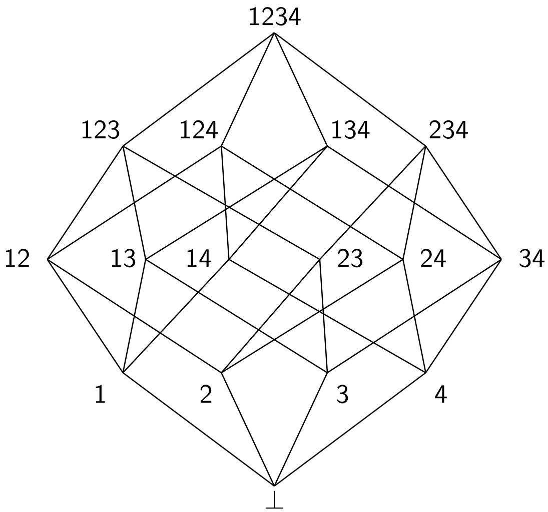
Clause Generalization

- A CTI is a cube
  - e.g., $s = x_1 \wedge \neg x_2 \wedge x_3$
- The negation of a CTI is a clause
  - e.g., $\neg s = \neg x_1 \vee x_2 \vee \neg x_3$
- Conjoining $\neg s$ to a reachability assumption F_i excludes the CTI from it
- Generalization extracts a subclause from $\neg s$ that excludes more states that are "like the CTI"
  - e.g., $\neg x_3$ may be a subclause of $\neg s$ that excludes states that, like the CTI, are not reachable in i steps
  - Every literal dropped doubles the number of states excluded by a clause
  - Generalization is time-consuming, but critical to performance

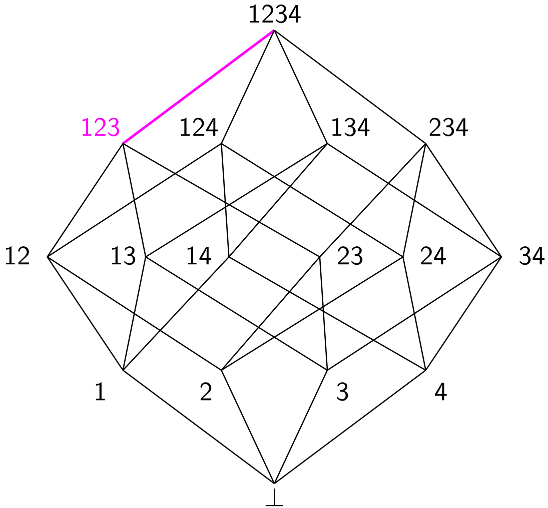
Generalization

- Crucial for efficiency
- Generalization in IC3 produces a minimal inductive clause (MIC)
- The MIC algorithm is based on DOWN and UP.
- DOWN extracts the (unique) maximal subclause
- UP finds a small, but not necessarily minimal subclause
- MIC recurs on subclauses of the result of UP

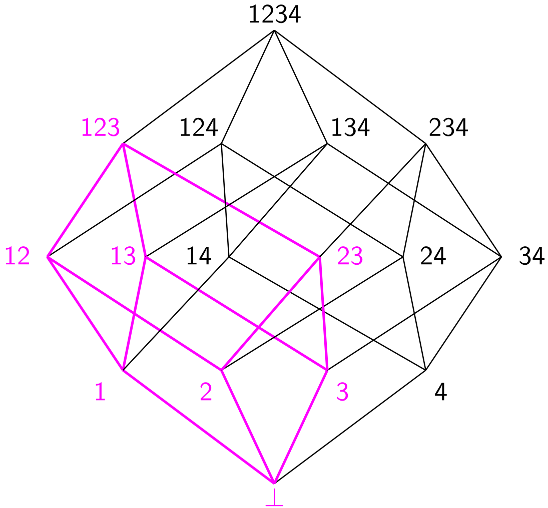
Minimal Inductive Clause



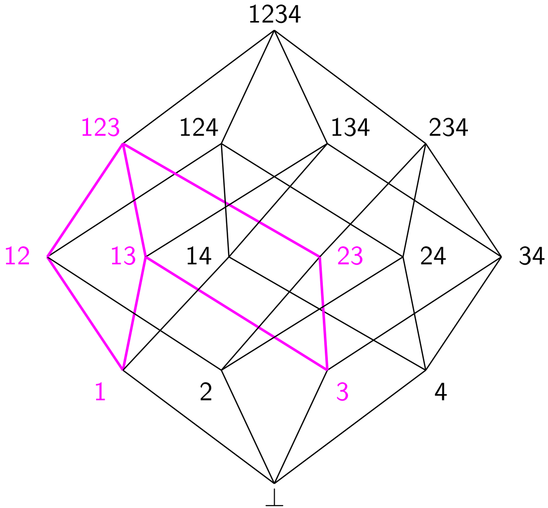
Minimal Inductive Clause



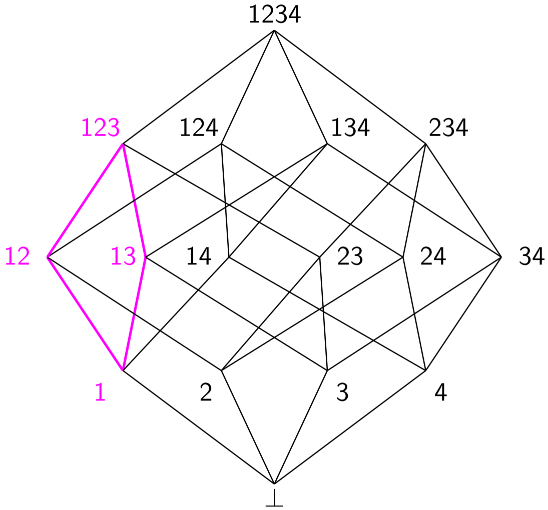
Minimal Inductive Clause



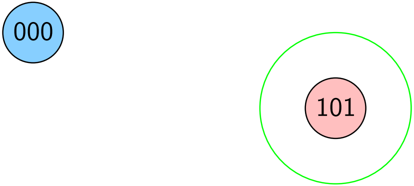
Minimal Inductive Clause



Minimal Inductive Clause

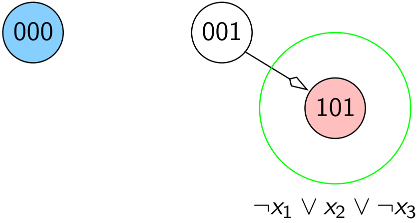


Maximal Inductive Subclause (DOWN)

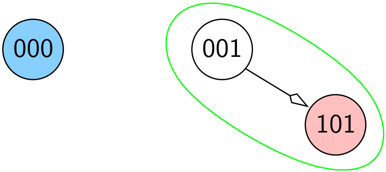


$\neg x_1 \vee x_2 \vee \neg x_3$

Maximal Inductive Subclause (DOWN)

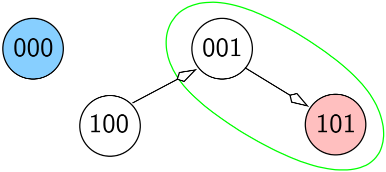


Maximal Inductive Subclause (DOWN)



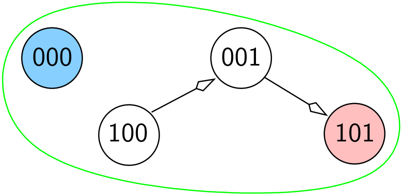
$x_2 \lor \neg x_3$

Maximal Inductive Subclause (DOWN)



$x_2 \vee \neg x_3$

Maximal Inductive Subclause (DOWN)



x_2

Use of UNSAT Cores

- $\neg s \wedge F_i \wedge T \Rightarrow \neg s'$ if and only if $\neg s \wedge F_i \wedge T \wedge s'$ is unsatisfiable
- The literals of s' are (unit) clauses in the SAT query
- If the implication holds, the SAT solver returns an unsatisfiable core
- Any literal of s' not in the core can be removed from s' because it does not contribute to the implication ...
- and from $\neg s$ because strengthening the antecedent preserves the implication

Use of UNSAT Core Example

- $\neg s \wedge F_0 \wedge T \Rightarrow \neg s'$ with

$$\begin{aligned} \neg s &= \neg x_1 \vee \neg x_2 \\ F_0 &= \neg x_1 \wedge \neg x_2 \\ T &= (\neg x_1 \wedge \neg x_2 \wedge \neg x_1' \wedge \neg x_2') \vee \cdots \end{aligned}$$

- The SAT query, after some simplification, is

$$\neg x_1 \wedge \neg x_2 \wedge \neg x_1' \wedge \neg x_2' \wedge x_1' \wedge x_2'$$

- Two UNSAT cores are

$$\begin{aligned} &\neg x_1' \wedge x_1' \\ &\neg x_2' \wedge x_2' \end{aligned}$$

from which the two generalizations we saw before follow

Clause Clean-Up

- As IC3 proceeds, clauses may be added to some F_is that subsume other clauses
- The weaker, subsumed clauses no longer contribute to the definition of F_i
- However, a weaker clause may propagate to F_{i+1} when the stronger clause does not
- Weak clauses are eliminated by subsumption only between major iterations and after propagation

More Efficiency-Related Issues

- State encoding determines what clauses are derived
- Incremental vs. monolithic
  - Reachability assumptions carry global information
  - ... but are built incrementally
- Semantic vs. syntactic approach
  - Generalization "jumps over large distances"
- Long counterexamples at low k
  - Typically more efficient than increasing k
- Consequences of no unrolling
  - Many cheap (incremental) SAT calls
- Ability to parallelize
  - Clauses are easy to exchange

IC3 and Interpolation

- An interesting analysis to be presented on Tuesday by Een, Mishchenko, and Brayton
- In the tutorial paper:
  - Both methods address the failure of consecution from an over-approximating *i*-step set.
  - Interpolation unrolls to produce an (interpolant-based) abstract post operator. When consecution fails, a greater unrolling refines the abstract post operator, yielding more refined over-approximating stepwise sets.
  - IC3 uses the CTI from the failure to direct the refinement of F_i (and $F_1, \ldots, F_{i-1}$).
  - In other words, they focus on refining **different parts of consecution**.
  - IC3 is more incremental and does not require unrolling the transition relation.

Applications

Checking all ω-regular properties

- Cycle detection reduced to several reachability queries
- Inductive proofs of unreachability refine partition of state space into SCC-closed regions

Incremental verification

- A proof from one revision of a circuit provides a starting point for the proof of the next revision
- Same for counterexample
- Some “patching” may be needed

More coming

Bibliography I

- A. R. Bradley, *k*-step relative inductive generalization," CU Boulder, Tech. Rep., March 2010, `http://arxiv.org/abs/1003.3649`.
- A. R. Bradley, "SAT-based model checking without unrolling," in *Verification, Model Checking, and Abstract Interpretation (VMCAI'11)*, Austin, TX, 2011, pp. 70–87, LNCS 6538.
- Z. Manna and A. Pnueli, *Temporal Verification of Reactive Systems: Safety*. Springer-Verlag, 1995.
- A. R. Bradley and Z. Manna, "Checking safety by inductive generalization of counterexamples to induction," in *Formal Methods in Computer Aided Design (FMCAD'07)*, Austin, TX, 2007, pp. 173–180.

Bibliography II (Fresh from the Oven)

- N. Een, A. Mishchenko, and R. K. Brayton, "Efficient Implementation of Property Directed Reachability," in *Formal Methods in Computer Aided Design (FMCAD'11)*, Austin, TX, 2011.
- H. Chockler, A. Ivrii, A. Matsliah, S. Moran, and Z. Nevo, "Incremental Formal Verification of Hardware," in *Formal Methods in Computer Aided Design (FMCAD'11)*, Austin, TX, 2011.
- A. R. Bradley, F. Somenzi, Z. Hassan, and Y. Zhang, "An incremental approach to model checking progress properties," in *Formal Methods in Computer Aided Design (FMCAD'11)*, Austin, TX, 2011.