Notation

\[ S : (\overline{x}, \overline{i}, I(\overline{x}), T(\overline{x}, \overline{i}, \overline{x}')) \] Invariant property : \( P \)

- \( \overline{x} \): State variables
- \( \overline{i} \): Inputs
- \( I(\overline{x}) \): Initial condition
- \( T(\overline{x}, \overline{i}, \overline{x}') \): Transition relation
- \( P(\overline{x}) \): Invariant property (“good states”)

**Problem**: Show all reachable states satisfy \( P \)
SAT-Based Model Checking:

*Just Unroll*
Bounded Model Checking (BMC)

For $k = 0, 1, 2, \ldots$, SAT query:

$$I(x_0) \land \bigwedge_{j=1}^{k} T(x_{j-1}, i_{j-1}, x_j) \land \neg P(x_k)$$

until an error is found or the diameter is reached.
Induction

Mathematical induction over $S$:

- $I(\overline{x}) \Rightarrow P(\overline{x})$ (initiation)
- $P(\overline{x}) \land T(\overline{x}, \overline{i}, \overline{x'}) \Rightarrow P(\overline{x'})$ (consecution)

Failure does not imply that $P$ does not hold.

Inductive strengthening: $F$ such that $F \land P$ is inductive
$P$ is inductive
$k$-Induction

Initiation: (BMC)

\[ I(\bar{x}_0) \land \bigwedge_{j=1}^{k} T(\bar{x}_{j-1}, \bar{i}_{j-1}, \bar{x}_j) \Rightarrow P(\bar{x}_k) \]

Consecution:

\[ \text{LoopFree} \land \bigwedge_{j=1}^{k} \left( P(\bar{x}_{j-1}) \land T(\bar{x}_{j-1}, \bar{i}_{j-1}, \bar{x}_j) \right) \Rightarrow P(\bar{x}_k) \]
Longest loop-free path

\[ k = 6 \]

k-Induction
Interpolant-based Model Checking (ITP)

Post-condition operator:

\[
\text{post}(F)(\overline{x}) = \exists x_0, i_0. \ F(x_0) \land T(x_0, i_0, \overline{x})
\]

Abstract post-condition operator:

\[
\text{post}(F)(\overline{x}) \Rightarrow \widehat{\text{post}}(F)(\overline{x})
\]
Interpolant-based Model Checking (ITP)

If this query is UNSAT

\[
F(\overline{x}_0) \land \bigwedge_{j=1}^{k} T(\overline{x}_{j-1}, \overline{i}_{j-1}, \overline{x}_j) \Rightarrow P(\overline{x}_k)
\]

then extract \( G \) such that

\[
F(\overline{x}_0) \land T(\overline{x}_0, \overline{i}_0, \overline{x}_1) \Rightarrow G(\overline{x}_1)
\]

and

\[
G(\overline{x}_1) \land \bigwedge_{j=2}^{k} T(\overline{x}_{j-1}, \overline{i}_{j-1}, \overline{x}_j) \Rightarrow P(\overline{x}_k)
\]

Then

\[
\hat{\text{post}}(F)(\overline{x}) := G(\overline{x})
\]
SAT-Based Model Checking:  
*Don’t Unroll!*
Next K! Get in line!

Mommy!
Incremental

Monolithic
Impossible?
Yes, but is there a simple reason?

FSIS (Finite State Inductive Strengthening)
- FMCA 2007
- From backward reachable state s to $c \leq t$ s.t.
  \[ F \land c \land T \Rightarrow c' \] (and $I \Rightarrow c$)
- On top of explicit backward enumeration.

\textit{Lasting idea:}
Relative Inductive Generalization
Inductive Generalization

**Given:** cube $s$ (usually based on backward-reachable state)

**Find:** $c \subseteq \neg s$ such that

- **Initiation:**
  \[ I(x) \Rightarrow c(x) \]

- **Consecution (relative to information $G$):**
  \[ G(x) \land c(x) \land T(x, i, x') \Rightarrow c(x') \]

- **Minimality:** No strict subclause of $c$ is inductive relative to $G$
Use inductive generalization to incrementally construct over-approximating sets.

\( F_i \): over-approximates set of states reachable in at most \( i \) steps

Four invariants:

1. \( I(\overline{x}) \Rightarrow F_0(\overline{x}) \)
2. \( \forall i. \ F_i(\overline{x}) \Rightarrow F_{i+1}(\overline{x}) \)
3. \( \forall i. \ F_i(\overline{x}) \land T(\overline{x}, \overline{i}, \overline{x}') \Rightarrow F_{i+1}(\overline{x}') \)
4. \( \forall i \leq k. \ F_i(\overline{x}) \Rightarrow P(\overline{x}) \)

\(^1\)Incremental Construction of Inductive Clauses for Indubitable Correctness
Sometimes called Property Directed Reachability (PDR)
Refinement: In response to proof obligation \( \langle s, j \rangle \),

- Attempt inductive generalization relative to \( F_j \): \( c \subseteq \neg s \)
- Success: Conjoin \( c \) to \( F_1, \ldots, F_{j+1} \)
- Failure:
  - Predecessor \( t \)
  - Enqueue new obligation \( \langle t, j - 1 \rangle \)
When

\[ F_k \land T(\overline{x}, \overline{i}, \overline{x}') \Rightarrow P(\overline{x}') \]

- Propagate clauses forward with relative induction
- Increment \( k \) (unless converged)
\[ F_k (\land P) \land T \Rightarrow P' \]
Converges when $\exists j \leq k. F_j = F_{j+1}$. Then:

1. $I(\overline{x}) \Rightarrow F_j(\overline{x})$
2. $F_j(\overline{x}) \land T(\overline{x}, \overline{i}, \overline{x}') \Rightarrow F_j(\overline{x}')$
3. $F_j(\overline{x}) \Rightarrow P(\overline{x})$

$\therefore F_j$ is an inductive strengthening of $P$. 
Research Inspired by IC3: 
*Incremental, Inductive Model Checking*
Improvements/Extensions to IC3

- Lift predecessor state $s$ to set of predecessors $\bar{s}$:
  - with kCOI, statically (original paper)
  - with ternary simulation [Een et al. '11]
  - with SAT [Chockler et al. '11]
- Improve proofs [Bradley et al. '11]
  - Strengthen, weaken, shrink
  - Used in FAIR, IICTL
- Apply IC3 in design/verify cycle [Chockler et al. '11]
  - Extract inductive core from previous run
  - Accelerate analysis of mutated design or similar property
- Improve generalization [Hassan et al. '13]
  - Apply inductive generalization to counterexamples to generalization (CTGs)
  - Not just explicitly discovered backward reachable states
  - Essentially uniform improvement $\therefore$ better
Localization Reduction

- Extract information from incomplete concrete run to guide refinements [Baumgartner et al. ’12]
  - Level at which variable is first used
  - Reduction in abstract model size in practice
- Lazy abstraction [Vizel et al. ’12]
  - Visible variables abstraction $U_0 \subseteq U_1 \subseteq \cdots \subseteq U_k$
  - Refinement: run IC3 on concrete model at $k$
  - Then use unsat core of $F_i \land T \Rightarrow F'_{i+1}$ to derive new $U_i$
IC3 $\Rightarrow$ IIIV (Incremental, Inductive Verification)

- Hypothesize "lasso" "skeleton"
- Attempt to "flesh out"
- Failure explained by inductive proof...
- ... that refines space of hypotheses

G(p -> Fq)

Eep! Uhh... maybe?

IICTL $\Rightarrow$ "Local" style
- Proof-based generalization
- Lifting-based generalization

FAIR $\downarrow$

IC3
Search for lasso as usual

Top-level SAT query:
  ▶ Find set of states in one “arena” that satisfy all Büchi conditions
  ▶ If UNSAT, property holds

Reachability queries to connect states:
  ▶ Stem: From initial state to one of states
  ▶ Cycle: From state to state

Refinement from inductive strengthenings:
  ▶ Stem: Global reachability
  ▶ Cycle: Transection of state space—loop must be on one side
CTL Model Checking [Hassan et al. ’12]

- “Local” method + generalization
- Incrementally refine lower/upper bounds on subformulas
- Generalize from queries involving explicit states:
  - $EX\psi$: SAT (unsat core)
  - $EF\psi$: reachability, e.g., IC3 (inductive strengthening)
  - $EG\psi$: constrained cycle, e.g., FAIR (inductive strengthening)
- Generalize traces with aggressive lifting
Other decidable domains

- Timed systems [Hoder et al. ’12, Kindermann et al. ’12]
- Petri nets (and more general) [Kloos et al. ’13]
- Finite-state safety games [Morgenstern ’13]
IC3 with SMT

- Combination with lazy abstraction [Cimatti et al. ’12]
- Constrained Horn Clauses [Hoder et al. ’12]
- Polyhedra [Welp et al. ’13]
Use inductive generalization to locally construct interpolant
[Vizel et al. '13]
Conclusion

Main ideas:

▶ Induction as a mechanism for generalization
▶ Incremental, local (state-triggered) reasoning

Complements monolithic reasoning, which sometimes wins
Thanks! Questions?