Incremental, Inductive Model Checking

Aaron Bradley

September 28, 2013
Notation

\[ S : (\bar{x}, \bar{i}, I(\bar{x}), T(\bar{x}, \bar{i}, \bar{x}')) \]  \hspace{1cm} \text{Invariant property : } P

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

**Problem:** Show all reachable states satisfy \( P \)
SAT-Based Model Checking: 
Just Unroll
For $k = 0, 1, 2, \ldots$, SAT query:

$$I(\overline{x}_0) \land \bigwedge_{j=1}^{k} T(\overline{x}_{j-1}, \overline{i}_{j-1}, \overline{x}_j) \land \neg P(\overline{x}_k)$$

until an error is found or the diameter is reached.
Mathematical induction over $S$:

- $I(x) \Rightarrow P(x)$ (initiation)
- $P(x) \land T(x, i, x') \Rightarrow P(x')$ (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)

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

Consecution:

\[
\text{LoopFree} \land \bigwedge_{j=1}^{k} \left( P(\overline{x}_{j-1}) \land T(\overline{x}_{j-1}, \overline{i}_{j-1}, \overline{x}_j) \right) \Rightarrow P(\overline{x}_k)
\]
Longest loop-free path
\[ k = 6 \]

\( k \)-Induction
Interpolant-based Model Checking (ITP)

Post-condition operator:

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

Abstract post-condition operator:

$$\text{post}(F)(\overline{x}) \Rightarrow \widetilde{\text{post}}(F)(\overline{x})$$
Interpolant-based Model Checking (ITP)

If this query is UNSAT

\[ F(x_0) \land \bigwedge_{j=1}^k T(x_{j-1}, i_{j-1}, x_j) \Rightarrow P(x_k) \]

then extract \( G \) such that

\[ F(x_0) \land T(x_0, i_0, x_1) \Rightarrow G(x_1) \]

and

\[ G(x_1) \land \bigwedge_{j=2}^k T(x_{j-1}, i_{j-1}, x_j) \Rightarrow P(x_k) \]

Then

\[ \hat{\text{post}}(F)(\overline{x}) := G(\overline{x}) \]
SAT-Based Model Checking: 
Don’t Unroll!
Incremental

Monolithic
FSIS (Finite State Inductive Strengthening)

- FMGAD 2007
- From backward reachable state $s$ to $c < -s \ \text{s.t.}
  \begin{align*}
  F \land a \land T \Rightarrow a' \quad \text{(and I} \Rightarrow a) \\
  \text{on top of explicit backward enumeration.}
  \end{align*}

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 $\langle t, j - 1 \rangle$
Inductive Generalization
When

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

- Propagate clauses forward with relative induction
- Increment \( k \) (unless converged)
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

- "Local" style
  - Proof-based generalization
  - Lifting-based generalization

FAIR

IC3
LTL ($\omega$-regular) Model Checking [Bradley et al. ’11]

- 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:
  - $\text{EX} \psi$: SAT (unsat core)
  - $\text{EF} \psi$: reachability, e.g., IC3 (inductive strengthening)
  - $\text{EG} \psi$: 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]
Combination with ITP

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?