Latticed k-Induction with an Application to Probabilistic Programs

Kevin Batz, Mingshuai Chen, Benjamin Lucien Kaminski,
Joost-Pieter Katoen, Christoph Matheja, Philipp Schröer

MOVES Seminar

April 20, 2021

k-Induction [Sheeran et al. 2000]

- SAT-based technique for verifying invariant properties of finite transition systems
- Later: Verification of *infinite-state* transition systems via SMT solving
- Applications: Hardware- and software model checking

" [k-induction] easily integrates with existing SAT-solvers [...]. The simplicity of applying k-induction made it the go-to technique for SMT-based infinite-state model checking."[1]

Is k-induction applicable to

(possibly infinite-state) probabilistic program verification?

Yes. Enables *fully automatic* verification of non-trivial properties.

[1] [Krishnan *et al.* 2018]

Classical k-Induction for Transition Systems

Given: $TS = (S, I, T)$, invariant property $P \subseteq S$

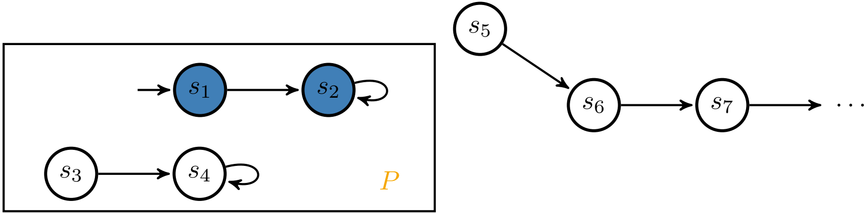
Goal: Prove that P covers all reachable states of TS

By induction. If

$$I \subseteq P$$

and $$\forall s, t \in S\colon \quad s \in P \ \wedge \ T(s,t) \quad \Longrightarrow \quad t \in P \ ,$$

then P is an inductive invariant covering all reachable states.



Classical k-Induction for Transition Systems

Given: $\mathit{TS} = (S, I, T)$, invariant property $P \subseteq S$

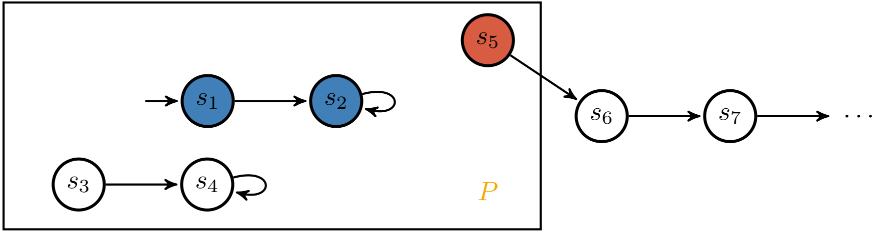
Goal: Prove that P covers all reachable states of TS

By 1-induction. If

$$I \subseteq P$$

and $$\forall s, t \in S\colon \quad s \in P \;\wedge\; T(s,t) \quad\Longrightarrow\quad t \in P\ ,$$

then P is an 1-inductive invariant covering all reachable states.



Classical k-Induction for Transition Systems

Given: $\mathit{TS} = (S, I, T)$, invariant property $P \subseteq S$

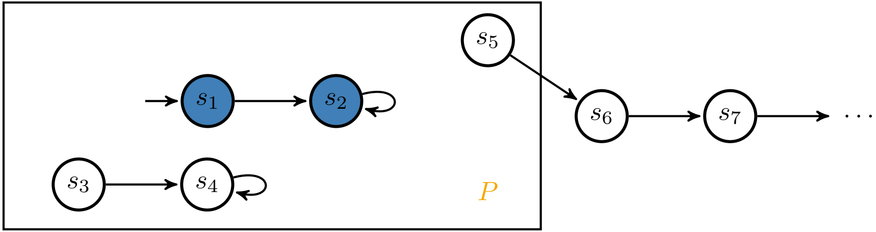
Goal: Prove that P covers all reachable states of TS

By 2-induction. If

$$I \subseteq P \quad \text{and} \quad \mathit{Succs}(I) \subseteq P$$

$$\text{and} \qquad \forall s, t, u \in S\colon \quad s \in P \;\wedge\; T(s,t) \;\wedge\; t \in P \;\wedge\; T(t,u) \quad \Longrightarrow \quad u \in P\;,$$

then P is a 2-inductive invariant covering all reachable states.



Classical k-Induction for Transition Systems

Given: $TS = (S, I, T)$, invariant property $P \subseteq S$

Goal: Prove that P covers all reachable states of TS

Let $k \geq 1$. If the following two formulae are valid

$$\underbrace{I(s_1) \wedge T(s_1, s_2) \wedge \ldots \wedge T(s_{k-1}, s_k)}_{\text{all states reachable within } k-1 \text{ steps}} \implies \underbrace{P(s_1) \wedge \ldots \wedge P(s_k)}_{\text{are } P\text{-states}}$$

$$\underbrace{P(s_1) \wedge T(s_1, s_2) \wedge \ldots \wedge P(s_k)}_{\text{assuming we stay in } P \text{ for } k-1 \text{ steps,}} \wedge \underbrace{T(s_k, s_{k+1})}_{\text{after step } k,} \implies \underbrace{P(s_{k+1})}_{\text{we end up in } P \text{ again}} ,$$

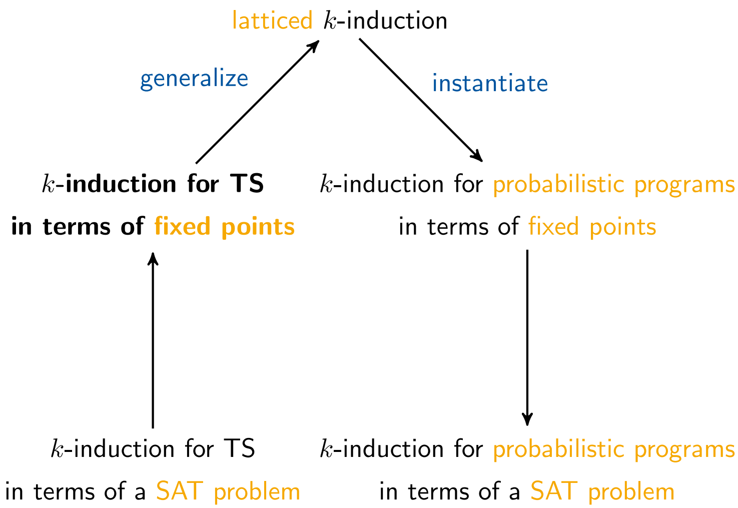
then P is a k-inductive invariant covering all reachable states of TS.

For verifying probabilistic programs, we have to ...

... leave the Boolean domain and reason about quantities

... reason about sets of paths rather than individual paths

k-Induction


latticed k-induction
generalize
instantiate
k-induction for TS in terms of fixed points
k-induction for probabilistic programs in terms of fixed points
k-induction for TS in terms of a SAT problem
k-induction for probabilistic programs in terms of a SAT problem

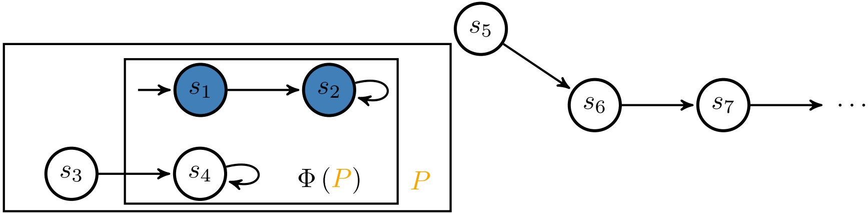
Let $TS = (S, I, T)$ and $P \subseteq S$. Define $\Phi \colon 2^S \to 2^S$ on the complete lattice $(2^S, \subseteq)$ by

$$\Phi\left(F\right) = I \cup \mathit{Succs}(F)\ . \qquad \text{Then:} \quad \mathit{Reach}\left(\mathit{TS}\right) = \mathsf{lfp}\ \Phi$$

Goal: Prove $\mathsf{lfp}\ \Phi \subseteq P$

By 1-induction. If

$$\Phi\left(P\right) \subseteq P \qquad \text{then} \qquad \mathsf{lfp}\ \Phi \subseteq P\ .$$



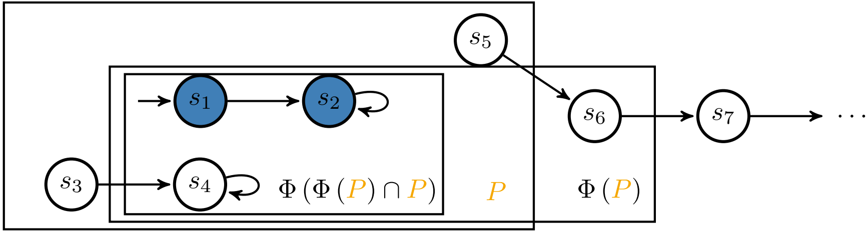
Let *TS* $= (S, I, T)$ and $P \subseteq S$. Define $\Phi \colon 2^S \to 2^S$ on the complete lattice $(2^S, \subseteq)$ by

$$\Phi(F) = I \cup \mathit{Succs}(F)~. \qquad \text{Then:} \quad \mathit{Reach}(\mathit{TS}) = \mathrm{lfp}~\Phi$$

Goal: Prove $\mathrm{lfp}~\Phi \subseteq P$

By 2-induction. If

$$\Phi(\Phi(P) \cap P) \subseteq P \qquad \text{then} \qquad \mathrm{lfp}~\Phi \subseteq P~.$$



Let *TS* $= (S, I, T)$ and $P \subseteq S$. Define $\Phi\colon 2^S \to 2^S$ on the complete lattice $(2^S, \subseteq)$ by

$$\Phi(F) = I \cup \mathit{Succs}(F)\ . \qquad \text{Then:} \quad \mathit{Reach}(\mathit{TS}) = \mathsf{lfp}\ \Phi$$

Goal: Prove $\mathsf{lfp}\ \Phi \subseteq P$

By 2-induction. If

$$\Phi(\Phi(P) \cap P) \subseteq P \qquad \text{then} \qquad \mathsf{lfp}\ \Phi \subseteq P\ .$$

By 3-induction. If

$$\Phi(\Phi(\Phi(P) \cap P) \cap P) \subseteq P \qquad \text{then} \qquad \mathsf{lfp}\ \Phi \subseteq P\ .$$

⋮

Let $\mathit{TS} = (S, I, T)$ and $P \subseteq S$. Define $\Phi \colon 2^S \to 2^S$ on the complete lattice $(2^S, \subseteq)$ by

$$\Phi(F) = I \cup \mathit{Succs}(F) \ . \qquad \text{Then:} \quad \mathit{Reach}(\mathit{TS}) = \mathrm{lfp}\ \Phi$$

Goal: Prove $\mathrm{lfp}\ \Phi \subseteq P$

Define $\Psi_P \colon 2^S \to 2^S$ by

$$\Psi_P(F) = \Phi(F) \cap P \ .$$

For every $k \geq 1$,

$$\Phi\left(\Psi_P^{k-1}(P)\right) \subseteq P \qquad \text{implies} \qquad \mathrm{lfp}\ \Phi \subseteq P \ .$$

latticed k-induction

generalize

instantiate

k-induction for TS
in terms of fixed points

k-induction for probabilistic programs
in terms of fixed points

k-induction for TS
in terms of a SAT problem

k-induction for probabilistic programs
in terms of a SAT problem

Let $(E, \sqsubseteq)$ be a complete lattice. Furthermore, let $\Phi\colon E \to E$ be monotonic and $f \in E$.

Goal: Prove $\mathsf{lfp}\ \Phi \sqsubseteq f$.

Define $\Psi_f\colon E \to E$ by

$$\Psi_f\,(g) = \Phi\,(g) \sqcap f\ .$$

Theorem (Latticed k-Induction)

For every $k \geq 1$,

$$\Phi\left(\Psi_f^{k-1}(f)\right) \sqsubseteq f \quad \textit{implies} \quad \mathsf{lfp}\ \Phi \sqsubseteq f\ .$$

We call such f k-inductive invariant.

k-Induction generalizes Park induction $\triangleq$ 1-induction.

Can be generalized to transfinite κ-induction (not in this talk).

Latticed k-Induction

Theorem (Park Induction from k-Induction)

$$\underbrace{\Phi\left(\Psi_f^{k-1}(f)\right) \sqsubseteq f}_{f \text{ is } k\text{-inductive invariant}} \quad \textit{iff} \quad \underbrace{\Phi\left(\Psi_f^{k-1}(f)\right) \sqsubseteq \Psi_f^{k-1}(f)}_{\Psi_f^{k-1}(f) \text{ is inductive invariant}}$$

Lemma

Iterating Ψ_f on f yields a descending chain, i.e.,

$$f \sqsupseteq \Psi_f(f) \sqsupseteq \Psi_f^2(f) \sqsupseteq \Psi_f^3(f) \sqsupseteq \ldots .$$

Hence, if f is k-inductive invariant, then

- $\Psi_f^{k-1}(f)$ is an inductive invariant,
- which is stronger than f.

Latticed k-induction generalizes classical k-induction for TS:

Theorem

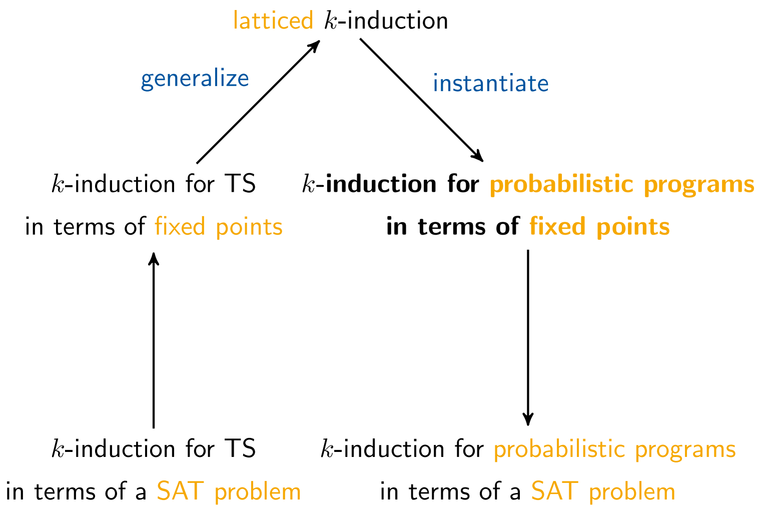
Let TS $= (S, I, T)$ *and* $P \subseteq S$. *For every* $k \geq 1$, *the formulae*

$$I(s_1) \wedge T(s_1, s_2) \wedge \ldots \wedge T(s_{k-1}, s_k) \quad \Longrightarrow \quad P(s_1) \wedge \ldots \wedge P(s_k)$$

$$P(s_1) \wedge T(s_1, s_2) \wedge \ldots \wedge P(s_k) \wedge T(s_k, s_{k+1}) \quad \Longrightarrow \quad P(s_{k+1})$$

are valid if and only if

$$\Phi\left(\Psi_P^{k-1}(P)\right) \subseteq P~.$$


latticed k-induction
generalize
instantiate
k-induction for TS
in terms of fixed points
k-induction for probabilistic programs
in terms of fixed points
k-induction for TS
in terms of a SAT problem
k-induction for probabilistic programs
in terms of a SAT problem

Consider the complete lattice $(\mathbb{E}, \leq)$ of *expectations*:

$$\mathbb{E} = \{ f \mid f \colon \Sigma \to \mathbb{R}_{\geq 0}^{\infty} \} \qquad \text{with} \qquad f \leq g \quad \text{iff} \quad \forall \sigma \in \Sigma\colon f(\sigma) \leq g(\sigma)$$

Weakest preexpectation transformer [Kozen, McIver & Morgan]:

$$\mathsf{wp}[\![C]\!] \colon \mathbb{E} \to \mathbb{E} \qquad \mathsf{wp}[\![C]\!]\,(g)\,(\sigma) \triangleq \text{expected value of } g \text{ evaluated in final states reached after executing } C \text{ on } \sigma$$

$$\mathsf{wp}[\![x \coloneqq 5]\!]\,(x) \;=\; 5$$

$$\mathsf{wp}[\![\{\, \texttt{skip} \,\}\, [1/2]\, \{\, x \coloneqq x + 2 \,\}]\!]\,(x) \;=\; \frac{1}{2} \cdot x + \frac{1}{2} \cdot (x+2) \;=\; x + 1$$

$$\mathsf{wp}[\![\{\, \texttt{skip} \,\}\, [1/2]\, \{\, x \coloneqq x + 2 \,\}]\!]\,([x = 4]) \;=\; \frac{1}{2} \cdot [x = 4] + \frac{1}{2} \cdot [x = 2]$$

$$\mathsf{wp}[\![\texttt{while}\,(\, c = 1 \,)\,\{\, \{\, c \coloneqq 0 \,\}\, [1/2]\, \{\, x \coloneqq x + 1 \,\}\,\}]\!]\,(x) \;=\; [c = 1] \cdot (x + 1) + [c \neq 1] \cdot x$$

Given: Loop $C = \texttt{while}\,(\,\varphi\,)\,\{\,C'\,\}$ and $f, g \in \mathbb{E}$

Goal: Prove $\mathsf{wp}[\![C]\!]\,(g) \leq f$

We have

$$\mathsf{wp}[\![C]\!]\,(g) = \mathsf{lfp}\ \Phi \qquad \text{with } \Phi\colon \mathbb{E} \to \mathbb{E} \text{ monotonic} \ .$$

Hence, latticed k-induction applies:

Corollary

For every $k \geq 1$,

$$\Phi\left(\Psi_f^{k-1}(f)\right) \leq f \quad \textit{implies} \quad \mathsf{wp}[\![C]\!]\,(g) \leq f \ .$$

Here

$$\Psi_f(h) = \Phi\,(h) \sqcap f \qquad \text{where for } h, h' \in \mathbb{E}, \quad h \sqcap h' = \lambda\sigma\text{.}\ \min\{h(\sigma), h'(\sigma)\} \ .$$

Given *linear* $C = \texttt{while}\,(\,\varphi\,)\,\{\,C'\,\}$ and *linear* $f, g \in \mathbb{E}$, our tool

kipro2 : k-**Induction for PRObabilistic PROgrams**

not: Kevin is programming 2

semi-decides via SMT solving:

Is there $k \geq 1$ such that $\mathsf{wp}[\![C]\!]\,(g) \leq f$ is k-inductive?

Furthermore, if $\mathsf{wp}[\![C]\!]\,(g) \not\leq f$, $\textsc{kipro2}$ finds via *bounded model checking* some $\sigma \in \Sigma$ with

$$\mathsf{wp}[\![C]\!]\,(g)\,(\sigma) > f(\sigma)\ .$$

For C_{geo} given by

$$\texttt{while}\,(\,c = 1\,)\,\{\,\{\,c \coloneqq 0\,\}\,[\,1/2\,]\,\{\,x \coloneqq x + 1\,\}\,\}\ ,$$

the property

$$\mathsf{wp}[\![C_{\text{geo}}]\!]\,(x) \leq x + 1$$

is 2-inductive. Does

$$\mathsf{wp}[\![C_{\text{geo}}]\!]\,(x) \leq x + 0.99$$

also hold? No, bounded model checking yields a counterexample: $c = 1, x = 6$.

For C_{brp} given by

$$\begin{aligned}
&\texttt{while}\,(\, sent < toSend \wedge fail < maxFail \,)\,\{\\
&\quad \{\, fail \coloneqq 0\,;\, sent \coloneqq sent + 1 \,\}\,[\,0.9\,]\,\{\, fail \coloneqq fail + 1\,;\, totalFail \coloneqq totalFail + 1 \,\}\\
&\}
\end{aligned}$$

the property

$$\mathsf{wp}[\![C_{\mathsf{brp}}]\!]\,(totalFail) \leq [toSend \leq 3] \cdot (totalFail + 1) + [toSend > 3] \cdot \infty$$

is 4-inductive. Does

$$\mathsf{wp}[\![C_{\mathsf{brp}}]\!]\,(totalFail) \leq totalFail + 1$$

also hold? No: $toSend = 6052, maxFail = 2, sent = 6042, fail = 0, totalFail = 1$

Sampling uniformly from $\{elow, \ldots, ehigh\}$ using fair coin flips only [Lumbroso 2013]:

```
while(running = 0){

  v := 2*v;
  {c := 2*c+1}[0.5]{c := 2*c};
  if((not (v<n))){
    if((not (n=c)) & (not (n<c))){ # terminate
      running := 1
    }{
      v := v-n;
      c := c-n;
    }
  }{
    skip
  }

  # On termination, determine correct index
  if((not (running = 0))){
    c := elow + c;
  }{
    skip
  }
}
```

Conclusion

- k-Induction for transition systems in terms of fixed points
- latticed k-induction
- fully automatic k-induction for probabilistic programs

Further topics:

- incremental SMT encoding (theory: QF_UFLIRA)
- k-induction for expected run-times
- transfinite κ-induction
- (in)completeness of k-induction
- latticed bounded model checking (refute $\text{lfp}\ \Phi \sqsubseteq f$)

Thank you!

Backup: Runtimes

Table 2: Empirical results for the first benchmark set (time in seconds).

	postexpectation	variant	result	k	#formulae	formulae_t	sat_t	total_t
brp	*totalFail*	1	ind	5	285	0.15	0.01	0.28
		2	ind	11	2812	1.77	0.12	2.03
		3	ind	23	26284	17.68	28.09	45.94
		4	TO	–	–	–	–	–
		5	ref	13	949	0.84	14.39	15.28
		6	TO	–	–	–	–	–
		7	TO	–	–	–	–	–
geo	c	1	ind	2	18	0.01	0.00	0.08
		2	ref	11	103	0.04	0.01	0.09
		3	ref	46	1223	0.39	0.04	0.48
rabin	$[i = 1]$	1	ind	1	21	0.01	0.00	0.15
		2	ind	5	1796	1.27	0.03	1.44
		3	TO	–	–	–	–	–
		4	ref	4	458	0.31	0.03	0.40
		5	ref	8	10508	8.76	2.85	11.68
unif_gen	$[c = i]$	1	ind	2	267	0.27	0.02	0.56
		2	ind	3	1402	1.45	0.10	1.81
		3	ind	3	1402	1.48	0.11	1.86
		4	ind	5	40568	47.31	15.70	63.28
		5	TO	–	–	–	–	–