Aiming Low is Harder

Induction for Lower Bounds in Probabilistic Program Verification

Marcel Hark    Benjamin Kaminski    Jürgen Giesl    Joost-Pieter Katoen
Motivation

Setting the stage
Motivation

Setting the stage

- Expected outcome of probabilistic programs $P$ w.r.t. expectation $f : \Sigma \rightarrow \mathbb{R}_{\geq 0}$.

Expectation $\leftrightarrow$ Random Variable.

→ Weakest preexpectation of $f$:

$$\text{wp/llbracket} P \text{/rrbracket}(f) : \Sigma \rightarrow \mathbb{R}_{\geq 0}.$$ (total correctness).

→ Quantitative version of Dijkstra’s wp-calculus: [Kozen 81, McIver, Morgan 96].

Bottleneck: $\text{wp/llbracket} \text{while} (\phi) \{ C \} /\text{rrbracket}(f)$ (loop invariants).

...
Motivation

Setting the stage

• Expected outcome of probabilistic programs $P$ w.r.t. expectation $f : \Sigma \rightarrow \mathbb{R}_{\geq 0}$.
  Expectation $\leftrightarrow$ Random Variable.

$\exists \tau_1, \tau_2, \tau_3 \ldots$ $f(\tau_1) \Rightarrow wp[\llbracket P \rrbracket](f)(s) = \text{Exp}[\ldots]$

$\text{Probability that } x \text{ is 10 after termination.}$

$f = x^2 + y^2 : wp[\llbracket P \rrbracket](f) \Rightarrow \text{Expected outcome of } x^2 + y^2 \text{ after termination.}$
Motivation

Setting the stage

- Expected outcome of probabilistic programs $P$ w.r.t. expectation $f : \Sigma \rightarrow \mathbb{R}_{\geq 0}$.

\[ \mathbb{E}[f(T_1)] \quad \mathbb{E}[f(T_2)] \quad \mathbb{E}[f(T_3)] \]

\[ \mathbb{E}[P] \]

\[ S \]

\[ \mathbb{E}[f(T)] \]

\[ f : \Sigma \rightarrow \mathbb{R}_{\geq 0} \]

- Weakest preexpectation of $f$:
  \[ \text{wp} / \llbracket P \rrbracket (f) : \Sigma \rightarrow \mathbb{R}_{\geq 0} \]

- Quantitative version of Dijkstra's wp-calculus: [Kozen 81, McIver, Morgan 96].

- Bottleneck: $\text{wp} / \llbracket \text{while} (\phi) \{ C \} \rrbracket (f)$ (loop invariants).
Motivation

Setting the stage

- Expected outcome of probabilistic programs $P$ w.r.t. expectation $f : \Sigma \rightarrow \mathbb{R}^\infty_{\geq 0}$.
  - Weakest preexpectation of $f$: $\text{wp}[P](f) : \Sigma \rightarrow \mathbb{R}^\infty_{\geq 0}$

$$\text{wp}[P] (f) (s) = \text{Exp}\left[ f(\tau_1) f(\tau_2) f(\tau_3) \right]$$
Motivation

Setting the stage

- Expected outcome of probabilistic programs $P$ w.r.t. expectation $f : \Sigma \rightarrow \mathbb{R}_\geq 0$.
  - $\rightarrow$ Weakest preexpectation of $f$: $\text{wp}[P](f) : \Sigma \rightarrow \mathbb{R}_\geq 0$ (total correctness).

\[
\text{wp}[P](f)(s) = \text{Exp}\left[ f(\tau_1) f(\tau_2) f(\tau_3) \right]
\]
Motivation

Setting the stage

- Expected outcome of probabilistic programs $P$ w.r.t. expectation $f : \Sigma \rightarrow \mathbb{R}_{\geq 0}$. 
  - Weakest preexpectation of $f$: $wp\llbracket P \rrbracket (f) : \Sigma \rightarrow \mathbb{R}_{\geq 0}$ (total correctness).

\[
wp\llbracket P \rrbracket (f) (s) = \text{Exp} \left( f(\tau_1) \right) = f(\tau_2) = f(\tau_3)
\]

$f = [x = 10]$: $wp\llbracket P \rrbracket (f) \leadsto$ Probability that $x$ is 10 after termination.
Motivation

Setting the stage

- Expected outcome of probabilistic programs $P$ w.r.t. expectation $f : \Sigma \rightarrow \mathbb{R}_{\geq 0}$.
  - Weakest preexpectation of $f$: $\text{wp}\llbracket P \rrbracket (f) : \Sigma \rightarrow \mathbb{R}_{\geq 0}$ (total correctness).

\[ \text{wp}\llbracket P \rrbracket (f)(s) = \text{Exp}\left[ f(\tau_1) f(\tau_2) f(\tau_3) \right] \]

$f = x^2 + y^2$: $\text{wp}\llbracket P \rrbracket (f) \rightsquigarrow$ Expected outcome of $x^2 + y^2$ after termination.
Motivation

Setting the stage

- Expected outcome of probabilistic programs $P$ w.r.t. expectation $f : \Sigma \rightarrow \mathbb{R}_{\geq 0}$.
  - Weakest preexpectation of $f$: $wp[\llbracket P \rrbracket (f) : \Sigma \rightarrow \mathbb{R}_{\geq 0}$ (total correctness).
  - Quantitative version of Dijkstra’s wp-calculus: [Kozen 81, McIver, Morgan 96].

$$wp[\llbracket P \rrbracket (f)(s) = \text{Exp}\left[ f(\tau_1) \quad f(\tau_2) \quad f(\tau_3) \right]$$

$$f = x^2 + y^2: \text{wp}[\llbracket P \rrbracket (f) \sim \text{Expected outcome of } x^2 + y^2 \text{ after termination.}$$
Motivation

Setting the stage

- Expected outcome of probabilistic programs $P$ w.r.t. expectation $f : \Sigma \to \mathbb{R}_\geq 0$.
  - Weakest preexpectation of $f$: $\text{wp}[P](f) : \Sigma \to \mathbb{R}_\geq 0$ (total correctness).
  - Quantitative version of Dijkstra's wp-calculus: [Kozen 81, McIver, Morgan 96].

$$\text{wp}[P](f)(s) = \text{Exp}\left[ f(\tau_1) \quad f(\tau_2) \quad f(\tau_3) \right]$$
Motivation

Setting the stage

• Expected outcome of probabilistic programs $P$ w.r.t. expectation $f : \Sigma \rightarrow \mathbb{R}_\geq 0$.
  
  $\rightarrow$ Weakest preexpectation of $f$: $\text{wp}[P](f) : \Sigma \rightarrow \mathbb{R}_\geq 0$ (total correctness).
  
  $\rightarrow$ Quantitative version of Dijkstra's wp-calculus: [Kozen 81, Mclver, Morgan 96].

$\text{wp}[P](f)(s) = \text{Exp}\left[f(\tau_1) \quad f(\tau_2) \quad f(\tau_3)\right]$

$\rightarrow$ Bottleneck: $\text{wp}[\text{while}(\varphi)\{C\}](f)$
Motivation

Setting the stage

- Expected outcome of probabilistic programs $P$ w.r.t. expectation $f : \Sigma \rightarrow \mathbb{R}_\geq 0$.
  - Weakest preexpectation of $f$: $\text{wp}[P](f) : \Sigma \rightarrow \mathbb{R}_\geq 0$ (total correctness).
  - Quantitative version of Dijkstra's wp-calculus: [Kozen 81, Mclver, Morgan 96].

$$\text{wp}[P](f)(s) = \text{Exp}\left[ f(\tau_1) f(\tau_2) f(\tau_3) \right]$$

→ Bottleneck: $\text{wp}[\text{while}(\varphi)\{C\}](f)$ (loop invariants).
Motivation

Setting the stage
Motivation

Setting the stage

while \(a = 1\) {
    \{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \}
}

\[ \langle \text{lfp} \Phi \rangle \]
\[ \langle \text{lfp} \lambda E. \left[ a \neq 1 \right] \cdot f + \left[ a = 1 \right] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x + 1]) \rangle \]

\[ \text{→Initial value of } x \text{ is unknown.} \]
\[ \text{• } \text{lfp } \Phi x = \lim_{n \to \omega} \Phi^n x(0) \text{ (incomputable).} \]

\[ \text{→We need bounds on least fixed points.} \]
Motivation

Setting the stage

```
while (a = 1) {
    { a := 0 } [\frac{1}{2}] { x := x + 1 }
}
\langle f \rangle
```

Initial value of \( x \) is unknown.

\( \text{lfp} \Phi x = \lim_{n \to \omega} \Phi^n x (0) \) (incomputable).

We need bounds on least fixed points.
Motivation

Setting the stage

\[ \langle \text{lfp } \Phi_f \rangle \]

\[
\text{while } (a = 1) \{
\quad \{ a := 0 \} \quad \left[ \frac{1}{2} \right] \quad \{ x := x + 1 \}
\}
\]

\[ \langle f \rangle \]

→ Initial value of \( x \) is unknown.

• \( \text{lfp } \Phi_x = \lim_{n \to \omega} \Phi^n_x(0) \) (incomputable).

→ We need bounds on least fixed points.
Motivation

Setting the stage

\[
\langle \text{lfp } \lambda E. \ [a \neq 1] \cdot f + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x + 1]) \rangle
\]
while \((a = 1)\) {
\[
\{ a := 0 \} \ [\frac{1}{2}] \ \{ x := x + 1 \}
\]
\}
\langle f \rangle

→ Initial value of \(x\) is unknown.

• \(\text{lfp } \Phi x = \lim_{n \to \omega} \Phi^n x (0)\) (incomputable).

→ We need bounds on least fixed points.
Motivation

Setting the stage

\[
\langle \text{lfp } \lambda E. [a \neq 1] \cdot f + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x + 1]) \rangle
\]

while \( a = 1 \) {
\[
\{ a := 0 \} \begin{bmatrix} 1 \\ 2 \end{bmatrix} \{ x := x + 1 \}
\]
}\langle x \rangle

→ Initial value of \( x \) is unknown.

\[ \text{lfp } \Phi x = \lim_{n \to \omega} \Phi^n x (0) \text{ (incomputable).} \]

→ We need bounds on least fixed points.
Motivation

Setting the stage

\[ \langle \text{lfp } \lambda E. \ [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x + 1]) \rangle \]

while \((a = 1)\) {
\{ \ a := 0 \} \ [1/2] \ \{ \ x := x + 1 \} 
}

\langle x \rangle

Initial value of \(x\) is unknown.

- \(\text{lfp } \Phi x = \lim_{n \to \omega} \Phi^n x (0)\) (incomputable).

- We need bounds on least fixed points.
Motivation

Setting the stage

\[
\langle \text{lfp } \lambda E. [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x+1]) \rangle
\]

while \( (a = 1) \) {
\[
\{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \}
\]
}

\[ \langle x \rangle \]

\[ \rightarrow \text{Initial value of } x \text{ is unknown.} \]
Motivation

Setting the stage

\[
\langle \text{lfp } \lambda E. \ [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x + 1]) \rangle
\]

\[
\text{while } (a = 1) \{
\begin{cases}
    a := 0 & \left[\frac{1}{2}\right] \\
    x := x + 1
\end{cases}
\}
\]

\[
\langle x \rangle
\]
Motivation

Setting the stage

\[
\langle \text{lfp } \lambda E. [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x + 1]) \rangle
\]

\[
\text{while (a = 1)} \{
\{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \}
\}
\]

\[
\langle x \rangle
\]

• \text{lfp } \Phi_x = \lim_{n \to \omega} \Phi_x^n(0) \text{ (incomputable).}
Motivation

Setting the stage

\[ \langle \text{lfp } \lambda E. \ [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x + 1]) \rangle \]

while \( (a = 1) \{
{a := 0} [\frac{1}{2}] \ {x := x + 1}
\}
\]

\[ \langle x \rangle \]

- \( \text{lfp } \Phi_x = \lim_{n \to \omega} \Phi_x^n(0) \) (incomputable).

\[ \rightarrow \text{We need bounds on least fixed points.} \]
Motivation

Outline

Motivation

Upper Bounds [Park]

Lower Bounds

Conclusion
Upper Bounds [Park]

Outline

Motivation

Upper Bounds [Park]

Lower Bounds

Conclusion
Park Induction

```plaintext
while (a = 1) {
    {a := 0} [1/2] {x := x + 1}
}
```

\[
\Phi x (E) = [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x+1]) \geq \Phi x (E) \Rightarrow E \geq \text{lfp } \Phi x (\text{Superinvariant}).
\]

\[
E := x + [a = 1] \text{ (Intuitive guess)}
\]

\[
\Phi x (E) = E \rightarrow E \text{ is an upper bound on } \text{lfp } \Phi x.
\]
Park Induction

\[
\text{while} (a = 1) \{
\{ a := 0 \} \quad \{ x := x + 1 \}
\}
\]

\[
\Phi_x(E) = [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x + 1])
\]
Park Induction

\[
\text{while}(a = 1) \{
\begin{align*}
&\{ a := 0 \} [\frac{1}{2}] \{ x := x + 1 \}
\end{align*}
\}
\]

\[
\Phi_x(E) = [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E [a/0] + E [x/x + 1])
\]

\[ E \geq \Phi_x(E) \implies E \geq \text{lfp } \Phi_x \text{ (Superinvariant).} \]
Park Induction

\[
\begin{align*}
\text{while} (a = 1) & \{ \!
\begin{align*}
a & := 0 \{ \frac{1}{2} \} \{ x := x + 1 \}
\end{align*}
\}\ \\
\Phi_x(E) & = [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x + 1])
\end{align*}
\]

\[E \geq \Phi_x(E) \implies E \geq \text{lfp } \Phi_x \quad \text{(Superinvariant).}\]

\[E := x + [a = 1] \quad \text{(Intuitive guess)}\]
Upper Bounds [Park]

Park Induction

\[
\text{while}(a = 1)\{
\begin{array}{l}
\{ a := 0 \} [\frac{1}{2}] \{ x := x + 1 \}
\end{array}
\}
\]

\[
\Phi_x(E) = [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x + 1])
\]

\[ E \geq \Phi_x(E) \implies E \geq \text{lfp } \Phi_x \text{ (Superinvariant)}. \]

\[ E := x + [a = 1] \text{ (Intuitive guess)} \]

\[ \Phi_x(E) = E \]
Upper Bounds [Park]

Park Induction

while (a = 1) {
    { a := 0 } \frac{1}{2} { x := x + 1 }
}

\[ \Phi_x(E) = [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E [a/0] + E [x/x + 1]) \]

\[ E \geq \Phi_x(E) \implies E \geq \text{lfp } \Phi_x \text{ (Superinvariant).} \]

\[ E := x + [a = 1] \text{ (Intuitive guess)} \]

\[ \Phi_x(E) = E \]

\[ \to E \text{ is an upper bound on } \text{lfp } \Phi_x. \]
Upper Bounds [Park]

Upper Bounds

• Rule for upper bounds $\Phi f(E) \geq E$ is simple. (Inductive)
• Not a surprise, bound a least fixed point from above.
→ Enough to bound any fixed point from above.

$E \Phi x(E) \Phi^2 x(E) \Phi^\omega x(E)$
Upper Bounds

- Rule for upper bounds $\Phi_f(E) \geq E$ is simple.
Upper Bounds

- Rule for upper bounds $\Phi_f(E) \geq E$ is simple. (Inductive)

- Not a surprise, bound a least fixed point from above.

$\rightarrow$ Enough to bound any fixed point from above.
Upper Bounds

- Rule for upper bounds $\Phi_f(E) \geq E$ is simple. (Inductive)
- Not a surprise, bound a least fixed point from above.
  \[ E \rightarrow \Phi_x(E) \]
  \[ \Phi_2(E) \]
  \[ \Phi_\omega(E) \]
  \[ \text{lfp } \Phi_x \]
Upper Bounds

- Rule for upper bounds $\Phi_f(E) \geq E$ is simple. (Inductive)
- Not a surprise, bound a least fixed point from above.
  $\Rightarrow$ Enough to bound any fixed point from above.

$E \Phi^2_x(E) \Phi^\omega_x(E) \text{lfp } \Phi_x$
Upper Bounds

Rule for upper bounds $\Phi_f(E) \geq E$ is simple. (Inductive)

Not a surprise, bound a least fixed point from above.

$\rightarrow$ Enough to bound any fixed point from above.

$E$

$\Phi^\omega_x(E)$

$lfp \ \Phi_x$
Upper Bounds

- Rule for upper bounds $\Phi_f(E) \geq E$ is simple. (Inductive)
- Not a surprise, bound a least fixed point from above.
Upper Bounds

- Rule for upper bounds $\Phi_f(E) \geq E$ is simple. (Inductive)
- Not a surprise, bound a least fixed point from above.
  $\rightarrow$ Enough to bound any fixed point from above.
Upper Bounds [Park]

Upper Bounds
Upper Bounds [Park]

Upper Bounds

No information on quality of the bound.
Upper Bounds [Park]

Upper Bounds

No information on quality of the bound.

→ We also need lower bounds.
Lower Bounds

Outline

Motivation

Upper Bounds [Park]

Lower Bounds

Conclusion
Lower Bounds

Subinvariants

\[ E \geq \Phi_x(E) = \Rightarrow E \geq \text{lfp } \Phi_x \text{(Superinvariant)} \]

\[ E \leq \Phi_x(E) = \Rightarrow E \leq \text{lfp } \Phi_x \text{(Subinvariant)} \]

Not absurd: Sound for deterministic programs. [Frohn et al. 16]
Lower Bounds

Subinvariants

\[ E \geq \Phi_x(E) \implies E \geq \text{lfp } \Phi_x \text{ (Superinvariant).} \]

Not absurd: Sound for deterministic programs. [Frohn et al. 16]
Lower Bounds

Subinvariants

\[ E \leq \Phi_x(E) \implies E \leq \text{lfp } \Phi_x \text{ (Subinvariant)} \]
Lower Bounds

Subinvariants

\[ E \leq \Phi_x(E) \implies E \leq \text{lfp } \Phi_x \text{ (Subinvariant)} \]

Not absurd: Sound for deterministic programs. [Frohn et al. 16]
Lower Bounds

\[ E \leq \Phi_x(E) \implies E \leq \text{lfp } \Phi_x \]

Subinvariants

```plaintext
while (a = 1) {
    { a := 0 } [1/2] { x := x + 1 }
}

\[ \Phi_x(E) = [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x + 1]). \]
```
Lower Bounds

\[ E \leq \Phi_x(E) \implies E \leq \text{lfp } \Phi_x \]

Subinvariants

\[
\text{while}(a = 1)\
\{\
\{ a := 0 \} \left\lceil \frac{1}{2} \right\rceil \{ x := x + 1 \}\
\}
\]

\[ \Phi_x(E) = [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x + 1]). \]

\[ E' := x + [a = 1] \cdot (1 + 2^x) \]
Subinvariants

\[
\text{while}(a = 1)\{ \\
\quad \{ a := 0 \} \ [\frac{1}{2}] \ \{ x := x + 1 \} \\
\}
\]

\[
\Phi_x(E) = [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x + 1]).
\]

\[
E' := x + [a = 1] \cdot (1 + 2^x)
\]

\[ E' \leq \Phi_x(E'), \text{ so } E' \text{ is a lower bound} \]
Lower Bounds

\[ E \leq \Phi_x(E) \implies E \leq \text{lfp } \Phi_x \]

Subinvariants

```plaintext
while (a = 1) {
  \{ a := 0 \} \frac{1}{2} \{ x := x + 1 \}
}
```

\[ \Phi_x(E) = [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x + 1]). \]

\[ E' := x + [a = 1] \cdot (1 + 2^x) \]

\[ \rightarrow E' \leq \Phi_x(E'), \text{ so } E' \text{ is a lower bound} \]

• Already seen upper bound by superinvariant \( E = x + [a = 1]. \)
Lower Bounds

\[ E \leq \Phi_x(E) \implies E \leq \text{lfp} \Phi_x \]

Subinvariants

```plaintext
while (a = 1) {
    { a := 0 } [\frac{1}{2}] { x := x + 1 }
}
```

\[ \Phi_x(E) = [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E[a/0] + E[x/x + 1]). \]

\[ E' := x + [a = 1] \cdot (1 + 2^x) \]

\[ \rightarrow E' \leq \Phi_x(E'), \text{ so } E' \text{ is a lower bound} \]

- Already seen upper bound by superinvariant \( E = x + [a = 1] \).

\[ E' \not\leq E. \]
Lower Bounds

\[ E \leq \Phi_x(E) \implies E \leq \text{lfp } \Phi_x \]

Subinvariants

while \( (a = 1) \) {
\{
\{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \}
\}
\]

\[ \Phi_x(E) = [a \neq 1] \cdot x + [a = 1] \cdot \frac{1}{2} \cdot (E [a/0] + E [x/x + 1]) \]

\[ E' := x + [a = 1] \cdot (1 + 2^x) \]

\[ \implies E' \leq \Phi_x(E'), \text{ so } E' \text{ is a lower bound} \]

- Already seen upper bound by superinvariant \( E = x + [a = 1] \).

\[ E' \nprecedes E. \]
Lower Bounds

\[ E \leq \Phi_x(E) \implies E \leq \text{lfp } \Phi_x \]

Subinvariants

\[ \Phi_\omega(x) (E) \]

\[ E \leq \text{lfp } \Phi_x \]
Lower Bounds

\[ E \leq \Phi_x(E) \implies E \leq \text{lfp } \Phi_x \]

Subinvariants

→ Subinvariants are i.e. not sound for lower bounds.
Lower Bounds

\[ E \leq \Phi_x(E) \implies E \leq \text{lfp } \Phi_x \]

Subinvariants

→ Subinvariants are i.g. not sound for lower bounds.

→ Additional requirements to extract lower bound.
Lower Bounds

\[ E \leq \Phi_x(E) \implies E \leq \text{lfp } \Phi_x \]

Subinvariants

→ Subinvariants are i.g. not sound for lower bounds.

→ Additional requirements to extract lower bound.
Lower Bounds

\[ \text{wp}[\text{while}(\varphi)\{C\}](f) \]

Our Rule
\[ E \leq \Phi_f(E) \land \text{additional requirements} \]
\[ \text{easily checkable} \implies E \leq \text{lfp}\Phi_f. \]

1. Expected finite number of loop iterations.
2. Expected change is bounded by constant.

\[ \text{Exp}(T \neg \varphi) < \infty \]

\[ \lambda_s. [\varphi] \text{wp} / [C] (|E - E(s)|) \leq K \text{ for } K \in \mathbb{R} \geq 0 \]
\[ \implies E \leq \text{lfp}\Phi_f. \]
Lower Bounds

Our Rule

\[ E \leq \Phi_f(E) \land \text{additional requirements} \leq \text{easily checkable} \implies E \leq \text{lfp } \Phi_f. \]
**Lower Bounds**

<table>
<thead>
<tr>
<th>Our Rule</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E \leq \Phi_f(E) \land$</td>
</tr>
<tr>
<td>1. $\Exp (T^\lnot \varphi) &lt; \infty$</td>
</tr>
<tr>
<td>2. $\lambda s. \wp [C] (</td>
</tr>
</tbody>
</table>

\[ wp[\text{while}(\varphi)\{C\}] (f) \]
Our Rule

\[
E \leq \Phi_f(E) \land \\
1. \text{Exp} (T^-\varphi) < \infty \\
2. \lambda s. [\varphi]_{\text{wp}} \llbracket C \rrbracket (|E - E(s)|) \leq K \text{ for } K \in \mathbb{R}_{\geq 0} \implies E \leq \text{lfp } \Phi_f.
\]

1. Expected finite number of loop iterations.
Our Rule

\[ E \leq \Phi_f(E) \land \]
\[ 1. \ \text{Exp}(T^{\neg \varphi}) < \infty \]
\[ 2. \ \lambda s. [\varphi] \text{wp}[C](|E - E(s)|) \leq K \text{ for } \Rightarrow E \leq \text{lfp } \Phi_f. \]

1. Expected finite number of loop iterations.
Lower Bounds

Our Rule

\( E \leq \Phi_f(E) \land \)

1. Expected finite number of loop iterations.

2. Expected change is bounded by constant.

\[
\left\lceil \begin{array}{c}
\text{Exp}\left( T^\neg \varphi \right) < \infty \\
\lambda s. \left[ \varphi \right] \wp \left[ C \right] \left( |E - E(s)| \right) \leq K \text{ for } K \in \mathbb{R}_{\geq 0}
\end{array} \right\rceil
\]

\( \iff E \leq \text{lfp } \Phi_f. \)
Our Rule

\[ E \leq \Phi_f(E) \land (1. \ \text{Exp}(T^{\neg \varphi}) < \infty \land 2. \ \lambda s. [\varphi]_{wp}[C](|E - E(s)|) \leq K \text{ for } K \in \mathbb{R}_{\geq 0}) \implies E \leq \text{lfp} \Phi_f. \]

1. Expected finite number of loop iterations.
2. Expected change is bounded by constant.

1. Expected finite number of loop iterations.
2. Expected change is bounded by constant.
Our Rule

<table>
<thead>
<tr>
<th>Condition</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( E \leq \Phi_f(E) ) ^ ( \wedge )</td>
<td></td>
</tr>
<tr>
<td>1. ( \text{Exp}(T^{\neg\varphi}) &lt; \infty )</td>
<td></td>
</tr>
<tr>
<td>2. ( \lambda s. [\varphi] \text{wp}[C],(</td>
<td>E - E(s)</td>
</tr>
</tbody>
</table>
## Lower Bounds

\[ \text{wp} \left[ \text{while}(\varphi)\{ C \} \right](f) \]

### Our Rule

\[
E \leq \Phi_f(E) \land
\begin{align*}
1. \quad & \text{Exp}(T^{\neg \varphi}) < \infty \\
2. \quad & \lambda s. [\varphi] \text{wp}[C]\left(|E - E(s)|\right) \leq K \quad \text{for} \quad K \in \mathbb{R}_{\geq 0} \implies E \leq \text{wp}[\text{loop}](f).
\end{align*}
\]
### Lower Bounds

$$\text{wp} \left[ \text{while}(\varphi)\{C\} \right](f)$$

<table>
<thead>
<tr>
<th>Our Rule</th>
</tr>
</thead>
</table>
| \[ E \leq \Phi_f(E) \land \begin{align*}
1. \quad & \text{Exp} \left( T^{\neg \varphi} \right) < \infty \\
2. \quad & \lambda s. \left[ \varphi \right]_{\text{wp}} \left[ C \right] \left( |E - E(s)| \right) \leq K \quad \text{for} \quad K \in \mathbb{R}_{\geq 0}
\end{align*} \implies E \leq \text{wp} \left[ \text{loop} \right] (f). \] |

---

**POPL 2020 – Aiming Low is Harder – Hark, Kaminski, Giesl, Katoen – 1/25/20 – 14**
Our Rule

\[
E \leq \Phi_f(E) \land (1. \ \text{Exp}(T^{\neg \varphi}) < \infty \land 2. \ \lambda s. [\varphi] wp [C] (|E - E(s)|) \leq K \text{ for } K \in \mathbb{R}_{\geq 0})
\]

\[
\text{while}(a = 1)\{
\{a := 0\}^{\frac{1}{2}} \{x := x + 1\}
\}
\]
Our Rule

\[ E \leq \Phi_f(E) \land \]
\[ 1. \, \text{Exp} (T^{\neg \varphi}) < \infty \]
\[ 2. \, \lambda s. [\varphi] \wp [C] (|E - E(s)|) \leq K \text{ for } K \in \mathbb{R}_{\geq 0} \]
\[ \implies E \leq \wp [\text{loop}] (f). \]

while (a = 1) {
    { a := 0 } \left[ \frac{1}{2} \right] { x := x + 1 }
}
\[ E = x + [a = 1] \]
Our Rule

\[ E \leq \Phi_f(E) \land \begin{array}{l}
1. \text{Exp}(T^\lnot \varphi) < \infty \\
2. \lambda s. [\varphi] \text{wp}[C](|E - E(s)|) \leq K \text{ for } K \in \mathbb{R}_{\geq 0} \end{array} \Rightarrow E \leq \text{wp}[\text{loop}](f). \]

\[
\text{while}(a = 1)\
\{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \} 
\}

\[ E = x + [a = 1] \]
Our Rule

\[
E \leq \Phi_f(E) \land \\
1. \text{Exp}(T^{\neg\varphi}) < \infty \\
2. \lambda s. [\varphi] \wp[[C]](|E - E(s)|) \leq K \quad \text{for} \quad K \in \mathbb{R}_{\geq 0} \\
\implies E \leq \wp[[\text{loop}]] (f).
\]

while (a = 1) {
  \{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \}
}

\[
E = x + [a = 1] \\
\lambda s. [a = 1] \cdot \wp[[\{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \}]] (|E - E(s)|) (s)
\]
Our Rule

\[ E \leq \Phi_f(E) \land \]
\[ \begin{align*}
1. & \; \text{Exp}(T^{-\varphi}) < \infty \\
2. & \; \lambda s. [\varphi] \text{wp}[C](|E - E(s)|) \leq K \text{ for } K \in \mathbb{R}_{\geq 0}
\end{align*} \]

\[ \implies E \leq \text{wp}[\text{loop}](f). \]

\textbf{while} \((a = 1)\{\)
\begin{align*}
\{ & \; a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \} \\
\}
\end{align*}

\[ E = x + [a = 1] \]

\[ \lambda s. [a = 1] \cdot \text{wp}[\{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \}] (|E - E(s)|) (s) \]
\[ = \frac{1}{2} \cdot |x + [0 = 1] - (x + [a = 1])| + \frac{1}{2} \cdot |x + 1 + [a = 1] - (x + [a = 1])| \]
Our Rule

\[ E \leq \Phi_f(E) \land \]
\[ 1. \ \text{Exp}(T^{\neg \varphi}) < \infty \]
\[ 2. \ \lambda s. \ [\varphi] \wp [C] (|E - E(s)|) \leq K \quad \text{for} \quad K \in \mathbb{R}_{\geq 0} \quad \implies E \leq \wp [\text{loop}] (f). \]

while \((a = 1)\)\

\{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \}\

\[
E = x + [a = 1] \\
\lambda s. [a = 1] \cdot \wp \left[ \left\{ a := 0 \right\} \left[ \frac{1}{2} \right] \left\{ x := x + 1 \right\} \right] (|E - E(s)|) \quad (s) \\
= \frac{1}{2} \cdot |x + [0 = 1] - (x + [a = 1])| + \frac{1}{2} \cdot |x + 1 + [a = 1] - (x + [a = 1])| \\
= \frac{1}{2} \cdot |\neg [a = 1]| + \frac{1}{2} \cdot |1| \]
Lower Bounds

Our Rule

\[
E \leq \Phi_f(E) \land \\
1. \Exp(T^{-\varphi}) < \infty \\
2. \lambda s. [\varphi] \wp[[C]](|E - E(s)|) \leq K \quad \iff \quad E \leq \wp[[\text{loop}]](f).
\]

while \((a = 1)\)\{
  \{a := 0\} \left[\frac{1}{2}\right] \{x := x + 1\}
}\}

\[
E = x + \lfloor a = 1 \rfloor \\
\lambda s. [a = 1] \cdot \wp[[\{a := 0\} \left[\frac{1}{2}\right] \{x := x + 1\}]](|E - E(s)|) (s) \\
= \frac{1}{2} \cdot |x + [0 = 1] - (x + [a = 1])| + \frac{1}{2} \cdot |x + 1 + [a = 1] - (x + [a = 1])| \\
= \frac{1}{2} \cdot |- [a = 1]| + \frac{1}{2} \cdot |1| \leq 1 = K \quad \text{(constant)}
\]
Lower Bounds

Our Rule

\[
E \leq \Phi_f(E) \quad \land \quad \lambda s. [\varphi] \text{wp}[\{C\}] (|E - E(s)|) \leq K \quad \text{for} \quad K \in \mathbb{R}_{\geq 0} \quad \implies \quad E \leq \text{wp}[\text{loop}] (f).
\]

while \((a = 1)\)\
\{ \{ a := 0 \} [\frac{1}{2}] \{ x := x + 1 \} \}
\]

\[
E = x + [a = 1]
\]
Our Rule

\[
\begin{align*}
E & \leq \Phi_f(E) \quad \land \\ & \quad 1. \ \text{Exp} \left( T^\neg \varphi \right) < \infty \\
& \quad 2. \ \lambda s. \left[ \varphi \right] \text{wp} \left[ C \right] (|E - E(s)|) \leq K \quad \text{for} \quad K \in \mathbb{R}_{\geq 0}
\end{align*}
\]

\[\text{while}(a = 1)\{\]
\[\{ a := 0 \} \frac{1}{2} \{ x := x + 1 \}\]
\[\}\]

\[E = x + [a = 1]\]

Expected finite looping time: \(\text{Exp} \left( T^{a \neq 1} \right) \leq 2 \cdot [a = 1] < \infty\).
Our Rule

\[
E \leq \Phi_f(E) \land
1. \ \text{Exp}(T^{\neg \varphi}) < \infty
2. \ \lambda s. [\varphi] wp [C] (|E - E(s)|) \leq K \text{ for } K \in \mathbb{R}_{\geq 0} \implies E \leq wp [\text{loop}] (f).
\]

\[
\text{while}(a = 1)\
\{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \}
\}
\]

\[
E = x + [a = 1]
\]
Lower Bounds

\[ \text{wp}[\text{while}(\varphi)\{C\}](f) \]

Our Rule

\[ E \leq \Phi_f(E) \land \begin{align*}
1. \quad & \text{Exp}(T^{\neg\varphi}) < \infty \\
2. \quad & \lambda s. [\varphi] \text{wp}[C](|E - E(s)|) \leq K \quad \text{for} \quad K \in \mathbb{R}_{\geq 0}
\end{align*} \]

while \((a = 1)\{\)
\[ \{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \} \]
\}

\[ E = x + [a = 1] \]

\[ \rightarrow E \text{ is a lower bound.} \]
Our Rule

\[ E \leq \Phi_f(E) \land \begin{cases} \exp(T^{\neg \varphi}) < \infty \\ \lambda s. [\varphi] \wp[[C]](|E - E(s)|) \leq K \text{ for } K \in \mathbb{R}_{\geq 0} \end{cases} \Rightarrow E \leq \wp[[\text{loop}]](f). \]

while \( a = 1 \) \{ \\
{ a := 0 } \} \left[ \frac{1}{2} \right] \{ x := x + 1 \} \}

\[ E = x + [a = 1] \]

\[ \rightarrow E \text{ is a lower bound.} \]

\[ \rightarrow E = \wp[[\text{while}(a = 1)\{\{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \} \}]](x). \]
### Our Rule

\[
E \leq \Phi_f(E) \land \begin{align*}
1. \quad & \text{Exp}(T^{\neg \varphi}) < \infty \\
2. \quad & \lambda s. [\varphi] \text{wp} [C] (|E - E(s)|) \leq K \quad \text{for} \quad K \in \mathbb{R}_{\geq 0} \implies E \leq \text{wp}[\text{loop}](f).
\end{align*}
\]

while (a = 1) {
  \{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \}
}

\[wp[\text{while}(\varphi)\{C\}](f)\]
## Lower Bounds

**Our Rule**

\[
E \leq \Phi_f(E) \land \\
1. \ Exp(T^{\neg \varphi}) < \infty \\
2. \ \lambda s. [\varphi] wp[C] (|E - E(s)|) \leq K \quad \text{for} \quad K \in \mathbb{R}_{\geq 0} \implies E \leq wp[loop](f).
\]

while \((a = 1)\{
\{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \}
\}
\]

\[
E' = x + [a = 1] \cdot (1 + 2^x)
\]
**Lower Bounds**

\[ \text{wp} \left[ \text{while}(\varphi) \{ C \} \right](f) \]

---

**Our Rule**

\[ E \leq \Phi_f(E) \land \]

1. \( \text{Exp}(T^{\neg \varphi}) < \infty \)

2. \( \lambda s. [\varphi] \text{wp} \left[ C \right] (|E - E(s)|) \leq K \) for \( K \in \mathbb{R}_{\geq 0} \)

\[ \implies E \leq \text{wp} \left[ \text{loop} \right] (f). \]

---

**while** \((a = 1)\{ \)

\[ \{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \} \]

\[ \}

\[ E' = x + [a = 1] \cdot (1 + 2^x) \]

\[ \lambda s. [a = 1] \cdot \text{wp} \left[ \left\{ a := 0 \right\} \left[ \frac{1}{2} \right] \left\{ x := x + 1 \right\} \right] (|E' - E'(s)|)(s) \]
Lower Bounds

\[ \wp[\text{while}(\varphi)\{C\}](f) \]

\begin{itemize}
  \item Our Rule
  \[ E \leq \Phi_f(E) \land \\
  1. \text{Exp}(T^{\neg\varphi}) < \infty \\
  2. \lambda s. [\varphi] \wp[\llbracket C \rrbracket](|E - E(s)|) \leq K \text{ for } K \in \mathbb{R}_{\geq 0} \implies E \leq \wp[\llbracket \text{loop} \rrbracket](f). \]
\end{itemize}

while \((a = 1)\{ \\
  \{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \} \} \)

\[ E' = x + [a = 1] \cdot (1 + 2^x) \]

\[ \lambda s. [a = 1] \cdot \wp[\llbracket \{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \} \rrbracket](|E' - E'(s)|)(s) \geq [a = 1] \cdot (1 + 2^x) \]
Our Rule

\[
E \leq \Phi_f(E) \land \\
1. \text{Exp}(T^{-\varphi}) < \infty \land \\
2. \lambda s. [\varphi] wp[[C]](|E - E(s)|) \leq K \text{ for } K \in \mathbb{R}_{\geq 0} \implies E \leq wp[\text{loop}](f).
\]

while \((a = 1)\)\
\[
\{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \}
\]

\[
E' = x + [a = 1] \cdot (1 + 2^x)
\]

\[
\lambda s. [a = 1] \cdot wp[[\{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \}]](|E' - E'(s)|)(s) \\
\geq [a = 1] \cdot (1 + 2^x)
\]
Lower Bounds

Our Rule

\[ E \leq \Phi_f(E) \land \left( 1. \text{Exp} \left( T^{\neg \varphi} \right) < \infty \right) \land \left. 2. \lambda s. [\varphi] wp [C] (|E - E(s)|) \leq K \right) \text{for } K \in \mathbb{R}_{\geq 0} \implies E \leq wp [\text{loop}] (f). \]

while\((a = 1)\)\{ \\
  \{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \} \\
\}

\[ E' = x + [a = 1] \cdot (1 + 2^x) \]

\(\rightarrow\) Rule is not applicable!
### Lower Bounds

- **Our Rule**

\[
E \leq \Phi_f(E) \land \\
1. \ \text{Exp} \left(T^{-\varphi}\right) < \infty \\
2. \ \lambda s. \left[\varphi\right] \wp \llbracket C \rrbracket (|E - E(s)|) \leq K \text{ for } K \in \mathbb{R}_{\geq 0} \implies E \leq \wp \llbracket \text{loop} \rrbracket (f).
\]

### Example

```plaintext
while (a = 1) {
    { a := 0 } [\frac{1}{2}] { x := x + 1 }
}

E' = x + [a = 1] \cdot (1 + 2^x)

\rightarrow \text{Good, since } E' \text{ is not a lower bound.}
```
Our Rule

\[
E \leq \Phi_f(E) \quad \land \\
1. \text{Exp} (T^-\varphi) < \infty \\
2. \lambda s. [\varphi] \text{wp} [C] (|E - E(s)|) \leq K \quad \text{for} \quad K \in \mathbb{R}_{\geq 0} \\
\implies E \leq \text{wp} [\text{loop}] (f).
\]

while \((a = 1)\{
\{ a := 0 \} \left[ \frac{1}{2} \right] \{ x := x + 1 \} 
\}

\text{Expected finite looping time: } \text{Exp} (T^-\varphi) \leq 2 \cdot \lambda s. [\varphi] \text{wp} [C] (|E - E(s)|) < \infty.

\text{→ Rule is not applicable!} \quad \text{→ Good, since } E' \text{ is not a lower bound.} \quad \text{→ Easily checkable.}
### Our Rule

\[
E \leq \Phi_f(E) \land \\
1. \ \text{Exp}(T^{\neg \varphi}) < \infty \land \\
2. \ \lambda s. \ [\varphi] \text{wp} [C] (|E - E(s)|) \leq K \quad \text{for} \ K \in \mathbb{R}_{\geq 0} \implies E \leq \text{wp}[\text{loop}](f).
\]

### Example

```latex
while(a = 1)\
  \{ a := 0 \} [\frac{1}{2}] \{ x := x + 1 \}
```

→ Easily checkable.
Lower Bounds

Contribution

\[ \text{wp} \left[ \text{while} (\varphi) \{ C \} \right] (f) \]
Contribution

To best of our knowledge:

→ First inductive rule for lower bounds.
Contribution

To best of our knowledge:

→ First inductive rule for lower bounds.

→ No reasoning about limits of sequences.
Lower Bounds

Lower Bounds for Expected Runtimes

• Similar rule for expected runtimes (ert [Kaminski et al. 16]).
• Only side condition: $\lambda_s$. 

```latex
\text{Coupon Collector:} \ x := 0 \text{ while}(x < N)
\{ i := N + 1 \text{ while}(x < i) 
\{ i := \text{Unif}[1..N] \}
\} \ x := x - 1
\}
\} \]
T[x/N] = \left\lfloor 0 < x \leq N \right\rfloor \cdot N \cdot H_N + \left\lfloor N < x \right\rfloor \cdot (N \cdot H_N + N - x) = N \cdot \left( 1 + \frac{1}{2} + \cdots + \frac{1}{N} \right)
```

is lower bound.
Lower bound is strict but asymptotically optimal.
$H_N$ appears in real world algorithm-analysis!
Lower Bounds for Expected Runtimes

- Similar rule for expected runtimes (ert [Kaminski et al. 16]).
Lower Bounds for Expected Runtimes

• Similar rule for expected runtimes (ert [Kaminski et al. 16]).
• Only side condition: $\lambda s. \wp \left[ C \right] (|T - T(s)|) \leq K$ for $K \in \mathbb{R}_{\geq 0}$.

$\wp[\text{while}(\varphi)\{C\}](f)$

Lower Bounds
Lower Bounds

Lower Bounds for Expected Runtimes

- Similar rule for expected runtimes (ert [Kaminski et al. 16]).
- Only side condition: \( \lambda s. [\varphi] \text{wp}[C] (|T - T(s)|) \leq K \) for \( K \in \mathbb{R}_{\geq 0} \).

Coupon Collector:

\[
T \left[ \frac{x}{N} \right] = \begin{cases} 
0 < x \leq N \cdot N \cdot H_x + \left[ N < x \right] \cdot (N \cdot H_N + N - x) \\
\end{cases}
\]

is lower bound.

Lower bound is strict but asymptotically optimal.

\( H_N \) appears in real world algorithm-analysis!
Lower Bounds for Expected Runtimes

- Similar rule for expected runtimes (ert [Kaminski et al. 16]).
- Only side condition: $\lambda s. [\varphi]_{\text{wp}} \llbracket C \rrbracket (|T - T(s)|) \leq K$ for $K \in \mathbb{R}_{\geq 0}$.

**Coupon Collector:**

```
x := N
while (0 < x) {
    i := N + 1
    while (x < i) {
        i := Unif[1..N]
    }
    x := x - 1
}
```
Lower Bounds for Expected Runtimes

- Similar rule for expected runtimes (ert [Kaminski et al. 16]).
- Only side condition: \( \lambda s. [\varphi]_{\text{wp}} [\llbracket C \rrbracket] (|T - T(s)|) \leq K \) for \( K \in \mathbb{R}_{\geq 0} \).

**Coupon Collector:**

\[
x := N \\
\text{while}(0 < x)\{
  i := N + 1 \\
  \text{while}(x < i)\{
    i := \text{Unif}[1..N] \\
  \}
  x := x - 1
\}
\]

\[
T = [0 < x \leq N] \cdot N \cdot H_x + [N < x] \cdot (N \cdot H_N + N - x)
\]
Lower Bounds

**Lower Bounds for Expected Runtimes**

- Similar rule for expected runtimes (ert [Kaminski et al. 16]).
- Only side condition: $\lambda s. [\varphi ] wp \llbracket C \rrbracket (|T - T(s)|) \leq K$ for $K \in \mathbb{R}_{\geq 0}$.

**Coupon Collector:**

```
x := N
while (0 < x)
    { i := N + 1
        while (x < i)
            { i := Unif[1..N]
                }
        x := x - 1
    }
```

\[ T = \left[ 0 < x \leq N \right] \cdot N \cdot H_x + \left[ N < x \right] \cdot \left( N \cdot H_N + N - x \right) \]

\[ T[x/N] \]

is lower bound.
Lower Bounds for Expected Runtimes

- Similar rule for expected runtimes (ert [Kaminski et al. 16]).
- Only side condition: \( \lambda_s \cdot [\varphi]_{\text{wp}}\left[ C \right] (|T - T(s)|) \leq K \) for \( K \in \mathbb{R}_{\geq 0} \).

**Coupon Collector:**

```plaintext
x := N
while (0 < x)
{
    i := N + 1
    while (x < i)
    {
        i := Unif[1..N]
    }
    x := x - 1
}
```

\[ T = [0 < x \leq N] \cdot N \cdot \mathcal{H}_x + [N < x] \cdot (N \cdot \mathcal{H}_N + N - x) \]

\( N \cdot \mathcal{H}_N = N \cdot (1 + \frac{1}{2} + \cdots + \frac{1}{N}) \) is lower bound.
Lower Bounds for Expected Runtimes

- Similar rule for expected runtimes (ert [Kaminski et al. 16]).
- Only side condition: \( \lambda_s [\varphi] \text{wp} \left[ C \right] (|T - T(s)|) \leq K \) for \( K \in \mathbb{R}_{\geq 0} \).

**Coupon Collector:**

\[
x := N \\
\text{while}(0 < x)\{ \\
\quad i := N + 1 \\
\quad \text{while}(x < i)\{ \\
\quad \quad i := \text{Unif}[1..N] \\
\quad \} \\
\quad x := x - 1 \\
\}\]

\[
T = [0 < x \leq N] \cdot N \cdot H_x + [N < x] \cdot (N \cdot H_N + N - x)
\]

\( N \cdot H_N = N \cdot \left(1 + \frac{1}{2} + \cdots + \frac{1}{N}\right) \) is lower bound.

Lower bound is strict but asymptotically optimal.
Lower Bounds for Expected Runtimes

- Similar rule for expected runtimes (ert [Kaminski et al. 16]).
- Only side condition: \( \lambda s. [\varphi] \text{wp } [C] (|T - T(s)|) \leq K \) for \( K \in \mathbb{R}_{\geq 0} \).

**Coupon Collector:**

\[
x := N \\
\text{while}(0 < x) \{ \\
\quad i := N + 1 \\
\quad \text{while}(x < i) \{ \\
\quad \quad i := \text{Unif}[1..N] \\
\quad \} \\
\quad x := x - 1 \\
\} \\
\]

\[
T = [0 < x \leq N] \cdot N \cdot \mathcal{H}_x + [N < x] \cdot (N \cdot \mathcal{H}_N + N - x)
\]

\( N \cdot \mathcal{H}_N = N \cdot (1 + \frac{1}{2} + \cdots + \frac{1}{N}) \) is lower bound.

Lower bound is strict but asymptotically optimal.

\( \mathcal{H}_N \) appears in real world algorithm-analysis!
Conclusion

Outline

Motivation

Upper Bounds [Park]

Lower Bounds

Conclusion
Conclusion

Summary
Conclusion

Summary

• Inductive rule for deriving lower bounds on (unbounded) postexpectations.
Conclusion

Summary

- Inductive rule for deriving lower bounds on (unbounded) postexpectations.
- Inductive rule for deriving lower bounds on expected runtimes.
Conclusion

Summary

- Inductive rule for deriving lower bounds on (unbounded) postexpectations.
- Inductive rule for deriving lower bounds on expected runtimes.
Conclusion

Summary

- Inductive rule for deriving lower bounds on (unbounded) postexpectations.
- Inductive rule for deriving lower bounds on expected runtimes.

Future Work
Conclusion

Summary
- Inductive rule for deriving lower bounds on (unbounded) postexpectations.
- Inductive rule for deriving lower bounds on expected runtimes.

Future Work
- Generalization to mixed sign expectations.
Conclusion

Summary

- Inductive rule for deriving lower bounds on (unbounded) postexpectations.
- Inductive rule for deriving lower bounds on expected runtimes.

Future Work

- Generalization to mixed sign expectations.
- Automation.
Conclusion

Summary
- Inductive rule for deriving lower bounds on (unbounded) postexpectations.
- Inductive rule for deriving lower bounds on expected runtimes.

Future Work
- Generalization to mixed sign expectations.
- Automation.
Conclusion

Summary

- Inductive rule for deriving lower bounds on (unbounded) postexpectations.
- Inductive rule for deriving lower bounds on expected runtimes.

Future Work

- Generalization to mixed sign expectations.
- Automation.

Thank you