Lower Bounds for Possibly Divergent Probabilistic Programs

Shenghua Feng, **Mingshuai Chen**, Han Su, Benjamin L. Kaminski,
Joost-Pieter Katoen, Naijun Zhan

OOPSLA · Lisbon · October 2023

A Fun Fact

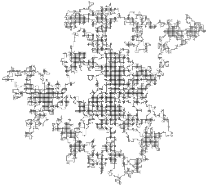
"A drunk man will find his way home, but a drunk bird may get lost forever."

— Shizuo Kakutani

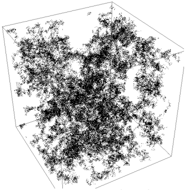
A Fun Fact

"A drunk man will find his way home, but a drunk bird may get lost forever."

— Shizuo Kakutani



©Wikipedia



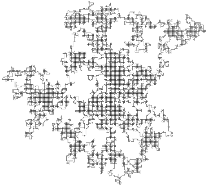
©StackExchange

A 2-D symmetric random walk on a lattice returns to the origin *almost-surely*, yet *not* its 3-D counterpart [Pólya, Math. Ann. '21].

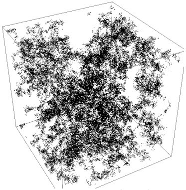
A Fun Fact

"A drunk man will find his way home, but a drunk bird may get lost forever."

— Shizuo Kakutani



©Wikipedia



©StackExchange

A 2-D symmetric random walk on a lattice returns to the origin *almost-surely*, yet *not* its 3-D counterpart [Pólya, Math. Ann. '21].

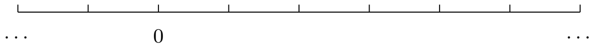
Question : How to compute *sound approx.* of the returning probability of the bird?

Probabilistic Programs

$$C_{\mathsf{brw}}: \quad \texttt{while}\,(\,n > 0\,)\,\{\,n \coloneqq n - 1\ [1/3]\ n \coloneqq n + 1\,\}$$

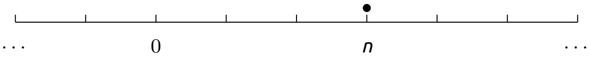
Probabilistic Programs

$$C_{\text{brw}}: \quad \texttt{while}\,(\,n > 0\,)\,\{\,n := n - 1\;[1/3]\;n := n + 1\,\}$$



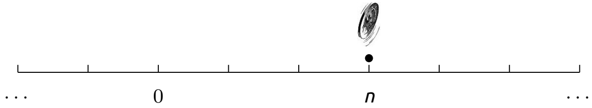
Probabilistic Programs

$$C_{\mathsf{brw}}: \quad \texttt{while}\,(\,n > 0\,)\,\{\,n := n - 1\;[1/3]\;n := n + 1\,\}$$



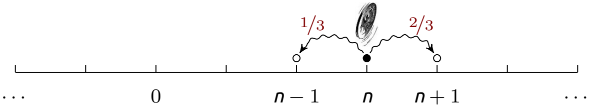
Probabilistic Programs

$$C_{\mathsf{brw}}\colon \quad \texttt{while}\,(\,n > 0\,)\,\{\, n \coloneqq n - 1 \; [1/3] \; n \coloneqq n + 1 \,\}$$



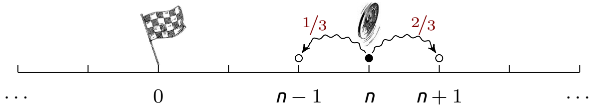
Probabilistic Programs

$$C_{\mathsf{brw}}: \quad \texttt{while}\,(\, n > 0\,)\,\{\, n \coloneqq n-1 \;[1/3]\; n \coloneqq n+1 \,\}$$



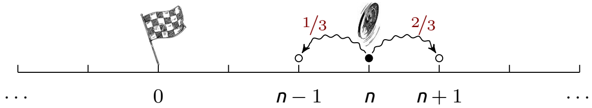
Probabilistic Programs

C_{brw} : $\texttt{while}\,(\,n > 0\,)\,\{\, n := n - 1 \;[1/3]\; n := n + 1 \,\}$



Probabilistic Programs

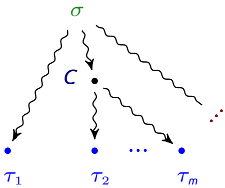
$$C_{\mathsf{brw}}: \quad \texttt{while}\,(\,n > 0\,)\,\{\, n \coloneqq n - 1 \; [1/3] \; n \coloneqq n + 1 \,\}$$



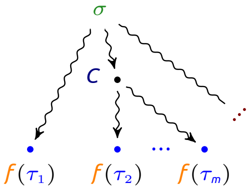
"The crux of probabilistic programming is to treat normal-looking programs as if they were probability distributions."

— Michael Hicks, The PL Enthusiast

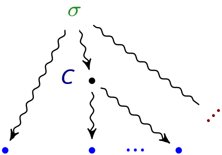
Quantitative Reasoning about Probabilistic Loops [Kozen; McIver, Morgan; Kaminski]



Quantitative Reasoning about Probabilistic Loops [Kozen; McIver, Morgan; Kaminski]



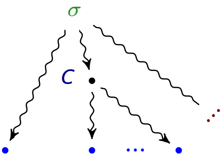
Quantitative Reasoning about Probabilistic Loops [Kozen; McIver, Morgan; Kaminski]



$$\mathrm{wp}[\![C]\!](f)(\sigma) \triangleq \mathrm{Exp}\Big[\, f(\tau_1) \qquad f(\tau_2) \qquad f(\tau_m) \,\Big]$$

Quantitative Reasoning about Probabilistic Loops

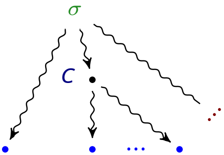
[Kozen; McIver, Morgan; Kaminski]



$$\mathsf{wp}[\![C]\!](f)(\sigma) \triangleq \mathsf{Exp}\big[\, f(\tau_1) \quad f(\tau_2) \quad \cdots \quad f(\tau_m) \,\big]$$

$$\mathsf{wp}[\![n := 5]\!]\,(n) = 5$$

Quantitative Reasoning about Probabilistic Loops [Kozen; McIver, Morgan; Kaminski]

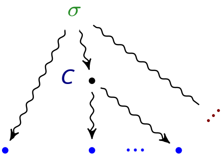


$$\mathsf{wp}[\![C]\!](f)(\sigma) \triangleq \mathsf{Exp}\left[\, f(\tau_1) \quad f(\tau_2) \quad f(\tau_m) \right]$$

$$\mathsf{wp}[\![n := 5]\!]\,(n) \;=\; 5$$

$$\mathsf{wp}[\![n := n-1\ [1/3]\ n := n+1]\!]\,(n) \;=\; 1/3 \cdot (n-1) + 2/3 \cdot (n+1) \;=\; n + 1/3$$

Quantitative Reasoning about Probabilistic Loops [Kozen; McIver, Morgan; Kaminski]



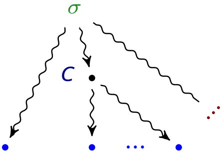
$$\mathrm{wp}[\![C]\!](f)(\sigma) \triangleq \mathrm{Exp}\big[\, f(\tau_1) \quad f(\tau_2) \quad \cdots \quad f(\tau_m) \,\big]$$

$$\mathrm{wp}[\![n := 5]\!]\,(n) \;=\; 5$$

$$\mathrm{wp}[\![n := n-1\;[1/3]\;n := n+1]\!]\,(n) \;=\; 1/3 \cdot (n-1) + 2/3 \cdot (n+1) \;=\; n + 1/3$$

$$\mathrm{wp}[\![\texttt{while}\,(\,n > 0\,)\,\{\, n := n-1\;[1/3]\;n := n+1 \,\}]\!]\,(1) \;=\; ?$$

Quantitative Reasoning about Probabilistic Loops [Kozen; McIver, Morgan; Kaminski]



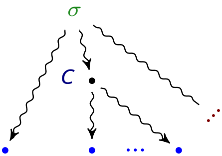
$$\mathsf{wp}[\![C]\!](f)(\sigma) \triangleq \mathsf{Exp}\left[f(\tau_1) \quad f(\tau_2) \quad f(\tau_m) \right]$$

$$\mathsf{wp}[\![n := 5]\!]\,(n) \;=\; 5$$

$$\mathsf{wp}[\![n := n-1\ [1/3]\ n := n+1]\!]\,(n) \;=\; 1/3 \cdot (n-1) + 2/3 \cdot (n+1) \;=\; n + 1/3$$

$$\mathsf{wp}[\![\mathtt{while}\,(\,n > 0\,)\,\{\,n := n-1\ [1/3]\ n := n+1\,\}]\!]\,(1) \;=\; [n < 0] + [n \geq 0] \cdot (1/2)^n$$

Quantitative Reasoning about Probabilistic Loops [Kozen; McIver, Morgan; Kaminski]



$$\mathsf{wp}[\![C]\!](f)(\sigma) \triangleq \mathsf{Exp}\left[\, f(\tau_1) \quad f(\tau_2) \quad \cdots \quad f(\tau_m) \,\right]$$

$$\mathsf{wp}[\![n := 5]\!]\,(n) \;=\; 5$$

$$\mathsf{wp}[\![n := n-1\ [1/3]\ n := n+1]\!]\,(n) \;=\; 1/3 \cdot (n-1) + 2/3 \cdot (n+1) \;=\; n + 1/3$$

$$\mathsf{wp}[\![\texttt{while}\,(\,n > 0\,)\,\{\,n := n-1\ [1/3]\ n := n+1\,\}]\!]\,(1) \;=\; [n < 0] + [n \geq 0] \cdot (1/2)^n$$

$$\mathsf{wp}[\![\texttt{while}\,(\,\varphi\,)\,\{\,C\,\}]\!]\,(f) \;=\; \mathsf{lfp}\;\Phi_f \;=\; ?$$

Bounding the Least Fixed Point

$$l \;\preceq\; \mathsf{lfp}\,\Phi_f \;\preceq\; u$$

Bounding the Least Fixed Point

$$l \;\preceq\; \mathrm{lfp}\,\Phi_f \;\preceq\; u$$

- Upper bounds (Park induction) :

$$\Phi_f(u) \preceq u \quad \text{implies} \quad \mathrm{lfp}\,\Phi_f \preceq u\,.$$

Bounding the Least Fixed Point

$$l \;\preceq\; \mathsf{lfp}\,\Phi_f \;\preceq\; u$$

- Upper bounds (Park induction) :

$$\Phi_f(u) \;\preceq\; u \quad \text{implies} \quad \mathsf{lfp}\,\Phi_f \;\preceq\; u\,.$$

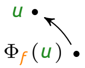
u •

Bounding the Least Fixed Point

$$l \preceq \mathsf{lfp}\, \Phi_f \preceq u$$

- Upper bounds (Park induction) :

$$\Phi_f(u) \preceq u \quad \text{implies} \quad \mathsf{lfp}\, \Phi_f \preceq u\,.$$

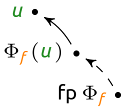


Bounding the Least Fixed Point

$$l \;\preceq\; \mathrm{lfp}\;\Phi_f \;\preceq\; u$$

- Upper bounds (Park induction) :

$$\Phi_f(u) \;\preceq\; u \quad \text{implies} \quad \mathrm{lfp}\;\Phi_f \;\preceq\; u\,.$$

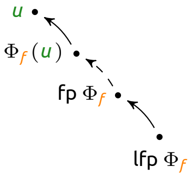


Bounding the Least Fixed Point

$$l \;\preceq\; \mathsf{lfp}\,\Phi_f \;\preceq\; u$$

- Upper bounds (Park induction) :

$$\Phi_f(u) \;\preceq\; u \quad \text{implies} \quad \mathsf{lfp}\,\Phi_f \;\preceq\; u\,.$$



Bounding the Least Fixed Point

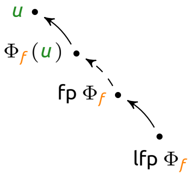
$$l \preceq \mathsf{lfp}\ \Phi_f \preceq u$$

- Upper bounds (Park induction) :

$$\Phi_f(u) \preceq u \quad \text{implies} \quad \mathsf{lfp}\ \Phi_f \preceq u\,.$$

- Lower bounds :

$$l \preceq \Phi_f(l) \quad \text{implies} \quad l \preceq \mathsf{lfp}\ \Phi_f\,.$$



Bounding the Least Fixed Point

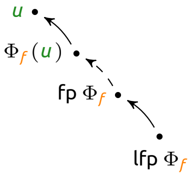
$$l \;\preceq\; \mathsf{lfp}\ \Phi_f \;\preceq\; u$$

- Upper bounds (Park induction) :

$$\Phi_f(u) \;\preceq\; u \quad \text{implies} \quad \mathsf{lfp}\ \Phi_f \;\preceq\; u\,.$$

- Lower bounds :

$$l \;\preceq\; \Phi_f(l) \quad \text{implies} \quad l \;\preceq\; \mathsf{lfp}\ \Phi_f\,.$$



Bounding the Least Fixed Point

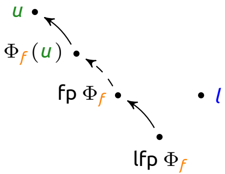
$$l \;\preceq\; \mathsf{lfp}\ \Phi_f \;\preceq\; u$$

- Upper bounds (Park induction) :

$$\Phi_f(u) \preceq u \quad \text{implies} \quad \mathsf{lfp}\ \Phi_f \preceq u\,.$$

- Lower bounds :

$$l \preceq \Phi_f(l) \quad \text{~~implies~~} \quad l \preceq \mathsf{lfp}\ \Phi_f\,.$$



Bounding the Least Fixed Point

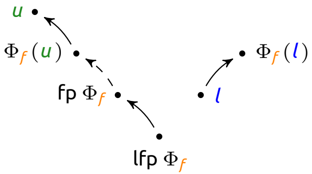
$$l \;\preceq\; \mathrm{lfp}\,\Phi_f \;\preceq\; u$$

- Upper bounds (Park induction) :

$$\Phi_f(u) \;\preceq\; u \quad \text{implies} \quad \mathrm{lfp}\,\Phi_f \;\preceq\; u\,.$$

- Lower bounds :

$$l \;\preceq\; \Phi_f(l) \quad \cancel{\text{implies}} \quad l \;\preceq\; \mathrm{lfp}\,\Phi_f\,.$$



Bounding the Least Fixed Point

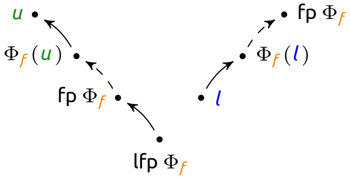
$$l \;\preceq\; \mathsf{lfp}\ \Phi_f \;\preceq\; u$$

- Upper bounds (Park induction) :

$$\Phi_f(u) \preceq u \quad \text{implies} \quad \mathsf{lfp}\ \Phi_f \preceq u\,.$$

- Lower bounds :

$$l \preceq \Phi_f(l) \quad \cancel{\text{implies}} \quad l \preceq \mathsf{lfp}\ \Phi_f\,.$$



Bounding the Least Fixed Point

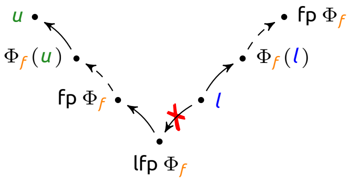
$$l \;\preceq\; \mathsf{lfp}\,\Phi_f \;\preceq\; u$$

- Upper bounds (Park induction) :

$$\Phi_f(u) \preceq u \quad \text{implies} \quad \mathsf{lfp}\,\Phi_f \preceq u\,.$$

- Lower bounds :

$$l \preceq \Phi_f(l) \quad \cancel{\text{implies}} \quad l \preceq \mathsf{lfp}\,\Phi_f\,.$$



Bounding the Least Fixed Point

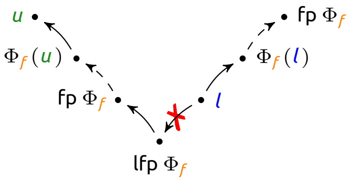
$$l \preceq \mathrm{lfp}\,\Phi_f \preceq u$$

- Upper bounds (Park induction) :

$$\Phi_f(u) \preceq u \quad \text{implies} \quad \mathrm{lfp}\,\Phi_f \preceq u\,.$$

- Lower bounds ([Hark et al., POPL '20]) :

$$l \preceq \Phi_f(l) \;\wedge\; l \text{ is uni. int.} \quad \text{implies} \quad l \preceq \mathrm{lfp}\,\Phi_f\,.$$



Bounding the Least Fixed Point

$$l \;\preceq\; \mathrm{lfp}\,\Phi_f \;\preceq\; u$$

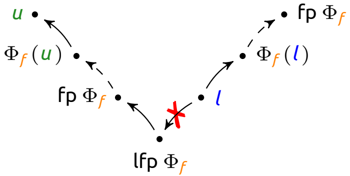
- Upper bounds (Park induction) :

$$\Phi_f(u) \preceq u \quad \text{implies} \quad \mathrm{lfp}\,\Phi_f \preceq u\,.$$

- Lower bounds ([Hark et al., POPL '20]) :

$$l \preceq \Phi_f(l) \;\wedge\; \boxed{l \text{ is uni. int.}} \quad \text{implies} \quad l \preceq \mathrm{lfp}\,\Phi_f\,.$$

almost-sure termination (AST)
bounded expectations
...



A New Proof Rule for Lower Bounds

Theorem (Guard-Strengthening Rule)

$$C_{\text{loop}}\colon\quad \texttt{while}\,(\,\varphi\,)\,\{\,C\,\} \quad\rightsquigarrow\quad C'_{\text{loop}}\colon\quad \texttt{while}\,(\,\varphi'\,)\,\{\,C\,\}$$

A New Proof Rule for Lower Bounds

Theorem (Guard-Strengthening Rule)

$$C_{\text{loop}}: \quad \texttt{while}\,(\,\varphi\,)\,\{\,C\,\} \quad \rightsquigarrow \quad C'_{\text{loop}}: \quad \texttt{while}\,(\,\varphi'\,)\,\{\,C\,\}$$

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathrm{wp}[\![C'_{\text{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \mathrm{wp}[\![C_{\text{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

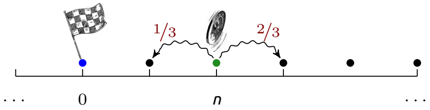
A New Proof Rule for Lower Bounds

Theorem (Guard-Strengthening Rule)

$$C_{\text{loop}}: \quad \texttt{while}\,(\,\varphi\,)\,\{\,C\,\} \quad \rightsquigarrow \quad C'_{\text{loop}}: \quad \texttt{while}\,(\,\varphi'\,)\,\{\,C\,\}$$

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathrm{wp}[\![C'_{\text{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \mathrm{wp}[\![C_{\text{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

$$C_{\text{brw}}: \quad \texttt{while}\,(0 < n)\,\{ \\ n := n-1\ [1/3]\ n := n+1\,\}$$



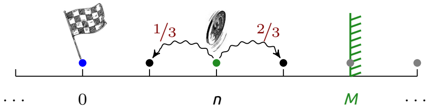
A New Proof Rule for Lower Bounds

Theorem (Guard-Strengthening Rule)

$$C_{\text{loop}}: \quad \texttt{while}\,(\,\varphi\,)\,\{\,C\,\} \quad \rightsquigarrow \quad C'_{\text{loop}}: \quad \texttt{while}\,(\,\varphi'\,)\,\{\,C\,\}$$

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathsf{wp}[\![C'_{\text{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \mathsf{wp}[\![C_{\text{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

$$C_{\text{brw}}: \quad \texttt{while}\,(0 < n)\,\{\; n := n-1\;[1/3]\; n := n+1\,\} \quad \rightsquigarrow \quad C^{M}_{\text{brw}}: \quad \texttt{while}\,(0 < n < M)\,\{\; n := n-1\;[1/3]\; n := n+1\,\}$$



A New Proof Rule for Lower Bounds

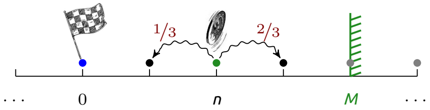
Theorem (Guard-Strengthening Rule)

$$C_{\text{loop}}: \quad \texttt{while}\,(\,\varphi\,)\,\{\,C\,\} \quad \rightsquigarrow \quad C'_{\text{loop}}: \quad \texttt{while}\,(\,\varphi'\,)\,\{\,C\,\}$$

$$\frac{\varphi' \implies \varphi \qquad l \preceq \text{wp}[\![C'_{\text{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \text{wp}[\![C_{\text{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

$$C_{\text{brw}}: \quad \texttt{while}\,(0 < n)\,\{\; n := n-1\;[1/3]\;n := n+1\,\} \quad \rightsquigarrow \quad C^M_{\text{brw}}: \quad \texttt{while}\,(0 < n < M)\,\{\; n := n-1\;[1/3]\;n := n+1\,\}$$

$$\text{wp}[\![C^M_{\text{brw}}]\!]\,([n \leq 0]\cdot 1) \;\preceq\; \text{wp}[\![C_{\text{brw}}]\!]\,(1)$$



A New Proof Rule for Lower Bounds

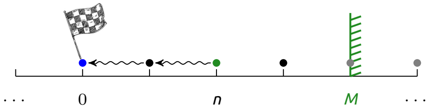
Theorem (Guard-Strengthening Rule)

$$C_{\text{loop}}\colon \quad \texttt{while}\,(\,\varphi\,)\,\{\,C\,\} \quad\rightsquigarrow\quad C'_{\text{loop}}\colon \quad \texttt{while}\,(\,\varphi'\,)\,\{\,C\,\}$$

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathrm{wp}[\![C'_{\text{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \mathrm{wp}[\![C_{\text{loop}}]\!]\,(f)}\ \text{(Guard-Strengthening)}$$

$$C_{\text{brw}}\colon \quad \texttt{while}\,(0 < n)\,\{\; n := n-1\;[1/3]\; n := n+1\,\} \quad\rightsquigarrow\quad C^{M}_{\text{brw}}\colon \quad \texttt{while}\,(0 < n < M)\,\{\; n := n-1\;[1/3]\; n := n+1\,\}$$

$$\boxed{\mathrm{wp}[\![C^{M}_{\text{brw}}]\!]\,([n \le 0]\cdot 1) \;\preceq\; \mathrm{wp}[\![C_{\text{brw}}]\!]\,(1)}$$



A New Proof Rule for Lower Bounds

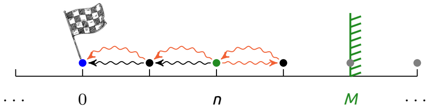
Theorem (Guard-Strengthening Rule)

$$C_{\text{loop}}: \quad \texttt{while}\,(\varphi)\,\{\,C\,\} \quad \rightsquigarrow \quad C'_{\text{loop}}: \quad \texttt{while}\,(\varphi')\,\{\,C\,\}$$

$$\frac{\varphi' \implies \varphi \qquad l \preceq \text{wp}[\![C'_{\text{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \text{wp}[\![C_{\text{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

$$C_{\text{brw}}: \quad \texttt{while}\,(0 < n)\,\{\; n := n-1\;[1/3]\; n := n+1\,\} \quad \rightsquigarrow \quad C^{M}_{\text{brw}}: \quad \texttt{while}\,(0 < n < M)\,\{\; n := n-1\;[1/3]\; n := n+1\,\}$$

$$\text{wp}[\![C^{M}_{\text{brw}}]\!]\,([n \leq 0]\cdot 1) \;\preceq\; \text{wp}[\![C_{\text{brw}}]\!]\,(1)$$



A New Proof Rule for Lower Bounds

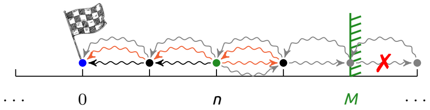
Theorem (Guard-Strengthening Rule)

$$C_{\text{loop}}\colon \quad \texttt{while}\,(\,\varphi\,)\,\{\,C\,\} \quad \rightsquigarrow \quad C'_{\text{loop}}\colon \quad \texttt{while}\,(\,\varphi'\,)\,\{\,C\,\}$$

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathrm{wp}[\![C'_{\text{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \mathrm{wp}[\![C_{\text{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

$$C_{\text{brw}}\colon \quad \texttt{while}\,(0 < n)\,\{\; n := n-1\;[1/3]\; n := n+1\,\} \quad \rightsquigarrow \quad C^{M}_{\text{brw}}\colon \quad \texttt{while}\,(0 < n < M)\,\{\; n := n-1\;[1/3]\; n := n+1\,\}$$

$$\mathrm{wp}[\![C^{M}_{\text{brw}}]\!]\,([n \le 0]\cdot 1) \;\preceq\; \mathrm{wp}[\![C_{\text{brw}}]\!]\,(1)$$



A New Proof Rule for Lower Bounds

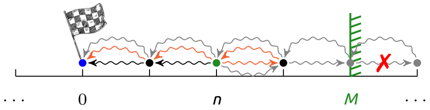
Theorem (Guard-Strengthening Rule)

$$C_{\text{loop}}: \quad \texttt{while}\,(\,\varphi\,)\,\{\,C\,\} \quad \rightsquigarrow \quad C'_{\text{loop}}: \quad \texttt{while}\,(\,\varphi'\,)\,\{\,C\,\}$$

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathrm{wp}[\![C'_{\text{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \mathrm{wp}[\![C_{\text{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

$$C_{\text{brw}}: \quad \texttt{while}\,(0 < n)\,\{\; n := n-1\ [1/3]\ n := n+1\,\} \quad \rightsquigarrow \quad C^{M}_{\text{brw}}: \quad \texttt{while}\,(0 < n < M)\,\{\; n := n-1\ [1/3]\ n := n+1\,\}$$

$$\mathrm{wp}[\![C^{M}_{\text{brw}}]\!]\,([n \leq 0]\cdot 1) \;\preceq\; \mathrm{wp}[\![C_{\text{brw}}]\!]\,(1)$$



- C'_{loop} features a **stronger** termination property (e.g., becoming AST).
- **Easier** to verify the uni. int. of l and the boundedness of expectations.

Behind the Proof Rule

Theorem (wp-Difference)

$$
\begin{aligned}
\mathsf{wp}[\![C_{\mathsf{loop}}]\!]\,(f) - \mathsf{wp}[\![C'_{\mathsf{loop}}]\!]\,(f) \;=\;& \\
&\mathsf{wp}[\![\mathtt{while}\,(\,\varphi \wedge \varphi'\,)\,\{\,C\,\}]\!]\,([\neg\varphi \wedge \varphi'] \cdot f) + \lambda\sigma\mathbf{.}\int_A f_{C_{\mathsf{loop}}}\,\mathrm{d}\,({}^{\sigma}\mathbb{P}) - \\
&\mathsf{wp}[\![\mathtt{while}\,(\,\varphi \wedge \varphi'\,)\,\{\,C\,\}]\!]\,([\varphi \wedge \neg\varphi'] \cdot f) - \lambda\sigma\mathbf{.}\int_B f_{C'_{\mathsf{loop}}}\,\mathrm{d}\,({}^{\sigma}\mathbb{P})
\end{aligned}
$$

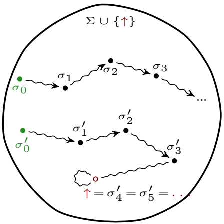


Figure – Infinite prog. traces.

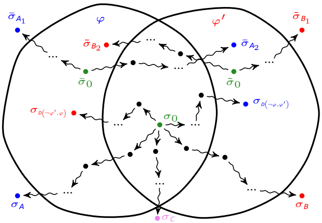


Figure – Illustration of wp-Difference.

- Potentially applicable to *sensitivity analysis* and *model repair*.

Properties of the Proof Rule

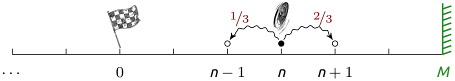
$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathrm{wp}[\![C'_{\mathrm{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \mathrm{wp}[\![C_{\mathrm{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

- (Trivially) **complete :** where there's an l, there's a φ' (albeit not "good" enough).

Properties of the Proof Rule

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathrm{wp}[\![C'_{\mathrm{loop}}]\!]\left([\neg\varphi]\cdot f\right)}{l \preceq \mathrm{wp}[\![C_{\mathrm{loop}}]\!]\left(f\right)} \quad \text{(Guard-Strengthening)}$$

- (Trivially) **complete :** where there's an l, there's a φ' (albeit not "good" enough).
- **General :** applicable to *possibly divergent* C_{loop} and unbounded expectations f, l:



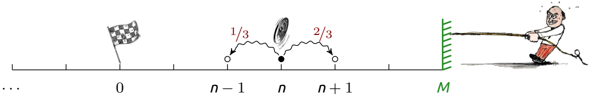
$$l_M \;=\; [n<0] + [0\leq n\leq M]\cdot\left(\left(1/2\right)^n - \left(1/2\right)^M\right)$$

$$\forall M\in\mathbb{N}:\quad l_M \overset{\text{Hark}}{\preceq} \mathrm{wp}[\![C^M_{\mathrm{brw}}]\!]\left([n\leq 0]\cdot 1\right) \overset{\text{Stren.}}{\preceq} \mathrm{wp}[\![C_{\mathrm{brw}}]\!]\left(1\right)$$

Properties of the Proof Rule

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathsf{wp}[\![C'_{\mathsf{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \mathsf{wp}[\![C_{\mathsf{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

- (Trivially) **complete :** where there's an l, there's a φ' (albeit not "good" enough).
- **General :** applicable to *possibly divergent* C_{loop} and unbounded expectations f, l:



$$l_M \;=\; [n < 0] + [0 \leq n \leq M] \cdot \left((1/2)^n - (1/2)^M \right)$$

$$\forall M \in \mathbb{N}\colon \quad l_M \overset{\text{Hark}}{\preceq} \mathsf{wp}[\![C^M_{\mathsf{brw}}]\!]\,([n \leq 0]\cdot 1) \overset{\text{Stren.}}{\preceq} \mathsf{wp}[\![C_{\mathsf{brw}}]\!]\,(1)$$

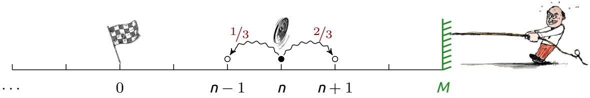
- **Tight :** the underapproximation error approaches 0 as $\varphi' \to \varphi$:

$$[n < 0] + [n \geq 0] \cdot (1/2)^n \;=\; \lim_{M\to\infty} l_M \;\preceq\; \mathsf{wp}[\![C_{\mathsf{brw}}]\!]\,(1)$$

Properties of the Proof Rule

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathrm{wp}[\![C'_{\mathrm{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \mathrm{wp}[\![C_{\mathrm{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

- (Trivially) **complete :** where there's an l, there's a φ' (albeit not "good" enough).
- **General :** applicable to *possibly divergent* C_{loop} and unbounded expectations f, l:



$$l_M \;=\; [n<0] + [0 \leq n \leq M]\cdot\Big((1/2)^n - (1/2)^M\Big)$$

$$\forall M\in\mathbb{N}\colon\quad l_M \overset{\text{Hark}}{\preceq} \mathrm{wp}[\![C^M_{\mathrm{brw}}]\!]\,([n\leq 0]\cdot 1) \overset{\text{Stren.}}{\preceq} \mathrm{wp}[\![C_{\mathrm{brw}}]\!]\,(1)$$

- **Tight :** the underapproximation error approaches 0 as $\varphi' \to \varphi$:

$$[n<0] + [n\geq 0]\cdot(1/2)^n \;=\; \lim_{M\to\infty} l_M \overset{\text{Park}}{=} \mathrm{wp}[\![C_{\mathrm{brw}}]\!]\,(1)$$

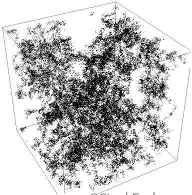
Properties of the Proof Rule

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathrm{wp}[\![C'_{\mathrm{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \mathrm{wp}[\![C_{\mathrm{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

- **Automatable :** reducible to *probabilistic model checking* for finite-state C'_{loop} :

$\texttt{while}\,(\,x \neq 0 \vee y \neq 0 \vee z \neq 0\,)\,\{$

$\quad x \coloneqq x-1 \,\oplus\, x \coloneqq x+1 \,\oplus\, y \coloneqq y-1 \,\oplus\, y \coloneqq y+1 \,\oplus\, z \coloneqq z-1 \,\oplus\, z \coloneqq z+1\,\}$



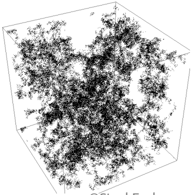
©StackExchange

Properties of the Proof Rule

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathsf{wp}[\![C'_{\mathsf{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \mathsf{wp}[\![C_{\mathsf{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

- **Automatable :** reducible to *probabilistic model checking* for finite-state C'_{loop} :

$$\texttt{while}\,(\,x \neq 0 \vee y \neq 0 \vee z \neq 0\,)\,\{$$
$$x \coloneqq x-1 \oplus x \coloneqq x+1 \oplus y \coloneqq y-1 \oplus y \coloneqq y+1 \oplus z \coloneqq z-1 \oplus z \coloneqq z+1\,\}$$



©StackExchange

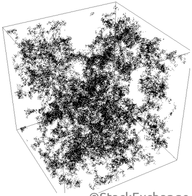
$$\mathcal{P} = 1 - \left(\frac{3}{(2\pi)^3}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\frac{\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z}{3-\cos x-\cos y-\cos z}\right)^{-1} = 0.3405373296\ldots$$

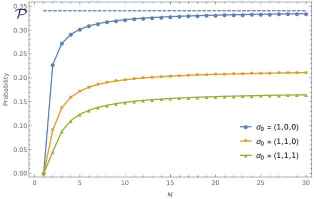
Properties of the Proof Rule

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathrm{wp}[\![C'_{\mathrm{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \mathrm{wp}[\![C_{\mathrm{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

- **Automatable :** reducible to *probabilistic model checking* for finite-state C'_{loop} :

$$\texttt{while}\,(\,x \neq 0 \vee y \neq 0 \vee z \neq 0\,)\,\{$$
$$x \coloneqq x-1 \oplus x \coloneqq x+1 \oplus y \coloneqq y-1 \oplus y \coloneqq y+1 \oplus z \coloneqq z-1 \oplus z \coloneqq z+1\,\}$$





$$\mathcal{P} = 1 - \left(\frac{3}{(2\pi)^3}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}\frac{\mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z}{3-\cos x-\cos y-\cos z}\right)^{-1} = 0.3405373296\ldots$$

A “Real” Application : Zeroconf Protocol [Bohnenkamp et al. 2003]

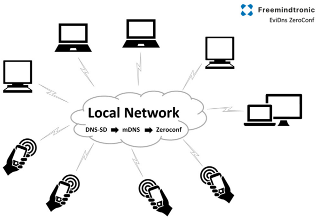
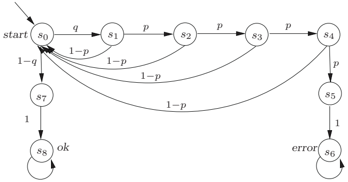


Figure – Self-configuring IP network.



©[Baier & Katoen 2008]

Figure – Markov-chain snippet ($N = 4$).

C_{zc} :

$$
\begin{array}{l}
\mathit{start} = 1 \,⨟\, \mathit{established} = 0 \,⨟\, \mathit{probe} = 0 \,⨟ \\
\texttt{while}\ (\, \mathit{start} \leq 1 \wedge \mathit{established} \leq 0 \wedge \mathit{probe} < N \wedge N \geq 4 \,)\ \{ \\
\quad \texttt{if}\ (\, \mathit{start} = 1 \,)\ \{ \\
\qquad \{\, \mathit{start} := 0 \,\}\ [0.5]\ \{\, \mathit{start} := 0 \,⨟\, \mathit{established} := 1 \,\}\ \} \\
\quad \texttt{else}\ \{\ \{\, \mathit{probe} := \mathit{probe} + 1 \,\}\ [0.001]\ \{\, \mathit{start} := 1 \,⨟\, \mathit{probe} := 0 \,\}\ \}\ \}
\end{array}
$$

A "Real" Application : Zeroconf Protocol [Bohnenkamp et al. 2003]

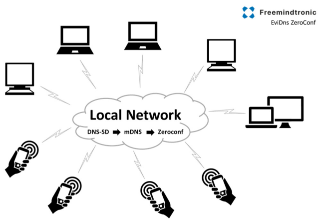
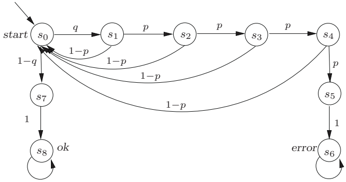


Figure – Self-configuring IP network.



©[Baier & Katoen 2008]

Figure – Markov-chain snippet ($N = 4$).

C_{zc} : $\quad$ *start* $= 1 \fatsemi$ *established* $= 0 \fatsemi$ *probe* $= 0 \fatsemi$

while (*start* $\leq 1 \wedge$ *established* $\leq 0 \wedge$ *probe* $< N \wedge N \geq 4$) {

if (*start* $= 1$) {

{ *start* $:= 0$ } $[0.5]$ { *start* $:= 0 \fatsemi$ *established* $:= 1$ } }

else { { *probe* $:=$ *probe* $+ 1$ } $[0.001]$ { *start* $:= 1 \fatsemi$ *probe* $:= 0$ } } }

Pr("starting within the loop guard, C_{zc} terminates with *established* $= 1$") ≥ 0.99999999999

A “Real” Application : Zeroconf Protocol [Bohnenkamp et al. 2003]

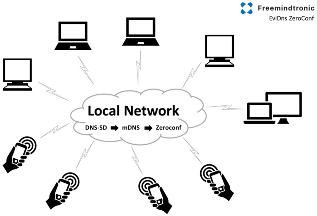


Figure – Self-configuring IP network.

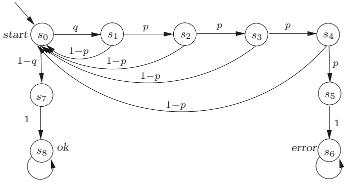


Figure – Markov-chain snippet ($N = 4$).

C_{zc} :
$$\begin{array}{l} \mathit{start} = 1 ⨾ \mathit{established} = 0 ⨾ \mathit{probe} = 0 ⨾ \\ \texttt{while}\ (\ \mathit{start} \leq 1 \wedge \mathit{established} \leq 0 \wedge \mathit{probe} < N \wedge N \geq 4\ )\ \{ \\ \quad \texttt{if}\ (\ \mathit{start} = 1\ )\ \{ \\ \quad\quad \{\ \mathit{start} := 0\ \}\ [0.5]\ \{\ \mathit{start} := 0 ⨾ \mathit{established} := 1\ \}\ \} \\ \quad \texttt{else}\ \{\ \{\ \mathit{probe} := \mathit{probe} + 1\ \}\ [0.001]\ \{\ \mathit{start} := 1 ⨾ \mathit{probe} := 0\ \}\ \}\ \} \end{array}$$

Pr(“starting within the loop guard, C_{zc} terminates with *established* $= 1$”) $\overset{?}{\geq} 0.99999999999$

A "Real" Application : Zeroconf Protocol [Bohnenkamp et al. 2003]

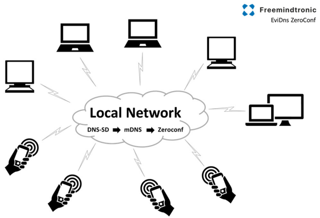
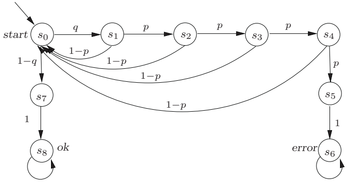


Figure – Self-configuring IP network.



©[Baier & Katoen 2008]

Figure – Markov-chain snippet ($N = 4$).

C_{zc} : $\quad$ *start* $= 1 \,⨟\,$ *established* $= 0 \,⨟\,$ *probe* $= 0 \,⨟$

while (*start* $\leq 1 \wedge$ *established* $\leq 0 \wedge$ *probe* $< N \wedge N \geq 4 \wedge N \leq 10$) {

if (*start* $= 1$) {

{ *start* $:= 0$ } $[0.5]$ { *start* $:= 0 \,⨟\,$ *established* $:= 1$ } }

else { { *probe* $:=$ *probe* $+ 1$ } $[0.001]$ { *start* $:= 1 \,⨟\,$ *probe* $:= 0$ } } }

Pr("starting within the loop guard, C_{zc} terminates with *established* $= 1$") $\overset{\checkmark}{\geq} 0.99999999999$

Summary

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathrm{wp}[\![C'_{\mathrm{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \mathrm{wp}[\![C_{\mathrm{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

- a new lower bound rule based on wp-difference and guard-strengthening;

⇒ Feng, Chen, Su, Kaminski, Katoen, Zhan : *Lower Bounds for Poss. Divergent Prob. Prog.* OOPSLA '23.

Summary

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathrm{wp}[\![C'_{\mathrm{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \mathrm{wp}[\![C_{\mathrm{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

- a new lower bound rule based on wp-difference and guard-strengthening;
- first lower bound rule admitting divergent loops with unbounded expectations;

⇒ Feng, Chen, Su, Kaminski, Katoen, Zhan : *Lower Bounds for Poss. Divergent Prob. Prog.* OOPSLA '23.

Summary

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathrm{wp}[\![C'_{\mathrm{loop}}]\!]\,([\neg\varphi] \cdot f)}{l \preceq \mathrm{wp}[\![C_{\mathrm{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

- a new lower bound rule based on wp-difference and guard-strengthening;
- first lower bound rule admitting divergent loops with unbounded expectations;
- tight lower bounds for 3-D random walks on $\mathbb{Z}^3$ and the Zeroconf protocol.

⇒ Feng, Chen, Su, Kaminski, Katoen, Zhan : *Lower Bounds for Poss. Divergent Prob. Prog.* OOPSLA '23.

Summary

$$\frac{\varphi' \implies \varphi \qquad l \preceq \mathsf{wp}[\![C'_{\mathrm{loop}}]\!]\,([\neg\varphi]\cdot f)}{l \preceq \mathsf{wp}[\![C_{\mathrm{loop}}]\!]\,(f)} \quad \text{(Guard-Strengthening)}$$

- a new lower bound rule based on wp-difference and guard-strengthening;
- first lower bound rule admitting divergent loops with unbounded expectations;
- tight lower bounds for 3-D random walks on $\mathbb{Z}^3$ and the Zeroconf protocol.

More in the paper :

- how to find a "good" strengthening $\varphi' \implies \varphi$?
- how to generate a non-trivial lower bound for C'_{loop}?
- corner cases where guard strengthening is insufficient;
- ...

⇒ Feng, Chen, Su, Kaminski, Katoen, Zhan : *Lower Bounds for Poss. Divergent Prob. Prog.* OOPSLA '23.