Inferring Expected Runtimes of Probabilistic Programs

Jürgen Giesl

LuFG Informatik 2, RWTH Aachen University, Germany

joint work with Marcel Hark and Fabian Meyer

Complexity Analysis for Integer Programs

- Termination analysis of imperative programs: ranking functions

Complexity Analysis for Integer Programs

- Termination analysis of imperative programs: ranking functions
- **Goal**: use ranking functions for complexity analysis

Complexity Analysis for Integer Programs

- Termination analysis of imperative programs: ranking functions
- **Goal**: use ranking functions for complexity analysis
- **Problem**: complexity from combination of ranking functions

Complexity Analysis for Integer Programs

- Termination analysis of imperative programs: ranking functions
- **Goal**: use ranking functions for complexity analysis
- **Problem**: complexity from combination of ranking functions

while $\mathtt{i} > 0$ **do**

  $\mathtt{i} = \mathtt{i} - 1$

done

while $\mathtt{x} > 0$ **do**

  $\mathtt{x} = \mathtt{x} - 1$

done

Complexity Analysis for Integer Programs

- Termination analysis of imperative programs: ranking functions
- **Goal**: use ranking functions for complexity analysis
- **Problem**: complexity from combination of ranking functions

Termination: lexicographic combination of

$$f_1(\mathtt{x}, \mathtt{i}) = \mathtt{i}$$
$$f_2(\mathtt{x}, \mathtt{i}) = \mathtt{x}$$

while $\mathtt{i} > 0$ **do**

  $\mathtt{i} = \mathtt{i} - 1$

done

while $\mathtt{x} > 0$ **do**

  $\mathtt{x} = \mathtt{x} - 1$

done

Complexity Analysis for Integer Programs

- Termination analysis of imperative programs: ranking functions
- **Goal**: use ranking functions for complexity analysis
- **Problem**: complexity from combination of ranking functions

Termination: lexicographic combination of

$$f_1(\mathtt{x}, \mathtt{i}) = \mathtt{i}$$
$$f_2(\mathtt{x}, \mathtt{i}) = \mathtt{x}$$

Complexity: linear

while $\mathtt{i} > 0$ **do**

$\quad \mathtt{i} = \mathtt{i} - 1$

done

while $\mathtt{x} > 0$ **do**

$\quad \mathtt{x} = \mathtt{x} - 1$

done

Complexity Analysis for Integer Programs

- Termination analysis of imperative programs: ranking functions
- **Goal**: use ranking functions for complexity analysis
- **Problem**: complexity from combination of ranking functions

Termination: lexicographic combination of

$$f_1(\mathtt{x}, \mathtt{i}) = \mathtt{i}$$
$$f_2(\mathtt{x}, \mathtt{i}) = \mathtt{x}$$

Complexity: linear

while $\mathtt{i} > 0$ **do**
  $\mathtt{x} = \mathtt{x} + \mathtt{i}$
  $\mathtt{i} = \mathtt{i} - 1$
done

while $\mathtt{x} > 0$ **do**
  $\mathtt{x} = \mathtt{x} - 1$
done

Complexity Analysis for Integer Programs

- Termination analysis of imperative programs: ranking functions
- **Goal**: use ranking functions for complexity analysis
- **Problem**: complexity from combination of ranking functions

Termination: lexicographic combination of

$$f_1(\mathtt{x}, \mathtt{i}) = \mathtt{i}$$
$$f_2(\mathtt{x}, \mathtt{i}) = \mathtt{x}$$

Complexity: quadratic

while $\mathtt{i} > 0$ **do**
  $\mathtt{x} = \mathtt{x} + \mathtt{i}$
  $\mathtt{i} = \mathtt{i} - 1$
done

while $\mathtt{x} > 0$ **do**
  $\mathtt{x} = \mathtt{x} - 1$
done

Complexity Analysis for Integer Programs

- Termination analysis of imperative programs: ranking functions
- **Goal**: use ranking functions for complexity analysis
- **Problem**: complexity from combination of ranking functions

Termination: lexicographic combination of

$$f_1(\mathtt{x}, \mathtt{i}) = \mathtt{i}$$
$$f_2(\mathtt{x}, \mathtt{i}) = \mathtt{x}$$

Complexity: quadratic

$$\mathtt{i}_0 + \text{“size”}(\mathtt{x})$$

while $\mathtt{i} > 0$ **do**
  $\mathtt{x} = \mathtt{x} + \mathtt{i}$
  $\mathtt{i} = \mathtt{i} - 1$
done

while $\mathtt{x} > 0$ **do**
  $\mathtt{x} = \mathtt{x} - 1$
done

Complexity Analysis for Integer Programs

- Termination analysis of imperative programs: ranking functions
- **Goal**: use ranking functions for complexity analysis
- **Problem**: complexity from combination of ranking functions

Termination: lexicographic combination of

$$f_1(\mathtt{x}, \mathtt{i}) = \mathtt{i}$$
$$f_2(\mathtt{x}, \mathtt{i}) = \mathtt{x}$$

Complexity: quadratic

$$\mathtt{i}_0 + \text{“size”}(\mathtt{x})$$

while $\mathtt{i} > 0$ **do**
$\quad \mathtt{x} = \mathtt{x} + \mathtt{i}$
$\quad \mathtt{i} = \mathtt{i} - 1$
done

while $\mathtt{x} > 0$ **do**
$\quad \mathtt{x} = \mathtt{x} - 1$
done

- **Solution**: modular approach which alternates between finding runtime and size bounds

Complexity Analysis for Probabilistic Programs

```
while i > 0 do
    x = x + i
    i = i - 1
done

while x > 0 do
  x = x - 1
done
```

Complexity Analysis for Probabilistic Programs

while $\mathtt{i} > 0$ **do**

$$\left\{ \begin{array}{l} \mathtt{x} = \mathtt{x} + \mathtt{i} \\ \mathtt{i} = \mathtt{i} - 1 \end{array} \right\} \; [\tfrac{1}{2}] \; \left\{ \begin{array}{l} \mathtt{x} = \mathtt{x} \\ \mathtt{i} = \mathtt{i} \end{array} \right\}$$

done

while $\mathtt{x} > 0$ **do**

$\mathtt{x} = \mathtt{x} - 1$

done

Complexity Analysis for Probabilistic Programs

- Probabilistic ranking functions

$$
\begin{array}{l}
\textbf{while } \mathtt{i} > 0 \textbf{ do} \\
\quad \left\{ \begin{array}{l} \mathtt{x} = \mathtt{x} + \mathtt{i} \\ \mathtt{i} = \mathtt{i} - 1 \end{array} \right\} \left[\tfrac{1}{2}\right] \left\{ \begin{array}{l} \mathtt{x} = \mathtt{x} \\ \mathtt{i} = \mathtt{i} \end{array} \right\} \\
\textbf{done}
\end{array}
$$

$$
\begin{array}{l}
\textbf{while } \mathtt{x} > 0 \textbf{ do} \\
\quad \mathtt{x} = \mathtt{x} - 1 \\
\textbf{done}
\end{array}
$$

Complexity Analysis for Probabilistic Programs

- Probabilistic ranking functions
- Expected value of ranking function must decrease by at least 1

while $\mathtt{i} > 0$ **do**

$$\left\{ \begin{array}{l} \mathtt{x} = \mathtt{x} + \mathtt{i} \\ \mathtt{i} = \mathtt{i} - 1 \end{array} \right\} \left[\tfrac{1}{2}\right] \left\{ \begin{array}{l} \mathtt{x} = \mathtt{x} \\ \mathtt{i} = \mathtt{i} \end{array} \right\}$$

done

while $\mathtt{x} > 0$ **do**

$\quad \mathtt{x} = \mathtt{x} - 1$

done

Complexity Analysis for Probabilistic Programs

- Probabilistic ranking functions
- Expected value of ranking function must decrease by at least 1

Probabilistic ranking functions for each loop

$$f_1(\mathtt{x}, \mathtt{i}) = 2 \cdot \mathtt{i}$$
$$f_2(\mathtt{x}, \mathtt{i}) = \mathtt{x}$$

while $\mathtt{i} > 0$ **do**

$$\left\{ \begin{array}{l} \mathtt{x} = \mathtt{x} + \mathtt{i} \\ \mathtt{i} = \mathtt{i} - 1 \end{array} \right\} \left[\tfrac{1}{2}\right] \left\{ \begin{array}{l} \mathtt{x} = \mathtt{x} \\ \mathtt{i} = \mathtt{i} \end{array} \right\}$$

done

while $\mathtt{x} > 0$ **do**

$\quad \mathtt{x} = \mathtt{x} - 1$

done

Complexity Analysis for Probabilistic Programs

- Probabilistic ranking functions
- Expected value of ranking function must decrease by at least 1

$$f_1(\mathtt{x},\mathtt{i}) \;\geq\; \tfrac{1}{2}\cdot f_1(\mathtt{x}+\mathtt{i},\mathtt{i}-1)+\tfrac{1}{2}\cdot f_1(\mathtt{x},\mathtt{i})+1$$

Probabilistic ranking functions for each loop

$$f_1(\mathtt{x},\mathtt{i}) = 2\cdot\mathtt{i}$$
$$f_2(\mathtt{x},\mathtt{i}) = \mathtt{x}$$

while $\mathtt{i} > 0$ **do**

$$\left\{\begin{array}{l}\mathtt{x}=\mathtt{x}+\mathtt{i}\\ \mathtt{i}=\mathtt{i}-1\end{array}\right\}\;\left[\tfrac{1}{2}\right]\;\left\{\begin{array}{l}\mathtt{x}=\mathtt{x}\\ \mathtt{i}=\mathtt{i}\end{array}\right\}$$

done

while $\mathtt{x} > 0$ **do**

$\mathtt{x} = \mathtt{x} - 1$

done

Complexity Analysis for Probabilistic Programs

- Probabilistic ranking functions
- Expected value of ranking function must decrease by at least 1

$$f_1(\mathtt{x},\mathtt{i}) \ \geq \ \tfrac{1}{2} \cdot f_1(\mathtt{x}+\mathtt{i},\mathtt{i}-1) + \tfrac{1}{2} \cdot f_1(\mathtt{x},\mathtt{i}) + 1$$

Probabilistic ranking functions for each loop

$$f_1(\mathtt{x},\mathtt{i}) = 2 \cdot \mathtt{i}$$
$$f_2(\mathtt{x},\mathtt{i}) = \mathtt{x}$$

Expected runtime: quadratic

while $\mathtt{i} > 0$ **do**

$$\left\{ \begin{array}{l} \mathtt{x} = \mathtt{x}+\mathtt{i} \\ \mathtt{i} = \mathtt{i}-1 \end{array} \right\} \ [\tfrac{1}{2}] \ \left\{ \begin{array}{l} \mathtt{x} = \mathtt{x} \\ \mathtt{i} = \mathtt{i} \end{array} \right\}$$

done

while $\mathtt{x} > 0$ **do**

$\mathtt{x} = \mathtt{x} - 1$

done

Complexity Analysis for Probabilistic Programs

- Probabilistic ranking functions
- Expected value of ranking function must decrease by at least 1

$$f_1(\mathtt{x},\mathtt{i}) \;\geq\; \tfrac{1}{2}\cdot f_1(\mathtt{x}+\mathtt{i},\mathtt{i}-1)+\tfrac{1}{2}\cdot f_1(\mathtt{x},\mathtt{i})+1$$

Probabilistic ranking functions for each loop

$$f_1(\mathtt{x},\mathtt{i}) = 2\cdot\mathtt{i}$$
$$f_2(\mathtt{x},\mathtt{i}) = \mathtt{x}$$

Expected runtime: quadratic

$$2\cdot\mathtt{i}_0 + \text{``expected size''}(\mathtt{x})$$

while $\mathtt{i} > 0$ **do**

$$\left\{\begin{array}{l}\mathtt{x}=\mathtt{x}+\mathtt{i}\\ \mathtt{i}=\mathtt{i}-1\end{array}\right\}\;[\tfrac{1}{2}]\;\left\{\begin{array}{l}\mathtt{x}=\mathtt{x}\\ \mathtt{i}=\mathtt{i}\end{array}\right\}$$

done

while $\mathtt{x} > 0$ **do**

$\mathtt{x} = \mathtt{x} - 1$

done

Complexity Analysis for Probabilistic Programs

- Probabilistic ranking functions
- Expected value of ranking function must decrease by at least 1

$$f_1(\mathtt{x},\mathtt{i}) \;\geq\; \tfrac{1}{2}\cdot f_1(\mathtt{x}+\mathtt{i},\mathtt{i}-1)+\tfrac{1}{2}\cdot f_1(\mathtt{x},\mathtt{i})+1$$

Probabilistic ranking functions for each loop

$$f_1(\mathtt{x},\mathtt{i}) = 2\cdot \mathtt{i}$$
$$f_2(\mathtt{x},\mathtt{i}) = \mathtt{x}$$

Expected runtime: quadratic

$$2\cdot \mathtt{i}_0 + \text{“expected size”}(\mathtt{x})$$

while $\mathtt{i} > 0$ **do**

$$\left\{\begin{array}{l}\mathtt{x}=\mathtt{x}+\mathtt{i}\\ \mathtt{i}=\mathtt{i}-1\end{array}\right\}\ [\tfrac{1}{2}]\ \left\{\begin{array}{l}\mathtt{x}=\mathtt{x}\\ \mathtt{i}=\mathtt{i}\end{array}\right\}$$

done

while $\mathtt{x} > 0$ **do**

$\mathtt{x} = \mathtt{x} - 1$

done

- **Solution**: modular approach which alternates between finding runtime and size bounds

Runtime and Size Bounds

while $\mathtt{i} > 0$ **do**

$$\left\{ \begin{array}{l} \mathtt{x} = \mathtt{x} + \mathtt{i} \\ \mathtt{i} = \mathtt{i} - 1 \end{array} \right\} \left[\tfrac{1}{2}\right] \left\{ \begin{array}{l} \mathtt{x} = \mathtt{x} \\ \mathtt{i} = \mathtt{i} \end{array} \right\}$$

done

while $\mathtt{x} > 0$ **do**

$\mathtt{x} = \mathtt{x} - 1$

done

Runtime and Size Bounds

while $\mathtt{i} > 0$ **do**

$$\left\{ \begin{array}{l} \mathtt{x} = \mathtt{x} + \mathtt{i} \\ \mathtt{i} = \mathtt{i} - 1 \end{array} \right\} \left[\tfrac{1}{2}\right] \left\{ \begin{array}{l} \mathtt{x} = \mathtt{x} \\ \mathtt{i} = \mathtt{i} \end{array} \right\}$$

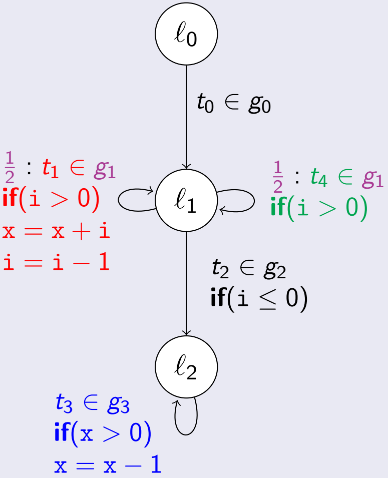
done

while $\mathtt{x} > 0$ **do**

$\quad \mathtt{x} = \mathtt{x} - 1$

done

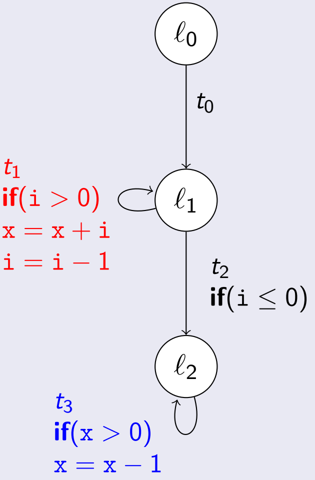
$$\Longrightarrow$$



Runtime and Size Bounds

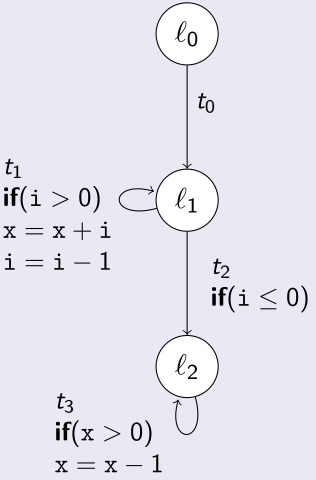
while $\mathtt{i} > 0$ **do**
$\quad \mathtt{x} = \mathtt{x} + \mathtt{i}$
$\quad \mathtt{i} = \mathtt{i} - 1$
done

while $\mathtt{x} > 0$ **do**
$\quad \mathtt{x} = \mathtt{x} - 1$
done



Runtime and Size Bounds

Goal: find complexity bounds w.r.t.
the *sizes* (absolute values) of the input variables

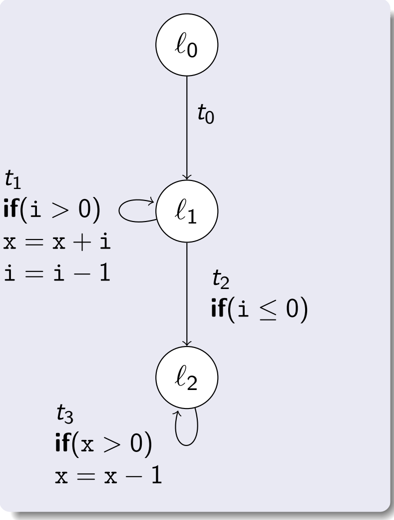


Runtime and Size Bounds

Goal: find complexity bounds w.r.t.
the *sizes* (absolute values) of the input variables

- Runtime bound $\mathcal{R}(t)$:

  bound on number of times
  that transition t occurs in executions



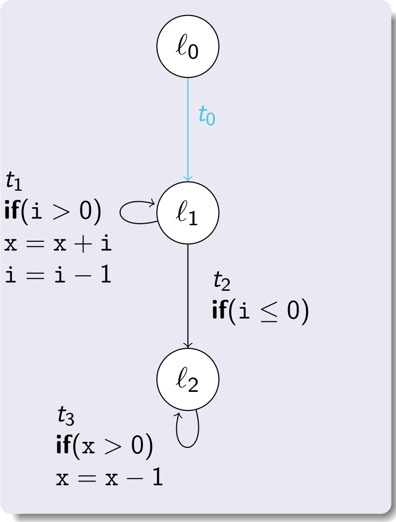
Runtime and Size Bounds

Goal: find complexity bounds w.r.t.
the *sizes* (absolute values) of the input variables

- Runtime bound $\mathcal{R}(t)$:

  bound on number of times
  that transition t occurs in executions

  $\mathcal{R}(t_0) = 1$



Runtime and Size Bounds

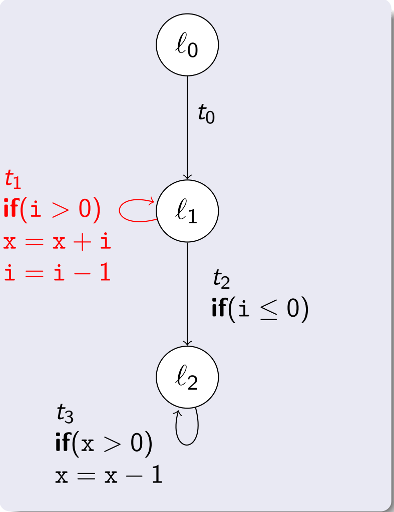
Goal: find complexity bounds w.r.t.
the *sizes* (absolute values) of the input variables

- Runtime bound $\mathcal{R}(t)$:

  bound on number of times
  that transition t occurs in executions

  $\mathcal{R}(t_0) = 1$

  $\mathcal{R}(t_1) = \mathtt{i}_0$



Runtime and Size Bounds

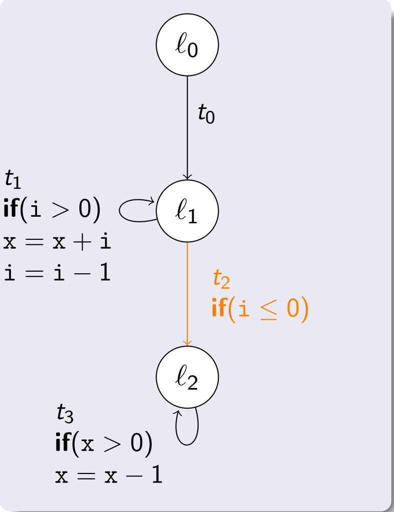
Goal: find complexity bounds w.r.t.
the *sizes* (absolute values) of the input variables

- Runtime bound $\mathcal{R}(t)$:

  bound on number of times
  that transition t occurs in executions

  $\mathcal{R}(t_0) = 1$ $\qquad$ $\mathcal{R}(t_2) = 1$

  $\mathcal{R}(t_1) = \mathtt{i}_0$



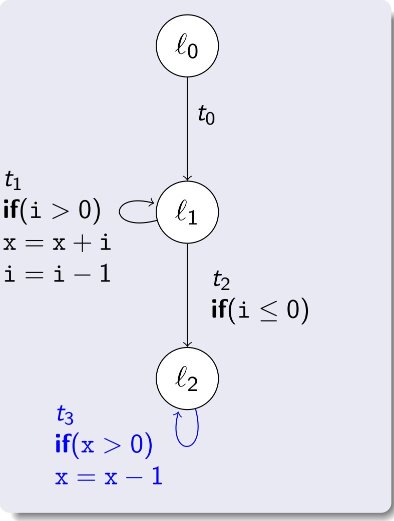
Runtime and Size Bounds

Goal: find complexity bounds w.r.t.
the *sizes* (absolute values) of the input variables

- Runtime bound $\mathcal{R}(t)$:

  bound on number of times
  that transition t occurs in executions

$$\mathcal{R}(t_0) = 1 \qquad \mathcal{R}(t_2) = 1$$
$$\mathcal{R}(t_1) = \mathtt{i}_0 \qquad \mathcal{R}(t_3) = \mathtt{x}_0 + \mathtt{i}_0^2$$



Runtime and Size Bounds

Goal: find complexity bounds w.r.t.
the *sizes* (absolute values) of the input variables

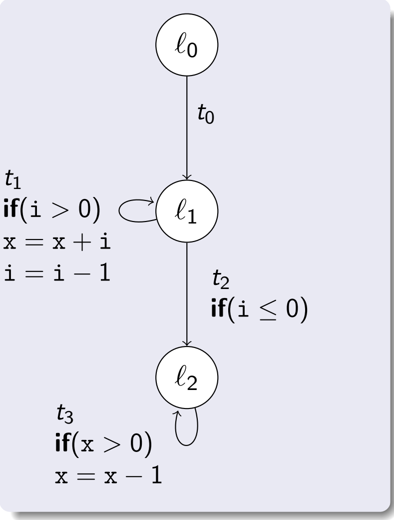
- Runtime bound $\mathcal{R}(t)$:

  bound on number of times
  that transition t occurs in executions

  $$\mathcal{R}(t_0) = 1 \qquad \mathcal{R}(t_2) = 1$$
  $$\mathcal{R}(t_1) = \mathtt{i}_0 \qquad \mathcal{R}(t_3) = \mathtt{x}_0 + \mathtt{i}_0^2$$

- Size bound $\mathcal{S}(t, v)$:

  bound on size of v
  after using transition t in program executions



Runtime and Size Bounds

Goal: find complexity bounds w.r.t.
the *sizes* (absolute values) of the input variables

- Runtime bound $\mathcal{R}(t)$:

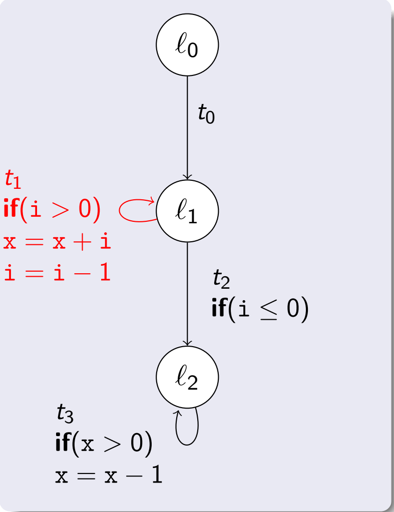
  bound on number of times
  that transition t occurs in executions

  $$\mathcal{R}(t_0) = 1 \qquad \mathcal{R}(t_2) = 1$$
  $$\mathcal{R}(t_1) = \mathtt{i}_0 \qquad \mathcal{R}(t_3) = \mathtt{x}_0 + \mathtt{i}_0^2$$

- Size bound $\mathcal{S}(t, v)$:

  bound on size of v
  after using transition t in program executions

  e.g., $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$



Runtime and Size Bounds

Goal: find complexity bounds w.r.t.
the *sizes* (absolute values) of the input variables

- Runtime bound $\mathcal{R}(t)$:

  bound on number of times
  that transition t occurs in executions

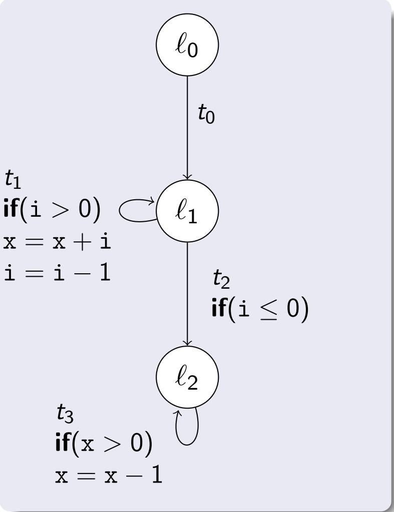
  $\mathcal{R}(t_0) = 1$ $\quad\quad$ $\mathcal{R}(t_2) = 1$

  $\mathcal{R}(t_1) = \mathtt{i}_0$ $\quad\quad$ $\mathcal{R}(t_3) = \mathtt{x}_0 + \mathtt{i}_0^2$

- Size bound $\mathcal{S}(t, v)$:

  bound on size of v
  after using transition t in program executions

  e.g., $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$

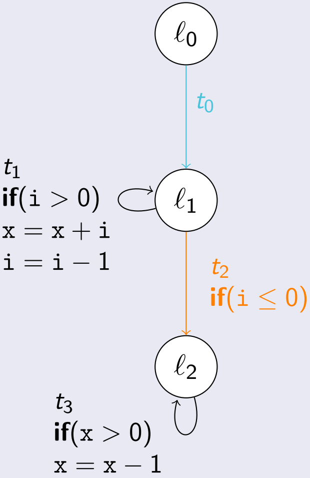


Overall runtime is bounded by $\mathcal{R}(t_0) + \ldots + \mathcal{R}(t_3) = 1 + \mathtt{i}_0 + 1 + \mathtt{x}_0 + \mathtt{i}_0^2$.

Runtime Bounds from Ranking Functions

Initial bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$ as t_0 and t_2 are not in loops

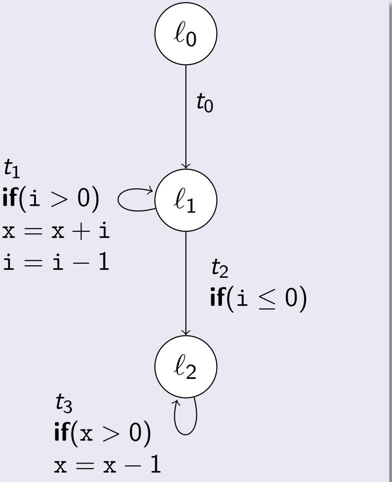


Runtime Bounds from Ranking Functions

Initial bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$ as t_0 and t_2 are not in loops

Ranking function $\mathfrak{r}$ for program $\mathcal{P}$



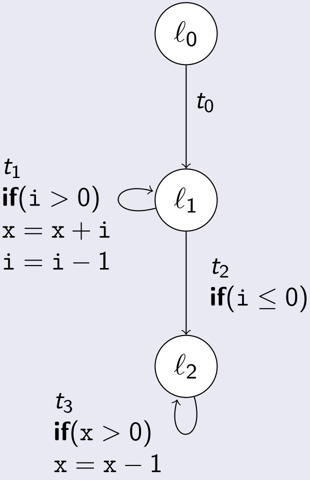
Runtime Bounds from Ranking Functions

Initial bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$ as t_0 and t_2 are not in loops

Ranking function $\mathfrak{r}$ for program $\mathcal{P}$

- $\mathfrak{r}$ maps *locations* to $\mathbb{R}[v_1, \ldots, v_n]$



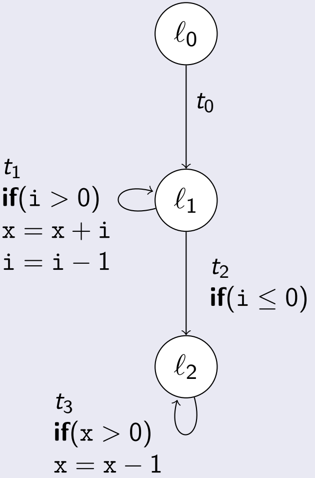
Runtime Bounds from Ranking Functions

Initial bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$ as t_0 and t_2 are not in loops

Ranking function $\mathfrak{r}$ for program $\mathcal{P}$

- $\mathfrak{r}$ maps *locations* to $\mathbb{R}[v_1, \ldots, v_n]$
- **Non-Increase:** no transition in $\mathcal{P}$ increases value of $\mathfrak{r}$



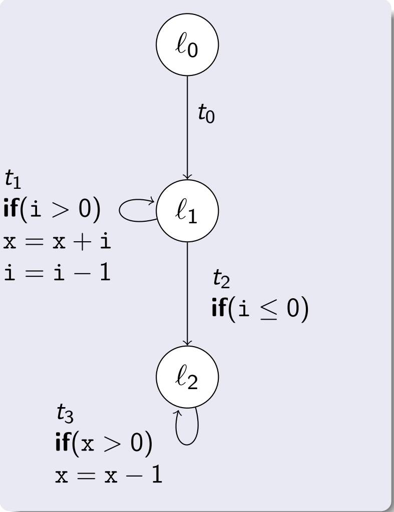
Runtime Bounds from Ranking Functions

Initial bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$ as t_0 and t_2 are not in loops

Ranking function $\mathfrak{r}$ for program $\mathcal{P}$

- $\mathfrak{r}$ maps *locations* to $\mathbb{R}[v_1, \ldots, v_n]$
- **Non-Increase:** no transition in $\mathcal{P}$ increases value of $\mathfrak{r}$
- **Decrease:** value of $\mathfrak{r}$ decreases by at least 1 for $\mathcal{P}_{\succ} \subseteq \mathcal{P}$



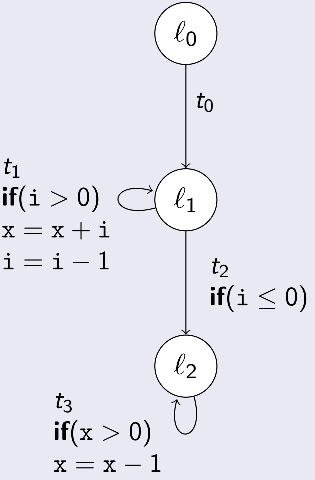
Runtime Bounds from Ranking Functions

Initial bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$ as t_0 and t_2 are not in loops

Ranking function $\mathfrak{r}$ for program $\mathcal{P}$

- $\mathfrak{r}$ maps *locations* to $\mathbb{R}[v_1, \ldots, v_n]$
- **Non-Increase:** no transition in $\mathcal{P}$ increases value of $\mathfrak{r}$
- **Decrease:** value of $\mathfrak{r}$ decreases by at least 1 for $\mathcal{P}_\succ \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_\succ \subseteq \mathcal{P}$



Runtime Bounds from Ranking Functions

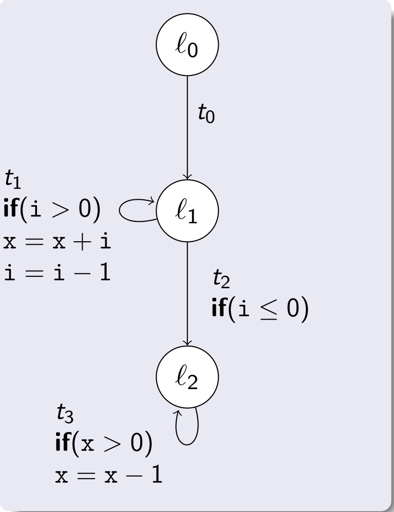
Initial bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$ as t_0 and t_2 are not in loops

Ranking function $\mathfrak{r}$ for program $\mathcal{P}$

- $\mathfrak{r}$ maps *locations* to $\mathbb{R}[v_1, \ldots, v_n]$
- **Non-Increase:** no transition in $\mathcal{P}$ increases value of $\mathfrak{r}$
- **Decrease:** value of $\mathfrak{r}$ decreases by at least 1 for $\mathcal{P}_\succ \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_\succ \subseteq \mathcal{P}$

- $\mathfrak{r}(\ell) = \mathtt{i}$ for all locations ℓ



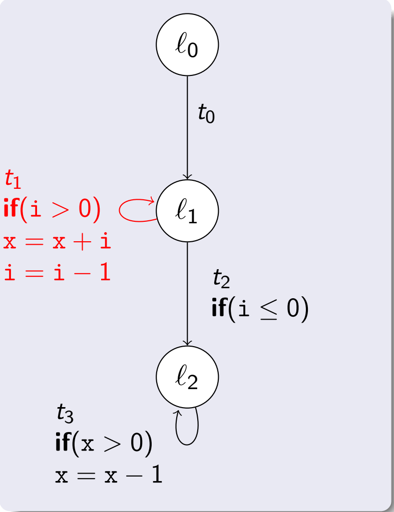
Runtime Bounds from Ranking Functions

Initial bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$ as t_0 and t_2 are not in loops

Ranking function $\mathfrak{r}$ for program $\mathcal{P}$

- $\mathfrak{r}$ maps *locations* to $\mathbb{R}[v_1, \ldots, v_n]$
- **Non-Increase:** no transition in $\mathcal{P}$ increases value of $\mathfrak{r}$
- **Decrease:** value of $\mathfrak{r}$ decreases by at least 1 for $\mathcal{P}_\succ \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_\succ \subseteq \mathcal{P}$



- $\mathfrak{r}(\ell) = \mathtt{i}$ for all locations ℓ
- Thus: $t_1 \in \mathcal{P}_\succ$

Runtime Bounds from Ranking Functions

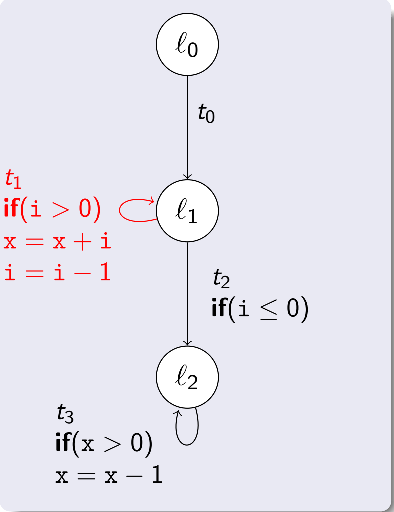
Initial bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$ as t_0 and t_2 are not in loops

Ranking function $\mathfrak{r}$ for program $\mathcal{P}$

- for all $t \in \mathcal{P}_{\succ}$, set $\mathcal{R}(t) = \mathfrak{r}(\ell_0)$
- **Non-Increase:** no transition in $\mathcal{P}$ increases value of $\mathfrak{r}$
- **Decrease:** value of $\mathfrak{r}$ decreases by at least 1 for $\mathcal{P}_{\succ} \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_{\succ} \subseteq \mathcal{P}$

- $\mathfrak{r}(\ell) = \mathtt{i}$ for all locations ℓ
- Thus: $t_1 \in \mathcal{P}_{\succ}$



Runtime Bounds from Ranking Functions

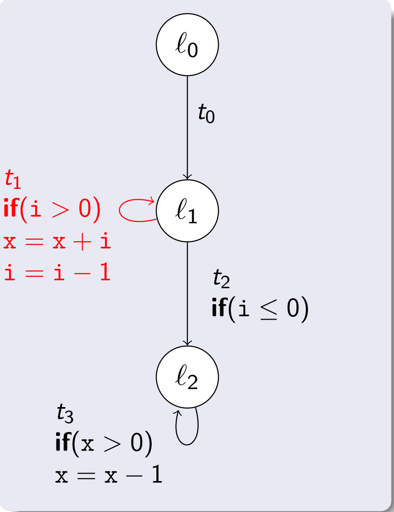
Initial bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$

Ranking function $\mathfrak{r}$ for program $\mathcal{P}$

- for all $t \in \mathcal{P}_\succ$, set $\mathcal{R}(t) = \mathfrak{r}(\ell_0)$
- **Non-Increase:** no transition in $\mathcal{P}$ increases value of $\mathfrak{r}$
- **Decrease:** value of $\mathfrak{r}$ decreases by at least 1 for $\mathcal{P}_\succ \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_\succ \subseteq \mathcal{P}$

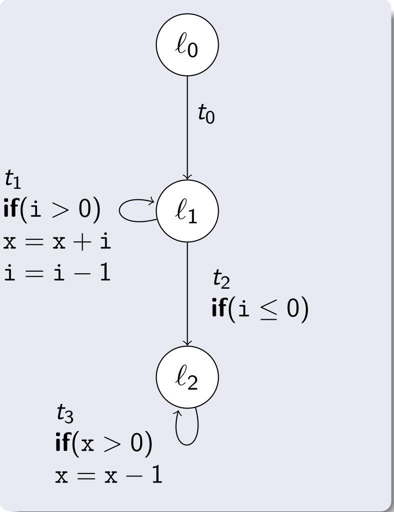
- $\mathfrak{r}(\ell) = \mathtt{i}$ for all locations ℓ
- Thus: $t_1 \in \mathcal{P}_\succ$



Modular Runtime Bounds from Ranking Functions

Current bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$

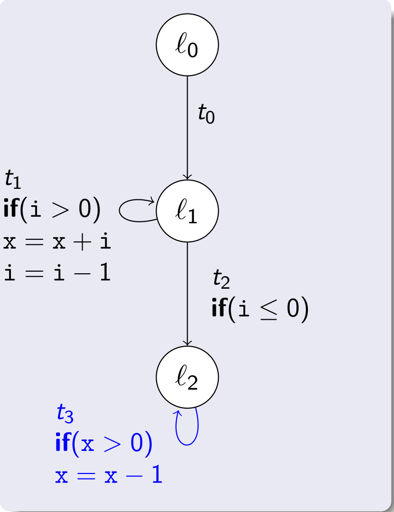


Modular Runtime Bounds from Ranking Functions

Current bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$

- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$



Modular Runtime Bounds from Ranking Functions

Current bounds

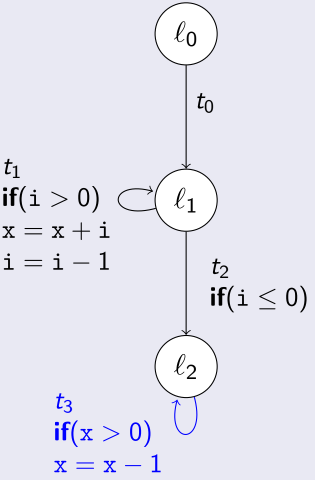
$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$

Computing runtime bound for $t \in \mathcal{P}'$

$$\mathcal{R}(t) \;=\; \qquad \mathfrak{r}(\ell)$$

- ℓ: entry location of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$



Modular Runtime Bounds from Ranking Functions

Current bounds

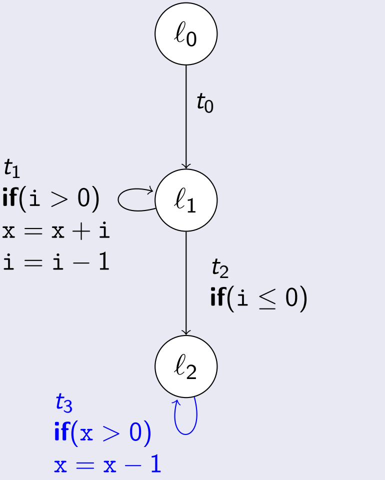
$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$

Computing runtime bound for $t \in \mathcal{P}'$

$$\mathcal{R}(t) \;=\; \mathfrak{r}(\ell)$$

- ℓ: entry location of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$



Modular Runtime Bounds from Ranking Functions

Current bounds

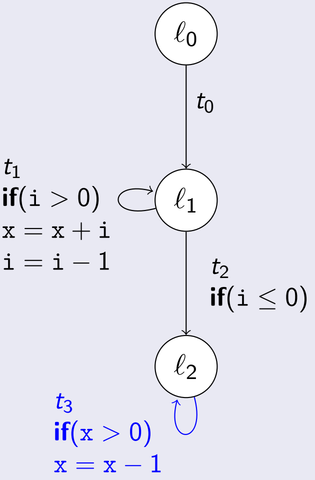
$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$

Computing runtime bound for $t \in \mathcal{P}'$

$$\mathcal{R}(t) \;=\; \qquad \mathfrak{r}(\ell)$$

- ℓ: entry location of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$



Modular Runtime Bounds from Ranking Functions

Current bounds

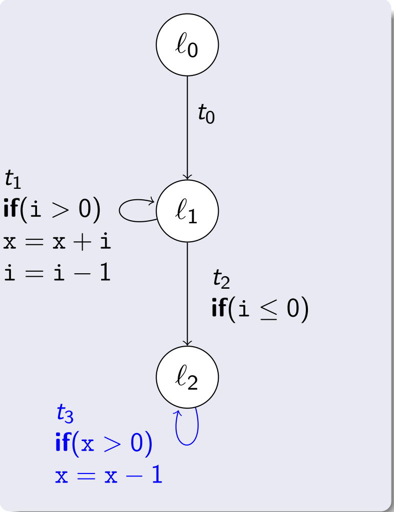
$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$, $\mathcal{R}(t_3) = \quad \mathfrak{r}(\ell_2)$

Computing runtime bound for $t \in \mathcal{P}'$

$$\mathcal{R}(t) \;=\; \qquad \mathfrak{r}(\ell)$$

- ℓ: entry location of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$



Modular Runtime Bounds from Ranking Functions

Current bounds

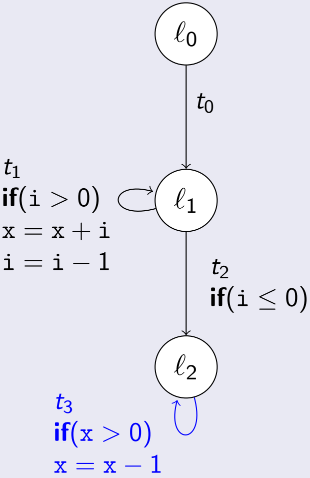
$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$, $\mathcal{R}(t_3) = \quad \mathfrak{r}(\ell_2)$

Computing runtime bound for $t \in \mathcal{P}'$

$$\mathcal{R}(t) \;=\; \qquad \mathfrak{r}(\ell)$$

- ℓ: entry location of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.



Modular Runtime Bounds from Ranking Functions

Current bounds

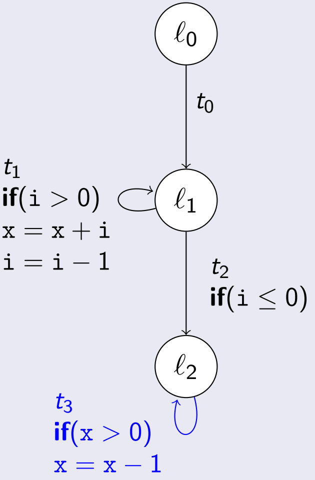
$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$, $\mathcal{R}(t_3) = \qquad \mathfrak{r}(\ell_2)$

Computing runtime bound for $t \in \mathcal{P}'$

$$\mathcal{R}(t) \;=\; \qquad \mathfrak{r}(\ell)$$

- ℓ: entry location of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:



Modular Runtime Bounds from Ranking Functions

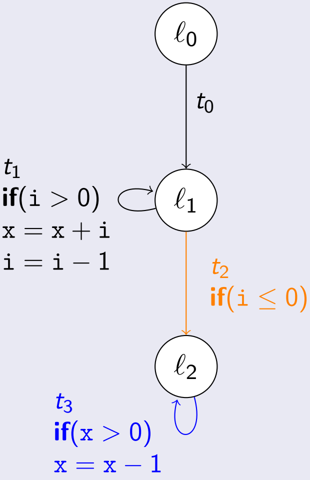
Current bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$, $\mathcal{R}(t_3) = \qquad \mathfrak{r}(\ell_2)$

Computing runtime bound for $t \in \mathcal{P}'$

$$\mathcal{R}(t) \;=\; \qquad \mathfrak{r}(\ell)$$

- ℓ: entry location of $\mathcal{P}'$



- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)

Modular Runtime Bounds from Ranking Functions

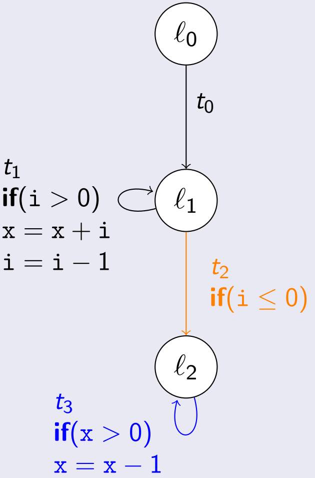
Current bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$, $\mathcal{R}(t_3) = \quad \mathfrak{r}(\ell_2)$

Computing runtime bound for $t \in \mathcal{P}'$

$$\mathcal{R}(t) \;=\; \qquad \mathfrak{r}(\ell)$$

- ℓ: entry location of $\mathcal{P}'$



- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)

$\Rightarrow$ multiply t_2's runtime bound $\mathcal{R}(t_2)$ with local bound $\mathfrak{r}(\ell_2)$

Modular Runtime Bounds from Ranking Functions

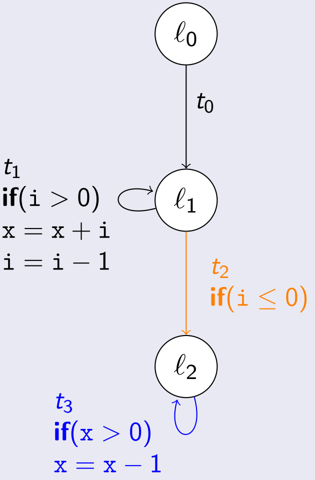
Current bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$, $\mathcal{R}(t_3) = \mathcal{R}(t_2) \cdot \mathfrak{r}(\ell_2)$

Computing runtime bound for $t \in \mathcal{P}'$

$$\mathcal{R}(t) \;=\; \mathfrak{r}(\ell)$$

- ℓ: entry location of $\mathcal{P}'$



- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)

$\Rightarrow$ multiply t_2's runtime bound $\mathcal{R}(t_2)$ with local bound $\mathfrak{r}(\ell_2)$

Modular Runtime Bounds from Ranking Functions

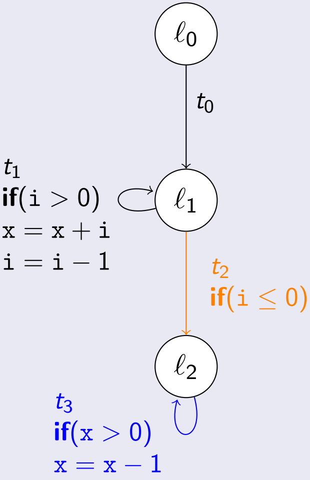
Current bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$, $\mathcal{R}(t_3) = \mathcal{R}(t_2) \cdot \mathfrak{r}(\ell_2)$

Computing runtime bound for $t \in \mathcal{P}'$

$$\mathcal{R}(t) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)$$

- ℓ: entry location of $\mathcal{P}'$
- t': pre-transition of $\mathcal{P}'$



- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)

$\Rightarrow$ multiply t_2's runtime bound $\mathcal{R}(t_2)$ with local bound $\mathfrak{r}(\ell_2)$

Modular Runtime Bounds from Ranking Functions

Current bounds

$\mathcal{R}(t_0) = 1,\ \mathcal{R}(t_2) = 1,\ \mathcal{R}(t_1) = \mathtt{i}_0,\ \mathcal{R}(t_3) = \mathcal{R}(t_2) \cdot \mathfrak{r}(\ell_2)$

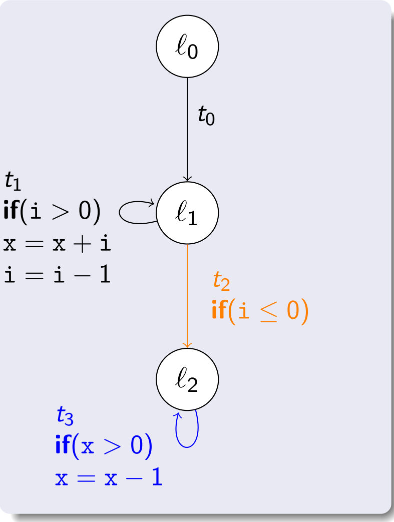
Computing runtime bound for $t \in \mathcal{P}'$

$$\mathcal{R}(t) \ = \ \mathcal{R}(t') \ \cdot \ \mathfrak{r}(\ell)$$

- ℓ: entry location of $\mathcal{P}'$
- t': pre-transition of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

$\Rightarrow$ multiply t_2's runtime bound $\mathcal{R}(t_2)$ with local bound $\mathfrak{r}(\ell_2)$



Modular Runtime Bounds from Ranking Functions

Current bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$, $\mathcal{R}(t_3) = \mathcal{R}(t_2) \cdot \mathfrak{r}(\ell_2)$

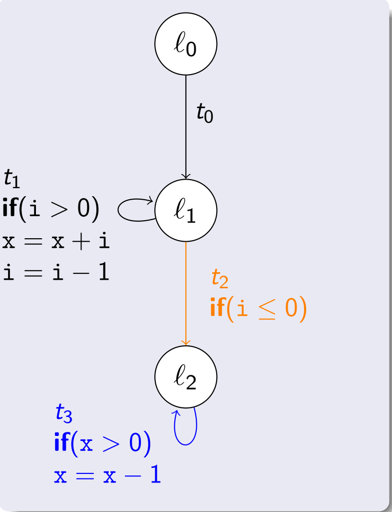
Computing runtime bound for $t \in \mathcal{P}'$

$$\mathcal{R}(t) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)$$

- ℓ: entry location of $\mathcal{P}'$
- t': pre-transition of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

$\Rightarrow$ replace $\mathfrak{r}(\ell_2)$ by $\mathfrak{r}(\ell_2)\,[\mathtt{x} \,/\, \mathcal{S}(t_2, \mathtt{x})]$



Modular Runtime Bounds from Ranking Functions

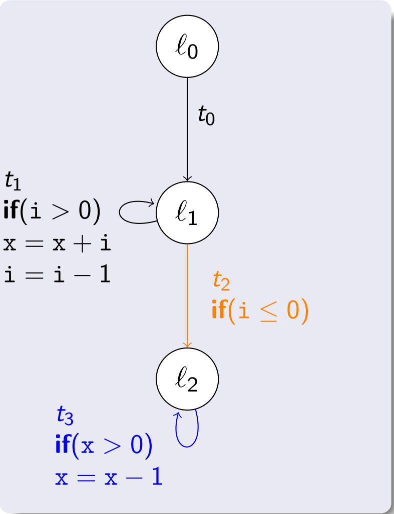
Current bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$, $\mathcal{R}(t_3) = \mathcal{R}(t_2) \cdot \mathfrak{r}(\ell_2)\,[\mathtt{x} \,/\, \mathcal{S}(t_2, \mathtt{x})]$

Computing runtime bound for $t \in \mathcal{P}'$

$$\mathcal{R}(t) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)$$

- ℓ: entry location of $\mathcal{P}'$
- t': pre-transition of $\mathcal{P}'$



- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

$\Rightarrow$ replace $\mathfrak{r}(\ell_2)$ by $\mathfrak{r}(\ell_2)\,[\mathtt{x} \,/\, \mathcal{S}(t_2, \mathtt{x})]$

Modular Runtime Bounds from Ranking Functions

Current bounds

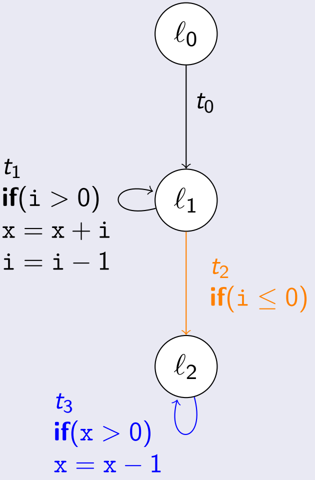
$\mathcal{R}(t_0) = 1,\ \mathcal{R}(t_2) = 1,\ \mathcal{R}(t_1) = \mathtt{i}_0,\ \mathcal{R}(t_3) = \mathcal{R}(t_2) \cdot \mathfrak{r}(\ell_2)\,[\mathtt{x} \,/\, \mathcal{S}(t_2, \mathtt{x})]$

Computing runtime bound for $t \in \mathcal{P}'$

$$\mathcal{R}(t) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}(t', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- t': pre-transition of $\mathcal{P}'$

use *size bounds* to compute *runtime bounds*



- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

$\Rightarrow$ replace $\mathfrak{r}(\ell_2)$ by $\mathfrak{r}(\ell_2)\,[\mathtt{x} \,/\, \mathcal{S}(t_2, \mathtt{x})]$

Modular Runtime Bounds from Ranking Functions

Current bounds

$\mathcal{R}(t_0) = 1,\ \mathcal{R}(t_2) = 1,\ \mathcal{R}(t_1) = \mathrm{i}_0,\ \mathcal{R}(t_3) = \quad 1 \quad \cdot\ \mathfrak{r}(\ell_2)\,[\mathrm{x}\,/\,\mathcal{S}(t_2,\mathrm{x})]$

Computing runtime bound for $t \in \mathcal{P}'$

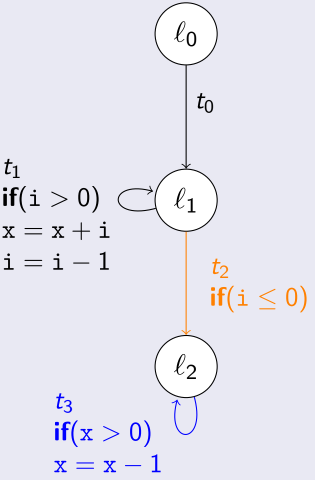
$$\mathcal{R}(t) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v\,/\,\mathcal{S}(t',v)]$$

- ℓ: entry location of $\mathcal{P}'$
- t': pre-transition of $\mathcal{P}'$

use *size bounds* to compute *runtime bounds*

- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathrm{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathrm{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider value of $\mathcal{P}'$'s initial variable x in full run

$\Rightarrow$ replace $\mathfrak{r}(\ell_2)$ by $\mathfrak{r}(\ell_2)\,[\mathrm{x}\,/\,\mathcal{S}(t_2,\mathrm{x})]$



Modular Runtime Bounds from Ranking Functions

Current bounds

$\mathcal{R}(t_0) = 1,\ \mathcal{R}(t_2) = 1,\ \mathcal{R}(t_1) = \mathtt{i}_0,\ \mathcal{R}(t_3) = \quad 1 \quad \cdot \quad \mathtt{x} \quad [\mathtt{x} \,/\, \mathcal{S}(t_2, \mathtt{x})]$

Computing runtime bound for $t \in \mathcal{P}'$

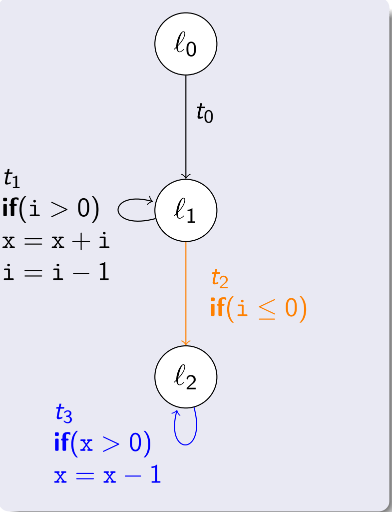
$$\mathcal{R}(t) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}(t', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- t': pre-transition of $\mathcal{P}'$

use *size bounds* to compute *runtime bounds*

- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

$\Rightarrow$ replace $\mathfrak{r}(\ell_2)$ by $\mathfrak{r}(\ell_2)\,[\mathtt{x} \,/\, \mathcal{S}(t_2, \mathtt{x})]$



Modular Runtime Bounds from Ranking Functions

Current bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$, $\mathcal{R}(t_3) = \quad 1 \quad \cdot \quad \mathcal{S}(t_2, \mathtt{x})$

Computing runtime bound for $t \in \mathcal{P}'$

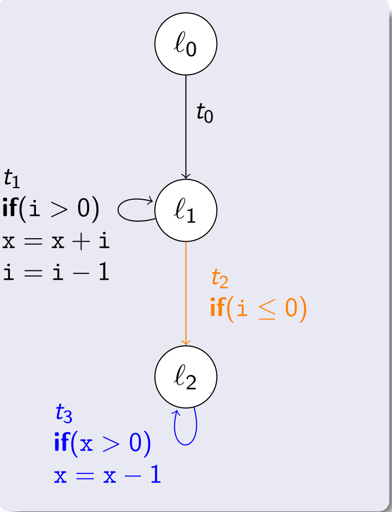
$$\mathcal{R}(t) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}(t', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- t': pre-transition of $\mathcal{P}'$

use *size bounds* to compute *runtime bounds*

- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

$\Rightarrow$ replace $\mathfrak{r}(\ell_2)$ by $\mathfrak{r}(\ell_2)\,[\mathtt{x} \,/\, \mathcal{S}(t_2, \mathtt{x})]$



Modular Runtime Bounds from Ranking Functions

Current bounds

$\mathcal{R}(t_0) = 1,\ \mathcal{R}(t_2) = 1,\ \mathcal{R}(t_1) = \mathtt{i}_0,\ \mathcal{R}(t_3) = \quad 1 \quad \cdot \quad (\,\mathtt{x}_0 + \mathtt{i}_0^2\,)$

Computing runtime bound for $t \in \mathcal{P}'$

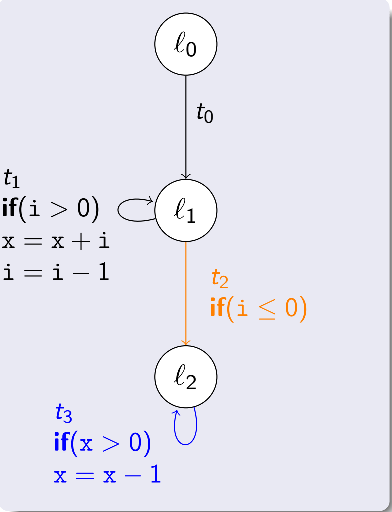
$$\mathcal{R}(t) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}(t', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- t': pre-transition of $\mathcal{P}'$

use *size bounds* to compute *runtime bounds*

- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

$\Rightarrow$ replace $\mathfrak{r}(\ell_2)$ by $\mathfrak{r}(\ell_2)\,[\mathtt{x} \,/\, \mathcal{S}(t_2, \mathtt{x})]$



Modular Runtime Bounds from Ranking Functions

Runtime bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$, $\mathcal{R}(t_3) = \mathtt{x}_0 + \mathtt{i}_0^2$

Computing runtime bound for $t \in \mathcal{P}'$

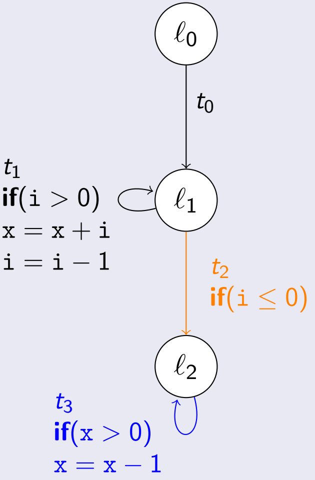
$$\mathcal{R}(t) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v\,/\,\mathcal{S}(t', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- t': pre-transition of $\mathcal{P}'$

use *size bounds* to compute *runtime bounds*

- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

$\Rightarrow$ replace $\mathfrak{r}(\ell_2)$ by $\mathfrak{r}(\ell_2)\,[\mathtt{x}\,/\,\mathcal{S}(t_2, \mathtt{x})]$



Modular Runtime Bounds from Ranking Functions

Runtime bounds

$\mathcal{R}(t_0) = 1$, $\mathcal{R}(t_2) = 1$, $\mathcal{R}(t_1) = \mathtt{i}_0$, $\mathcal{R}(t_3) = \mathtt{x}_0 + \mathtt{i}_0^2$

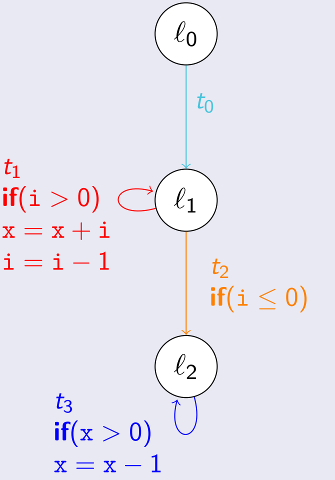
Computing runtime bound for $t \in \mathcal{P}'$

$$\mathcal{R}(t) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v\,/\,\mathcal{S}(t',v)]$$

- ℓ: entry location of $\mathcal{P}'$
- t': pre-transition of $\mathcal{P}'$

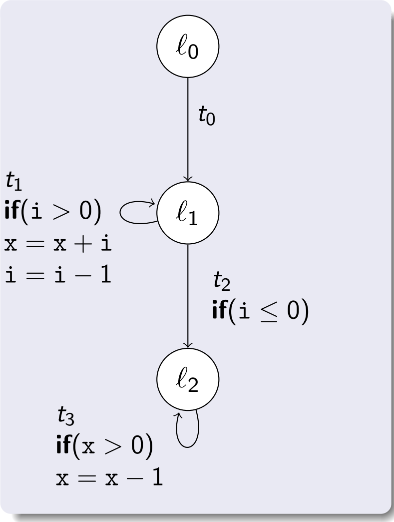
use *size bounds* to compute *runtime bounds*

- Modular use of ranking function for subset $\mathcal{P}' = \{t_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $t_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use t_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

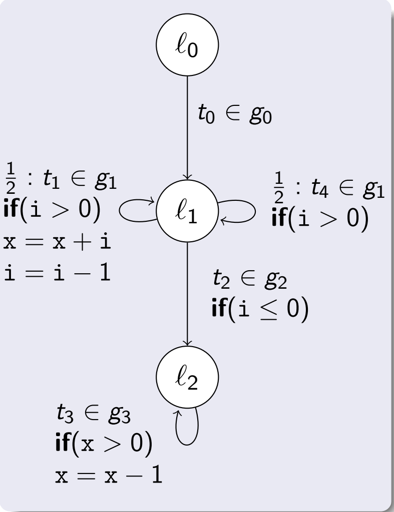


Overall runtime is bounded by $\mathcal{R}(t_0) + \ldots + \mathcal{R}(t_3) = 1 + \mathtt{i}_0 + 1 + \mathtt{x}_0 + \mathtt{i}_0^2$.

Expected Runtime Bounds from *Probabilistic* Ranking Functions



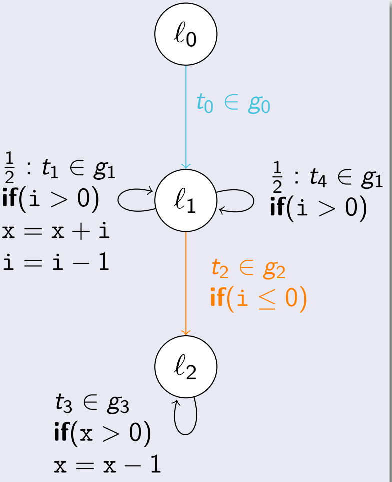
Expected Runtime Bounds from *Probabilistic* Ranking Functions



Expected Runtime Bounds from *Probabilistic* Ranking Functions

Initial bounds

$\mathcal{R}_{\mathbb{E}}(g_0)=1$, $\mathcal{R}_{\mathbb{E}}(g_2)=1$ as g_0 and g_2 are not in loops



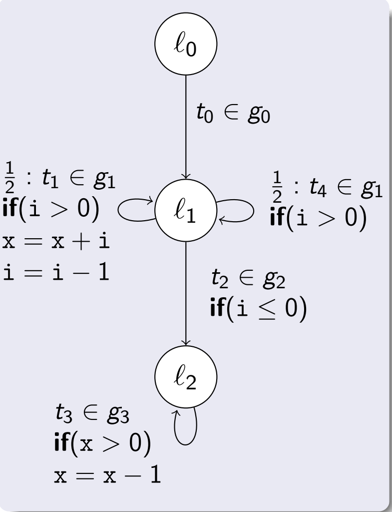
Expected Runtime Bounds from *Probabilistic* Ranking Functions

Initial bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$ as g_0 and g_2 are not in loops

ranking function $\mathfrak{r}$ for program $\mathcal{P}$

- $\mathfrak{r}$ maps *locations* to $\mathbb{R}[v_1, \ldots, v_n]$
- **Non-Increase:** no transition in $\mathcal{P}$ increases value of $\mathfrak{r}$
- **Decrease:** value of $\mathfrak{r}$ decreases by 1 for $\mathcal{P}_{\succ} \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_{\succ} \subseteq \mathcal{P}$



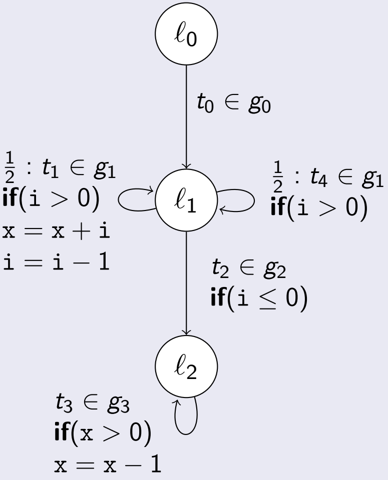
Expected Runtime Bounds from *Probabilistic* Ranking Functions

Initial bounds

$\mathcal{R}_{\mathbb{E}}(g_0)=1$, $\mathcal{R}_{\mathbb{E}}(g_2)=1$ as g_0 and g_2 are not in loops

Probabilistic ranking function $\mathfrak{r}$ for program $\mathcal{P}$

- $\mathfrak{r}$ maps *locations* to $\mathbb{R}[v_1, \ldots, v_n]$
- **Non-Increase:** no transition in $\mathcal{P}$ increases value of $\mathfrak{r}$
- **Decrease:** value of $\mathfrak{r}$ decreases by 1 for $\mathcal{P}_\succ \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_\succ \subseteq \mathcal{P}$



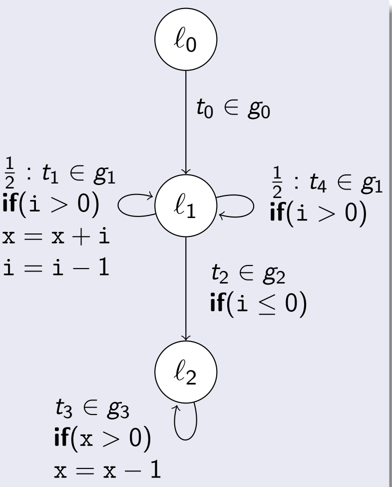
Expected Runtime Bounds from *Probabilistic* Ranking Functions

Initial bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$ as g_0 and g_2 are not in loops

Probabilistic ranking function $\mathfrak{r}$ for program $\mathcal{P}$

- for all $g \in \mathcal{P}_{\succ}$, set $\mathcal{R}_{\mathbb{E}}(g) = \mathfrak{r}(\ell_0)$
- **Non-Increase:** no transition in $\mathcal{P}$ increases value of $\mathfrak{r}$
- **Decrease:** value of $\mathfrak{r}$ decreases by 1 for $\mathcal{P}_{\succ} \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_{\succ} \subseteq \mathcal{P}$



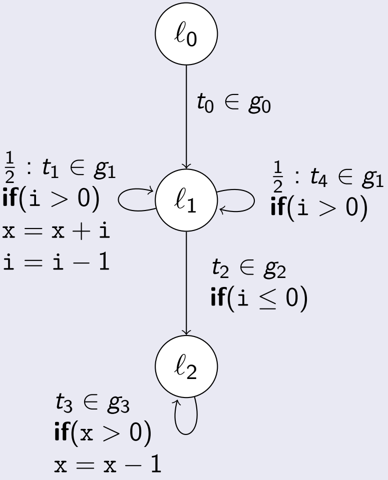
Expected Runtime Bounds from *Probabilistic* Ranking Functions

Initial bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$ as g_0 and g_2 are not in loops

Probabilistic ranking function $\mathfrak{r}$ for program $\mathcal{P}$

- for all $g \in \mathcal{P}_{\succ}$, set $\mathcal{R}_{\mathbb{E}}(g) = \mathfrak{r}(\ell_0)$
- **Non-Increase:** no transition in $\mathcal{P}$ increases *expected* value of $\mathfrak{r}$
- **Decrease:** value of $\mathfrak{r}$ decreases by 1 for $\mathcal{P}_{\succ} \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_{\succ} \subseteq \mathcal{P}$



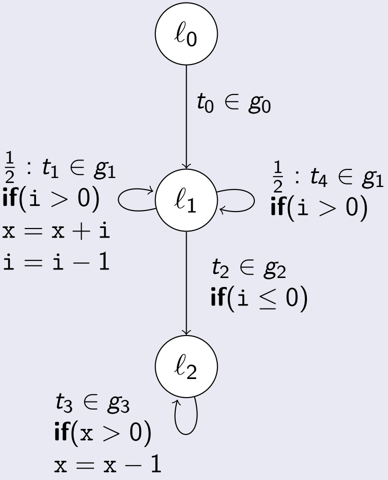
Expected Runtime Bounds from *Probabilistic* Ranking Functions

Initial bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$ as g_0 and g_2 are not in loops

***Probabilistic* ranking function $\mathfrak{r}$ for program $\mathcal{P}$**

- for all $g \in \mathcal{P}_{\succ}$, set $\mathcal{R}_{\mathbb{E}}(g) = \mathfrak{r}(\ell_0)$
- **Non-Increase:** no transition in $\mathcal{P}$ increases *expected* value of $\mathfrak{r}$
- **Decrease:** *expected* value of $\mathfrak{r}$ decreases by 1 for $\mathcal{P}_{\succ} \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_{\succ} \subseteq \mathcal{P}$



Expected Runtime Bounds from *Probabilistic* Ranking Functions

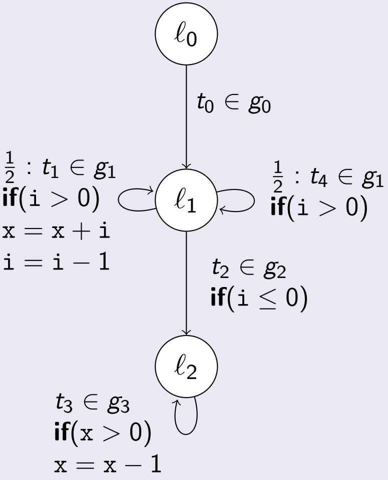
Initial bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$ as g_0 and g_2 are not in loops

Probabilistic **ranking function $\mathfrak{r}$ for program $\mathcal{P}$**

- for all $g \in \mathcal{P}_{\succ}$, set $\mathcal{R}_{\mathbb{E}}(g) = \mathfrak{r}(\ell_0)$
- **Non-Increase:** no transition in $\mathcal{P}$ increases *expected* value of $\mathfrak{r}$
- **Decrease:** *expected* value of $\mathfrak{r}$ decreases by 1 for $\mathcal{P}_{\succ} \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_{\succ} \subseteq \mathcal{P}$

- $\mathfrak{r}(\ell) = 2 \cdot \mathtt{i}$ for all locations ℓ



Expected Runtime Bounds from *Probabilistic* Ranking Functions

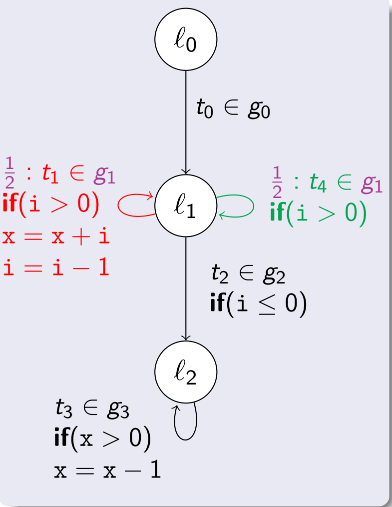
Initial bounds

$\mathcal{R}_{\mathbb{E}}(g_0)=1$, $\mathcal{R}_{\mathbb{E}}(g_2)=1$ as g_0 and g_2 are not in loops

***Probabilistic* ranking function $\mathfrak{r}$ for program $\mathcal{P}$**

- for all $g \in \mathcal{P}_{\succ}$, set $\mathcal{R}_{\mathbb{E}}(g) = \mathfrak{r}(\ell_0)$
- **Non-Increase:** no transition in $\mathcal{P}$ increases *expected* value of $\mathfrak{r}$
- **Decrease:** *expected* value of $\mathfrak{r}$ decreases by 1 for $\mathcal{P}_{\succ} \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_{\succ} \subseteq \mathcal{P}$

- $\mathfrak{r}(\ell) = 2 \cdot \mathtt{i}$ for all locations ℓ
- Thus: $g_1 \in \mathcal{P}_{\succ}$



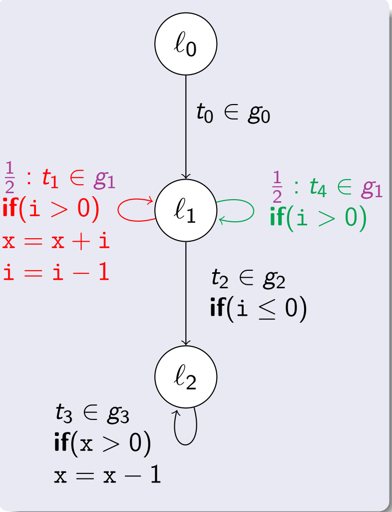
Expected Runtime Bounds from *Probabilistic* Ranking Functions

Initial bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$ as g_0 and g_2 are not in loops

Probabilistic ranking function $\mathfrak{r}$ for program $\mathcal{P}$

- for all $g \in \mathcal{P}_{\succ}$, set $\mathcal{R}_{\mathbb{E}}(g) = \mathfrak{r}(\ell_0)$
- **Non-Increase:** no transition in $\mathcal{P}$ increases *expected* value of $\mathfrak{r}$
- **Decrease:** *expected* value of $\mathfrak{r}$ decreases by 1 for $\mathcal{P}_{\succ} \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_{\succ} \subseteq \mathcal{P}$



- $\mathfrak{r}(\ell) = 2 \cdot \mathtt{i}$ for all locations ℓ
- Thus: $g_1 \in \mathcal{P}_{\succ}$

$$\mathfrak{r}(\ell_1) \geq \frac{1}{2} \cdot \mathfrak{r}(\ell_1)\,[\mathtt{x} \,/\, \mathtt{x} + \mathtt{i},\ \mathtt{i} \,/\, \mathtt{i} - 1] + \frac{1}{2} \cdot \mathfrak{r}(\ell_1)\,[\mathtt{x} \,/\, \mathtt{x},\ \mathtt{i} \,/\, \mathtt{i}] + 1$$

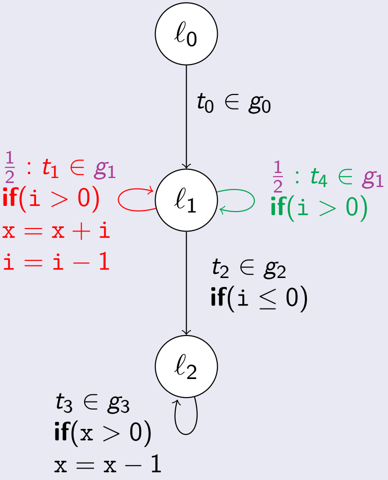
Expected Runtime Bounds from *Probabilistic* Ranking Functions

Initial bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$ as g_0 and g_2 are not in loops

***Probabilistic* ranking function $\mathfrak{r}$ for program $\mathcal{P}$**

- for all $g \in \mathcal{P}_{\succ}$, set $\mathcal{R}_{\mathbb{E}}(g) = \mathfrak{r}(\ell_0)$
- **Non-Increase:** no transition in $\mathcal{P}$ increases *expected* value of $\mathfrak{r}$
- **Decrease:** *expected* value of $\mathfrak{r}$ decreases by 1 for $\mathcal{P}_{\succ} \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_{\succ} \subseteq \mathcal{P}$



- $\mathfrak{r}(\ell) = 2 \cdot \mathtt{i}$ for all locations ℓ
- Thus: $g_1 \in \mathcal{P}_{\succ}$

$$2 \cdot \mathtt{i} \;\geq\; \tfrac{1}{2} \cdot \mathfrak{r}(\ell_1)\,[\mathtt{x} \,/\, \mathtt{x} + \mathtt{i},\ \mathtt{i} \,/\, \mathtt{i} - 1] \;+\; \tfrac{1}{2} \cdot \mathfrak{r}(\ell_1)\,[\mathtt{x} \,/\, \mathtt{x},\ \mathtt{i} \,/\, \mathtt{i}] \;+\; 1$$

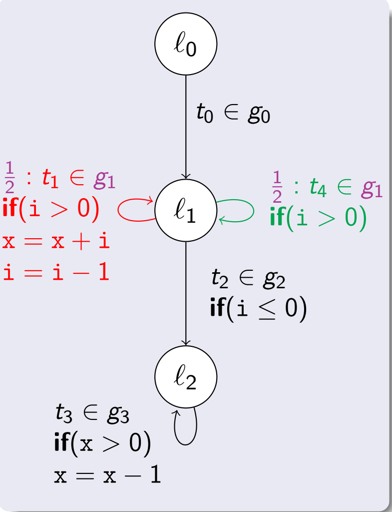
Expected Runtime Bounds from *Probabilistic* Ranking Functions

Initial bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$ as g_0 and g_2 are not in loops

Probabilistic ranking function $\mathfrak{r}$ for program $\mathcal{P}$

- for all $g \in \mathcal{P}_{\succ}$, set $\mathcal{R}_{\mathbb{E}}(g) = \mathfrak{r}(\ell_0)$
- **Non-Increase:** no transition in $\mathcal{P}$ increases *expected* value of $\mathfrak{r}$
- **Decrease:** *expected* value of $\mathfrak{r}$ decreases by 1 for $\mathcal{P}_{\succ} \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_{\succ} \subseteq \mathcal{P}$



- $\mathfrak{r}(\ell) = 2 \cdot \mathtt{i}$ for all locations ℓ
- Thus: $g_1 \in \mathcal{P}_{\succ}$

$$2 \cdot \mathtt{i} \;\geq\; \tfrac{1}{2} \cdot \quad 2 \cdot (\mathtt{i} - 1) \quad + \; \tfrac{1}{2} \cdot \mathfrak{r}(\ell_1)\,[\mathtt{x} \,/\, \mathtt{x},\ \mathtt{i} \,/\, \mathtt{i}] \; + 1$$

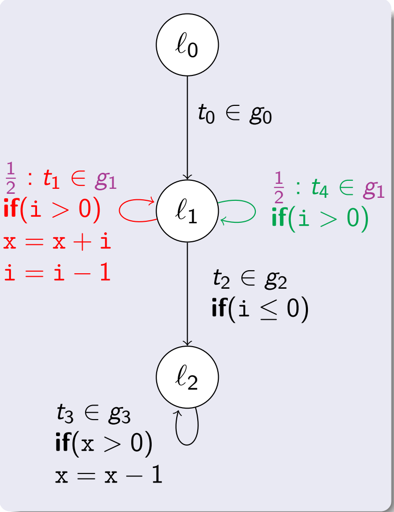
Expected Runtime Bounds from *Probabilistic* Ranking Functions

Initial bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$ as g_0 and g_2 are not in loops

Probabilistic **ranking function $\mathfrak{r}$ for program $\mathcal{P}$**

- for all $g \in \mathcal{P}_{\succ}$, set $\mathcal{R}_{\mathbb{E}}(g) = \mathfrak{r}(\ell_0)$
- **Non-Increase:** no transition in $\mathcal{P}$ increases *expected* value of $\mathfrak{r}$
- **Decrease:** *expected* value of $\mathfrak{r}$ decreases by 1 for $\mathcal{P}_{\succ} \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_{\succ} \subseteq \mathcal{P}$



- $\mathfrak{r}(\ell) = 2 \cdot \mathtt{i}$ for all locations ℓ
- Thus: $g_1 \in \mathcal{P}_{\succ}$

$$2 \cdot \mathtt{i} \quad \geq \quad \frac{1}{2} \cdot \qquad 2 \cdot (\mathtt{i} - 1) \qquad + \frac{1}{2} \cdot \qquad 2 \cdot \mathtt{i} \qquad + 1$$

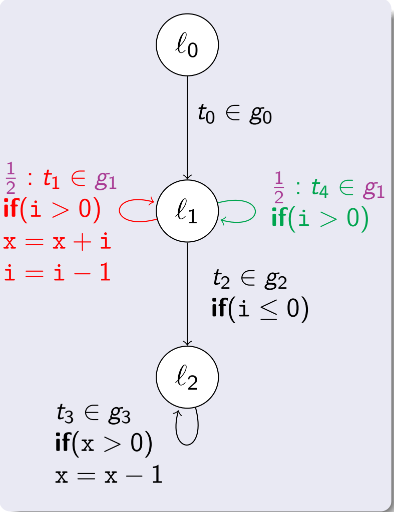
Expected Runtime Bounds from *Probabilistic* Ranking Functions

Initial bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$, $\mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$

***Probabilistic* ranking function $\mathfrak{r}$ for program $\mathcal{P}$**

- for all $g \in \mathcal{P}_{\succ}$, set $\mathcal{R}_{\mathbb{E}}(g) = \mathfrak{r}(\ell_0)$
- **Non-Increase:** no transition in $\mathcal{P}$ increases *expected* value of $\mathfrak{r}$
- **Decrease:** *expected* value of $\mathfrak{r}$ decreases by 1 for $\mathcal{P}_{\succ} \subseteq \mathcal{P}$
- **Boundedness:** $\mathfrak{r} \geq 0$ after $\mathcal{P}_{\succ} \subseteq \mathcal{P}$



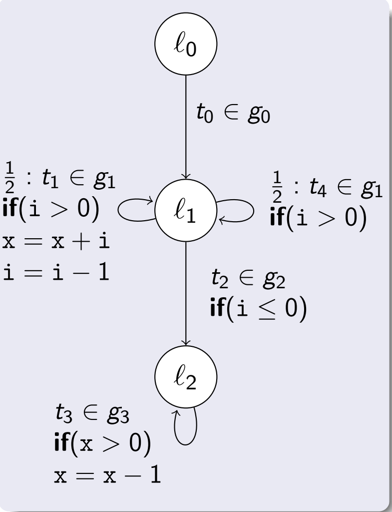
- $\mathfrak{r}(\ell) = 2 \cdot \mathtt{i}$ for all locations ℓ
- Thus: $g_1 \in \mathcal{P}_{\succ}$

$$2 \cdot \mathtt{i} \geq \frac{1}{2} \cdot 2 \cdot (\mathtt{i} - 1) + \frac{1}{2} \cdot 2 \cdot \mathtt{i} + 1$$

Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

$\mathcal{R}_{\mathbb{E}}(g_0)=1,\ \mathcal{R}_{\mathbb{E}}(g_2)=1,\ \mathcal{R}_{\mathbb{E}}(g_1)=2\cdot \mathtt{i}_0$



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

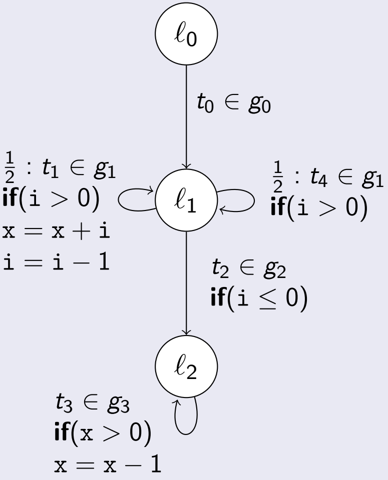
Current bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1,\ \mathcal{R}_{\mathbb{E}}(g_2) = 1,\ \mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$

Computing runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}\ (g) = \mathcal{R}(g') \cdot \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}\ (g', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g' : pre-transition of $\mathcal{P}'$



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

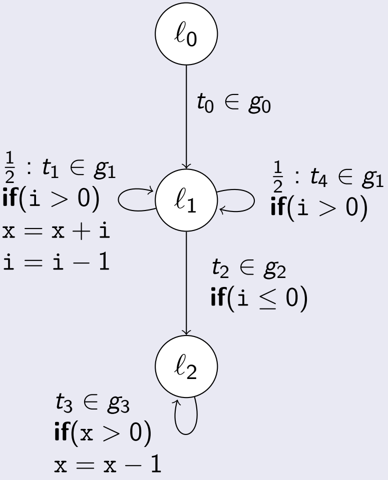
Current bounds

$\mathcal{R}_{\mathbb{E}}(g_0)=1,\ \mathcal{R}_{\mathbb{E}}(g_2)=1,\ \mathcal{R}_{\mathbb{E}}(g_1)=2\cdot \mathtt{i}_0$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}\ (g)\ =\ \mathcal{R}(g')\ \cdot\ \mathfrak{r}(\ell)\,[v\,/\,\mathcal{S}\ (g',v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g' : pre-transition of $\mathcal{P}'$



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

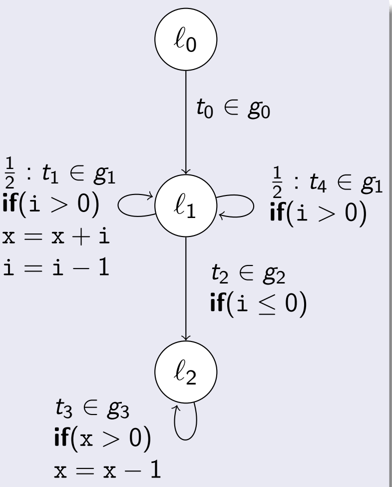
Current bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$, $\mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathbb{E}(\; \mathcal{R}(g') \qquad \cdot \qquad \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S} \;\; (g', v)] \;)$$

- ℓ: entry location of $\mathcal{P}'$
- g' : pre-transition of $\mathcal{P}'$



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

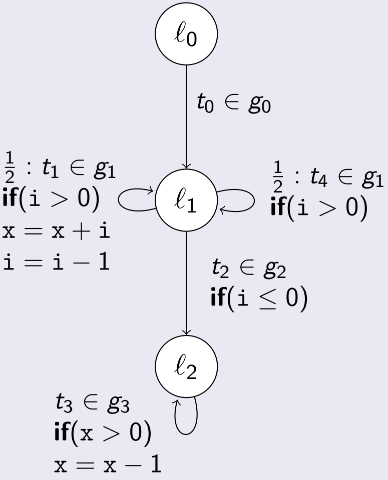
$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$, $\mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathbb{E}(\; \mathcal{R}(g') \quad \cdot \quad \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}\;(g', v)]\;)$$

- ℓ: entry location of $\mathcal{P}'$
- g' : pre-transition of $\mathcal{P}'$

Expected value *not* multiplicative!



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

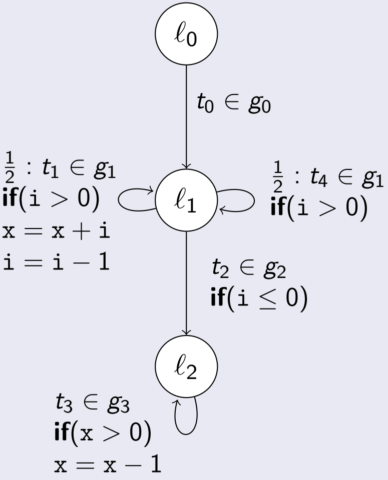
$\mathcal{R}_{\mathbb{E}}(g_0)=1$, $\mathcal{R}_{\mathbb{E}}(g_2)=1$, $\mathcal{R}_{\mathbb{E}}(g_1)=2\cdot \mathtt{i}_0$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathcal{R}(t') \cdot \mathbb{E}(\mathfrak{r}(\ell)\,[v\,/\,\mathcal{S}\,(g', v)]\,)$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

Expected value *not* multiplicative!



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

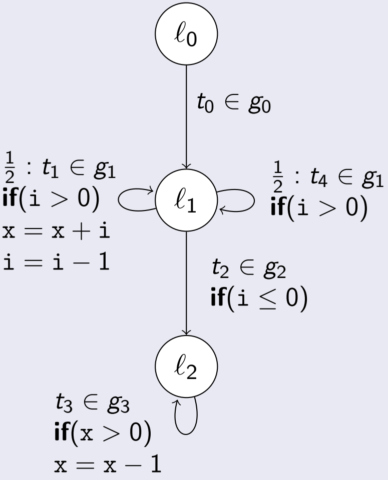
$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$, $\mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathcal{R}(t') \;\cdot\; \mathbb{E}(\,\mathfrak{r}(\ell)\,[v\,/\,\mathcal{S}\;(g', v)]\,)$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

Expected value *not* multiplicative!
$\Rightarrow$ restrict to *linear* ranking functions $\mathfrak{r}$



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

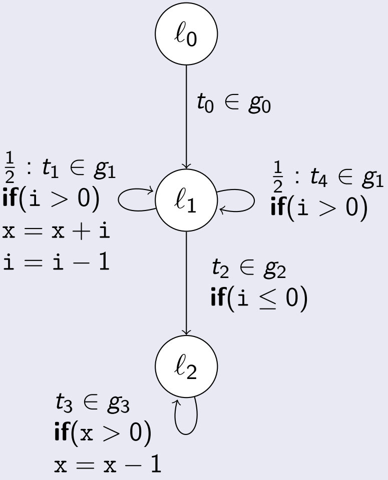
$\mathcal{R}_{\mathbb{E}}(g_0) = 1,\ \mathcal{R}_{\mathbb{E}}(g_2) = 1,\ \mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \ = \ \mathcal{R}(t') \ \cdot \ \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}_{\mathbb{E}}(g', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

Expected value *not* multiplicative!
$\Rightarrow$ restrict to *linear* ranking functions $\mathfrak{r}$



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

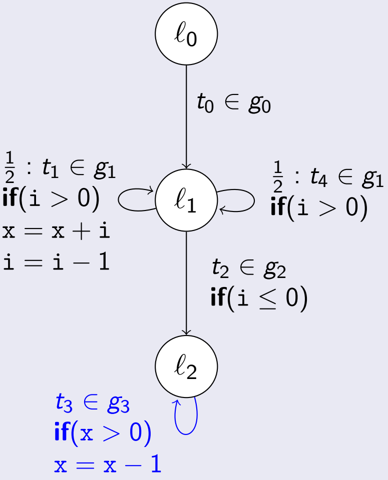
$\mathcal{R}_{\mathbb{E}}(g_0) = 1,\ \mathcal{R}_{\mathbb{E}}(g_2) = 1,\ \mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \; = \; \mathcal{R}(t') \; \cdot \; \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}_{\mathbb{E}}(g', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

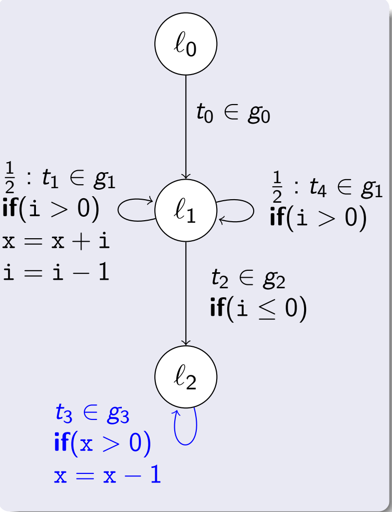
$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$, $\mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}_{\mathbb{E}}(g', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

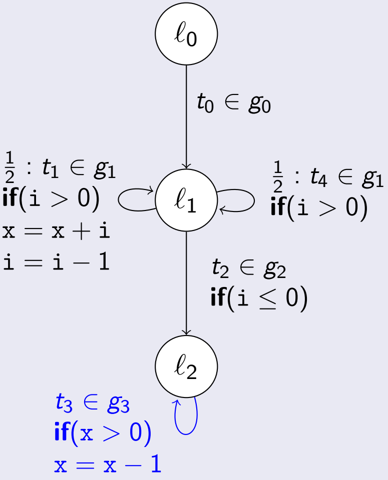
$\mathcal{R}_{\mathbb{E}}(g_0)=1$, $\mathcal{R}_{\mathbb{E}}(g_2)=1$, $\mathcal{R}_{\mathbb{E}}(g_1)=2\cdot \mathtt{i}_0$, $\mathcal{R}_{\mathbb{E}}(g_3)=$ $\mathfrak{r}(\ell_2)$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v\,/\,\mathcal{S}_{\mathbb{E}}(g', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $g_3 \in \mathcal{P}'_{\succ}$



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

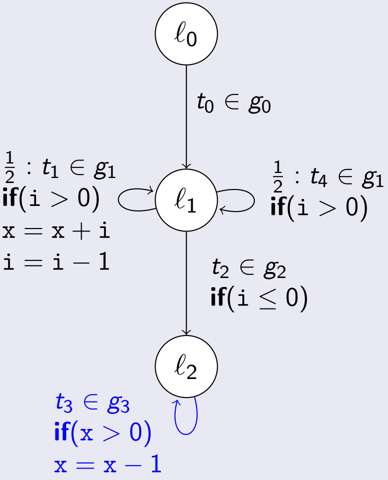
$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$, $\mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$, $\mathcal{R}_{\mathbb{E}}(g_3) =$ $\mathfrak{r}(\ell_2)$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v\,/\,\mathcal{S}_{\mathbb{E}}(g', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $g_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use g_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

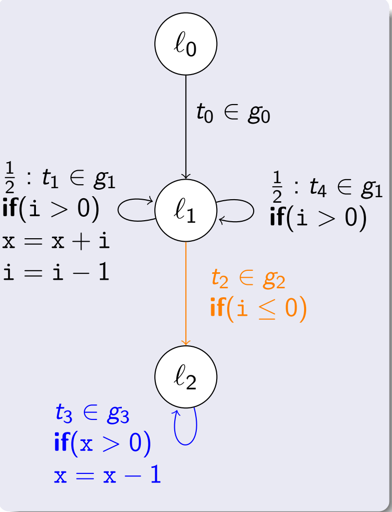
$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$, $\mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$, $\mathcal{R}_{\mathbb{E}}(g_3) = \quad \mathfrak{r}(\ell_2)$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}_{\mathbb{E}}(g', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $g_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use g_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

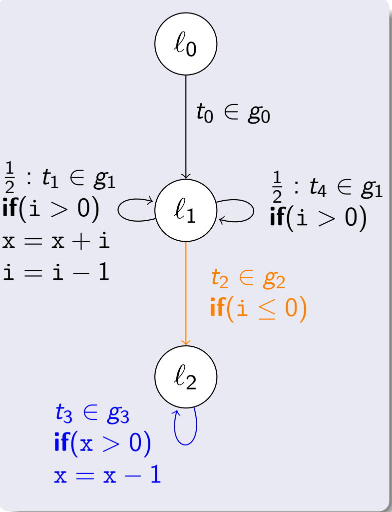
$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$, $\mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$, $\mathcal{R}_{\mathbb{E}}(g_3) =$ $\mathfrak{r}(\ell_2)$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \; = \; \mathcal{R}(t') \; \cdot \; \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}_{\mathbb{E}}(g', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $g_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use g_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)



$\Rightarrow$ multiply t_2's *non-probabilistic* runtime bound $\mathcal{R}(t_2)$ with local bound $\mathfrak{r}(\ell_2)$

Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

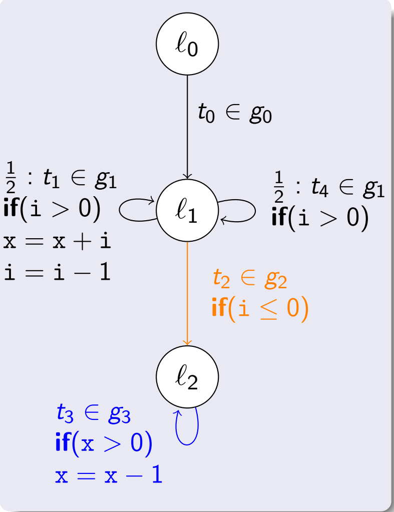
Current bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1,\ \mathcal{R}_{\mathbb{E}}(g_2) = 1,\ \mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0,\ \mathcal{R}_{\mathbb{E}}(g_3) = \mathcal{R}(t_2) \cdot \mathfrak{r}(\ell_2)$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \ = \ \mathcal{R}(t') \ \cdot \ \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}_{\mathbb{E}}(g', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$



- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $g_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use g_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)

$\Rightarrow$ multiply t_2's *non-probabilistic* runtime bound $\mathcal{R}(t_2)$ with local bound $\mathfrak{r}(\ell_2)$

Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

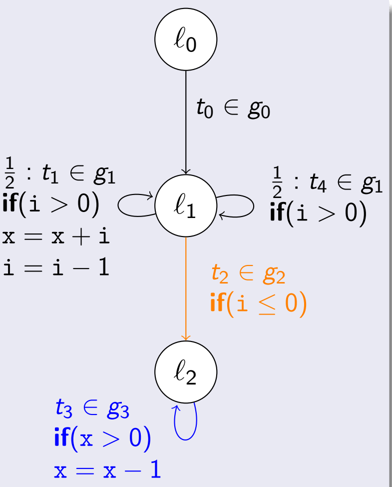
$\mathcal{R}_{\mathbb{E}}(g_0)=1,\ \mathcal{R}_{\mathbb{E}}(g_2)=1,\ \mathcal{R}_{\mathbb{E}}(g_1)=2\cdot \mathtt{i}_0,\ \mathcal{R}_{\mathbb{E}}(g_3)=\quad 1 \quad \cdot\ \mathfrak{r}(\ell_2)$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \ = \quad \mathcal{R}(t') \quad \cdot \quad \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}_{\mathbb{E}}(g', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $g_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use g_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)



$\Rightarrow$ multiply t_2's *non-probabilistic* runtime bound $\mathcal{R}(t_2)$ with local bound $\mathfrak{r}(\ell_2)$

Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

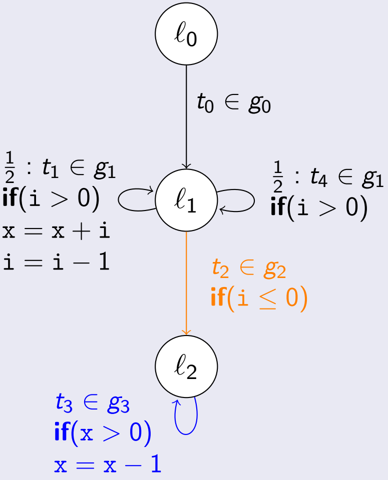
$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$, $\mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$, $\mathcal{R}_{\mathbb{E}}(g_3) = \mathfrak{r}(\ell_2)$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v\,/\,\mathcal{S}_{\mathbb{E}}(g', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $g_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use g_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)



$\Rightarrow$ multiply t_2's *non-probabilistic* runtime bound $\mathcal{R}(t_2)$ with local bound $\mathfrak{r}(\ell_2)$

Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

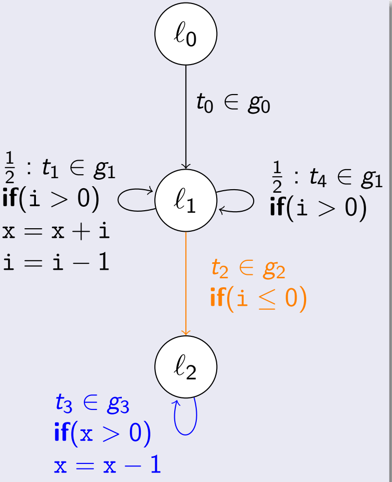
Current bounds

$\mathcal{R}_{\mathbb{E}}(g_0)=1,\ \mathcal{R}_{\mathbb{E}}(g_2)=1,\ \mathcal{R}_{\mathbb{E}}(g_1)=2\cdot \mathtt{i}_0,\ \mathcal{R}_{\mathbb{E}}(g_3)=\mathfrak{r}(\ell_2)$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v\,/\,\mathcal{S}_{\mathbb{E}}(g',v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$



- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $g_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use g_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider *expected* value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

$\Rightarrow$ multiply t_2's *non-probabilistic* runtime bound $\mathcal{R}(t_2)$ with local bound $\mathfrak{r}(\ell_2)$

Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$, $\mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$, $\mathcal{R}_{\mathbb{E}}(g_3) = \mathfrak{r}(\ell_2)$

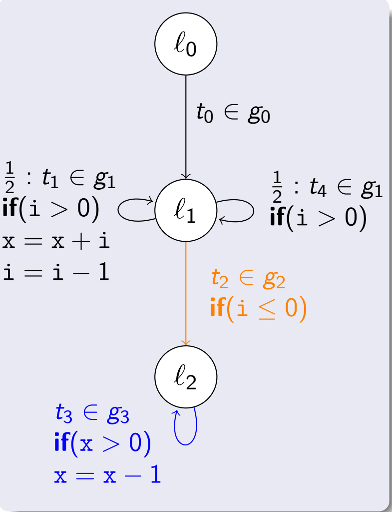
Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}_{\mathbb{E}}(g', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $g_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use g_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider *expected* value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

$\Rightarrow$ replace $\mathfrak{r}(\ell_2)$ by $\mathfrak{r}(\ell_2)\,[\mathtt{x} \,/\, \mathcal{S}_{\mathbb{E}}(g_2, \mathtt{x})]$



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$, $\mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$, $\mathcal{R}_{\mathbb{E}}(g_3) = \mathfrak{r}(\ell_2)\,[\mathtt{x} / \mathcal{S}_{\mathbb{E}}(g_2, \mathtt{x})]$

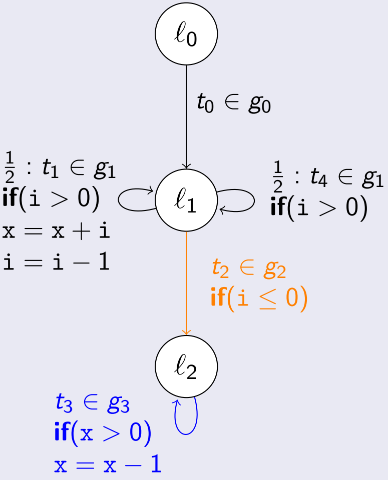
Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}_{\mathbb{E}}(g', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $g_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use g_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider *expected* value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

$\Rightarrow$ replace $\mathfrak{r}(\ell_2)$ by $\mathfrak{r}(\ell_2)\,[\mathtt{x} \,/\, \mathcal{S}_{\mathbb{E}}(g_2, \mathtt{x})]$



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

$\mathcal{R}_{\mathbb{E}}(g_0)=1,\ \mathcal{R}_{\mathbb{E}}(g_2)=1,\ \mathcal{R}_{\mathbb{E}}(g_1)=2\cdot \mathtt{i}_0,\ \mathcal{R}_{\mathbb{E}}(g_3)= \quad \mathtt{x} \quad [\mathtt{x}/\mathcal{S}_{\mathbb{E}}(g_2,\mathtt{x})]$

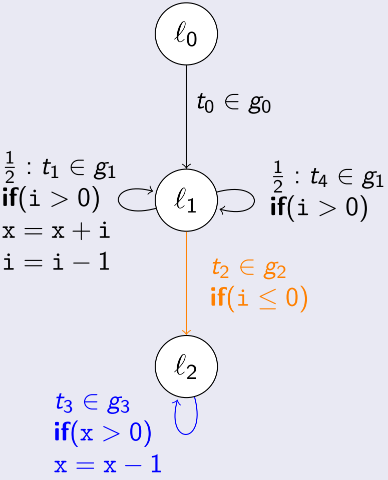
Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathcal{R}(t') \quad\cdot\quad \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}_{\mathbb{E}}(g',v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $g_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use g_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider *expected* value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

$\Rightarrow$ replace $\mathfrak{r}(\ell_2)$ by $\mathfrak{r}(\ell_2)\,[\mathtt{x} \,/\, \mathcal{S}_{\mathbb{E}}(g_2,\mathtt{x})]$



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Current bounds

$\mathcal{R}_{\mathbb{E}}(g_0)=1,\ \mathcal{R}_{\mathbb{E}}(g_2)=1,\ \mathcal{R}_{\mathbb{E}}(g_1)=2\cdot\mathtt{i}_0,\ \mathcal{R}_{\mathbb{E}}(g_3)=\mathcal{S}_{\mathbb{E}}(g_2,\mathtt{x})$

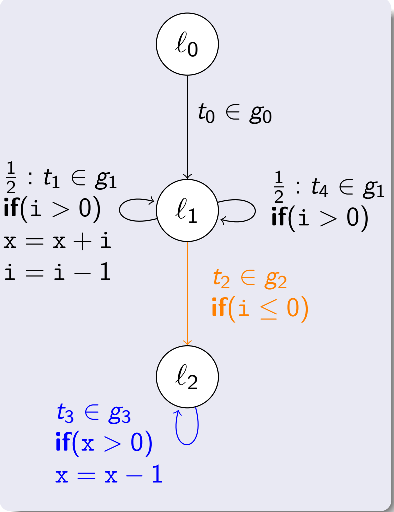
Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}_{\mathbb{E}}(g',v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $g_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use g_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider *expected* value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

$\Rightarrow$ replace $\mathfrak{r}(\ell_2)$ by $\mathfrak{r}(\ell_2)\,[\mathtt{x} \,/\, \mathcal{S}_{\mathbb{E}}(g_2,\mathtt{x})]$



Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

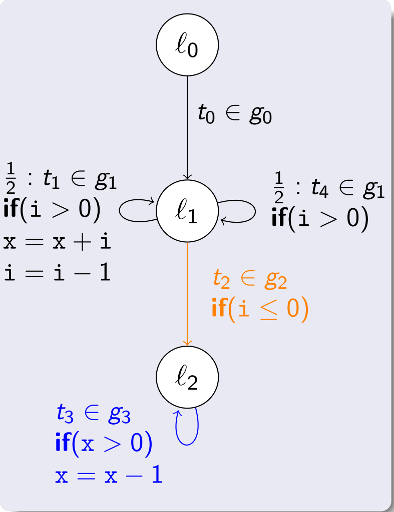
Expected runtime bounds

$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$, $\mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$, $\mathcal{R}_{\mathbb{E}}(g_3) = \mathtt{x}_0 + \mathtt{i}_0^2$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v\,/\,\mathcal{S}_{\mathbb{E}}(g', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$



- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $g_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use g_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider *expected* value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

$\Rightarrow$ replace $\mathfrak{r}(\ell_2)$ by $\mathfrak{r}(\ell_2)\,[\mathtt{x}\,/\,\mathcal{S}_{\mathbb{E}}(g_2, \mathtt{x})]$

Modular *Expected* Runtime Bounds from *Probabilistic* Ranking Functions

Expected runtime bounds

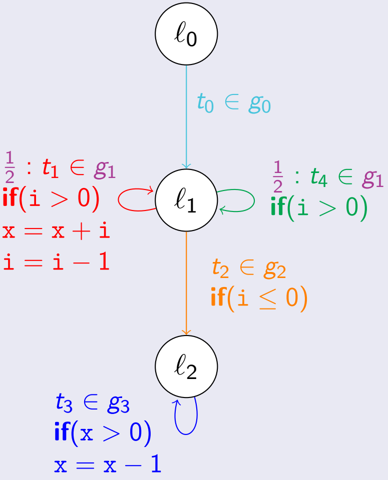
$\mathcal{R}_{\mathbb{E}}(g_0) = 1$, $\mathcal{R}_{\mathbb{E}}(g_2) = 1$, $\mathcal{R}_{\mathbb{E}}(g_1) = 2 \cdot \mathtt{i}_0$, $\mathcal{R}_{\mathbb{E}}(g_3) = \mathtt{x}_0 + \mathtt{i}_0^2$

Computing *expected* runtime bound for $g \in \mathcal{P}'$

$$\mathcal{R}_{\mathbb{E}}(g) \;=\; \mathcal{R}(t') \;\cdot\; \mathfrak{r}(\ell)\,[v \,/\, \mathcal{S}_{\mathbb{E}}(g', v)]$$

- ℓ: entry location of $\mathcal{P}'$
- g', t': pre-transition of $\mathcal{P}'$

- Modular use of ranking function for subset $\mathcal{P}' = \{g_3\}$
- $\mathfrak{r}(\ell_2) = \mathtt{x}$ Thus: $g_3 \in \mathcal{P}'_{\succ}$
- Executions of $\mathcal{P}'$ starting in ℓ_2 use g_3 at most $\mathfrak{r}(\ell_2) = \mathtt{x}$ times.
- For global result:
  - consider how often $\mathcal{P}'$ is reached (by t_2)
  - consider *expected* value of $\mathcal{P}'$'s initial variable $\mathtt{x}$ in full run

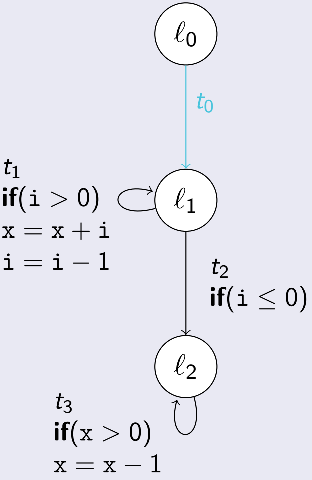


Overall *expected* runtime is bounded by $\mathcal{R}_{\mathbb{E}}(g_0) + \ldots + \mathcal{R}_{\mathbb{E}}(g_3) = 1 + 2 \cdot \mathtt{i}_0 + 1 + \mathtt{x}_0 + \mathtt{i}_0^2$.

Size Bounds

Size bounds

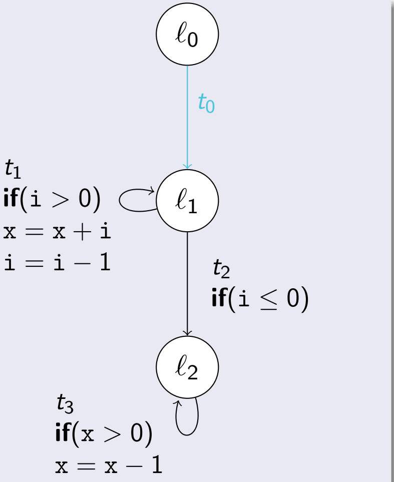
$\mathcal{S}(t_0, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_0, \mathtt{x}) = \mathtt{x}_0$



Size Bounds

Size bounds

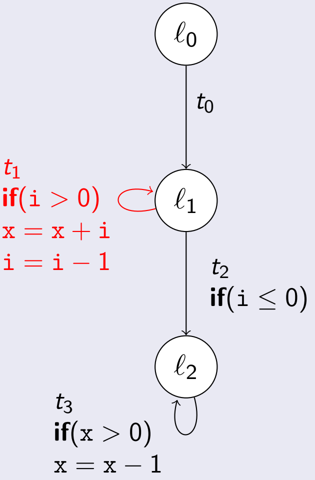
$\mathcal{S}(t_0, v) = v_0$



Size Bounds

Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$



Size Bounds

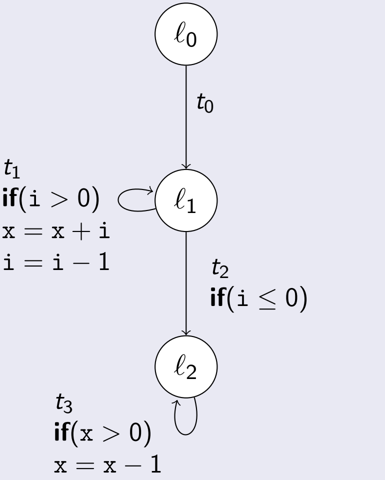
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t



Size Bounds

Size bounds

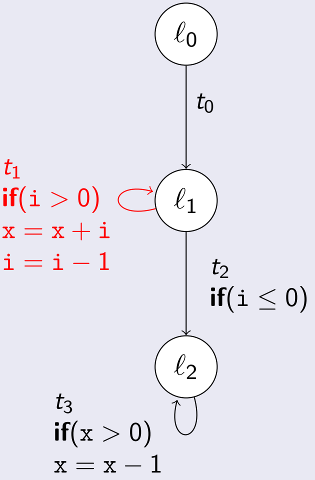
$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \ = \qquad \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t

- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$



Size Bounds

Size bounds

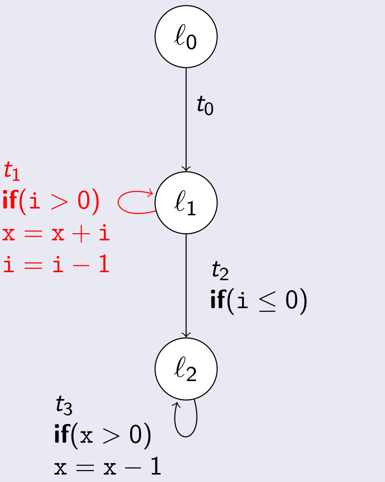
$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \qquad \mathcal{LC}(t_1, \mathtt{x})$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \qquad \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t

- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$



Size Bounds

Size bounds

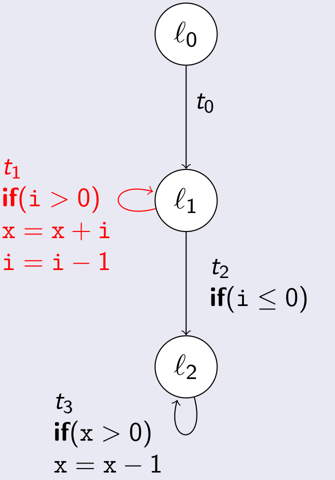
$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \qquad \mathcal{LC}(t_1, \mathtt{x})$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \qquad \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t

- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:



Size Bounds

Size bounds

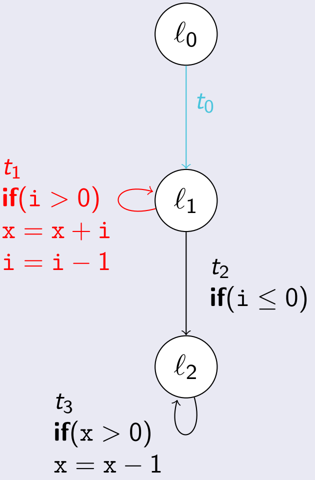
$\mathcal{S}(t_0, v) = v_0,\ \mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0,\ \mathcal{S}(t_1, \mathtt{x}) = \qquad \mathcal{LC}(t_1, \mathtt{x})$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \ = \qquad \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t

- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)



Size Bounds

Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \qquad \mathcal{LC}(t_1, \mathtt{x})$

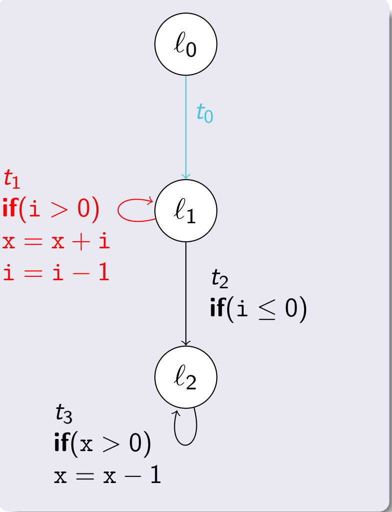
Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \qquad \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t

- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)

$\Rightarrow$ add size bound $\mathcal{S}(t_0, \mathtt{x})$ to $\mathcal{LC}(t_1, \mathtt{x})$



Size Bounds

Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathcal{S}(t_0, \mathtt{x}) + \mathcal{LC}(t_1, \mathtt{x})$

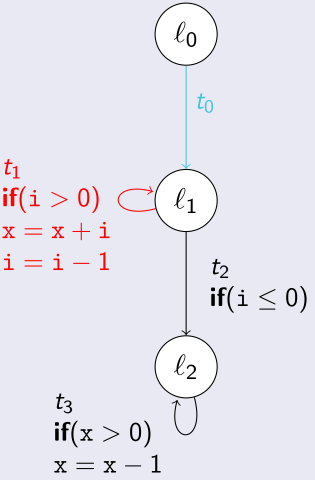
Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) = \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t

- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)

$\Rightarrow$ add size bound $\mathcal{S}(t_0, \mathtt{x})$ to $\mathcal{LC}(t_1, \mathtt{x})$



Size Bounds

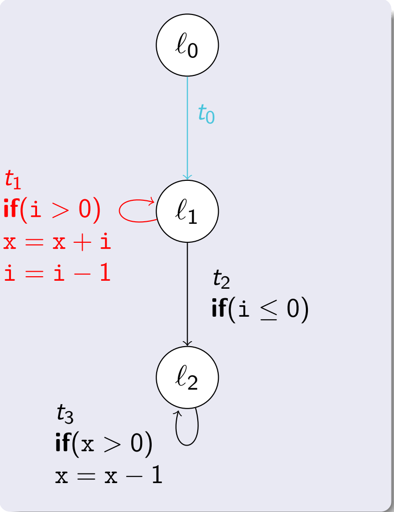
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathcal{S}(t_0, \mathtt{x}) + \quad \mathcal{LC}(t_1, \mathtt{x})$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \qquad \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)

$\Rightarrow$ add size bound $\mathcal{S}(t_0, \mathtt{x})$ to $\mathcal{LC}(t_1, \mathtt{x})$

Size Bounds

Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \quad \mathcal{LC}(t_1, \mathtt{x})$

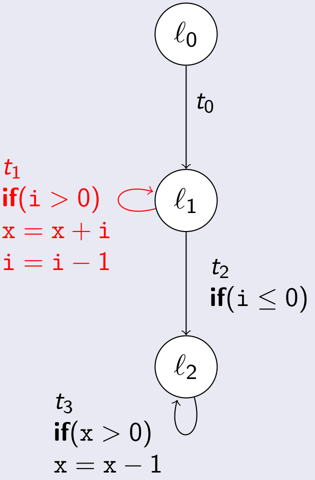
Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+ \qquad \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t

- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)

$\Rightarrow$ add size bound $\mathcal{S}(t_0, \mathtt{x})$ to $\mathcal{LC}(t_1, \mathtt{x})$



Size Bounds

Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \quad \mathcal{LC}(t_1, \mathtt{x})$

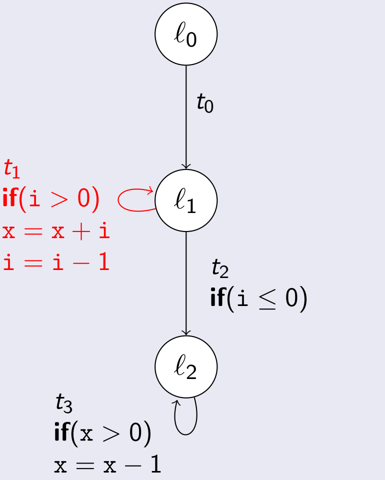
Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \qquad \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t

- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed

$\Rightarrow$ add size bound $\mathcal{S}(t_0, \mathtt{x})$ to $\mathcal{LC}(t_1, \mathtt{x})$



Size Bounds

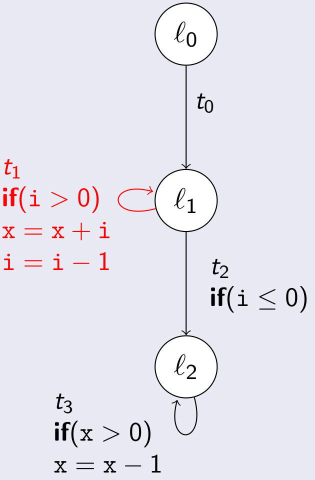
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \qquad \mathcal{LC}(t_1, \mathtt{x})$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+ \qquad \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed

$\Rightarrow$ multiply t_1's runtime bound $\mathcal{R}(t_1)$ with local change $\mathcal{LC}(t_1, \mathtt{x})$

Size Bounds

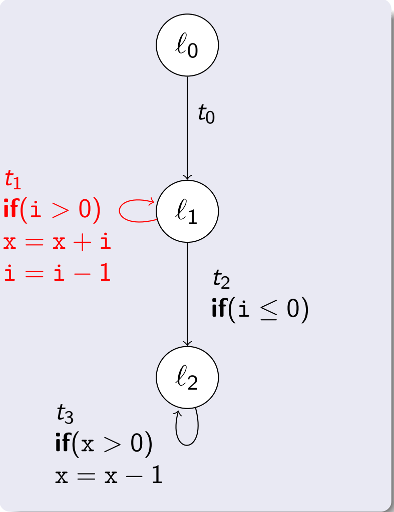
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathcal{R}(t_1) \cdot \mathcal{LC}(t_1, \mathtt{x})$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \qquad\qquad \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed

$\Rightarrow$ multiply t_1's runtime bound $\mathcal{R}(t_1)$ with local change $\mathcal{LC}(t_1, \mathtt{x})$

Size Bounds

Size bounds

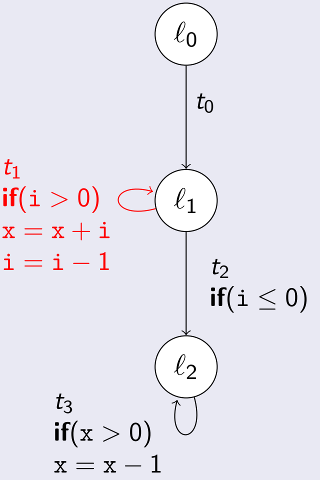
$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathcal{R}(t_1) \cdot \mathcal{LC}(t_1, \mathtt{x})$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t

use *runtime bounds* to compute *size bounds*



- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed

$\Rightarrow$ multiply t_1's runtime bound $\mathcal{R}(t_1)$ with local change $\mathcal{LC}(t_1, \mathtt{x})$

Size Bounds

Size bounds

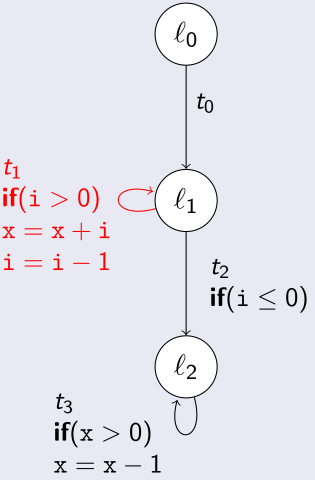
$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \quad \mathtt{i}_0 \cdot \mathcal{LC}(t_1, \mathtt{x})$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t

use *runtime bounds* to compute *size bounds*



- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed

$\Rightarrow$ multiply t_1's runtime bound $\mathcal{R}(t_1)$ with local change $\mathcal{LC}(t_1, \mathtt{x})$

Size Bounds

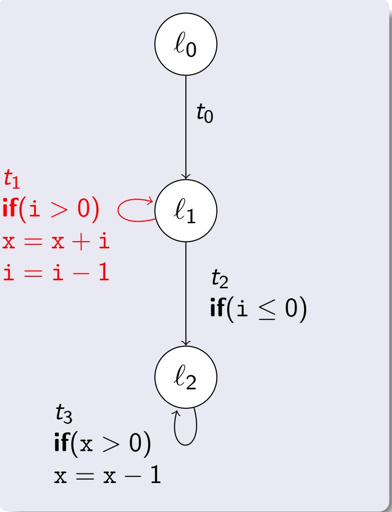
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \quad \mathtt{i}_0 \cdot \mathcal{LC}(t_1, \mathtt{x})$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed
  - consider values of $\mathcal{LC}(t_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ multiply t_1's runtime bound $\mathcal{R}(t_1)$ with local change $\mathcal{LC}(t_1, \mathtt{x})$

Size Bounds

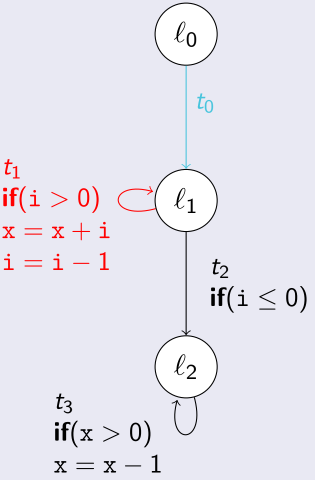
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \quad \mathtt{i}_0 \cdot \mathcal{LC}(t_1, \mathtt{x})$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed
  - consider values of $\mathcal{LC}(t_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}(t_1, \mathtt{x})$ by $\mathcal{LC}(t_1, \mathtt{x})\,[\mathtt{i}\ /\ \max(\,\mathcal{S}(t_0, \mathtt{i}),\ \mathcal{S}(t_1, \mathtt{i})\,)]$

Size Bounds

Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \quad \mathtt{i}_0 \cdot \mathcal{LC}(t_1, \mathtt{x})$

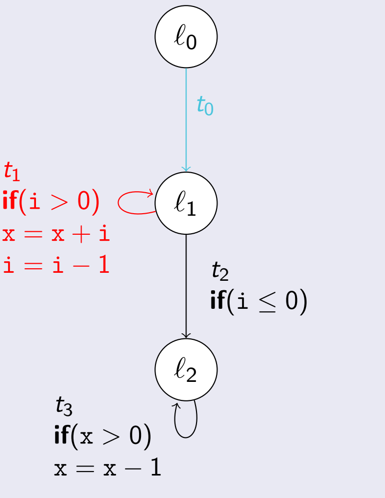
Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \; = \; \mathcal{S}(t', v) \; + \; \mathcal{R}(t) \; \cdot \; \mathcal{LC}(t, v) \, [u \, / \, \max(\, \mathcal{S}(t', u), \, \mathcal{S}(t, u) \,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t

- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed
  - consider values of $\mathcal{LC}(t_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}(t_1, \mathtt{x})$ by $\mathcal{LC}(t_1, \mathtt{x}) \, [\mathtt{i} \, / \, \max(\, \mathcal{S}(t_0, \mathtt{i}), \, \mathcal{S}(t_1, \mathtt{i}) \,)]$



Size Bounds

Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \quad \mathtt{i}_0 \cdot \mathcal{LC}(t_1, \mathtt{x})$

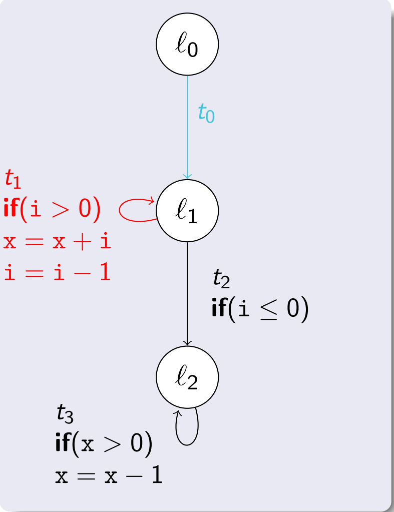
Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u), \mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t

- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed
  - consider values of $\mathcal{LC}(t_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}(t_1, \mathtt{x})$ by $\mathcal{LC}(t_1, \mathtt{x})\,[\mathtt{i} \,/\, \max(\,\mathtt{i}_0, \mathtt{i}_0\,)]$



Size Bounds

Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \quad \mathtt{i}_0 \cdot \mathcal{LC}(t_1, \mathtt{x})$

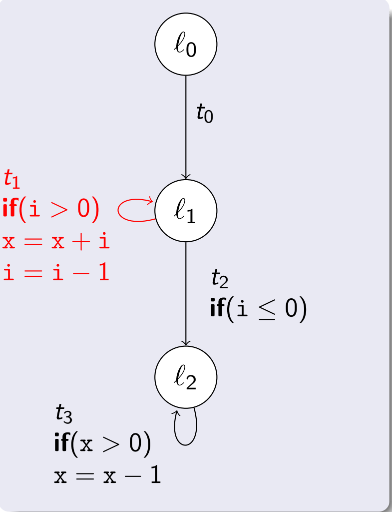
Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \; = \; \mathcal{S}(t', v) \; + \; \mathcal{R}(t) \; \cdot \; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u), \, \mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t, v)$: local change by one application of t
- t': pre-transition of t

- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed
  - consider values of $\mathcal{LC}(t_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}(t_1, \mathtt{x})$ by $\mathcal{LC}(t_1, \mathtt{x})\,[\mathtt{i} \,/\, \mathtt{i}_0]$



Size Bounds

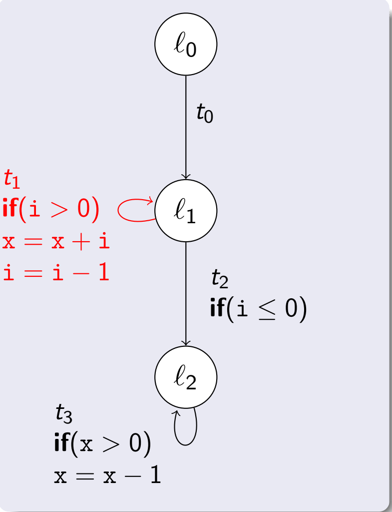
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \quad \mathtt{i}_0 \cdot \mathcal{LC}(t_1, \mathtt{x})\ [\mathtt{i} \,/\, \mathtt{i}_0]$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u), \mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed
  - consider values of $\mathcal{LC}(t_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}(t_1, \mathtt{x})$ by $\mathcal{LC}(t_1, \mathtt{x})\,[\mathtt{i} \,/\, \mathtt{i}_0]$

Size Bounds

Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \quad \mathtt{i}_0 \cdot \qquad \mathtt{i}\ [\mathtt{i} \,/\, \mathtt{i}_0]$

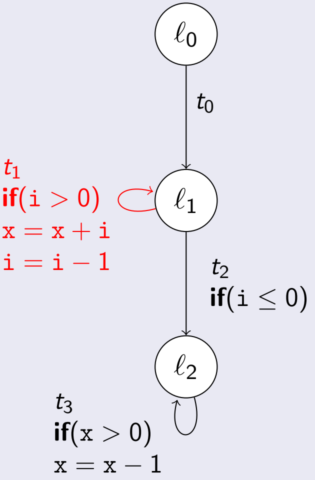
Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u),\, \mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t

- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed
  - consider values of $\mathcal{LC}(t_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}(t_1, \mathtt{x})$ by $\mathcal{LC}(t_1, \mathtt{x})\,[\mathtt{i} \,/\, \mathtt{i}_0]$



Size Bounds

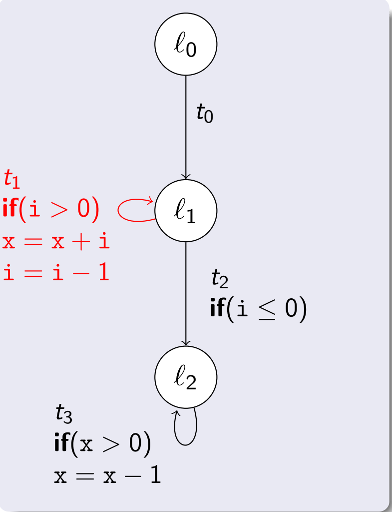
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \quad \mathtt{i}_0 \cdot \quad\quad \mathtt{i}_0$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u), \mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed
  - consider values of $\mathcal{LC}(t_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}(t_1, \mathtt{x})$ by $\mathcal{LC}(t_1, \mathtt{x})\,[\mathtt{i} \,/\, \mathtt{i}_0]$

Size Bounds

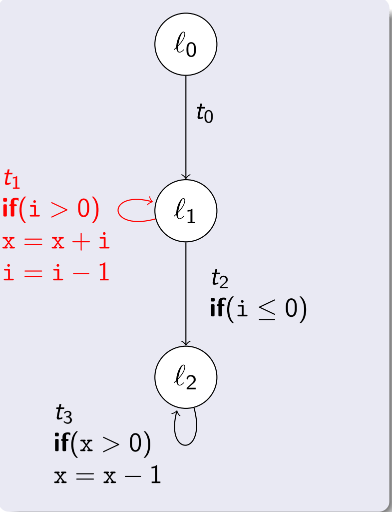
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u), \mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_1, \mathtt{x}) = \mathtt{i}$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed
  - consider values of $\mathcal{LC}(t_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}(t_1, \mathtt{x})$ by $\mathcal{LC}(t_1, \mathtt{x})\,[\mathtt{i} \,/\, \mathtt{i}_0]$

Size Bounds

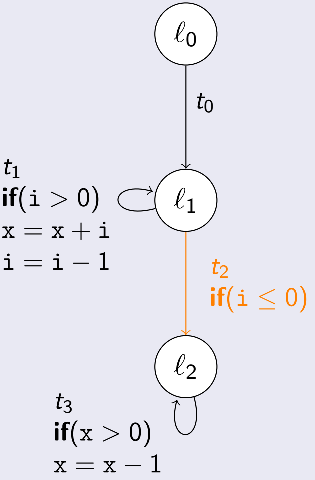
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u),\, \mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_2, \mathtt{x}) = 0$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed
  - consider values of $\mathcal{LC}(t_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}(t_1, \mathtt{x})$ by $\mathcal{LC}(t_1, \mathtt{x})\,[\mathtt{i} \,/\, \mathtt{i}_0]$

Size Bounds

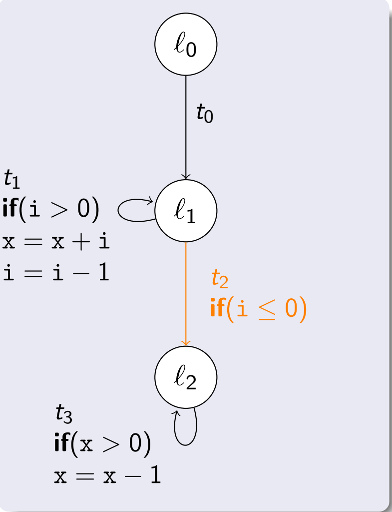
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$, $\mathcal{S}(t_2, \mathtt{x}) = \mathcal{LC}(t_2, \mathtt{x})$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u), \,\mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_2, \mathtt{x}) = 0$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_1 is executed
  - consider values of $\mathcal{LC}(t_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}(t_1, \mathtt{x})$ by $\mathcal{LC}(t_1, \mathtt{x})\,[\mathtt{i} \,/\, \mathtt{i}_0]$

Size Bounds

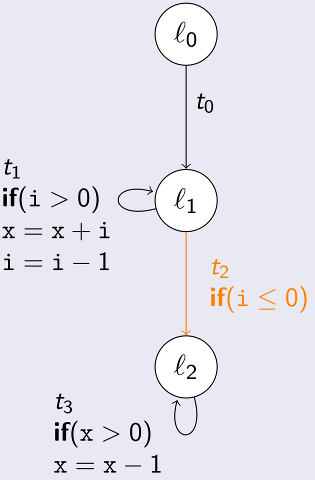
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$, $\mathcal{S}(t_2, \mathtt{x}) = \mathcal{LC}(t_2, \mathtt{x})$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u),\, \mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_2, \mathtt{x}) = 0$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_2 is executed
  - consider values of $\mathcal{LC}(t_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}(t_1, \mathtt{x})$ by $\mathcal{LC}(t_1, \mathtt{x})\,[\mathtt{i} \,/\, \mathtt{i}_0]$

Size Bounds

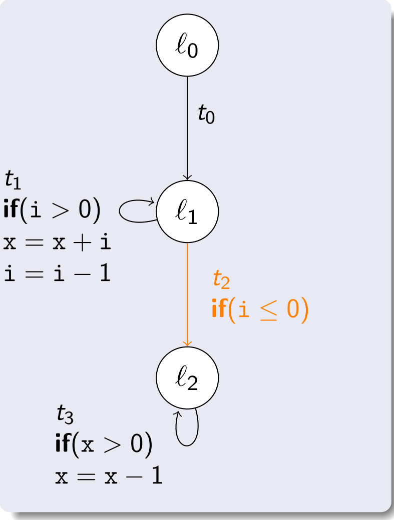
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$, $\mathcal{S}(t_2, \mathtt{x}) = \mathcal{LC}(t_2, \mathtt{x})$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u), \,\mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t, v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_2, \mathtt{x}) = 0$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_2 is executed
  - consider values of $\mathcal{LC}(t_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}(t_2, \mathtt{x})$ by $\mathcal{R}(t_2) \cdot \mathcal{LC}(t_2, \mathtt{x})[\ldots]$

Size Bounds

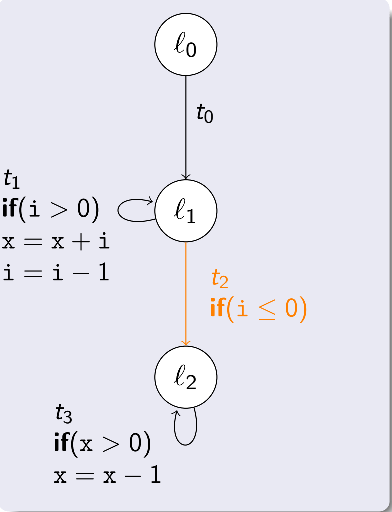
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$, $\mathcal{S}(t_2, \mathtt{x}) = \mathcal{LC}(t_2, \mathtt{x})$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u),\, \mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_2, \mathtt{x}) = 0$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_2 is executed
  - consider values of $\mathcal{LC}(t_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}(t_2, \mathtt{x})$ by 0

Size Bounds

Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$, $\mathcal{S}(t_2, \mathtt{x}) =$

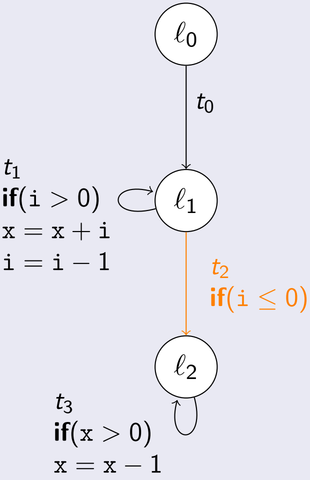
Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u),\, \mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t

- $\mathcal{LC}(t_2, \mathtt{x}) = 0$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_1 (after t_0)
  - consider how often t_2 is executed
  - consider values of $\mathcal{LC}(t_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}(t_2, \mathtt{x})$ by 0



Size Bounds

Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$, $\mathcal{S}(t_2, \mathtt{x}) =$

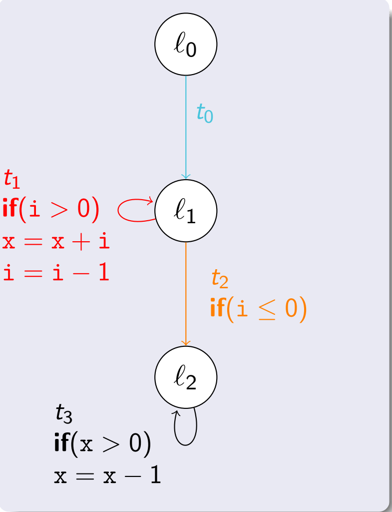
Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u),\, \mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t

- $\mathcal{LC}(t_2, \mathtt{x}) = 0$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_2 (after t_0 or t_1)
  - consider how often t_2 is executed
  - consider values of $\mathcal{LC}(t_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}(t_2, \mathtt{x})$ by 0



Size Bounds

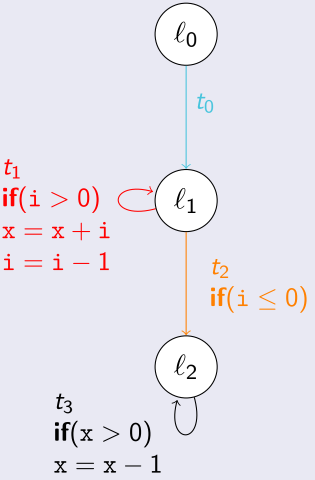
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$, $\mathcal{S}(t_2, \mathtt{x}) =$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u),\, \mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_2, \mathtt{x}) = 0$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_2 (after t_0 or t_1)
  - consider how often t_2 is executed
  - consider values of $\mathcal{LC}(t_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ add $\max(\,\mathcal{S}(t_0, \mathtt{x}),\, \mathcal{S}(t_1, \mathtt{x})\,)$ to $\mathcal{LC}(t_2, \mathtt{x})$

Size Bounds

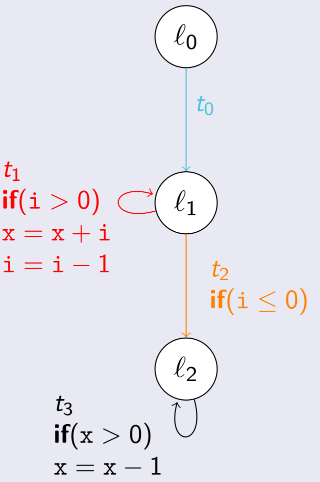
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$, $\mathcal{S}(t_2, \mathtt{x}) =$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u),\, \mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_2, \mathtt{x}) = 0$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_2 (after t_0 or t_1)
  - consider how often t_2 is executed
  - consider values of $\mathcal{LC}(t_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ add $\max(\quad \mathtt{x}_0 \quad, \mathcal{S}(t_1, \mathtt{x})\ )$ to $\mathcal{LC}(t_2, \mathtt{x})$

Size Bounds

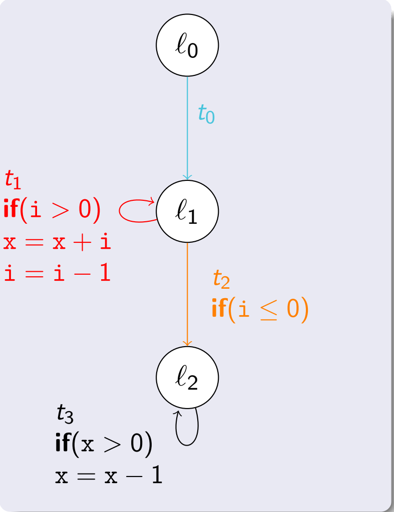
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$, $\mathcal{S}(t_2, \mathtt{x}) =$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \; = \; \mathcal{S}(t', v) \; + \; \mathcal{R}(t) \; \cdot \; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u), \, \mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_2, \mathtt{x}) = 0$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_2 (after t_0 or t_1)
  - consider how often t_2 is executed
  - consider values of $\mathcal{LC}(t_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ add $\max(\quad \mathtt{x}_0 \quad , \; \mathtt{x}_0 + \mathtt{i}_0^2 \;)$ to $\mathcal{LC}(t_2, \mathtt{x})$

Size Bounds

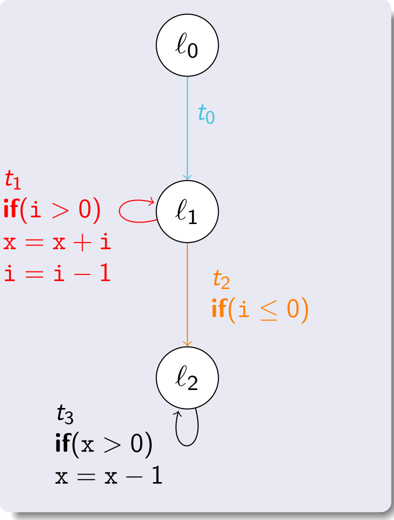
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$, $\mathcal{S}(t_2, \mathtt{x}) =$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u\,/\,\max(\,\mathcal{S}(t', u), \mathcal{S}(t, u)\,)]$$

- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



- $\mathcal{LC}(t_2, \mathtt{x}) = 0$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_2 (after t_0 or t_1)
  - consider how often t_2 is executed
  - consider values of $\mathcal{LC}(t_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ add $\mathtt{x}_0 + \mathtt{i}_0^2$ to $\mathcal{LC}(t_2, \mathtt{x})$

Size Bounds

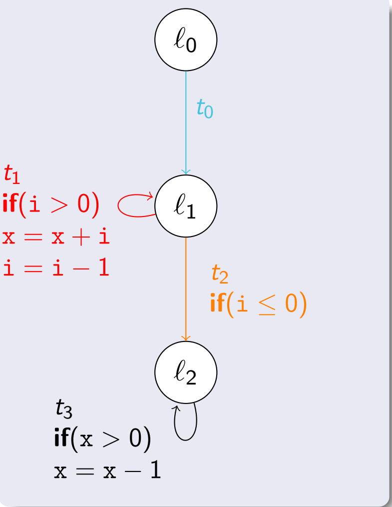
Size bounds

$\mathcal{S}(t_0, v) = v_0$, $\mathcal{S}(t_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}(t_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$, $\mathcal{S}(t_2, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$

Computing size bound for variable v after transition t

$$\mathcal{S}(t, v) \;=\; \mathcal{S}(t', v) \;+\; \mathcal{R}(t) \;\cdot\; \mathcal{LC}(t, v)\,[u \,/\, \max(\,\mathcal{S}(t', u), \,\mathcal{S}(t, u)\,)]$$

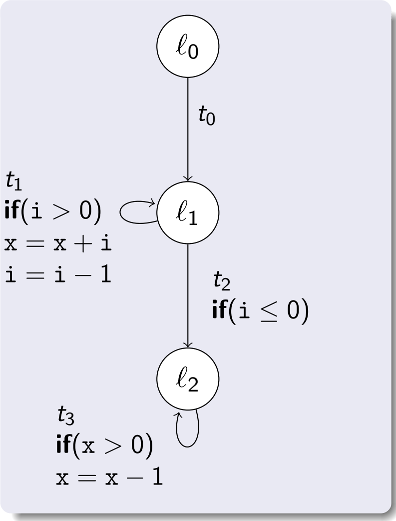
- $\mathcal{LC}(t,v)$: local change by one application of t
- t': pre-transition of t



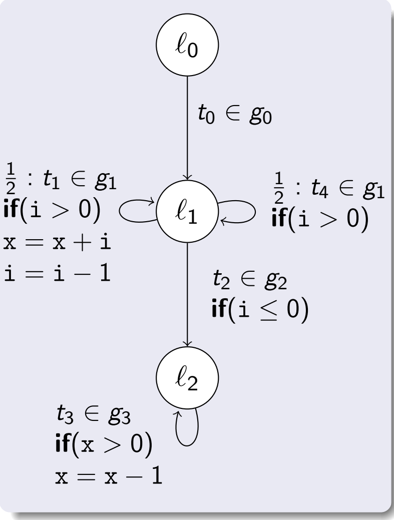
- $\mathcal{LC}(t_2, \mathtt{x}) = 0$
- For global result:
  - consider value of $\mathtt{x}$ before reaching t_2 (after t_0 or t_1)
  - consider how often t_2 is executed
  - consider values of $\mathcal{LC}(t_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ add $\mathtt{x}_0 + \mathtt{i}_0^2$ to $\mathcal{LC}(t_2, \mathtt{x})$

Expected Size Bounds



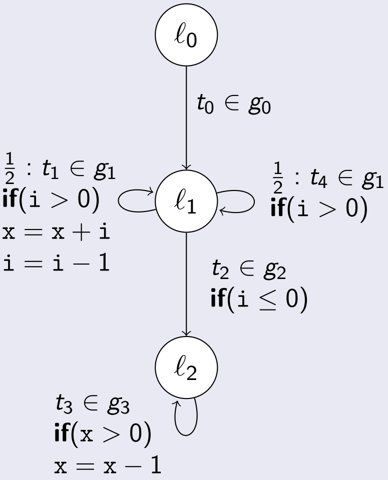
Expected Size Bounds



Expected Size Bounds

Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$



Expected Size Bounds

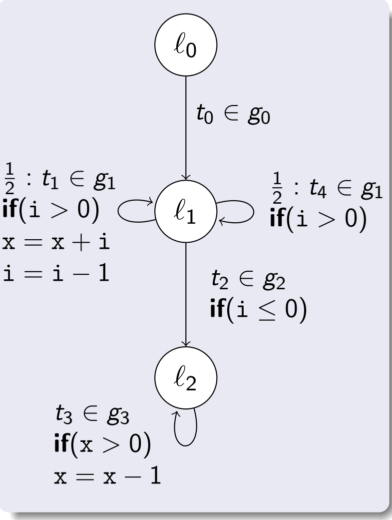
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$

Computing size bound for variable v after transition g

$$\mathcal{S}\ (g, v) = \mathcal{S}(g', v) + \mathcal{R}(g) \cdot \mathcal{LC}\ (g, v)\,[u/\max(\mathcal{S}(g', u), \ldots)]$$

- $\mathcal{LC}\ (g, v)$: local change by g
- g' : pre-transition of g



Expected Size Bounds

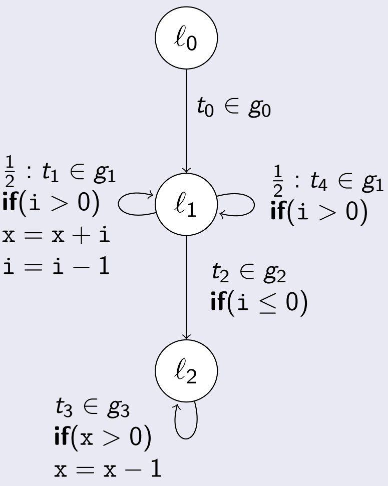
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}\ (g, v) = \quad \mathcal{S}(g', v) \ + \quad \mathcal{R}(g) \cdot \mathcal{LC}\ (g, v)\,[u/\max(\mathcal{S}(g', u), \ldots)]$$

- $\mathcal{LC}\ (g, v)$: local change by g
- g' : pre-transition of g



Expected Size Bounds

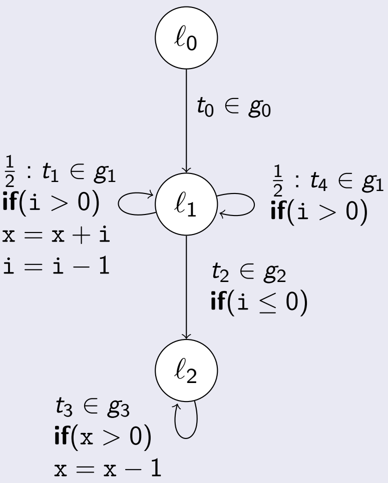
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$

Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathbb{E}(\, \mathcal{S}(g', v) + \mathcal{R}(g) \cdot \mathcal{LC}(g, v)\,[u/\max(\mathcal{S}(g', u), \ldots)]\,)$$

- $\mathcal{LC}(g,v)$: local change by g
- g' : pre-transition of g



Expected Size Bounds

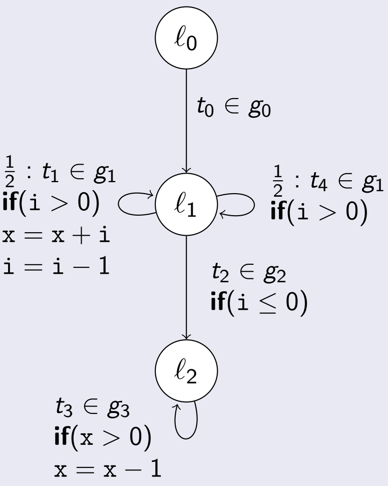
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) \ = \ \mathbb{E}(\ \mathcal{S}(g', v)\ ) + \mathbb{E}(\ \mathcal{R}(g) \cdot \mathcal{LC}\ \ (g, v)\,[u/\max(\mathcal{S}(g', u), \ldots)]\ )$$

- $\mathcal{LC}\ \ (g,v)$: local change by g
- g' : pre-transition of g



Expected Size Bounds

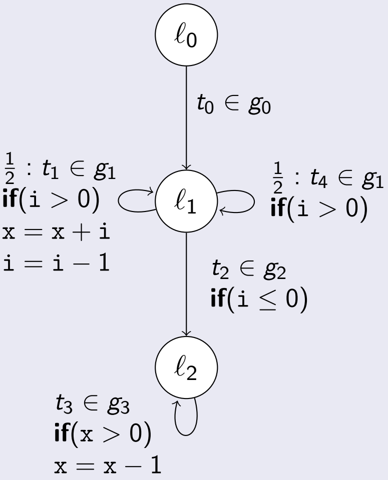
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$

Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathbb{E}(\mathcal{R}(g) \cdot \mathcal{LC}(g, v)\,[u/\max(\mathcal{S}(g', u), \ldots)])$$

- $\mathcal{LC}(g,v)$: local change by g
- g' : pre-transition of g



Expected Size Bounds

Expected size bounds

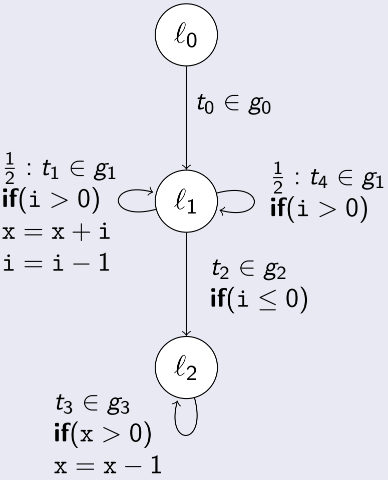
$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathbb{E}(\mathcal{R}(g) \cdot \mathcal{LC}(g, v)[u/\max(\mathcal{S}(g', u), \dots)])$$

- $\mathcal{LC}(g, v)$: local change by g
- g' : pre-transition of g

Expected value *not* multiplicative!



Expected Size Bounds

Expected size bounds

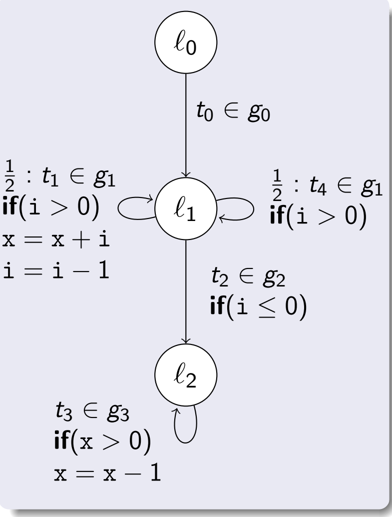
$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathbb{E}(\mathcal{R}(g) \cdot \mathcal{LC}(g, v)[u / \max(\mathcal{S}(g', u), \ldots)])$$

- $\mathcal{LC}$ (g,v): local change by g
- g' : pre-transition of g

Expected value *not* multiplicative!
But: $\mathcal{LC}$ independent of runtime



Expected Size Bounds

Expected size bounds

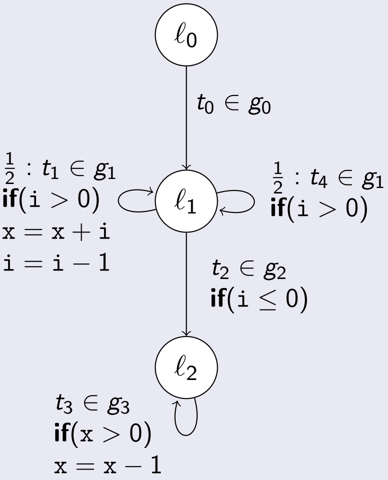
$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \quad \mathcal{S}_{\mathbb{E}}(g', v) \;+\; \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g,v)$: *expect.* local change by g
- g', t': pre-transition of g

Expected value *not* multiplicative!
But: $\mathcal{LC}$ independent of runtime



Expected Size Bounds

Expected size bounds

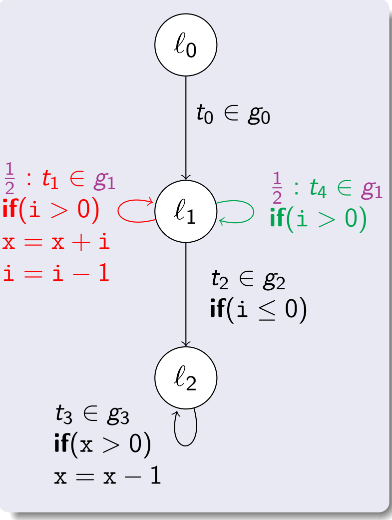
$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) =$ $\frac{\mathtt{i}}{2}$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g,v)$: *expect.* local change by g
- g', t': pre-transition of g

- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$



Expected Size Bounds

Expected size bounds

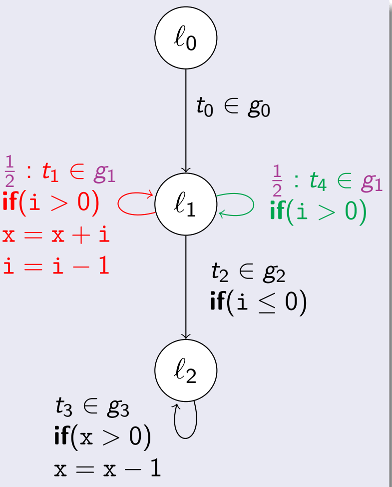
$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) =$ $\frac{\mathtt{i}}{2}$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g,v)$: *expect.* local change by g
- g', t': pre-transition of g

- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$
- For global result:



Expected Size Bounds

Expected size bounds

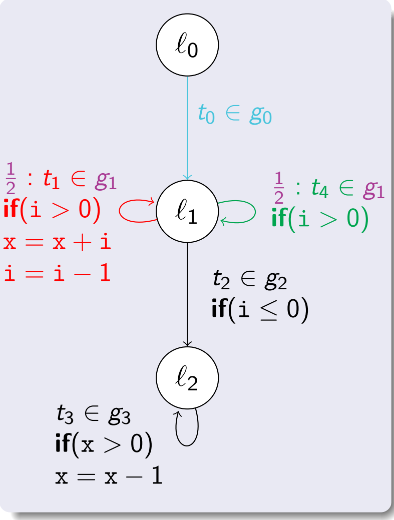
$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \qquad \frac{\mathtt{i}}{2}$

Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g, v)$: *expect.* local change by g
- g', t': pre-transition of g

- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_1 (after g_0)



Expected Size Bounds

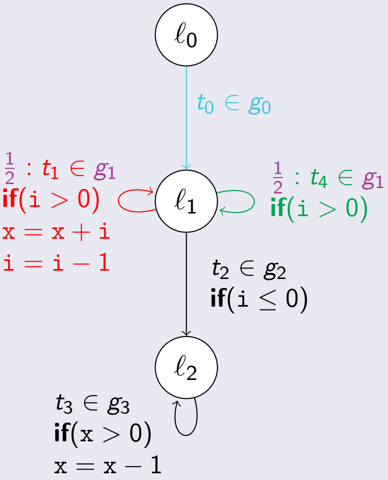
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \quad \frac{\mathtt{i}}{2}$

Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g, v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_1 (after g_0)

$\Rightarrow$ add *expected* size bound $\mathcal{S}_{\mathbb{E}}(g_0, \mathtt{x})$ to $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$

Expected Size Bounds

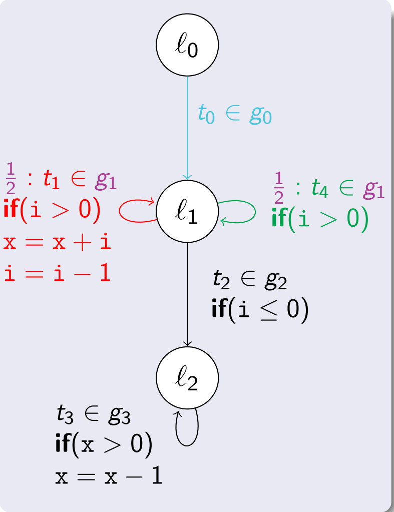
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathcal{S}_{\mathbb{E}}(g_0, \mathtt{x}) + \quad \frac{\mathtt{i}}{2}$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \dots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g,v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_1 (after g_0)

$\Rightarrow$ add *expected* size bound $\mathcal{S}_{\mathbb{E}}(g_0, \mathtt{x})$ to $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$

Expected Size Bounds

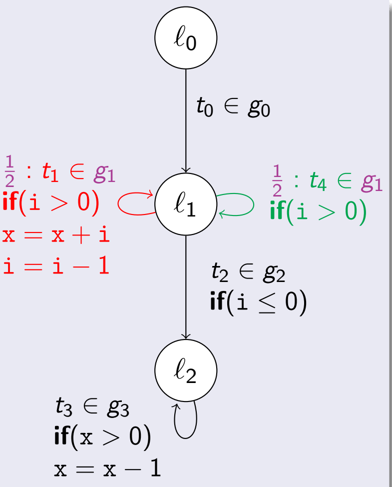
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + \qquad \frac{\mathtt{i}}{2}$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g, v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_1 (after g_0)

$\Rightarrow$ add *expected* size bound $\mathcal{S}_{\mathbb{E}}(g_0, \mathtt{x})$ to $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$

Expected Size Bounds

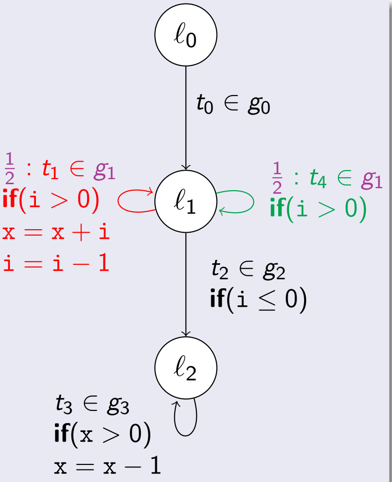
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + \qquad \frac{\mathtt{i}}{2}$

Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g, v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_1 (after g_0)
  - consider how often g_1 is *expected* to be executed

$\Rightarrow$ add *expected* size bound $\mathcal{S}_{\mathbb{E}}(g_0, \mathtt{x})$ to $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$

Expected Size Bounds

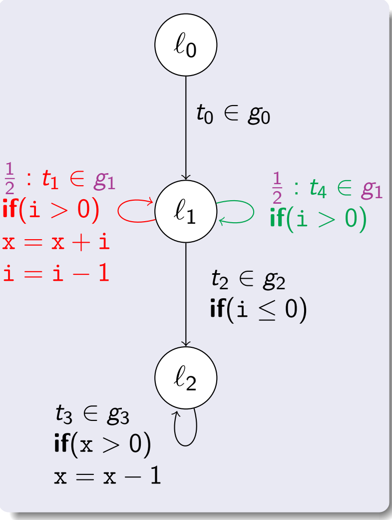
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + \qquad \frac{\mathtt{i}}{2}$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g,v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_1 (after g_0)
  - consider how often g_1 is *expected* to be executed

$\Rightarrow$ multiply g_1's *expected* runtime bound $\mathcal{R}_{\mathbb{E}}(g_1)$ with local change $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$

Expected Size Bounds

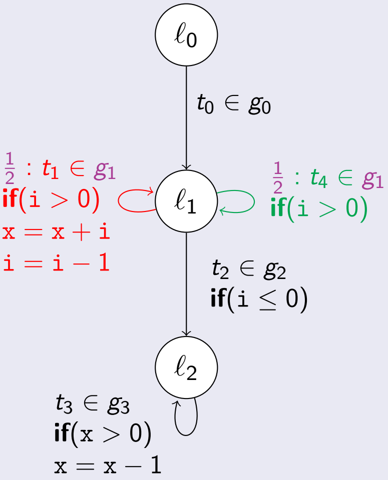
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + \mathcal{R}_{\mathbb{E}}(g_1) \cdot \frac{\mathtt{i}}{2}$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g,v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_1 (after g_0)
  - consider how often g_1 is *expected* to be executed

$\Rightarrow$ multiply g_1's *expected* runtime bound $\mathcal{R}_{\mathbb{E}}(g_1)$ with local change $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$

Expected Size Bounds

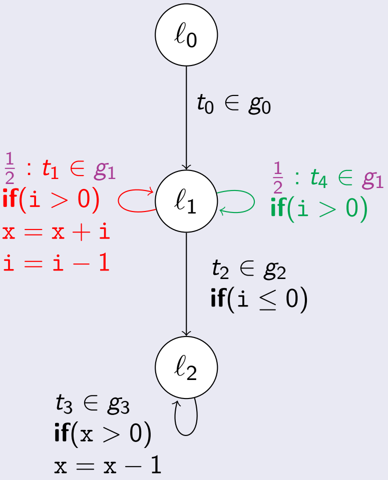
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + 2 \cdot \mathtt{i}_0 \cdot \frac{\mathtt{i}}{2}$

Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g,v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_1 (after g_0)
  - consider how often g_1 is *expected* to be executed

$\Rightarrow$ multiply g_1's *expected* runtime bound $\mathcal{R}_{\mathbb{E}}(g_1)$ with local change $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$

Expected Size Bounds

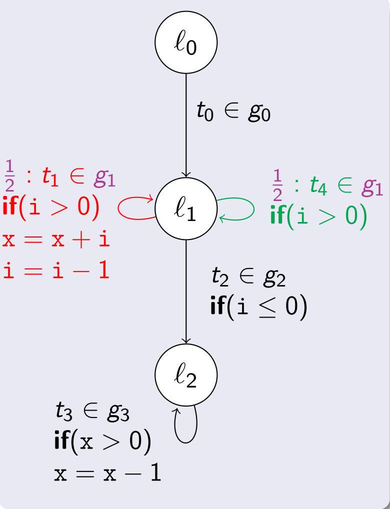
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + 2 \cdot \mathtt{i}_0 \cdot \frac{\mathtt{i}}{2}$

Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g,v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_1 (after g_0)
  - consider how often g_1 is *expected* to be executed
  - consider values of $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ multiply g_1's *expected* runtime bound $\mathcal{R}_{\mathbb{E}}(g_1)$ with local change $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$

Expected Size Bounds

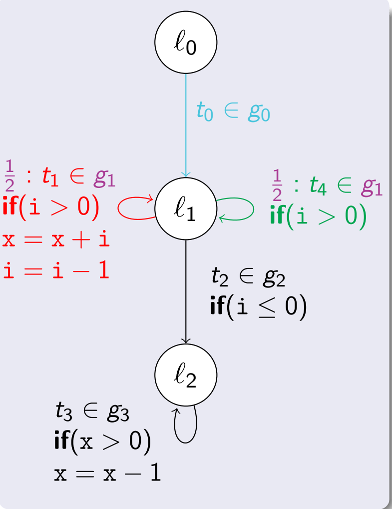
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + 2 \cdot \mathtt{i}_0 \cdot \frac{\mathtt{i}}{2}$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g,v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_1 (after g_0)
  - consider how often g_1 is *expected* to be executed
  - consider values of $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$ by $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})\,[\mathtt{i}\,/\,\max(\,\mathcal{S}(t_0, \mathtt{i}), \mathcal{S}(t_1, \mathtt{i}), \mathcal{S}(t_4, \mathtt{i})\,)\,]$

Expected Size Bounds

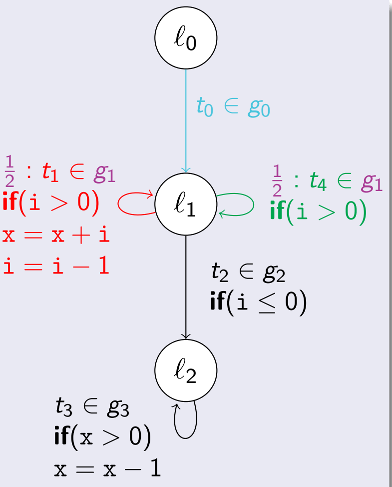
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + \ 2 \cdot \mathtt{i}_0 \cdot \ \frac{\mathtt{i}}{2}$

Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \quad \mathcal{S}_{\mathbb{E}}(g', v) \ + \ \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g,v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_1 (after g_0)
  - consider how often g_1 is *expected* to be executed
  - consider values of $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$ by $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})\,[\mathtt{i}\ /\ \max(\ \mathtt{i}_0, \mathtt{i}_0, \mathtt{i}_0\ )\,]$

Expected Size Bounds

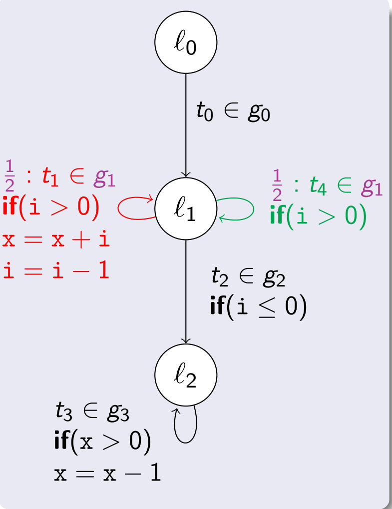
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + 2 \cdot \mathtt{i}_0 \cdot \frac{\mathtt{i}}{2}$

Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g, v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$
- For global result:
  - consider *expected* value of x before reaching g_1 (after g_0)
  - consider how often g_1 is *expected* to be executed
  - consider values of $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$ by $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})\,[\mathtt{i} \,/\, \mathtt{i}_0\,]$

Expected Size Bounds

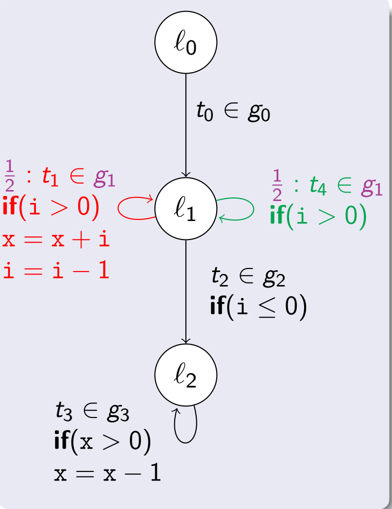
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + 2 \cdot \mathtt{i}_0 \cdot \frac{\mathtt{i}}{2}\ [\mathtt{i} / \mathtt{i}_0]$

Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u / \max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g, v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_1 (after g_0)
  - consider how often g_1 is *expected* to be executed
  - consider values of $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$ by $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})\,[\mathtt{i} / \mathtt{i}_0]$

Expected Size Bounds

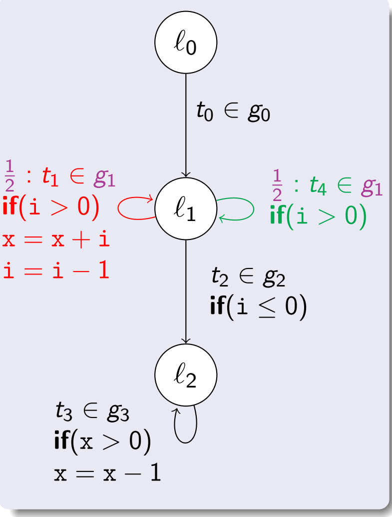
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + \; 2 \cdot \mathtt{i}_0 \cdot \; \frac{\mathtt{i}_0}{2}$

Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \quad \mathcal{S}_{\mathbb{E}}(g', v) \; + \; \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g,v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x}) = \frac{\mathtt{i}}{2}$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_1 (after g_0)
  - consider how often g_1 is *expected* to be executed
  - consider values of $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$ by $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})\,[\mathtt{i} \,/\, \mathtt{i}_0\,]$

Expected Size Bounds

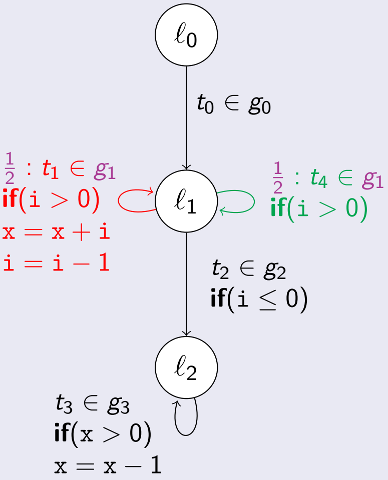
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \texttt{i}) = \texttt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \texttt{x}) = \texttt{x}_0 + \texttt{i}_0^2$

Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u / \max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g, v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_1, \texttt{x}) = \frac{\texttt{i}}{2}$
- For global result:
  - consider *expected* value of $\texttt{x}$ before reaching g_1 (after g_0)
  - consider how often g_1 is *expected* to be executed
  - consider values of $\mathcal{LC}_{\mathbb{E}}(g_1, \texttt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}_{\mathbb{E}}(g_1, \texttt{x})$ by $\mathcal{LC}_{\mathbb{E}}(g_1, \texttt{x})\,[\texttt{i} \,/\, \texttt{i}_0\,]$

Expected Size Bounds

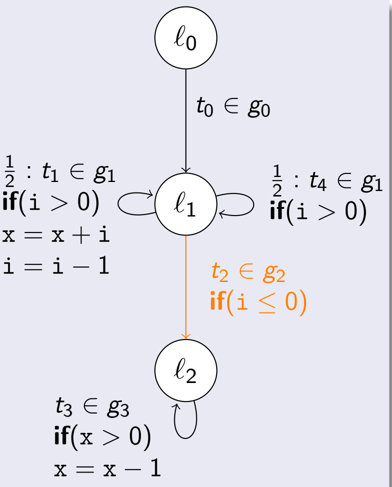
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \quad \mathcal{S}_{\mathbb{E}}(g', v) \ + \ \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g, v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x}) = 0$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_1 (after g_0)
  - consider how often g_1 is *expected* to be executed
  - consider values of $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$ by $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})\,[\mathtt{i} \,/\, \mathtt{i}_0\,]$

Expected Size Bounds

Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$

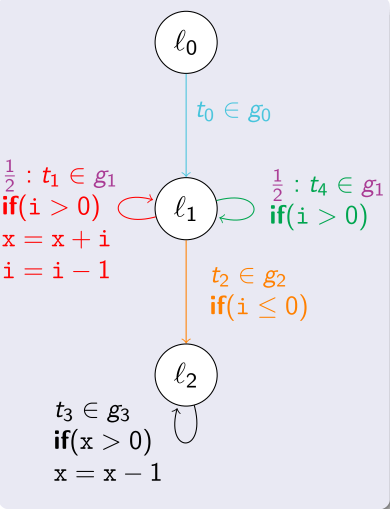
Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u / \max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g,v)$: *expect.* local change by g
- g', t': pre-transition of g

- $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x}) = 0$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_2 (after g_0 or g_1)
  - consider how often g_2 is *expected* to be executed
  - consider values of $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ replace $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})$ by $\mathcal{LC}_{\mathbb{E}}(g_1, \mathtt{x})\,[\mathtt{i} / \mathtt{i}_0]$



Expected Size Bounds

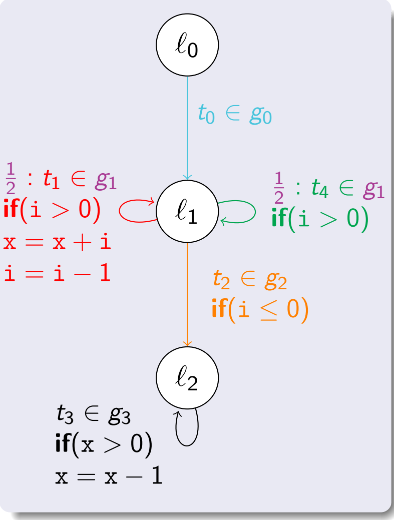
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$

Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g,v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x}) = 0$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_2 (after g_0 or g_1)
  - consider how often g_2 is *expected* to be executed
  - consider values of $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ add $\max(\ \mathcal{S}_{\mathbb{E}}(g_0, \mathtt{x}),\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x})\ )$ to $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x})$

Expected Size Bounds

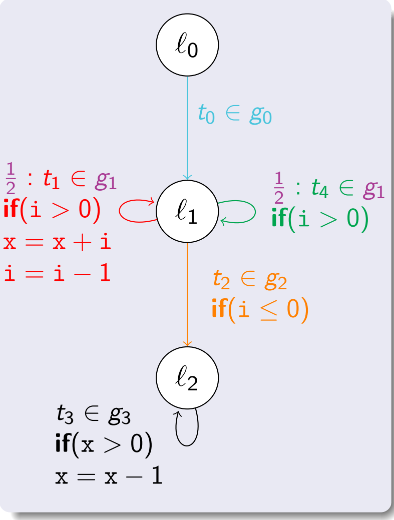
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$

Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g, v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x}) = 0$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_2 (after g_0 or g_1)
  - consider how often g_2 is *expected* to be executed
  - consider values of $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ add $\max(\ \mathtt{x}_0\ ,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x})\ )$ to $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x})$

Expected Size Bounds

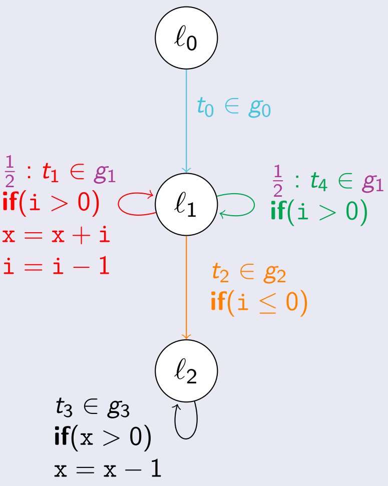
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0$, $\mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g, v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x}) = 0$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_2 (after g_0 or g_1)
  - consider how often g_2 is *expected* to be executed
  - consider values of $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ add $\max(\ \mathtt{x}_0\ ,\ \mathtt{x}_0 + \mathtt{i}_0^2\ )$ to $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x})$

Expected Size Bounds

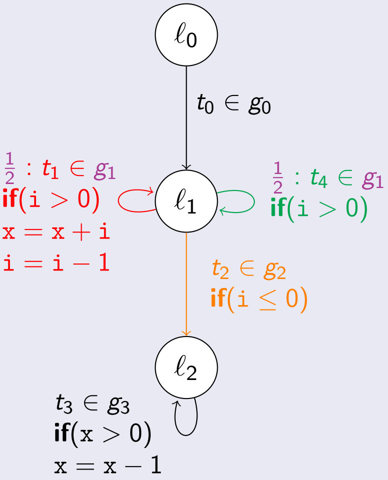
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \dots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g,v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x}) = 0$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_2 (after g_0 or g_1)
  - consider how often g_2 is *expected* to be executed
  - consider values of $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ add $\mathtt{x}_0 + \mathtt{i}_0^2$ to $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x})$

Expected Size Bounds

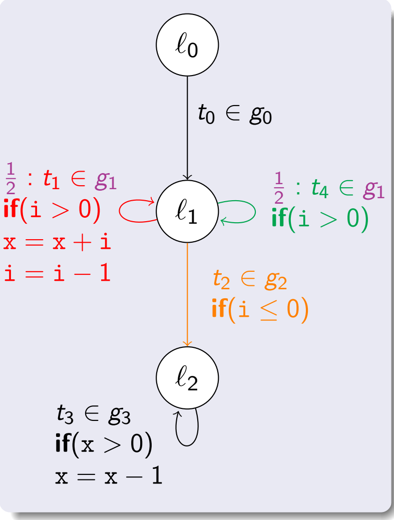
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2$

Computing _expected_ size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g, v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x}) = 0$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_2 (after g_0 or g_1)
  - consider how often g_2 is *expected* to be executed
  - consider values of $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ add $\mathtt{x}_0 + \mathtt{i}_0^2$ to 0

Expected Size Bounds

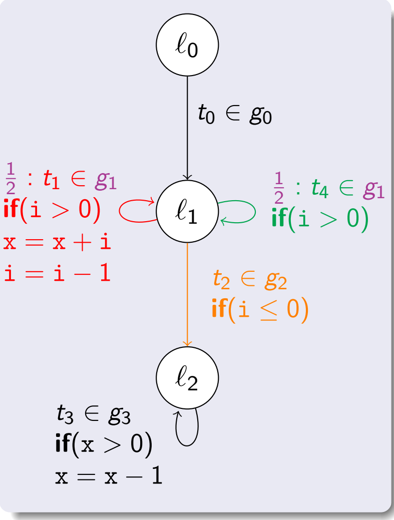
Expected size bounds

$\mathcal{S}_{\mathbb{E}}(g_0, v) = v_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{i}) = \mathtt{i}_0,\ \mathcal{S}_{\mathbb{E}}(g_1, \mathtt{x}) = \mathtt{x}_0 + \mathtt{i}_0^2 = \mathcal{S}_{\mathbb{E}}(g_2, \mathtt{x})$

Computing *expected* size bound for variable v after transition g

$$\mathcal{S}_{\mathbb{E}}(g, v) = \mathcal{S}_{\mathbb{E}}(g', v) + \mathcal{R}_{\mathbb{E}}(g) \cdot \mathcal{LC}_{\mathbb{E}}(g, v)\,[u/\max(\mathcal{S}(t', u), \ldots)]$$

- $\mathcal{LC}_{\mathbb{E}}(g,v)$: *expect.* local change by g
- g', t': pre-transition of g



- $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x}) = 0$
- For global result:
  - consider *expected* value of $\mathtt{x}$ before reaching g_2 (after g_0 or g_1)
  - consider how often g_2 is *expected* to be executed
  - consider values of $\mathcal{LC}_{\mathbb{E}}(g_2, \mathtt{x})$'s variables in full run

$\Rightarrow$ add $\mathtt{x}_0 + \mathtt{i}_0^2$ to 0

Inferring *Expected* Runtimes of *Probabilistic* Programs

- Implementation in `KoAT`

Inferring *Expected* Runtimes of *Probabilistic* Programs

- Implementation in `KoAT`
  - all 46 benchmarks from `Absynth`

Inferring *Expected* Runtimes of *Probabilistic* Programs

- Implementation in `KoAT`
  - all 46 benchmarks from `Absynth`
  - 29 new benchmarks including examples from TPDB enriched with randomization

Inferring *Expected* Runtimes of *Probabilistic* Programs

- Implementation in `KoAT`
  - all 46 benchmarks from `Absynth`
  - 29 new benchmarks including examples from TPDB enriched with randomization
  - timeout of 5 minutes

Inferring *Expected* Runtimes of *Probabilistic* Programs

Bound	KoAT	Absynth	eco-imp
$\mathcal{O}(1)$	8	7	8
$\mathcal{O}(n)$	42	35	35
$\mathcal{O}(n^2)$	15	9	15
$\mathcal{O}(n^{>2})$	2	0	0
EXP	1	0	0
∞	7	15	14
TO	0	9	3

- Implementation in KoAT
  - all 46 benchmarks from Absynth
  - 29 new benchmarks including examples from TPDB enriched with randomization
  - timeout of 5 minutes

Inferring *Expected* Runtimes of *Probabilistic* Programs

- Implementation in KoAT
  - all 46 benchmarks from Absynth
  - 29 new benchmarks including examples from TPDB enriched with randomization
  - timeout of 5 minutes

Bound	KoAT	Absynth	eco-imp
$\mathcal{O}(1)$	8	7	8
$\mathcal{O}(n)$	42	35	35
$\mathcal{O}(n^2)$	15	9	15
$\mathcal{O}(n^{>2})$	2	0	0
EXP	1	0	0
∞	7	15	14
TO	0	9	3
Avg. Time	4.26 s	3.53 s	0.93 s

Inferring *Expected* Runtimes of *Probabilistic* Programs

- Implementation in KoAT
  - all 46 benchmarks from Absynth
  - 29 new benchmarks including examples from TPDB enriched with randomization
  - timeout of 5 minutes

Bound	KoAT	Absynth	eco-imp
$\mathcal{O}(1)$	8	7	8
$\mathcal{O}(n)$	42	35	35
$\mathcal{O}(n^2)$	15	9	15
$\mathcal{O}(n^{>2})$	2	0	0
EXP	1	0	0
∞	7	15	14
TO	0	9	3
Avg. Time	4.26 s	3.53 s	0.93 s
Success	91 %	68 %	77 %

Inferring *Expected* Runtimes of *Probabilistic* Programs

- alternate finding expected runtime and size bounds

- Implementation in KoAT
  - all 46 benchmarks from Absynth
  - 29 new benchmarks including examples from TPDB enriched with randomization
  - timeout of 5 minutes

Bound	KoAT	Absynth	eco-imp
$\mathcal{O}(1)$	8	7	8
$\mathcal{O}(n)$	42	35	35
$\mathcal{O}(n^2)$	15	9	15
$\mathcal{O}(n^{>2})$	2	0	0
EXP	1	0	0
∞	7	15	14
TO	0	9	3
Avg. Time	4.26 s	3.53 s	0.93 s
Success	91 %	68 %	77 %

Inferring *Expected* Runtimes of *Probabilistic* Programs

- alternate finding expected runtime and size bounds
  - compute size bounds by combining local change bound with runtime bounds

- Implementation in `KoAT`
  - all 46 benchmarks from `Absynth`
  - 29 new benchmarks including examples from TPDB enriched with randomization
  - timeout of 5 minutes

Bound	`KoAT`	`Absynth`	`eco-imp`
$\mathcal{O}(1)$	8	7	8
$\mathcal{O}(n)$	42	35	35
$\mathcal{O}(n^2)$	15	9	15
$\mathcal{O}(n^{>2})$	2	0	0
EXP	1	0	0
∞	7	15	14
TO	0	9	3
Avg. Time	4.26 s	3.53 s	0.93 s
Success	91 %	68 %	77 %

Inferring *Expected* Runtimes of *Probabilistic* Programs

- alternate finding expected runtime and size bounds
  - compute size bounds by combining local change bound with runtime bounds
  - compute runtime bounds for program parts based on size bounds for preceding parts

- Implementation in KoAT
  - all 46 benchmarks from Absynth
  - 29 new benchmarks including examples from TPDB enriched with randomization
  - timeout of 5 minutes

Bound	KoAT	Absynth	eco-imp
$\mathcal{O}(1)$	8	7	8
$\mathcal{O}(n)$	42	35	35
$\mathcal{O}(n^2)$	15	9	15
$\mathcal{O}(n^{>2})$	2	0	0
EXP	1	0	0
∞	7	15	14
TO	0	9	3
Avg. Time	4.26 s	3.53 s	0.93 s
Success	91 %	68 %	77 %

Inferring *Expected* Runtimes of *Probabilistic* Programs

- alternate finding expected runtime and size bounds
  - compute size bounds by combining local change bound with runtime bounds
  - compute runtime bounds for program parts based on size bounds for preceding parts
  - based on both *expected* and *non-probabilistic* bounds for program parts

- Implementation in `KoAT`
  - all 46 benchmarks from `Absynth`
  - 29 new benchmarks including examples from TPDB enriched with randomization
  - timeout of 5 minutes

Bound	`KoAT`	`Absynth`	`eco-imp`
$\mathcal{O}(1)$	8	7	8
$\mathcal{O}(n)$	42	35	35
$\mathcal{O}(n^2)$	15	9	15
$\mathcal{O}(n^{>2})$	2	0	0
EXP	1	0	0
∞	7	15	14
TO	0	9	3
Avg. Time	4.26 s	3.53 s	0.93 s
Success	91 %	68 %	77 %

Inferring *Expected* Runtimes of *Probabilistic* Programs

- alternate finding expected runtime and size bounds
  - compute size bounds by combining local change bound with runtime bounds
  - compute runtime bounds for program parts based on size bounds for preceding parts
  - based on both *expected* and *non-probabilistic* bounds for program parts
- **modular:** only consider small program parts at a time
- Implementation in `KoAT`
  - all 46 benchmarks from `Absynth`
  - 29 new benchmarks including examples from TPDB enriched with randomization
  - timeout of 5 minutes

Bound	`KoAT`	`Absynth`	`eco-imp`
$\mathcal{O}(1)$	8	7	8
$\mathcal{O}(n)$	42	35	35
$\mathcal{O}(n^2)$	15	9	15
$\mathcal{O}(n^{>2})$	2	0	0
EXP	1	0	0
∞	7	15	14
TO	0	9	3
Avg. Time	4.26 s	3.53 s	0.93 s
Success	91 %	68 %	77 %

Inferring *Expected* Runtimes of *Probabilistic* Programs

- alternate finding expected runtime and size bounds
  - compute size bounds by combining local change bound with runtime bounds
  - compute runtime bounds for program parts based on size bounds for preceding parts
  - based on both *expected* and *non-probabilistic* bounds for program parts

- **modular:** only consider small program parts at a time
  - linear probabilistic ranking functions

- Implementation in `KoAT`
  - all 46 benchmarks from `Absynth`
  - 29 new benchmarks including examples from TPDB enriched with randomization
  - timeout of 5 minutes

Bound	`KoAT`	`Absynth`	`eco-imp`
$\mathcal{O}(1)$	8	7	8
$\mathcal{O}(n)$	42	35	35
$\mathcal{O}(n^2)$	15	9	15
$\mathcal{O}(n^{>2})$	2	0	0
EXP	1	0	0
∞	7	15	14
TO	0	9	3
Avg. Time	4.26 s	3.53 s	0.93 s
Success	91 %	68 %	77 %

Inferring *Expected* Runtimes of *Probabilistic* Programs

- alternate finding expected runtime and size bounds
  - compute size bounds by combining local change bound with runtime bounds
  - compute runtime bounds for program parts based on size bounds for preceding parts
  - based on both *expected* and *non-probabilistic* bounds for program parts

- **modular:** only consider small program parts at a time
  - linear probabilistic ranking functions
  - approach scales to larger programs

- Implementation in `KoAT`
  - all 46 benchmarks from `Absynth`
  - 29 new benchmarks including examples from TPDB enriched with randomization
  - timeout of 5 minutes

Bound	`KoAT`	`Absynth`	`eco-imp`
$\mathcal{O}(1)$	8	7	8
$\mathcal{O}(n)$	42	35	35
$\mathcal{O}(n^2)$	15	9	15
$\mathcal{O}(n^{>2})$	2	0	0
EXP	1	0	0
∞	7	15	14
TO	0	9	3
Avg. Time	4.26 s	3.53 s	0.93 s
Success	91 %	68 %	77 %