Synthesizing Ranking Functions from Bits and Pieces

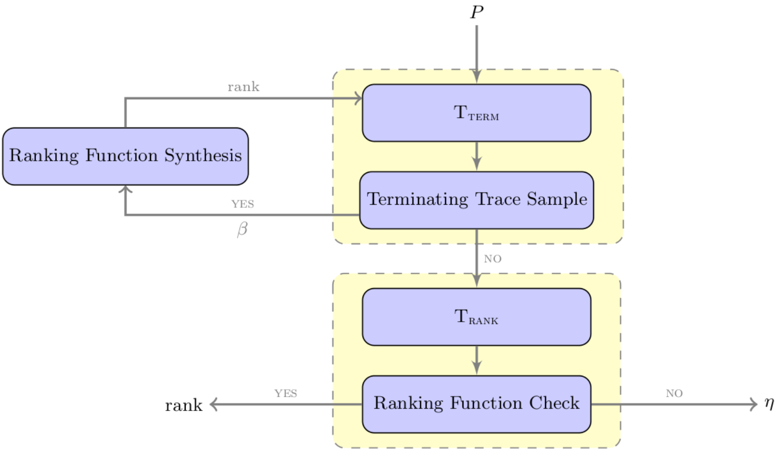
Caterina Urban et al.

February 1, 2021

Overview

- Synthesizing ranking function to prove termination based on safety checking.
- Bits: **bits** of information from terminating executions. Pieces: Extrapolate from these bits to obtain ranking functions **pieces** and combine them into ranking functions.
- Algorithm implemented in SeaHorn targeting on C code.

Overview



P
rank
T_{TERM}
Ranking Function Synthesis
Terminating Trace Sample
YES
β
NO
T RANK
YES
rank
Ranking Function Check
NO
η

Preliminaries

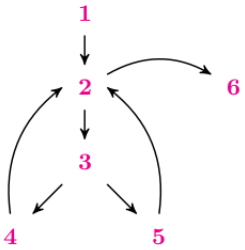
- Transition System: $\langle \Sigma, \tau \rangle$: states and transition relation.
- $s \in \Sigma$ is a pair $\langle l, \bar{x} \rangle$ where l is program control point and $\bar{x}$ is the vector of values of variables.
- A state is *terminating*, *non-terminating*, *potentially terminating* and *potentially nonterminating*.
- Ranking function: $\forall s, s'.\tau(s, s') \Rightarrow rank(s') < rank(s)$, $\forall s.rank(s) \geq 0$.
- Control flow graph induced by the transition system.

Loops in Control Flow Graph

- Loop header, entry e dge, loop edge and exit edge.

$\texttt{int}\ {}^{1}x := \texttt{?}$
$\texttt{while}\ {}^{2}(x \neq 0)\ \texttt{do}$
$\quad \texttt{if}\ {}^{3}(x < 10)\ \texttt{then}$
$\qquad {}^{4}x := x + 1$
$\quad \texttt{else}$
$\qquad {}^{5}x := -x$
$\quad \texttt{fi}$
$\texttt{od}^{6}$

(a)



(b)

Verifying Safety Properties

Verifying Nontermination:

$\texttt{int}\ {}^{1}x := 0,\ y := 9$
$\texttt{while}\ {}^{2}(x \neq y)\ \texttt{do}$
$\quad {}^{3}x := x + 1$
$\quad {}^{4}y := y + 1$
$\texttt{od}^{5}$

(a)

$\texttt{int}\ {}^{1}x := 0,\ y := 9$
$\texttt{while}\ {}^{2}(x \neq y)\ \texttt{do}$
$\quad {}^{3}x := x + 1$
$\quad {}^{4}y := y + 1$
$\texttt{od}$
$\texttt{assert}\ (\texttt{false})^{5}$

(b)

Verifying a ranking function:

$\texttt{int}\ {}^{1}x := \texttt{?},\ r := \texttt{max}\{-x, 21 - x, x + 1\}$
$\texttt{while}\ {}^{2}(x \neq 0)\ \texttt{do}$
$\quad r := r - 1$
$\quad \texttt{assert}\ (r \geq 0)$
$\quad \texttt{if}\ {}^{3}(x < 10)\ \texttt{then}\ {}^{4}x := x + 1\ \texttt{else}\ {}^{5}x := -x\ \texttt{fi}$
$\texttt{od}^{6}$

Verifying Termination via Safety: the Algorithm

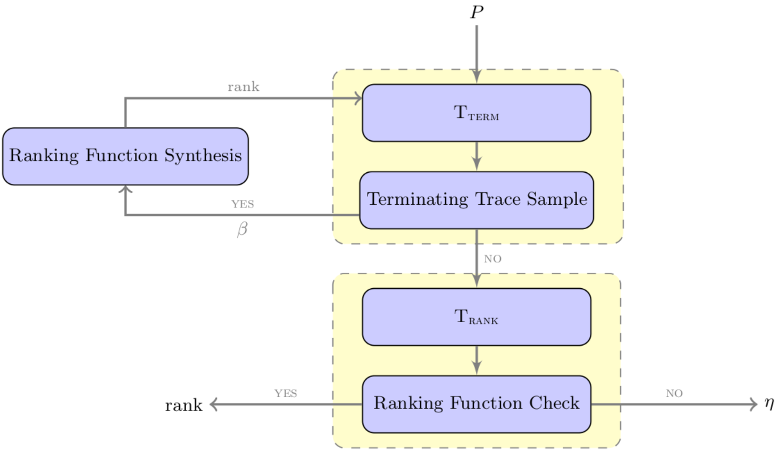
Algorithm 1 : Program Termination

```
function IsTerminating(⟨Σ, τ⟩)
    R ← ∅
    for h ∈ getLoops(⟨Σ, τ⟩) do                    ▷ h is a loop header in the program
        r: ρ ← isLoopTerminating(h, ⟨Σ, τ⟩)
        if r then                                   ▷ the loop is terminating
            R ← R [h ↦ ρ]
        else return false: ρ                        ▷ ρ is a potentially non-terminating state
    return true: R                                  ▷ R is a ranking function for the program
```

Algorithm 2 : Loop Termination

```
function IsLoopTerminating(h,⟨Σ, τ⟩)                ▷ h is the loop header
    rank ← 0                                        ▷ candidate ranking function initialization
    B ← ∅
    while true do
        β ← getTerminatingTrace(h, ⟨Σ, τ⟩, rank)
        if β then                                   ▷ there are terminating traces violating rank
            B ← B ∪ β
            rank ← getCandidateRankingFunction(rank, B)
        else                                        ▷ there are no terminating traces violating rank
            η ← isRankingFunction(rank)
            if η then                               ▷ η is a potentially non-terminating state
                return false: η
            else                                    ▷ rank is a ranking function for the loop
                return true: rank
```

Verifying Termination via Safety



Search for Ranking Function Counterexamples

$\mathbf{T_{TERM}}$ **Transformation**: For a program $\langle \Sigma, \tau \rangle$ and candidate ranking function $rank$. Construct new set of state $\Sigma' = (\Sigma \times \mathbb{Z}) \cup \{\omega\}$, and modified transition relation τ':

- For each loop entry transition $\tau(s, \langle h, \bar{x} \rangle)$, there exists an τ^{rank} s.t.

$$\tau^{rank}(\langle s, r \rangle, \langle \langle h, \bar{x} \rangle, r' \rangle) \Leftrightarrow \tau(s, \langle h, \bar{x} \rangle) \wedge r' = rank(\bar{x})$$

- For each loop transition $tau(\langle h, \bar{x} \rangle, s)$, there exists a loop transition $r^{\ominus}$ s.t.

$$r^{\ominus}(\langle \langle h, \bar{x} \rangle, r \rangle, \langle s, r' \rangle) \Leftrightarrow \tau(\langle h, \bar{x} \rangle, s) \wedge r' = r \ominus 1$$

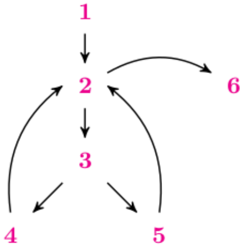
- For each loop exit transition $\tau(s, s')$, there exists a transition $\tau^{<}$ to the error state ω s.t.

$$\tau^{<}(\langle s, r \rangle, \omega) \stackrel{def}{=} r < 0$$

Example: $\mathrm{T}_{\mathrm{TERM}}$ Transformation

$\texttt{int}\ {}^{1}x := \texttt{?}$
$\texttt{while}\ {}^{2}(x \neq 0)\ \texttt{do}$
$\quad \texttt{if}\ {}^{3}(x < 10)\ \texttt{then}$
$\qquad {}^{4}x := x + 1$
$\quad \texttt{else}$
$\qquad {}^{5}x := -x$
$\quad \texttt{fi}$
$\texttt{od}^{6}$

(a)



(b)

$\texttt{int}\ {}^{1}x := \texttt{?},\ r := rank$
$\texttt{while}\ {}^{2}(x \neq 0)\ \texttt{do}$
$\quad r := r - 1$
$\quad \texttt{if}\ {}^{3}(x < 10)\ \texttt{then}\ {}^{4}x := x + 1\ \texttt{else}\ {}^{5}x := -x\ \texttt{fi}$
$\texttt{od}$
$\texttt{assert}\ (r \geq 0)^{6}$

Search for Ranking Function Counterexamples

Theorem

The error state ω *is reachable*
iff.
A finite trace jump out of loop with header h *and visit* h *more than* $rank(\bar{x})$ *times.*

Validating Ranking Function

$\textsc{Trank}$ **Transformation**:

$$\tau^{rank}(\langle s, r\rangle, \langle\langle h, \bar{x}\rangle, r'\rangle) \Leftrightarrow \tau(s, \langle h, \bar{x}\rangle) \wedge r' = rank(\bar{x})$$

$$r^{\ominus}(\langle\langle h, \bar{x}\rangle, r\rangle, \langle s, r'\rangle) \Leftrightarrow \tau(\langle h, \bar{x}\rangle, s) \wedge r' = r \ominus 1$$

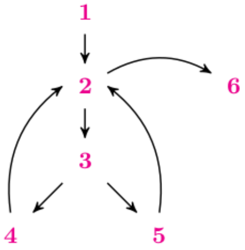
For **loop transitions** $\tau(s, s')$.

$$\tau^{<}(\langle s, r\rangle, \omega) \stackrel{def}{=} r < 0$$

Example: $\mathrm{T_{RANK}}$ Transformation

$\texttt{int}\ {}^{1}x := \texttt{?}$
$\texttt{while}\ {}^{2}(x \neq 0)\ \texttt{do}$
$\quad \texttt{if}\ {}^{3}(x < 10)\ \texttt{then}$
$\qquad {}^{4}x := x + 1$
$\quad \texttt{else}$
$\qquad {}^{5}x := -x$
$\quad \texttt{fi}$
$\texttt{od}^{6}$

(a)



(b)

$\texttt{int}\ {}^{1}x := \texttt{?},\ r := \texttt{max}\{-x, 21 - x, x + 1\}$
$\texttt{while}\ {}^{2}(x \neq 0)\ \texttt{do}$
$\quad r := r - 1$
$\quad \texttt{assert}\ (r \geq 0)$
$\quad \texttt{if}\ {}^{3}(x < 10)\ \texttt{then}\ {}^{4}x := x + 1\ \texttt{else}\ {}^{5}x := -x\ \texttt{fi}$
$\texttt{od}^{6}$

Validating Ranking Functions

Theorem

The error state ω *is reachable*
iff.
The corresponding finite trace is the prefix of a an infinite trace which visits the loop header h *more than* $rank(\bar{x})$ *times.*

Synthesis of Candidate Ranking Function

- Target: affine ranking function pieces.
- $\{\langle \bar{x}_1, r_1\rangle, \langle \bar{x}_2, r_2\rangle, \ldots\}$ is the set of pairs mapping the initial states of terminating traces to number of iterations need for termination.
- Template: $\bar{m} \cdot \bar{x} + q$

$$\begin{gathered} \bar{m} \cdot \bar{x}_1 + q = r_1 \\ \bar{m} \cdot \bar{x}_2 + q = r_2 \\ \vdots \end{gathered}$$

Then utilize these ranking pieces to form piecewise, multi-phase and lexicographic ranking functions.

Implementation and Experiments

Implemented in SeaHorn.
Experimental evaluation is conducted on SV-COMP 2015.

	Tot	Time
SeaHorn	135	1.71s
AProVE [27]	129	10.77s
FuncTion [29]	111	0.55s
HIPTnT+ [21]	152	0.62s
Ultimate [15]	109	8.45s

Experimental Evaluation