Westfälische
Wilhelms-Universität
Münster

Weighted Dynamic Pushdown Networks

MEMICS 2009 / 14.11.09

Alexander Wenner

wissen.leben
WWU Münster

Motivation

- Weighted Pushdown Systems [RSJM05]
  - Extend Pushdown Systems with Weights
  - Generalize Interprocedural Dataflow Analysis
- Dynamic Pushdown Networks [BMOT05]
  - Model Network of Parallel Processes
  - Dynamic Creation of Processes
- Weighted Dynamic Pushdown Networks
  - Combination of WPDS and DPN
  - Generalize Interprocedural Dataflow Analysis of Parallel Processes with Dynamic Process Creation

Weighted Dynamic Pushdown System

- WDPN $\mathcal{W} = (\mathcal{M}, \mathcal{S}, f)$
  - DPN $\mathcal{M} = (P, \Gamma, \Delta)$
    - Configurations $(P\Gamma^*)^*$
    - Transition Rules $p\gamma \hookrightarrow cp'w'$, $c \in (P\Gamma^*)^*$, $p'w'P\Gamma^*$
  - Semiring $\mathcal{S} = (D, \oplus, \odot, 0, 1)$
  - Weight Function $f : \Delta \to D$
- Interleaving Semantics $c \xrightarrow{\rho} c'$
  - Paths$(c, C) = \{\rho \mid \exists c' \in C \; c \xrightarrow{\rho} c'\}$
- Abstraction $\alpha(r_1 \ldots r_n) = f(r_1) \odot \cdots \odot f(r_n)$

Generalised Pushdown Predecessor Problem

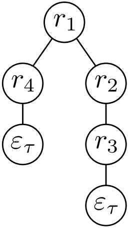
$$\delta(c, C) = \bigoplus\{\alpha(\rho) \mid \rho \in \text{Paths}(c, C)\}$$

Example Bitvector Analysis

- Semiring $D = \{\text{GEN}, \text{ID}, \text{KILL}, \text{ZERO}\}$
- DPN $\Delta = \{r_1, r_2, r_3, r_4\}$, Weight Function f
  - $r_1 = p\gamma_1 \hookrightarrow p\gamma_3 p\gamma_4,\ f(r_1) = \text{ID}$
  - $r_2 = p\gamma_4 \hookrightarrow p,\ f(r_2) = \text{ID}$
  - $r_3 = p\gamma_2 \hookrightarrow p\gamma_5,\ f(r_3) = \text{KILL}$
  - $r_4 = p\gamma_3 \hookrightarrow p\gamma_6,\ f(r_4) = \text{GEN}$
- $c = p\gamma_1\gamma_2,\ C = \{p\gamma_6 p\gamma_5\}$
- $\text{Paths}(c, C) = \{r_1 r_4 r_2 r_3, r_1 r_2 r_4 r_3, r_1 r_2 r_3 r_4\}$
- $\delta(c, C) = (\text{GEN} \odot \text{KILL}) \oplus (\text{KILL} \odot \text{GEN}) = \text{GEN}$
- But: Set of Paths can not be directly abstracted.

Execution Hedges

- Branching instead of Interleaving
- Tree for each initial Process
- Hedge of Trees for Configuration
- Branching Semantics $c \overset{\sigma}{\Longrightarrow} c'$
  - $\text{Hedges}(c, C) = \{\sigma \mid \exists c' \in C \; c \overset{\sigma}{\Longrightarrow} c'\}$
- Interleaving ψ

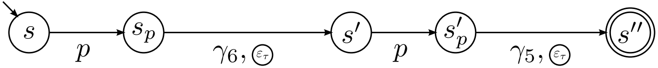


Reaching Paths and Hedges

$$\text{Paths}(c, C) = \psi(\text{Hedges}(c, C))$$

Computing Hedges of a DPN

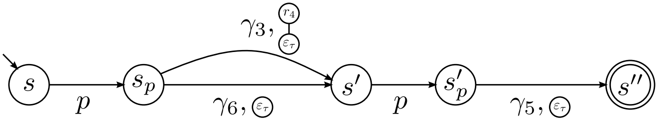
$$r_1 = p\gamma_1 \hookrightarrow p\gamma_3 p\gamma_4$$
$$r_2 = p\gamma_4 \hookrightarrow p$$
$$r_3 = p\gamma_2 \hookrightarrow p\gamma_5$$
$$r_4 = p\gamma_3 \hookrightarrow p\gamma_6$$



- Bottom-Up *PRE**-Construction, Concatenation ;
- Constraints collect Trees
  - $\mathsf{L}[(s, \gamma, s')] \supseteq \{\varepsilon_\tau\}$
  - $\mathsf{L}[(s_p, \gamma, s')] \supseteq r(\pi_{\mathsf{L}}(s, c, s'))$ for $p\gamma \hookrightarrow c$

Computing Hedges of a DPN

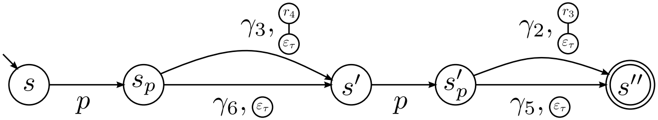
$r_1 = p\gamma_1 \hookrightarrow p\gamma_3 p\gamma_4$
$r_2 = p\gamma_4 \hookrightarrow p$
$r_3 = p\gamma_2 \hookrightarrow p\gamma_5$
$r_4 = p\gamma_3 \hookrightarrow p\gamma_6$



- Bottom-Up *PRE**-Construction, Concatenation ;
- Constraints collect Trees
  - $L[(s, \gamma, s')] \supseteq \{\varepsilon_\tau\}$
  - $L[(s_p, \gamma, s')] \supseteq r(\pi_L(s, c, s'))$ for $p\gamma \hookrightarrow c$

Computing Hedges of a DPN

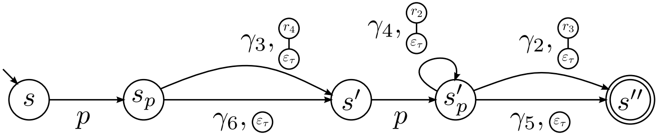
$r_1 = p\gamma_1 \hookrightarrow p\gamma_3 p\gamma_4$
$r_2 = p\gamma_4 \hookrightarrow p$
$r_3 = p\gamma_2 \hookrightarrow p\gamma_5$
$r_4 = p\gamma_3 \hookrightarrow p\gamma_6$



- Bottom-Up *PRE**-Construction, Concatenation **;**
- Constraints collect Trees
  - $\mathsf{L}[(s, \gamma, s')] \supseteq \{\varepsilon_\tau\}$
  - $\mathsf{L}[(s_p, \gamma, s')] \supseteq r(\pi_{\mathsf{L}}(s, c, s'))$ for $p\gamma \hookrightarrow c$

Computing Hedges of a DPN

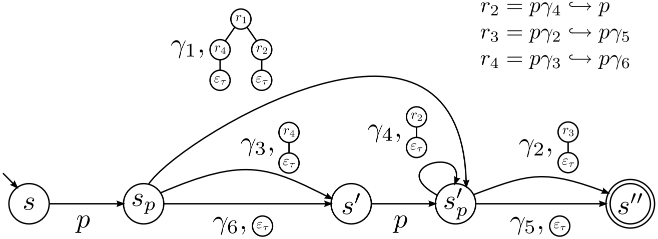
$$r_1 = p\gamma_1 \hookrightarrow p\gamma_3 p\gamma_4$$
$$r_2 = p\gamma_4 \hookrightarrow p$$
$$r_3 = p\gamma_2 \hookrightarrow p\gamma_5$$
$$r_4 = p\gamma_3 \hookrightarrow p\gamma_6$$



- Bottom-Up *PRE**-Construction, Concatenation ;
- Constraints collect Trees
  - $\mathsf{L}[(s, \gamma, s')] \supseteq \{\varepsilon_\tau\}$
  - $\mathsf{L}[(s_p, \gamma, s')] \supseteq r(\pi_{\mathsf{L}}(s, c, s'))$ for $p\gamma \hookrightarrow c$

Computing Hedges of a DPN

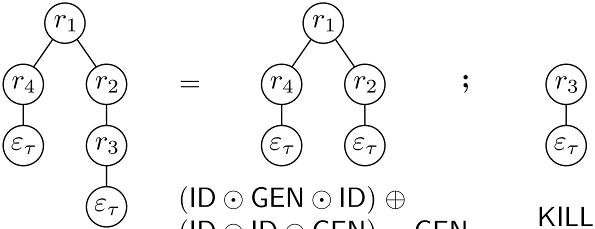
$r_1 = p\gamma_1 \hookrightarrow p\gamma_3 p\gamma_4$
$r_2 = p\gamma_4 \hookrightarrow p$
$r_3 = p\gamma_2 \hookrightarrow p\gamma_5$
$r_4 = p\gamma_3 \hookrightarrow p\gamma_6$



- Bottom-Up *PRE**-Construction, Concatenation ;
- Constraints collect Trees
  - $\mathsf{L}[(s, \gamma, s')] \supseteq \{\varepsilon_\tau\}$
  - $\mathsf{L}[(s_p, \gamma, s')] \supseteq r(\pi_\mathsf{L}(s, c, s'))$ for $p\gamma \hookrightarrow c$

Weights for Hedges

- Consider Bitvector Analysis



$(\mathsf{ID} \odot \mathsf{GEN} \odot \mathsf{ID}) \oplus$
$(\mathsf{ID} \odot \mathsf{ID} \odot \mathsf{GEN}) = \mathsf{GEN}$

KILL

$\mathsf{GEN} \odot \mathsf{KILL} = \mathsf{KILL}$

- But: Interleaving of Semiring Weights not sound.

Branching Weighted Dynamic Pushdown System

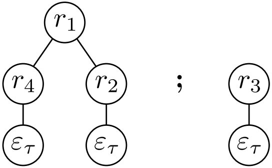
- BWDPN $\mathcal{B} = (\mathcal{M}, \mathcal{E}, \bar{f})$
  - DPN $\mathcal{M} = (P, \Gamma, \Delta)$
  - Extended Semiring $\mathcal{E} = (E, \bar{\oplus}, \bar{\odot}, \bar{\otimes}, \bar{0}, \bar{1})$
    - $(e_1 \bar{\otimes} e_2) \bar{\odot} e_3 = e_1 \bar{\otimes} (e_2 \bar{\odot} e_3)$
  - Weight Function $\bar{f} : \Delta \to E$
- Branching Semantics
- Abstraction $\beta(r(\tau_1 \dots \tau_n)) = \bar{f}(r) \bar{\odot} (\beta(\tau_1) \bar{\otimes} \dots \bar{\otimes} \beta(\tau_n))$

Branching Generalised Pushdown Predecessor Problem

$$\theta(c, C) = \bar{\bigoplus} \{\beta(\sigma) \mid \sigma \in \text{Hedges}(c, C)\}$$

Connection

- Consider Bitvector Analysis
- Extended Semiring $E = \{\overline{\mathsf{GEN}}, \mathsf{GEN}, \mathsf{ID}, \mathsf{KILL}, \mathsf{ZERO}\}$
- Extension $(\mathcal{S}, \mathcal{E}, \iota, \eta)$
  - Embedding $\iota : D \rightarrow E$
  - Projection $\eta : E \rightarrow D$



$\mathsf{GEN} \,\bar{\otimes}\, \mathsf{ID} = \overline{\mathsf{GEN}}$ $\quad$ KILL

$\overline{\mathsf{GEN}} \,\bar{\odot}\, \mathsf{KILL} = \overline{\mathsf{GEN}}$

Connection

$$\delta(c, C) = \eta(\theta(c, C)) \text{ für } \bar{f}(r) = \iota(f(r))$$

Applications/Future Work

- Applications
  - Shortest Path
  - Bitvector Analyses
  - KILL/GEN Analyses
- Future Work
  - Iterate Method
    - Synchronisation
  - Additional Weight Domains
    - Generalize Construction
    - Weights for Hedges, e.g. Locks [LMOW09]

Bibliography

A. Bouajjani, M. Müller-Olm, and T. Touili.
Regular symbolic analysis of dynamic networks of pushdown systems.
In *CONCUR*, volume 3653 of *LNCS*. Springer, 2005.

P. Lammich, M. Müller-Olm, and A. Wenner.
Predecessor sets of dynamic pushdown networks with tree-regular constraints.
In *CAV*, volume 5643 of *LNCS*. Springer, 2009.

T. Reps, S. Schwoon, S. Jha, and D. Melski.
Weighted pushdown systems and their application to interprocedural dataflow analysis.
Sci. Comput. Program., 58(1-2), 2005.