Deciding Branching Bisimilarity between BPA and Finite-State Systems

Hongfei Fu

BASICS Laboratory
Department of Computer Science
Shanghai Jiao Tong University

APLAS 2009 Paper

Outline

Background

Formal Verification

- formal specification: finite-state system
- real implementation: infinite-state system

Comparative semantics

Bisimulation Semantics:

- strong bisimulation (Park)
- weak bisimulation (Milner)
- branching bisimulation (van Glabbeek and Weijland)

The Problem We Study

Polynomial time algorithms deciding branching bisimilarity between:

- BPA (Basic Process Algebra) and finite-state systems (FS)
- Normed BPP (Basic Parallel Processes) and finite-state systems

Definitions

Notations

- Let V be a finite alphabet of symbols.
  Symbols of V are ranged over by $X, Y, Z \ldots$.
- The set of words over V is denoted V^*.

Notations

We presume a set of actions Act_τ.

We always use

- Γ to refer to a FS
- $State(\Gamma)$ to refer to the state set of Γ
- $f, g, h \ldots$ to range over $State(\Gamma)$

BPA Processes

A BPA system is a tuple (V, Δ) where

- V is a finite alphabet of symbols.
- Δ is a finite set of *rules* for which each rule has the form $X \xrightarrow{a} \alpha$ where $X \in V$, $\alpha \in V^*$ and $a \in Act$.

BPA Processes

A BPA system (V, Δ) defines an LTS where

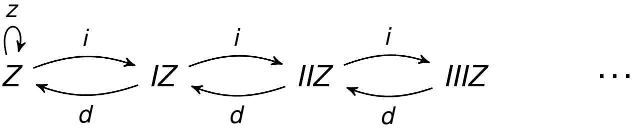
- states are elements of V^*.
- for $\alpha, \beta \in V^*$, $\alpha \xrightarrow{a} \beta$ if $\alpha = Y\gamma$, $Y \xrightarrow{a} \gamma' \in \Delta$ and $\beta = \gamma'\gamma$.

A state (word) $\alpha \in V^*$ is *normed* if $\alpha \rightarrow^* \varepsilon$.
$\alpha \in V^*$ is *unnormed* if α is not normed.

BPA: An Example

$V = \{I, Z\}$

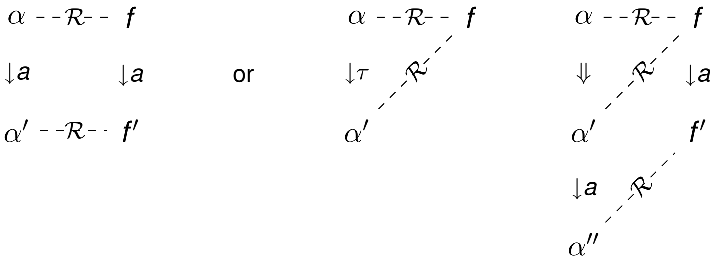
$\Delta = \{Z \xrightarrow{z} Z, Z \xrightarrow{i} IZ, I \xrightarrow{i} II, I \xrightarrow{d} \varepsilon\}$



Branching Bisimulation Under "Contraction"

Let (V, Δ) be a BPA system and Γ be a FS system.

A binary relation $\mathcal{R} \subseteq V^* \times \mathit{State}(\Gamma)$ is a *branching bisimulation* if whenever $(\alpha, f) \in \mathcal{R}$ then for each $a \in \mathit{Act}$:



The *branching bisimilarity* $\approx_{\text{br}}$ is the largest branching bisimulation.

Bisimulation Base Technique

Description

- an effective technique developed by D. Caucal to decide bisimilarity concerning infinite-state systems
- a finite relation "*bisimulation base*" from which the whole bisimilarity relation can be *effectively generated*.
- We can decide the bisimulation problem if we can compute the corresponding bisimulation base.

Previous Application

- strong bisimulation over BPA/BPP
  S. Christensen, H. Hüttel, and C. Stirling 1992/
  S. Christensen, Y. Hirshfeld, and F. Moller 1993
- strong bisimulation over normed BPA/normed BPP
  Y. Hirshfeld, M. Jerrum and F. Moller, 1994/1996

Previous Application

- weak bisimulation between FS and BPA/normed BPP (†)
  Antonín Kučera and Richard Mayr (2002)
- various bisimulations between FS and pushdown processes.
  Antonín Kučera and Richard Mayr (2004)

Our Application

- the application on the branching bisimilarity between a BPA system and a FS system.
- We follow and rely on the scheme of the previous work on weak bisimulation (†).

The Bisimulation Base between BPA and FS

- We fix a BPA system (V, Δ) and a FS Γ.
- We construct the BPA system $(V', \Delta') = (V \cup State(\Gamma), \Delta \cup \Gamma)$
- Special attention on words of the form αf with $\alpha \in V^*$, $f \in State(\Gamma)$.

The Bisimulation Base between BPA and FS

We also define:

- $\mathcal{G}_\Gamma$ to be the $\approx_{\mathrm{br}}$ over $\Gamma \cup \{\varepsilon\}$.
- $\mathit{Normed}(V) = \{X \in V | X \text{ is normed}\}$.

Well-formed Relations

A relation K is *well-formed* if $\mathcal{G}_\Gamma \subseteq K$ and K is a subset of the relation $\mathcal{G}$ defined by:

$$\begin{aligned}\mathcal{G} &= ((\mathit{Normed}(V) \cdot \mathit{State}(\Gamma)) \times \mathit{State}(\Gamma)) \cup \\ &\quad (V \times \mathit{State}(\Gamma)) \cup \mathcal{G}_\Gamma\end{aligned}$$

$\mathcal{G}$ is the largest well-formed relation.

The Bisimulation Base

The *bisimulation base* $\mathcal{B}$, is a well-formed relation as follows:

$$\begin{aligned}\mathcal{B} &= \{(Yf, g) \in Normed(V) \cdot State(\Gamma) | Yf \approx_{\mathrm{br}} g, Y \in Normed(V)\} \\ &\cup \{(Y, g) \in V \times State(\Gamma) | Y \approx_{\mathrm{br}} g\} \cup \mathcal{G}_\Gamma\end{aligned}$$

Closure

Let K be a well-formed relation. The *closure* of K, denoted $Cl(K)$, is the least relation M such that:

$$K \subseteq M \qquad \frac{(Yf, g) \in K, (\alpha, f) \in M}{(Y\alpha, g) \in M} \qquad \frac{(Yf, g) \in K, (\alpha h, f) \in M}{(Y\alpha h, g) \in M}$$

$$\frac{(\alpha, g) \in M, \alpha \text{ is unnormed}}{(\alpha\beta, g), (\alpha\beta h, g) \in M \text{ for every } \beta \in V^* \text{ and } h \in State(\Gamma)}$$

$Cl(K)$ will only contain pairs of two form: (α, g) and $(\alpha f, g)$.

The Idea Behind Closure

The basic properties w.r.t sequential computation:

- $Yf \approx_{\mathrm{br}} g$, $\alpha \approx_{\mathrm{br}} f$ implies $Y\alpha \approx_{\mathrm{br}} g$.
- $\alpha \approx_{\mathrm{br}} g$, α is unnormed implies $\alpha\beta \approx_{\mathrm{br}} g$, $\alpha\beta h \approx_{\mathrm{br}} g$

Regular Property of the Closure

Theorem

Let K be a well-formed relation. For each $g \in State(\Gamma)$ there is a finite automaton $\mathcal{A}_g^K$ constructible in polynomial time such that $L(\mathcal{A}_g^K) = \{\alpha | (\alpha, g) \in Cl(K)\} \cup \{\alpha f | (\alpha f, g) \in Cl(K)\}$

$(\alpha, g) \in Cl(K)$ iff $\alpha \in L(\mathcal{A}_g^K)$.

The Bisimulation Base: The Key Property

Theorem

$$Cl(\mathcal{B}) = \{(\alpha, g) | \alpha \approx_{\mathrm{br}} g\} \cup \{(\alpha f, g) | \alpha f \approx_{\mathrm{br}} g\}.$$

$\alpha \approx_{\mathrm{br}} g$ iff $\alpha \in \mathcal{A}_g^{\mathcal{B}}$

Computing The Bisimulation Base

The Work Flow

- First we develop an expansion function *Exp* over well-formed relations such that $Exp(K) \subseteq K$ for every K.
- Then we iteratively apply *Exp* to $\mathcal{G}$:
  $\mathcal{B}^0 = \mathcal{G}$, $\mathcal{B}^1 = Exp(\mathcal{B}^0)$,$\ldots$, $\mathcal{B}^{k+1} = Exp(\mathcal{B}^k)$, $\ldots$
- Finally we obtain a fixed point of *Exp* which is exactly the bisimulation base

The Expansion Function: The Idea

The idea naturally follows the definition of branching bisimulation.

Let K be a well-formed relation. Roughly a pair, say, (X, g) *expands* in K by the following conditions:

$$
\begin{array}{ccc}
X & K & g \\
\downarrow a & & \downarrow a \\
\alpha' & Cl(K) & g'
\end{array}
\qquad \text{or} \qquad
\begin{array}{ccc}
X & K & g \\
\downarrow \tau & Cl(K) & \\
\alpha' & &
\end{array}
\qquad
\begin{array}{ccc}
X & K & g \\
\Downarrow & Cl(K) & \downarrow a \\
\alpha' & & g' \\
\downarrow a & Cl(K) & \\
\alpha'' & &
\end{array}
$$

$Exp(K)$ is the set of pairs of K that expands in K.

Branching Bisimulation Under "Contraction" (van Glabbeek and P. Weijland)

A binary relation $\mathcal{R} \subseteq V^* \times State(\Gamma)$ is a *branching bisimulation* if whenever $(\alpha, f) \in \mathcal{R}$ then for each $a \in Act$:

$$\begin{array}{ccc} \alpha & \text{-}\text{-}\mathcal{R}\text{-}\text{-} & f \\ \downarrow a & & \downarrow a \\ \alpha' & \text{-}\text{-}\mathcal{R}\text{-}\text{-} & f' \end{array} \qquad \text{or} \qquad \begin{array}{ccc} \alpha & \text{-}\text{-}\mathcal{R}\text{-}\text{-} & f \\ \downarrow \tau & \diagup\mathcal{R} & \\ \alpha' & & \end{array} \qquad \begin{array}{ccc} \alpha & \text{-}\text{-}\mathcal{R}\text{-}\text{-} & f \\ \Downarrow & \diagup\mathcal{R} & \downarrow a \\ \alpha' & & f' \\ \downarrow a & \diagup\mathcal{R} & \\ \alpha'' & & \end{array}$$

The *branching bisimilarity* $\approx_{\mathrm{br}}$ is the largest branching bisimulation.

The Real Situation

Let K be a well-formed relation. For each $g \xrightarrow{a} h \in \Gamma$, we define an auxiliary language $L^K_{[g,a,h]}$ which approximates the language:

$$\overline{L}^K_{[g,a,h]} \triangleq \{E | (E, g) \in Cl(K) \wedge \exists E'(E \xrightarrow{a} E' \wedge (E', h) \in Cl(K))\}$$

The Real Situation

Let K be a well-formed relation. A pair (X, g) *expands* in K by the following conditions:

$$
\begin{array}{ccc}
X & K & g \\
\downarrow a & & \downarrow a \\
\alpha' & Cl(K) & g'
\end{array}
\quad \text{or} \quad
\begin{array}{ccc}
X & K & g \\
\downarrow \tau & Cl(K) & \\
\alpha' & &
\end{array}
\qquad
\begin{array}{ccc}
X & K & g \\
\Downarrow & & \downarrow a \\
\alpha' & & g' \\
\pitchfork & & \\
L^{K}_{[g,a,g']} & &
\end{array}
$$

The Auxiliary Language: Formal Definition

Let K be a well-formed relation. For each $g \xrightarrow{a} h \in \Gamma$, a pair $(X, g), (Xf, g) \in K$ satisfies the condition $\phi^K_{[g,a,h]}$, if:

- For (X, g): X is unnormed and there is some $X \xrightarrow{a} \alpha$ such that $(\alpha, h) \in Cl(K)$
- For (Xf, g): there is some $X \xrightarrow{a} \alpha$ such that $(\alpha f, h) \in Cl(K)$

Then, the (regular) language $L^K_{[g,a,h]}$ is defined as the union of the regular languages below:

- $\{X\} \cdot L(\mathcal{A}^K_f)$, for (Xf, g) satisfies $\phi^K_{[g,a,h]}$
- $\{X\} \cdot V^* + \{X\} \cdot V^* \cdot State(\Gamma)$, for (X, g) satisfies $\phi^K_{[g,a,h]}$

The Expansion Function: The Formal Definition

Let K be a well-formed relation. A pair in K *expands* in K iff

- For (Y, g): whenever $Y \xrightarrow{a} \alpha$, there is some $g \xrightarrow{a} g'$ such that $(\alpha, g') \in Cl(K)$, or $a = \tau$ and $(\alpha, g) \in Cl(K)$; and whenever $g \xrightarrow{a} g'$, there is some $Y \Rightarrow \alpha$ such that $\alpha \in L^{K}_{[g,a,g']}$.
- For (Yf, g): ....
- For pairs in $\mathcal{G}_\Gamma$: ....

The set $Exp(K)$ is defined as all pairs in K that expand in K.

Consistency with Branching Bisimulation

Lemma

Let K be a well-formed relation such that $Exp(K) = K$. Then $Cl(K)$ is a branching bisimulation.

Procedure: $\mathcal{B}^0 = \mathcal{G}$, $\mathcal{B}^{k+1} = Exp(\mathcal{B})$.

Theorem

There is a polynomial bounded natural number j such that $\mathcal{B}^j = \mathcal{B}^{j+1}$. Moreover, $\mathcal{B}^j = \mathcal{B}$.

Theorem

Let K be a well-formed relation. The set $Exp(K)$ can be computed in polynomial time.

The Main Theorem

Theorem

Branching bisimilarity between BPA and FS is polynomial time decidable.

Summary

- We've developed a polynomial time algorithm deciding branching bisimilarity between BPA and FS.
- Our algorithm is not new, but improves a previous result by Antonín Kučera and Richard Mayr (2004).
- We've shown the scheme of the bisimulation base technique.

Questions?