# Process Algebra (2IMF10)

## Recursion

Bas Luttik

MF 6.072
s.p.luttik@tue.nl
http://www.win.tue.nl/~luttik

Lectures 6–7

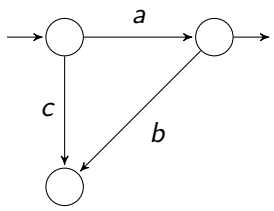


## Expressiveness (1)



### Question

Can we express the following transition system




in the syntax of BSP($A$).

### Answer

Yes! The transition system generated by $a.(b.0 + 1) + c.0$ is isomorphic to the transition system above.

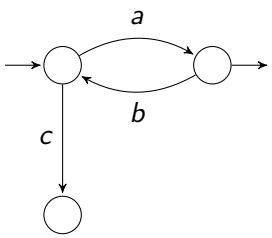


## Expressiveness (2)



### Question

Can we express the following transition system




in the syntax of BSP($A$).

### Answer

No! Infinite behaviour cannot be expressed in the syntax of BSP($A$).

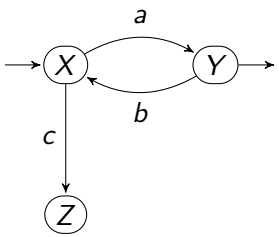


## Expressiveness (3)



General method for expressing transition systems:

1. label states with *process names* (a.k.a. recursion variables);
2. associate behaviour to every recursion variable by means of an equation specifying for every recursion variable its transition- and termination-behaviour.




$$X = a.Y + c.Z$$
$$Y = b.X + 1$$
$$Z = 0$$

## Recursive specifications



Let $\Sigma$ be a signature and let $V_R$ be a set of *recursion variables*

A recursive equation over $\Sigma$ and $V_R$ is an equation of the form

$$X = t \ ,$$

with $X \in V_R$ and $t$ a term over $\Sigma$ and $V_R$. We say that the recursive equation $X = t$ *defines* $X$.

A recursive specification over $\Sigma$ and $V_R$ is a set of recursive equations over $\Sigma$ and $V_R$ consisting of precisely one recursive equation defining $X$ for every $X \in V_R$.



## Recursion: operational semantics



[ Warning: notation slightly simplified; we will generalise later! ]

Let $E$ be a recursive specification over $\Sigma$ and $V_R$ including, for every $X \in V_R$, a defining equation $X = t_X$.

Operational rules for recursion

$$\frac{t_X \xrightarrow{a} t'_X}{X \xrightarrow{a} t'_X} \qquad \frac{t_X\downarrow}{X\downarrow}$$



## Term model (simplified)



Let $E$ be a recursive specification over $\mathrm{BSP}(A)$ and $V_R$.

Definition

The term algebra for $\mathrm{BSP}(A)+E$ is the algebra

$$\mathbb{P}(\mathrm{BSP}(A)+E) = (\mathcal{C}(\mathrm{BSP}(A)+E), +, (a.)_{a\in A}, 0, 1, (X)_{X\in V_R}) \ .$$

Proposition

Bisimilarity is a congruence on $\mathbb{P}(\mathrm{BSP}(A)+E)$.

Theorem

*The equational theory* $\mathrm{BSP}(A)+E$ *is a sound axiomatisation of* $\mathbb{P}(\mathrm{BSP}(A)+E)/\leftrightarrow$.

Is $\mathrm{BSP}(A)+E$ also ground-complete for $\mathbb{P}(\mathrm{BSP}(A)+E)/\leftrightarrow$?



## Equivalence of recursion variables



Example

Consider the recursive specification

$$\left\{ \begin{array}{l} X = a.X \ , \\ Y = a.a.Y \end{array} \right\}$$

Then $X \leftrightarrow Y$.

But this equation cannot be derived from $\mathrm{BSP}(A) + E$. (Prove!)

Conclusion: we need additional methods to reason about the equivalence of recursion variables.

[ Remark: we shall not discuss a full-fledged ground-complete axiomatisation of the algebra $\mathbb{P}(\mathrm{BSP}(A)+E)/\leftrightarrow$, but we'll come close.]

## Recursion: algebraic approach



Let $E$ be a rec. spec. over signature $\Sigma$ and set of variables $V_R$.

Furthermore, let $\mathbf{A}$ be a $\Sigma$-algebra and $\iota$ the associated interpretation.

Definition

A solution of $E$ in $\mathbf{A}$ is an extension $\kappa$ of $\iota$ with interpretations of the recursion variables in $V_R$ as elements of $\mathbf{A}$ such that $\mathbf{A}, \kappa \models X = t_X$ for every equation $X = t_X$ in $E$. If $\kappa$ is a solution of $E$ and $X \in V_R$, then we shall call $\kappa(X)$ a *solution* of $X$ in $E$.

NB: a solution of $E$ in an algebra of transition systems modulo bisimilarity is an assignment of closed terms to recursion variables such that the recursion equations are true up to bisimilarity.



## Solutions: examples



1. The recursive specification $E_1 = \{X = a.1\}$ has a solution both in $\mathbb{P}(\mathrm{BSP}(A))/\underline{\leftrightarrow}$ and in $\mathbb{P}(\mathrm{BSP}(A)+E_1)/\underline{\leftrightarrow}$.
2. The recursive specification $E_2 = \{X = a.X\}$ has a solution in $\mathbb{P}(\mathrm{BSP}(A)+E_2)/\underline{\leftrightarrow}$, but not in $\mathbb{P}(\mathrm{BSP}(A))/\underline{\leftrightarrow}$.
3. The recursive specification $E_3 = \{X = X\}$ has many solutions, both in $\mathbb{P}(\mathrm{BSP}(A))/\underline{\leftrightarrow}$ and in $\mathbb{P}(\mathrm{BSP}(A)+E_3)/\underline{\leftrightarrow}$.



## Example: equivalence of rec. vars.



Consider the recursive specification

$$\left\{ \begin{array}{l} X = a.X \ , \\ Y = a.a.Y \end{array} \right\} .$$

Note that we can argue that every solution of $X$ is a solution of $Y$ too:

$$X = a.X = a.a.X \ .$$

Hence, any solution $\kappa$ of $E$ in some algebra $\mathbf{A}$ satisfies

$$\kappa(X) = \kappa(a.a.X) = \mathbf{a}.\mathbf{a}.\kappa(X) \qquad (\mathbf{a}. \text{ the interpretation of } a. \text{ in } \mathbf{A}) \ ,$$

so $\kappa(X)$ is a solution of $Y$ in $E$.



## Example: equivalence of rec. vars.



Consider the recursive specification

$$\left\{ \begin{array}{l} X = a.X \ , \\ Y = a.a.Y \end{array} \right\} .$$

The reasoning on the previous slide allows us to conclude that every solution of $X$ in whatever algebra (!) must also be a solution of $Y$ in that algebra.

The converse, however, need not hold:

Exercise: construct a model of $\mathrm{BSP}(A)$ in which $Y$ has a solution that is not also a solution of $X$.

## Example: equivalence of rec. vars.



Consider the recursive specification

$$\left\{ \begin{array}{l} X = a.X \ , \\ Y = a.a.Y \end{array} \right\} .$$

Perhaps if we exclude some models (e.g., the answers to the exercise on the previous slide), then we may be able to say more about the equivalence of $X$ and $Y$ in the above recursive specification.

Note that, for models in which $X$ and $Y$ both have a unique solution, the reasoning on slide 14 would suffice to conclude that $X$ and $Y$ indeed denote the same process!



## Guardedness



### Definition

An occurrence of a recursion variable $X$ in a closed term $s$ is guarded if it occurs in the scope of an action prefix.

A term $s$ is completely guarded if all occurrences of all recursion variables in $s$ are guarded.

A recursive specification $E$ is completely guarded if all right-hand sides of all equations in $E$ are completely guarded.

### Exercise 5.5.1

Determine whether, in the following terms, the occurrences of the recursion variables $X$ and $Y$ are guarded, unguarded, or both:

$$a.X \ , \quad Y + b.X \ , \quad b.(X + Y) \ , \quad a.Y + X \ .$$



## Guardedness



### Example

1. The recursive specification

   $$E_1 = \{X_1 = a.X_1, \ Y_1 = a.X_1\}$$

   is completely guarded.
2. The recursive specification

   $$E_2 = \{X_1 = a.X_1, \ Y_1 = X_1\}$$

   is *not* completely guarded.

But $E_1$ and $E_2$ have exactly the same solutions in every model!



## Guardedness



### Definition

A recursive specification $E$ is guarded if there exists a completely guarded recursive specification $F$ with $V_R(E) = V_R(F)$ and $\mathrm{BSP}(A)+E \vdash X = t$ for all $X = t \in F$.

### Example

Although the recursive specification

$$E_2 = \{X_1 = a.X_1, \ Y_1 = X_1\}$$

is not completely guarded, it is guarded.

## Guardedness



### Definition

A recursive specification $E$ is guarded if there exists a completely guarded recursive specification $F$ with $V_R(E) = V_R(F)$ and $\mathrm{BSP}(A)+E \vdash X = t$ for all $X = t \in F$.

### Exercise 5.5.2

Determine whether the following recursive specifications are guarded or unguarded:

1. $\{X = Y,\ Y = a.X\}$; and
2. $\{X = a.Y + Z,\ Y = b.Z + X,\ Z = c.X + Y\}$.



## Recursive Specification Principle



### Definition

Let $\Sigma$ be a signature; we say that $\Sigma$-algebra $\mathbf{A}$ satisfies the Recursive Specification Principle (RSP) if every *guarded* recursive specification $E$ over $\Sigma$ and some set $V_R$ of variables has *at most* one solution.



## RSP as a proof principle



### Example

Consider rec. spec. $E$ consisting of the following equations:

$$X = a.X + b.X \ ,$$
$$Y = a.Y + b.Z \ , \text{ and}$$
$$Z = a.Z + b.Y \ .$$

We can prove that $X = Y$ in the context of $E$ as follows:

Define two sequences of terms $\tilde{t} = t_X, t_Y, t_Z$ and $\tilde{u} = u_X, u_Y, u_Z$ by $t_X \equiv X$, $t_Y \equiv Y$, $t_Z \equiv Z$, and $u_X \equiv X$, $u_Y \equiv X$, $u_Z \equiv X$.

Then both $\tilde{t}$ and $\tilde{u}$ denote solutions of $E$ (verify!).

Since $E$ is guarded, by RSP, $\tilde{t} = \tilde{u}$, so $X \equiv u_Y = t_Y \equiv Y$.



## Extension with recursion



[Not in the book!]

Let $A$ be a set of actions, and let $T(A)$ be some process theory.

Let $Rec$ be the collection of all recursive specifications over $T(A)$.

$T(A)_{\mathrm{rec}}$: extension of $T(A)$ with for every rec. spec. $E$ over $T(A)$ and for every $(X = t_X) \in E$:

1. a constant symbol $\mu X.E$, and
2. an axiom $\mu X.E = \mu t_X.E$ (Rec).

[We write $\mu t.E$ for the term obtained from $t$ by replacing all occurrences of a recursion variable $X$ by $\mu X.E$ (see book for an inductive definition).]

## $\mathrm{BSP}_{\mathrm{rec}}(A)$



So: $\mathrm{BSP}_{\mathrm{rec}}(A)$ is the extension of $\mathrm{BSP}(A)$ with, for every rec. spec. $E$ over $\mathrm{BSP}(A)$ and for every $(X = t_X) \in E$, a constant symbol

$$\mu X.E$$

and an equation

$$\mu X.E = \mu t_X.E$$

The term algebra $\mathbb{P}(\mathrm{BSP}_{\mathrm{rec}}(A))$ for $\mathrm{BSP}_{\mathrm{rec}}(A)$ is the algebra

$$(\mathcal{C}(\mathrm{BSP}_{\mathrm{rec}}(A)), +, (a.)_{a\in A}, 0, 1, (\mu X.E)_{E\in Rec, X\in V_R(E)})$$

We use operational semantics and bisimilarity to build a model for $\mathrm{BSP}_{\mathrm{rec}}(A)$ from $\mathbb{P}(\mathrm{BSP}_{\mathrm{rec}}(A))$.



## Term model



$$\frac{}{a.x \xrightarrow{a} x} \qquad \frac{x \xrightarrow{a} x'}{x + y \xrightarrow{a} x'} \qquad \frac{y \xrightarrow{a} y'}{x + y \xrightarrow{a} y'}$$

$$\frac{}{1\downarrow} \qquad \frac{x\downarrow}{(x+y)\downarrow} \qquad \frac{y\downarrow}{(x+y)\downarrow}$$

$$\frac{\mu t_X.E \xrightarrow{a} t'_X}{\mu X.E \xrightarrow{a} t'_X} \qquad \frac{\mu t_X.E\downarrow}{\mu X.E\downarrow}$$

1. Bisimilarity is a congruence on $\mathbb{P}(\mathrm{BSP}_{\mathrm{rec}}(A))$.
2. $\mathbb{P}(\mathrm{BSP}_{\mathrm{rec}}(A))/\underline{\leftrightarrow}$ is a model for the equational theory $\mathrm{BSP}(A)$, the so called term model for $\mathrm{BSP}(A)$.



## Recursive Definition Principle (RDP)



**Definition**
Let $\Sigma$ be a signature; we say that $\Sigma$-algebra $\mathbf{A}$ satisfies the Recursive Definition Principle if every recursive specification $E$ over $\Sigma$ and some set $V_R$ of variables has *at least* one solution.

**Theorem**
$\mathbb{P}(\mathrm{BSP}_{\mathrm{rec}}(A))/\underline{\leftrightarrow} \models \mathrm{RDP}$.

**Proof.**
Let $E$ be a recursive specification. Define $\kappa$ as the extension of $\iota$ such that, for every recursion variable $X$ in $E$,

$$\kappa(X) = [\mu X.E]_{\underline{\leftrightarrow}} \ .$$

Then $\kappa(X) = \kappa(t_X)$ for every recursion variable $X$ in $E$ (verify!), so $\kappa$ is indeed a solution of $E$ in $\mathbb{P}(\mathrm{BSP}_{\mathrm{rec}}(A))/\underline{\leftrightarrow}$. □



## Recursive Specification Principle (RSP)



**Definition**
Let $\Sigma$ be a signature; we say that $\Sigma$-algebra $\mathbf{A}$ satisfies the Recursive Specification Principle (RSP) if every *guarded* recursive specification $E$ over $\Sigma$ and some set $V_R$ of variables has *at most* one solution.

**Theorem**
*The term model* $\mathbb{P}(\mathrm{BSP}_{\mathrm{rec}}(A))/\underline{\leftrightarrow}$ *satisfies RSP (notation:* $\mathbb{P}(\mathrm{BSP}_{\mathrm{rec}}(A))/\underline{\leftrightarrow} \models \mathrm{RSP}$*).*

**Proof.**
[Postponed.] □

## TSP with recursion (example)



Consider the guarded recursive specification $E$ over $\mathrm{TSP}(A)$ consisting of:

$$X = a.(X \cdot Y) + c.1$$
$$Y = b.1 + 1 \ .$$

a. Draw the transition system associated with $\mu X.E$.
b. Prove that the process denoted by $\mu X.E$ in $(\mathrm{TSP} + \mathrm{PR})_{\mathrm{rec}}(A)$ is not finitely definable over $\mathrm{BSP}(A)$.
c. Find a recursive specification $F$ over $\mathrm{BSP}(A)$ defining a recursion variable $Z$ such that $\mu X.E \leftrightarrows \mu Z.F$.





## Example



Let

$$E = \{X = a.a.X\}$$

and let

$$F = \{Y = a.(Y \cdot b.1)\} \ .$$

Prove that

$$\mathrm{TSP}_{\mathrm{rec}}(A) + RSP \vdash \mu X.E = \mu Y.F \ .$$