Bigraphical Reactive Systems

Marino Miculan
MADS lab
(with results by many people)

MeMo Workshop, June 6, 2014

Bigraphical Reactive System

A discrete reactive system is composed by a set of states and a transition relation.

A Bigraphical Reactive System is a RS where:
- States are bigraphs: data structures rendering explicitly the positions and connections of system’s components;
- State transitions are bigraph rewritings defined by a set of local reaction rules.

So BRSs propose as an operational metamodel.

Static

Dynamics

Conclusions

Metatheory & Tools
Bigraphical Reactive Systems

Marino Miculan
MADS lab
(with results by many people)

MeMo Workshop, June 6, 2014
Models for concurrent, distributed systems

Distributed systems are cool, but so damn complex and error prone...

In the last 40 years, TCS have developed hundreds of models and calculi, each focusing on some specific aspects, and providing
- mathematical theories
- prototypes (simulation tools)
- model checkers
- editors
- other nice stuff (even some Right Stuff!)
Models for concurrent, distributed systems

Models are cool, but still so damn complex and error prone...
  • in different models, many definitions and results look almost the same
  • everytime we have to start over (almost) from scratch
  • Implementing tools is time consuming!

"The final model" does not exist
We have to accept a plethora of specific models
**Bigraphs and Bigraphical Reactive Systems**

Introduced by R. Milner et al. (2001) as a formal, graphical **meta-model** for (distributed) systems (but lots of work by many people, since then)

**Main aims:**

- a theoretical framework covering many models dealing with *localities* and *connections*
- general results, tools and techniques which can be readily instantiated to specific calculi
- a common ground where different models can be formally compared and merged

---

*Another (unexpected?) application:*

a data structure for representing semi-structured informations, like knowledge, architectures,...

*In this tutorial: a gentle (and not very abstract) introduction to bigraphs*

For more details:
- Milner's book "The Space and Motion of Communicating Agents", 2009
- bigraph.org
- many other works...
a theoretical framework covering many models dealing with *localities* and *connections*
general results, tools and techniques which can be readily instantiated to specific calculi
a common ground where different models can be formally compared and merged
Another (unexpected?) application:
a data structure for representing *semi-structured informations*, like knowledge architectures,...
In this tutorial: a gentle (and not very abstract) introduction to bigraphs

For more details:
- Milner's book "The Space and Motion of Communicating Agent", 2009
- bigraph.org
- many other works...
Bigraphical Reactive System

A discrete reactive system is composed by a set of states and a transition relation.

A Bigraphical Reactive System is a RS where
- States are bigraphs: data structures rendering explicitly the positions and connections of system's components
- State transitions are bigraph rewritings defined by a set of local reaction rules

So BRSs propose as an operational metamodel.
What's a Bigraph?

A **bigraph** $G$ is a pair of graphs on the same (finite) set $V$ of nodes:

- **the place graph**: a forest representing the topology of the system
  - $G^P: m \to n$
  - Roots ...
  - Sites ...

- **the link graph**: a hypergraph representing the connections
  - $G^L: X \to Y$
  - Outer names ...
  - Inner names ...

```
      v_0
       \  /\  /\
        \ /\ /  \
         \ / \  \
          v_2   v_3
           |     |
           v_1   |
            |     |
            v_1   |
             |     |
             v_0
```

```
      y_0
       /  /
      /  /  \     \
     /  /   \   /  \
    /  /     \ /  \
   /  /       /  \
  /  /         /  \
 x_0 v_2 v_1 x_1
    \        /  \
     \      /  \
      \  /  \
       \ /  \
        \ /  \
         y_1
```
What's a Bigraph?

Compact notation: place graph is represented by nesting nodes; sites are grey holes, and roots (or regions) are outlined.

**bigraph**

\( G: \langle m, X \rangle \rightarrow \langle n, Y \rangle \)

**place graph**

\( G^p: m \rightarrow n \)

**link graph**

\( G^l: X \rightarrow Y \)

Each node is given a type (called **control**) taken from a set \( \Sigma \) called the **signature**.
The control specifies

- whether the node is **atomic**: atomic nodes must be empty
- the number of **ports** of the node
Each node is given a type (called **control**) taken from a set $\Sigma$ called the **signature**.
The control specifies
- whether the node is **atomic**: atomic nodes must be empty
- the number of **ports** of the node
Anatomy of a Bigraph

place = root or node or site

\[ G^P = (V, \text{ctrl}, \text{prnt}) : m \to n \]  (place graph)

\[ G^L = (V, E, \text{ctrl}, \text{edge}, \text{link}) : X \to Y \]  (link graph)

\[ G = (V, E, \text{ctrl}, \text{edge}, \text{prnt}, \text{link}) : \langle m, X \rangle \to \langle n, Y \rangle \]  (bigraph)

\[ = (G^P, G^L) \]  A pair \(<m, X>\) is an interface
• Bigraphs have a formal graphical language
• But complete textual languages (kind of graph grammars) are available
• "People want pictures. Coding is for nerds."
Horizontal composition of bigraphs (tensor)

Bigraphs can be juxtaposed when interfaces do not share names

Given $G_1 : \langle m_1, X_1 \rangle \rightarrow \langle n_1, Y_1 \rangle$, $G_2 : \langle m_2, X_2 \rangle \rightarrow \langle n_2, Y_2 \rangle$
the horizontal composition is

$$G_1 \otimes G_2 : \langle m_1 + m_2, X_1 \uplus X_2 \rangle \rightarrow \langle n_1 + n_2, Y_1 \uplus Y_2 \rangle$$

given by disjoint union of nodes, edges, and maps (possibly with renamings).
Notice that:

1. Order is important: $G_1 \otimes G_2 \neq G_2 \otimes G_1$
2. No links are added between the two summands
3. No roots are merged
Vertical Composition of bigraphs

Bigraphs can be composed when interfaces are compatible
- subbigraphs' roots are grafted in sites (holes)
- names are wired to same names

\[ H = G \circ (F_1 \otimes F_2) \]

(What is the identity for composition?)
Bigraphs form a monoidal category

Given a signature $\Sigma$, $(\text{Big}(\Sigma), \otimes, I)$ is the monoidal category where

- objects of $\text{Big}(\Sigma)$: interfaces
- morphisms of $\text{Big}(\Sigma)$: bigraphs over the signature $\Sigma$
- composition is vertical composition
- $\otimes$ is horizontal composition

(Categories of place graphs and link graphs can be defined likewise)

Categories of bigraphs are akin Lawvere theories
How to encode a process algebra - syntax

Basic strategy

<table>
<thead>
<tr>
<th>Process Algebra</th>
<th>Bigraphs</th>
</tr>
</thead>
<tbody>
<tr>
<td>syntactic constructor with n variables</td>
<td>control with arity n</td>
</tr>
<tr>
<td>syntactic tree</td>
<td>place graph</td>
</tr>
<tr>
<td>variable</td>
<td>outer name</td>
</tr>
<tr>
<td>name</td>
<td>edge</td>
</tr>
</tbody>
</table>

Useful shorthands: nil is just "nothing", | (parallel) can be "omitted";
Example: CCS

Original syntax: \[ P ::= 0 \mid a.P \mid \bar{a}.P \mid P\mid Q \]

Bigraphs:
Signature has two controls: send, recv

Example: encoding of \[ \bar{a}.b.0\mid a.\bar{b}.a.0 \]
Modeling "informal" systems

This is a way to give a formal syntax to informal systems
Sortings

Often, bigraphs generated by a signature are too many.
**Sorting** = discipline for ruling out unwanted bigraphs

\[
\begin{align*}
\text{sorted bigraphs} & \quad = \quad \text{well-formed terms} \\
\text{unsorted bigraphs} & \quad = \quad \text{context-free terms}
\end{align*}
\]

Can be specified in several ways (e.g. predicates in some logic); see work by Hildebrandt, Debois, Perrone, ...
Example: binding bigraphs

Some ports of controls are marked as binding

Requirement over bigraphs:
“all points linked to a binding port of a node \( u \) lie inside \( u \)”

Example: encoding of \( \pi \)-calculus terms

\[
\bar{x}y. P | x(z). Q
\]

Diagram:

- Node 0 labeled “send” connected to node 1 labeled “get”
- Port labeled “y” connected to node 0
- Port labeled “x” connected to node 1
How does a bigraph evolve?

Graph rewriting: a sub-bigraph (redex) is replaced by another (reactum), with the same outer interface
A parametric (reaction) rule has the form

\[(R : \langle m, X \rangle \rightarrow J, R' : \langle n, X \rangle \rightarrow J, \rho : m \rightarrow n)\]

Given a set \(\mathcal{R}\) of reaction rules, the *reaction relation* \(\rightarrow\) is defined by

\[
(R, R', \rho) \in \mathcal{R} \quad \text{D active}
\]

\[
G = D \circ (id_Z \otimes R) \circ \bar{d}
\]

\[
G' = D \circ (id_Z \otimes R') \circ \rho(\bar{d})
\]

\[
G \rightarrow G'
\]

(A context \(D\) is *active* when its sites are only below active nodes. Active controls are indicated in the signature.)
Example reaction rules

\[ \pi\text{-calculus: } x\langle y \rangle.P | x(z).Q \rightarrow P|Q\{y/z\} \]

A wide rule: "long distance" communication
Example: vesicle formation
Matching of bigraphs

In the definition of reaction relations:

\[(R, R', \rho) \in \mathcal{R} \quad D \text{ active} \]
\[G = D \circ (id_Z \otimes R) \circ \vec{d} \]
\[G' = D \circ (id_Z \otimes R') \circ \rho(\vec{d}) \]
\[G \rightarrow G' \]

a key step:

- the matching problem: Given an agent \( G \) and a rule with redex \( R \), find all matchings of \( R \) inside the agent \( G \).

The matching problem is NP-complete, but it is exponential in the width of redexes, which is fixed for a given BRSs (and usually \( \leq 3 \))

Several algorithms have been proposed (inductive [Birkedal et al.], graph-based, with reduction to SAT [Sevegnani...], to CSP [MP2012]...
Execution policies

Once all matchings have been computed, how to choose that to be applied?

- Bigraphs are agnostic about the rewriting policy: can be non-deterministic, probabilistic, weighted, fair, etc.
- In fact, many variations have been developed. See e.g. Stochastic Bigraphs (for biological purposes).
- Non-interfering reactions can be executed concurrently
Metatheory & Tools
Deriving a good LTS

Often the semantics of a process algebra is given by means of a Labelled Transition System

\[ P \xrightarrow{a} Q \]

Useful for defining bisimilarity, model checking, etc.

**Problem:** how to define a good LTS for a process algebra?
*Good* = it induces a *compositional* bisimilarity

\[ P \sim Q \text{ iff for all } C[\_]: \; C[P] \sim C[Q] \]

In general, it is a difficult and error-prone task.
(cf. the LTS for mobile ambients)
Labels from contexts

In a BRS we can define labels for an agent as the *minimal* contexts (i.e. bigraphs) which are required to make a reaction

\[
\frac{C[G] \rightarrow G' \quad C[\ ] \text{ minimal}}{G \xrightarrow{C} G'}
\]

C[ ] is "what G is missing" to make a reaction

In bigraphs, minimality is formally given by the categorical notion of *idempotent pushout* (IPO).
Labels from reactions

**Theorem:** in a BRSs, the bisimilarity given by the LTS whose labels are defined by IPOs, is always a congruence.

Hence, in order to get a LTS with a compositional bisimilarity for a process algebra:

- Encode the process algebra as a BRS
- Calculate the IPO labels

Often the resulting bisimilarity coincides with the knew one
### Example: \( \pi \)-calculus

<table>
<thead>
<tr>
<th>( a )</th>
<th>( L )</th>
<th>( a' )</th>
<th>conditions</th>
<th>( \tilde{L} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( W \circ (\text{send}_{xy}</td>
<td>b) )</td>
<td>( \text{get}_x</td>
<td>\text{id}, (z)d \otimes \text{id} )</td>
<td>( a_0 )</td>
</tr>
<tr>
<td>( W \circ ((\text{get}_x \circ (z)d)</td>
<td>b) )</td>
<td>( \text{send}_{xy}</td>
<td>\text{id} )</td>
<td>( a_0' )</td>
</tr>
<tr>
<td>( W \circ (\text{send}_{xy}</td>
<td>(\text{get}_u \circ (z)d)</td>
<td>b) )</td>
<td>( x/y</td>
<td>\text{id} )</td>
</tr>
<tr>
<td>( W \circ (\text{send}_{xy}</td>
<td>(\text{get}_x \circ (z)d)</td>
<td>b) )</td>
<td>( \text{id} )</td>
<td>( a_0' )</td>
</tr>
</tbody>
</table>

where \( a_0' = W \circ (x | (y/z \circ d) | b) \)

![Diagram](image)

**Proposition:** The bisimilarity induced by IPOs coincides with the strong early bisimilarity.
Tools

Simulation tools:
- nondeterministic execution engines (BAM [Perrone et al.])
- stochastic engines (Gillespie-based [Danos, Krivine,...])
- distributed [Mansutti, Peressotti, M., 2014] (based on various algorithms for solving matching problem)

Model checkers [Perrone 2012, MPM in progress]

Graphical editors [Faithful, Hildebrandt]

...
LibBig: a Java Library for Bigraphs

An implementation of the machinery for defining and manipulating bigraphical reactive systems. Matching is implemented as a CSP. Easily extensible and adaptable to other variants

http://mads.dimii.uniud.it/wordpress/downloads/libbig/
Conclusions
Bigraphs are a good operational metamodel

- Bigraphical Reactive Systems are a general operational meta-model which can be instantiated to many models and systems

- Provides a theory of general results and tools
- Graphically oriented, yet rigorously defined in category theory
- Many ideas have been ported to other contexts (e.g. IPOs are used in PROP categories)

- In fact, the "bigraphical way of thinking" is often used as a guideline in the design and analysis of distributed systems
Other cool stuff we had not time to see here

- Categorical formulations
- Application to barbed equivalence
- Agent-based programming [Pereira et al, Mansutti et al.]
- Programming languages refinement and engineering [Perrone et al, Grohmann et al]
- Variants (directed, stochastic, typed, etc.)
- Generalization: multi-graphs
- Computational bigraphs [Debois & Milner]
- Spatial logic (BiLog [Conforti et al])
- ...
Still to come...

- General BRSs analysis using Abstract Interpretation techniques (e.g. CFA, termination, interference...)
- Further development of library and tools
- CTL*-like spatial-temporal logic
- Applications (especially in agent-oriented programming)
- Overall, the model can evolve in different ways, so **feedback is very welcome!**

http://bigraph.org
**Bigraphical Reactive Systems**

Marino Miculan
MADS lab
(with results by many people)

MeMo Workshop, June 6, 2014

---

**Bigraphical Reactive System**

A discrete reactive system is composed by a set of states and a transition relation.

A *Bigraphical Reactive System* is a RS where
- States are bigraphs: data structures rendering explicitly the positions and connections of system's components
- State transitions are bigraph rewritings defined by a set of local reaction rules

So BRSs propose as an operational metamodel.