Measuring the Expressiveness of Rewriting Systems through Event Structures

Part II: Normal Rewriting Systems

Damiano Mazza
Laboratoire d'Informatique de Paris Nord
CNRS–Université Paris 13

Concerto final workshop
Torino, 10 June 2010

Motivations

- Interaction nets (Lafont, 1990) are a model of deterministic computation, born as a generalization of linear logic proof nets (Girard, 1987).

- How expressive are they? They are Turing-complete... but this means nothing! What about parallelism?

- In addition, there are several non-deterministic variants:
  - *multiwire* (Alexiev 1999, Beffara-Maurel 2006);
  - *multiport* (Alexiev 1999, Khalil 2003, Mazza 2005);
  - *multirule* (Alexiev 1999, Ehrhard-Regnier 2006).

- How do these relate to each other? Can they model concurrency?

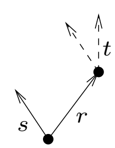
- We are not only interested in *what* we compute, but also *how*.

Rewriting systems

Rewriting systems are defined as pairs $\mathcal{S} = (\mathcal{G}, R)$, where $\mathcal{G}$ is a *graph*

$$\mathcal{G} = \quad \mathcal{G}_0 \underset{\mathrm{trg}}{\overset{\mathrm{src}}{\leftleftarrows}} \mathcal{G}_1$$

and R a *residue structure*, $i.e.$, a relation $R \subseteq \mathcal{G}_1^3$ such that $(r, s, t) \in R$ implies $\mathrm{src}(r) = \mathrm{src}(s)$ and $\mathrm{src}(t) = \mathrm{trg}(r)$:



In other words, a residue structure describes "what happens" to an arrow (called *radical*) if we follow a radical which is coinitial to it.

Pre-normal rewriting systems

- The notion of residue can be extended to *reductions*, i.e., the paths of $\mathcal{G}$: $[f]r$ is the set of residues of a radical r after the reduction f. We can then define *equivalence* of reductions: $f \rightleftharpoons g$ iff f and g are coinitial, cofinal, and for all coinitial r, $[f]r = [g]r$.

- A rewriting system is *pre-normal* if, for all coinitial radicals r, s:

  affinity: $\sharp[r]s \leq 1$; in case this set is a singleton, we denote its only element by s^r;

  symmetry: $\sharp[r]s = \sharp[s]r$; in case these sets are singletons, we say that r and s are *independent*;

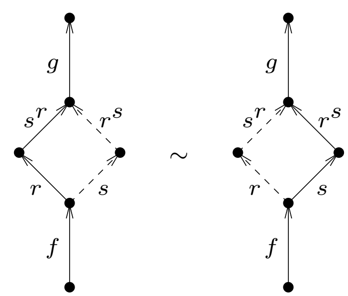
  tiling: $rs^r \rightleftharpoons sr^s$.

Homotopy

- Semi-normal rewriting systems allow the definition of *homotopy* as the smallest equivalence relation $\sim$ on reductions such that

$$f r s^r g \sim f s r^s g$$

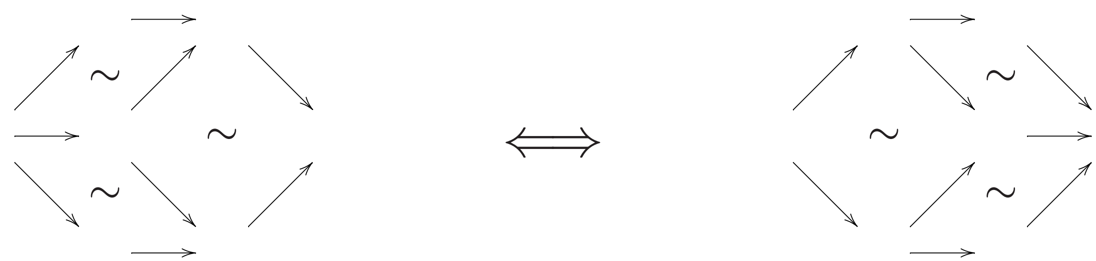
whenever r, s are independent radicals, and f, g are generic reductions:



- We then define the preorder $f \lesssim g$ iff $\exists h$ s.t. $fh \sim g$, which induces a partial order on homotopy classes: $[f] \leq [g]$ iff $f \lesssim g$.

The cube property

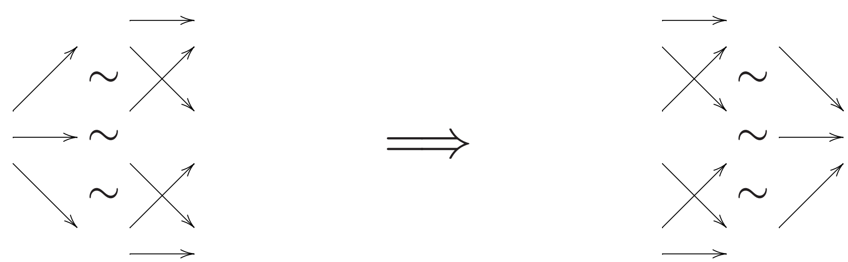
- A pre-normal rewriting system $\mathcal{S}$ is said to have the *cube property* if $\mathcal{S}$ contains the structure below on the left iff it contains the structure on the right:



- The terminology is borrowed from Mimram (2008). Previously studied also by Nielsen, Plotkin and Winskel (1981) (as *Mazurkiewicz traces*), and by Melliès (2004) (as *asynchronous graphs*).

The cubic pushout property

- A pre-normal rewriting system $\mathcal{S}$ is said to have the *cubic pushout property* if, whenever $\mathcal{S}$ contains the structure below on the left, it contains the structure on the right:



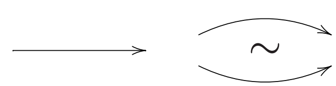
- Also considered by Nielsen, Plotkin and Winskel (1981).

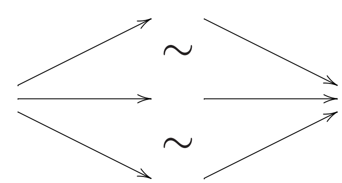
Normal rewriting systems

- A pre-normal rewriting system $\mathcal{S}$ is *normal* if it has the **cube property**, the **cubic pushout property**, and the following two additional axioms hold:

  self-conflict: for every radical r, $[r]r = \emptyset$;
  injectivity: for all radicals r, s, t with r, t and s, t independent, $r^t = s^t$ implies $r = s$.

- The following configurations are excluded in normal rewriting systems:





Normal rewriting systems and event structures

- We can prove the following:

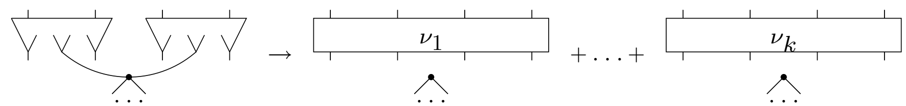
  Theorem 1. *Let $\mathcal{S}$ be a normal rewriting system, let μ be an object of $\mathcal{S}$, and let $\mathcal{H}_\mu(\mathcal{S})$ be the set of all homotopy classes of source μ. Then, $(\mathcal{H}_\mu(\mathcal{S}), \leq)$ is a configuration poset.*

- These results allow us to associate an event structure with every object of a normal rewriting system! Namely, we define $\mathrm{Ev}(\mu) = \Psi(\mathcal{H}_\mu(\mathcal{S}), \leq)$ (the interest of configuration posets is here).

- Therefore, as soon as two computational processes admit a description in terms of normal rewriting systems, we can use bisimilar embeddings to compare them.

Back to interaction nets

- We consider a general form of interaction nets which includes multirule, multiwire, and multiport extensions, all at the same time:



- Any interaction net system $\mathcal{S}$, with its reductions, induces a graph $\mathcal{G}_\mathcal{S}$: a radical is uniquely determined by an active pair, and a way to reduce it.

- The residue structure is defined by $(r, s, t) \in R_\mathcal{S}$ iff the active pairs associated with r, s belong to the same net, *have no cell in common*, and t is, by locality of interaction, "the same" radical as s after reducing r.

Proposition 2. *For every interaction nets system $\mathcal{S}$, $(\mathcal{G}_\mathcal{S}, R_\mathcal{S})$ is a normal rewriting system.*

Confusion-free rewriting systems

Let r, s be two coinitial radicals of a normal rewriting system.

- We say that r and s are *separated* if every radical t coinitial with r, s is independent with at least one of r, s.

- We say that r and s are *contemporary* if, for all radical r_0 and reduction h such that $r = r_0^h$, there exists a radical s_0 such that $s = s_0^h$. We say that r and s are in *simple conflict* if they are contemporary and not independent.

- A normal rewriting system $\mathcal{S}$ is *confusion-free* if all coinitial radicals are either separated or in simple conflict.

  Proposition 3. *A normal rewriting system $\mathcal{S}$ is confusion-free iff, for all object μ of $\mathcal{S}$, $\mathrm{Ev}(\mu)$ is confusion-free.*

Application to interaction nets

Proposition 4. *The rewriting system associated with a multirule interaction net system is always confusion-free.*

Corollary 5. *Multirule nets are strictly less expressive than multiwire and multiport nets. Moreover, there is no embedding of finite* CCS *in them.*

Lemma 6. *The rewriting system associated with a finite multirule or multiport interaction net system has finite degree of non-determinism.*

Corollary 7. *There is no finite universal system of multirule or multiport combinators not introducing divergence.*

There are also some positive results:

Proposition 8. *Lafont (*$i.e.$*, deterministic) interaction nets are able to generate all finite posets (*$i.e.$*, conflict-free event structures), and multirule interaction nets are able to generate all finite confusion-free event structures.*

Discussion

- How meaningful is all this? In other words:

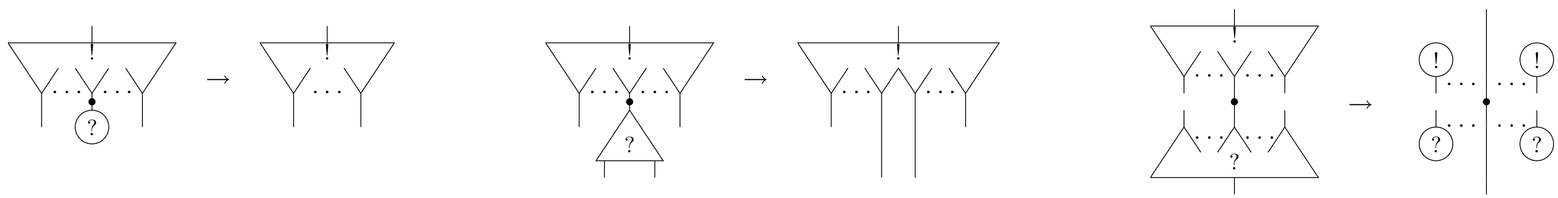
  (i) how many computational models can be rephrased in terms of normal rewriting systems?
  (ii) how sensible is our notion of bisimilar embedding?

- For (i), Turing machines, Petri nets, all process calculi can be seen as normal rewriting systems. However, the natural residue structure of the λ-calculus and proof nets is *not* pre-normal (affinity fails).

- For (ii), some well known encodings induce bisimilar embeddings ($e.g.$, Lafont's translations for interaction nets). However, there are surprises: apart from the problem with non-deterministic Turing machines seen in Part I, also the encodings of π-calculus into interaction nets do not work anymore.

Encoding the π-calculus in interaction nets

- A simple solution: turn differential interaction nets from multirule (which will *never* work, $cf.$ Corollary 5) into multiport:



- Ehrhard and Laurent's (2007) encoding, if reduced according to the above rules instead of the usual ones, yields a bisimilar embedding. This is what goes wrong with the usual reduction rules: