Modelling and Using Application-Level QoS

Dan Hirsch **Alberto Lluch-Lafuente**
{dhirsch,lafuente}@di.unipi.it

Dipartimento di Informatica,
Università di Pisa
Italy

Emilio Tuosto
et52@mcs.le.ac.uk

Dept. of Computer Science
University of Leicester
UK

Work supported by
EC projects FET SENSORIA IST-2005-16004

&

RTN SEGRAVIS 2-2001-00346

Motivations

What do we mean for application level QoS?

- Distributed computing is moving toward SOC
- Integration of software and (heterogeneous) networks of (heterogeneous) systems (e.g., Internet & mobile phones, wireless & wired networks)

- Distributed computing is moving toward SOC
- Integration of software and (heterogeneous) networks of (heterogeneous) systems (e.g., Internet & mobile phones, wireless & wired networks)

- SOC architectures are
  - distributed
  - interconnected
  - based on different communication infrastructures:
    - IP, wireless, satellites...
    - overlay networks
  - Designers, programmers and end-users may ignore the stratification and complexity

- Distributed computing is moving toward SOC
- Integration of software and (heterogeneous) networks of (heterogeneous) systems (e.g., Internet & mobile phones, wireless & wired networks)

- SOC architectures are
  - distributed
  - interconnected
  - based on different communication infrastructures:
    - IP, wireless, satellites...
    - overlay networks
  - Designers, programmers and end-users may ignore the stratification and complexity

- SOC applications (SOAs) are soups of services
  - programmable coordination
  - "autonomous"
  - independent
  - mobile/stationary
  - "interconnected" through interfaces

  and published, searched and binded ... offline and in a mostly ad-hoc way

What do we mean for application level QoS?

- Distributed computing is moving toward SOC
- Integration of software and (heterogeneous) networks of (heterogeneous) systems (e.g., Internet & mobile phones, wireless & wired networks)

- SOC architectures are
  - distributed
  - interconnected
  - based on different communication infrastructures:
    - IP, wireless, satellites...
    - overlay networks
  - Designers, programmers and end-users may ignore the stratification and complexity

- SOC applications (SOAs) are soups of services
  - programmable coordination
  - "autonomous"
  - independent
  - mobile/stationary
  - "interconnected" through interfaces

  and published, searched and binded … offline and in a mostly ad-hoc way

- May we search & glue services dynamically?
- What should drive service searching?

Lifting QoS issues to applications

- with programmable application level QoS
- QoS in designing and implementing SOAs

Lifting QoS issues to applications

- with programmable application level QoS
- QoS in designing and implementing SOAs

Search and bind wrt application level QoS

- application-related, e.g.
  - price
  - payment mode
  - transactions
  - data available in a given format
- low-level related (e.g., throughput, response time) not directly referred but abstracted for expressing how they are "perceived" at application level

Lifting QoS issues to applications

- with programmable application level QoS
- QoS in designing and implementing SOAs

Search and bind wrt application level QoS

- application-related, e.g.
  - price
  - transactions
  - payment mode
  - data available in a given format
- low-level related (e.g., throughput, response time) not directly referred but abstracted for expressing how they are "perceived" at application level

We uses c-semiring as

- a synchronisation mechanism...
- and for interpreting a logic!

An algebraic structure $\langle S, +, \star, \mathbf{0}, \mathbf{1} \rangle$ is a c-semiring iff $\mathbf{0} \neq \mathbf{1} \in S$, and

$$\begin{aligned} x + y &= y + x \\ (x + y) + z &= x + (y + z) \\ (x + y) \star z &= (x \star z) + (y \star z) \\ x + \mathbf{1} &= \mathbf{1} \end{aligned}$$

$$\begin{aligned} x \star y &= y \star x \\ (x \star y) \star z &= x \star (y \star z) \\ x \star \mathbf{1} &= x \\ x \star \mathbf{0} &= \mathbf{0} \end{aligned}$$

An algebraic structure $\langle S, +, \star, \mathbf{0}, \mathbf{1}\rangle$ is a c-semiring iff $\mathbf{0} \neq \mathbf{1} \in S$, and

$$x + y = y + x$$
$$(x + y) + z = x + (y + z)$$
$$(x + y) \star z = (x \star z) + (y \star z)$$
$$x + \mathbf{1} = \mathbf{1}$$

$$x \star y = y \star x$$
$$(x \star y) \star z = x \star (y \star z)$$
$$x \star \mathbf{1} = x$$
$$x \star \mathbf{0} = \mathbf{0}$$

- Implicit partial order: $a \leq b \iff a + b = b$ "b is better than a"
- c-semirings preserved by many mathematical constructions

Constraint semirings

An algebraic structure $\langle S, +, \star, \mathbf{0}, \mathbf{1} \rangle$ is a c-semiring iff $\mathbf{0} \neq \mathbf{1} \in S$, and

$$x + y = y + x$$
$$(x + y) + z = x + (y + z)$$
$$(x + y) \star z = (x \star z) + (y \star z)$$
$$x + \mathbf{1} = \mathbf{1}$$

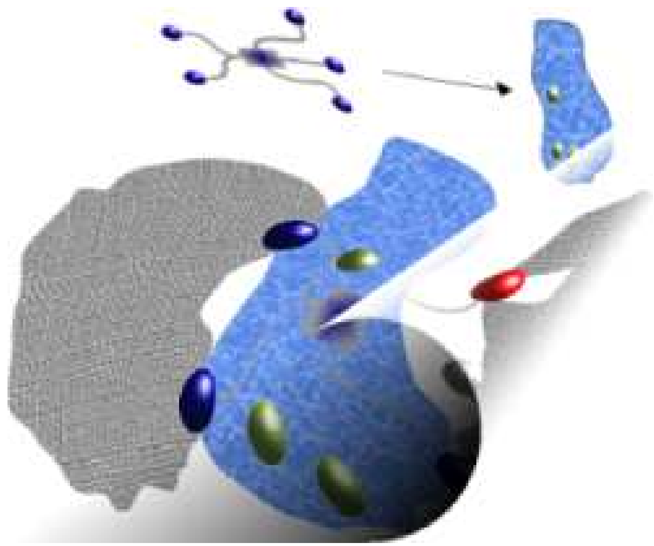
$$x \star y = y \star x$$
$$(x \star y) \star z = x \star (y \star z)$$
$$x \star \mathbf{1} = x$$
$$x \star \mathbf{0} = \mathbf{0}$$

- Implicit partial order: $a \leq b \iff a + b = b$ "b is better than a"
- c-semirings preserved by many mathematical constructions

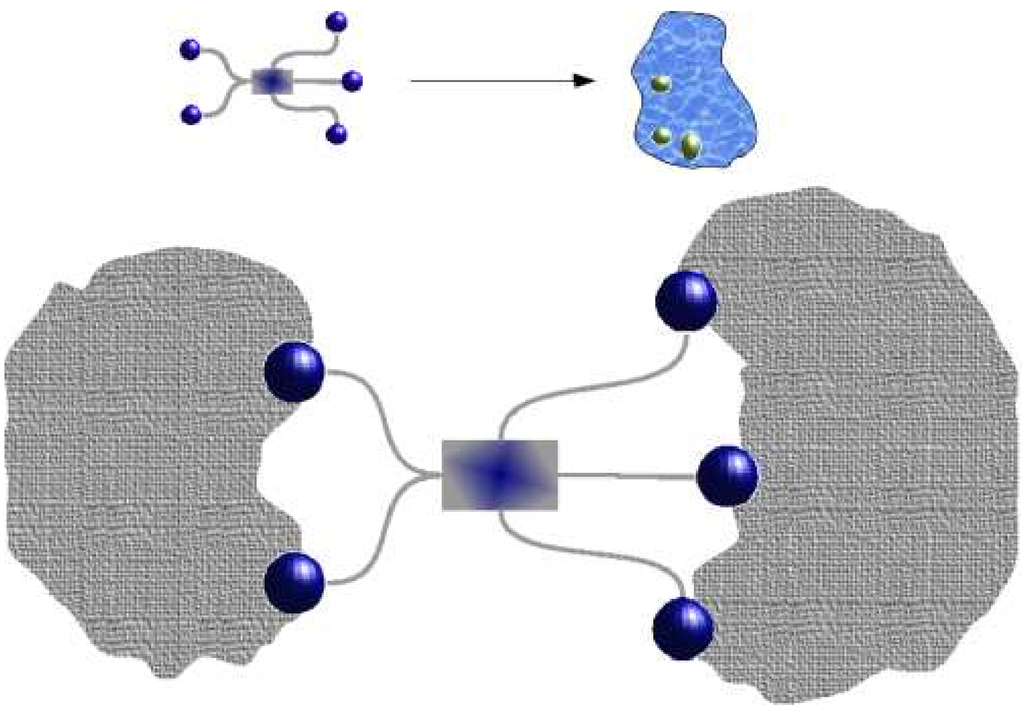
Examples

- $\langle \mathrm{Real}+, max, min, 0, +\infty \rangle$ (max/min): bandwidth, priority
- $\langle \{true, false\}, \vee, \wedge, false, true \rangle$ (boolean): availability
- $\langle \mathrm{Real}+, min, +, +\infty, 0 \rangle$ (optimization): price, propagation delay
- $\langle [0, 1], max, \cdot, 0, 1 \rangle$ (probabilistic): performance and rates
- $\langle [0, 1], max, min, 0, 1 \rangle$ (fuzzy): performance and rates
- $\langle 2^N, \cup, \cap, \emptyset, N \rangle$ (set-based, where N is a set): capabilities and access rights

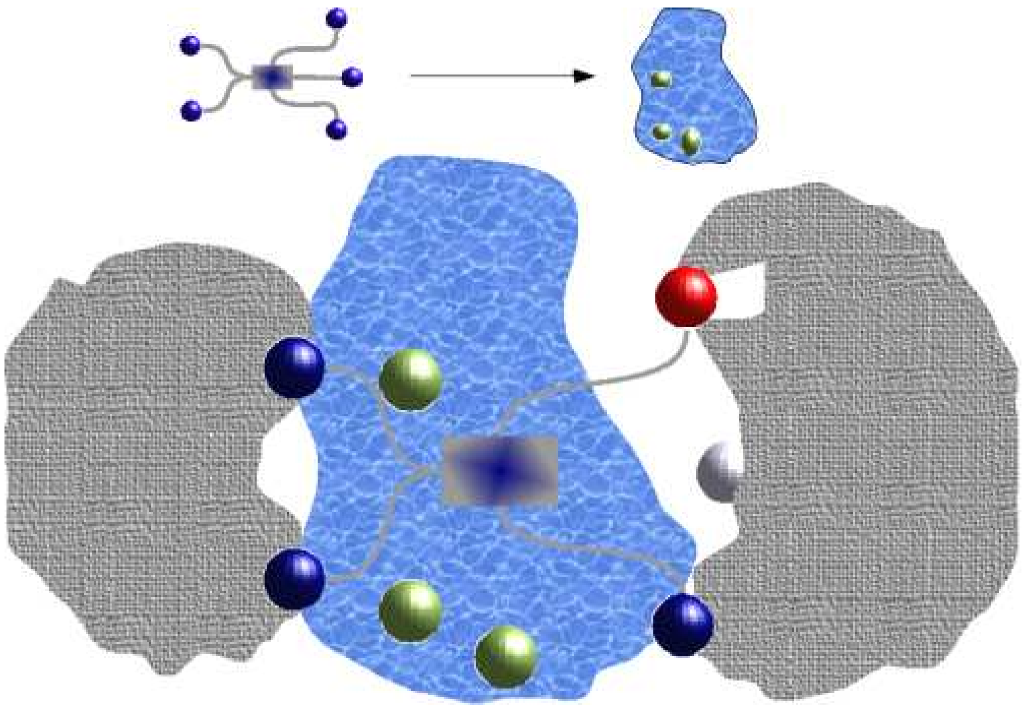
Synchornised Hyperedge Replacement



Synchronised Hyperedge Replacement

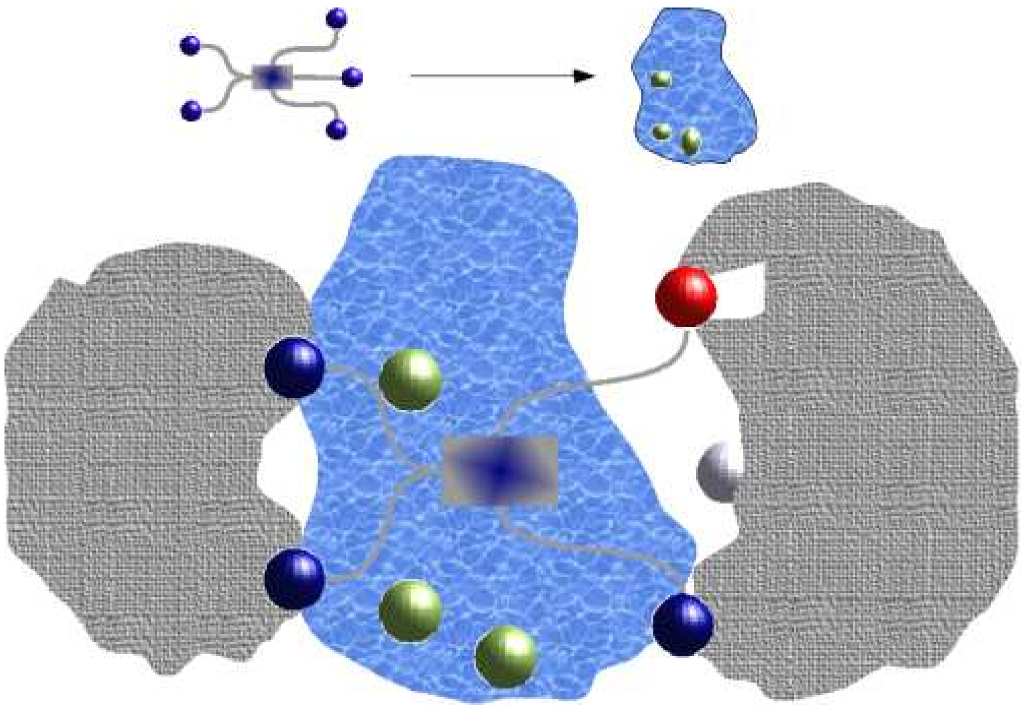


Synchronised Hyperedge Replacement



- Edge replacement: local
- Synchronisation as distributed constraint solving
- Multi-party synchronisation
- New node creation
- Node fusion: model of mobility and communication

Synchronised Hyperedge Replacement



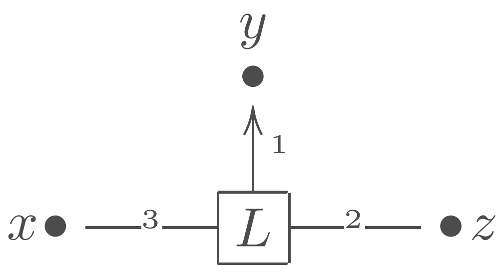
- Edge replacement: local
- Synchronisation as distributed constraint solving
- Multi-party synchronisation
- New node creation
- Node fusion: model of mobility and communication

Benefits:

- Uniform framework
  - for π, π-I, fusion [Tuo03, Hir03, LM04]
  - LTS for Ambient ... [FMT01]
- Expressive for
  - ... for Klaim ... [DFM$^+$03]
  - "sophisticated synchronisations" [HT05, HLT05, LT05]

Given nodes $\mathcal{N}$, hyperedges connect any number of nodes (generalisation of edge)

$$L : 3,\ \ L(y, z, x),$$

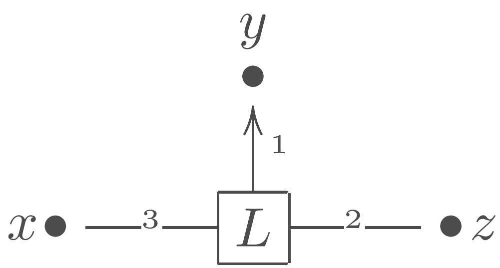


$$\begin{array}{lcl} G & ::= & nil \\ & | & L(\tilde{x}) \\ & | & G|G \end{array}$$

Syntactic Judgement $\quad x_1 : s_1, \ldots, x_n : s_n \vdash G, \qquad \mathrm{n}(G) \subseteq \{x_1, \ldots, x_n\}$

Given nodes $\mathcal{N}$, hyperedges connect any number of nodes (generalisation of edge)

$L : 3, \;\; L(y, z, x),$



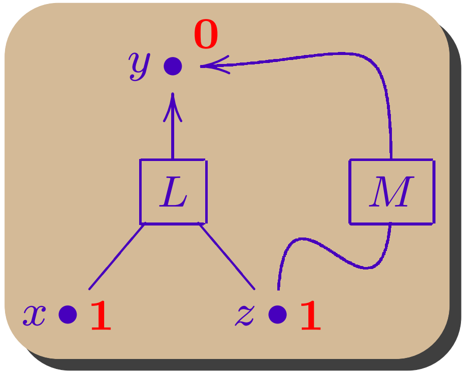
$$\begin{array}{lcl} G & ::= & nil \\ & | & L(\tilde{x}) \\ & | & G|G \end{array}$$

Syntactic Judgement $\quad x_1 : s_1, \ldots, x_n : s_n \vdash G, \qquad \mathrm{n}(G) \subseteq \{x_1, \ldots, x_n\}$

An example:

$$L : 3, \quad M : 2$$

$$x : \mathbf{1}, y : \mathbf{0}, z : \mathbf{1} \vdash (L(y, z, x) \mid M(y, z))$$



SHReQ

Productions are the context free rules upon which hypergraph rewriting is defined

production $\chi \triangleright L(\tilde{x}) \xrightarrow{\Lambda} G$

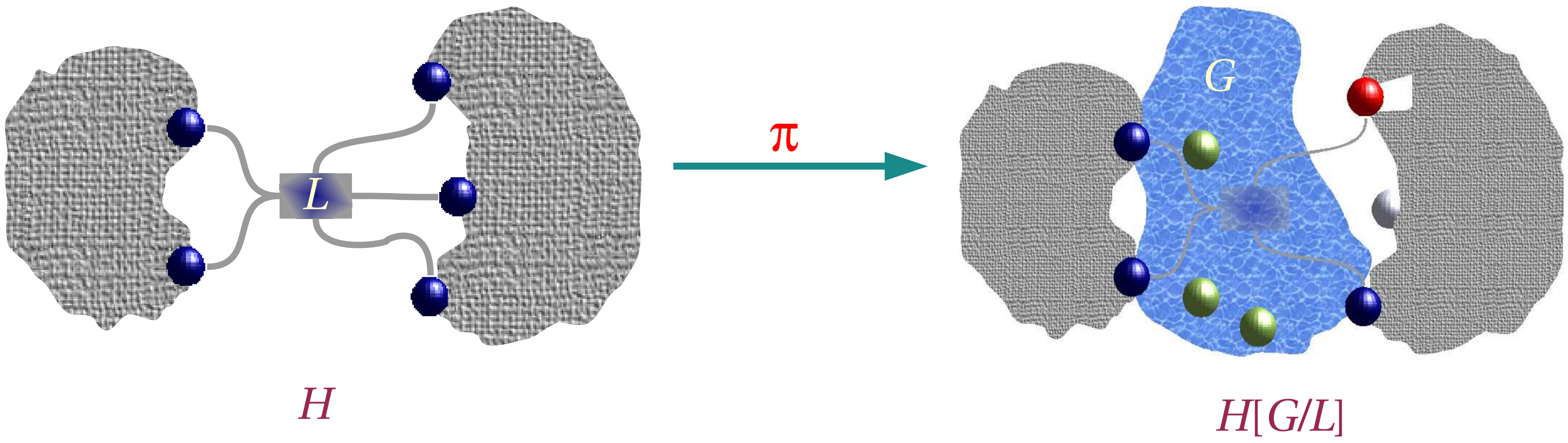
Once χ is satisfied, $L(\tilde{x})$ rewrites as G synchronising with the edges connected to nodes $\tilde{x}$ according to Λ

- $\tilde{x}$ is a tuple of pairwise distinguished nodes and $L : |\tilde{x}|$
- $\chi : \{\!|\tilde{x}|\!\} \to S$ is a weighting function
  - eache node in the interface has an associated c-semiring value...
  - that will "drive" synchronisations
- communication function $\Lambda : \{\!|\tilde{x}|\!\} \to \mathcal{R}$ associates requirements to the interface of L
  - $\mathcal{R} = S \times \mathcal{N}^*$ is the set of events, where S is an alphabet of actions
  - $\mathrm{n}(\Lambda) = \{z \mid \exists x \in \mathrm{dom}(\Lambda).z \in \Lambda(x)\}$
- G is a graph s.t. $\mathrm{n}(G) \subseteq \{\!|\tilde{x}|\!\} \cup \mathrm{n}(\Lambda)$

Consider a production

$$\pi : \chi \triangleright L(\tilde{x}) \xrightarrow{\Lambda} G$$

and a graph H having an arc labelled by L, e.g.:



- Replacing L with G in H according to π requires that H satisfies the conditions expressed by χ on the attachment nodes of L

- Once χ is satisfied in H, $L(\tilde{x})$ participate to the rewriting by “offering” Λ in the synchronisation with all the edges connected to nodes in $\tilde{x}$

Consider

$$x_1 : u_1, x_2 : u_2, x_3 : u_3 \triangleright L(x_2, x_3, x_1) \xrightarrow{\substack{(x_1, r, \langle x_1, x_3 \rangle),\\ (x_2, s, \langle\rangle),\\ (x_3, t, \langle\rangle)}} nil$$

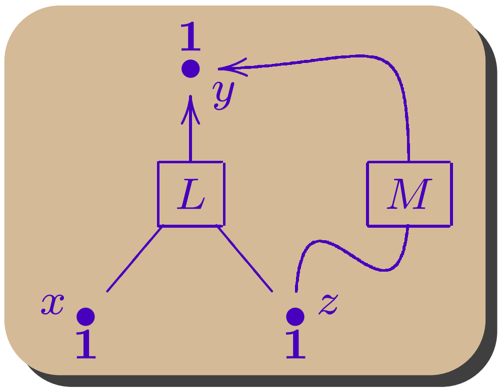
$$x' : v_1, y' : v_2 \triangleright M(x', y') \xrightarrow{\substack{(x', r', \langle x', x' \rangle),\\ (y', \mathbf{1}, \langle\rangle)}} L(x', y', y')$$

Consider

$$x_1 : u_1, x_2 : u_2, x_3 : u_3 \triangleright L(x_2, x_3, x_1) \xrightarrow{\substack{(x_1, r, \langle x_1, x_3\rangle),\\ (x_2, s, \langle\rangle),\\ (x_3, t, \langle\rangle)}} nil$$

$$x' : v_1, y' : v_2 \triangleright M(x', y') \xrightarrow{\substack{(x', r', \langle x', x'\rangle),\\ (y', \mathbf{1}, \langle\rangle)}} L(x', y', y')$$

Let'u apply these productions to the hypergraph

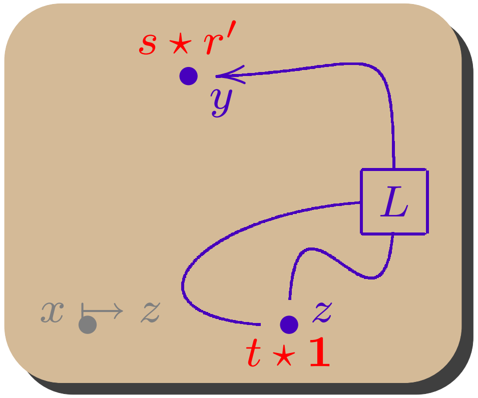


Consider

$$x_1 : u_1, x_2 : u_2, x_3 : u_3 \rhd L(x_2, x_3, x_1) \xrightarrow{\begin{array}{c}(x_1, r, \langle x_1, x_3\rangle),\\ (x_2, s, \langle\rangle),\\ (x_3, t, \langle\rangle)\end{array}} nil$$

$$x' : v_1, y' : v_2 \rhd M(x', y') \xrightarrow{\begin{array}{c}(x', r', \langle x', x'\rangle),\\ (y', \mathbf{1}, \langle\rangle)\end{array}} L(x', y', y')$$

Let'u apply these productions to the hypergraph



C-semiring values for

Synchronisation

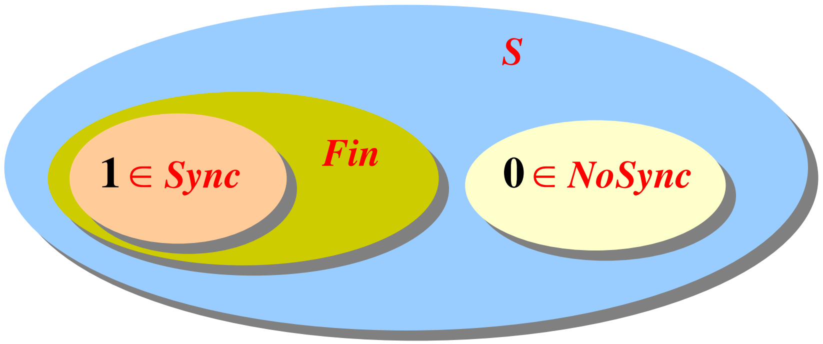
$Sync$ and Fin s.t.

- $Sync \subseteq Fin \subseteq S$
- $\mathbf{1} \in Sync$

No synchronisation

$NoSync \subseteq S \setminus Fin$ s.t.

- $S \star NoSync \subseteq NoSync$
- $\mathbf{0} \in NoSync$



Synchronising with c-semirings (I)

C-semiring values for

Synchronisation

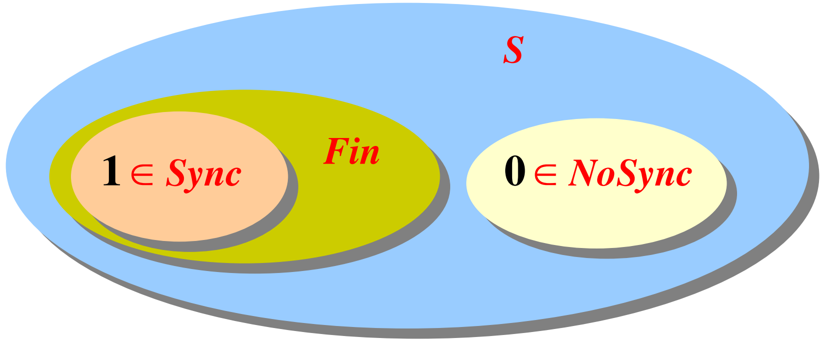
$Sync$ and Fin s.t.

- $Sync \subseteq Fin \subseteq S$
- $\mathbf{1} \in Sync$

No synchronisation

$NoSync \subseteq S \setminus Fin$ s.t.

- $S \star NoSync \subseteq NoSync$
- $\mathbf{0} \in NoSync$



The Hoare c-semiring on $\mathcal{H} = Act \cup \{\mathbf{1}_H, \mathbf{0}_H, \perp\}$ (where Act is a set of actions) is specified according to

- $\forall a \in \mathcal{H}.\ a +_H a = a$,
  $\forall a, b \in Act \cup \{\perp\}.\ b \neq a \implies a +_H b = \perp$
  plus the c-semiring axioms for the sum
- $\forall a \in Act.\ a \star_H a = a$
  $\forall a, b \in Act \cup \{\perp\} : b \neq a \implies a \star_H b = \perp$
- plus commutative rules and the ones for $\mathbf{0}$ and $\mathbf{1}$

Hoare synchronisations take place only when all interacting components agree on their actions. This is reflected in the Hoare c-semiring multiplicative equations.

Let Ω be a finite multiset over $\mathcal{N} \times \mathcal{R}$, **mgu** Ω is an idempotent substitution which is an mgu of

Synchronising with c-semirings (II)

Let Ω be a finite multiset over $\mathcal{N} \times \mathcal{R}$, **mgu** Ω is an idempotent substitution which is an mgu of

$$\left\{ \tilde{u}_i = \tilde{v}_i \quad \middle| \quad \begin{array}{l} \exists (x, s, \tilde{u}), (x, t, \tilde{v}) \in \Omega : \\ \\ 1 \leq i \leq |\tilde{u}| \end{array} \right\}$$

defined iff

$$\|\Omega @ x\| > 1 \implies \prod_{(x,s,\tilde{y}) \in \Omega @ x} s \notin NoSync$$

Let Ω be a finite multiset over $\mathcal{N} \times \mathcal{R}$, **mgu** Ω is an idempotent substitution which is an mgu of

$$\left\{ \tilde{u}_i = \tilde{v}_i \;\middle|\; \begin{array}{l} \exists (x, s, \tilde{u}), (x, t, \tilde{v}) \in \Omega : \\ 1 \leq i \leq |\tilde{u}| \end{array} \right\}$$

$$\begin{array}{cccccc} (x, & s_1, & u_1^1, & \cdots \; u_i^1 & \cdots & u_n^1) \\ & \vdots & & \vdots & & \vdots \\ (x, & s_m, & u_1^m, & \cdots \; u_i^m & \cdots & u_n^m) \end{array}$$

defined iff

$$\|\Omega @ x\| > 1 \implies \prod_{(x,s,\tilde{y}) \in \Omega @ x} s \notin NoSync$$

Synchronising with c-semirings (II)

Let Ω be a finite multiset over $\mathcal{N} \times \mathcal{R}$, **mgu** Ω is an idempotent substitution which is an mgu of

$$\left\{ \tilde{u}_i = \tilde{v}_i \quad \middle| \quad \begin{array}{l} \exists (x, s, \tilde{u}), (x, t, \tilde{v}) \in \Omega : \\ \\ 1 \leq i \leq |\tilde{u}| \end{array} \right\}$$

defined iff

$$\|\Omega @ x\| > 1 \implies \prod_{(x,s,\tilde{y}) \in \Omega @ x} s \notin NoSync$$

$$\begin{array}{ccccccc}
(x, & s_1, & u_1^1, & \cdots & u_i^1 & \cdots & u_n^1) \\
 & \star & \downarrow & & \downarrow & & \downarrow \\
 & \vdots & & \vdots & & \vdots & \\
 & & \downarrow & & \downarrow & & \downarrow \\
 & \star & & & & & \\
(x, & s_m, & u_1^m, & \cdots & u_i^m & \cdots & u_n^m) \\
 & \notin NoSync & u_1 & & u_i & & u_n
\end{array}$$

Let Ω be a finite multiset over $\mathcal{N} \times \mathcal{R}$, **mgu** Ω is an idempotent substitution which is an mgu of

$$\left\{ \tilde{u}_i = \tilde{v}_i \;\middle|\; \begin{array}{c} \exists (x, s, \tilde{u}), (x, t, \tilde{v}) \in \Omega : \\ \\ 1 \leq i \leq |\tilde{u}| \end{array} \right\}$$

defined iff

$$\|\Omega@x\| > 1 \implies \prod_{(x,s,\tilde{y}) \in \Omega@x} s \notin NoSync$$

$$\begin{array}{cccccccc}
(x, & s_1, & u_1^1, & \cdots & u_i^1 & \cdots & u_n^1) \\
 & \star & \downarrow & & \downarrow & & \downarrow \\
 & \vdots & & \vdots & & \vdots & \\
 & \star & \downarrow & & \downarrow & & \downarrow \\
(x, & s_m, & u_1^m, & \cdots & u_i^m & \cdots & u_n^m) \\
 & \notin NoSync & u_1 & & u_i & & u_n
\end{array}$$

The communication function induced by Ω is the function $\underline{\Omega} : \mathrm{dom}(\Omega) \to \mathcal{R}$ defined as

$$\underline{\Omega}(x) = \begin{cases} (t, \tilde{y}\rho), & t = \prod_{(x,s,\tilde{y}) \in \Omega@x} s \notin Sync, \quad \rho = \textbf{mgu}\ \Omega \\ \\ (t, \langle\rangle), & t = \prod_{(x,s,\tilde{y}) \in \Omega@x} s \in Sync \end{cases}$$

Basically, $\underline{\Omega}(x)$ yields the synchronisation of requirements in $\Omega@x$ according to the c-semiring product.

Synchronising with c-semirings (III)

The weighting function induced by Γ and Ω is the function $\Gamma_\Omega : \mathrm{dom}(\Gamma) \to S$ s.t.

$$\Gamma_\Omega(x) = \begin{cases} \mathbf{1}, & x \in \mathrm{new}(\underline{\Omega}) \\ \Gamma(x), & \|\Omega @ x\| = 1 \\ \boxed{\Gamma_\Omega(x) = \underline{\Omega}(x){\downarrow_1}}, & \text{otherwise} \end{cases}$$

The weighting function computes the new weights of graphs after the synchronisations induced by Ω.

The weighting function induced by Γ and Ω is the function $\Gamma_\Omega : \mathrm{dom}(\Gamma) \to S$ s.t.

$$\Gamma_\Omega(x) = \begin{cases} \mathbf{1}, & x \in \mathrm{new}(\underline{\Omega}) \\ \Gamma(x), & \|\Omega @ x\| = 1 \\ \boxed{\Gamma_\Omega(x) = \underline{\Omega}(x){\downarrow_1}}, & \text{otherwise} \end{cases}$$

The weighting function computes the new weights of graphs after the synchronisations induced by Ω.

rewriting system: $(\mathcal{QP}, \Gamma \vdash G)$

where $\mathcal{QP}$ is the set of quasi-productions on $\mathcal{P}$ and is s.t.

- $\mathcal{P} \subseteq \mathcal{QP}$
- $\chi \triangleright L(\tilde{x}) \xrightarrow{\Omega} G \in \mathcal{QP} \;\wedge\; y \in \mathcal{N} \setminus \mathrm{new}(\Omega) \implies \chi' \triangleright L(\tilde{x}\{^y/_x\}) \xrightarrow{\Omega\{^y/_x\}} G\{^y/_x\} \in \mathcal{QP}$

where

$$\chi' : \{\!|\tilde{x}|\!\} \setminus \{x\} \cup \{y\} \to S \qquad \chi'(z) = \begin{cases} \chi(z), & z \in \{\!|\tilde{x}|\!\} \setminus \{x, y\} \\ \chi(x) + \chi(y), & z = y \wedge y \in \tilde{x} \\ \chi(x), & z = y \wedge y \notin \{\!|\tilde{x}|\!\} \end{cases}$$

$$\frac{\chi \triangleright L(\tilde{x}) \xrightarrow{\Omega} G \in \mathcal{QP} \qquad \rho = \mathbf{mgu}\ \Omega \qquad \bigwedge_{x \in \mathrm{dom}(\chi)} \chi(x) \leq \Gamma(x)}{\Gamma \vdash L(\tilde{x}) \xrightarrow{\Omega} \Gamma_\Omega \vdash G\rho}$$

$$\frac{\Gamma_1 \vdash G_1 \xrightarrow{\Omega_1} \Gamma'_1 \vdash G'_1 \quad \Gamma_2 \vdash G_2 \xrightarrow{\Omega_2} \Gamma'_2 \vdash G'_2 \quad \rho = \mathbf{mgu}\ \Omega \qquad \bigwedge_{x \in \mathrm{dom}(\Gamma_1) \cap \mathrm{dom}(\Gamma_2)} \Gamma_1(x) = \Gamma_2(x)}{\Gamma_1 \cup \Gamma_2 \vdash G_1 \mid G_2 \xrightarrow{\Omega} (\Gamma_1 \cup \Gamma_2)_\Omega \vdash G'_1 \mid G'_2\rho}$$

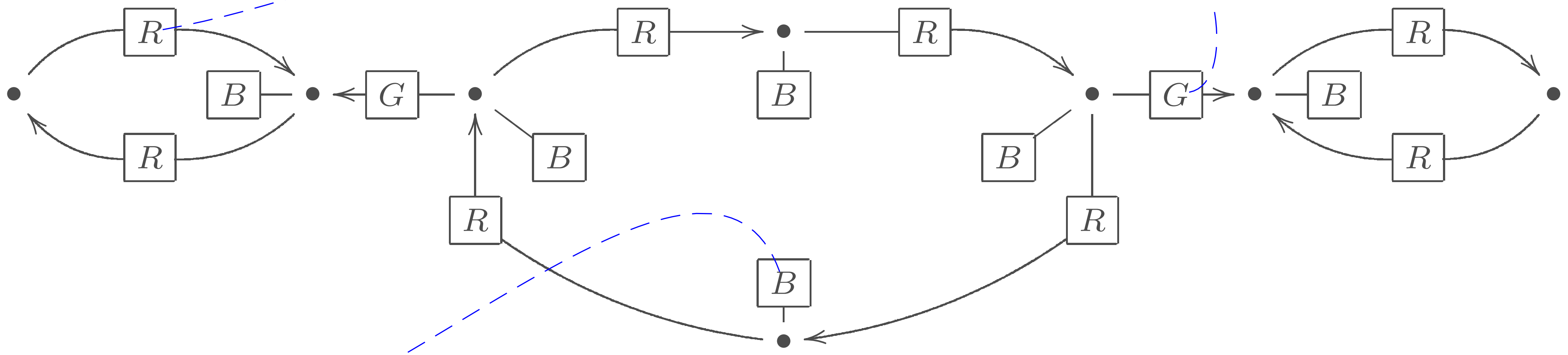
where $\Omega = \Omega_1 \sqcup \Omega_2$

$$\frac{\chi \triangleright L(\tilde{x}) \xrightarrow{\Omega} G \in \mathcal{QP} \qquad \rho = \mathbf{mgu}\ \Omega \qquad \bigwedge_{x \in \mathrm{dom}(\chi)} \chi(x) \leq \Gamma(x)}{\Gamma \vdash L(\tilde{x}) \xrightarrow{\Omega} \Gamma_\Omega \vdash G\rho}$$

$$\frac{\Gamma_1 \vdash G_1 \xrightarrow{\Omega_1} \Gamma'_1 \vdash G'_1 \quad \Gamma_2 \vdash G_2 \xrightarrow{\Omega_2} \Gamma'_2 \vdash G'_2 \quad \rho = \mathbf{mgu}\ \Omega \qquad \bigwedge_{x \in \mathrm{dom}(\Gamma_1) \cap \mathrm{dom}(\Gamma_2)} \Gamma_1(x) = \Gamma_2(x)}{\Gamma_1 \cup \Gamma_2 \vdash G_1 \mid G_2 \xrightarrow{\Omega} (\Gamma_1 \cup \Gamma_2)_\Omega \vdash G'_1 \mid G'_2\rho}$$

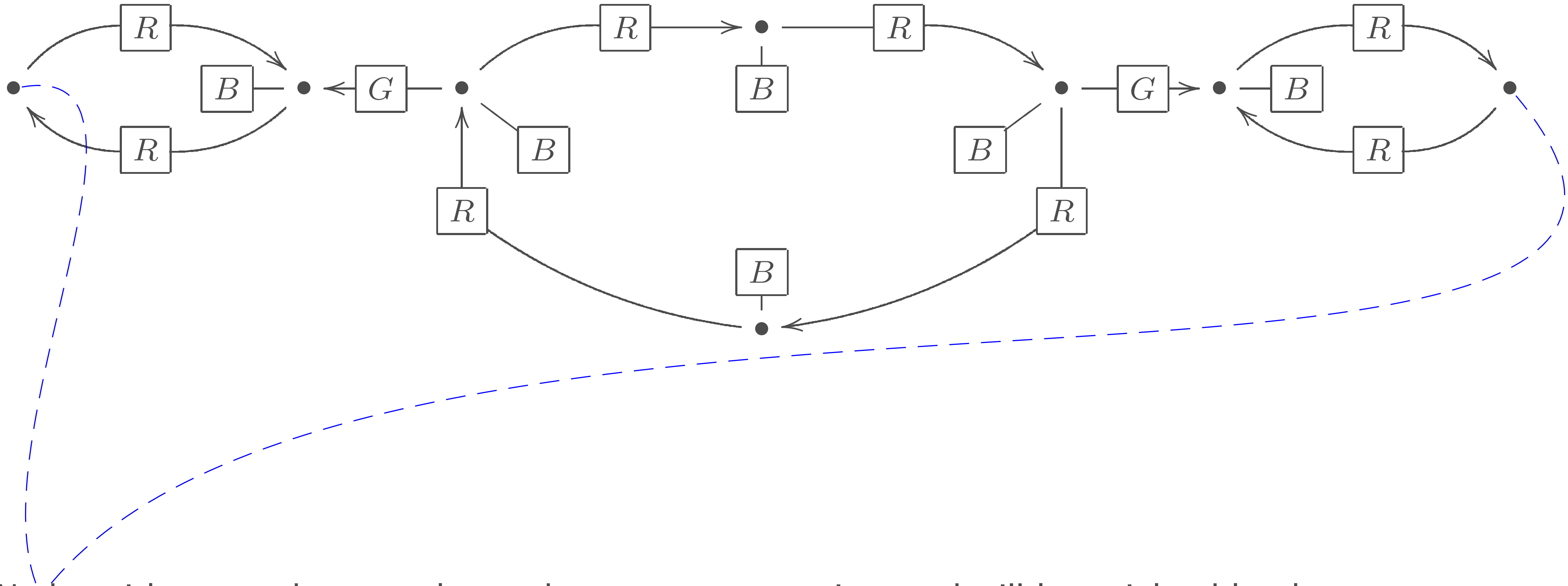
where $\Omega = \Omega_1 \sqcup \Omega_2$

A network of rings consists of “rings” of different sizes connected by gates.



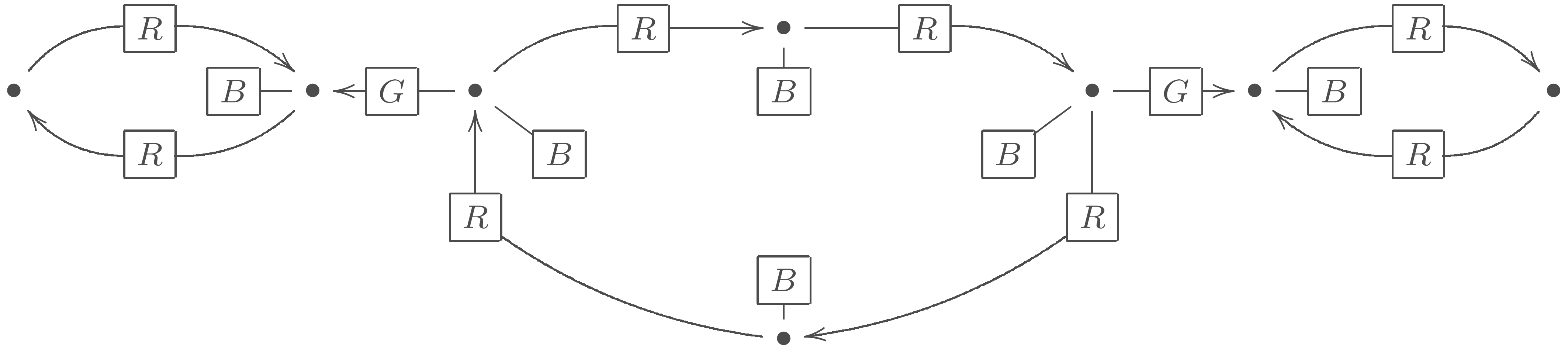
The B-edges avoid new gates to be attached on the node they insist on

A network of rings consists of "rings" of different sizes connected by gates.



Nodes with no B-edges, can be used to generate new rings and will be weighted by the amount of available resource.

A network of rings consists of “rings” of different sizes connected by gates.

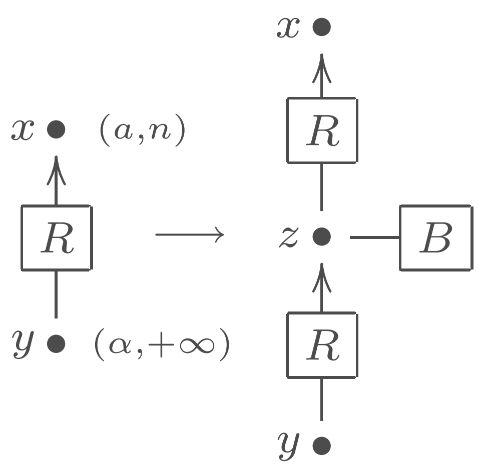


The c-semiring is $\mathfrak{H}\mathfrak{R}$, the cartesian product of the Hoare c-semiring $\mathfrak{H} = \langle \mathcal{H}, +_H, \star_H, \mathbf{0}_H, \mathbf{1}_H \rangle$ (where $\mathcal{H} = \{a, b, c, \mathbf{1}_H, \mathbf{0}_H, \perp\}$) and $\mathfrak{R} = \langle \omega_\infty, max, min, 0, +\infty \rangle$
The idea is that

- $\mathfrak{H}$ coordinates the network rewritings
- $\mathfrak{R}$ handles resource availability

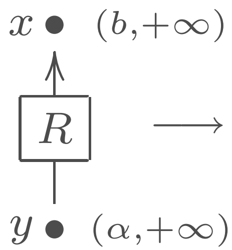
Productions for the ring case study

Create Brother $(n < u)$



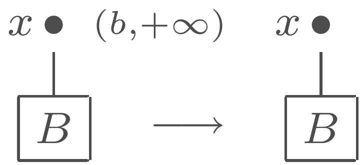
$$x : (\mathbf{0}_H, u), y : \mathbf{0} \triangleright R(x,y) \xrightarrow{\substack{x \,\mapsto\, (a,\, n) \\ y \,\mapsto\, (\alpha,\, +\infty)}} R(x,z) \mid R(z,y) \mid B(z)$$

Accept Syncrhonisation R



$$x : (\mathbf{0}_H, +\infty), y : \mathbf{0} \triangleright R(x,y) \xrightarrow{\substack{x \,\mapsto\, (b,\, +\infty) \\ y \,\mapsto\, (\alpha,\, +\infty)}} R(x,y)$$

Accept Syncrhonisation B

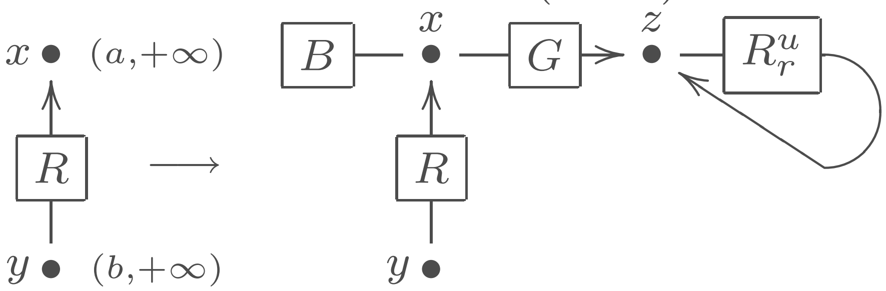


$$x : \mathbf{0} \triangleright B(x) \xrightarrow{x \mapsto (b, +\infty)} B(x)$$

where $\alpha \in \{a, b\}$

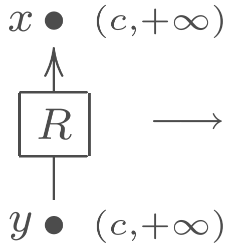
Productions for the ring case study[2]

Create Gate $(r > 0)$



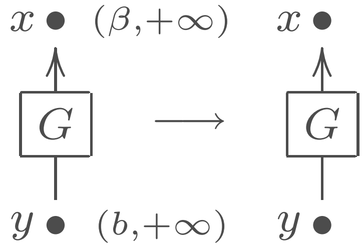
$$x : (\mathbf{0}_H, u), y : \mathbf{0} \triangleright R(x,y) \xrightarrow{\substack{x \,\mapsto\, (a,\, +\infty) \\ y \,\mapsto\, (b,\, +\infty)}} R(x,y) \mid B(x) \mid G(z,x) \mid R_r^u(z,z)$$

Accept Synchronisation Init



$$x : \mathbf{0}, y : \mathbf{0} \triangleright R(x,y) \xrightarrow{\substack{x \,\mapsto\, (c,\, +\infty) \\ y \,\mapsto\, (c,\, +\infty)}} R(x,y)$$

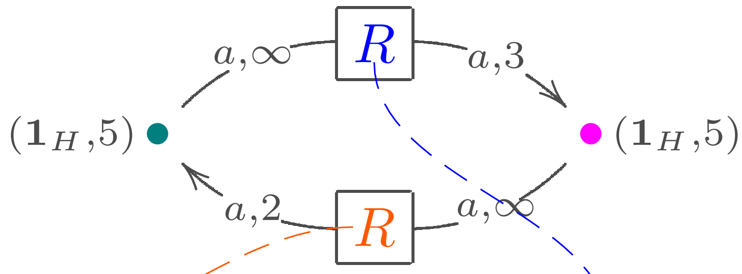
Accept Synchronisation Gate



$$x : \mathbf{0}, y : \mathbf{0} \triangleright G(x,y) \xrightarrow{\substack{x \,\mapsto\, (\beta,\, +\infty) \\ y \,\mapsto\, (b,\, +\infty)}} G(x,y)$$

where $\beta \in \{b, c\}$

Let's start the derivation from

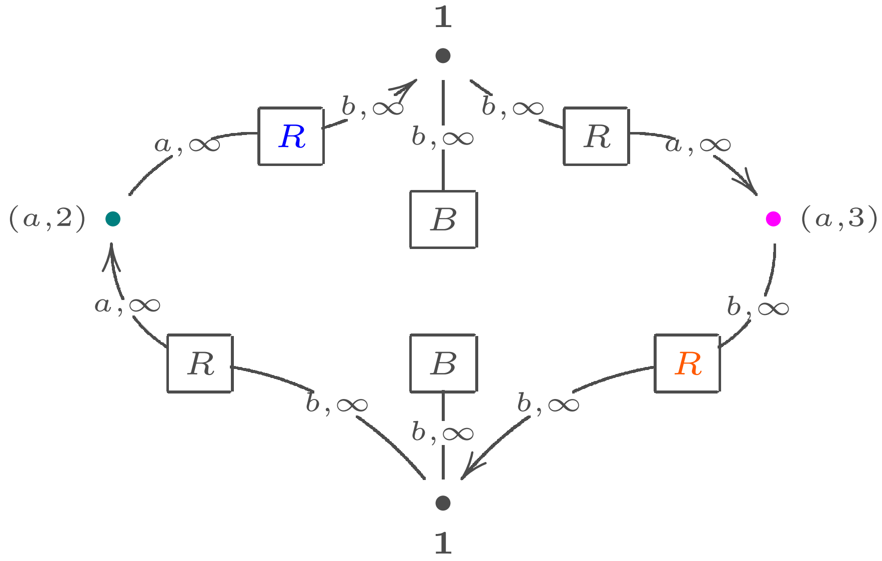


$$\textbf{CreateBrother}(u = 5, n = 2) \times \textbf{CreateBrother}(u = 4, n = 3)$$

- R chooses the production **Create Brother** with $u = 5$ and $n = 2$
- R chooses $u = 4$ and $n = 3$

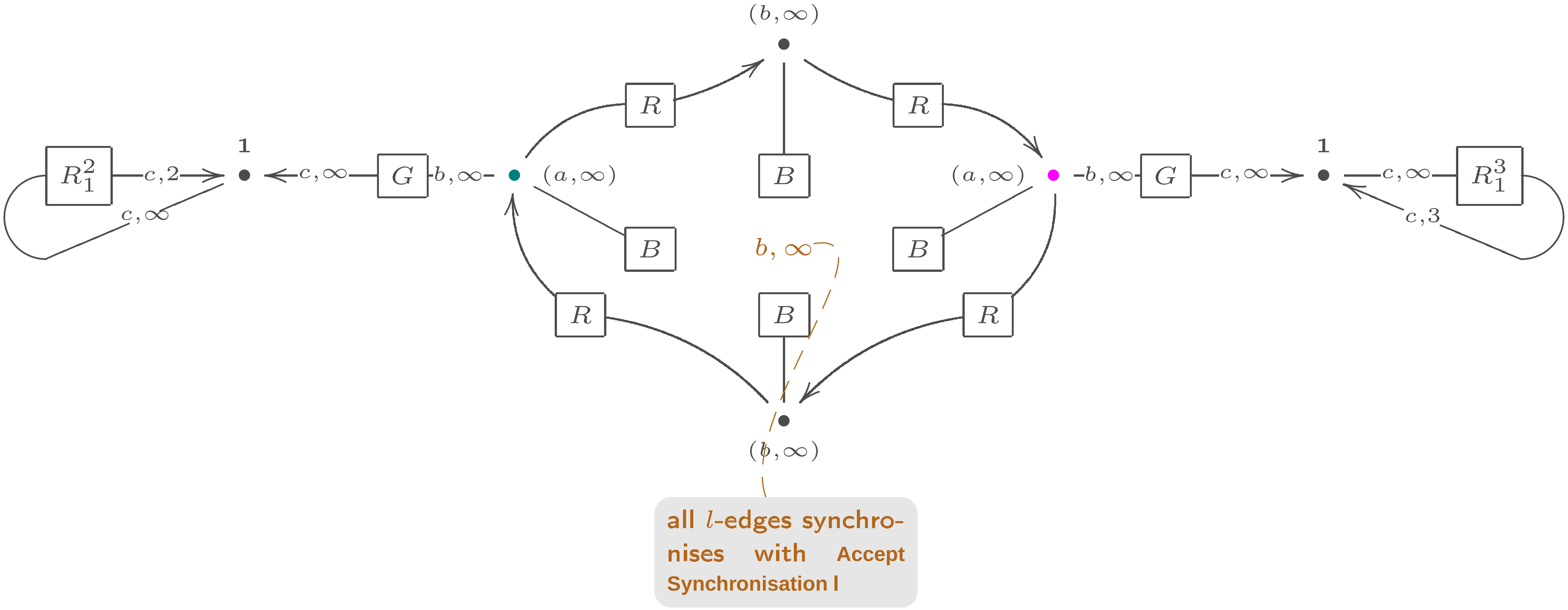
The resulting synchronisation produces the new weights for the nodes as,

$$(a, 2) = (a, 2) \star (a, +\infty) = (a \star_H a, \min(2, +\infty))$$
$$(a, 3) = (a, 3) \star (a, +\infty) = (a \star_H a, \min(3, +\infty)).$$



$\mathbf{CreateGate}(r = 1, u = 2) \times \mathbf{CreateGate}(r = 1, u = 3) \times \mathbf{Accept}^*$

Only R and R can create brothers or gates and they use the remaining resources to create gates to two 2-rings ($r = 1$); the other edges apply the **Accept** productions.



Note that • and • are now blocked.

C-semiring logic for
SHReQ

ΣΠ

A spatio-temporal logic for SHReQ has been defined in [HLT05]
$V_{\mathcal{N}}$ is the set of are node variables, $V_{\mathcal{N}} \cap \mathcal{N} = \emptyset$ and V_R is the set of recursion variables

$$
\begin{array}{rcl}
\phi & ::= & nil \mid \phi|\phi \mid \phi\|\phi \\
 & \mid & \Gamma(\xi) \mid \mathfrak{L}(\tilde{\xi}) \\
 & \mid & f(\phi, \ldots, \phi) \\
 & \mid & \sum_u \phi \mid \prod_u \phi \\
 & \mid & [\Sigma]\,\phi \mid [\Pi]\,\phi \\
 & \mid & u = v \\
 & \mid & \mathfrak{r}(\tilde{\xi}) \mid (\mu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi} \mid (\nu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi}
\end{array}
$$

- $\mathfrak{L} \subseteq \mathcal{L}$ denotes a finite set of labels
- ξ ranges over $\mathcal{N} \cup V_{\mathcal{N}}$
- f is a symbol for c-semiring function
- $u, v \in V_{\mathcal{N}}$
- $\mathfrak{r} \in V_R$

A spatio-temporal logic for SHReQ has been defined in [HLT05]
$V_{\mathcal{N}}$ is the set of are node variables, $V_{\mathcal{N}} \cap \mathcal{N} = \emptyset$ and V_R is the set of recursion variables

$$
\begin{array}{rcl}
\phi & ::= & nil \mid \phi|\phi \mid \phi\|\phi \\
& \mid & \Gamma(\xi) \mid \mathfrak{L}(\tilde{\xi}) \\
& \mid & f(\phi, \ldots, \phi) \\
& \mid & \sum_u \phi \mid \prod_u \phi \\
& \mid & [\Sigma]\,\phi \mid [\Pi]\,\phi \\
& \mid & u = v \\
& \mid & \mathfrak{r}(\tilde{\xi}) \mid (\mu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi} \mid (\nu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi}
\end{array}
$$

$\mathcal{F}$ is the set of c-semiring function $S^i \to S$, $i \geq 0$. Set $\mathcal{F}$ obviously contains c-semiring addition and multiplication as binary functions, and values as zero-adic functions

A spatio-temporal logic for SHReQ has been defined in [HLT05]
$V_{\mathcal{N}}$ is the set of are node variables, $V_{\mathcal{N}} \cap \mathcal{N} = \emptyset$ and V_R is the set of recursion variables

$$\begin{array}{rcl} \phi & ::= & nil \mid \phi|\phi \mid \phi\|\phi \\ & \mid & \Gamma(\xi) \mid \mathfrak{L}(\tilde{\xi}) \\ & \mid & f(\phi, \ldots, \phi) \\ & \mid & \sum_u \phi \mid \prod_u \phi \\ & \mid & [\Sigma]\,\phi \mid [\Pi]\,\phi \\ & \mid & u = v \\ & \mid & \mathfrak{r}(\tilde{\xi}) \mid (\mu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi} \mid (\nu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi} \end{array}$$

Spatial and temporal modalities (dual modalities are necessary for the lack of negation in c-semirings). Quantification is modelled on top of c-semiring summation and moltiplication.

A spatio-temporal logic for SHReQ has been defined in [HLT05]

$V_{\mathcal{N}}$ is the set of are node variables, $V_{\mathcal{N}} \cap \mathcal{N} = \emptyset$ and V_R is the set of recursion variables

$$
\begin{array}{rcl}
\phi & ::= & nil \mid \phi|\phi \mid \phi\|\phi \\
& \mid & \Gamma(\xi) \mid \mathfrak{L}(\tilde{\xi}) \\
& \mid & f(\phi, \ldots, \phi) \\
& \mid & \sum_u \phi \mid \prod_u \phi \\
& \mid & [\Sigma]\,\phi \mid [\Pi]\,\phi \\
& \mid & u = v \\
& \mid & \mathfrak{r}(\tilde{\xi}) \mid (\mu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi} \mid (\nu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi}
\end{array}
$$

Notice that

- C-semiring values $\mathbf{0}$ and $\mathbf{1}$ model falsity and truth
- Absence of negation [LM05]
- Basic operations are represented by functions f

- Formulae are interpreted as maps $\mathcal{G} \to S$, where $\mathcal{G}$ is the set of all weighted graphs
- Let $(\mathcal{QP}, \Gamma \vdash G)$ be a SHReQ rewriting system and $\langle S, +, \star, \mathbf{0}, \mathbf{1} \rangle$ a c-semiring
- $\kappa \in \{\Sigma, \Pi\}$, $\sigma : V_{\mathcal{N}} \to \mathcal{N}$ and $\rho : V_R \to \mathcal{G} \to S$

- Formulae are interpreted as maps $\mathcal{G} \to S$, where $\mathcal{G}$ is the set of all weighted graphs
- Let $(\mathcal{QP}, \Gamma \vdash G)$ be a SHReQ rewriting system and $\langle S, +, \star, \mathbf{0}, \mathbf{1} \rangle$ a c-semiring
- $\kappa \in \{\Sigma, \Pi\}$, $\sigma : V_{\mathcal{N}} \to \mathcal{N}$ and $\rho : V_R \to \mathcal{G} \to S$

$$
\begin{array}{rcl}
[\![\ nil\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & G \equiv nil \\
[\![\ \mathfrak{L}(\tilde{\xi})\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \sum_{L \in \mathfrak{L}} \{G \equiv L(\tilde{\xi}\sigma)\} \\
[\![\ \xi = \xi'\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \xi\sigma = \xi'\sigma \\
[\![\ \Gamma(\xi)\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & (\xi\sigma \in \mathrm{dom}(\Gamma)) \star \Gamma(\xi\sigma) \\
[\![\ f(\phi_1, \ldots, \phi_n)\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & f([\![\ \phi_1\ ]\!]_{\sigma;\rho}(\Gamma \vdash G), \ldots, [\![\ \phi_n\ ]\!]_{\sigma;\rho}(\Gamma \vdash G)) \\
[\![\ \phi_1 | \phi_2\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \sum_{(G_1, G_2) \in \Theta(G)} \{[\![\ \phi_1\ ]\!]_{\sigma;\rho}(\Gamma \vdash G_1) \star [\![\ \phi_2\ ]\!]_{\sigma;\rho}(\Gamma \vdash G_2)\} \\
[\![\ \phi_1 \| \phi_2\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \prod_{(G_1, G_2) \in \Theta(G)} \{[\![\ \phi_1\ ]\!]_{\sigma;\rho}(\Gamma \vdash G_1) + [\![\ \phi_2\ ]\!]_{\sigma;\rho}(\Gamma \vdash G_2)\} \\
[\![\ \kappa_u\ \phi\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \kappa_{x \in \mathrm{dom}(\Gamma)} [\![\ \phi\ ]\!]_{\sigma[^x/_u];\rho}(\Gamma \vdash G) \\
[\![\ [\kappa]\,\phi\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \kappa_{\Gamma \vdash G \xrightarrow{\Lambda} \Gamma' \vdash G'} [\![\ \phi\ ]\!](\Gamma' \vdash G') \\
[\![\ \mathfrak{r}(\tilde{\xi})\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \mathfrak{r}\rho(\tilde{\xi}\sigma) \\
[\![\ (\mu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi}\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & lfp(\lambda\mathfrak{r}'.\lambda\tilde{v}.[\![\ \phi\ ]\!]_{\sigma[^{\tilde{v}}/_{\tilde{u}}],\rho[\mathfrak{r} \mapsto \mathfrak{r}']})(\tilde{\xi}\sigma)(\Gamma \vdash G) \\
[\![\ (\nu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi}\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & gfp(\lambda\mathfrak{r}'.\lambda\tilde{v}.[\![\ \phi\ ]\!]_{\sigma[^{\tilde{v}}/_{\tilde{u}}],\rho[\mathfrak{r} \mapsto \mathfrak{r}']})(\tilde{\xi}\sigma)(\Gamma \vdash G)
\end{array}
$$

- Formulae are interpreted as maps $\mathcal{G} \to S$, where $\mathcal{G}$ is the set of all weighted graphs
- Let $(\mathcal{QP}, \Gamma \vdash G)$ be a SHReQ rewriting system and $\langle S, +, \star, \mathbf{0}, \mathbf{1} \rangle$ a c-semiring
- $\kappa \in \{\Sigma, \Pi\}$, $\sigma : V_{\mathcal{N}} \to \mathcal{N}$ and $\rho : V_R \to \mathcal{G} \to S$

$$
\begin{array}{rcl}
[\![\, nil \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & G \equiv nil \\
[\![\, \mathfrak{L}(\tilde{\xi}) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \sum_{L \in \mathfrak{L}} \{G \equiv L(\tilde{\xi}\sigma)\} \\
[\![\, \xi = \xi' \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \xi\sigma = \xi'\sigma \\
[\![\, \Gamma(\xi) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & (\xi\sigma \in \mathrm{dom}(\Gamma)) \star \Gamma(\xi\sigma) \\
[\![\, f(\phi_1, \ldots, \phi_n) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & f([\![\, \phi_1 \,]\!]_{\sigma;\rho}(\Gamma \vdash G), \ldots, [\![\, \phi_n \,]\!]_{\sigma;\rho}(\Gamma \vdash G)) \\
[\![\, \phi_1 | \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \sum_{(G_1,G_2) \in \Theta(G)} \{[\![\, \phi_1 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_1) \star [\![\, \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_2)\} \\
[\![\, \phi_1 \| \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \prod_{(G_1,G_2) \in \Theta(G)} \{[\![\, \phi_1 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_1) + [\![\, \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_2)\} \\
[\![\, \kappa_u \; \phi \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \kappa_{x \in \mathrm{dom}(\Gamma)} [\![\, \phi \,]\!]_{\sigma[^x/_u];\rho}(\Gamma \vdash G) \\
[\![\, [\kappa] \, \phi \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \kappa_{\Gamma \vdash G \xrightarrow{\Lambda} \Gamma' \vdash G'} [\![\, \phi \,]\!](\Gamma' \vdash G') \\
[\![\, \mathfrak{r}(\tilde{\xi}) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \mathfrak{r}\rho(\tilde{\xi}\sigma) \\
[\![\, (\mu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi} \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \mathit{lfp}(\lambda\mathfrak{r}'.\lambda\tilde{v}.[\![\, \phi \,]\!]_{\sigma[^{\tilde{v}}/_{\tilde{u}}],\rho[\mathfrak{r} \mapsto \mathfrak{r}']})(\tilde{\xi}\sigma)(\Gamma \vdash G) \\
[\![\, (\nu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi} \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \mathit{gfp}(\lambda\mathfrak{r}'.\lambda\tilde{v}.[\![\, \phi \,]\!]_{\sigma[^{\tilde{v}}/_{\tilde{u}}],\rho[\mathfrak{r} \mapsto \mathfrak{r}']})(\tilde{\xi}\sigma)(\Gamma \vdash G)
\end{array}
$$

- Formulae are interpreted as maps $\mathcal{G} \to S$, where $\mathcal{G}$ is the set of all weighted graphs
- Let $(\mathcal{QP}, \Gamma \vdash G)$ be a SHReQ rewriting system and $\langle S, +, \star, \mathbf{0}, \mathbf{1} \rangle$ a c-semiring
- $\kappa \in \{\Sigma, \Pi\}$, $\sigma : V_{\mathcal{N}} \to \mathcal{N}$ and $\rho : V_R \to \mathcal{G} \to S$

$$
\begin{array}{rcl}
[\![\, nil \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & G \equiv nil \\
[\![\, \mathfrak{L}(\tilde{\xi}) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \sum_{L \in \mathfrak{L}} \{G \equiv L(\tilde{\xi}\sigma)\} \\
[\![\, \xi = \xi' \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \xi\sigma = \xi'\sigma \\
[\![\, \Gamma(\xi) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & (\xi\sigma \in \mathrm{dom}(\Gamma)) \star \Gamma(\xi\sigma) \\
[\![\, f(\phi_1, \ldots, \phi_n) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & f([\![\, \phi_1 \,]\!]_{\sigma;\rho}(\Gamma \vdash G), \ldots, [\![\, \phi_n \,]\!]_{\sigma;\rho}(\Gamma \vdash G)) \\
[\![\, \phi_1 | \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \sum_{(G_1, G_2) \in \Theta(G)} \{[\![\, \phi_1 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_1) \star [\![\, \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_2)\} \\
[\![\, \phi_1 \| \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \prod_{(G_1, G_2) \in \Theta(G)} \{[\![\, \phi_1 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_1) + [\![\, \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_2)\} \\
[\![\, \kappa_u \; \phi \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \kappa_{x \in \mathrm{dom}(\Gamma)} [\![\, \phi \,]\!]_{\sigma[^x/_u];\rho}(\Gamma \vdash G) \\
[\![\, [\kappa] \, \phi \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \kappa_{\Gamma \vdash G \xrightarrow{\Lambda} \Gamma' \vdash G'} [\![\, \phi \,]\!](\Gamma' \vdash G') \\
[\![\, \mathfrak{r}(\tilde{\xi}) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \mathfrak{r}\rho(\tilde{\xi}\sigma) \\
[\![\, (\mu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi} \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & lfp(\lambda\mathfrak{r}'.\lambda\tilde{v}.[\![\, \phi \,]\!]_{\sigma[^{\tilde{v}}/_{\tilde{u}}],\rho[\mathfrak{r} \mapsto \mathfrak{r}']})(\tilde{\xi}\sigma)(\Gamma \vdash G) \\
[\![\, (\nu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi} \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & gfp(\lambda\mathfrak{r}'.\lambda\tilde{v}.[\![\, \phi \,]\!]_{\sigma[^{\tilde{v}}/_{\tilde{u}}],\rho[\mathfrak{r} \mapsto \mathfrak{r}']})(\tilde{\xi}\sigma)(\Gamma \vdash G)
\end{array}
$$

- Formulae are interpreted as maps $\mathcal{G} \to S$, where $\mathcal{G}$ is the set of all weighted graphs
- Let $(\mathcal{QP}, \Gamma \vdash G)$ be a SHReQ rewriting system and $\langle S, +, \star, \mathbf{0}, \mathbf{1} \rangle$ a c-semiring
- $\kappa \in \{\Sigma, \Pi\}$, $\sigma : V_{\mathcal{N}} \to \mathcal{N}$ and $\rho : V_R \to \mathcal{G} \to S$

$$
\begin{array}{rcl}
[\![\ nil\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & G \equiv nil \\
[\![\ \mathfrak{L}(\tilde{\xi})\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \sum_{L \in \mathfrak{L}} \{G \equiv L(\tilde{\xi}\sigma)\} \\
[\![\ \xi = \xi'\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \xi\sigma = \xi'\sigma \\
[\![\ \Gamma(\xi)\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & (\xi\sigma \in \mathrm{dom}(\Gamma)) \star \Gamma(\xi\sigma) \\
[\![\ f(\phi_1, \ldots, \phi_n)\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & f([\![\ \phi_1\ ]\!]_{\sigma;\rho}(\Gamma \vdash G), \ldots, [\![\ \phi_n\ ]\!]_{\sigma;\rho}(\Gamma \vdash G)) \\
[\![\ \phi_1 | \phi_2\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \sum_{(G_1,G_2) \in \Theta(G)} \{[\![\ \phi_1\ ]\!]_{\sigma;\rho}(\Gamma \vdash G_1) \star [\![\ \phi_2\ ]\!]_{\sigma;\rho}(\Gamma \vdash G_2)\} \\
[\![\ \phi_1 \| \phi_2\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \prod_{(G_1,G_2) \in \Theta(G)} \{[\![\ \phi_1\ ]\!]_{\sigma;\rho}(\Gamma \vdash G_1) + [\![\ \phi_2\ ]\!]_{\sigma;\rho}(\Gamma \vdash G_2)\} \\
[\![\ \kappa_u\ \phi\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \kappa_{x \in \mathrm{dom}(\Gamma)} [\![\ \phi\ ]\!]_{\sigma[^x/_u];\rho}(\Gamma \vdash G) \\
[\![\ [\kappa]\, \phi\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \kappa_{\Gamma \vdash G \xrightarrow{\Lambda} \Gamma' \vdash G'} [\![\ \phi\ ]\!](\Gamma' \vdash G') \\
[\![\ \mathfrak{r}(\tilde{\xi})\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \mathfrak{r}\rho(\tilde{\xi}\sigma) \\
[\![\ (\mu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi}\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \mathit{lfp}(\lambda\mathfrak{r}'.\lambda\tilde{v}.[\![\ \phi\ ]\!]_{\sigma[^{\tilde{v}}/_{\tilde{u}}],\rho[\mathfrak{r} \mapsto \mathfrak{r}']})(\tilde{\xi}\sigma)(\Gamma \vdash G) \\
[\![\ (\nu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi}\ ]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \mathit{gfp}(\lambda\mathfrak{r}'.\lambda\tilde{v}.[\![\ \phi\ ]\!]_{\sigma[^{\tilde{v}}/_{\tilde{u}}],\rho[\mathfrak{r} \mapsto \mathfrak{r}']})(\tilde{\xi}\sigma)(\Gamma \vdash G)
\end{array}
$$

- Formulae are interpreted as maps $\mathcal{G} \to S$, where $\mathcal{G}$ is the set of all weighted graphs
- Let $(\mathcal{QP}, \Gamma \vdash G)$ be a SHReQ rewriting system and $\langle S, +, \star, \mathbf{0}, \mathbf{1} \rangle$ a c-semiring
- $\kappa \in \{\Sigma, \Pi\}$, $\sigma : V_{\mathcal{N}} \to \mathcal{N}$ and $\rho : V_R \to \mathcal{G} \to S$

$$
\begin{array}{rcl}
[\![\, nil \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & G \equiv nil \\
[\![\, \mathfrak{L}(\tilde{\xi}) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \sum_{L \in \mathfrak{L}} \{G \equiv L(\tilde{\xi}\sigma)\} \\
[\![\, \xi = \xi' \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \xi\sigma = \xi'\sigma \\
[\![\, \Gamma(\xi) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & (\xi\sigma \in \mathrm{dom}(\Gamma)) \star \Gamma(\xi\sigma) \\
[\![\, f(\phi_1, \ldots, \phi_n) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & f([\![\, \phi_1 \,]\!]_{\sigma;\rho}(\Gamma \vdash G), \ldots, [\![\, \phi_n \,]\!]_{\sigma;\rho}(\Gamma \vdash G)) \\
[\![\, \phi_1 | \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \sum_{(G_1, G_2) \in \Theta(G)} \{[\![\, \phi_1 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_1) \star [\![\, \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_2)\} \\
[\![\, \phi_1 \| \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \prod_{(G_1, G_2) \in \Theta(G)} \{[\![\, \phi_1 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_1) + [\![\, \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_2)\} \\
[\![\, \kappa_u \; \phi \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \kappa_{x \in \mathrm{dom}(\Gamma)} [\![\, \phi \,]\!]_{\sigma[{}^x/_u];\rho}(\Gamma \vdash G) \\
[\![\, [\kappa] \, \phi \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \kappa_{\Gamma \vdash G \xrightarrow{\Lambda} \Gamma' \vdash G'} [\![\, \phi \,]\!](\Gamma' \vdash G') \\
[\![\, \mathfrak{r}(\tilde{\xi}) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \mathfrak{r}\rho(\tilde{\xi}\sigma) \\
[\![\, (\mu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi} \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \mathit{lfp}(\lambda\mathfrak{r}'.\lambda\tilde{v}.[\![\, \phi \,]\!]_{\sigma[{}^{\tilde{v}}/_{\tilde{u}}],\rho[\mathfrak{r} \mapsto \mathfrak{r}']})(\tilde{\xi}\sigma)(\Gamma \vdash G) \\
[\![\, (\nu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi} \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \mathit{gfp}(\lambda\mathfrak{r}'.\lambda\tilde{v}.[\![\, \phi \,]\!]_{\sigma[{}^{\tilde{v}}/_{\tilde{u}}],\rho[\mathfrak{r} \mapsto \mathfrak{r}']})(\tilde{\xi}\sigma)(\Gamma \vdash G)
\end{array}
$$

- Formulae are interpreted as maps $\mathcal{G} \to S$, where $\mathcal{G}$ is the set of all weighted graphs
- Let $(\mathcal{QP}, \Gamma \vdash G)$ be a SHReQ rewriting system and $\langle S, +, \star, \mathbf{0}, \mathbf{1} \rangle$ a c-semiring
- $\kappa \in \{\Sigma, \Pi\}$, $\sigma : V_{\mathcal{N}} \to \mathcal{N}$ and $\rho : V_R \to \mathcal{G} \to S$

$$
\begin{array}{rcl}
[\![\, nil \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & G \equiv nil \\
[\![\, \mathfrak{L}(\tilde{\xi}) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \sum_{L \in \mathfrak{L}} \{G \equiv L(\tilde{\xi}\sigma)\} \\
[\![\, \xi = \xi' \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \xi\sigma = \xi'\sigma \\
[\![\, \Gamma(\xi) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & (\xi\sigma \in \mathrm{dom}(\Gamma)) \star \Gamma(\xi\sigma) \\
[\![\, f(\phi_1, \ldots, \phi_n) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & f([\![\, \phi_1 \,]\!]_{\sigma;\rho}(\Gamma \vdash G), \ldots, [\![\, \phi_n \,]\!]_{\sigma;\rho}(\Gamma \vdash G)) \\
[\![\, \phi_1 | \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \sum_{(G_1, G_2) \in \Theta(G)} \{[\![\, \phi_1 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_1) \star [\![\, \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_2)\} \\
[\![\, \phi_1 \| \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \prod_{(G_1, G_2) \in \Theta(G)} \{[\![\, \phi_1 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_1) + [\![\, \phi_2 \,]\!]_{\sigma;\rho}(\Gamma \vdash G_2)\} \\
[\![\, \kappa_u \; \phi \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \kappa_{x \in \mathrm{dom}(\Gamma)} [\![\, \phi \,]\!]_{\sigma[^x/_u];\rho}(\Gamma \vdash G) \\
[\![\, [\kappa] \, \phi \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \kappa_{\Gamma \vdash G \xrightarrow{\Lambda} \Gamma' \vdash G'} [\![\, \phi \,]\!](\Gamma' \vdash G') \\
[\![\, \mathfrak{r}(\tilde{\xi}) \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \mathfrak{r}\rho(\tilde{\xi}\sigma) \\
[\![\, (\mu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi} \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \mathit{lfp}(\lambda\mathfrak{r}'.\lambda\tilde{v}.[\![\, \phi \,]\!]_{\sigma[^{\tilde{v}}/_{\tilde{u}}],\rho[\mathfrak{r} \mapsto \mathfrak{r}']})(\tilde{\xi}\sigma)(\Gamma \vdash G) \\
[\![\, (\nu\mathfrak{r}(\tilde{u}).\phi)\tilde{\xi} \,]\!]_{\sigma;\rho}(\Gamma \vdash G) & = & \mathit{gfp}(\lambda\mathfrak{r}'.\lambda\tilde{v}.[\![\, \phi \,]\!]_{\sigma[^{\tilde{v}}/_{\tilde{u}}],\rho[\mathfrak{r} \mapsto \mathfrak{r}']})(\tilde{\xi}\sigma)(\Gamma \vdash G)
\end{array}
$$

A formula expressing that there is a path between nodes u, v made of edges labelled by elements in $\mathfrak{L}$:

$$\textbf{path}(u, v, \mathfrak{L}) \stackrel{\text{def}}{=} \mu\mathfrak{r}(u, v).(u = v) \; + \; \sum_{w} \mathfrak{L}(u, w)|\mathfrak{r}(w, v).$$

A pair of nodes u, v belong to a ring whenever there are two disjoint paths from u to v made of R-edges

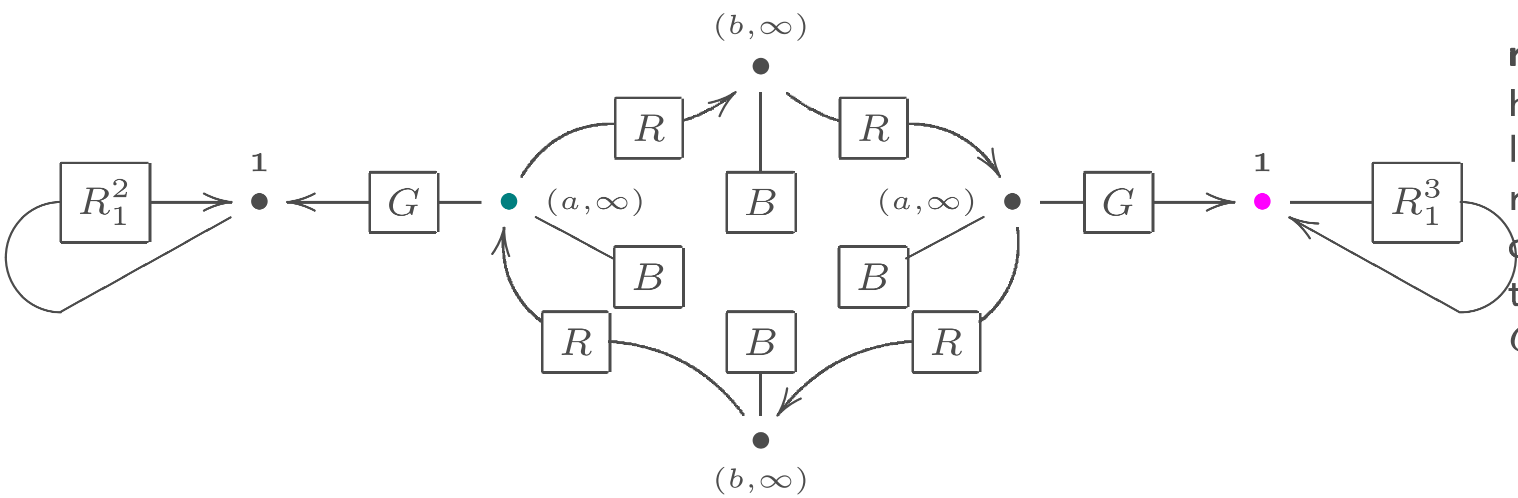
$$\textbf{ring}(u, v) \stackrel{\text{def}}{=} \textbf{path}(u, v, \{R\}) \mid \textbf{path}(v, u, \{R\})$$

A formula expressing that there is a path between nodes u, v made of edges labelled by elements in $\mathfrak{L}$:

$$\textbf{path}(u, v, \mathfrak{L}) \stackrel{\text{def}}{=} \mu\mathfrak{r}(u, v).(u = v) \; + \; \sum_{w} \mathfrak{L}(u, w) | \mathfrak{r}(w, v).$$

A pair of nodes u, v belong to a ring whenever there are two disjoint paths from u to v made of R-edges

$$\textbf{ring}(u, v) \stackrel{\text{def}}{=} \textbf{path}(u, v, \{R\}) \mid \textbf{path}(v, u, \{R\})$$



$\textbf{ring}(u, v)$ does not hold if u and v belongs to different rings because they are connected with a path that must contain a G-edge.

A formula expressing that there is a path between nodes u, v made of edges labelled by elements in $\mathfrak{L}$:

$$\textbf{path}(u, v, \mathfrak{L}) \stackrel{\text{def}}{=} \mu\mathfrak{r}(u, v).(u = v) \; + \; \sum_{w} \mathfrak{L}(u, w) | \mathfrak{r}(w, v).$$

A pair of nodes u, v belong to a ring whenever there are two disjoint paths from u to v made of R-edges

$$\textbf{ring}(u, v) \stackrel{\text{def}}{=} \textbf{path}(u, v, \{R\}) \mid \textbf{path}(v, u, \{R\})$$

$$\textbf{summation}(\phi) \stackrel{\text{def}}{=} \mu\mathfrak{r}.\phi + [\Sigma]\, \mathfrak{r}$$

The boolean interpretation of **summation**(ϕ) is **eventually**(ϕ)

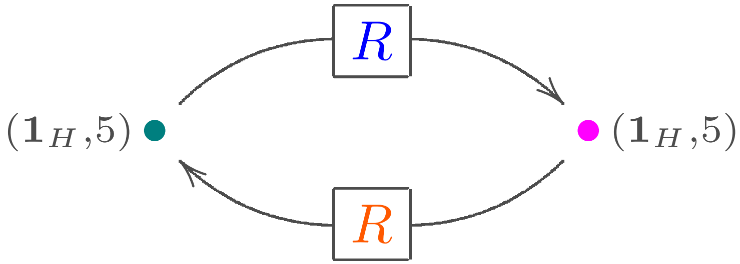
$$\prod_u \neg(\{B\}(u) \mid \mathbf{1}) \rightarrow \textsf{\textbf{summation}}\left(\sum_v (\{G\}(v,u) \mid \mathbf{1})\right)$$

where $\neg \in \mathcal{F}$ maps $\mathbf{0}$ into $\mathbf{1}$ and every value distinct from $\mathbf{0}$ to $\mathbf{0}$ and $\rightarrow$ is $\leq_S$.

$$\prod_u \neg(\{B\}(u) \mid \mathbf{1}) \rightarrow \textbf{summation}\left(\sum_v (\{G\}(v,u) \mid \mathbf{1})\right)$$

$$\prod_u \neg(\{B\}(u) \mid \mathbf{1}) \rightarrow \textbf{summation}\left(\sum_v (\{G\}(v,u) \mid \mathbf{1})\right)$$

- its interpretation on graph



will range over its two nodes • and •

- The antecedent $\mathbf{1}$ since no B-edge is present; the whole formula holds only if the consequent is $\mathbf{1}$ as well.
- The initial graph eventually evolves to a graph containing gates connected to that nodes, namely the consequent is evaluated to $\mathbf{1}$, as required.

A non-trivially valued formula

We now introduce a formula concerning QoS aspects and whose interpretation might return values different from $\mathbf{0}$ or $\mathbf{1}$.

We now introduce a formula concerning QoS aspects and whose interpretation might return values different from $\mathbf{0}$ or $\mathbf{1}$.

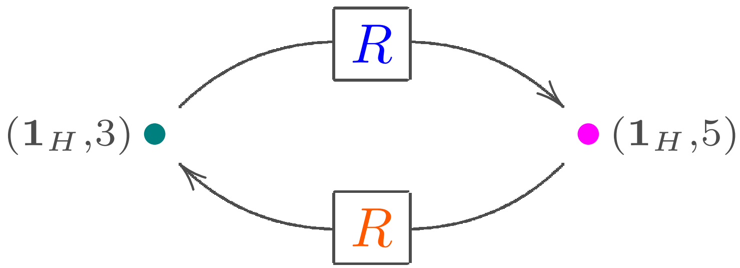
$$\sum_u \Gamma(u) \star \neg(\{B\}(u) \mid \mathbf{1})$$

A non-trivially valued formula

We now introduce a formula concerning QoS aspects and whose interpretation might return values different from $\mathbf{0}$ or $\mathbf{1}$.

$$\sum_u \Gamma(u) \star \neg(\{B\}(u) \mid \mathbf{1})$$

- its interpretation on graph

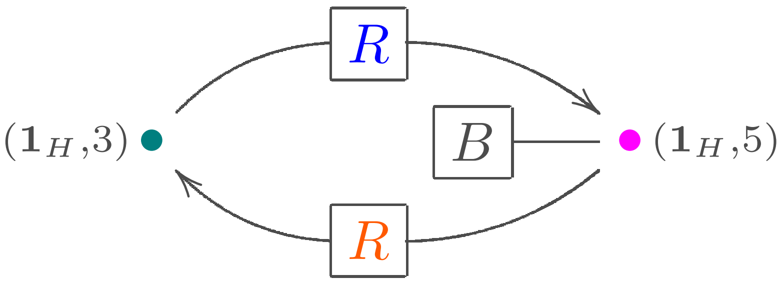


will range over its two nodes • and • and is evaluated to $(\mathbf{1}_H, 5)$

We now introduce a formula concerning QoS aspects and whose interpretation might return values different from $\mathbf{0}$ or $\mathbf{1}$.

$$\sum_{u} \Gamma(u) \star \neg(\{B\}(u) \mid \mathbf{1})$$

- but, its interpretation on graph



is evaluated to $(\mathbf{1}_H, 3)$

Our last example states a property related to structure, behaviour and QoS.

$$\textbf{resource} \quad \equiv \quad \sum_{w,v} \quad (\{R_0^u\}(w,v) \mid \mathbf{1})\star$$

$$([\Sigma]\,(\{R\}(w,v) \mid \mathbf{1}) \star \Gamma(w))$$

Our last example states a property related to structure, behaviour and

First, the resources of newly generated rings are computed by looking over a graph containing a R_0^u-edge

$$\textbf{resource} \equiv \sum_{w,v} (\{R_0^u\}(w,v) \mid \mathbf{1})\star$$

Then, after a rewriting step, the weights of the first attachment node of the R-edge are summed up.

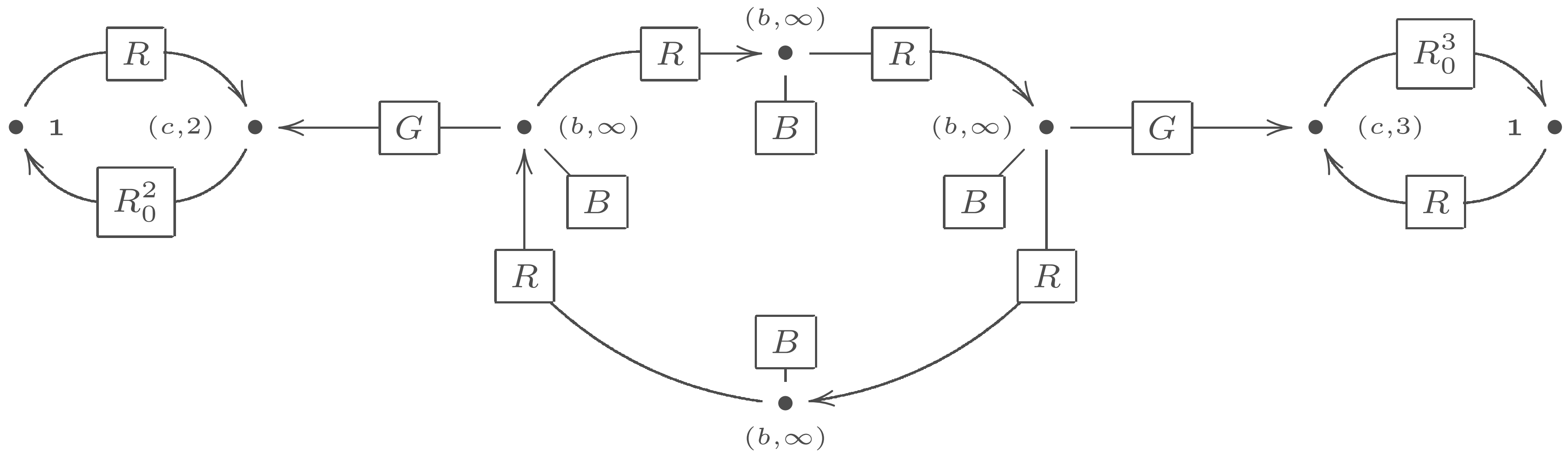
$$([\Sigma]\,(\{R\}(w,v) \mid \mathbf{1}) \star \Gamma(w))$$

Our last example states a property related to structure, behaviour and QoS.

$$\textbf{resource} \equiv \sum_{w,v} (\{R_0^u\}(w,v) \mid \mathbf{1})\star$$

$$([\Sigma]\,(\{R\}(w,v) \mid \mathbf{1}) \star \Gamma(w))$$

- Formula **summation**(**resource**) computes the best value of **resource** in every reachable state, i.e., the best initial resource value of every ring ever created
- interpreting **summation**(**resource**) on



finds the two new rings over which the maximum is chosen resulting in $(\mathbf{0}_H, 3)$.

Conclusions and future directions

We presented c-semiring as

- an abstract model of application level QoS
- a synchronisation mechanism
- an interpretation of a spatio-temporal logic

Future work

- comparisons of c-semiring and other algebraic structures
- decidability of the logic

References

[DFM+03] Rocco De Nicola, Gianluigi Ferrari, Ugo Montanari, Rosario Pugliese, and Emilio Tuosto. A Formal Basis for Reasoning on Programmable QoS. In Nachum Dershowitz, editor, *International Symposium on Verification – Theory and Practice – Honoring Zohar Manna's 64th Birthday*, volume 2772 of *Lecture Notes in Computer Science*, pages 436–479. Springer-Verlag, 2003.

[FMT01] Gianluigi Ferrari, Ugo Montanari, and Emilio Tuosto. A LTS Semantics of Ambients via Graph Synchronization with Mobility. In *Italian Conference on Theoretical Computer Science*, volume 2202 of *Lecture Notes in Computer Science*, Torino (Italy), October 4-6, 2001. Springer-Verlag.

[Hir03] Dan Hirsch. *Graph Transformation Models for Software Architecture Styles*. PhD thesis, Departamento de Computación, Universidad de Buenos Aires, 2003. http://www.di.unipi.it/˜dhirsch.

[HLT05] Dan Hirsch, Alberto Lluch-Lafuente, and Emilio Tuosto. A Logic for Application Level QoS. In *3nd Workshop on Quantitative Aspects of Programming Languages*, April 2005.

[HT05] Dan Hirsch and Emilio Tuosto. SHReQ: A Framework for Coordinating Application Level QoS. In K. Aichernig Bernhard and Beckert Bernhard, editors, *3rd IEEE International Conference on Software Engineering and Formal Methods*, pages 425–434. IEEE Computer Society, 2005.

[LM04] I. Lanese and U. Montanari. Mapping fusion and synchronized hyperedge replacement into logic programming. *Theory and Practice of Logic Programming, Special Issue on Multiparadigm Languages and Constraint Programming*, 2004. Submitted.

[LM05] A. Lluch Lafuente and U. Montanari. Quantitative mu-calculus and ctl defined over constraint semirings. *TCS special issue on quantitative aspects of programming languages*, 2005. To appear.

[LT05] Ivan Lanese and Emilio Tuosto. Synchronized Hyperedge Replacement for Heterogeneous Systems. In Jean-Marie Jacquet and Gian Pietro Picco, editors, *International Conference on Coordination Models and Languages*, volume 3454 of *Lecture Notes in Computer Science*, pages 220 – 235. Springer-Verlag, April 2005.

[Tuo03] Emilio Tuosto. *Non-Functional Aspects of Wide Area Network Programming*. PhD thesis, Dipartimento di Informatica, Università di Pisa, May 2003. TD-8/03.