Site graph rewriting

Russ Harmer

Rewriting

A *site graph* G is defined by (i) two finite sets, $\mathcal{A}$ and $\mathcal{S}$, of *agents* and *sites* respectively; (ii) a function $\sigma : \mathcal{S} \to \mathcal{A}$; (iii) a subset $\mathcal{F} \subseteq \mathcal{S}$ of *stubs*; and (iv) a symmetric binary relation $\mathcal{E}$ on sites.

The relation $\mathcal{E}$ defines a simple, undirected graph structure on $\mathcal{S}$, while the idea of a stub is that of a site capable of supporting a further incident edge; note that both aspects of the graph structure are 'extensional' in the sense that (i) a site either is or is not a stub; and (ii) there is at most one edge between any two sites. It is possible for a site to be in neither $\mathcal{F}$ nor $\|\mathcal{E}\|$ [the support of $\mathcal{E}$]. The function σ induces an 'intensional' equivalence relation on $\mathcal{S}$, *i.e.* the equivalence classes are 'indexed' by the agents.

A *homomorphism* $h : G \to G'$ of site graphs consists of a pair of functions, $h_\mathcal{S} : \mathcal{S} \to \mathcal{S}'$ and $h_\mathcal{A} : \mathcal{A} \to \mathcal{A}'$, such that (i) $h_\mathcal{A} \circ \sigma = \sigma' \circ h_\mathcal{S}$; (ii) if $s \in \mathcal{F}$ then $h_\mathcal{S}(s) \in \mathcal{F}'$; and (iii) if $s_1 \mathrel{\mathcal{E}} s_2$ then $h_\mathcal{S}(s_1) \mathrel{\mathcal{E}'} h_\mathcal{S}(s_2)$. A site in neither $\mathcal{F}$ nor $\mathcal{E}$ can be mapped anywhere—modulo preservation of its corresponding agent—and corresponds to a 'no binding status' wild-card.

The category **SGrph** of site graphs and homomorphisms has all pull-backs and all push-outs. An arrow h of **SGrph** is a mono if, and only if, $h_\mathcal{S}$ and $h_\mathcal{A}$ are both injective. The category **SGrph** additionally has all *pull-back complements over monos*. We can thus perform sesqui-push-out rewriting over *rules* defined as arbitrary spans in **SGrph** using arbitrary monos as *matchings*.

SGrph also has an initial object [the empty graph] and a terminal object [the graph with one agent that has one site that is a stub and has a self-loop]. As such, **SGrph** is finitely complete and co-complete.

Realizable site graphs

A site graph G is *realizable* iff the graph structure of G is 'deterministic' in the following sense: (i) if $s \in \mathcal{F}$ then $s \notin \|\mathcal{E}\|$; (ii) if $s \mathrel{\mathcal{E}} s_1$ and $s \mathrel{\mathcal{E}} s_2$ then $s_1 = s_2$; and (iii) $\mathcal{E}$ is irreflexive. In words, a site that already has an incident edge cannot solicit another [by having a stub]; nor can it have multiple incident edges to begin with—including self-loops.

If $h : G \rightarrowtail G'$ and G' is realizable then so is G: a site with a stub and an incident edge would have to preserve both in G'; a site with multiple incident edges would have to preserve them all and, by injectivity of $h_\mathcal{S}$, maintain them distinct in G'; and any self-loops in G would also have to be preserved in G'.

The class of realizable site graphs is closed under rewriting by the so-called 'linear' rules, *i.e.* spans of monos with realizable co-domains. There is no straightforward characterization of the sub-class of linear rules without side effects; but one sufficient condition is to ask for 'mass preservation'.