Abstract

In this dissertation, we study bigraphical languages—languages based on the theory for bigraphs and bigraphical reactive systems developed by Milner and coworkers.

We begin by examining algebraic theory for binding bigraphs. We give a term language for binding bigraphs and develop a complete axiomatization of structural equivalence. Along the way, we develop a set of normal form theorems, which prove convenient for the analysis of bigraphical structure.

We examine next the central problem of matching, that is, to determine when and where the left-hand side of a bigraphical reaction rule matches a bigraph. We give an inductive characterization in the form of a set of rules for inferring matching for binding bigraphs and show that the characterization is both sound and complete.

We then develop a matching algorithm that works on terms for bigraphs. We start by specializing the characterization above to include application of structural congruence. We isolate a class of *normal* inferences, and prove that normal inferences are sufficient for inferring all matches. The matching algorithm relies on building normal inferences mechanically. An implementation of the algorithm is at the core of the BPL Tool, a prototype tool for experimenting with bigraphical reactive systems.

In a second line of work, we study bigraphical reactive systems as a vehicle for developing a language to model biochemical reactions at the level of cells and proteins. We discuss and isolate $\mathcal{B}^{\Sigma,\mathcal{R}}$-calculi, a family of bigraphical reactive systems that we deem sufficient for the language. We develop a self-contained presentation of the syntax and operational semantics for $\mathcal{B}^{\Sigma,\mathcal{R}}$-calculi that exploits the restrictions we have made on the family. We also treat how one may extend certain bigraphical reaction rules to include negative contextual side-conditions. As an example, we show that the non-deterministic κ-calculus (due to Danos and Laneve) can be modelled.

Finally, we build on our study above and develop a formal language, the $\mathcal{C}$-calculus, for modelling low-level interaction inside and among cells. At the core of the calculus lies a model of formal proteins and membranes. In addition, formal channels between compartments allow us to model an intermediate state in cell fusion or division, regulated by diffusion. A user models in the $\mathcal{C}$-calculus by refining a set of core rules, each of which encapsulates a core biological reaction. We illustrate the calculus with two examples, one that models the firing of a G-protein coupled receptor protein, and another that models the formation of clathrin-coated cytoplasmic vesicles through budding.