論 文 の 内 容 の 要 旨

論文題目 Geometry of Parallelism: A Uniform Analysis of Evaluation Strategies and Effects by Synchronous Interaction Abstract Machine
（並行性の幾何：同期付き相互作用抽象機械による評価戦略および副作用の統一的な解析）

氏　　名　　由水　輝

This thesis introduces the notion of Synchronous Interaction Abstract Machine that is a novel variation of the Geometry of Interaction (GoI) framework. We provide a thorough account on the newly introduced framework and employ it to obtain adequate semantics of a class of programming languages possibly with memories and effects, uniformly treating the call-by-name and the call-by-value strategies.

The Geometry of Interaction (Girard, LC 1988) is originally a scheme to give dynamic semantics to linear logic proofs. Later, it has proved to be a useful framework in computer science in general: when represented as a token machine called the Interaction Abstract Machine (IAM) (Danos & Regnier, Electr. Notes Theor. Comput. Sci. 1996), it can be seen as a compilation scheme of possibly higher-order computation into first-order, low-level computation.

This character of GoI is theoretically interesting and at the same time provides a compiler implementation technique based on semantics (e.g. Mackie, POPL 1995). As such, GoI has resulted in a number of results in computer science such as programming language semantics (Hoshino, FoSSaCS 2011), optimal reduction strategy (Gonthier et al., POPL 1992), and defunctionalization (Schöpp, TLCA 2013).

The main achievement of the thesis is to better understand the capability of GoI (and in general dialogue-based) semantics compared to other kinds of semantics. It is presented via interpretation of a quantum programming language with higher-order functions and recursion by an extension of the GoI token machine semantics, namely utilizing multiple tokens at a time. Establishing such a multi-token machine framework itself is non-trivial and is one of the main contributions of the thesis. Moreover, a seemingly different result, namely distinction of the call-by-name and the call-by-value behaviors turns out to follow from the same notion of multi-token machine. We expect that the concurrent and fine structure of our multi-token machine also provides possibilities of applications to semantic approach to inherent parallelism of functional programs and a unified framework for type systems for concurrent computation. Concretely, the contributions of this thesis are as follows.

In the thesis, we provide a thorough account on a multi-token machine on a much more expressive proof net system (that we call SMEYLL proof nets) that accommodates higher-order functions, recursions, and branchings. For linear calculi such a multi-token machine have been recently proposed (Dal Lago & Faggian, QPL 2011, Yoshimizu et. al., ESOP 2014), but compared to the linear cases it turns out to be much more non-trivial whether such a multi-token system behaves well, e.g. whether the machine can deadlock or not. We examine properties on the proof net system, the multi-token system, and the relation between the two, by using both traditional notions in linear logic and new proof techniques we introduce. As a result, it is shown that those systems behave surprisingly well despite its inherent complexity. The main results include deadlock-freedom of the SIAM, invariance and adequacy of the multi-token semantics with respect to net reduction.

Using those results shown, we interpret a PCF-like calculus by SMEYLL nets and the SIAM (appropriately extended with a notion of integer memory). The SIAM is shown to possess superiority over the standard IAM: our SIAM can distinguish the call-by-name and call-by-value reduction strategies without any special construction other than the two corresponding translation of calculus into proof nets, while the IAM (or GoI model in general) can only exhibit the call-by-name character even if we apply the call-by-value translation. This is made precise as an adequacy result between the calculus and the extended SMEYLL nets, which is in turn adequately interpreted by the extended SIAM. The fact that such a parallel, non-standard token machine can adequately interpret a standard sequential calculus is already of theoretical interest, and moreover the parallelism is the source of enrichment of the distinguishing power on reduction strategies.

Furthermore, we again extend the SMEYLL proof net system and the SIAM system by equipping them with a notion of memory structure and (possibly) probabilistic transitions depending on contents of a memory. Instances of memory structures include deterministic, probabilistic, and quantum memories.

The systems are again shown to satisfy desirable properties in the probabilistic setting; to prove those properties, we also introduce a probabilistic variation of abstract reduction system, of which the definition is of independent interest. Finally we interpret a calculus equipped with a memory structure by SMEYLL proof nets and the SIAM with a memory structure. All the proofs are done in a parametric way with respect to memory structure, thus we obtain adequacy result for all the three instances (and any other instance) at a time.