

Title	分散コミュニケーションのためのPi-calculusの拡張とその抽象機械の提案
Author(s)	福田, 博之
Citation	
Issue Date	2002-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1528
Rights	
Description	Supervisor:大堀 淳, 情報科学研究科, 修士

分散コミュニケーションのための Pi-calculus の拡張と その抽象機械の提案

福田 博之 (010098)

北陸先端科学技術大学院大学 情報科学研究科

2002 年 2 月 15 日

キーワード: Pi-calculus, 操作的意味論, 抽象機械, 分散コミュニケーション.

1 研究の背景及び目的

現在, 並行プログラミング言語の計算モデルとして Pi-calculus が注目されている. Pi-calculus は 1990 年に Milner, Parrow, Walker によって提案された. この計算モデルは, 並行動作するプロセス間でのチャンネルを介したコミュニケーションを形式化する. また, チャンネルによるチャンネルの伝達を許すことで, モバイリティの要素を導入し, 動的にコミュニケーショントポロジーを変化させる並行システムの仕様の記述と正当性の検証を可能にする. さらに, チャンネルによるモバイリティは, 表現力を著しく増大させ, 簡潔な意味論と扱いやすい代数理論を伴って, Pi-calculus を高水準な並行性を幅広く記述することができる計算モデルへと導いた.

一方で, λ -calculus は簡潔さと表現力を伴って, さかんに理論的な研究が行われ, 逐次的な関数型言語に対する基礎理論となった. それならば, Pi-calculus からどのような高級言語が構築されるのか興味がひかれるところである. そのような動機によって, Pi-calculus に基づく言語のあり方を探るべく実験的に設計された言語として Pict がある.

Pict は, Pierce と Turner によって 1992 年から開発された. Pict は, 計算の唯一のメカニズムとしてコミュニケーションを利用し, 純粋に Pi-calculus に基づく言語と言える. これまでの Pi-calculus を用いた言語は, 関数型言語に Pi-calculus ライクなコミュニケーションを組み合わせたものであり, その点で Pict は大きく異なる. Pict は, Turner が提案した抽象機械を採用することで, Pi-calculus での基本的な計算メカニズムであるチャンネルを介したコミュニケーションを効率よく実装している. しかし, Turner の抽象機械はローカルな環境でのコミュニケーションを実現するものであり, 分散環境への実装が今後の研究課題となっている.

そこで本研究では, Pi-calculus の操作的意味論について考察し, 分散環境へ実装する上でどのような問題が起こるのかを明らかにし, その解決方法を探る. そして, Pi-calculus

を拡張した分散環境へ実装可能な計算モデルを提案し, Turner の抽象機械を拡張して, その計算モデルの実装を試みる.

この研究によって期待される効果は, 今日の大規模, 複雑化する傾向にある分散アプリケーションの動作の正当性や安全性の検証が可能になるということである. なぜなら, 計算モデルに基づいて言語を設計することによって, その言語によって書かれたプログラムの意味付けに対して数学的な基礎を与えることができるからである. これによって, プログラムの振る舞いの特性を証明することができるようになり, プログラムの信頼性の向上につながる. そのため, プログラムの基礎理論に対する研究は, 今後ますますその重要性を増す. 特に, 分散・並行プログラミング言語に対する基礎理論はまだ不十分であり, 今後の大きな研究課題である.

2 分散コミュニケーションのための拡張

Pi-calculus を実装する上で重要な事は, この計算モデルでの基本的な計算メカニズムであるコミュニケーションをいかに効率よく実装するかという事である. Pi-calculus では, コミュニケーションは以下の簡約によって定義される.

$$x!a.P \mid x?(y).Q \rightarrow P \mid \{a/y\}Q$$

この簡約では, 同一チャネルに対する入力プロセスと出力プロセスが現れたとき, 値またはチャネルの送受信が行われるということを表している. ここで, この簡約を分散環境で実装する事を考える. この簡約は, 見ての通り並行動作している2つのプロセスの状態に依存している. もし, この2つのプロセスが別々のサイトに存在していた場合, コミュニケーションを行うには, 2つのサイトの状態を把握しておく必要があるということになる. 分散環境では, 不特定多数のサイトが存在し, しかもサイトは動的に起動, 停止されるものであり, 一時的に存在するものも含め, これら全ての状態を把握するのは困難である. つまり, 上の簡約を分散環境で効率よく実装するのは不可能と言える. そこで, 同期通信のメカニズムを導入して, 他のサイトの状態に依存しない4つの簡約(RCOM1, RCOM2, RCOM3, RETURN)によって分散コミュニケーションを実現する. これによって, 実際のチャネルを介したコミュニケーションはあるサイト内に限定され, Pi-calculus の特性をそのままにして, 分散コミュニケーションを実現することができる.

3 結論

本研究によって得られた結論は以下の通りである.

- 同期通信のメカニズムを導入することによって, Pi-calculus の特性をそのまま持つ分散環境へ実装可能な計算モデル(以下, 拡張 Pi-calculus と呼ぶ)を提案することができた.

- 拡張 Pi-calculus によって分散コミュニケーションを明確に形式化することができるようになった .
- Turner の抽象機械を拡張することで , 拡張 Pi-calculus に対する実装を与えることができた .
- 拡張 Pi-calculus に基づくことによって , 分散コミュニケーションを容易に , かつ柔軟に記述することができる言語システムを実装することができた .
- 実際に分散環境へ拡張 Pi-calculus に基づく言語システムを実装したことによって , 今後の研究における実験のベースを与えることができた .