

動的プランニングへの π -計算の適用の研究

3 F - 6

岩田 員典 伊藤 暢浩 杜 小勇 石井 直宏
名古屋工業大学

1 はじめに

エージェントが行動決定を適切におこない目標を達成するためにはプランニングが必要である。しかし、従来の人工知能の分野で研究されてきたプランニングでは動的に変化する環境に対応することが困難であった。そこで環境の変化に応じてプランを生成し直す再プランニングが重要になってきている。この再プランニングに関する課題の一つに「再プランニングに適した表現は何か」というものがある [2, 3]。

本研究では、この課題に対して多引数 π -計算 (Polyadic π -calculus) [1, 4] を用いる。まず、再プランニングに適した言語を多引数 π -計算に基づいて構成する。この言語を PDL (Plan Description Language) と呼ぶ。

次に、PDL を処理するためのインタプリタを構築する。そこで、システムを構築するために多引数 π -計算に基づいた言語 Polyadic PiL (Polyadic Pi-calculus Language) を提案する。Polyadic PiL は数学的な表現をもつ多引数 π -計算を計算機上で使用するのに適した表現に変換したものである。このシステムが Polyadic PiL で記述されたプログラムを正確に実行できることを検証する。

最後に、PDL で記述されたプランがシステムで実行可能なことを確認する。これにより PDL が実際のシステムに使用可能なことを示す。

2 π -計算によるプランの表現

本研究では、プランを表現するために π -計算の特徴である「プロセス間のリンクや構造の動的変化」という性質を利用した。

プランを実行に移すときには π -計算の名前通信の機能を利用してパラメータを渡す。また、プランの各ゴールを π -計算のプロセスとみなし、ゴール間の関係をリンクとする。このとき、構造の動的変化をプランの動的変化に適用できる。このため、プランの変更が容易におこなえる。

以下に PDL の定義の一部を示す。

定義 2.1 プランの構成

1. goal を満たすために直接メソッド (method) を呼び出す場合

$$\text{Plan}(\text{goal}(\vec{x}), \text{method}(\vec{y}))$$

$$\stackrel{\text{def}}{=} (\nu l) \quad \text{Cons}(\text{goal}(\vec{x}), \text{method}(\vec{y}), l(\vec{x})) \mid \text{Nil}(l(\vec{x})) \mid \text{!Declare}(\text{method}(\vec{y}))$$

ただし $|\vec{x}| = |\vec{y}|$ とする。

2. goal を満たすために subgoal へ分解する場合

$$\text{Plan}(\text{goal}(\vec{x}), \text{subgoal}_1(\vec{x}_1), \dots, \text{subgoal}_n(\vec{x}_n))$$

$$\stackrel{\text{def}}{=} (\nu l_1 \dots l_n) \quad \text{Cons}(\text{goal}(\vec{x}), \text{subgoal}_1(\vec{x}_1), l_1(\vec{x})) \mid \text{Cons}(l_1(\vec{x}), \text{subgoal}_2(\vec{x}_2), l_2(\vec{x})) \mid \dots \mid \text{Cons}(l_{n-2}(\vec{x}), \text{subgoal}_{n-1}(\vec{x}_{n-1}), l_{n-1}(\vec{x})) \mid \text{Cons}(l_{n-1}(\vec{x}), \text{subgoal}_n(\vec{x}_n), l_n(\vec{x})) \mid \text{Nil}(l_n(\vec{x}))$$

ただし、 $\vec{x}_i \subseteq \vec{x}$ ($1 \leq i \leq n$) を満たすとする。

この定義を利用することにより以下に示すような特徴が得られる。これらの特徴は π -計算の遷移規則と構造帰納法を用いることにより証明することができる。

1. 再プランニングを容易に行うことができる
2. メソッドを並行に呼び出すことができる

再プランニングのイメージを図 1 に並行実行のイメージを図 2 に示す。

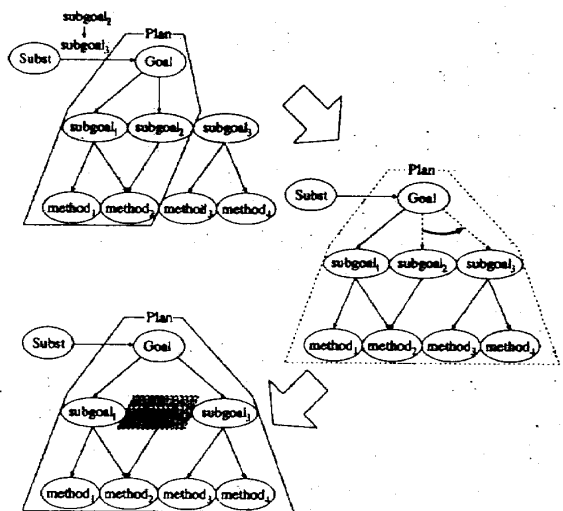


図 1: 再プランニングのイメージ

Applying π -calculus to Dynamic Planning for Agents
Kazunori Iwata, Nobuhiro Ito, Xiaoyong Du and Naohiro Ishii
Nagoya Institute of Technology
Gokiso-cho, Showa-ku, Nagoya, 466-8555, Japan
E-mail: kazunori@egg.ics.nitech.ac.jp
TEL: 052-735-5474
FAX: 052-735-5477

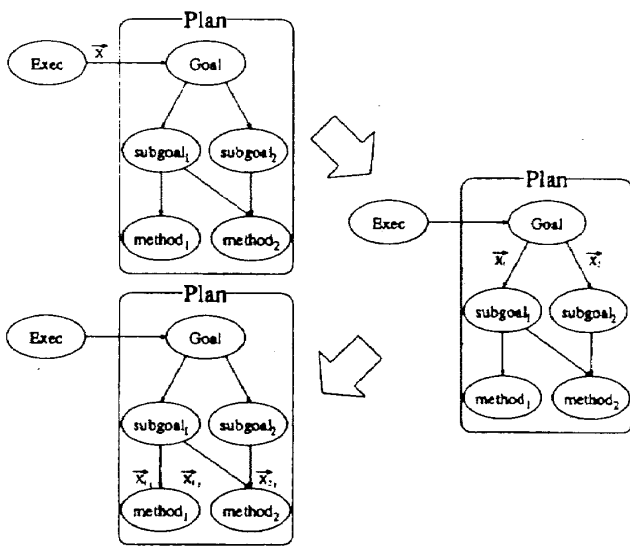


図 2: メソッドの並行実行のイメージ

3 π-計算に基づく処理系の構築

多引数 π-計算の構文にしたがい、かつ計算機上の表現に適した言語 Polyadic PiL を提案する。Polyadic PiL の構文を次のように定義する。

$P ::=$	$(out\ a\ v_1 \dots v_n).P$	出力動作
	$(in\ a\ x_1 \dots x_n).P$	入力動作
	$[a = b].P$	マッチング
	$[a! = b].P$	非マッチング
	$(new\ a_1 \dots a_n).P$	名前制限
	$(out\ a\ v_1 \dots v_n);$	終端出力動作
	$(in\ a\ x_1 \dots x_n);$	終端入力動作
	$\{P_1, P_2, \dots, P_n\};$	並行合成
	$+P_1, P_2, \dots, P_n+;$	非決定和
	$\langle P' \rangle;$	プロセスコール

この構文にしたがい処理をおこなうことにより π-計算の動作をシミュレートするシステムの構築をおこなった。

プロセス P, Q, R をこの構文にしたがって次のように定義する。このプログラムを本システムにより実行した結果を図 3, 図 4, 図 5 に示す。

P	$\stackrel{def}{=} (out\ a\ b);$
Q	$\stackrel{def}{=} (in\ a\ x).(out\ x\ v);$
R	$\stackrel{def}{=} (in\ b\ y);$

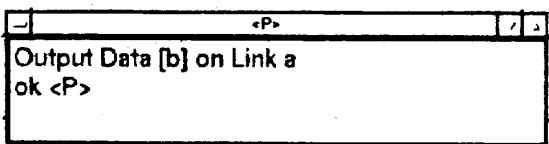


図 3: 実行結果 (プロセス P)

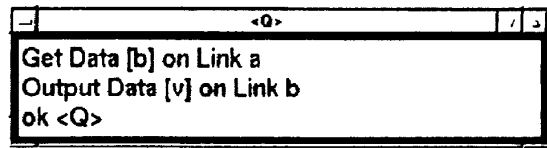


図 4: 実行結果 (プロセス Q)

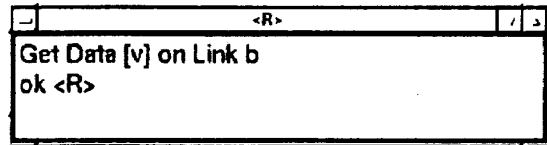


図 5: 実行結果 (プロセス R)

4 まとめと今後の課題

本研究では、プランを表現するための言語 PDL を多引数 π-計算を利用し構成した。そして、PDL で記述したプランが持つ性質を定理として証明した。この性質はプランの動的変更が容易におこなえることを示している。したがって、言語 PDL で記述されたプランは再プランニングに適しているといえる。

PDL を処理するシステムを構築するために言語 Polyadic PiL を提案した。そして、Polyadic PiL を処理するシステムを構築した。本システムは π-計算の動作をシミュレートできることを示した。さらに、PDL で記述したプランが本システムで実行可能なことを確認した。これにより PDL が実装面にも適用できることを示した。

今後は、提案した言語を実際のエージェントシステムに使用して評価することが必要である。また、マルチエージェントシステムに適用することも考えている。そのためには、複数のエージェントのプランを表現する方法について検討する必要がある。

参考文献

- [1] Robin Milner. Polyadic π-calculus: a tutorial. LFCS Report Series ECS-LFCS-91-180, Laboratory for Foundation of Computer Science, 1991.
- [2] S. Au and N. Parameswaran. Plan execution in a dynamic world. In *International Conference on Computing and Information 98*, pp. 331-338, 1998.
- [3] 山田 誠二. エージェントのプランニング. 人工知能学会誌, Vol. 10, No. 5, pp. 677-682, 1995.
- [4] 吉田 展子, 久保 誠, 本田 耕平. π-計算とその周辺. 情報処理, Vol. 37, No. 4, pp. 319-326, 1996.