

6Y-8

Prologへの
並列化述語 dosim の導入

金田悠紀夫 下谷 清 松田秀雄 秋吉克己 前川禎男
神戸大学 工学部

1. はじめに

現在筆者らの研究室では、論理型言語である Prolog に並列計算機能を付加する研究を行っている。密結合型のマルチプロセッサ計算機 (Sequent社の Balance8000) 上に並列化機能を導入した Prolog のインタプリタを実現し、その性能を評価したので報告する。

2. 組込み述語 dosim

本研究では、並列で解を求める (並列にゴールを実行する) 組込み述語 dosim を実現した。この述語の構文、意味を以下に記す。

構文: dosim(引数リスト, 入力変数リスト, ゴール, 出力変数リスト, 解リスト)

例 dosim([a1, a2, a3], X, p(X, Y), Y, [S1, S2, S3])

意味: 入力変数 X に入力リスト [a1, a2, a3] の各要素をひとつずつユニフィケーションし、並列にゴール p(X, Y) を呼び出す。そして得られた解 (出力変数) Y によって解リスト [S1, S2, S3] を作成する。

- ① 入力変数、出力変数には変数を1つ書けるだけでなく、変数リストを書くことができる。
- ② それぞれの変数について、ゴールの解が無いとき (ゴールが失敗したとき) には '\$fail\$' という特殊なアトムが解となる。
- ③ それぞれの引数について、ゴールの解が複数ある時にはまず一組の解を (リストにして) 返す、以下バックトラックが生じてリトライするたびに一組の別解を求め、返す。

④ 全ての引数についてゴールの実行が失敗し (全部の子プロセスが失敗し)、解リストの要素がすべて '\$fail\$' になったとき dosim の実行は失敗する。

⑤ dosim は入れ子構造にすることができる。

下に例を示す通り dosim の中で dosim を呼び出すことができる。

例: ?-dosim([[a, b], [c, d]], X, p(X, Y), Y, [M, N]).
p(X, Y):-dosim(X, A, q(A, B), B, Y).

3. 実行手順

dosim がプログラム中に現われたときの実行手順を図1. に示す。

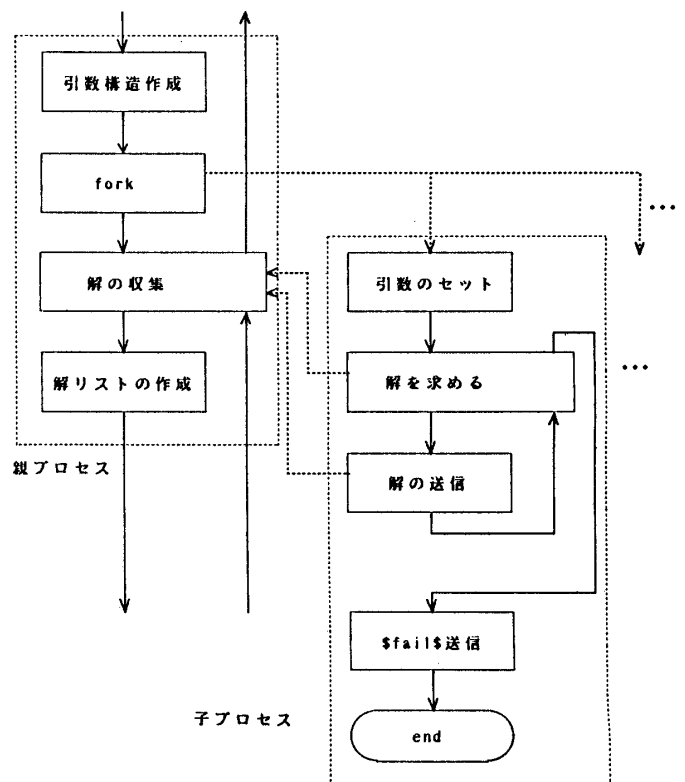


図1. dosim の実行手順

Introduction of The Dosim Predicate to Prolog
 Yukio Kaneda, Kiyoshi Shimotani, Hideo Matsuda,
 Katsuki Akiyoshi, Sadao Maekawa
 Kobe University

図1. において下向きの矢印は `dosim` の実行過程を、上向きの矢印はバックトラックによるリトライを表す。 `dosim` の実行は1つの親プロセスと引数リストの要素数個の子プロセスによって実行される。

I. 親プロセス

(1) 引数構造作成

`dosim` のゴール部の並列実行に先立って引数リストの分解を行う。本処理系では、引数リストの第一要素・第二要素・・・以下第 n 要素までが、それぞれ並列に実行されるゴールの引数（入力変数）となる。

このときプロセスの管理を行う“ID番号”と、何番目の引数に関してゴールを実行したかを管理する“子プロセス番号”をつけておく。この処理によって共有領域に引数構造が構成される。この構造例を図2. に示す。

ID番号 小プロセス番号 引数の内容

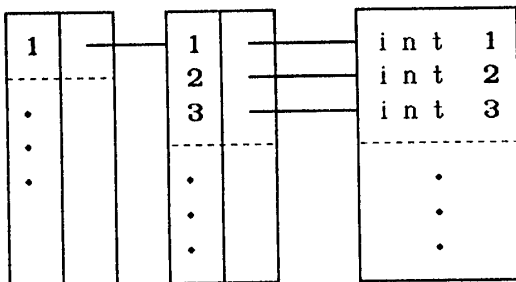


図2. 引数構造例（[1, 2, 3]）

(2) fork

(1)の引数リストの分解によって引数の個数 n が得られている。そこで n 個の子プロセスを、`fork` システムコールによって生成する。これらの子プロセスはBalance8000のOSであるDYNIXによって並列に実行される。

(3) 解の収集

子プロセスによって共通領域に書き込まれた解を集める。

(4) 解リストの作成

(3)によって集められた解を一つのリスト構造にする、このとき与えられた引数の順に解リストを構成する。そして作成されたリストを `dosim` 述語の解リストにユニフィケーションする。

II. 子プロセス

(1) 引数のセット

`fork` によって生成された各子プロセスが自分のプロセス番号と一致する引数を共有領域から得る。

(2) 解の送信

それぞれの子プロセスが求めた解を、親プロセスがアクセスできるように共有領域に書き込む。

4. 実行結果

`nqueen` 問題について、 $n=6, 7, 8$ の場合の全解探索を行い実行時間の計測を行った。この結果を表1に示す。

表1. 実行時間

単位は [秒]

n	逐次実行	並列実行	速度比
6	13.7	2.5	5.5
7	57.1	9.0	6.3
8	254.7	34.8	7.3

参考文献

- 1) Warren, D.H.D: An Abstract Prolog Instruction Set, SRI Technical Note 309 (1983).
- 2) Sequent Computer Systems, Inc: Guide to Parallel Programming (1985).