

# ハイパーキューブアルゴリズムの可視化に関する研究\*

5 E - 2

樋上 和彦<sup>†</sup><sup>†</sup>大阪電気通信大学

## 1 はじめに

高速な計算機を得るための手法として並列計算機がある。そのアルゴリズムは通信などのため理解が難しいという問題がある。並列アルゴリズムの理解を促すために、アルゴリズムアニメーションを利用するという解決法がよく用いられている。

本研究では、並列計算機の一つであるハイパーキューブ結合計算機に注目し、ユーザが書き換え可能な疑似コードにもとづいて、SIMDハイパーキューブアルゴリズムの動作を WS 上でシミュレートできる教育用可視化システムを作成し、いくつかのアルゴリズムを可視化した。これを利用することで、ハイパーキューブアルゴリズムを理解することが容易になると思われる。

## 2 ハイパーキューブ結合計算機

ハイパーキューブ結合は  $2^n$  個のプロセッサを使用し、プロセッサインデックスの二進表現のハミング距離が1のプロセッサを結合したものである [1, 2]。図 1 に構成図を示す。

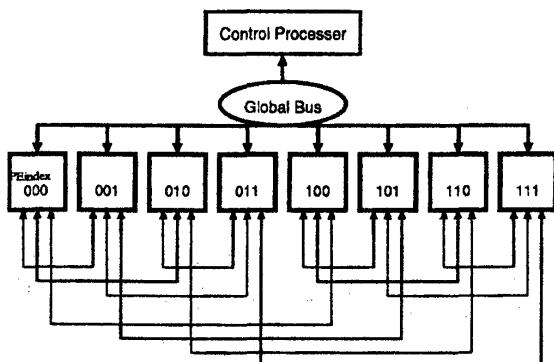


図 1: 想定したハイパーキューブ結合計算機の例 (SIMD)

## 3 処理の流れ

処理の流れを図 2 に示す。このシステムは lex/yacc で記述したトランслーター部と tcl インタープリタ言語と tk ツールキットで記述した可視化部の 2 部で構成される。

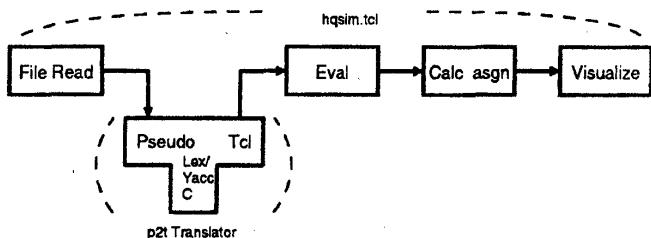


図 2: 処理の流れ

### 3.1 言語変換部分

システムに読み込まれた疑似コードは tcl コードに変換される。並列代入は tcl で記述した手続きに変換される。

### 3.2 可視化実装部分

変換されたコードは tcl インタープリタの "eval" コマンドで tcl コードとして実行される。並列代入手続きが呼ばれると、画面表示を行う手続きをさらに呼びだす。そして、リンクや変数の着色などで可視化をおこなう。プロセッサの表示は一次元である。tcl への変換時に、実行している行を表示する手続きを付加してあるので、代入操作が行われると疑似コードの実行行が赤に着色される。変数に対しては、プロセッサ内での代入先はオレンジ色に、通信代入元は緑、通信代入先は赤で着色している。また、通信がおこなわれるとリンクも着色される。

## 4 疑似言語について

SIMD ハイパーキューブアルゴリズムの記述に必要な構文を定義した。文献 [1] の疑似コードをもとに、C 言語が一般に普及している事から、利用者の利便を考え for while 二項演算子などは、C 言語風の仕様とした。細かな特徴を列挙すると、

- 関数はサポートしていない。
- コメントは # から行末までである。
- 大文字から始まる変数はハイパーキューブ内の変数、小文字から始まるものはコントロールプロセッサの変数である。
- 演算子の優先順位と評価は C とほぼ同じである。正確には tcl 言語のものをそのまま利用している。
- 変数の型は整数のみで、宣言は不用。
- INDEX はプロセッサインデックスを示す予約変数である。

\*Visualization of hypercube algorithm

<sup>†</sup>Osaka Electro-Communication University

などがある。ここでプレフィックスサムでの例で説明する。

```
# SIMD Prefix Sum
true = 1
false = 0
d = 4 # dim of hq
#init val
A : INDEX,(true) #intra asgn
S : A, (true)
T : A, (true)
for ( i = 0; i < d; i = i + 1)
BEGIN
    B @ i @ T ,(true) #inter asgn
    S : S + B ,(INDEX & (1 << i))
    T : T + B ,(true)
END
```

変数  $d$  にプロセッサ数を  $\log_2$  の形で格納する。簡単のため、この変数に代入があるとプロセッサ枠の表示をしている。プロセッサ内の並列代入は、

識別子：式，(プロセッサマスク)

と記述する。プロセッサマスクは、式の値が 0 以外で真となり実行される。INDEX はプロセッサインデックスを示す予約変数である。プロセッサ間代入は、

識別子 ◎ 通信次元 ◎ 識別子，(プロセッサマスク)

である。通信と同時に計算を行うことはできない。

## 5 実行画面

プロセッサ数  $2^4$  個でプレフィックスサムを実行したときの画面を示す。図 3 がハイパーキューブプロセッサの変数と通信の様子である。1 ステップごとに、変数と通信の様子が更新される。図 4 が実行行が着色された疑似コードである。他に SIMD shift, Consecutive Sum を可視化した。

## 6まとめ

ハイパーキューブ結合計算機のアルゴリズムの可視化システムを構築し、いくつかのアルゴリズムを可視化した。これを利用することで、ハイパーキューブアルゴリズムを理解することが容易になると思われる。

今後、より複雑な画像処理アルゴリズムや文字列アルゴリズムの可視化を行うための言語仕様の改善と、システムの改良などを行う必要がある。

## 参考文献

- [1] Ranka, S. and Sahni, S.: *Hypercube Algorithms*, Springer-Verlag (1990).
- [2] 梅尾博司：超並列計算機アーキテクチャとそのアルゴリズム，共立出版 (1991)。

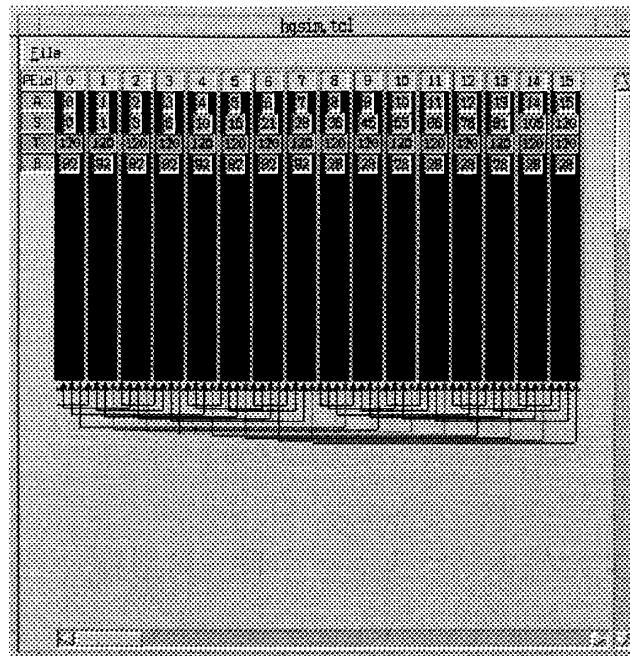


図 3: 変数と通信を表示するウインドウ

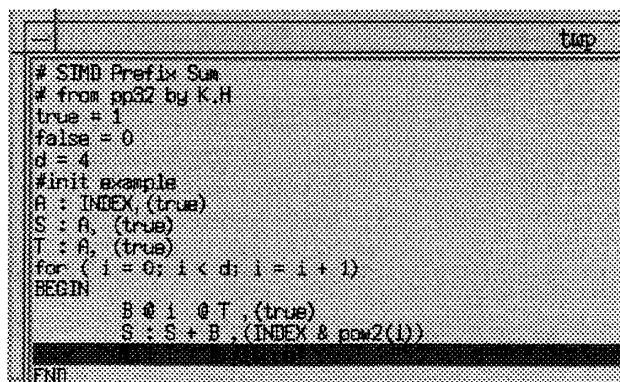


図 4: 疑似コード表示ウインドウ