7L−3

相互結合網シミュレータによるメッセージ生成規則とバッファ構成方式の研究

藤本茂訓　加納卓也　広田勝久　Andrew Flavell　高橋義造
徳島大学工学部知能情報工学科

1　はじめに

超並列計算機を構成する相互結合網において、高速なメッセージ通信は最も重要な要素である。そこで我々はメッセージ通信の要である通信バッファの構造をシミュレーションによって評価し、メッセージ生成規則とバッファ構成方式との間にある関係を明かにした。また、この関係を数学モデルを用いて検証した。その結果、可変ブロックサイズをもつバッファ構造が有効であることが判った。

2　シミュレータの説明

今回、開発した相互結合網シミュレータ(DIENS: DIversified Evaluation Network Simulator)[1]では、多数のプロセッサを計算機上で仮想的に結合することによって相互結合網を構築し、これを用いてパケット交換方式によるプロセッサ間通信をシミュレートすることができる。特定条件下における1回のシミュレーションの結果はメッセージの平均通信時間およびパケットの衝突によってできる通信時間の遅延割合として得られる。ここでは、多種の相互結合網を様々な条件下でシミュレートできるように開発されたDIENSの機能を説明する。

2.1　シミュレート可能な設定範囲

表1、表2は、DIENSがシミュレート可能な相互結合網のトポロジ、バッファの構成（ブロックの数とサイズ）、およびメッセージの生成規則（メッセージのサイズと生成速度）の設定範囲を表している。

表1 相互結合網とその設定範囲

相互結合網名	設定可能範囲	
Mandala結合網[2]	レベル 1～5	クラスタサイズ 3～16
Hypercube	次元 1～8	基数 2～16
無向Kautz[3]	桁数 1～7	基数 2～8
多進木	高さ 1～7	基数 2～15
トーラス	次元 1～8	1辺数（基数） 2～16

表2 バッファの構成とメッセージの生成規則の設定範囲

設定項目	設定範囲	設定値の単位
ブロック数	4,8,16,32,･･･,16k	[word (2bytes)]
ブロックサイズ	1,2,4,8,16,･･･,128	[block/buffer]
メッセージサイズ	1,4,16,64,･･･,256k	[word (2bytes)]
メッセージ生成速度	0.01, 0.1, 1, ･･,10万	[message/sec]

Study of message generation model and buffer configuration with interconnection network simulator. Shigenori FUJIMOTO, Takuya KANOH, Katsuhisa HIROTA, Andrew FLAVELL, Yoshizo TAKAHASHI. Department of Information Science and Intelligent Systems, University of Tokushima.

2.2　機能および特徴

DIENSは次のような機能をもつ。
・局所性のある通信パターンの発生。
・通信のデッドロックの検出。
また対象とする相互結合網を次のように仮定した。
・ルータ間の通信速度を20MB/secとする。
・1チャネルは16ビット幅とする。
・パケットのヘッダは6バイトとする。
・プロセッサとルータ間にあるNIU(Network Interface Unit)がパケット生成を行う。

3　シミュレーションによるバッファ構造の決定

2台のプロセッサ間でデータの転送を行う場合、このデータを小さなメッセージに分けて送信した方が、通信遅れが少なくて済む。またこれとは逆に送信手続き等のオーバヘッドの影響を少なくするために、データを1つのメッセージにまとめて送信した方がよい。このため、最適なメッセージサイズが存在する。

そこで相互結合網に要求されるのは、生成されたメッセージをできるだけ速く宛先のプロセッサに送り届けることである。ここではメッセージの高速通信を実現するために、ブロックサイズを変化させることが出来るバッファ構造が有効であることを、シミュレーションの結果を用いて説明する。

3.1　バッファ構造と通信時間の関係

まず最初にメッセージのサイズと生成速度が決まっているとき、メッセージ通信時間が最小となるような最適なバッファの構造（ブロックの数とサイズ）が一意に決まることを図1を使って示す。この図は表3の条件下でシミュレートした結果をグラフにしたものである。すなわちブロック数とブロックサイズを変化させていき、どの組合せのときが最も通信時間が短かったかを調べている。したがって横軸にブロックサイズ、縦軸にブロック数をとってあり、グラフ上の曲線は通信時間が等しくなるような等時間曲線を表わしている。このグラフをみると最適なバッファ構造はブロックサイズが256バイトでブロック数が2であることが分かる。

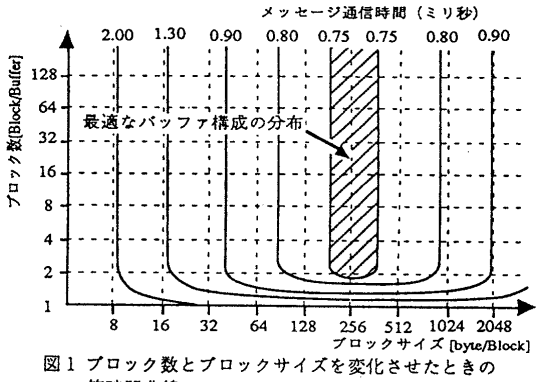

図1 ブロック数とブロックサイズを変化させたときの等時間曲線

表3 図1のグラフ作成用のシミュレーション条件

固定条件	レベル2，クラスタサイズ4のMandala結合網 メッセージサイズ　8kバイト メッセージ生成速度　1秒間に10個
変化条件	ブロックサイズ　8〜4kバイト ブロック数　1〜128

さて、このようなグラフが成り立つのは、次の理由によるものと考えられる。

(1)ブロックサイズが小さくなるほどメッセージのヘッダ部分の割合が増加するためオーバーヘッドが生じる。

(2) Store-and-Forwardの場合は、隣のルータに1パケット分送信し終わるまでその次のルータへ送信開始が出来ないので、パケットサイズが大きくなりすぎると通信遅れが生じる。

(3)メッセージ生成時にこれをパケットに分解するとき、断片化が生じる。

(4)ブロック数が小さくなると、1つのルータが1度にバッファリングできるパケット数が少なくなるため、パケットの衝突が頻繁に起こる。

上記の理由により最適なバッファ構造が一意に求まるわけである。

3.2 メッセージサイズとバッファサイズの関係

図2はメッセージサイズにを変化させて、それぞれ最適なブロックサイズを前節と同様にして求め、これらの対応をグラフ上にプロットしたものである。このグラフをみるとメッセージサイズの増加に伴い、最適なブロックサイズも単調増加しているのが分かる。したがってこの表から最適なブロックサイズとメッセージサイズの関係を近似式で表すと式1のようになる。ただしこの場合のAは2〜4である。

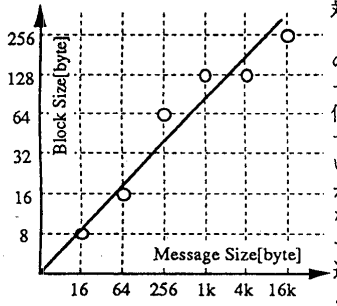

図2 メッセージサイズと最適なバッファサイズの関係

$$\text{最適なブロックサイズ} = A\sqrt{\text{メッセージサイズ}} \quad \ldots (1)$$

この結果、次のことが言える。

・ルータの設計において、ブロックサイズが可変であるようなバッファを採用すれば、送信したいメッセージのサイズから式1を用いて最適なパケットサイズ（＝ブロックサイズ）を求めることができる。これによっ最短時間でのメッセージ通信を行うことが可能となる。

4 数学モデルによる最適なバッファサイズの評価

ここでは、シミュレーションによって得られた最適なブロックサイズとメッセージサイズの関係式（式1）の正当性を、数学モデルを用いて検証する。またこのモデルを使ってより深い考察を行なう。

4.1 モデルの説明と関係式の導出

図3は1つのメッセージをNIU1からNIU2へ送信した場合の通信時間を表している。図中の1区間"⊢⊣"は1つのパケットを隣のルータ又はNIUへ送信する時間を表しており、図全体の時間の流れは左から右へ進んでいる。またNIU-ルータ間およびルータ-ルータ間の通信時間は同じであるとする。ただしこの図の場合、通信距離は2であり、また通信の衝突は起こらないと仮定している。

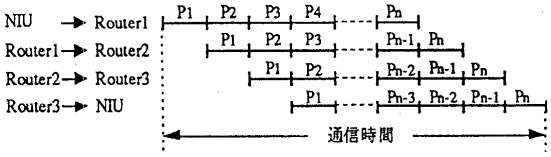

図3 数学モデルによるメッセージ通信時間

さて、ここで以下のような変数を定義する。

m：メッセージの長さ(byte)
x：ブロックサイズ
h：ヘッダの長さ(byte)
D：メッセージの通信距離
t：1バイト送信する時間

パケット数はm/xであり、1パケットの送信時間は(x+h)tとなるので、1メッセージの通信時間Tは次式で表される。

$$T = \left(\frac{m}{x} + D + 1\right)(x+h) \quad \ldots (2)$$

したがって、通信時間Tが最小値となるブロックサイズxは、

$$x = \sqrt{\frac{mh}{D+1}} \quad \ldots (3)$$

である。

4.2 最適なブロックサイズの関係式の考察

通信の衝突があまり生じないとき、式3から最適なブロックサイズはメッセージサイズの1/2乗に比例することが言えるので、式1は正しいと言える。さらに、ヘッダの長さhは一般に固定であるので、最適なブロックサイズは、正確には通信距離とメッセージサイズの両方に影響を受けることがわかる。また実際のメッセージ通信では、式3を用いて予め最適なブロックサイズを示すテーブルを設けておくとよい。

5 おわりに

バッファサイズには、より高速なメッセージ通信を可能にするための最適値が存在する。この最適値はメッセージサイズと通信距離から式3を用いて決定できる。従って、メッセージサイズに応じてブロックサイズが可変であるバッファを用いれば、より高速な通信が可能となる。

参考文献

[1]加納卓也, 広田勝久, 藤本茂訓, Andrew Flavell, 高橋義造: 階層構造を持つ分散メモリ型超並列計算機MANDALAの設計, 情報処理学会第87回計算機アーキテクチャ研究会 No.11 (1992)

[2]Andrew Flavell, Takuya Kanoh, Yoshizo Takahashi : Mandala: An Interconnection Network For A Scalable Massively Parallel Computer, 情報処理学会第43回全国大会, 4Q-13 (1991)

[3]Gerard J. M. Smit, Paul J. M. Havinga, Pierre G. Jansen : Generating Node Disjoint in Kautz Digraphs, private communicatin (1991)