社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2008－ARC－179 (16)
2008／8／6

相互結合網のスケール効果に関する初期評価

横田隆史[†1] 大津金光[†1] 馬場敬信[†1]

並列計算機，特に超並列規模の並列計算機システムにおいて，相互結合技術の重要性は論を待たない．これまで相互結合網に関しては，トポロジ，ルーティングアルゴリズム，フロー制御，仮想チャネル制御，デッドロック回避等の観点から活発に議論されてきた．これらの多くは，比較的小規模な網での評価結果をもって議論を行っており，我々はこれらの結果を外挿して超並列規模のシステムを予想している．しかし，こうした外挿の妥当性についてはあまり議論されていない．我々は，相互結合網シミュレータを並列化することで超並列規模に対応させ相互結合網のスケール性について評価を行う．本稿では，解析的に検討するとともに，基礎的な通信パターンについてのシミュレーション評価結果を示す．

Preliminary Discussions on Scaling Effects of Interconnection Networks

Takashi Yokota,[†1] Kanemitsu Ootsu[†1] and Takanobu Baba[†1]

Interconnection networks are of importance in massively parallel multiomputers. There are so many active researches on interconnection networks. They discuss topologies, routing algorithms, flow-control methods, virtual channels, deadlock avoidance, and so on. Most researches assume extrapolation of small-scale results to discuss the performance of large-scale systems. In this paper, we question the belief. We extended our interconnection network simulator to parallelized execution. We first discuss scalablity issues analytically. Then, we show some fundamental evaluation results by using the parallelized simulator.

1. はじめに

大規模並列計算機には，その処理性能に見合っただけの相互結合網が求められる．そのため相互結合技術の研究は，トポロジ，ルーティングアルゴリズム，フロー制御・仮想チャネル制御手法，デッドロック回避方法等，様々な観点から活発に検討されてきている．

こうした議論の中で，我々は，暗黙のうちに，規模に対するスケール性が成り立つことを前提としていることが多い．たとえば，ある新しい手法を提案し，その有効性をシミュレータにより評価したとする．我々は，この新手法が，超並列規模のシステムにおいても同様に有効なのか否か，どのように判断するだろうか．暗黙のうちに規模に対するスケール性が成り立つことを前提としていないだろうか．こうした疑問が本稿の出発点である．

我々は簡素化したモデルに基づきセルオートマトンを用いた評価により，$N \times N$ のトーラス網において．輻輳の発生・成長は $O(N)$ の速度で進行するが，解消は概ね $O(N^2)$ の速度になる事を示している[1),2)]．こうした速度差が相互結合網の性質の違いとして表れるなら，それはつまり網の規模によって性質が変化することを意味する．つまり，スケール性の前提が成り立たない可能性がある．

このため本稿では，できるだけ現実に近いモデルで評価するため相互結合網シミュレータを並列拡張し，2 次元トーラス網 (N-ary 2-cube) を対象として，一貫して相互結合網の規模と性能に関するスケール性の問題を議論する．このため，まず，2 節で解析的な手法により議論する．次に 3 節で並列化した相互結合網シミュレータについて述べ，4 に評価結果を示す．

†1 宇都宮大学大学院工学研究科
Utsunomiya University, Graduate School of Engineering

2. 次元順ルーティングの臨界転送負荷の解析

各ノードから送信頻度 r でパケットが送信されるものとする．パケットは 1 フリット (flit) 以上の長さを持ち，1 クロックサイクルごとに高々1 フリットが送信される．ここで r は，実行時間中にフリットを送信している時間の割合を指し，平均パケット長を l_p [flits]，パケット間のギャップの平均を g_p [cycles] としたとき，以下のようになる．

$$r = l_p/(l_p + g_p) \quad (1)$$

以下，この r を転送負荷と呼ぶ．

ここでランダム通信，すなわち，各ノードからランダムな送信先にパケットが生成されている状況を仮定する．二次元トーラスのサイズを $N \times N$ とする．次元順ルーティングでは，生成されたパケットのうち，$(N-1)/N$ の割合が最初に x 次元方向に移動し，残り

の $1/N$ が y 次元方向（x 軸成分が同じアドレス）に行く．

生成されたパケットのうち最初に $x+$ 方向に進むものの割合いは $(N-1)/2N$ であり，$x+$ 方向に 1 ホップ進んだのち，さらに $x+$ 方向に進むのは $(N-3)/2N$ となる．このように，h ホップ分離れたところでは，ノードから送信されたパケットのうち $(N-2h-1)/2N$ の割合のものが $x+$ 方向に進行する．

同様に，あるノードに着目したとき，そこから h ホップだけ離れた位置にあるノードから送出されたパケットがそのノードを通過する割合いも $(N-2h-1)/2N$ で表される．隣接ノード間の物理リンクはどのパケットでも共有されるから，リンク単体の通信負荷は個々のノードから発生した負荷の単純な和になる．ここで e-cube ルーティング[3),4)] を仮定し，リンクの負荷を検討する．

ルーティングは x 軸を優先する次元順で行う（次元順ルーティング）．どのパケットも仮想チャネル番号 0 で生成され，ラップアラウンドリンクを通過するときのみ仮想チャネル番号を増加する．$(0,0)$ ノードは $x-$ 方向から VC 0 の入力を持たないため，ここでの転送負荷は $(0,0)$ から $x+$ 方向に送信されるパケットのみである．このリンクの負荷は $l^{x+}_{(0,0)} = r(N-1)/2N$ である．上で述べたようにリンクの通信負荷は個々のノードから発生した負荷の単純な和であるから，結局，$(h,0)$ での $x+$ 方向のリンクの負荷は，

$$l^{x+}_{(h,0)} = \sum_{i=0}^{h} r(N-2i-1)/2N$$
$$= r(h+1)(N-h-1)/2N \qquad (2)$$

となる．なお，2 次元トーラス網を想定しているため，個々のノードからの $x+$ 方向の影響範囲は $N/2$ までである．したがって，上記の式は $0 \le h < N/2$ の範囲でのみ成立する．

$l_{(h,0)}$ は物理リンクの転送負荷を表しているから，$-\le l_{(h,0)} \le 1$ である[⋆1]．これにより最大値をとる h の位置と，そこでの転送負荷の値をもとに各ノードの最大転送レートを求めることができる．

ここで議論している，ラップアラウンド線を dateline とする単純な次元順ルーティングでは，$h = N/2-1$ が最大負荷となる地点である．このときの臨界転送負荷 r_c は式 (2) に $h = N/2-1$ を代入して，$r_c = 8/N$ となる．

3. 相互結合網シミュレータの並列拡張

我々の相互結合網シミュレータは，当初よりハードウェア実装を念頭に置いた設計になっており，演算ノード，ルータ等の構成要素を C++のクラスで表現し，ルータ間，ルータ・ノード間をポートで結んだ構成になっている．ルータは，各クロックサイクルの開始時に，隣接ルータ（およびノード）の出力ポートの情報を読み自身の入力ポートの値としてから，入力ポートおよびパケットバッファ内にあるフリットに対して所定の動作を行い，必要に応じて出力ポートを更新する．

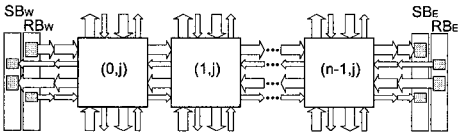


図 1 並列化シミュレータの通信部の構成

PC クラスタの使用により大規模な結合網に対応させる理由から，並列化にあたっては，MPI (MPICH 2[5)]) を用いる方針とした．各クラスタノードが $n \times n$ のノードを担当する．ただしノードは 2 次元メッシュ状に接続する．辺縁部のノードでの通信には，通信用のバッファを用いる．このバッファは，ルータの入出力ポートに直接対応する．クラスタノード間の通信は，このバッファを単位として `MPI_Send()`, `MPI_Recv()` のみを用いて行う[⋆2]．

逐次版シミュレータでは，パケットに関する統計情報の管理を行うクラスを設けることで集中管理できたが，並列版では分散管理が必要になる．たとえば，パケットが隣のクラスタノードに移動したときには，そのパケットの付随情報も同時に転送しなければならない[⋆3]．このためにポートの構成を一部改変している．また，クラスタノード境界で転送中のパケットに関する統計情報が（一時的ではあるが）両クラスタノードで重複するため，シミュレーション実行途中の統計情報の求め方に一部変更が必要であった．しかし全体的には，並列化のための作業は比較的軽微であった．

ルーティングアルゴリズムについては紙面の制約上詳細は省略するが，非適応型（決定的）ルーティングの典型として次元順 (do) ルーティング，適応型ルーティングとして大域的な輻輳情報を用いる Cross-Line ルーティング (cl)[6)]，また隣接ノードの輻輳情報を使うのみに制約した単純適応ルーティング (sa) の 3 者を用いている．

4. 並列シミュレータによる評価

4.1 結合網評価手法

これまで相互結合網の評価は，規模やルーティングアルゴリズム，負荷等のパラメータを設定したのち，系内にパケットが存在しない状態からシミュレーションを開始し，一定時間経過して系内の挙動が安定したのち（ウォーミングアップ），さらに一定時間の測定データの平均値を得ていた．転送特性を得るには，パ

⋆1 ここでは議論を簡素化するために VC 0 のみを考える．

⋆2 非同期通信等は使用しなかった．

⋆3 送信ノード，受信ノード，生成時刻，各種属性,... など．

ラメータを変えながら何度もシミュレーションを行う必要があった．

本稿での評価の場合，対象の相互結合網の規模が一定でないため，適切なウォーミングアップ時間を設定しにくい．また，大規模な結合網では，並列化により高速化したとはいえ，膨大な時間を要するためパラメータごとに独立したシミュレーションを繰り返す方法では限界がある．しかし，できるだけ滑らかな転送特性のデータを取得したい．

こうした理由から，我々は，1回のシミュレーション実行の中で，パラメータ（送信頻度）を僅かずつ変化させることで，連続的に転送性能の評価データを得ることにした．具体的には，シミュレーション開始時に与えた送信頻度の初期値 r_0，傾き sl (slew rate)，送信頻度の最大値 r_{max} の各パラメータから，時刻 t [cycle] での送信頻度 $r(t)$ を

$$r(t) = r_0 + t/sl \text{ (ただし} r \leq r_{max} \leq 1) \quad (3)$$

により求める．パケットは長さ l_p フリットの固定長であり，パケット間のギャップ g_p は式 (1) で求められる．各ノードは $1/g_p$ の確率でパケットの送出を開始する．一度送出を開始したら，1クロックに1フリットの割合で連続して送出する．

こうして，パケットの送出頻度を連続的に変化させながら，一定のウインドウ時間（100または10クロック）ごとの性能パラメータ（パケット数，送信パケット数，受信パケット数，平均レイテンシ，最大レイテンシ等）を測定・記録した．これにより，転送特性を1度のシミュレーション実行で取得できるほか，連続的な転送特性のグラフが一度で描ける*1．

4.2 ランプ応答

送信頻度を時間とともに漸増させたときの動的な挙動（過渡応答）を測定する．本来は時刻0での送信頻度をゼロとし，式 (3) に従って漸増させるののだが，大規模網ではシミュレーションの実行時間の点で大きな問題があることがわかった．このため，シミュレーション開始時に一定の送信頻度を与えるウォーミングアップ時間を設け，その後，slew rateを様々に変えて評価を行った．ウォーミングアップ時の送信頻度とその時間は，4.4節で行った結果を利用して決めている．

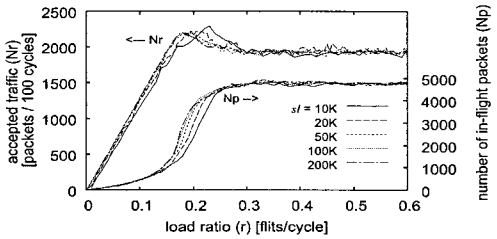
送信頻度の変化は以下のとおりである．送信頻度の初期値 $r_0 = 0.8r_c$，終了値を $r_x = 1.2r_c$，送信頻度の傾きを sl，ウォーミングアップ時間 $t_w = 10,000$ として，

$$r = \begin{cases} r_0 & (t < t_w) \\ r_0 + (t - t_w)/sl & (t_w \leq t < t_1) \\ r_x & (t_1 \leq t \leq t_2) \end{cases} \quad (4)$$

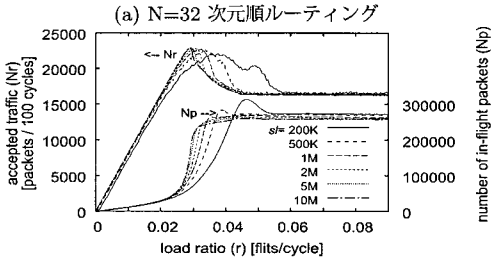
ただし，$t_1 = sl*(r_x - r_0) + t_w$, $t_2 = sl*(r_x - r_0) + t_1$.

図2に，式 (3) により転送負荷を変化させたときのスループットと平均レイテンシを示す．sl 値を大きくすることで安定した評価結果が得られることがわかる．ただし不必要に大きな sl ではシミュレーション時間が過大になる．図から，$N = 32$ の場合では $sl = 50K$ 程度で十分だが，$N = 256$ では $sl = 5M$ 以上の値が求められることがわかる．

*1 ただし，必要に応じて測定値のばらつきに対応する．



(a) N=32 次元順ルーティング



(b) N=256 次元順ルーティング

図2 ランプ応答特性（式 (3) の手法）

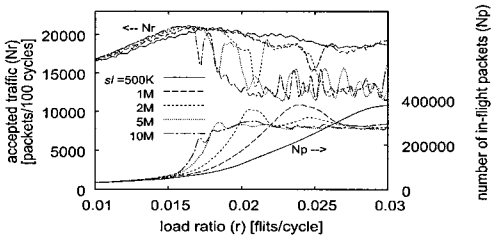


図3 ランプ応答特性（式 (4) の手法）

図3に，式 (4) により求めたランプ応答特性の評価結果例を示す．負荷の大きな領域において，スループットが不安定に変動する様子が観測されている．これに関しては後に議論する．

4.3 臨界転送負荷

相互結合網系が非飽和状態から飽和状態に遷移するときの送信頻度を臨界転送負荷 r_c とする．本稿では，臨界転送負荷を，受信パケット数の変化をもとに定義する．すなわち，非飽和区間では，送信頻度 r にほぼ比例した受信パケット数が得られるが，飽和状態に達すると，受信パケット数が飽和し，条件によっては逆に低下する．上述のように送信頻度を連続的に増加させていき，送信頻度と受信パケット数との比例関係が崩れる最小の送信頻度を臨界転送負荷とする．

具体的には，本稿では以下のようにして求めた．ま

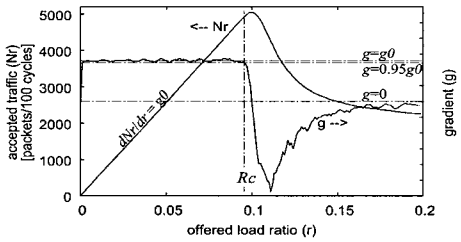


図 4　臨界転送負荷の定義

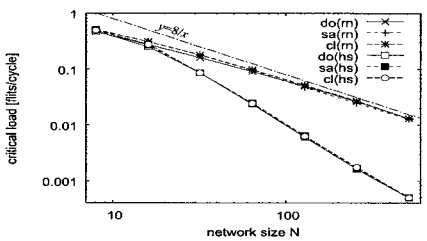


図 5　結合網のサイズ (N) と臨界転送負荷 (r_c) との関係

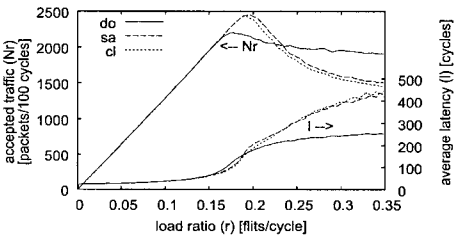
ず，非飽和区間における受信パケット数 (y) と送信頻度 (r) の関係を求める．網全体では 1 クロックサイクル間に $rN^2 \times N^2/(N^2-1)$ フリットが生成される．ここで $N^2/(N^2-1)$ を乗じているのは，自ノード宛のパケットが結合網を通過しないことの補正のためである．非飽和状態では，生成パケットと等しい分量だけ到着して網から除外されなければならない．したがって，単純に $y = rN^2 \times N^2/(N^2-1)$ [flits] である．

次に，ばらつきによる影響を排除するため，各ウインドウ時間での測定受信パケット数に対して，前後 200 サンプル（合計 401 サンプル）による移動平均を求める．同時に，各移動平均区間において最小自乗法による直線 ($y = a'r + b'$) へのフィッティングを行い，傾きを求める．この傾き (a') は，結合網が非飽和状態にあるとき，上で求めた a とほぼ等しい*1．網が飽和状態に遷移すると $a' < a$ となる．

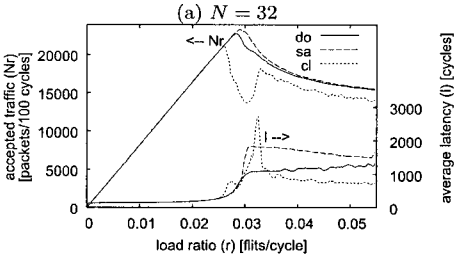
本稿では，移動平均による局所的な受信パケット数の傾き a' が，a の 95%を下回った時点を臨界とし，そのときの送信頻度を臨界転送負荷 r_c として定義している．**図 4** に臨界転送負荷を求めている様子を示す．

結合網の規模とルーティングアルゴリズムを変え，網の規模 (N) と臨界転送負荷 r_c との関係を求めた．その結果を**図 5** に示す．図には，次元順 (do)，単純適応 (sa)，Cross-Line (cl) のルーティングアルゴリズムと，ランダム通信 (rn)，1%ホットスポット通信 (hs) の通信パターンとの各組み合わせの結果を示している．また，同グラフ中には，2 節の結果（$r_c = 8/N$ のグラフ）も併せて示している．図 5 では判別しにくいが，網サイズによりルーティングアルゴリズムの優位性が変化する現象が確認されている．**図 6**(a), (b) に $N = 32$ と $N = 256$ での転送性能の測定結果を示す．ここでは，Cross-Line ルーティングと次元順ルーティングの優位性が逆転していることがわかる．網のサイズと臨界転送負荷についてマクロ的に見れば図 5 に示されるように，ルーティングアルゴリズムによらずほぼ一定の傾向が観測されるが，微視的に見ると個々のルーティングアルゴリズム間の相対的な性能は，網の規模の大小で保存されないことがわかる．

*1 むろん，移動平均をとった後でもばらつきを考慮する必要がある．



(a) $N = 32$



(b) $N = 256$

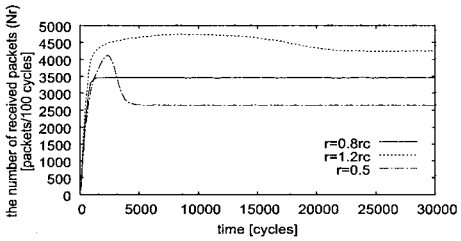
図 6　$N = 32$ および $N = 256$ での転送性能結果

4.4 ステップ応答

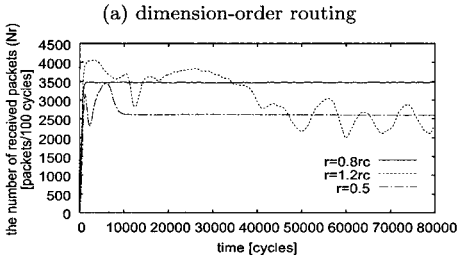
送信頻度を一定に保ったままシミュレーションを実行し，時刻 $t = 0$ からの動的な挙動（過渡応答）を測定する．急激な変化の様子を観測するため，ここでは，ウインドウ時間を 10 [cycles] としている．送信頻度は，前節で得られた臨界転送負荷 r_c を基準として，その 80%, 120%の負荷とした．また，r_c にかかわらず，$r = 0.5$ の場合の挙動も測定した．**図 7** に 512×512 トーラス網でのステップ応答の時系列変化の様子を示す．図 7(a) は次元順ルーティング，同図 (b) は Cross-Line ルーティングの場合である．

臨界転送負荷以下の負荷では比較的速やかに安定するが，$1.2r_c$ では収束までに多くの時間を要しており，また Cross-Line の場合は不安定な挙動が観測されている．荷重な負荷の場合 ($r = 0.5$)，応答当初に大きなオーバーシュートが表れるが，以降は安定している．

オーバーシュートは臨界転送負荷以上の負荷を与えた場合に共通して観測することができる．次元順ルーティングを用いた場合の，様々な網サイズでのステップ応答波形（時系列変化）を，時間，スループットと



(a) dimension-order routing



(b) Cross-Line routing

図 7 ステップ応答における時系列変化の様子（512 × 512 トーラス網）

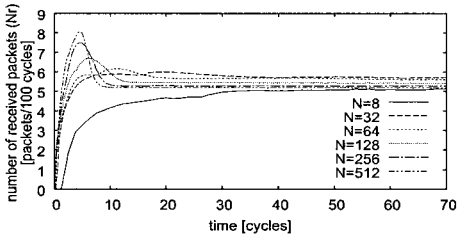
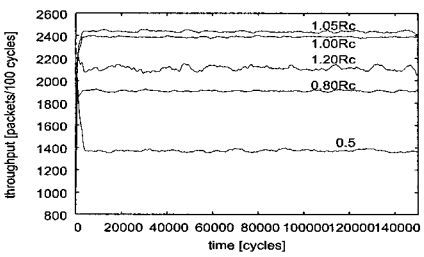


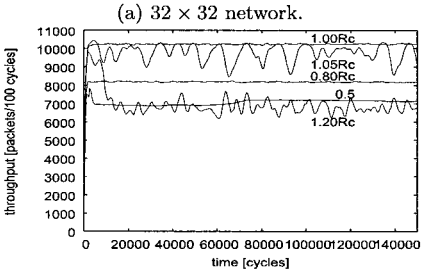
図 8 時間軸，スループットを網サイズ (N) により正規化したステップ応答

もに網サイズ (N) で正規化し同一のグラフ上に表したものを**図 8** に示す．このときの転送負荷は $r = 0.5$ である．このように，オーバーシュートが収束するまでの時間は網サイズにほぼ比例していることがわかる．なお，臨界転送負荷以下 ($0.8r_c$) ではオーバーシュートは観測されないが，安定するまでの時間は，ほぼ N に比例している．

図 7(b) のように，Cross-Line ルーティングでは臨界転送負荷をわずかに越える転送負荷での不安定性が観測されている．転送負荷量による不安定性が網の規模により変化する様子を**図 9** に示す．同図 (a) が $N = 32$ の場合，(b) が $N = 128$ の場合である．不安定性は網の規模により大きく変化することがわかる．臨界転送負荷をわずかに越えた程度の転送負荷で観測される，こうした不安定な挙動は，たとえば 7) で報告しているように，モデルを簡略化したセルオートマトンでの評価でも観測されている．



(a) 32 × 32 network.



(b) 128 × 128 network.

図 9 結合網規模によるステップ応答の時系列変化の違い（Cross-Line ルーティング）

4.5 インパルス応答

各ノードが一定の転送負荷（r_b: 背景負荷）でランダム通信をしている状態において，指定された時刻に一斉に指示された転送パターンのパケットを，指示された個数だけ連続して生成する．その後，各ノードは元の通り背景負荷でランダム通信をする状態に戻る．通信パターンとして

- ランダム通信:各ノードがランダムに転送先を選ぶ
- 転置通信：転移位置にあるノードに送信する
- ホットスポット通信：特定のノード（$N \times N$ 網において $(N/2, N/2)$ の位置にあるノード）に向けて送信する

の 3 種類を用いた．インパルスパケットの生成量は 1, 2, 5, 10, 20, 50 パケットとした．**図 10** に 128 × 128 網，Cross-Line ルーティング，転置通信でのインパルス応答波形の例を示す．応答波形自体は，結合網サイズによる大きな違いは見られなかった．

インパルスによる輻輳状態の持続時間 t_d を測定した．開始時間はシミュレーション中に明示するため特定されるが，インパルスにより発生した輻輳状態から復帰した時刻は検出が難しい．このため本稿では，ウインドウ時間内に観測された到着パケットの最大レイテンシを用いることとした．**図 11** にその例を示す．最大レイテンシがインパルス開始時間までの平均値に戻るまでの時間を計測している．**図 12** に，結合網の規模とインパルス応答の持続時間との関係を示す．転送パターンの違いにより，結合網規模に対する持続時間の傾きが異なる．また，図中では判別しにくいが，ルー

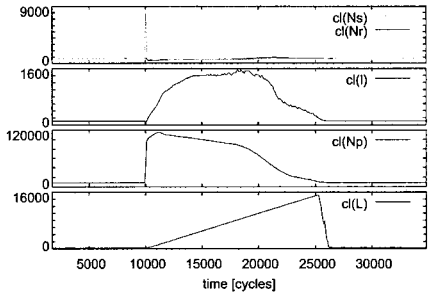


図 10 インパルス応答の時系列変化の例（128 × 128, Cross-Line）

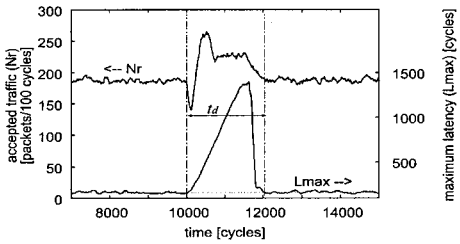


図 11 インパルス応答における持続時間 t_d の測定（32 × 32 トーラス網，Cross-Line ルーティング）

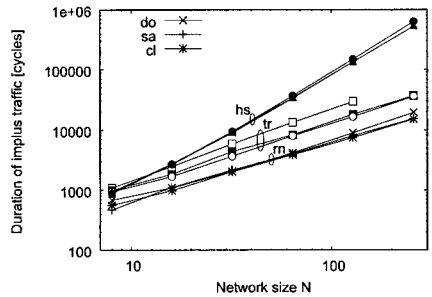


図 12 インパルス応答における持続時間の結合網サイズ依存性（$r_b = 0.8r_c$）

ティングアルゴリズム間の相対的な性能の優位性が，結合網規模によって変化する様子も観測されている．

5. 関連研究

相互結合網のスケーリング性に関して直接論じている研究はほとんどない．多くの場合，大規模システムでの評価を効率良く行うための方便としてスケール性を用いている．たとえば 8) などがある．これらは，スケール性が本当に（どこまで）成り立つのかの疑問に立脚している本稿とは逆のアプローチである．

臨界転送負荷に関しては，たとえば 9) で average lifetime が急増するポイントを critical load としている．本稿ではスループットが投入した転送負荷に対して比例関係を保てなくなるポイントとして臨界転送負荷を厳密に定義している．

6. おわりに

本稿では，2 次元トーラス網を対象として，相互結合網シミュレータを並列化することにより，網の規模に対する性能や特性のスケーリング性に関する基本的な評価を行った．具体的には，ランプ応答，ステップ応答，インパルス応答を設定し，網の規模と応答特性に関する基本的な評価結果を示した．

評価の過程において，スループットと転送負荷との比例関係の崩れをもとに臨界転送負荷を定義し，網の規模との関係を明らかにした．またインパルス応答での持続時間を定義し，臨界転送負荷と同様に網の規模との関係を明らかにした．これらの結果から，巨視的には結合網の性能と規模との間には一定のスケール性が認められるが，一方でルーティングアルゴリズム間の相対的な優位性は網の規模により変化することが明らかになった．

また，適応型ルーティングアルゴリズムでは，大規模化に伴い，臨界転送負荷をわずかに超えた転送負荷の状況において挙動が不安定になることが確認された．

謝辞 本研究は，一部日本学術振興会科学研究費補助金（基盤研究 (B)18300014，同 (C)19500037，若手研究 (B) 17700047），および宇都宮大学重点推進研究の援助による．

参考文献

1) 横田ほか，“セルオートマトンによる相互結合網の輻輳の解析”，情報処理学会論文誌：コンピューティングシステム，Vol.47, No.SIG 7 (ACS 14), pp.21-42, 2006 年 05 月．
2) T. Yokota, et al., "Phase Transition Phenomena in Interconnection Networks of Massively Parallel Computers," Journal of Physical Society of Japan, Vol.75, No.7, p.074801, June 26, 2006.
3) J. Duato, S. Yalamanchili, L. Ni, “Interconnection Networks: An Engineering Approach,” Morgan Kaufmann Pub., 2003.
4) W. J. Dally, B. Towles, “Principles and Practices of Interconnection Networks,” Morgan Kaufmann Pub., 2004.
5) “MPICH2: High-performance and Widely Portable MPI,” http://www.mcs.anl.gov/research/projects/mpich2/
6) 横田ほか，“大域的な情報を用いる相互結合網方式 Cross-Line”，情報処理学会論文誌：コンピューティングシステム，Vol.46, No.SIG 16(ACS-12), pp.28–42, 2005 年 12 月．
7) 横田ほか，“セルオートマトンによる相互結合網の間欠的な輻輳の解析”，情処研報，Vol.2006, No.88, pp.91–96, 2006 年 8 月．
8) L. Atzori, M. Isola, “A traffic scaling approach to speed up network simulations,” Proc. Global Telecommunications Conference, 2003 (GLOBECOM) Vol.7, pp.3862–3866, Dec. 2003.
9) T. Ohira, R. Sawatari, “Phase Transition in a Computer Network Traffic Model,” Physical Review E, Vol.58, No.1, pp.193–195, Jul. 1998.