

Title	再帰的相互結合網における適応型ルーティングに関する研究
Author(s)	川井, 雅之
Citation	
Issue Date	1999-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1256
Rights	
Description	Supervisor:堀口 進, 情報科学研究科, 修士

修士論文

再帰的相互結合網に対する 適応型ルーティングに関する研究

指導教官 堀口進 教授

北陸先端科学技術大学院大学
情報科学研究科情報システム学専攻

川井雅之

1999年2月15日

要 旨

超並列計算機の結合網には、科学技術計算に広く用いられている二次元格子結合を含み、ノード当りのリンク数が少数であるなど、実装性、耐故障性といった要件が求められている。再帰的相互結合網はメッシュ網やトーラス網を基に、再帰的にバイパスリンクを付加し構成された超並列計算機に適した結合網である。再帰的相互結合網はノード当りのリンク数を固定した階層構造を有する結合網であり、従来の相互結合網に比べて十分に優れた静的性能を有している。

再帰的相互結合網におけるルーティング（再帰ルーティング）は直径や平均距離などの点で十分に高い性能を有しているが、デッドロックフリーを保証していないものがある。また、転送経路が固定であるため混雑や故障に対応できないといった問題がある。

本論文では、再帰的相互結合網のデッドロックフリーな適応型ルーティング手法を提案する。シミュレーションにより適応型ルーティングの性能評価を行ない、従来手法と比較検討を行った。更に、提案する適応型ルーティングは仮想チャネルを増設する必要がなく、デッドロックフリーな固定型の再帰ルーティングに比べ非常に高い転送能力を有していることを示す。

目次

1	諸言	1
1.1	研究の背景と目的	1
1.2	本論文の構成	2
2	相互結合網とノード間通信	4
2.1	はじめに	4
2.2	超並列計算機における相互結合網	4
2.2.1	メッシュ結合網とトーラス結合網	5
2.2.2	階層構造相互結合網 TESH	6
2.2.3	再帰的相互結合網・一次元 SRT	8
2.2.4	再帰的相互結合網・二次元 SRT	9
2.3	ノード間通信	11
2.3.1	パケット転送方式	11
2.3.2	仮想チャネル	13
2.3.3	デッドロック	15
2.3.4	ルーティング	15
2.3.5	デッドロックフリー・ルーティング	16
2.3.6	適応型ルーティング	19
2.4	まとめ	21
3	一次元再帰網における適応型ルーティング	23
3.1	はじめに	23
3.2	諸定義	23
3.3	デッドロックフリールーティング	24
3.3.1	定義	25
3.3.2	一次元再帰網のデッドロックフリー・ルーティング	25
3.3.3	仮想チャネルの使用法	26
3.3.4	一次元 SRT におけるデッドロックフリー・ルーティング	27

3.4	デッドロックフリー・適応型ルーティング	29
3.4.1	一次元再帰網の適応型ルーティングの導出	29
3.4.2	一次元再帰網に対する適応型ルーティング	33
3.4.3	一次元 SRT における適応型ルーティング	33
3.5	シミュレーション評価	34
3.5.1	シミュレータ概要	34
3.5.2	シミュレーション概要	35
3.5.3	シミュレーション結果と考察	36
3.6	まとめ	37
4	二次元再帰網に対する適応型ルーティング	40
4.1	はじめに	40
4.2	次元オーダールーティングによる デッドロック回避と適応性	40
4.2.1	定義	41
4.2.2	デッドロック回避と Dimension order routing	42
4.3	二次元再帰網に適した適応型ルーティング	43
4.3.1	定義	44
4.3.2	二次元再帰網の適応型ルーティングの導出	44
4.3.3	二次元再帰網の適応型ルーティング	52
4.4	二次元 SRT における適応型ルーティング	53
4.5	シミュレーション評価	56
4.5.1	シミュレーションの概要	56
4.5.2	シミュレーション結果と考察	58
4.6	まとめ	62
5	適応型ルーティングの耐故障性	63
5.1	はじめに	63
5.2	適応型ルーティングの耐故障性	63
5.2.1	同次元迂回ルーティングの自由度	64
5.2.2	Dimension reversal routing の自由度	66
5.3	シミュレーションによる耐故障性の評価	69
5.3.1	シミュレーション結果と考察	71
5.4	まとめ	73
6	結言	74

目次

2.1	4 × 4 ノードから成るメッシュ網	5
2.2	4 × 4 ノードから成るトーラス網	5
2.3	2レベルから構成される TESH	6
2.4	4 × 4 構成の TESH の Basic Module)	7
2.5	32 ノードから成る基本型 1D-SRT	9
2.6	8 × 8 ノードから成る 1-shift 型 2D-SRT	10
2.7	一般的なルータの構成	11
2.8	パケットのブロック	13
2.9	仮想チャネルのあるノードの構成	14
2.10	ブロックの解消	14
2.11	デッドロックの例	16
2.12	仮想チャネルによるデッドロックの回避	17
2.13	二次元メッシュ上での Dimension order routing	18
2.14	2D-MESH 上での North-last 法	20
2.15	2D-MESH 上での静的な Dimension reversal routing	22
3.1	1D-SRT における再帰ルーティング	27
3.2	1D-SRT の再帰ルーティングの例	28
3.3	一次元再帰網の適応型ルーティング	33
3.4	1D-SRT のチャネル番号の割り当て	34
3.5	1D-SRT 上での同次元迂回ルーティングの例	35
3.6	32 ノードから成る 1-shift 型 1D-SRT	37
3.7	ランダム転送時の平均通信時間	38
3.8	他の結合網におけるランダム転送時の平均通信時間	39
3.9	仮想チャネル数と平均通信時間の関係	39
4.1	各ノードにおけるチャネル番号の割り当て	45
4.2	二次元再帰網の適応型ルーティング	55
4.3	2D-SRT での適応型ルーティングの例	57

4.4	ランダム転送時の平均通信時間	59
4.5	仮想チャネル数と平均通信時間の関係	59
4.6	256PEs での HC 網との比較	60
4.7	転送パターンが Matrix Transpose のときの仮想チャネル数と平均通信時間の関係	60
4.8	1024PEs 時の仮想チャネル数と平均通信時間	61
4.9	1024PEs での HC 網との比較	62
5.1	1D-SRT の自由度の変化	70
5.2	2D-SRT の自由度の変化	70
5.3	故障率とスループットの関係 (仮想チャネル 2)	72
5.4	故障率とスループットの関係 (仮想チャネル 4)	72

表 目 次

3.1	記号に関する定義	24
3.2	シミュレーションの概要 (1D-SRT での性能評価)	36
4.1	シミュレーションの概要— (2D-SRT での性能評価)	58
5.1	シミュレーションの概要 (耐故障性)	71

第 1 章

諸言

1.1 研究の背景と目的

自然科学におけるシミュレーションや VLSI 設計など，先端科学技術分野における大規模科学技術計算の需要は増大する一方であり，多数のプロセッサ (PE) を用いた超並列計算機による高速化が求められている．超並列計算機では，PE 間の通信性能が並列処理の効率に大きな影響を与えるために，様々な視点から数多くの相互結合網が提案されている [1]．

科学技術計算の多くは二次元または三次元構造のデータを対象とするため，格子型の結合網と適合しやすい．しかし，格子結合網はシステムの規模が大きくなると通信性能が急速に低下するという問題がある．そこで，超並列計算機用の相互結合網として，格子結合網やトーラス網を基本として，遠距離ノード間の結合を付加した相互結合網が多く提案されている [6, 7, 8, 3]．揚ら [6, 7] は，二次元の基本トーラス結合網に対して，格子間隔を $\sqrt{2}n$ (n は整数) 倍し 45 度傾けて再帰的に上位のトーラスを構成する相互結合網 RDT を提案した．Kirkman ら [8, 3] は，二次元の基本メッシュ結合網に対して，格子間隔を 2^n (n は整数) 倍した格子を，互いに重ならないように再帰的に重ね合せて構成する相互結合網 PEC について議論している．井口ら [4, 5] は，Shifted Recursive Torus (SRT) を提案している．トーラス結合網を基に，ノード間距離の異なるバイパスリンクを再帰的に付加した SRT は少ない次数で RDT や PEC と同程度の直径を実現している．

一方，ノード間のルーティング手法も通信性能を大きく左右する重要な要素であるため，これまでに多くの手法が提案されている．Dally ら [9] はルーティングアルゴリズムがデッドロックフリーとなる為の必要十分条件を示した．この必要十分条件は，現在多くのルーティングアルゴリズムでデッドロックフリーを保証するために用いられてる．Ni ら [12] はメッシュ網やハイパーキューブ網ではルーティングを次元順で行なうことでデッドロックが回避できることを述べている．また，Dally ら [10] は仮想チャネルを十分に設け，次元オーダを無視してもデッドロックフリー・ルーティングが可能となる手法 (Dimension

reversal routing) を提案した。Dally らの手法では、仮想チャネルの数に比例してメッセージは次元オーダを破ることができるため、仮想チャネルが多いほど適応性が高くなるという特徴がある。田辺ら [16] はハイパクロスバ網に対し、Dimension reversal routing が有効であることを確認している。Duato [13, 14] はデッドロックフリーを保証した適応型ルーティング手法を提案している。Duato の手法では、従来のデッドロックフリーなルーティングアルゴリズムに対し、仮想チャネルを新たに付加することで適応性が得ることができる。また、Duato はハイパーキューブに対し提案した手法を適用し、十分に高い性能が得られることを確認している。

Glass ら [15] は結合網内のリンクの論理的循環構造に着目し、その循環構造を回避する手法を提案した。Glass らの手法は Turn Model と呼ばれ、経路の選択性が他の手法と比べて高く、十分な適応性を得ることができる。また、Glass らはその手法をメッシュ結合網とハイパーキューブに適用し、その有効性を確認している。

しかしながら、いずれの手法もバイパスリンクを持つような再帰的相互結合網への応用は考慮されていない。また、再帰的相互結合網では、バイパスリンクを使用を視野に入れたルーティング手法(再帰ルーティング)を用いるが、その多くがデッドロックを保証していなかったり、適応性や耐故障性を有していないなどの問題がある。

RDT においては舟橋らに [17] より、Duato の必要十分条件と Turn Model を適用する方法が議論されているが、RDT のような再帰網の特色であるバイパスリンクの使用までは考慮されておらず、バイパスリンクを使用することにより通信距離を短縮できるという RDT の特性を十分に発揮してるとはいえない。

本研究では再帰的相互結合網に対して、その特色であるバイパスリンクの使用を考慮したデッドロックのない適応型再帰的ルーティング手法を提案する。そのため、再帰ルーティングがデッドロックフリーであるための必要条件を厳密に導出する。次に、バイパスリンクの性質に着目し、デッドロックフリーな再帰ルーティングとは異なるルーティング手法が可能となること示し、それらが選択的に使用可能であることを示す。そして、それらを複合的に用いる適応型ルーティングアルゴリズムを提案し、その動的通信性能をシミュレーションにより詳細に議論する。

1.2 本論文の構成

本論文の構成は次の通りである。第 2 章では、まず、代表的な格子型結合網であるメッシュ結合網、トーラス結合網について述べ、それらを拡張した結合網である、階層構造相互結合網 TESH について述べる。さらに再帰的相互結合網の最も代表的な結合網である SRT について詳細に述べる。次に一般的なノード間の通信方法について述べ、代表的なルーティング手法について議論する。第 3 章では、一次元再帰網に対する適応型ルーティング手法を提案し、第 4 章では、二次元再帰網に対する適応型ルーティング手法を提案す

る．また，各々にこれらがデッドロックフリーであることを証明し，シミュレーションにより提案した適応型ルーティング手法の動的通信性能について詳細に議論する．第5章では，提案する適応型ルーティング手法の耐故障性について，選択経路の自由度の解析と動的通信性能評価の2つの側面から議論する．提案した適応型ルーティング手法では経路選択の自由度が高いことを示す．また，シミュレーションによる動的通信性能評価により提案手法の高い耐故障性について詳しく議論する．第6章は結論である．

第 2 章

相互結合網とノード間通信

2.1 はじめに

超並列計算機の性能を決定する重要な要素である，結合方式(トポロジ)とノード間通信について詳しく議論する．2.2 節では代表的なトポロジについて述べる．

2.3 節ではノード間通信について述べる．ノード間通信はスイッチングや仮想チャネル，フロー制御，ルーティングアルゴリズムといったいくつかの機構，機能により実現される．代表的な 3 つにスイッチングについて詳細に議論し，フロー制御についても検討する．ルーティングアルゴリズムは大きく固定型ルーティングと適応型ルーティングとがあることを述べ，それぞれの特長について詳しく述べる．また，ネットワーク通信において重要な問題であるデッドロックについて述べ，デッドロックの回避法について検討を行ない，ルーティング手法との関連について詳しく述べる．

2.2 超並列計算機における相互結合網

数値計算やパターン処理など膨大な計算量が要求される科学技術計算において超並列計算機による高速化が強く望まれている．超並列計算機では多数のプロセッサ要素 (PE あるいはノード) を用いるが，多くの並列処理ではプロセス間に依存関係が生じるため，ノード間で通信を行ないながら処理が進められる．超並列計算機ではノード間の通信性能が並列処理の効率に大きな影響を与えるために，通信性能向上のために数多くの相互結合網が提案されている．

科学技術計算の多くは二次元または三次元構造のデータを対象とするため，格子型の結合網と適合しやすい．しかし，格子結合網はシステムの規模が大きくなると通信性能が急速に低下したり，レイアウト面積が増大してしまうなどの問題がある．そこで，超並列計算機用の相互結合網として，メッシュ結合網やトーラス結合網を基本として遠距離ノード

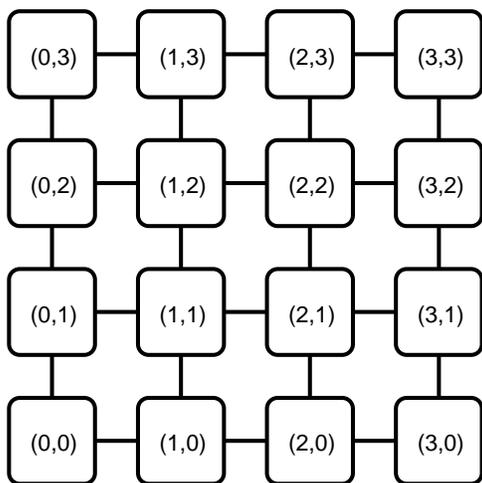


図 2.1: 4 × 4 ノードから成るメッシュ網

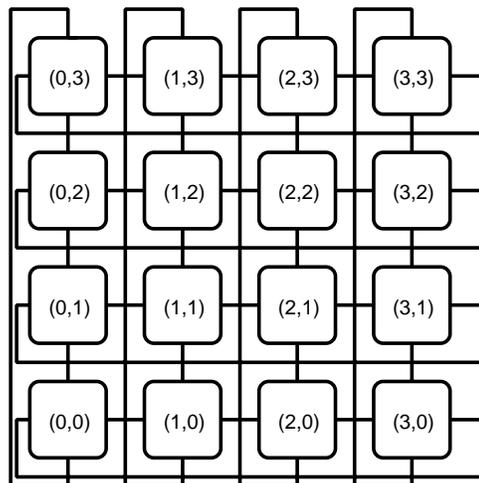


図 2.2: 4 × 4 ノードから成るトーラス網

間の結合を付加した結合網や、それらを階層的に接続した結合網などが注目されている。本節では基本的な結合網であるメッシュ結合網とトーラス結合網について述べ、それらを階層的に接続して構成される階層的結合網の代表的結合網である TESH について述べる。さらに代表的な再帰的相互結合網である一次元 SRT、二次元 SRT について詳細に述べる。

2.2.1 メッシュ結合網とトーラス結合網

数多く提案されているネットワークの中でも、メッシュ結合網とトーラス結合網は最も基本的なネットワークで、それを一般化した k -ary n -cube を含めると多くのネットワークに応用されている。

n 次元のメッシュ結合網やトーラス結合網は $K_0 \times K_1 \times \cdots \times K_{n-1}$ ノードから構成される。ここで k_i は各次元のノード数であり基数と呼ばれる。任意のノード X のアドレスは、

$$X = (x_{n-1})(x_{n-2}) \cdots (x_1)(x_0),$$

のように一意に割り当てられる。ここで x_i は $0 \leq x_i \leq K_i$ を満たす任意の整数である。任意の2つのノード X, Y は $x_i = y_i \pm 1$ が成り立つ i が一つだけ存在する時、接続することができる。ただし、メッシュ結合網では境界同士を結合することはできない。

図 2.1, 2.2 に 4 × 4 のメッシュ結合網, トーラス結合網をそれぞれ示す。

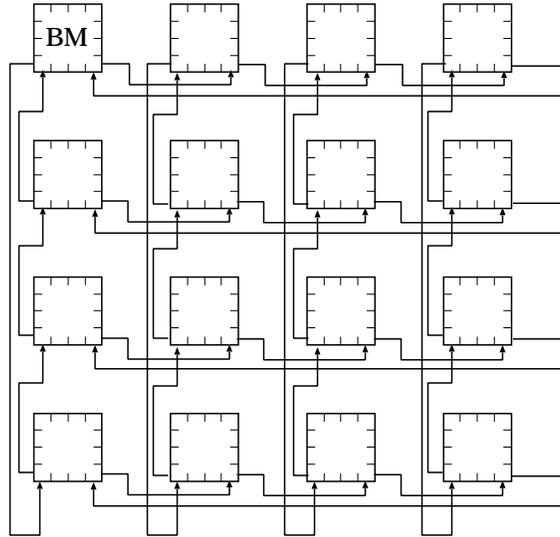


図 2.3: 2レベルから構成される TESH

2.2.2 階層構造相互結合網 TESH

階層構造相互結合網 Tori connected mESHes(TESH)[18] は 4×4 といったような比較的小さいサイズのメッシュ結合網を基本要素 (BM) とし, BM を 2 次元トーラスにより接続する階層的な相互結合網である. 図 2.3 に BM のサイズが 4×4 の 2 階層型の TESH を示す.

TESH のノード数 N は, BM のサイズが $2^m \times 2^m$ のとき, $N = 2^{2mL}$ と表される. ここで L は階層のレベルを表し, その取り得る最大値 L_{max} は $2^{m-q} + 1$ である. q は多重度を表し, 同じレベルとのリンクがどの程度あるかを示す.

任意のノード X は,

$$\begin{aligned} X &= x_{n-1}x_{n-2} \cdots x_3x_2x_1x_0, \\ &= x_{2L-1}x_{2L-2} \cdots x_3x_2x_1x_0, \\ &= (x_{2L-1}x_{2L-2}) \cdots (x_3x_2)(x_1x_0), \end{aligned}$$

と表される. 各 BM は図 2.4 に示されるように階層のレベルに応じたリンクの入出力位置があらかじめ決められている. 階層のレベルによって BM 間の距離は決まっているので, 各 BM はその距離に応じて, その距離だけ離れた BM とトーラス結合することになる. つまり, 階層のレベルが 2 の時, 各 BM は隣接する BM とトーラス結合する. 同様に階層のレベルが 3 の時も, 各 BM は 4 つ先, あるいは手前の BM と接続される. 階層のレベルが 4 以上であっても, 以上で述べた操作を繰り返すことにより BM 間を接続することができる.

2.2.3 再帰的相互結合網・一次元 SRT

一次元 SRT (1D-SRT) [4, 5] は一次元トーラス結合網 (リング結合網) にバイパスリンクを再帰的に付加することで構成することができる。1D-SRT は、ノード数 $N(N = 2^n)$ から成るリング網を基に構成される。リング網のあるノードを番号 0 とし、ノードを昇順に番号付け、次の基本トーラスを定義する。

定義 1 (基本トーラス) N ノードから成る基本トーラスのノード $x(0 \leq x < N)$ は、隣接する左右のノード $(x \pm 1) \bmod N$ と結合される。□

トーラスの両端は互いに結合されるので、ノード 0 の隣接ノードは 1 と $N - 1$ である。基本トーラスを構成する結合リンクをレベル 0 のリンクと呼ぶ。

トーラス結合網の直径および平均距離を短縮するために、基本トーラスにバイパスリンクを付加する。バイパスリンクにより構成される上位トーラスは次の定義に従う。

定義 2 (1D-SRT) $N(= 2^n)$ ノードから成る基本トーラスから、ノード番号 x が

$$x \bmod 2^l = \min(2^{l-1}, 2^T), \quad (2.1)$$

を満たすノードを取り出し、環状に接続する。この手続きをレベル l が 1 から $l_{max} - 1$ まで繰り返す。 l_{max} は基本型 1D-SRT の最大のレベルであり、 $l_{max} = \log_2 N = n$ である。また、 T は 1D-SRT のタイプを表す。□

(2.1) 式を満たすノードの集合をレベル l のノードと呼び V_l とする。 $l \geq 1$ のレベルを上位レベルと呼ぶ。1D-SRT では T の採り方によりレベル 0, l_{max} , $l_{max} - 1$ ノードの結合方法が若干異なる。 $T = n$ の場合を基本型と呼ぶ。基本型では、ノード 0 は、どのような $l \geq 1$ でも (2.1) 式を満たさないので、上位レベルを持たない。このため、ノード 0 のレベルは 0 とする。 V_l を環状に接続するリンクを上位リンクと呼び、 E_l とする。 x_l はレベル 0 のリンクとは別に、レベル l のリンクによって 2^l 離れた同じレベルのノード $((x_l \pm 2^l) \bmod N)$ に接続される。従って x_l は、 $(x_l \pm 1) \bmod N$, $(x_l \pm 2^l) \bmod N$, の 4 ノードに接続される。また、 $T = n - 2$ としたときを Long-Span 型、 $T = n - 3$ としたときを Short-Span 型と呼び、レベル 0, レベル l_{max} ノードでも上位レベルリンクを持つことができる。

図 2.5 に 32 ノードから成る 1D-SRT のリンク結合を示す。ノードはレベル毎に 2 本のリンクを持つため、1D-SRT のどのノードも、基本トーラス (レベル 0) を構成する 2 本と、上位リンクを構成する 2 本の合計 4 本のリンクを持つ。但しノード 0, $N/2$ は上位レベルが割り当てられないため上位リンクが無くレベルは 0 となる、またノード $\frac{1}{4}N$, $\frac{3}{4}N$ に接続される上位リンクは 1 つであり、レベルは l_{max} である。

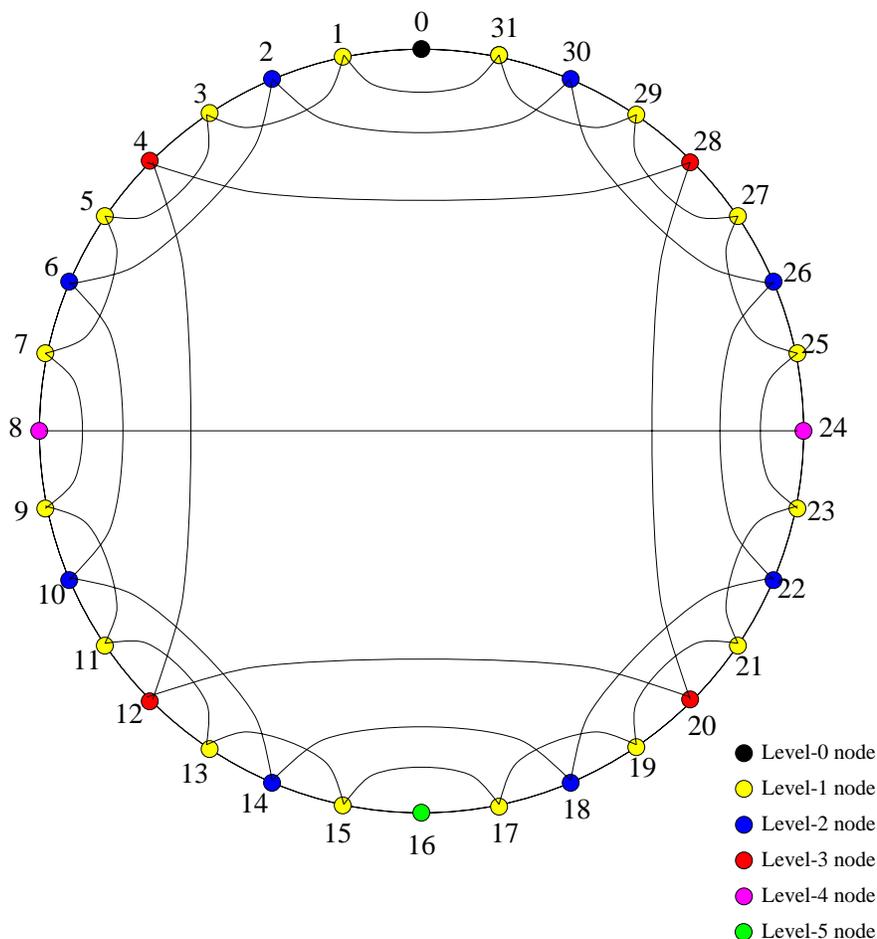


図 2.5: 32 ノードから成る基本型 1D-SRT .

2.2.4 再帰的相互結合網・二次元 SRT

本節では、1D-SRT を 2 次元に拡張した 2D-SRT について述べる。図 2.6 に示すように、 $N \times N (N = 2^n)$ のノードから成る 2D-SRT は、 x 方向に直線状に配置した N ノードから成る 1D-SRT を y 方向に積み重ねることにより構成される。2D-SRT は、 x 方向とともに、 y 方向にも 1D-SRT が構成可能でなくてはならない。そこで、積み重ねる段ごとに 1D-SRT の原点ノード (ノード 0) の位置をシフトする。シフト幅を変化させることにより、様々な 2D-SRT が構成可能である。ここでは、2D-SRT の一般形について説明する。

定義 3 (2D-SRT) $N \times N (N = 2^n)$ のノードから成るトーラス上で、アドレスを左下から順に与え、 (x, y) と表記する。1D-SRT と同様に、トーラス上のノードに上位レベルを割り当てる。レベル l のノード (x_l, y_l) は次の式を満すノードである。

$$(x_l + s_x \cdot y_l) \bmod \min(2^l, 2^T) = 2^{l-1} \quad (2.2)$$

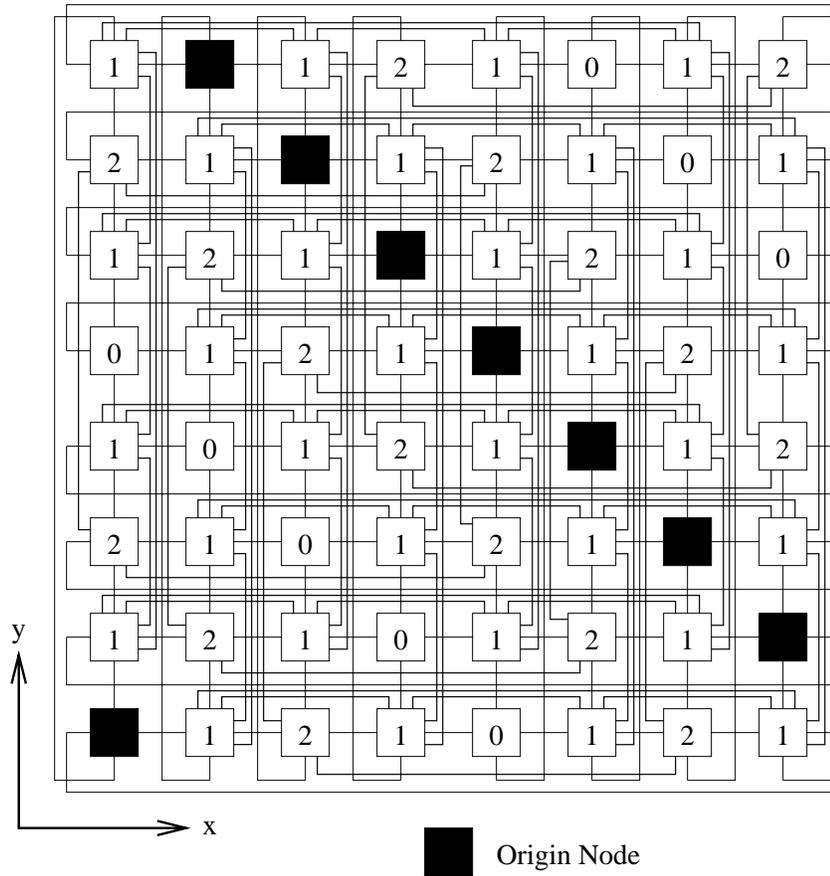


図 2.6: 8×8 ノードから成る 1-shift 型 2D-SRT .

ここで s_x は x 方向のシフト幅である . ノード (x_l, y_l) は隣接する 4 ノード $((x_l \pm 1) \bmod N, (y_l \pm 1) \bmod N)$ と , 2^l 離れた 4 ノード $((x_l \pm 2^l) \bmod N, (y_l \pm 2^l) \bmod N)$ と接続される . \square

x 方向に 1D-SRT を構成すると同時に , y 方向にも 1D-SRT が構成されるためには , シフト幅 s_x が以下の条件を満たす必要がある .

1. 原点ノードが各行・列に 1 つだけ存在する
2. x と y が転置関係にあること .

原点ノードとは , 1D-SRT のノード番号が 0 のノードのことである , また , 図 2.6 にシフト幅が 1 の 1-shift 型 2D-SRT を示す .

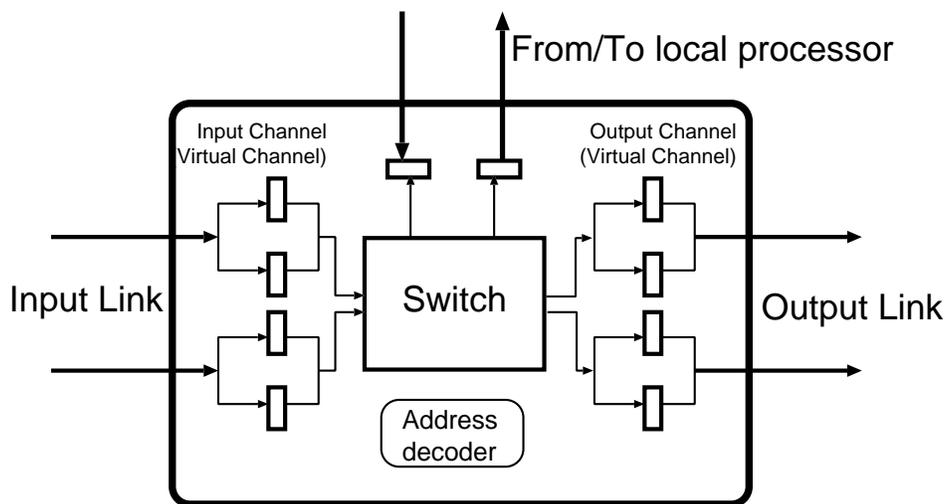


図 2.7: 一般的なルータの構成

2.3 ノード間通信

並列処理では処理されるデータに依存関係などが生じるため、ノード間で通信を行ないながら処理を進める。それゆえ、ノード間通信は並列処理の性能を左右する重要な要素である。そのため、従来よりノード間通信の性能を向上させる研究が盛んに行なわれている。

ノード間通信では、メッセージはパケットの形でノード間を転送される。通常、パケットは1クロックでパケット全体を転送することはできないので、フリット単位での転送が行なわれている。フリットとは各ノード間を1クロックで転送できる単位である。パケットの形式は並列計算機によって異なるが、多くは8bitから64bitのデータ幅を持ち、数フリットからなるヘッダとそれ以外からなるデータ本体で構成される。

ノード間通信はパケット転送方式、仮想チャネル、フロー制御方式、ルーティングアルゴリズムといったいくつかの機能により実現され、ルータと呼ばれる専用のハードウェアにより行なわれる。一般的なルータの構成を図2.7に示す。

また、以下に各々の機能について議論する。

2.3.1 パケット転送方式

パケット転送方式は大きく、Store and Forward方式(SF方式)、Wormhole方式(WH方式)、Virtual Cut Through方式(VT方式)の3つに分けることができる[12, 20]。

SF方式は初期の並列計算機で用いられた手法である。SF方式では各ノードは1パケット分のバッファを持ち、フリットは隣接するノードのバッファに逐次格納される。パケット全体が格納されたら、次のノードへ転送を開始する。この操作を繰り返すことにより目

的ノードまでパケットを転送する。SF方式は非常に簡単な方式であるが、1パケット分のバッファを必要とすること、ネットワークのサイズが大きくなるにつれ通信レイテンシが大きくなることといった問題がある。

WH方式では各ノードは基本的には1フリットを格納できるバッファを持てば十分である。WH方式では、パケットの先頭は、転送先のノードのバッファが空いていけば次々に先へと進むことができる。転送先のバッファが他のパケットにより使用されている場合は、そのパケットはそれ以上は進むことができず、転送先のバッファが空くのを待ってから転送を行なう。制御はSF方式に比べ複雑になるが、最近の並列計算機ではパケットの転送にルータと呼ばれる専用のハードウェアを用いるおりそれほど大きな問題にはならない。しかしながら、WH方式では各ノードは1パケット分のバッファしか持たないので、あるパケットを構成するフリットがいくつものノードにまたがり、その経路上のバッファを占有して転送することになる。そのため、あるパケットAが他のパケットBによりブロックされた場合、パケットAの経路上にあるバッファはパケットAそのものに占有されたままになり、新たなブロックを誘発してしまう可能性がある。

VT方式はSF方式とWH方式を組み合わせた方式で各ノードはSF方式同様、1パケット分のバッファを持つ。一方でパケットの転送方法は、WH方式のようにパケットの先頭は本体の到着を待たずに次の転送先へと転送が可能である。また、転送先のバッファが使用されていた場合は、SF方式のように、各ノードに設置されているバッファに次々と格納されていく。そのため、WH方式とは異なり、経路上に存在するフリットにより新たなブロックを誘発する可能性は低くなる。しかしながら、SF方式同様、各ノードに1パケット分のバッファが必要であるという問題がある。

次に各方式の通信時間について議論する。いま、ネットワークの直径を D 、パケットのヘッダ部のフリット数を F_h 、パケットのデータ部のフリット数を F_d とすると、SF方式では、いったんすべてのフリットが転送先で格納されるので、パケット全体の転送に

$$(F_h + F_d) \times D$$

だけ時間を要する。一方、WH方式、VT方式では各ノードではヘッダ部のみを格納すればよいので、パケット全体の転送に要する時間は

$$F_h \times D + F_d$$

となる。通常、ヘッダ部は数フリットで済むため、システムが大規模になるにつれ通信時間の点では、WH方式、VT方式のほうが有利になる。

現在の並列計算機ではノード間通信のための専用のルータを持つことと、システムが大規模になった場合、WH方式、VT方式のほうがSF方式に比べ通信時間の点で有利であることからWH方式、VT方式を用いるのが主流である。なお、WH方式とVT方式のどちらが有効かは、必要とするバッファ数、つまりパケット長により決まるが、後で述べる

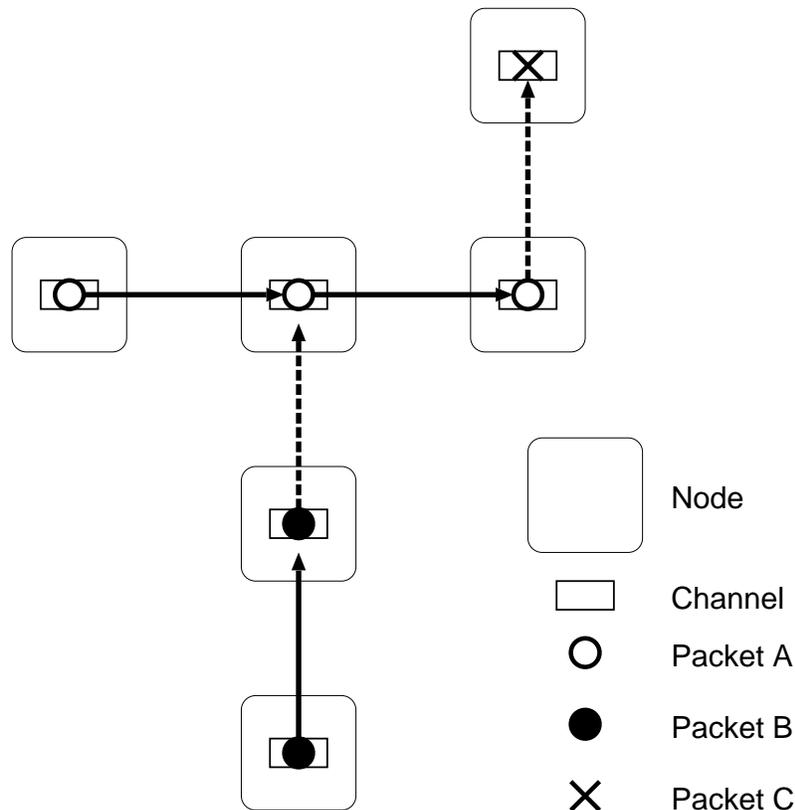


図 2.8: パケットのブロック

いくつかの手法により連鎖的なブロッキングを減らすことができるため，WH方式のほうがVT方式に比べ有効である場合が多い。

2.3.2 仮想チャネル

WH方式ではパケットの先頭がブロックされると連鎖的に新たなブロックを誘発する可能性がある。そのため図2.8におけるパケットBのヘッダのように物理的なリンクは空いているにもかかわらず，そのリンクを使用できないといった状況が生じる。そこで，図2.9のように，物理リンク1つにつき複数の論理的に独立なバッファ（仮想チャネル）を設け，マルチプレクサによりリンクを共有する手法が提案されている。図2.10に示すように，仮想チャネルによりバッファが塞がっていても他の空いているバッファを使用し転送を行い，全体のスループットを向上させることが可能である。仮想チャネルを用いたフロー制御を仮想チャネルフロー制御と呼ぶ[11]。仮想チャネルフロー制御は物理リンクを増設せずにノード間の転送性能を向上させることができ，転送方式を問わず適用できる。パケットにブロッキングが生じやすいWH方式に対して用いると特に効果的である。しかしながら，

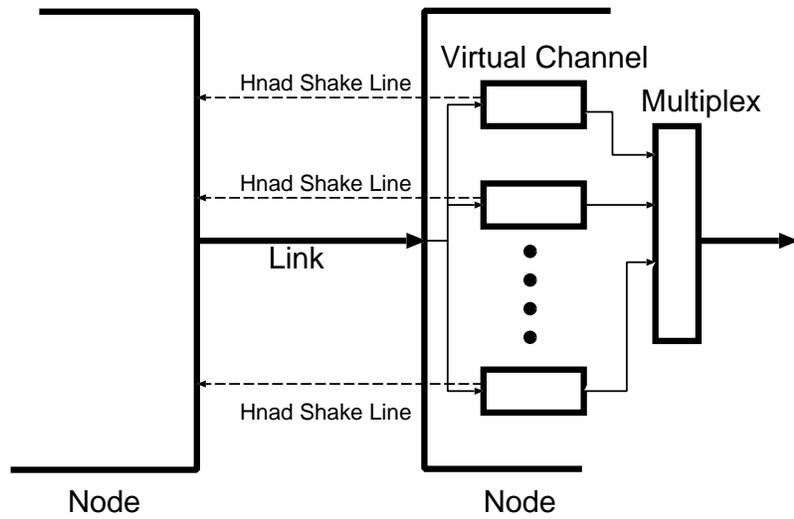


図 2.9: 仮想チャンネルのあるノードの構成

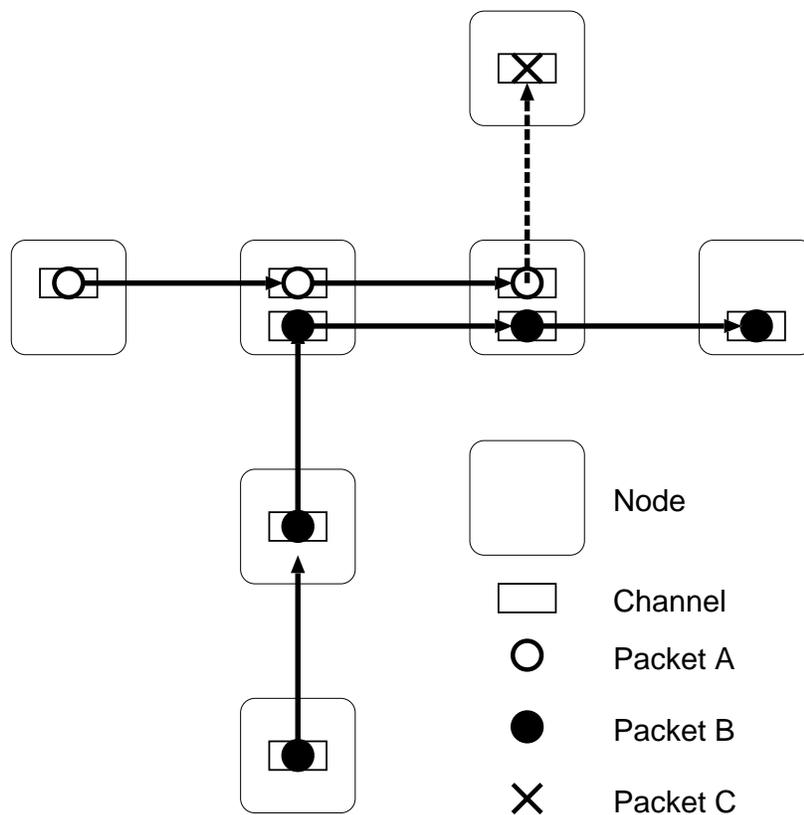


図 2.10: ブロックの解消

仮想チャネルフロー制御ではノード内に新たにバッファを追加する必要があり，同時に実際の転送に必要なハンドシェイクラインも新たに設置しなければならない．したがって，どの程度仮想チャネルを追加するかはパフォーマンスとのトレードオフになる．

仮想チャネルフロー制御を用いると，どのバッファに物理リンクを使用させるかといった問題が生じてくる．パケットに優先順位を付け，それに従いパケットの転送を行なう手法なども提案されているが，構造が簡単なラウンドロビンに比べそれほど性能が向上しないことから，多くの場合ラウンドロビンを用いている [12] ．

なお，本論文では以後，断りのない限り転送方式には WH 方式，仮想チャネルのフロー制御にはラウンドロビンを用いる．

2.3.3 デッドロック

ノード間通信は，メッセージを目的とするノードまでパケットが転送要求と転送を繰り返すことにより実現される．

ネットワークが十分に空いている場合はパケットの転送要求に滞りなく受け入れられ，転送が行なわれていく．しかし，ネットワークが混雑してくると複数のパケットが循環的にチャネルを要求するといった状況が生じる．一度このような状況が生じると，その循環的な要求が解除されるまで全体として全く通信が行なえなくなってしまう．こうした複数のパケットによるチャネルの循環要求をデッドロックと呼ぶ．図 2.11 にデッドロックの状態になっている例を示す．

デッドロックはフロー制御方式によらず生じる可能性があるが，パケットが複数のノードにまたがり，かつチャネルに 1 フリット（あるいは数フリット）分のバッファしか持たない WH 方式ではデッドロックを頻繁に引き起こす可能性がある．そのため何らかの方法によりデッドロックを回避する必要がある．多くの場合，図 2.12 のように仮想チャネルを付加したり，転送要求に循環が生じないように転送することでデッドロックを回避する．

2.3.4 ルーティング

相互結合網により接続された並列計算機におけるノード間通信では，メッセージはいくつかのノードを中継しながら転送される．あるメッセージがどのような経路を辿るかはルーティングアルゴリズムにより決定される．そのため，ルーティングアルゴリズムも並列計算機の性能を左右する重要な要素となる．

ルーティングアルゴリズムは経路の決定方法により大きく固定型と適応型に分けることができる．固定型ルーティングは目的ノードまでの経路が固定であり，方向転換などが制限される．そのため混雑や故障等を回避することはできないが，比較的容易にデッドロックフリーを実現できる．

適応型ルーティングは目的ノードまでの経路をある程度動的に選択することができる．

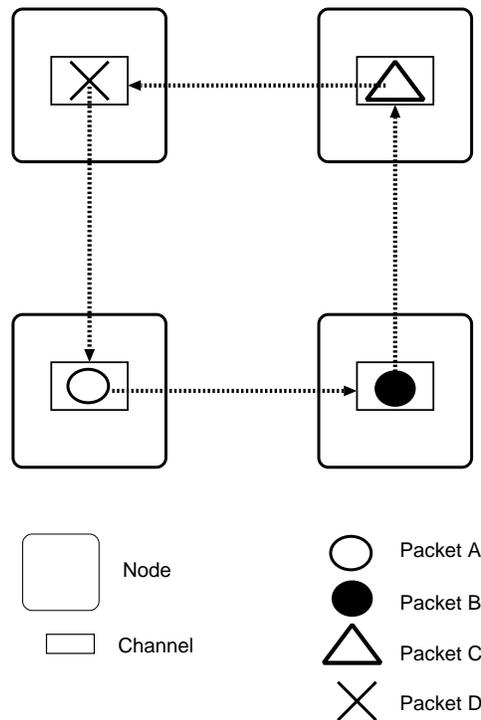


図 2.11: デッドロックの例

そのため混雑や故障などにも対処することが可能である。また、空いている経路を有効に用いることでネットワークの性能を十分に発揮できる可能性がある。したがって、性能面のみを考慮すれば固定型より適応型のほうが有利であると言える。しかしながら、デッドロックフリーの保証が容易でなかったり、アルゴリズムが複雑になるなどの問題がある。多くの場合、仮想チャネルを必要とするためハードウェア量が固定型に比べ増大する可能性もある。

本論文ではデッドロックフリーなルーティングアルゴリズムのみを議論の対象とする。そのため、固定型ルーティングとデッドロックフリールーティングは本質的には同義ではないが、固定型を用いる理由の大半がデッドロックの回避にあるので、固定型のデッドロックフリールーティングを単にデッドロックフリー・ルーティングと呼ぶ。また、適応型のデッドロックフリールーティングはデッドロックフリーを保証した上で適応性を得ることが目的であるので、単に適応型ルーティングと呼ぶ。

2.3.5 デッドロックフリー・ルーティング

パケットがお互いに循環的にチャネルを要求し、全体として全く通信が行なえなくなる状態、デッドロックを固定型ルーティングにより回避する手法について議論する。

デッドロックの原因は循環的なチャネル要求だけである。そのため、デッドロック

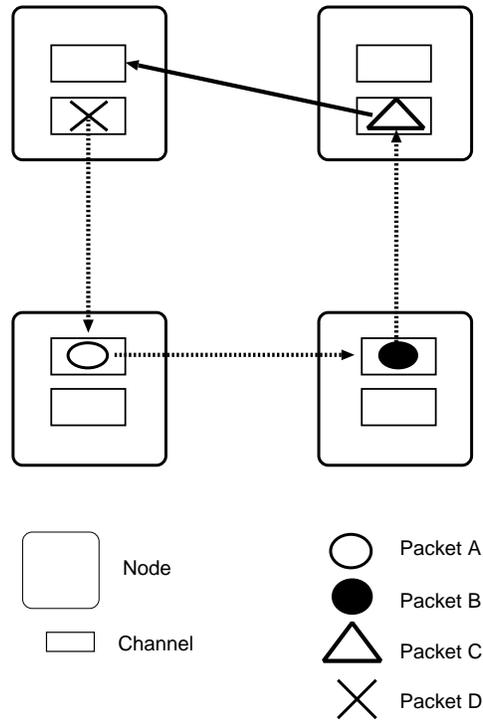


図 2.12: 仮想チャンネルによるデッドロックの回避

はチャンネル要求に循環が生じないようにすることで回避することができる。

デッドロックフリーか否かは、チャンネルに番号を割り当てたとき、ルーティングにより得られるチャンネル番号を順番に並べた時、そのリストが昇順あるいは降順になるか否かで判別できる [9]。チャンネル番号のリストが昇順（降順）でなる時は必ず終点が存在し、しかもパケットは終点の方向へのみ進むことになる。したがって、循環が生じないのでデッドロックフリーであるといえる。

Dimension order routing [12] はデッドロックを回避する手法として幅広く用いられる手法で、メッシュ網やハイパーキューブ網などの k -ary n -cube、CCC、De Bruijn 網等の多くの結合網において有効である。次元オーダールーティングではルーティングをネットワークの次元の低い（高い）ほうから順に行ないデッドロックを回避する。なお、パケットは後戻りはできないものとする。

二次元メッシュにおける Dimension order routing を例に説明する。パケットははじめに x 方向に x 方向のアドレスが一致するまで送られ、その後に y 方向に転送することで Dimension order routing 実現できる。図 2.13 に二次元メッシュでの Dimension order routing の例を示す。

定理 1 k -ary n -cube における Dimension order routing はデッドロックフリーである。

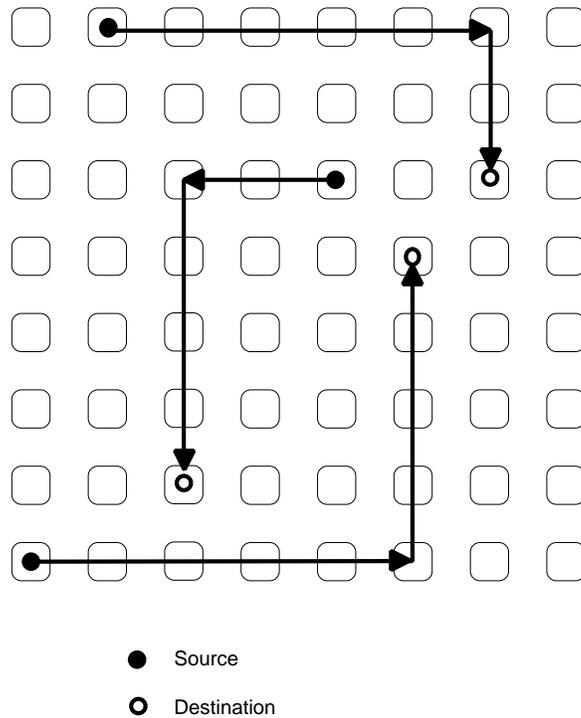


図 2.13: 二次元メッシュ上での Dimension order routing

証明 k -ary n -cube における任意のチャンネル番号 X を dn と置く．ここで d はチャンネルが接続されている次元， n はチャンネルがノード番号の大きくなる方向の時はノード番号，小さくなる時は方向性を考慮するため各次元のサイズに対する補数をとる．このとき，パケットは後戻りできないので， n は一方向にのみ進むことになり，また，チャンネル番号はパケットの進む方向にその番号が大きくなるので n は昇順である．

ルーティングは次元の低い方から行なわれるので d もまた昇順である．したがって，チャンネル番号は昇順で要求される．つまり， k -ary n -cube 上での Dimension order routing デッドロックフリーである．

なお，次元の高い方からルーティングが行なわれる場合も同様にして示すことができる．

また，各次元にループ（ラウンドトリップループ）がある場合は， k -ary n -cube における任意のチャンネル番号 X を dvn と置く．ここで v は仮想チャンネルのクラス番号である．

いま，各リンクの仮想チャンネルを $class0$ ， $class1$ の 2 つに分ける．各次元で最初は $class0$ の仮想チャンネルを使用しルーティングを開始する．パケットがラウンドトリップループを通過する際，仮想チャンネルのクラスを $class1$ に切替える．このように仮想チャンネルを使用することでラウンドトリップループを通過した時を除けば n は常に昇順となる．ラウンドトリップループを通過した時は v が 0 から 1 になるため全体として一様昇順が保存さ

れる。

□

2.3.6 適応型ルーティング

固定型ルーティングではルーティングの経路が固定のため，混雑があっても他の経路を選択することができない．また，経路上に故障リンクや故障ノードがあった場合はメッセージを目的ノードまで転送することさえできなくなるといった問題がある．そのためこれまでに多くの適応型ルーティングが提案されてきている．

適応型ルーティングは大きく最短型適応ルーティングと迂回型適応ルーティングに分けられる．以下に各々の代表的な手法である，Duatoの手法，Turn ModelとDimension reversal routingについて述べる．

2.3.6.1 Duatoの手法

Duato [13, 14]の手法は既存のデッドロックフリー・ルーティングに仮想チャンネルを付加する方法である．仮想チャンネルはデッドロックフリー・ルーティングに必要な部分(以下 *core* チャンネル)とそれ以外の部分(以下 *free* チャンネル)に分けられる．Duatoの手法では通信中のメッセージは任意のノードにおいて *core* チャンネルと最短経路上にある *free* チャンネルのどちらも選択できる．ただし *free* チャンネルはバッファに空きがある時のみ選択することができる．*core* チャンネルは常に選択対象として含まれ，更にメッセージは最短経路上に存在してるのでシステム全体でデッドロックフリーを保証することができる．同時に，*free* チャンネルはバッファが空いていればいつでも選択が可能なので適応型ルーティングが実現されている．

2.3.6.2 Turn Model

Turn ModelはGlassら[15]により提案された手法で，原則的には仮想チャンネルを付加することなく迂回ルーティングを実現できる．通常，デッドロックフリーを実現するためにルーティングの方向に制限を加える．つまりターンのパターンをいくつか制限することでデッドロックフリーを実現している．例えば，二次元メッシュのデッドロックフリールーティングでは x 方向， y 方向の順でルーティングが行なわれる．つまり x 方向から y 方向へのターンは可能であるが，その逆である y 方向から x 方向へのターンは禁止されている．

Turn Modelではこのようなターンの制限を最小限にすることでこれまで禁止されていたターンを可能にすることができる．禁止するターンのパターンの組み合わせによりいくつかのルーティングアルゴリズムが考えられるが，Glassら[15]では，二次元メッシュ網に対し，West-first，North-last，Negative-firstといったアルゴリズムが提案されてい

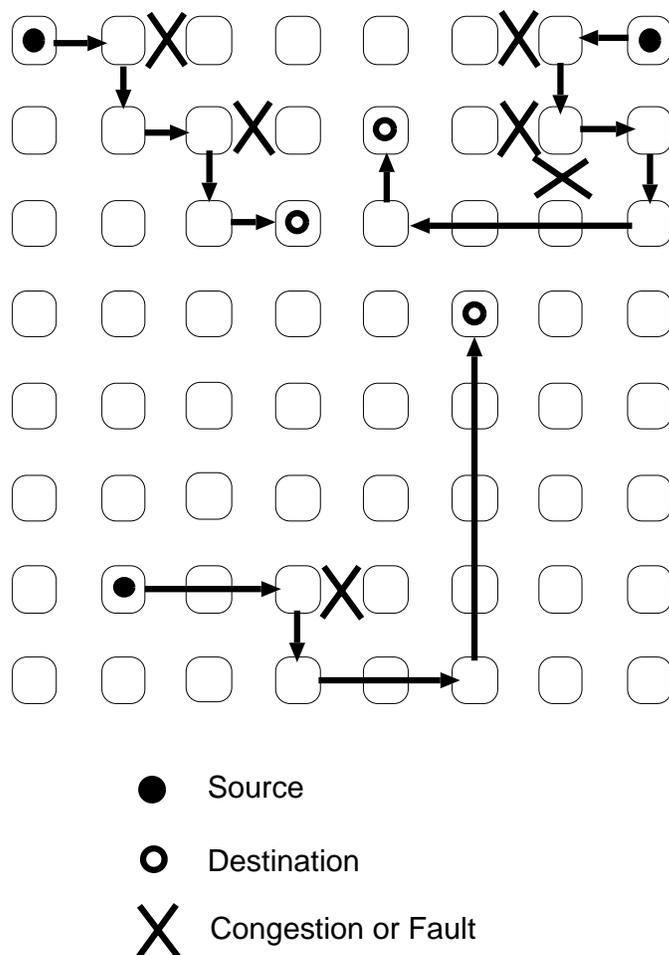


図 2.14: 2D-MESH 上での North-last 法

る．図 2.14 に二次元メッシュ網での North-last 法の例を示す．図 2.14 では，パケットは混雑や故障を避けて通信されていることが分かる．

2.3.6.3 Dimension reversal routing

Dimension reversal routing は Dally ら [10] により提案された手法である．この手法では，通常は Dimension order routing が行なわれるが，混雑や故障があった場合はそれに従わない方向へルーティングすることができる．Dimension reversal routing には静的ルーティングと動的ルーティングとの 2 種類がある．まず，静的ルーティングについて説明する．いま，仮想チャネルが r 本あるとし，パケットが i 番目（クラス i ）の仮想チャネルにあるとする．Dimension order routing による次の転送先に混雑等があった場合は，そのクラスに 1 を加えた， $i + 1$ 番目（クラス $i + 1$ ）の仮想チャネルに対し次元オーダを守ることなく転送することができる．そのため，次元オーダに従わないルーティングを行なう

たびに仮想チャネルのクラスが大きくなるためデッドロックが生じない。ただし，チャネルのクラスが $r - 1$ となった場合はそれ以上大きいクラスに転送することができないため Dimension order routing を行なわなければならない。

次に動的ルーティングについて説明する。動的ルーティングでは，チャネルは固定ルーティング用と適応ルーティング用の 2 つの仮想クラスに分割され，適応ルーティング用の仮想チャネルは更にその中でいくつかのクラスに分割される。適応ルーティング用の仮想チャネルは，仮想チャネルがフリーであればどんな方向へのルーティングも許される。ただし，このチャネルが空いていない場合はそれを待つことはできず，固定ルーティング用のチャネルを用いて Dimension order routing を行なうか，適応ルーティング用のチャネルのうち，自分のクラスよりも大きいクラスのチャネルへ転送しなければならない。このように，Dimension reversal routing では仮想チャネルを利用したデッドロックフリーな迂回型適応ルーティングが可能となるが，仮想チャネルを多く必要するといった問題もある。

図 2.15 に二次元メッシュ網での静的ルーティングの例を示す。図 2.15 で class は仮想チャネルのクラスであり，仮想チャネル数は 4 とする。どのパケットも混雑や故障があった場合は次元オーダを守ることなく Dimension reversal routing を行なうことができる。しかし，仮想チャネルのクラスが最大の 3 になった場合は Dimension order routing を行なうことになる。

2.4 まとめ

超並列計算機の代表的な結合網であるメッシュ結合網，トーラス結合網に加え，再帰的相互結合網の代表例として SRT について説明した。さらに，ノード間通信について議論を行ない，代表的なデッドロック回避の手法である Dimension order routing と適応型ルーティングについて説明した。

様々な適応型ルーティングが提案されているがどれも再帰的相互結合網の特長であるバイパスリンクの使用までは考慮されていない。本論文では 3 章，4 章で再帰的相互結合網の特性を考慮した適応型ルーティング手法を提案し，その性能について詳しく議論する。

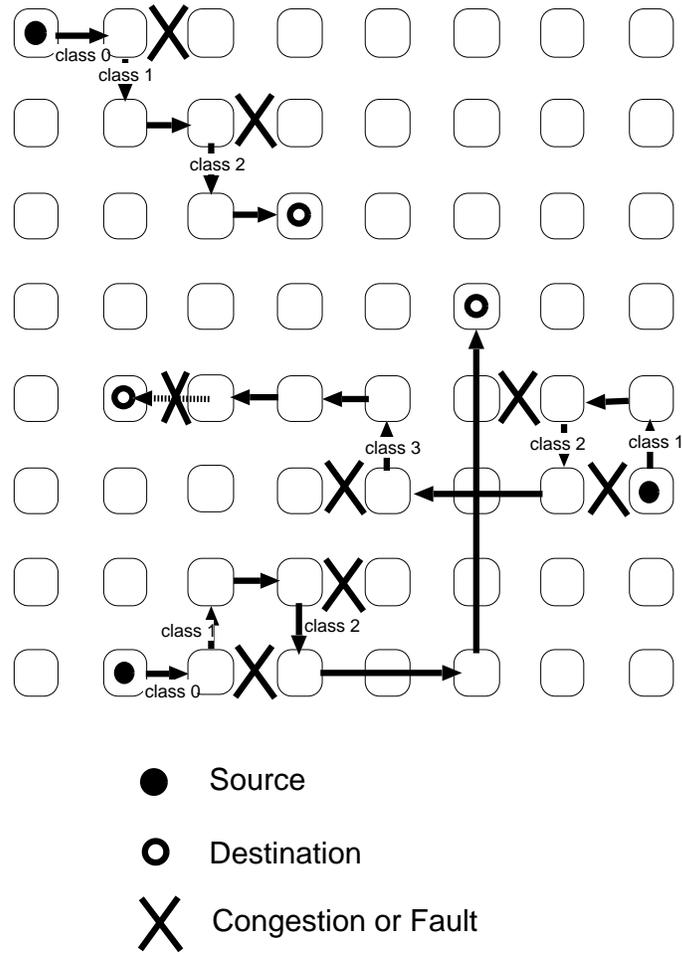


図 2.15: 2D-MESH 上での静的な Dimension reversal routing

第 3 章

一次元再帰網における適応型ルーティング

3.1 はじめに

一次元の再帰的結合網はリニア網あるいはリング網上に配置されたノードに対して、ある条件に従いながら再帰的にバイパスリンクを付加した結合網である。

一次元再帰網では、バイパスリンクにより直径を短縮できるという構造的な特長を生かすためにできる限り付加されたバイパスリンクを利用する方針でルーティングが行なわれる。一次元再帰網におけるルーティングでは、ある手続きを再帰的に繰り返して目的ノードまでの経路を算出する。各結合網により、このようなルーティング手法の呼び方は異なるが、本論文では SRT の呼び方を用い、再帰ルーティングと呼ぶ。

一次元再帰網では再帰ルーティングを行なうことにより、最適ではないものの、実用上十分な性能を得ることができる。たとえば、一次元 SRT (1D-SRT) でノード数が 1024 ノードの時、ネットワークを全探索し最短経路を得る最適ルーティングと再帰ルーティングでは、直径は等しく、平均距離では約 10 % 再帰ルーティングのほうが増加する程度である。しかしながら、従来法ではデッドロックフリーの保証までは考慮されていない。また、多くのもの再帰ルーティングは固定型ルーティングであるため、適応性や耐故障性といった側面からも問題がある。

本章では、一次元再帰網の基本構成がリニア結合あるいはリング結合であることを利用し、 k -ary n -cube に対する次元オーダルーティングによりデッドロックフリーが実現できることを示す。さらに、再帰網の特徴であるバイパスリンクの使用を考慮した適応型ルーティングを提案し、その動的性能特性をシミュレーションにより評価する。

3.2 諸定義

本節では本論文で必要な定義を行なう。

定義 4 (チャンネルの方向) 任意のチャンネルの方向はノード番号の大なる方向を正とし, その逆を負とする. また, 始点ノードから目的ノードまでのホップ数が小さい方向を順方向と呼び, 大きい方向を逆方向と呼ぶ. \square

定義 5 (ルーティング手法の呼び方) パケットがある次元で一方向のみ (monotonic order [12]) に転送されるルーティングを *Monotonic order routing* と呼び, *Monotonic order routing* を次元順で用いたルーティングを *Dimension order routing* と呼ぶ. また, 次元順を守らないルーティングを *Dimension reversal routing*[10] と呼ぶ.

定義 6 (チャンネル番号の大小関係) あるチャンネル番号を n 次元ベクトル $C = (c_{n-1}, c_{n-2}, \dots, c_1, c_0)$, $c \in Z$, で表した時, 任意の 2 つのチャンネル番号 $C_1 = (c_{1_{n-1}}, c_{1_{n-2}}, \dots, c_{1_1}, c_{1_0})$, $C_2 = (c_{2_{n-1}}, c_{2_{n-2}}, \dots, c_{2_1}, c_{2_0})$ には大小関係が定義され, 次の条件を満たす時 C_1 が C_2 より大きいものとし, $C_1 > C_2$ と書く.

(条件)

$$\exists i, \nexists j$$

$$s.t. \quad c_{1_i} - c_{2_i} > 0 \quad \text{and} \quad c_{1_j} - c_{2_j} < 0 \quad \text{and} \quad j > i \quad (3.1)$$

\square

定義 7 (記号に関する定義) 本論文で用いる記号を以下に示す.

表 3.1: 記号に関する定義

パケットの存在するノード	n_{cur}
パケットの目的ノード	n_{dst}
パケットの次の転送先	n_{next}
パケットが選択しているチャンネルのレベル	l_{cur}
パケットが次に選択するチャンネルのレベル	l_{next}
レベル l_* リンクの基本結合上でのホップ数	d_{l_*}

3.3 デッドロックフリールーティング

一次元再帰網はリニア網あるいはリング網に対して再帰的にバイパスリンクを付加した結合網である. リニア網, リング網は共に k -ary n -cube のクラスに属し, 具体的には $n = 1$ とした場合と等価である. したがって, k -ary n -cube 上でデッドロックフリーが保証され

ている Monotonic order routing を施すことで次元再帰網においてもデッドロックフリーを保証することができる。本節では、次元再帰網を一般的に定義し、Monotonic order routing によりデッドロックの回避が可能であることを示す。さらに、1D-SRT に対し具体的なルーティング方法について詳細に述べる。

3.3.1 定義

任意の次元再帰網のトポロジ、ノード、チャンネルを次のように定義する。

定義 8 (次元再帰網) N ノードから成る次元再帰網は全てのノードから構成される基本結合と一部のノードから構成される基本結合と同じ結合を持つ上位結合から構成される。基本結合はリニア結合あるいはリング結合とする。また、各ノードにはあるノードを基準にした一連のノード番号は割り当てられる。 □

定義 9 (レベル) 各ノードのリンクは接続されているノードまでの基本結合網上でホップ数により区別され、その区別をレベルと呼ぶ。基本結合をレベル 0 とし、バイパスリンクは接続されているノードまでの基本結合網上でホップ数に応じたレベル $l (l > 0)$ を割り当てられる。 □

定義 10 (チャンネル番号) 各ノードのチャンネルに対し、次のようなチャンネル番号 (v, n, l) を割り当てる。

- v : 仮想チャンネル番号
- n :
 - チャンネルの方向が正: ノード番号 n
 - チャンネルの方向が負: サイズに対する補数 $(N - 1 - n)$
- l : チャンネルのレベル

□

3.3.2 次元再帰網のデッドロックフリー・ルーティング

次元再帰網では、Monotonic order routing を適用することでデッドロックフリーを実現できる。ここで Monotonic order routing を適用するとは、

- (i) 転送方向の方向性、
 - (ii) ラウンドトリップループによるチャンネル循環の回避、
- を保証することである。

定理 2 (Monotonic order routing によるデッドロックの回避) 次元再帰網路上での *Monotonic order routing* はデッドロックフリーである。ただし、ラウンドトリップループのある再帰網では仮想チャンネルを必要とする。その場合、仮想チャンネルはラウンドトリップループを通過する際に仮想チャンネルを番号の高い方に切替える。

証明 Monotonic order routing ではメッセージは一方向にのみ進む。したがって、定義 10 より、チャンネル番号は単調増加するのみである。

ラウンドトリップループがある場合、ラウンドトリップループを通過する際に仮想チャンネルの番号が高い方に切り替わるので、その場合もチャンネル番号は単調増加である。

以上より、次元再帰網における Monotonic order routing デッドロックフリーである。

□

以上より次元再帰網路上での Monotonic order routing はデッドロックフリーであることが分かった。つまり、メッセージの転送される方向が一方向でさえあれば、途中、どんなレベルのチャンネルを使用してもデッドロックフリーは保証される。

3.3.3 仮想チャンネルの使用法

従来、トラス網のようにラウンドトリップループが存在するネットワークでは仮想チャンネルを用いる。ラウンドトリップループを通過した際に仮想チャンネルを切替えることでデッドロックフリーを実現してきた。従来の仮想チャンネルの使用法は次の通りである。

1. 仮想チャンネルに class 1, class 0 といった 2 つのクラスを割り当てる。
2. 最初は class 0 の仮想チャンネルを使用しルーティングを開始する。
3. ラウンドトリップループを用いる際、仮想チャンネルを class 1 に切替える。

従来の仮想チャンネルの使用法では、パケットがネットワーク内へ投入される時使用されるチャンネルが常にクラス class 0 に決められている。そのため、チャンネルの使用頻度に偏りが生じてしまい資源が有効に利用されないという問題がある。

そこで、パケットがネットワーク内に投入される時、どちらかの空いている方の仮想チャンネルを選択できる手法をとる。なお、ラウンドトリップループを使用する場合の仮想チャンネルの使用法は従来通りとする。提案する仮想チャンネルの使用法はデッドロックフリーである。Monotonic order routing はラウンドトリップループを通過しない限り、仮想チャンネルのクラスは固定である。したがって、パケットの投入の際に 2 つあるクラスのどちらを選択してもチャンネル番号は一様に昇順となり、デッドロックフリーは保証される。

任意のパケットがラウンドトリップループを通過するか否かの判断であるが、Monotonic order routing ではパケットは順方向しか転送が許されていないので、始点ノードと目的ノードとの差をとることで容易に判断できる。

```

RecursiveRouting( $x_s, x_d$ ) {
  if(  $x_s = x_d$  )      return(  $\phi$  );
  if(  $x_s \leq x_d$  )    dir = +1;
  else                  dir = -1;
   $l_r = FindMedLevel(x_s, x_d)$ ;
  ( $x_s^r, x_d^r$ ) = FindNearestNodes( $l_r, x_s, x_d$ );
   $l_u = FindUpLevel(x_s, x_d)$ ;
  ( $x_s^u, x_d^u$ ) = FindNearestNodes( $l_u, x_s, x_d$ );
  if(  $|x_s - x_s^u| + |x_d - x_d^u| < |x_s - x_s^r| + |x_d - x_d^r|$  ) {
     $l_r = l_u$ ;
     $x_s^r = x_s^u$ ;     $x_d^r = x_d^u$ ;
  }
  RoutingList = RecursiveRouting( $x_s, x_s^r$ );
  while(  $x_s^r \neq x_d^r$  ) {
    addlist(RoutingList,  $x_s^r$ );
     $x_s^r = x_s^r + dir * 2^{l_r}$ ;
  }
  addlist(RoutingList, RecursiveRouting( $x_d^r, x_d$ ));
  return( RoutingList );
}

```

図 3.1: 1D-SRT における再帰ルーティング

3.3.4 一次元 SRT におけるデッド ロックフリー・ルーティング

3.3.3 節では一次元再帰網において次元オーダールーティングを用いることでデッドロックを回避できることを示した。

本節では、まず、1D-SRT における準最適なルーティング手法である再帰ルーティングについて説明し、再帰ルーティングがデッドロックを引き起こす可能性があることを示す。次に、1D-SRT に対し Monotonic order routing を適用することでデッドロックフリーを実現できることを示す。

1D-SRT における再帰ルーティングを図 3.1 に示す。1D-SRT における再帰ルーティングは始点ノードを x_s 、終点ノードを x_d としたとき、関数 $FindMLevel$ によって、ルーティングに使用するリンクの最大レベル l_r を求め、関数 $FindNearestNodes$ で、レベル l_r ノード x_s^r, x_d^r を探す。始点ノード x_s と x_s^r, x_d^r と終点ノード x_d との間で、手続きを再帰的に呼び出し、経路を求める。

32 ノードから成る 1D-SRT における、ノード 0 ($x_s = 0$) からノード 15 ($x_d = 15$) まで

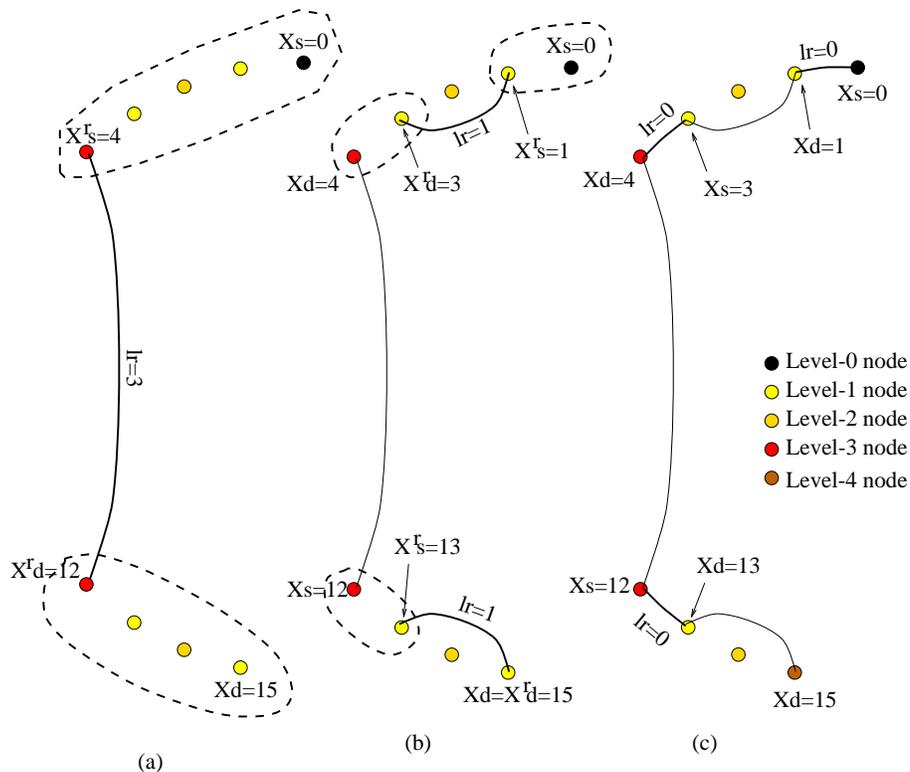


図 3.2: 1D-SRT の再帰ルーティングの例

の再帰ルーティングの例を図 3.2 に示す。

1D-SRT における再帰ルーティングでは上位レベルノードを算出する際より、 x_s に最も近いノードを探索するため、逆方向へルーティングする可能性がありチャンネルに循環を生じさせてしまう原因となる。また、再帰ルーティングを行なうために必要な仮想チャンネル数が明確にされていないため、ラウンドトリップループを用いると、チャンネルに循環が生じてしまい、デッドロックを引き起こす可能性がある。

1D-SRT における再帰ルーティングでは以上のような問題点があるが、定理 2 より、一次元再帰網上で Monotonic order routing はデッドロックフリーなので、1D-SRT に対しても、Monotonic order routing に従うよう再帰ルーティングに条件を付加することでデッドロックフリーが実現できる。付加する条件は次に示すような条件である。

(C1) ノード探索を一方向にのみ行なう

(C2) 物理リンク 1 本につき 2 つの仮想チャンネルを設ける

条件 (C1) により転送方向が一方向となることが保証され、条件 (C2) によりラウンドトリップループを用いた際のチャンネルの循環が回避される。従って、条件 (C1)、(C2) を付加することで再帰ルーティングは Monotonic order routing そのものとなる。

以上のことは次の系としてまとめられる。

系 1 $1D$ -SRTにおける再帰ルーティングに $(C1)$ 、 $(C2)$ の条件を付加したアルゴリズムはデッドロックフリーである。

3.4 デッドロックフリー・適応型ルーティング

3.3 節では、再帰ルーティングと Monotonic order routing を用いて一次元再帰網でデッドロックフリーを保証する手法を述べた。しかしながら、3.3 節で述べたデッドロックフリールーティングは固定型であり、2 章等でも述べたように適応性や耐故障性がない。

再帰網におけるデッドロックフリー・ルーティングは、従来からの代表的な手法である Dimension order routing を適用することで簡単に得ることができた。しかし、一次元再帰網における適応型ルーティングを考える場合、従来の代表的な適応化手法である Duato の手法や Turn Model を適用するにはいくつかの問題がある。

まず、Duato の手法であるが、一次元の場合単純に仮想チャンネルを付加したのと変わらないという問題がある。Duato の手法の利点は、付加する仮想チャンネルにより、いくつか存在する最短経路を選択できるという点にある。したがって、一次元の場合経路が固定となってしまう、どのチャンネルを選択するかといった面では適応性が得られるが、経路選択の自由度といった観点からは適応性が得られない。

次に Turn Model であるが、モデルを作成する際のいくつかの禁止事項に 180 度のターンの禁止がある。方向転換が 180 度の方向の一つしかない一次元結合網ではモデルを作成することは実質的に不可能である。

そこで Turn Model を用いずに、一次元再帰網でターンを可能にする手法を提案する。また、提案する手法の動的な性能評価をシミュレーションにより行ない、その性能について議論する。

3.4.1 一次元再帰網の適応型ルーティングの導出

Turn Model では結合網の循環構造を明確にし、方向転換をいくつか禁止することにより循環を断ち切りデッドロックフリーを保証した。しかし、Turn Model では 180 度のターンが禁止されており、一次元網では Turn Model を適用することは不可能である。そこで、180 度のターンが可能な領域（ノードの組み合わせ）を導き出し、その領域内ではチャンネル番号が常に昇順となることを示す。それを用いて Turn Model では禁止されていた 180 度のターンが可能なルーティングアルゴリズムを提案する。

Turn Model に限らず、従来のルーティングアルゴリズムは、アルゴリズムの提案が先に行なわれ、デッドロックフリーは、そのあとでチャンネルに適切な番号を与え、循環が生じないことを示すことで保証してきた。

ここで提案する手法はこれまでの手法とは全く逆のアプローチをとる。つまり、各チャ

ネルに対してはじめにチャンネル番号を割り当てる．そしてその範囲内で可能なルーティングを順次明確にしていく．

提案手法では先にチャンネル番号を与える必要があるが、ここでは次元再帰網の任意のチャンネルに対し定義 10 に従い、チャンネル番号を割り当てることとする．

次に 180 度のターンが可能な領域を示す．この領域では、パケットは目的ノードをバイパスリンクを使用して飛び越え、そののちに目的ノードへ引き返すことが可能である．つまり、この領域では行き過ぎと逆方向へのルーティングが可能となる．また、このような手法を同次元迂回ルーティングと呼ぶ．

定理 3 (同次元迂回ルーティング) 定義 10 に従いチャンネル番号を割り当てた次元再帰網で同次元迂回ルーティングが可能な必要十分条件は、パケットの存在するカレントノード n_{cur} と目的ノード n_{dst} が次の条件を満たすことである．

(条件)

$$n_{cur} < n_{dst}, \quad n_{cur} < \frac{N-1}{2} - \frac{d_{l_{cur}}}{2}, \quad (3.2)$$

あるいは、

$$n_{cur} > n_{dst}, \quad n_{cur} > \frac{N-1}{2} + \frac{d_{l_{cur}}}{2}. \quad (3.3)$$

ここで、 l_{cur} はカレントノードで選択されているチャンネルのレベルであり、 $d_{l_{cur}}$ はレベル l_{cur} リンクの基本結合上でのホップ数である．

証明

十分性の証明

十分性を示す為に条件式 (3.2), (3.3) が満足されていると仮定する．

(i) $n_{cur} < n_{dst}$ のとき

仮定より条件式 (3.2) は、 $n_{next} = n_{cur} + d_{l_{cur}}$ と書けるので、

$$\begin{aligned} n_{cur} &< \frac{N-1}{2} - \frac{d_{l_{cur}}}{2}, \\ 2n_{cur} &< N-1 - d_{l_{cur}}, \\ n_{cur} &< N-1 - n_{cur} - d_{l_{cur}}, \\ n_{cur} &< N-1 - n_{next}, \end{aligned} \quad (3.4)$$

となる．したがって、式 (3.4) より

$$(v, n_{cur}, l_{cur}) < (v, N-1 - n_{next}, l_{next}), \quad (3.5)$$

は常に成り立つ．したがって、条件を満たす負方向のチャンネル $(v, N-1 - n_{next}, l_{next})$ への転送は常にデッドロックが保証される．

(ii) $n_{cur} > n_{dst}$ のとき

(i) 同様，条件式 (3.3) は， $n_{next} = n_{cur} - d_{l_{cur}}$ と書けるので，

$$\begin{aligned}
n_{cur} &> \frac{N-1}{2} + \frac{d_{l_{cur}}}{2}, \\
2n_{cur} &> N-1 + d_{l_{cur}}, \\
n_{cur} &> N-1 - n_{cur} + d_{l_{cur}}, \\
N-1 - n_{cur} &< n_{cur} - d_{l_{cur}}, \\
N-1 - n_{cur} &< n_{next},
\end{aligned} \tag{3.6}$$

となる．したがって，式 (3.6) より，条件が満たされている時，負方向のチャンネル $(v, N-1 - n_{cur}, l_{cur})$ から正方向のチャンネル (v, n_{next}, l_{next}) への転送はデッドロックフリーである．

必要性の証明

必要性を示す為に，同次元迂回ルーティングが可能となる条件を実際に導出する．

(i) $n_{cur} < n_{dst}$ のとき

このとき，パケットは正の方向に進んでいる．そのパケットがあるノードで一旦，バイパスリンクにより目的ノードを飛び越し，その後，目的ノード側へ進む，つまり後戻りする状況を考えて，パケットの現在のノード n_{cur} と目的ノード n_{dst} ，そしてパケットの次の転送先 n_{next} の間には次の関係が成り立つ．

$$n_{cur} < n_{dst} < n_{next}.$$

また，パケットの通過するチャンネル番号は順次

$$\begin{aligned}
(v, n_{cur}, l_{cur}) &\longrightarrow (v, N-1 - n_{next}, l_{next}), \\
&\longrightarrow (v, N-1 - (n_{next} - d_{l_{next}}), l_{next}).
\end{aligned}$$

と書ける．

したがって，デッドロックフリーであることを示すためには上記のチャンネル番号に次のような大小関係が成り立てば良い．

$$(v, n_{cur}, l_{cur}) < (v, N-1 - n_{next}, l_{next}), \tag{3.7}$$

$$< (v, N-1 - (n_{next} - d_{l_{next}}), l_{next}). \tag{3.8}$$

式 (3.8) の大小関係が真であるのは自明であるので，ここでは式 (3.7) の大小関係が真となる条件を導く．式 (3.7) が真であるための条件は定義 6 より

$$n_{cur} < N - 1 - n_{next}.$$

となり， $n_{next} = n_{cur} + d_{l_{cur}}$ として，この不等式を解くと，

$$n_{cur} < \frac{N - 1}{2} - \frac{d_{l_{cur}}}{2}.$$

が成り立つ．

(ii) $n_{cur} > n_{dst}$ のとき

この場合も，(i) の場合と同様な議論ができ，カレントノード n_{cur} と目的ノード n_{dst} ，パケットの次の転送先 n_{next} の間には次の関係が成り立つ．

$$n_{cur} > n_{dst} > n_{next}.$$

また，パケットの通過するチャンネル番号は順次

$$\begin{aligned} (v, N - 1 - n_{cur}, l_{cur}) &\longrightarrow (v, n_{next}, l_{next}), \\ &\longrightarrow (v, (n_{next} + d_{l_{next}}), l_{next}). \end{aligned}$$

と書ける．したがって，デッドロックフリーであるための条件は

$$(v, n_{cur}, l_{cur}) < (v, N - 1 - n_{next}, l_{next}), \quad (3.9)$$

$$< (v, N - 1 - (n_{next} - d_{l_{next}}), l_{next}). \quad (3.10)$$

となる．式 (3.10) の大小関係が真であるのは自明であるので，先ほどと同様，式 (3.9) の大小関係が真となる条件を導くと，

$$N - 1 - n_{cur} < n_{next}.$$

となり， $n_{next} = n_{cur} - d_{l_{cur}}$ として，この不等式を解くと，

$$n_{cur} > \frac{N - 1}{2} + \frac{d_{l_{cur}}}{2}.$$

が成り立つ．

以上より，実際に同次元迂回ルーティングが可能となる条件が導出でき，それらは式 (3.2)，(3.3) と一致する． □

```

AdaptiveRouting( $n_{cur}, n_{dst}$ ){
    if(  $n_{cur} < n_{dst}$  ) {  $dir = +1$  }
    else{  $dir = -1$  }
    NextNode = RecursiveRouting( $n_{cur}, n_{dst}$ )
    if( NextNode == BUSY && Cond( $n_{cur}, n_{dst}$ ) == TRUE
        &&  $|n_{dst} - n_{src}| > \frac{d_{l_{cur}}}{2}$  ){
        NextNode =  $n_{cur} + dir \times d_{l_{cur}}$ 
    }
}

```

図 3.3: 一次元再帰網の適応型ルーティング

3.4.2 一次元再帰網に対する適応型ルーティング

定理 2、3 を用いると、Monotonic order routing による通常の再帰ルーティングと同次元迂回ルーティングを許容するルーティングとを選択的に使用する適応ルーティングが可能である。

一次元再帰網における適応型ルーティングは図 3.3 に示すアルゴリズムとしてまとめることができる。

図 3.3 において $RecursiveRouting(n_{cur}, n_{dst})$ は結合網毎に用意されるデッドロックフリーな再帰ルーティングである。デッドロックフリーが保証されていない場合は定理 2 を使いデッドロックを回避できる。

$Cond(n_{cur}, n_{dst})$ は n_{cur}, n_{dst} が式 (3.2)、(3.3) を満たすか否かを判別する関数である。これが満たされたときは上位レベルリンクを使用して $d_{l_{cur}}$ 離れたノードに転送される。また、 $|n_{dst} - n_{src}| > \frac{d_{l_{cur}}}{2}$ は行き過ぎによる目的ノードまでの距離が増大してしまうのを防ぐ為の条件である。ただし、どの程度まで増大を許容するかは任意であり、条件は加えられなくても良い。

3.4.3 一次元 SRT における適応型ルーティング

3.4.2 節では一般化した一次元再帰網に対する適応型ルーティングを提案した。

本節では一次元 SRT(1D-SRT) に対して、提案手法を適用し、同次元迂回ルーティングが可能となる条件をまとめる。

まず、定義 10 によるチャンネル番号の割り当てを 1D-SRT に対して行なう。1D-SRT の各ノードは基本レベルリンクとただ 1 つの上位レベルリンクを持つことから、1D-SRT の

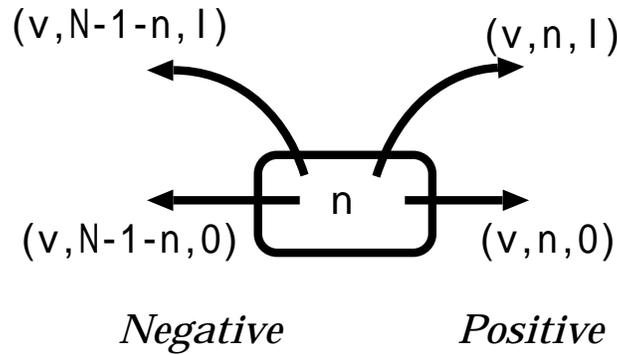


図 3.4: 1D-SRT のチャンネル番号の割り当て

各ノードのチャンネルは図 3.4 のようなチャンネル番号を割り当てる。

次に, 3.2 節より Monotonic order routing による再帰ルーティングのデッドロックフリーの保証は明らかなので, 定理 3 による適応化について述べる。

定理 3 より, 1D-SRT における同次元迂回ルーティング可能な領域は次のような系としてまとめられる。

系 2 1D-SRT において, カレント n_c とデスティネーション n_d が次の条件を満たす時, 同次元迂回ルーティングが可能である。

(条件)

$$n_c < n_d, \quad n_c < \frac{2^{l_{max}} - 1}{2} - 2^{l-1} \quad (3.11)$$

あるいは,

$$n_c > n_d, \quad n_c > \frac{2^{l_{max}} - 1}{2} + 2^{l-1} \quad (3.12)$$

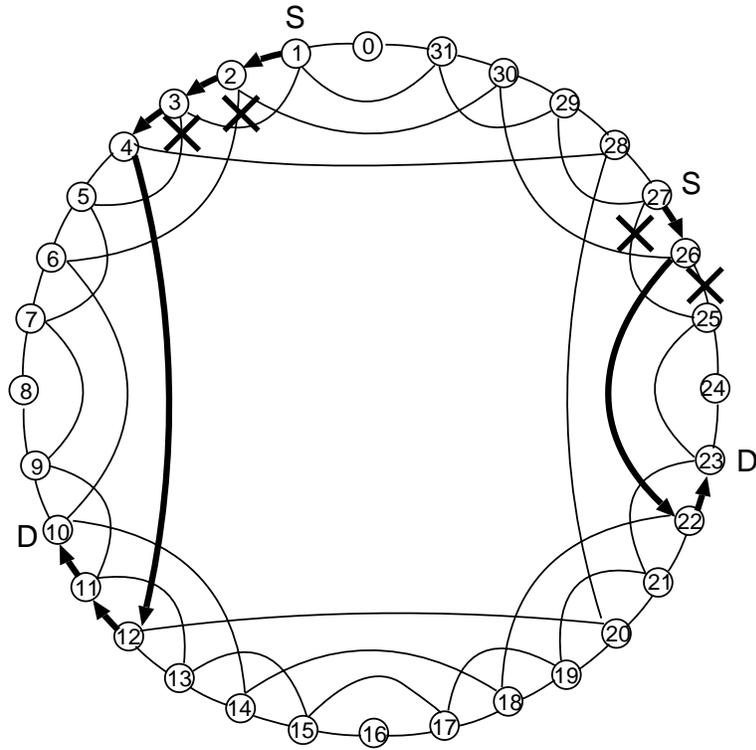
ここで $l_{max} = \log_2 N$ である。

図 3.5 に 1D-SRT 上でのルーティング例を示す。ブロックされたメッセージは迂回可能となっているのが分かる。

3.5 シミュレーション評価

3.5.1 シミュレータ概要

デッドロックフリーなルーティングアルゴリズムの性能を評価するために, C++によりシミュレータを作成し, 性能評価を行なった。シミュレータはフリットレベルでのシミュレートが可能であり, トポロジ, ルーティングアルゴリズム, パケットサイズ, メッセージの発生確率などが変更可能である。



S Source
D Destination

図 3.5: 1D-SRT 上での同次元迂回ルーティングの例

3.5.2 シミュレーション概要

一次元再帰網として 1D-SRT を採用する．シミュレーションの条件を表 4.1 に示す．ここで，SRT 以外の仮想チャネル数はデッドロックフリーを保證する最小数とした．

シミュレーションは，Short-Span 型の SRT に対して行なう．これは，基本型や Long-Span 型では最大レベルリンクの基本トラス上でのホップ数が $N/2$ であり，ルーティングが一方向に制限されていることから，ノード間通信の大半でそのリンクを使用できないからである．図 3.6 に 32 ノードから成る Short-Span 型 1D-SRT を示す．

また，動的性能評価に必要な平均通信時間，パケット発生確率は以下のように定義する．

平均通信時間

あるノードがパケットを最初のフリット（ヘッダ）をネットワーク内に投入した時間と目的ノードでそのパケットが全て受信された時間の差を平均通信時間と呼び，ルーティング手法の性能を測る指標とする．

表 3.2: シミュレーションの概要 (1D-SRT での性能評価)

トポロジ	Short-Spna 型 1D-SRT ,MESH, TESH
サイズ	256
パケット長	16flit
仮想チャネル数	2,3,4 (SRT) 1 (MESH) 4 (TESH)
転送パターン	ランダム転送
パケット発生確率	$10^{-4} \leq P \leq 1$ (flit/clock) (36ポイントで測定)
シミュレーション時間	10000 clock
評価	パケット発生確率 - 平均レイテンシ

パケット発生確率

パケット発生確率はフリットが毎クロックネットワーク内の投入される確率を 1 とする。

3.5.3 シミュレーション結果と考察

図 3.7 にランダム転送を行なった際のメッセージ発生確率と平均通信時間の関係を示す。提案した同次元迂回ルーティングによる適応型ルーティング (ADP) は、通常のデッドロックフリーな再帰ルーティング (DLF) に比べ高い性能が得られた。ネットワーク内の通信量が最大となり平均通信時間が飽和する付近では、再帰ルーティングが約 0.05(flit/clock) であるのに対し適応型ルーティングでは約 0.065(flit/clock) であり、性能向上比は約 1.3 倍であった。これは、256PEs の 1D-SRT では同次元迂回ルーティングを適用できる範囲が十分に大きく、迂回により混雑が回避できたためであると考えられる。

図 3.8 に他の相互結合網でのランダム転送によるメッセージ発生確率と平均通信時間の関係を示す。図 3.8 より、SRT は他の結合網と比較しても十分に高い性能を有していることが分かる。メッシュ網 (MESH) との比較では、平均通信時間が飽和する付近では、メッシュ網が約 0.035(flit/clock) であるのに対し 1D-SRT では約 0.065(flit/clock) と約 1.8 倍の性能向上が見られた。

また、TESH 網 (TESH) では階層間の多重度 q を最大である 2 にすることでメッシュ網と同等の性能が得られることが分かった。したがって、動的性能に重点を置く場合は SRT を、レイアウト等の実装性に重点を置く場合は TESH 網を選択するのがよいと考えられる。

図 3.9 に仮想チャネルを更に付加した場合の動的性能を示す。図中、DLF(L) はデッドロックフリーな再帰ルーティングを示し、ADP(L) は同次元迂回ルーティングを示す。また、括弧内 L は仮想チャネル数を表す。図 3.9 より、適応型ルーティングでは仮想チャネ

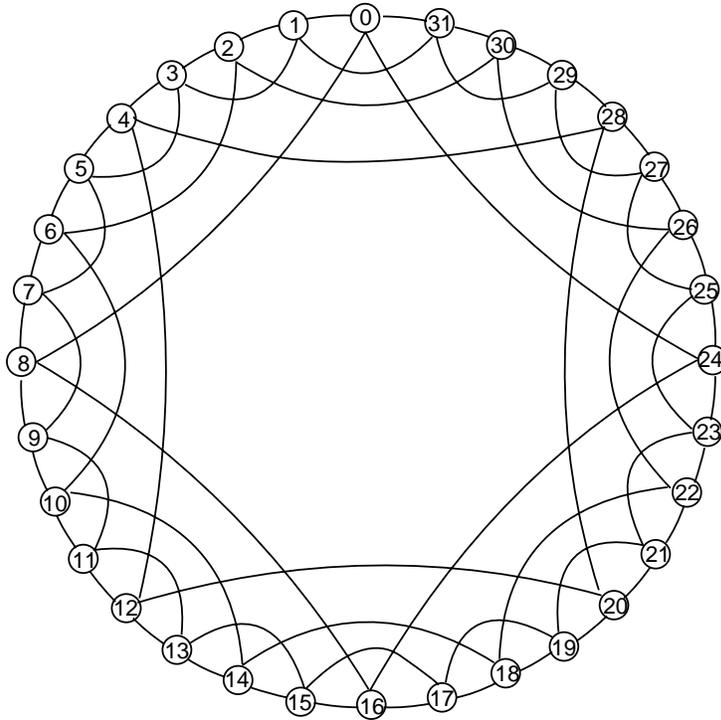


図 3.6: 32 ノードから成る 1-shift 型 1D-SRT .

ルを増設することにより，更に性能を上げることができた．特に，仮想チャネル数を 4 にしたときの性能向上が著しく，再帰ルーティング (DLF(4)) が約 0.045(flit/clock) で平均通信時間が飽和してしまうにもかかわらず，適応型ルーティング (ADP(4)) の飽和点は約 0.105(flit/clock) である．

以上より，本論文で提案した同次元迂回ルーティングでは，仮想チャネルを付加することなく，従来の再帰ルーティングに比べ十分に高い性能を得ることができた．また，仮想チャネルを増設することで飛躍的に性能が向上することもわかった．更に，他の相互結合網と比較しても高い性能が得られることをシミュレーションにより示した．

3.6 まとめ

本章では一次元再帰網のデッドロックフリー・適応型ルーティング手法を提案した．提案手法は同次元迂回ルーティングと呼ばれ，Turn Model では禁止されていた同次元上での迂回が可能である．同次元迂回ルーティングは既存のデッドロックフリーな再帰ルーティングと選択的に用いることができ，新たに仮想チャネルを付加する必要がないという利点がある．また，シミュレーションにより動的性能評価を行なった結果，従来の再帰ルーティングと比べ大幅な性能向上が実現できた．また，同じノード数のメッシュ網との比較

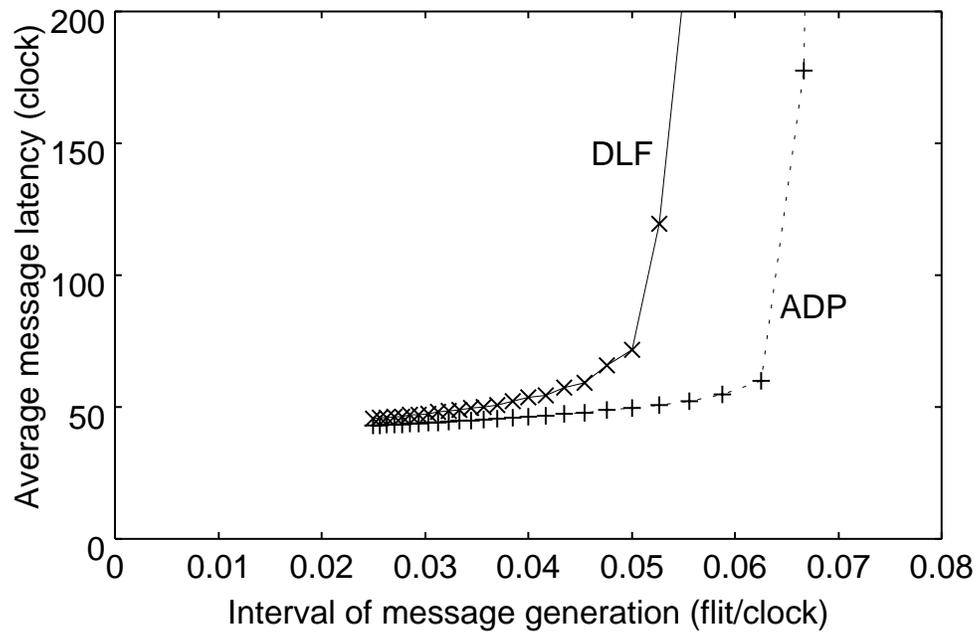


図 3.7: ランダム転送時の平均通信時間

では平均通信時間の飽和する付近で約 1.8 倍の性能向上が見られた。更に，仮想チャネルを新たに付加することで性能を向上できることが分かった。

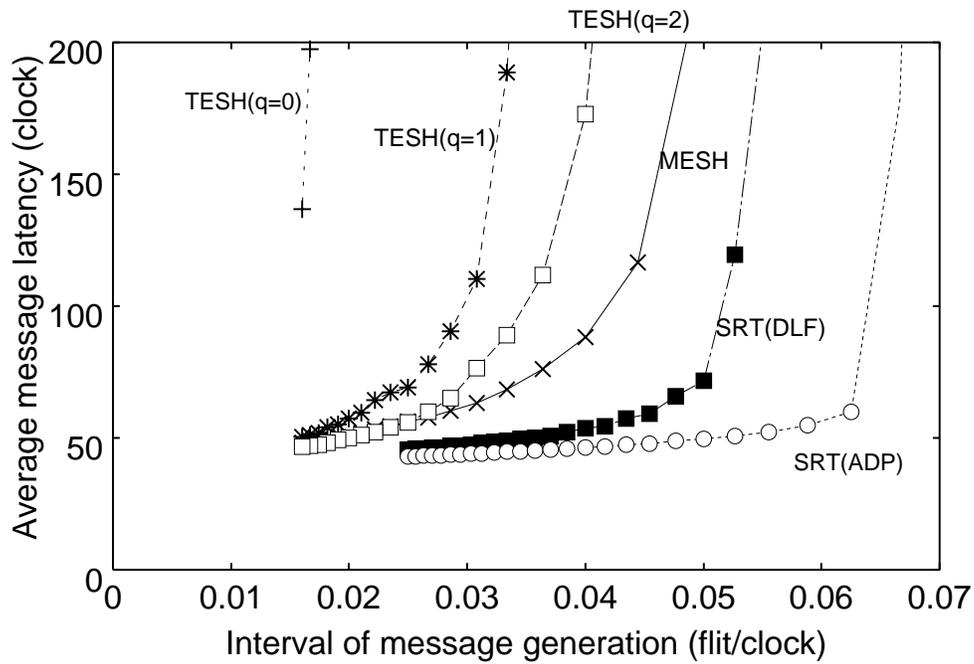


図 3.8: 他の結合網におけるランダム転送時の平均通信時間

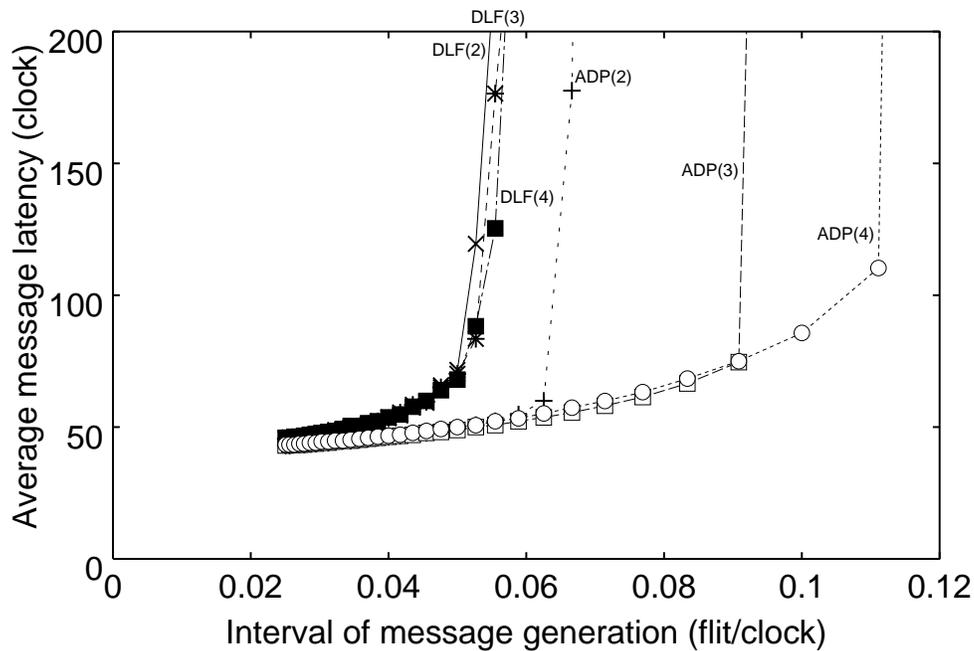


図 3.9: 仮想チャネル数と平均通信時間の関係

第 4 章

二次元再帰網に対する適応型ルーティング

4.1 はじめに

二次元の再帰的結合網には，一次元再帰網を y 方向へ拡張し構成するものと，メッシュ網あるいはトーラス網を元に一次元同様何らかの条件に従いながら構成するものがある．前者の代表的な結合網としては SRT, PEC 等が挙げられ，後者としては RDT 等が挙げられる．構成方法に違いはあるものの，両者とも各レベルではメッシュ結合網あるいはトーラス結合網が構成されている．二次元再帰網におけるルーティングは一次元再帰網同様，できる限り付加されたバイパスリンクを利用する方針で行なわれており，直径，平均距離といった点では十分に高い性能を得ることができる．しかしながら，二次元再帰網においても一次元再帰網と同様，適応性および耐故障性において問題がある．

本章では，二次元再帰網では Dimension order routing を用いることでデッドロックフリーが実現できることを示す．また，一次元再帰網における同次元迂回ルーティングを次元オーダで用いることで適応型ルーティングが実現できることを示す．さらに二次元再帰網の特徴であるの平面的構造とバイパスリンクの存在という 2 つの利点を考慮した適応型ルーティングを提案し，その動的性能特性についてシミュレーションにより詳しく議論する．

4.2 次元オーダールーティングによる デッドロック回避と適応性

二次元再帰網におけるデッドロックフリー・ルーティング，適応型ルーティングは一次元再帰網におけるそれを次元オーダで用いることで実現できる．本節では，一般の二次元再帰網を定義し，一次元再帰網におけるルーティング手法である Monotonic order routing と同次元迂回ルーティングが次元オーダで用いることによりそのまま有効であることを

示す．

4.2.1 定義

本節では一般的な二次元再帰網の定義を行なう．

任意の $N \times N$ ノードからなる二次元再帰網は一つの基本結合と一つ以上の上位結合から成る．基本結合，上位結合は次の定義に従う．

定義 11 (二次元再帰網) 基本結合は全てのノードから構成され，メッシュ結合あるいはトーラス結合で構成されている．上位結合は一部のノード同士がメッシュ結合あるいはトーラス結合で構成される．

二次元再帰網のノードアドレスを (x, y) とし，各レベルの基本結合上でのホップ数を d_l ，各ノードが持つ上位レベルの数を l_{xy} としたとき，任意のノード (x, y) は基本結合を構成する 4 ノードと，上位結合を構成する $4 \times l_{xy}$ ノードと結合する．多くの場合，次数が大きくなるのを防ぐため，各ノードが持つことのできる上位レベルの数を制限している．代表的な二次元再帰網である SRT，PEC などは予め上位レベルの数を 1 に制限している．RDT では各ノードが全ての上位リンクを持つ完全 RDT ($\text{RDT}(n, l_{max})$) と，上位リンクの数を m に制限した $\text{RDT}(n, l_{n_{xy}}, m)$ が定義されている．ここで n は上位レベルの 1 つ下のレベルにおけるホップ数である．ただし，RDT では実装面を考慮し $n = 2, l_{max} = 4, m = 1$ とした $\text{RDT}(2, 4, 1)$ / について検討がなされている．

次に任意の仮想チャネルのチャネル番号を定義する．

定義 12 (二次元再帰網のチャネル番号 (1)) 任意のノード (y, x) の各チャネルに対し次のようなチャネル番号 (d, v, n_y, n_x, l) を割り当てる．

d :

y 方向のチャネル: 1

x 方向のチャネル: 0

n_y, n_x :

正方向: ノード番号

負方向: その次元のサイズに対するノード番号の補数

また， v, l については定義 10 に従う．

4.2.2 デッドロック回避と Dimension order routing

任意の二次元再帰網では Dimension order routing によりデッドロックが回避でき、また、デッドロックフリーな同次元迂回ルーティングが可能であることを示す。

補題 1 任意の二次元再帰網の各次元上での *Monotonic order routing* はデッドロックフリーである。

証明 パケットが Monotonic order routing で x 方向に転送されているとき、要求されるチャンネル番号はパケットの存在する y アドレスとは無関係であり、一次元再帰網のそれと全く等価である。したがって、定理 2 より x 方向への Monotonic order routing はデッドロックフリーである。

次にパケットが y 方向に転送されているとき、要求されるチャンネル番号はパケットの存在する x アドレスとは無関係である。したがって、 x 方向の転送時と同じく y 方向への Monotonic order routing はデッドロックフリーである。

また、パケットがラウンドトリップループを通過する必要がある時は、始めに class 0 の仮想チャンネルにより転送を開始し、ラウンドトリップループを通過したとき、仮想チャンネルを class 1 に切替えることで単調増加が保存できる。

以上より、任意の二次元再帰網の各次元上での Monotonic order routing はデッドロックフリーである。□

補題 2 任意の二次元再帰網の各次元上で、 x 方向から y 方向へのターンはデッドロックフリーである。

証明 まず、ノード (x, y) にあるパケットの x の正方向から y 正方向へのターンについて考える。この時、要求されるチャンネル番号の順序は次のようになる。

$$(0, v, x, y, l_{cur}) \longrightarrow (1, v, x + d_{l_{cur}}, y, l_{next})$$

したがって、2つのチャンネルの間には常に

$$(0, v, x, y, l_{cur}) < (1, v, x + d_{l_{cur}}, y, l_{next})$$

が成り立つ。したがって、チャンネル番号は昇順となりデッドロックは生じない。

次に、 x の正方向から y 負方向、 x の負方向から y 正・負方向へのターンでも同様な議論ができ、デッドロックは生じない。したがって、 x 方向から y 方向へのターンはデッドロックフリーである。□

定理 4 任意の二次元再帰網において *Dimension order routing* はデッドロックフリーである。

証明 *Dimension order routing* では異なる次元への転送されるのは x の方向から y 方向のみである。

補題 1 より各次元上ではチャンネル番号は常に昇順となり，補題 2 より x の方向から y 方向への転送の際もチャンネル番号は常に昇順である。したがって，ルーティング全体でチャンネル番号が昇順となることが保証される。つまり *Dimension order routing* はデッドロックフリーである。□

以上より，任意の二次元再帰網に対し *Dimension order routing* が有効であることが分かった。また，二次元再帰網に対し *Dimension order routing* が有効であることは天野 [7]，川井ら [19] によって確認されている。

RDT では斜め方向へのリンクがあるため，単純に *Dimension order routing* を適用しても RDT の特長を發揮することは出来ない。そこで天野ら [7] はルーティングをレベルオーダ（文献中ランクオーダ）で行なうことによりデッドロックフリーを実現している。ただし，RDT に限らず x あるいは y 方向以外へのリンクが存在する場合は，その方向を新たな次元と捉えることにより定理 4 の考えが適用できる。

次に二次元再帰網でもデッドロックフリーな同次元迂回ルーティングが可能であることを示す。

定理 5 任意の二次元再帰網において，パケットの存在する各次元でのノード番号が式 (3.2)，(3.3) を満たす時，デッドロックフリーな同次元迂回ルーティングが可能である。

証明 定義より各次元のチャンネル番号は一次元再帰網におけるチャンネル番号と同じ体系である。したがって， x 方向， y 方向，それぞれに定理 3 が適用できる。つまり，パケットの存在するノードが式 (3.2)，(3.3) を満たす時，デッドロックフリーな同次元迂回ルーティングが可能である。□

以上より，任意の二次元再帰網において *Dimension order routing*，同次元迂回ルーティングが可能であることが分かった。したがって，二次元再帰網においてもそれらを選択的に用いた適応型ルーティングが可能となる。

4.3 二次元再帰網に適した適応型ルーティング

二次元再帰網に対する適応型ルーティング手法としては，第 3 章で提案した同次元迂回ルーティングの次元オーダでの適用の他に，従来からの代表的手法である Duato の手法 [13, 14] や Turn Model [15] などが挙げられる。しかしながら，Duato の手法は最短経路に

よる通信を保証する必要がある．RDTではルーティングの際，使用されるリンクのレベルに順序があるため同一レベルでの最短経路の保証は容易である．しかし，一般的には使用されるリンクのレベルの順序が規則的である保証はなく，最短経路を通過するという条件を満たすのは容易ではない．また，先に提案した同次元迂回ルーティングとの併用も困難となる．

二次元再帰網における Turn Model は一次元再帰網で見られていた問題が解消され，基本的には仮想チャネルを付加する必要がないなどの利点がある．さらに，最短経路をルーティングする必要がなく，Duatoの手法に比べても経路選択の自由度がかなり高い．しかしながら，Turn Modelではターンの方向については十分に議論されているものの，そのホップ数までは議論されておらず，目的ノードを飛び越した場合，目的ノードまでのパスが存在するか否かは保証されていない．また，そのことを防ぐために上位リンクの使用を制限した場合，ルーティングが保守的になってしまい，再帰網の特長を十分に発揮出来ない可能性がある．

そこで本論文では同次元迂回ルーティングの導出と同様な手法で Turn Model のような適応化手法を提案する．提案する手法では再帰ルーティングや同次元迂回ルーティングと併用して用いることができ，経路選択の点から更に自由度を上げることができる．

4.3.1 定義

同次元迂回ルーティングは一次元再帰網の各仮想チャネルに対しそれぞれ番号を割り当て，チャネル番号に基づき，同次元上で迂回可能な領域を導いた．二次元再帰網における適応型ルーティングでも同様な手法により，異なる次元の経路が可能となる領域を導く．しかしながら，従来のチャネル番号の割り当ては，Dimension order routing を行なうことを前提としているため，次元オーダを守らないとデッドロックが生じてしまう．

そこで二次元再帰網の各チャネルに対し定義 13 に示すようにチャネル番号を割り当て，次元オーダを守る必要のないルーティング手法の提案する．

定義 13 (二次元再帰網のチャネル番号 (2)) 各ノード (x, y) のチャネルに対し，図 4.1 ようなチャネル番号 (v, n_y, v, n_x, l) を割り当てる．

ここで v は仮想チャネルのクラス番号， l は各チャネルのレベル， R は任意の正の整数である． □

4.3.2 二次元再帰網の適応型ルーティングの導出

定義 13 により，Dimension order routing，同次元迂回ルーティングが可能であることを示す．

補題 3 定義 13 によるチャネル番号は，二次元再帰網においてある次元を一方向にルーティングしている時，そのチャネル要求は定義 12 によるそれと等価である．

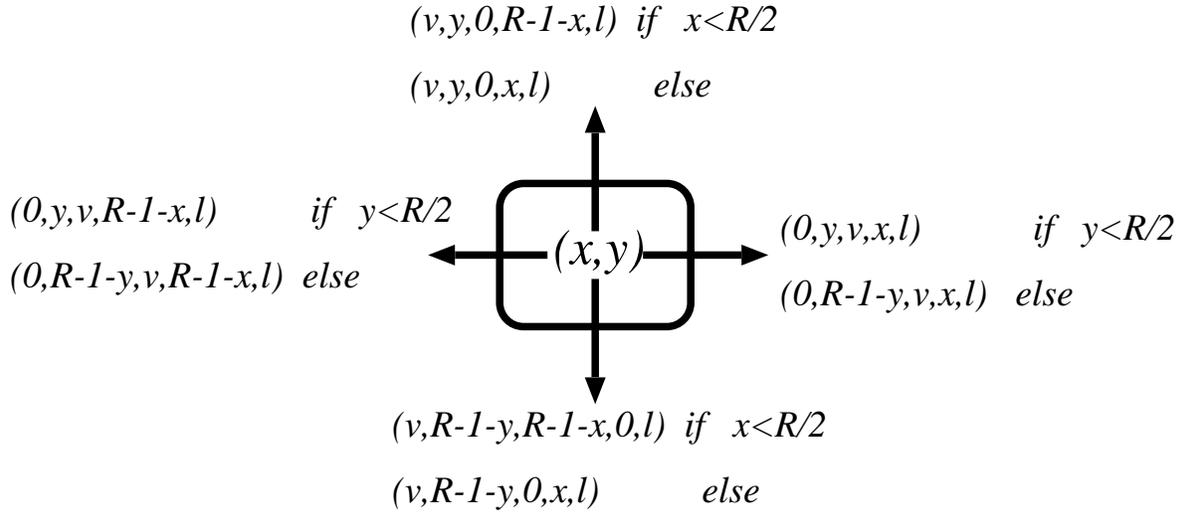


図 4.1: 各ノードにおけるチャネル番号の割り当て

証明 ノード (x, y) にあるパケットが x 方向にのみ転送されるとき，定義 12 によるチャネル要求は，

$$(0, v, x_{cur}, y_{cur}, l_{cur}) \longrightarrow (0, v, x_{cur} - d_{l_{cur}}, y_{cur}, l_{next}),$$

あるいは，

$$(0, v, N - 1 - x_{cur}, y_{cur}, l_{cur}) \longrightarrow (0, v, N - 1 - x_{cur} + d_{l_{cur}}, y_{cur}, l_{next}),$$

となり，常に，

$$x_{cur} < x_{cur} + d_{l_{cur}},$$

が成り立つ．したがって，通信前後でのチャネル番号の差 Δ_1 は，

$$\Delta_1 = d_{l_{cur}},$$

となる．ただし，ラウンドトリップループを通過した時を除く．

次に定義 13 によるチャネル要求は順次，

$$(0, y_{cur}, v, x_{cur}, l_{cur}) \longrightarrow (0, y_{cur}, v, x_{cur} + d_{l_{cur}}, l_{next}),$$

あるいは，

$$(0, y_{cur}, v, R - 1 - x_{cur}, l_{cur}) \longrightarrow (0, y_{cur}, v, R - 1 - x_{cur} + d_{l_{cur}}, l_{next}),$$

となる．したがって，

$$x_{cur} < x_{cur} + d_{l_{cur}},$$

が常に成り立ち，ラウンドトリップループを通過した時を除けば，通信前後でのチャンネル番号の差 Δ_2 は

$$\Delta_2 = d_{l_{cur}},$$

となる．したがって $\Delta_1 = \Delta_2$ となり，どちらも一定量変化している．つまり，パケットが x の方向にのみ転送されるとき 定義 12 と 定義 13 は全く等価なものであるといえる．

次に，パケットが y の方向にのみ転送されるときであるが，この時も，2つの系列での通信前後でのチャンネル番号の差は

$$\Delta_1 = \Delta_2 = d_{l_{cur}},$$

となり一定である．以上より，ある次元を一方向にルーティングしてるい時，2つのチャンネル番号体系は等価である． \square

補題 3 より，同一次元上のみで行なわれるルーティングに関し以下の系が言える．

系 3 定義 13 によりチャンネル番号を割り当てた二次元再帰網の各次元では，*Monotonic order routing* が可能である．

系 4 定義 13 によりチャンネル番号を割り当てた二次元再帰網では，各次元において条件 3.2, 3.3 が満たされるとき同次元迂回ルーティングが可能である．

次に， x 方向から y 方向へのターンがデッドロックフリーであることを示す．

定理 6 (二次元再帰網での x 方向から y 方向へのターン) 定義 13 によりチャンネル番号を割り当てた二次元再帰網において， x 方向から y 方向へのターンはデッドロックフリーである．

証明 パケットが x の正の方向から y の正の方向に転送されるときについて考える．

(i) $x_{next} < \frac{R}{2}$ のとき

要求されるチャンネル番号は，

$$(0, *, v, x_{cur}, l_{cur}) \longrightarrow (0, *, v, R - 1 - x_{cur} - d_{l_{cur}}, l_{next}),$$

となる．記号 $*$ は任意の y アドレスを示す．チャンネル番号が昇順である条件は，

$$x_{cur} < R - 1 - x - d_{l_{cur}},$$

であり，この不等式を解くと，

$$x_{cur} < \frac{R-1}{2} - \frac{d_{l_{cur}}}{2},$$

となり，仮定を満たす．

(ii) $x_{next} < \frac{R}{2}$ のとき

このとき，要求されるチャンネル番号は，

$$(0, *, v, x_{cur}, l_{cur}) \longrightarrow (0, *, v, x_{cur} + d_{l_{cur}}, l_{next}),$$

となり，チャンネル番号は常に昇順であるのは明らかである．ただし，記号 * は任意の x アドレスである．

(i), (ii) より，パケットが x の正の方向から y の正の方向に転送される時はデッドロックは生じない．残りの3つパターンでも同様に，チャンネル番号は常に単調増加となる．したがって x 方向から y 方向へのルーティングはデッドロックフリーである． \square

したがって，系3，定理6より Dimension order routing がデッドロックフリーであることが言える．

定理7 (定義13による二次元再帰網での Dimension order routing) 定義13によりチャンネル番号を割り当てた二次元再帰網では Dimension order routing が可能である．

証明 定理4と全く同様な議論ができ，チャンネル番号は常に昇順である． \square

以上より，定義13によりチャンネル番号が割り当てられた二次元再帰網でも Dimension order routing と同次元迂回ルーティングとを選択的に用いた適応型ルーティングが可能となる．

しかしながら，チャンネル番号が昇順となるのは，Monotonic order routing や同次元迂回ルーティングだけではない．そこで，本論文ではターンの制限を受けない領域，つまり， y 方向から x 方向へのターンを行なってもチャンネル番号が昇順となる領域を導き出し，より自由度の高い経路選択手法を提案する．なお，この y 方向から x 方向へのルーティングを Dimension reversal routing[10] と呼ぶ．

定理 8 (二次元再帰網での Dimension reversal routing) 定義 13 によりチャンネル番号を割り当てた二次元再帰網で, *Dimension reversal routing* がデッドロックフリーである必要十分条件は, パケットの存在するノードを (x_{cur}, y_{cur}) , 次の転送先を (x_{next}, y_{next}) としたとき, それらが式 (4.1) あるいは 式 (4.2) を満たすことである.

$$dir_{cur} \notin y_- \quad \text{and} \quad y_{cur} < \frac{R-1}{2} - \frac{d_{l_{cur}}}{2}, \quad (4.1)$$

あるいは,

$$dir_{cur} \notin y_+ \quad \text{and} \quad y_{cur} \geq \frac{R-1}{2} + \frac{d_{l_{cur}}}{2}, \quad (4.2)$$

ここで dir はパケットの進行方向を示し, y_+ は y の正方向, y_- は y の負方向を示す.

十分性の証明

十分性を示す為に, 条件は満たされているものとして仮定して y の方向から x の方向へのターンが可能となることを示す.

(I) $dir_{cur} \notin y_-$ のとき

式 (4.1) は,

$$\begin{aligned} y_{cur} &< \frac{R-1}{2} - \frac{d_{l_{cur}}}{2}, \\ 2y_{cur} &< R-1 - d_{l_{cur}}, \\ y_{cur} &< R-1 - y_{cur} - d_{l_{cur}}, \\ y_{cur} &< R-1 - (y_{cur} + d_{l_{cur}}), \end{aligned}$$

と変形できる. パケットは y の正方向のチャンネルを要求するので

$$y_{next} = y_{cur} + d_{l_{cur}},$$

が成り立ち,

$$y_{cur} < R-1 - y_{next},$$

となる. したがって, 任意の x_{cur}, x_{next} に対し,

$$(0, y_{cur}, 1, x_{cur}, l_{cur}) < (0, R-1 - y_{next}, v, x_{next}, l_{next}), \quad (4.3)$$

が成り立つ．つまり，

$$(0, R - 1 - y_{next}, v, x_{next}, l_{next}),$$

という形式を採る， x 方向のどんなチャンネルを要求しても式 (4.3) は満足される．したがって，デッドロックフリーな Dimension reversal routing が可能であるといえる．

(ii) $dir_{cur} \notin y_+$ のとき

式 (4.2) は，(i) と同様，

$$\begin{aligned} y_{cur} &\geq \frac{R-1}{2} + \frac{d_{l_{cur}}}{2}, \\ 2y_{cur} &\geq R-1 + d_{l_{cur}}, \\ y_{cur} &\geq R-1 - y_{cur} + d_{l_{cur}}, \\ y_{cur} &\geq R-1 - (y_{cur} - d_{l_{cur}}), \end{aligned}$$

と変形できる．パケットは y の負方向のチャンネルを要求するので

$$y_{next} = y_{cur} - d_{l_{cur}},$$

が成り立ち，

$$y_{cur} \geq R - 1 - y_{next},$$

となる．したがって，先ほど同様，任意の x_{cur} ， x_{next} に対し，

$$(0, R - 1 - y_{next}, v, x_{next}, l_{next})$$

という形式を採る x 方向の全てのチャンネルが要求可能である．

以上より，式 (4.1)，(4.2) が満たされるとき，Dimension reversal routing が可能で，かつ，デッドロックフリーである．

必要性の証明

必要性を示す為には，デッドロックが生じないための条件を導出し，導いた結果が式 (4.1)，式 (4.2) と一致すれば良い．

まず， y の正方向から x 方向へのターンについて条件を導く．

(I) $dir_{cur} \notin y_-$ のとき

(i) $y_{next} < \frac{R}{2}$ のとき

パケットの存在するノードを (x_{cur}, y_{cur}) ，次の転送先を (x_{next}, y_{next}) とするとパケットの通過するチャンネル番号は

$$(v, y_{cur}, 0, *, l_{cur}) \longrightarrow (v, y_{next}, 0, *, l_{next}),$$

となるので , デッドロックフリーであるための条件は

$$(v, y_{cur}, 0, *, l_{cur}) < (v, y_{next}, 0, *, l_{next}), \quad (4.4)$$

である . 更に , $y_{next} = y_{cur} + d_{l_{cur}}$ と表せるので , 式 (4.4) は常に成り立つ .

したがって , デッドロックフリーである条件は仮定そのものとなり

$$y_{next} < \frac{R}{2}, \quad (4.5)$$

である .

(ii) $y_{next} \geq \frac{R}{2}$ のとき

パケットの通過するチャンネル番号は

$$(v, y_{cur}, 0, *, l_{cur}) \longrightarrow (v, R - 1 - y_{next}, 0, *, l_{next}),$$

先ほどと同様に , デッドロックフリーである条件は

$$(v, y_{cur}, 0, *, l_{cur}) < (v, R - 1 - y_{next}, 0, *, l_{next}), \quad (4.6)$$

となる . したがって , 常に式 (4.6) が成り立つ条件は

$$y_{cur} < R - 1 - y_{next},$$

であり , $y_{next} = y_{cur} + d_{l_{cur}}$ と表せるので , 常にデッドロックフリーとなる条件は ,

$$y_{cur} < \frac{R - 1}{2} - \frac{d_{l_{cur}}}{2}, \quad (4.7)$$

となる .

式 (4.7) は

$$\begin{aligned} y_{cur} &< \frac{R - 1}{2} - \frac{d_{l_{cur}}}{2}, \\ y_{cur} + d_{l_{cur}} &< \frac{R - 1}{2} + \frac{d_{l_{cur}}}{2}, \\ y_{next} &< \frac{R - 1}{2} + \frac{d_{l_{cur}}}{2}, \end{aligned}$$

となる．よって，

$$\frac{R}{2} \leq \frac{R-1}{2} + \frac{d_{l_{cur}}}{2},$$

なる関係が導き出せる．したがって，式(4.7)は式(4.5)を包含している．

(i),(ii)より導いた，デッドロックフリーである為の条件は式(4.1)と一致する．

次に， y の負方向から x 方向へのルーティングについて議論する．

(II) $dir_{cur} \notin y_+$ のとき

(i) $y_{next} \geq \frac{R}{2}$ のとき

パケットの存在するノードを (x_{cur}, y_{cur}) ，次の転送先を (x_{next}, y_{next}) とするとパケットの通過するチャンネル番号は

$$(v, R-1-y_{cur}, 0, *, l_{cur}) \longrightarrow (v, R-1-y_{next}, 0, *, l_{next}) \quad (4.8)$$

となるので，デッドロックフリーであるための条件は

$$(v, R-1-y_{cur}, 0, *, l_{cur}) < (v, R-1-y_{next}, 0, *, l_{next}) \quad (4.9)$$

である．更に， $y_{next} = y_{cur} - d_{l_{cur}}$ と表せるので，式(4.9)は常に成り立つ．

したがって，デッドロックフリーである条件は仮定そのものとなり

$$y_{next} \geq \frac{R}{2} \quad (4.10)$$

である．

(ii) $y_{next} < \frac{R}{2}$ のとき

パケットの通過するチャンネル番号は

$$(v, R-1-y_{cur}, 0, *, l_{cur}) \longrightarrow (v, y_{next}, 0, *, l_{next}) \quad (4.11)$$

先ほどと同様に，デッドロックフリーである条件は

$$(v, R-1-y_{cur}, 0, *, l_{cur}) < (v, y_{next}, 0, *, l_{next}) \quad (4.12)$$

となる．したがって，常に式(4.12)が成り立つ条件は

$$R-1-y_{cur} < y_{next} \quad (4.13)$$

であり， $y_{next} = y_{cur} - d_{l_{cur}}$ と表せるので，常にデッドロックフリーとなる条件は，

$$y_{cur} > \frac{R-1}{2} + \frac{d_{l_{cur}}}{2} \quad (4.14)$$

となる．

式(4.14)は

$$y_{cur} < \frac{R-1}{2} + \frac{d_{l_{cur}}}{2} \quad (4.15)$$

$$y_{cur} - d_{l_{cur}} < \frac{R-1}{2} - \frac{d_{l_{cur}}}{2} \quad (4.16)$$

$$y_{next} < \frac{R-1}{2} - \frac{d_{l_{cur}}}{2} \quad (4.17)$$

となる．よって，

$$\frac{R}{2} \geq \frac{R-1}{2} - \frac{d_{l_{cur}}}{2} \quad (4.18)$$

なる関係が導き出せる．したがって，式(4.14)は式(4.12)を包含している．

よって，条件(4.2)に一致する．

以上(I),(II)より，Dimension reversal routingがデッドロックフリーとなるための条件が導けた．また，それらが式(4.1)，(4.2)と一致することが確認できた． □

以上より二次元再帰網では全領域で次元オーダによる再帰ルーティングが可能である．また，一部の領域において同次元迂回ルーティングルーティング，あるいは Dimension reversal routingが可能である．

4.3.3 二次元再帰網の適応型ルーティング

4.3.2節では，定義13によりチャンネル番号を割り当てた二次元再帰網では，Dimension order routing，同次元迂回ルーティング，Dimension reversal routingが可能であることを示した．そこで，本節では，それらを用いた二次元再帰網における適応型ルーティングアルゴリズムを提案する．

いずれのルーティング手法も定義13に基づく二次元再帰網ではチャンネル番号は単調増加である．したがって，それらを選択的に用いてもデッドロックフリーは保証される．つまり，定義13，定理7，系4，定理8に従ってさえいれば，どの手法を選択してもデッドロックフリーである．

二次元再帰網における適応型ルーティングアルゴリズムを図4.2に示す．

図4.2、 N は各次元のネットワークサイズ、 R は定義13における定数である．定理8の適用範囲は定数 R により変更することが可能である． $R = N$ のとき、Dimension reversal routingで y 方向のリンクが使用可能となる範囲は正方向，負方向を問わず均一である． $R > N$ のときは正方向のリンクの使用範囲が広くなり、 $R < N$ のときは負方向のリンクの使用範囲が広がる．したがって，あらかじめノード間通信に偏りがあることが分かっている場合は定数 R を調整し，通信性能の向上が望める．

また，一次元同様 $|dst - src| > \frac{d_{l_{cur}}}{2}$ といったような行き過ぎを抑制する条件を付加することも可能である．なお，この条件も任意であり，条件は加えられなくても良い．

次に仮想チャネルについて議論する．本論文で提案する手法では，通常のデッドロックフリーな再帰ルーティングに対し，同次元迂回ルーティングと Dimension reversal routing を単純に追加することが可能であり，再帰ルーティングそのものに変更を加える必要はない．また，提案手法では経路選択の自由度を上げることにより適応性を得ているため仮想チャネルは必要最小限でよく，仮想チャネルを新たに付加する必要がない．

しかし，トラス網上で Dimension reversal routing を用いる際，仮想チャネルのクラスは常に 0 である必要がある．これは，Dimension reversal routing を用いた後に，再帰ルーティングが用いられる可能性があるため，Dimension reversal routing でクラス 1 の仮想チャネルが使用されてしまうと，再帰ルーティングがラウンドトリップループを用いたルーティングを行なうことができなくなってしまうからである．

逆に，仮想チャネルの増設が可能な場合は，増設分については，デッドロックフリーを保証するチャネルが既に確保されているので，転送先のチャネルが空いている時は自由に選択できる．

4.4 二次元 SRT における適応型ルーティング

本節では $N \times N$ から成る二次元 SRT (2D-SRT) に対し提案手法を適用する．2D-SRT は x 方向， y 方向，どちらにも 1D-SRT が構成されているため，各方向に対し同次元迂回ルーティングを行なうことができる．定理 8 より，2D-SRT の各ノード (x, y) で式 (4.1)，(4.2) を満たせば Dimension reversal routing も可能となる．また，定理 6 より Dimension order routing が可能である．したがって，Dimension order routing，同次元迂回ルーティング，Dimension reversal routing を選択的に用いる適応型ルーティングが可能である．同次元迂回ルーティング，Dimension reversal routing が可能となる領域は次のようにまとめられる．なお，定理 8 における定数 R は $R = N$ とした．

系 5 2D-SRT において，同次元迂回ルーティングが可能となるのは，パケットの存在するノードを (x_{cur}, y_{cur}) ，そのレベルを l_{cur} ，目的ノードを (x_{dst}, y_{dst}) とした時，次の条件を満たされることである．

(条件)

(i) x 方向に同次元迂回ルーティングが可能な条件

$$x_{cur} < x_{dst}, \quad x_{cur} < \frac{2^{l_{max}} - 1}{2} - 2^{l_{cur}-1} \quad (4.19)$$

あるいは，

$$x_{cur} > x_{dst}, \quad x_{cur} > \frac{2^{l_{max}} - 1}{2} + 2^{l_{cur}-1} \quad (4.20)$$

(ii) y 方向に同次元迂回ルーティングが可能な条件

$$x_{cur} \neq x_{dst}, \quad y_{cur} < y_{dst}, \quad y_{cur} < \frac{2^{l_{max}} - 1}{2} - 2^{l_{cur}-1} \quad (4.21)$$

```

AdaptiveRouting( $x_{cur}, y_{cur}, x_{dst}, y_{dst}$ ){
    if(  $x_{cur} \neq x_{dst}$  )
        if(  $x_{cur} < x_{dst}$  ) {  $dir = +1$  }
        else{  $dir = -1$  }
         $Next1 = RecursiveRouting(x_{cur}, x_{dst})$ 
         $Next2 = DetourRouting(x_{cur}, x_{dst})$ 
         $Next3 = ReversalRouting(x_{cur}, x_{dst})$ 
        if(  $Next1 == BUSY \ \&\& \ Next2 == FREE$  ) {
            return(Next2)
        }
        elseif(  $Next1 == BUSY \ \&\& \ Next2 == BUSY \ \&\& \ Next3 \neq NULL \ \&\& \ Next3 == FREE$  ) {
            return(Next3)
        }
        else{
            return(Next1)
        }
    }
elseif(  $x_{cur} == x_{dst} \ \&\& \ y_{cur} \neq y_{dst}$  ) )
    if(  $y_{cur} < y_{dst}$  ) {  $dir = +1$  }
    else{  $dir = -1$  }
     $Next1 = RecursiveRouting(y_{cur}, y_{dst})$ 
     $Next2 = DetourRouting(y_{cur}, y_{dst})$ 

    if(  $Next1 == BUSY \ \&\& \ Next2 == FREE$  ) {
        return(Next2)
    }
    else{
        return(Next1)
    }
}
}

```

```

DetourRouting(cur, dst)
  if( cur < dst ) { dir = +1 }
  else{ dir = -1 }
  if( cur < dst && cur <  $\frac{N}{2} - \frac{d_{i_{cur}}}{2}$  ){
    return(cur + dir ×  $d_{i_{cur}}$ )
  }
  elseif( cur > dst && cur >  $\frac{N}{2} + \frac{d_{i_{cur}}}{2}$  ){
    return(cur + dir ×  $d_{i_{cur}}$ )
  }
  else{
    return(cur + dir)
  }
}

```

```

RiversalRouting(cur, dst)
  if( cur < dst ) { dir = +1 }
  else{ dir = -1 }
  if( dir == 1 && cur <  $\frac{R-1}{2} - \frac{d_{i_{cur}}}{2}$  ){
    return(cur + dir ×  $d_{i_{cur}}$ )
  }
  elseif( dir == -1 && cur >  $\frac{R-1}{2} + \frac{d_{i_{cur}}}{2}$  ){
    return(cur + dir ×  $d_{i_{cur}}$ )
  }
  else{
    return(NULL)
  }
}

```

図 4.2: 二次元再帰網の適応型ルーティング

あるいは,

$$x_{cur} \neq x_{dst} \quad y_{cur} > y_{dst}, \quad y_{cur} > \frac{2^{l_{max}} - 1}{2} + 2^{l_{cur}-1} \quad (4.22)$$

ここで $l_{max} = \log_2 N$ である .

系 6 2D-SRTにおいて, *Dimension reversal routing* が可能となるのは, パケットの存在するノードを (x_{cur}, y_{cur}) , そのレベルを l_{cur} , 目的ノードを (x_{dst}, y_{dst}) とした時, 次の条件を満たされることである .

(条件)

$$dir_{cur} \notin y_- \quad \text{and} \quad y_{cur} < \frac{2^{l_{max}} - 1}{2} - 2^{l_{cur}-1}, \quad (4.23)$$

あるいは,

$$dir_{cur} \notin y_+ \quad \text{and} \quad y_{cur} \geq \frac{2^{l_{max}} - 1}{2} + 2^{l_{cur}-1}. \quad (4.24)$$

ここで dir はパケットの進行方向を示し, y_+ は y の正方向, y_- は y の負方向を示す . また, $l_{max} = \log_2 N$ である .

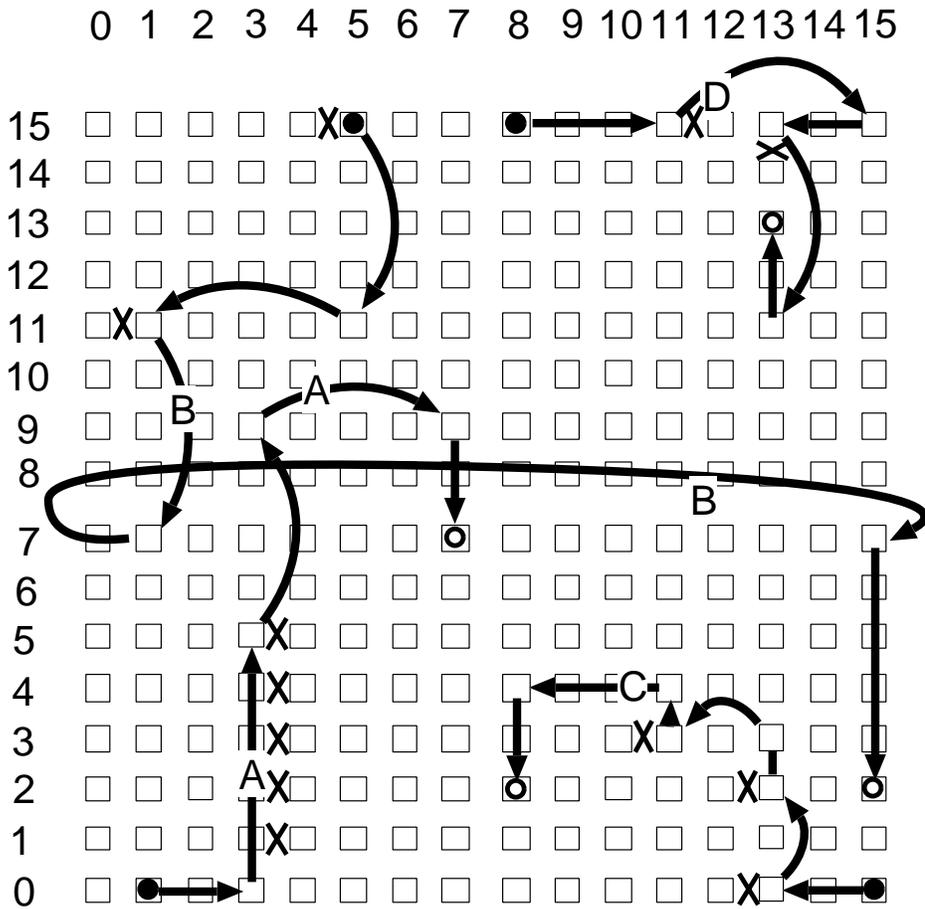
2D-SRTで Dimension order routing, 同次元迂回ルーティング, Dimension reversal routing を用いた時の例を図 4.3 に示す .

図 4.3 で, パケット A はノード (2,0) を出発し, 途中のノード (3,0) で y 方向へ Dimension reversal routing を行なっている . さらにノード (3,5) では上位レベルリンクを用いた Dimension reversal routing を行ない混雑を避けている . ノード (3,9) は Dimension reversal routing が可能な領域ではないため, それ以降 Dimension order routing を行なっている . パケット B は y の負方向での Dimension reversal routing の例である . パケット C は始点ノードが (15,0) で, 目的ノードが (8,2) であるため y の正方向へ転送されている場合はどのノードでも Dimension reversal routing が可能であるため, 何度もターンが可能になっている . また, パケット D は同次元迂回ルーティングを次元オーダで行なっている例である .

4.5 シミュレーション評価

4.5.1 シミュレーションの概要

二次元再帰網における, 同次元迂回ルーティング, Dimension reversal routing を用いた適応型ルーティングの動的性能評価をシミュレーションにより行なう . シミュレーションの概要を表 4.1 に示す .



-  Source
-  Destination
-  Use of Upper Level
-  Use of Level 0
-  Congestion or Fault

図 4.3: 2D-SRT での適応型ルーティングの例

表 4.1: シミュレーションの概要— (2D-SRTでの性能評価)

トポロジ	Short-Spna 型 1-Shift 2D-SRT, ハイパーキューブ
サイズ	256, 1024
パケット長	16flit
仮想チャネル数	2,3,4
転送パターン	ランダム転送, Matrix Transepose
パケット発生確率	$10^{-4} \leq P \leq 1$ (flit/clock) (36ポイントで測定)
シミュレーション時間	10000 clock
評価	パケット発生確率 - 平均レイテンシ

Matrix Transpose はノード番号が (x, y) のノードは $(N - 1 - y, N - 1 - x)$ のノードにパケットを送る通信パターンである。ここで N は各次元のノード数である。なお、通信パターン、平均通信時間、パケット発生確率については 3.5 節と同じものとする。

4.5.2 シミュレーション結果と考察

図 4.4 にランダム転送を行なった際のメッセージ発生確率と平均通信時間の関係を示す。提案した 2 つの適応化手法は通常のデッドロックフリーな再帰ルーティング (DLF) に比べ高い性能が得られた。しかし、同次元迂回ルーティングのみの手法 (ADP1) と同次元迂回ルーティングに Dimension reversal routing を加えた手法 (ADP2) とでは性能に差が見られなかった。これは、Dimension reversal routing では経路の選択性は増加するものの、一度方向転換したメッセージはある程度 y 方向に進んだ後、再度 x 方向に戻る必要がある。Dimension reversal routing の適用範囲に制限がない場合はボトルネックの心配はないが、提案する Dimension reversal routing では、その手法が適用できる範囲に制限がある為、適用可能な範囲の境界となる周辺で混雑が生じるボトルネックを引き起こす可能性がある。そのため、経路の選択性が向上しているにもかかわらず、動的性能はさほど向上していないものと考えられる。

図 4.5 に仮想チャネルの数と平均通信時間の関係について示す。図 4.5 より、256PEs の 2D-SRT では仮想チャネルの数の違いによる性能差はあまり見られなかった。仮想チャネル数が 4 の Dimension reversal routing の飽和点を見ると、メッセージ発生確率が 0.2(flit/clock) 付近であり、性能の向上率は 1D-SRT の場合と比較してそれほど大きくない。そのため、0.2(flit/clock) 付近がこの結合網の通信量の限界であると考えられる。

図 4.6 に他の結合網との性能差について示す。同次元迂回ルーティングあるいは Dimension reversal routing によるルーティングでは動的性能は HC とほぼ同程度であるといえる。

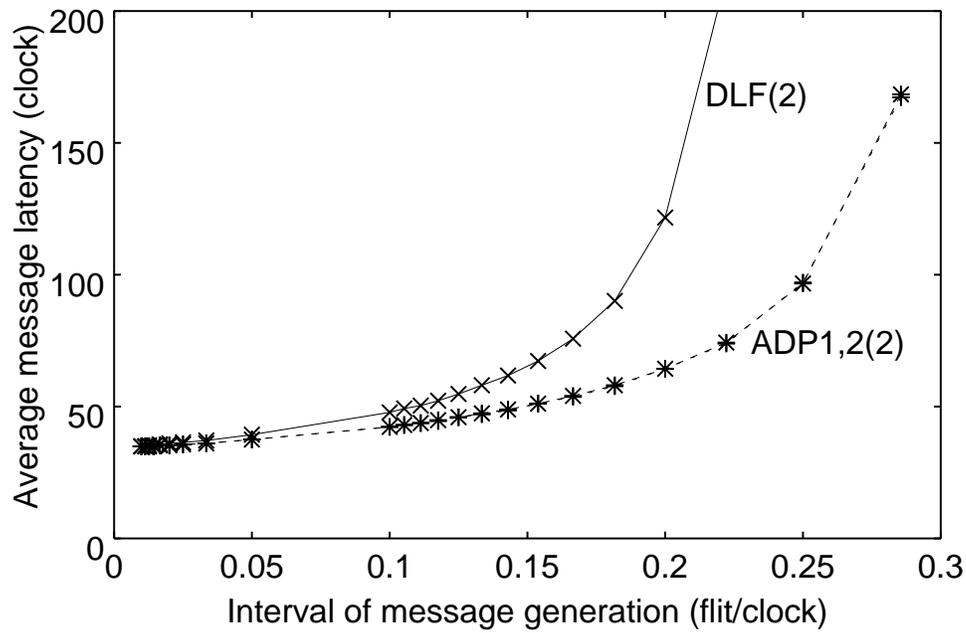


図 4.4: ランダム転送時の平均通信時間

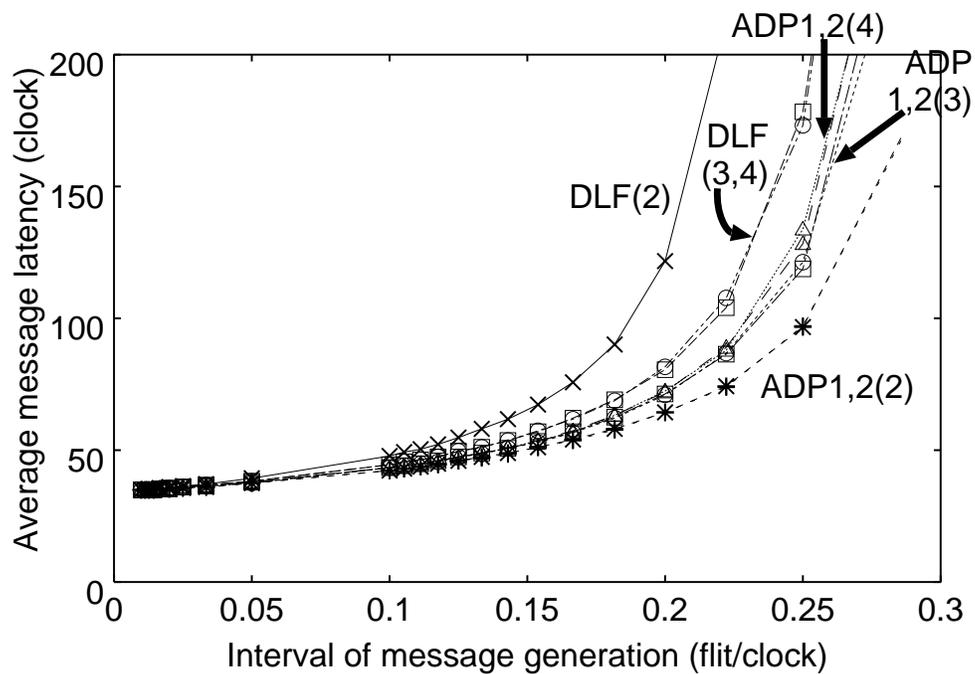


図 4.5: 仮想チャネル数と平均通信時間の関係

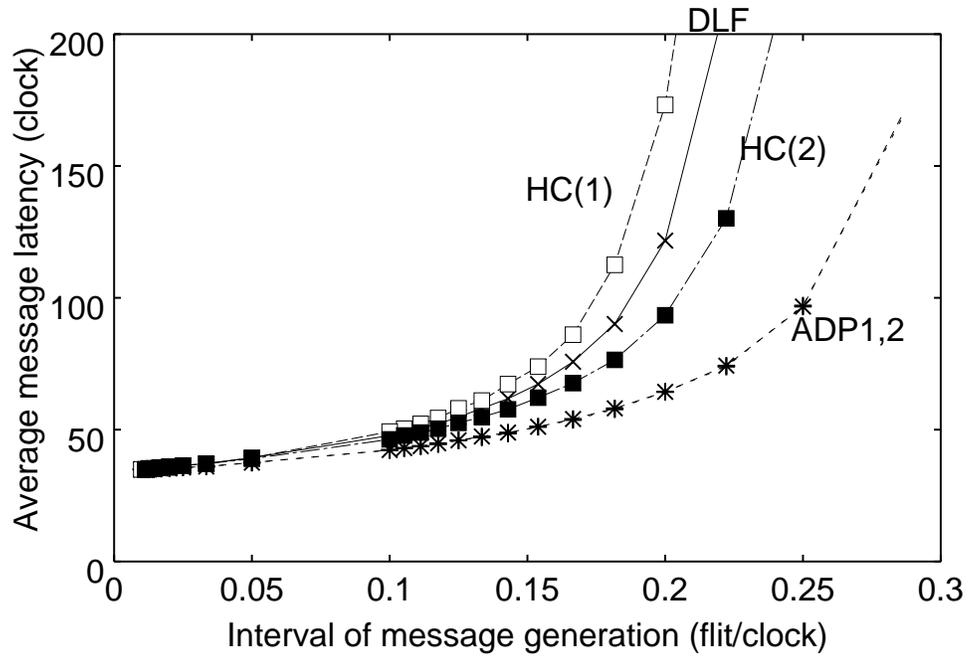


図 4.6: 256PEs での HC 網との比較

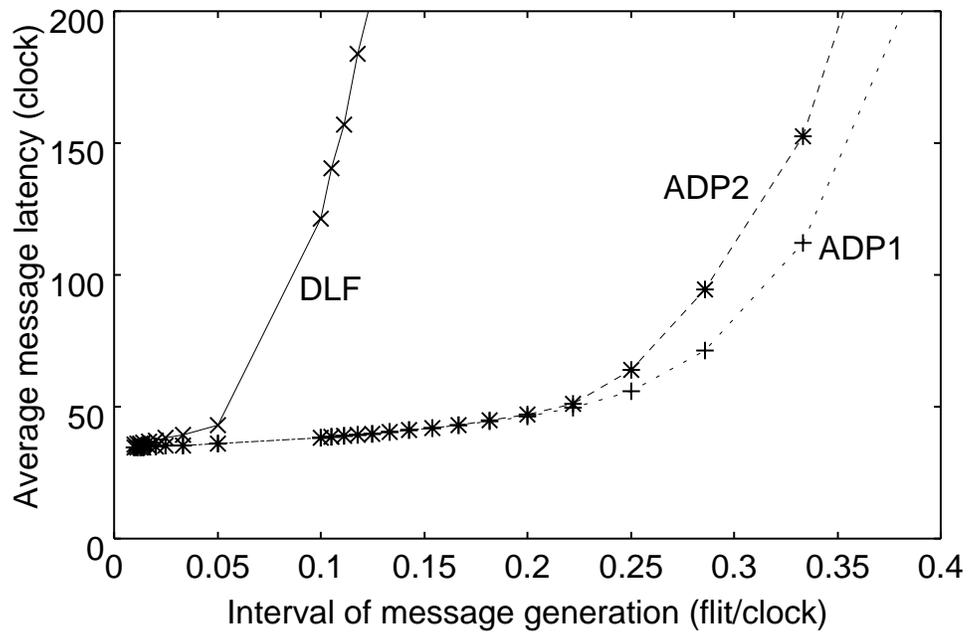


図 4.7: 転送パターンが Matrix Transpose のときの仮想チャネル数と平均通信時間の関係

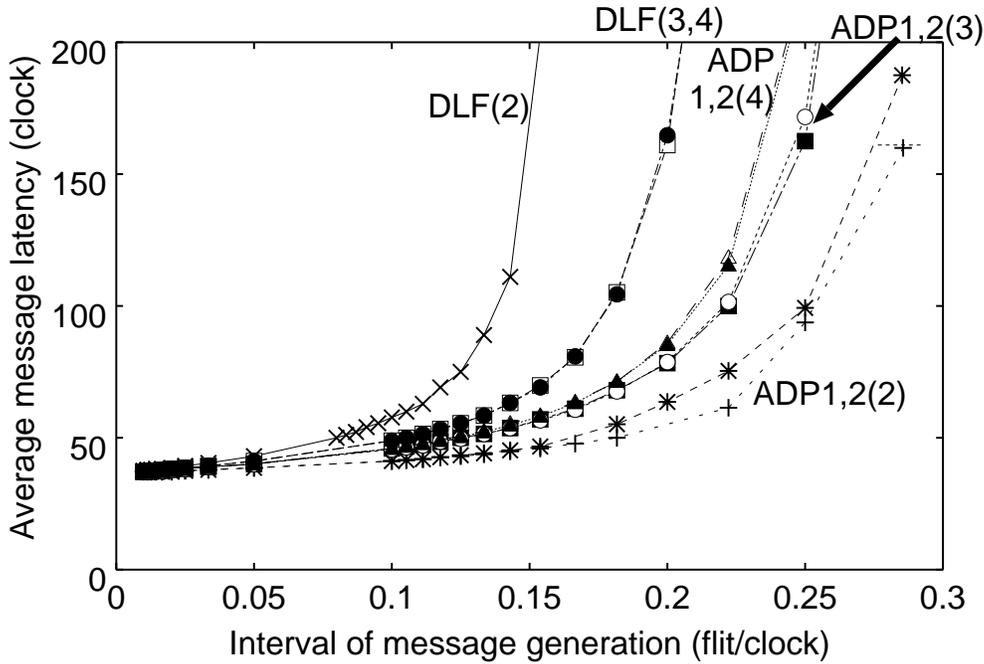


図 4.8: 1024PEs 時の仮想チャネル数と平均通信時間

通信パターンを matrix transpose にした際のメッセージ発生確率と平均通信時間の関係を図 4.7 に示す。図 4.7 より，matrix transpose のような通信パターンに偏りがあるような場合でも，通常の再帰ルーティングに比べ同次元迂回ルーティング，Dimension reversal routing は高い性能が得られることが分かった。

図 4.8 にノード数が 1024PEs のときの仮想チャネルの数と平均通信時間の関係について示す。図 4.8 より，1024PEs の 2D-SRT でも仮想チャネルの数の違いによる性能差はあまり見られなかった。ノード数が 1024PEs の時，各次元でのノード数は高々 32PEs であるため，動的通信性能にはさほど大きい差が現れなかったと考えられる。

ノード数が 1024PEs のときのハイパーキューブ網との性能比較を図 4.9 に示す。図 4.9 より，適応型ルーティングを行なった SRT の方がハイパーキューブ網 (HC) と比較して性能が高いことが分かる。特にパケット発生確率が 0.22(flit/clock) 付近での平均通信時間は仮想チャネル数が 2 の HC で約 150(clock) であるのに対し SRT では約 75(clock) と約 2 倍の差がでた。ノード数が 1024PEs の場合，HC はノードの次数が 10 であるのに対し SRT では 8 であるため性能比較にコストを含めた場合，さらに差が大きくなると考えられる。

以上より，本論文で提案する同次元迂回ルーティングは再帰ルーティングに比べ十分に高い性能を得ることができ，また，仮想チャネルを新たに付加する必要がないため，要求されるハードウェア量も少なく，十分に有効な手法であると言える。

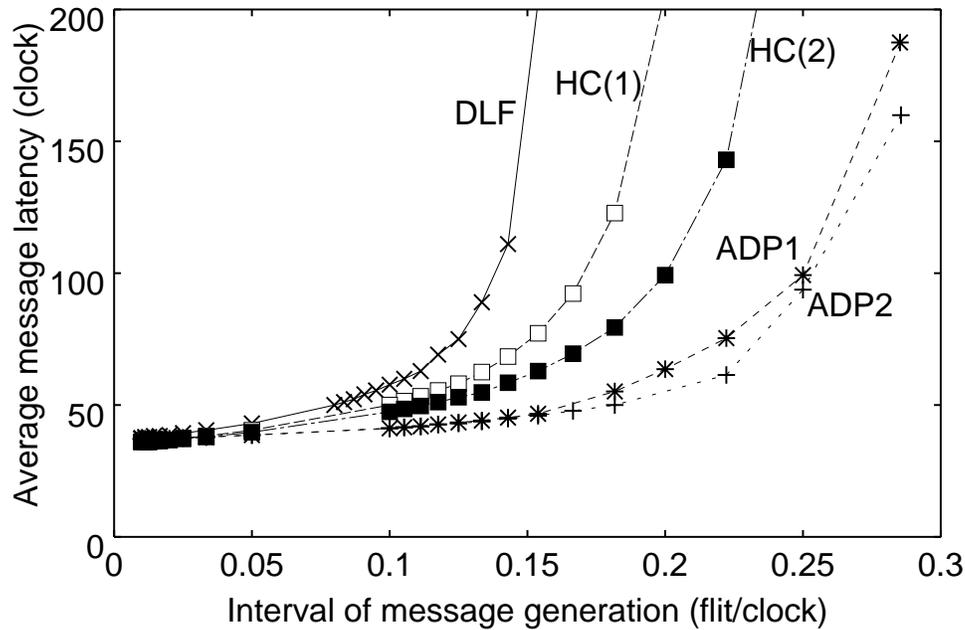


図 4.9: 1024PEs での HC 網との比較

4.6 まとめ

本章では，二次元再帰網のデッドロックフリーな適応型ルーティング手法を提案した．提案した適応化手法では，次元オーダを守らない Dimension reversal routing が可能である．また，仮想チャンネルを新たに付加する必要がないという利点がある．さらに，第3章で提案した同次元迂回ルーティングや従来のデッドロックフリーな再帰ルーティングとの併用が可能である．

また，提案手法を 2D-SRT に対し適用して，動的な性能特性を測った結果，十分に高い通信性能が得られた．また，ハイパーキューブ網と比較した結果，同等以上の結果が得られることが分かった．

第 5 章

適応型ルーティングの耐故障性

5.1 はじめに

本章では、3、4章で提案した適応型ルーティングの耐故障性について議論する。超並列計算機において PE の故障やリンクの故障などは避けられない問題である。従来の再帰ルーティングの経路決定方法は固定であるため、転送経路上に故障があった場合故障を避けることができず、それより先にメッセージを転送することはできない。そのため最悪の場合では、ただ 1 つのリンクの故障によりシステム全体が機能しなくなる可能性がある。

一方で、少なからず経路選択の自由度を持つ適応型ルーティングでは、転送経路上に混雑や故障が存在する場合、それを避けるようにルーティングすることが可能となる。如何なる故障に対しても対応できるわけではないが、固定型ルーティングに比べ、故障によるシステムの性能低下は小さくなることが期待できる。

耐故障性の評価方法として、本論文では経路選択の自由度という理論的評価と、故障によりネットワークのスループットがどのように推移するかという動的な性能評価の 2 つの評価指標を用いて耐故障性について議論する。

5.2 適応型ルーティングの耐故障性

従来の再帰ルーティングでは経路は固定であり、混雑や故障などがあってもそれを避けることは不可能である。一方で本論文で提案する適応型ルーティングでは、ある条件を満たした時、同次元迂回ルーティングや Dimension reversal routing が可能となる。本章では、提案した適応化手法が持つ、経路選択の自由度について議論する。

まず、評価の為の尺度を定義する。本章での耐故障性の評価は経路選択の自由度であるので、全ての方向に転送可能な場合に 1.0 となるような尺度を定義する。

定義 14 経路選択の自由度 ϵ は次式で定義される .

$$\epsilon = \sum_{i=1}^N \frac{dir_i}{deg_i}. \quad (5.1)$$

ここで , N はノード数 , dir_i はノード番号 i で選択可能な方向の数を表し , deg_i はノード番号 i の次数を表す .

任意の再帰網のノードの平均次数を deg_{rec} とすると再帰ルーティングの自由度 ϵ は ,

$$\epsilon = \frac{1}{deg_{rec}}, \quad (5.2)$$

となる .

例えば , 1D-SRT では , ノード次数は 4 なので再帰ルーティングによる自由度は $\frac{1}{4}$ となり , 2D-SRT では , ノード次数が 8 なので再帰ルーティングによる自由度は $\frac{1}{8}$ となる .

以下 , SRT を対象とする適応型ルーティングの自由度の評価を行なう .

5.2.1 同次元迂回ルーティングの自由度

同次元迂回ルーティングでは , 条件を満たす場合基本レベルリンクの他に上位レベルリンクを使用することができる . したがって , あるノードが上位レベルリンクを使用できる確率を r_{upper} とすると , 同次元迂回ルーティングの自由度は次式で与えられる .

$$\epsilon = \frac{1}{4}(1 + r_{upper}). \quad (5.3)$$

また , r_{upper} は次のようにして算出できる .

1D-SRT で 同次元迂回ルーティングが可能となる領域は

$$n_{cur} < n_{dst} \quad , \quad n_{cur} < \frac{2^{l_{max}} - 1}{2} - 2^{l_{cur}-1}, \quad (5.4)$$

$$n_{cur} > n_{dst} \quad , \quad n_{cur} > \frac{2^{l_{max}} - 1}{2} + 2^{l_{cur}-1}, \quad (5.5)$$

であり , 全てのノードで使用できるわけではなく , パケットの存在するカレントノードと目的ノードとの位置関係により決まってくる . 1D-SRT のレベル l ノードにおいて上位リンクが使用できる確率に関して次の定理が成り立つ .

定理 9 1D-SRT 上の任意のレベル l ノードにおいて , 同次元迂回ルーティングにより上位リンクを使用できる確率 $r_u(l)$ は

$$r_u(l) = \frac{N - 2^l}{N - 1} + \frac{2^l - 1}{N - 1} \frac{2^{l_{max}-l-1} - 1}{2^{l_{max}-l}}, \quad (5.6)$$

である . ただし , $l_{max} = \log_2 N$ である .

証明いま，パケットは正方向に転送されていると仮定し，カレントノードを n_{cur} ，目的ノードを n_{dst} とする．

(i) $n_{cur} + 2^l \leq n_{dst}$ となるとき

このとき，上位レベルリンクを使用することは，Monotonic order routing そのものなので可能である．したがって，任意のノードからの目的ノードの数 $N - 1$ に対し，仮定を満足する目的ノードは $N - 2^l$ だけ存在するので，上位リンクを使用できる確率は

$$\frac{N - 2^l}{N - 1}, \quad (5.7)$$

となる．

(ii) $n_{cur} + 2^l > n_{dst}$ となるとき

式(5.4)を満たすときのみ上位レベルリンクが使用可能となる．1D-SRTのノード0から見て i 番目のレベル l ノードは $2^l(i - 1) + 2^{l-1}$ と書けるので，パケットが正方向に転送されている時，同次元迂回ルーティングが可能なのは，

$$2^l(i - 1) + 2^{l-1} < \frac{2^{l_{max}} - 1}{2} - 2^{l-1},$$

を満たす時である．よって，

$$\begin{aligned} 2^l(i - 1) + 2^{l-1} &< \frac{2^{l_{max}} - 1}{2} - 2^{l-1}, \\ 2^l(i - 1) &< \frac{2^{l_{max}} - 1}{2} - 2^l, \\ i - 1 &< \frac{2^{l_{max}} - 1}{2^{l+1}} - 1, \\ i &< \frac{2^{l_{max}}}{2^{l+1}} - \frac{1}{2^{l+1}}, \end{aligned}$$

となり，条件を満たす最大の整数は $i = 2^{l_{max}-l-1} - 1$ である．したがって，レベル l ノードで上位レベルを使用できるのは確率は，レベル l ノードは全部で $\frac{N}{2^l} = 2^{l_{max}-l}$ だけあるので，

$$\frac{2^{l_{max}-l-1} - 1}{2^{l_{max}-l}}, \quad (5.8)$$

となり，仮定を満足する目的ノード数はある1ノードに対し $2^l - 1$ だけ存在するので，上位リンクを使用できる確率は，

$$\frac{2^l - 1}{N - 1} \frac{2^{l_{max}-l-1} - 1}{2^{l_{max}-l}}, \quad (5.9)$$

となる．

以上より，正方向の上位レベルリンクを使用できる確率は

$$r_u(l) = \frac{N - 2^l}{N - 1} + \frac{2^l - 1}{N - 1} \frac{2^{l_{max} - l - 1} - 1}{2^{l_{max} - l}} \quad (5.10)$$

となる．次に負方向に関する議論であるが，式 (5.4)，式 (5.5) より，負方向で同次元迂回ルーティングが可能な領域は正方向でそれが可能な場合とは独立であり，また，適用できる範囲の大きさは等しいので正方向と全く議論が可能である．更に，Monotonic order routing では，あるノードで選択できる方向は一方のみであり，あるノードでパケットがどちらの方向を選択するかは等確率であるので全体での確率も式 (5.10) となる． □

定理 9 より， r_{upper} を導くことができ，次のようにまとめられる．

系 7 上位レベルリンクを使用できるノードの割合 r_{upper} は，

$$r_{upper} = \frac{\sum_{l=1}^{l_{max}-1} r_u(l)}{l_{max} - 1}, \quad (5.11)$$

である．

証明 SRT において各ノードに割り当てられる上位レベルの数は 1 つなので，全てのノードで上位レベルが重複することはない．したがって，全てのレベルに対し定理 9 を考えれば良いので，

$$r_{upper} = \frac{\sum_{l=1}^{l_{max}-1} r_u(l)}{l_{max} - 1}, \quad (5.12)$$

となる． □

2D-SRT の場合，1D-SRT をシフトしながら積み上げる構成のため条件を満たすノード数が若干異なる．ここでは，簡単化のため上位レベルリンクを使用できる確率はどのノードも変わらないものとする．したがって，定理 9 が利用でき，各ノードで転送可能な方向は一方なので，2D-SRT での同次元迂回ルーティングを用いた際の自由度は

$$\epsilon = \frac{1}{8}(1 + r_{upper}), \quad (5.13)$$

となる．

5.2.2 Dimension reversal routing の自由度

Dimension reversal routing が可能な場合，基本レベルリンクと同次元迂回ルーティングによる各方向への上位レベルリンクの他に y 方向へのリンクを使用することが出来る．

したがって、あるノードにおいて y 方向への基本レベルリンクを使用できる確率を r_{y_0} 、レベル l リンクを使用できる確率を $r_{y_{upper}}$ とすると、Dimension reversal routing の自由度は

$$\epsilon = \frac{1}{8}(1 + r_{upper} + r_{y_0} + r_{y_{upper}}) \quad (5.14)$$

と与えることができる。

以下、 r_{y_0} 、 $r_{y_{upper}}$ の導出について述べる。

Dimension reversal routing による y 方向へのルーティングは全てのノードで行なえるわけではなく、カレントノードとそのレベル、そして、目的ノードとの位置関係により決まってくる。

2D-SRT 上のあるノードで Dimension reversal routing により y 方向のレベル l リンクを使用できる確率について定理 10 がいえる。

なお、2D-SRT は、1D-SRT をシフトしながら積み重ねた構造のため、ノードの絶対的位置（基本トラス上での位置）とそのレベルとが一對一に対応しない。ここでは、簡単化のため、ノード $(0,0)$ で y 方向に形成された 1D-SRT について考える。

定理 10 2D-SRT 上のあるノードで Dimension reversal routing により y 方向へのレベル l リンクを使用できる確率 $r_y(l)$ は

$$r_y(l) = \frac{N-2}{N-1} \frac{2^{l_{max}-1} - 2^l}{N} \quad (5.15)$$

である。

証明 2D-SRT のノード $(0,0)$ で y 方向に形成される 1D-SRT の任意の y アドレスは $2^l(i-1) + 2^{l-1}$ で与えられる。このとき、パケットが正方向に転送可能となる条件は、 $R = N$ とすると、

$$2^l(i-1) + 2^{l-1} < \frac{2^{l_{max}} - 1}{2} - 2^{l-1} \quad (5.16)$$

を満たす時である。よって、

$$\begin{aligned} 2^l(i-1) + 2^{l-1} &< \frac{2^{l_{max}} - 1}{2} - 2^{l-1}, \\ 2^l(i-1) &< \frac{2^{l_{max}} - 1}{2} - 2^l, \\ i-1 &< \frac{2^{l_{max}} - 1}{2^l} - 1, \\ i &< \frac{2^{l_{max}} - 1}{2^{l+1}}, \end{aligned}$$

となり、条件を満たす最大の整数は $i = 2^{l_{max}-l-1} - 1$ である。レベル l ノードは全部で $\frac{2^{l_{max}}}{2^l}$ だけあるので、レベル l リンクを使用できる確率は、

$$(2^{l_{max}-l-1} - 1) \frac{2^l}{2^{l_{max}}} = \frac{2^{l_{max}-1} - 2^l}{N}, \quad (5.17)$$

となる．さらに，あるノードに対する目的ノード数は $N^2 - 1$ だけ存在し， $c_x = d_x$ となる $(N - 1)$ 個のパターン以外は Dimension reversal routing により y 方向へルーティングする可能性がある．したがって，あるパケットの y 方向への転送の確率は

$$\frac{N^2 - 1 - N - 1}{N^2 - 1} = \frac{N - 2}{N - 1} \quad (5.18)$$

となる． y 方向へルーティングが可能となるのは式 (5.17)，式 (5.18) 双方を満たす場合のみであるので，あるノードで Dimension reversal routing により y 方向へのレベル l リンクを使用できる確率 $r_y(l)$ は

$$r_y(l) = \frac{N - 2}{N - 1} \frac{2^{l_{max}-1} - 2^l}{N}, \quad (5.19)$$

である．

次に負方向に関する議論であるが，同次元迂回ルーティングの時と同様，条件を満たす領域が正方向と独立なので，全く同じ議論が可能である．加えて，Dimension reversal routing で選択可能となる y 方向は正か負かのどちらか一方なので，どちらが選択されるかは等確率である．したがって，全体での確率も式 (5.19) となる．以上より，2D-SRT 上のあるノードで Dimension reversal routing により y 方向へのレベル l リンクを使用できる確率 $r_y(l)$ は

$$r_y(l) = \frac{N - 2}{N - 1} \frac{2^{l_{max}-1} - 2^l}{N}$$

である．

□

定理 10 より， $r_{y_0} = r_y(0)$ である．さらに， $r_{y_{upper}}$ も導くことができ，次のようにまとめることができる．

系 8 y 方向の上位レベルリンクを使用できる確率は $r_{y_{upper}}$ は

$$r_{y_{upper}} = \frac{\sum_{l=1}^{l_{max}-1} r_y(l)}{l_{max} - 1} \quad (5.20)$$

である．

証明 系 7 同様，SRT では全てのノードで上位レベルが重複することはない．したがって，全てのレベルに対し定理 10 を考えれば良く

$$r_{y_{upper}} = \frac{\sum_{l=1}^{l_{max}-1} r_y(l)}{l_{max} - 1} \quad (5.21)$$

である .

□

以上より , Dimension reversal routing の自由度をまとめると次のようになる .

$$\left\{ \begin{array}{l} \epsilon = \frac{1}{8}(1 + r_{upper} + r_{y_0} + r_{y_{upper}}) \\ r_{upper} = \frac{\sum_{l=1}^{l_{max}-1} r_u(l)}{l_{max}-1} \\ r_{y_0} = r_y(0) \\ r_{y_{upper}} = \frac{\sum_{l=1}^{l_{max}-1} r_y(l)}{l_{max}-1} \\ r_u(l) = \frac{N-2^l}{N-1} + \frac{2^l-1}{N-1} \frac{2^{l_{max}-l-1}-1}{2^{l_{max}-l}} \\ r_y(l) = \frac{N-2}{N-1} \frac{2^{l_{max}-1}-2^l}{N} \end{array} \right. \quad (5.22)$$

図 5.1 に 1D-SRT のノードサイズと自由度の関係を示す . ネットワークサイズが大きくなるにつれ 同次元迂回ルーティングを行なうことが可能な領域の比率が増していることが分かる . また , 再帰ルーティングのみの場合はネットワークサイズに関わらず , $\epsilon = 0.25$ で一定であり , 同次元迂回ルーティングの経路選択の自由度は十分に高いと言える . 次に , 2D-SRT のノードサイズと自由度の関係を図 5.2 に示す . 図 5.2 では ADP1 は同次元迂回ルーティングを , ADP2 は Dimension reversal routing を表す . 図 5.2 より , Dimension reversal routing のほうが同次元迂回ルーティングに比べ自由度が高いことが分かる . なお , 再帰ルーティングのみの場合はネットワークサイズに関わらず , $\epsilon = 0.125$ である . 一次元 , 二次元共にネットワークサイズが大きいくほど自由度が増すため , 本論文で提案する手法は大規模向きであると考えられる .

5.3 シミュレーションによる耐故障性の評価

5.2 節では , 経路選択の自由度を理論的に導出し , 耐故障性について考察を行なった . 本節では , 実際にネットワーク上のリンクがいくつか故障しているものとして , 故障率とスループットの関係をシミュレーションにより解析した . また , 故障はリンクに対しのみ発生させた . スループットは理想的な無衝突転送の場合を 1 として正規化し , あるノードが 1 クロックで受信できたフリットの数の平均値で表すものとする . この指標ではネットワーク全体の混雑状況を測ることができる .

シミュレーションの概要を表 5.1 に示す .

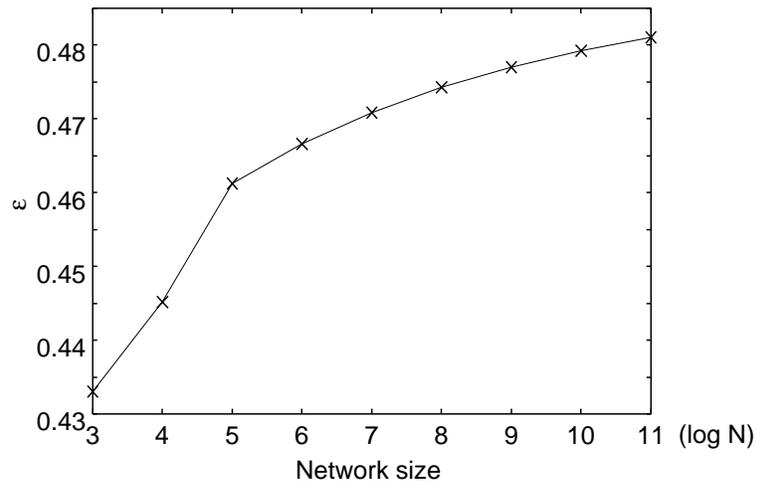


図 5.1: 1D-SRT の自由度の変化

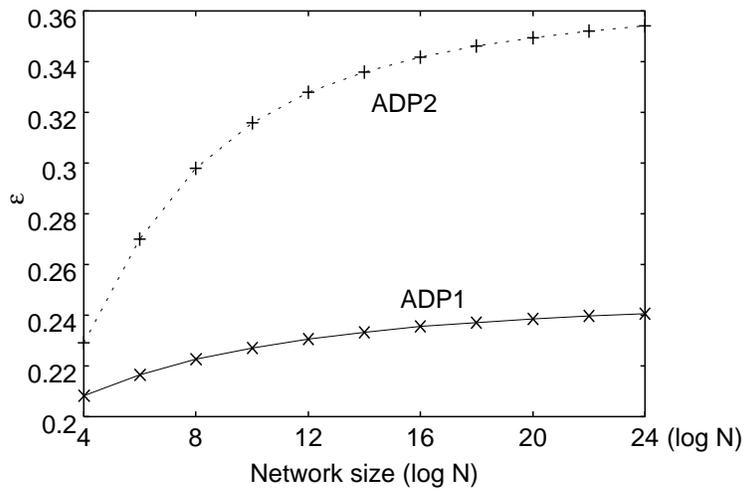


図 5.2: 2D-SRT の自由度の変化

表 5.1: シミュレーションの概要 (耐故障性)

トポロジ	Short-Spna 型 1-Shift 2D-SRT
サイズ	256
パケット長	16flit
仮想チャネル数	2
転送パターン	ランダム転送
故障パターン	ランダム
故障率	0 % ~ 20 %
パケット発生確率	0.25 (flit/clock)
シミュレーション時間	10000 clock
評価	故障率 - スループット

5.3.1 シミュレーション結果と考察

図 5.3 に仮想チャネル数が 2 , メッセージ発生確率が 0.25flit/clock 時の故障率とスループットの関係を示す . 図 5.3 で , DLF は従来の再帰ルーティング , ADP1 は同次元迂回ルーティングのみによる適応型ルーティング , ADP2 は同次元迂回ルーティングと Dimension reversal routing による適応型ルーティングを表す . 図 5.3 より , 明らかに適応化の有効性が確認できる . ルーティング手法が再帰ルーティングのみの場合 , 経路選択の自由度はなく , ただ一つの方向しか転送することが許されない . そのため急激な性能低下を招いている . 一方 , 適応型ルーティングでは同次元迂回ルーティングあるいは Dimension reversal routing といった手法により経路選択の自由度がある為 , 急激な性能低下を防ぐことができる . しかし , 適応化手法の違いによる性能差はほとんど見られなかった . これは , 4.5.2 節と同じ理由によるものであると考えられる .

また , 図 5.4 に仮想チャネル数が 4 , メッセージ発生確率が 0.25flit/clock 場合の故障率とスループットの関係を示す . 図 5.4 での記号は図 5.3 でのそれと同じである . 図 5.4 より , 再帰ルーティング (DLF) のみでは仮想チャネル数が 2 の場合に比べ性能低下は若干緩やかであるが , 数パーセントのリンク故障でネットワークが機能しなくなることが分かる . 一方 , 適応型ルーティングでは 2 つの手法に仮想チャネルの数による差は見られなかった .

以上より , 本論文で提案した同次元迂回ルーティングあるいは Dimension reversal routing によるデッドロックフリー適応型ルーティングは , 再帰ルーティングのみに比べ十分な耐故障性を有していることが分かった . 提案手法は仮想チャネルを新たに付加する必要がないにもかかわらずネットワークの急激な性能低下を防ぐことができる .

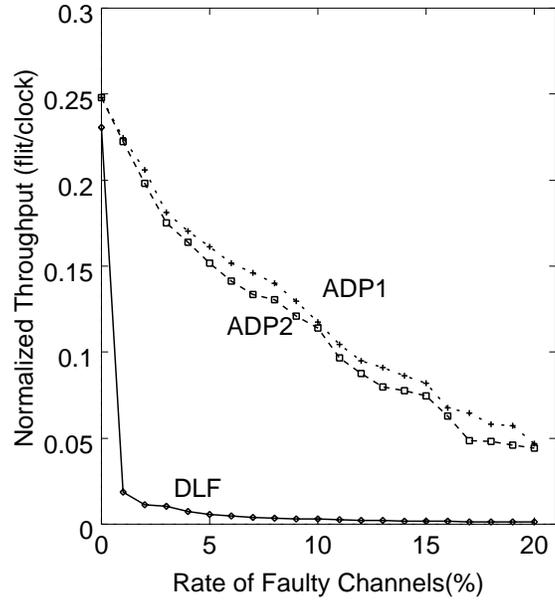


図 5.3: 故障率とスループットの関係 (仮想チャンネル 2)

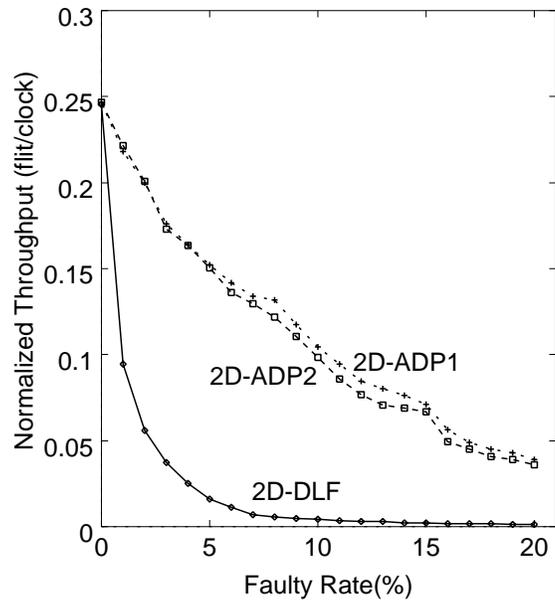


図 5.4: 故障率とスループットの関係 (仮想チャンネル 4)

5.4 まとめ

本章では、提案する同次元迂回ルーティングと Dimension reversal routing の耐故障性について理論的及び実験的な側面から議論した。1つは経路選択の自由度の解析である。解析の結果、提案手法は従来の再帰ルーティングに比べ高い経路選択性を有していることが分かった。また、提案した同次元迂回ルーティング、Dimension reversal routing 共にネットワークの規模が大きいほど、自由度が高くなることを示した。2つ目は故障リンクがある場合の動的性能評価である。シミュレーションは 2D-SRT に対し行なった。その結果、従来の再帰ルーティングでは数パーセントの故障でネットワークがほとんど稼働しなくなるが、提案手法では急激な性能低下が生じないことが分かった。したがって、同次元迂回ルーティングと Dimension reversal routing は十分な耐故障性を有しているといえる。

第 6 章

結言

高性能な超並列計算機を実現する上でノード間の結合方式と通信方式は性能に大きく影響するため、従来より、その性能改善のために様々な結合方式やルーティング手法が提案されてきた。再帰的相互結合網は従来の格子型結合網を含み、高い通信性能を有している。しかし、再帰網におけるルーティング手法はデッドロック回避、適応性、耐故障性といった重要な特性を有していなかった。

本論文では再帰的相互結合網のデッドロックフリーな適応型ルーティング手法をこれまでとは全くことなるアプローチにより導出し、SRTにおいて、シミュレーションにより動的な性能特性を評価した。

第三章では、まず、始めに、一次元再帰的結合網に対し、従来からのデッドロック回避の手法である Monotonic order routing を適用することでデッドロックフリーが保証されることを示した。次に、一次元再帰網の適応化手法として同次元迂回ルーティングを提案した。同次元迂回ルーティングでは同次元上で 180 度ターンすることが可能であり、従来のデッドロックフリールーティングとの併用が可能である。また、仮想チャネルを新たに付加する必要がないなどの特徴がある。さらに、同次元迂回ルーティングの動的な性能評価をシミュレーションにより行なった結果、大幅な性能の向上が見られた。

第四章では、二次元再帰的結合網において Dimension reversal routing が可能な領域を導出した。そのことにより従来からのデッドロックフリールーティングと同次元迂回ルーティング、Dimension reversal routing との併用が可能な適応性のあるルーティングが可能となった。また、同次元迂回ルーティング同様、新たな仮想チャネルを付加する必要がないという特徴を持っている。また、シミュレーションによる動的な性能評価では、従来法に比べ性能の向上がみられ、ハイパーキューブ網との比較では、ハイパーキューブに比べノード次数が小さいにもかかわらず、高い性能を得ることができた。適応化における課題である混雑解消の役割を果たしたえる。

第五章では、本論文で提案した適応化手法の耐故障性について 2 つの側面から議論した。あるノードでどの程度の経路選択に自由度があるかという理論的な評価では、適応的に

選択できる領域に制限があるものの、従来型に比べ高い自由度があることが分かった。また、故障を含んだ SRT に対する動的性能評価では、故障がない時と比べてかなり高い性能向上がみられ、同次元迂回ルーティング, Dimension reversal routing による適応化手法の有効性を確認できた。

今後の課題としては、新たに仮想チャネルを付加することが可能な場合についての議論が挙げられる。本論文で提案した手法は、デッドロックフリーが保証できる必要最小限の仮想チャネルで適応性を獲得する為の手法である。したがって、仮想チャネルを新たに付加する必要がなく、仮想チャネル数がそれより多い場合は、空いているチャネルを自由に使用することができる。しかしながら、適応ルーティングが可能な領域が制限されている為、仮想チャネルを増やしても、経路の選択性を挙げることはできない。そこで、新たに付加された分を別のネットワークと考え新たにチャネル番号を割り当てることで、適応化可能な領域を広げることが可能であると考えられる。

次に、同次元迂回ルーティング, Dimension reversal routing が可能となる領域を導出した方法による、逆方向へのルーティングが可能な領域の導出が考えられる。

また、本論文では動的性能評価の対象として SRT を扱ったが、他の再帰的結合網に対しても提案手法を適用し性能評価を行なう必要がある。

謝辞

本研究を進めるにあたり終始熱心な御指導，御教授いただいた北陸先端科学技術大学院大学 堀口 進 教授，阿部 亨 助教授に感謝致します．

サブテーマで熱心に御指導いただいた松澤 照男教授には感謝致します．

日頃より有意義な御教示，検討，議論を下さった山森 一人 助手，林 亮子 助手，井口 寧 助手に感謝致します．

また，日頃よりお世話になった堀口・阿部研究室の皆様に厚く御礼申し上げます．

参考文献

- [1] R. Duncan, "A Survey of Parallel Computer Architectures," IEEE Computer, vol.23, no.2, pp.5-16, 1990.
- [2] C.-C. Lin and V.K. Prasanna, "A Routing Algorithm for PEC Networks, Proceeding of the Fourth Symposium on the Frontiers of Massively Parallel Computation," pp.170-177, 1992.
- [3] C.-C. Lin and V.K. Prasanna, "Bounds on the diameter of one-dimentional PEC networks, Journal of Parallel and Distributed Computing," vol.29, no.1, pp.1-16, 1995.
- [4] Y. Inoguchi and S. Horiguchi, "Shifted Recursive Torus Network for Mesh-Oriented Interconnections," Proc. 31st Conference on Information Sciences and Systems, Baltimore, Mar. 1997.
- [5] Y. Inoguchi and S. Horiguchi, "Shifted Recursive Torus Interconnection for High Performance Computing," IEEE High Performance Computing in Asia Conference, Seoul, pp. 61-66, Apr. 1997.
- [6] 楊 愚魯, 天野 英晴, 柴村 英智, 末吉 敏則, "超並列計算機に向き結合網:RDT," 信学論 (D-I) vol. J78-D-I no.2 pp.118-128, 1995.
- [7] 天野 英晴, 楊 愚魯, "超並列計算機に向き結合網 RDT 上でのデッドロックフリールーティング" 信学技報 CPSY94-56 pp.109-116, Sep, 1994.
- [8] W.W. Kirkman and D. Quammen, "Packed Exporntial Connections - A Hierarchy of 2D-Meshes," Proceeding of the Fifth International Parallel Processing Symposium, pp.464-470, Apl. 1991.
- [9] W.J.Dally and C.L.Seitz, "Deadlock-Free message routing in multiprocessor interconnection networks," IEEE Trans. on Computers , Vol.C-36, pp.547-553,May 1987

- [10] W.J.Dally and H.Aoki, "Deadlock-Free Adaptive Routing in Multicomputer Networks Using Virtual Channels," IEEE Trans. on Parallel and Distributed Systems, Vol.4, No.4 1993. pp.466-475.
- [11] W.J.Dally, "Virtual Channel Flow Control," IEEE Trans. Parallel and Distributed Systems, Vol.3, No.2, Mar.1992. pp.194-205.
- [12] L.M. Ni , L.P. McKinley , "A Survey of Wormhole Routing Techniques in Direct Networks," IEEE Trans. on Computers, 1993.
- [13] J.Duato , "A New Theory of Deadlock-Free Adaptive Routing in Wormhole Networks," IEEE Trans. on Parallel and Distributed System, Vol.4, No.12, December 1993
- [14] J.Duato , "A Necessary and Sufficient Condition for Deadlock-Free Adaptive Routing in Wormhole Networks," IEEE Trans. on Parallel and Distributed System, Vol.6, No.10, December 1995
- [15] Glass,C.J , Ni, L.M. , "Maximally Fully Adaptive Routing in 2D Meshes," Proceedings of ISCA92, pp.278-287(1992)
- [16] 田辺, 原田, 朴"ハイパクロスバ網における迂回ルーティング・アルゴリズムに関する研究", 情報処理学会第 54 回全国大会, 1997.
- [17] 舟橋 啓, 埴 敏博, 工藤 知宏, 天野 英晴, "相互結合網 RDT における adaptive routing." CPSY-96-75, pp.47-54 Oct. 1996
- [18] V.K. Jain, T. Ghirmai and S. Horiguchi, "TESH: A New Hierarchical Interconnection Network for Massively Parallel Computing", IEICE Trans. Information and Systems, Vol.E80-D, No.9, pp.837-846 (1997 Sep)
- [19] 川井 雅之, 井口 寧, 堀口 進, "再帰トールス相互結合網 SRT におけるデッドロックフリー・ルーティング", 平成 10 年度電気関係学会北陸支部連合大会論文集, p.264 (Oct.1998)
- [20] 天野英晴, "並列コンピュータ" 昭晃堂,1997
- [21] William J. Dally and Hiromichi Aoki, "Deadlock-Free Adaptive Routing in Multicomputer Networks Using Virtual Channels", IEEE Trans. Parallel Distrib. Syst., vol.4, No.4, pp.466-475, Apr 1993

研究業績

- [1] 川井 雅之，井口 寧，堀口 進
“再帰トラス相互結合網 SRT におけるデッドロックフリー・ルーティング”，
平成 10 年度電気関係学会北陸支部連合大会論文集，p.264 (Oct.1998)
- [2] 川井雅之，井口寧，堀口進
“超並列計算機向き相互結合網 SRT のデッドロックフリー・ルーティング”
情処学論 (投稿中)
- [3] 川井雅之，井口寧，堀口進
“超並列計算機向き相互結合網 SRT における適応型ルーティング”，
第 6 回 HOKKE'99，Mar.1999 (発表予定)