

相互結合網 RSOT のルーチング方式

2L-6

秋山知之 田中英彦
東京大学工学系研究科

1 はじめに

相互結合網 RSOT (Recursive Orthogonal Torus) は次数 8、65536 ノード構成で平均距離 7.97 という特長を持つ。また、木構造のマルチキャストが可能、メッシュを内蔵するなどの特長も有する [1]。

本稿ではデッドロックを回避するためのルーチングや仮想チャネルの構成を示す。一般に、デッドロック回避の目的で仮想チャネルを設けると実効バッファサイズが小さくなるのが問題であるが、その問題を解決するために用いた適応フロー制御について説明する。

2 RSOT のトポロジ

RSOT は図 1(a) に示すレベル 0~4 の 5 種類の二次元トーラス網から構成される。各ノードはレベル 0 トーラスとレベル 1 から 4 のうちの一つのトーラスに接続され、次数は 8 となる。ノード配列を図 1(b) に示す。

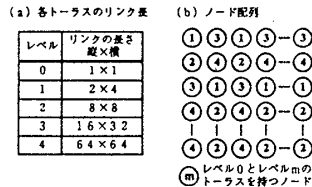


図 1: RSOT のトポロジ

3 RSOT のルーチング

k-ary n-cube は、使用する物理チャネルの方向 (次元) に順序を設けることによってデッドロックフリーのルーチングが実現できる。RSOT では、k-ary n-cube の次元はレベルとトーラス上の方向 (次元) に対応する。RSOT は、ダイレクトリベースキャッシュコヒーレンスプロトコルを支援することを主目的とする木構造のマルチキャストの機能を備えるが、そのとき使用するレベルの順序は上位レベルから下位レベルであるのに対して、マルチキャストされたコマンドパケットに対する Ack パケットは逆方向のレベルの使用順序をとる。これによるデッドロックを回避するためには 2 つの仮想チャネルを備える必要がある。1 対 1 通信に関しては、基本的に最適ルーチング (平均距離の最も小さいルーチング) を用いるが、デッドロックフリーであるためには、レベルの使用順序についての制限があるため、すべての始点-終点間で最適ルーチングが可能ではない。しかし、最適ルーチングが不可能となる場合は全体の 13 % にまで抑えられることが分かっており (4.3 節)、この場合平均距離の増加は 0.13 程度に抑えられる (16384 ノード構成時)。

4 物理チャネルの分類と仮想チャネルの設定法

物理チャネル、仮想チャネルの分類を図 2 に示す。

4.1 物理チャネルの分類

縦方向物理チャネルを V (vertical)、横方向物理チャネルを H (horizontal) と分類する。レベル 0 の物理チャネルについてはさらに、その物理チャネルによって低レベルのノードから高レベルのノードに移動するときは R (rise)、その逆のときは F (fall) のように分類する。レベル 0 の物理チャネルについて、さらに細かく分類するとき、その物理チャネルの始点、終点のレベル L_s, L_e を用いた (L_s, L_e) により区別する。

4.2 仮想チャネルの設定

レベル 1~4 の物理チャネルには u_n, u_c, d_n, d_c の 4 つの仮想チャネルを設定する。レベル 0 の物理チャネルにはその 4 つとさらに t_n, t_c の 2 つの仮想チャネルを設定する。

The Routing Scheme of RSOT
Tomoyuki AKIYAMA, Hidehiko TANAKA
Faculty of Engineering, University of Tokyo

物理チャネル		仮想チャネル	
レベル	方向	仮想チャネル	仮想チャネル
1			
2	H	u	n
3	V	d	c
4			

物理チャネル				仮想チャネル	
レベル	方向	レベル移動の向き	始点・終点のレベル	仮想チャネル	仮想チャネル
0	H	R	(13) (24)	u	n
		F	(31) (42)		
	V	R	(12) (23) (34) (14)		
		F	(21) (32) (43) (41)		

図 2: 物理チャネル、仮想チャネルの分類

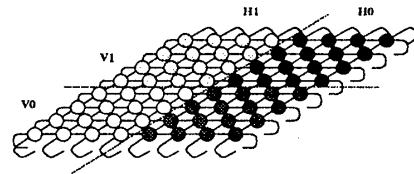


図 3: 二次元トーラス網の領域分割

n (not crossed), c (crossed) は二次元トーラス網中の横 [縦] 方向物理チャネルのみによって構成されるループによって生じるデッドロックを回避するためのもので、次のように使用される。あらかじめ、図 3 の様に各二次元トーラスに対して領域分割を行なう。パケットがあるレベルにおける二次元トーラスの横 [縦] 方向の移動を始めた時は、必ず n の仮想チャネルを用いるが、領域 H1 から H0 [V1 から V0] に移る時に、仮想チャネルを c に切り替える (図 6(a))。こうすることにより、特定レベルの特定方向内でのデッドロックは回避できる。次に特定レベル内において、使用する物理チャネルの方向に $H \rightarrow V \rightarrow H$ のようなループが生じないことを保証する必要がある。そのためには、あるレベル内では $H \rightarrow V$ あるいは $V \rightarrow H$ の順序で使用するという条件を満たすルーチングを行なえば良い。最後に、使用するレベルに例えば $0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 1$ のようなループが生じないことを保証する必要がある。そのためには、レベルを下位から上位の順に用いるなど、ある特定の順序に従ったルーチングを行なえば良い。厳密には、RSOT ではレベル 1 の次にレベル 2 を用いるということは出来ず、レベル 1 の次にレベル 0 によってレベル 2 を持つノードへ移動した後でレベル 2 を用いる必要がある。そのため、レベルを下位から上位の順に用いる場合を考えると、レベル間依存関係は $0 \rightarrow 1 \rightarrow 0 \rightarrow 2 \rightarrow 0 \rightarrow 3 \rightarrow 0 \rightarrow 4$ のようになる。このままでは依存関係にループが生じるので、レベル 0 をレベル間移動に用いるときは専用の仮想チャネル t (transit) を用いる。0t (レベル 0 の物理チャネル中の仮想チャネル t 、以下チャネルをこの様に記述する) の間の依存関係によってループが生じないことも保証する必要がある。前記の場合では、0t はそれぞれ 0(12)t, 0(23)t, 0(34)t, 0(13)t, 0(24)t, 0(14)t であり、すべて異なるため、ループは生じない。RSOT は、3 節に述べたように、少なくとも上位から下位、下位から上位の 2 つの順序でレベルを使用できる必要がある。そのためには上位から下位と下位から上位の場合で仮想チャネルを使い分ける必要がある。レベルを上位から下位の順で用いる場合仮想チャネル d (down) を使い、逆の場合は u (up) を使うことにする。

4.3 仮想チャネル間の依存関係

4.2 節で述べた仮想チャネル間の依存関係を図 4 に示す。(a) は基本形を示しており、この場合ルーチングはレベルの使用順序において、上位→下位、下位→上位の 2 つの順序に限定される。そのため、このルーチングに従った場合、平均距離は 0.347 増加する (16384 ノード構成時)。

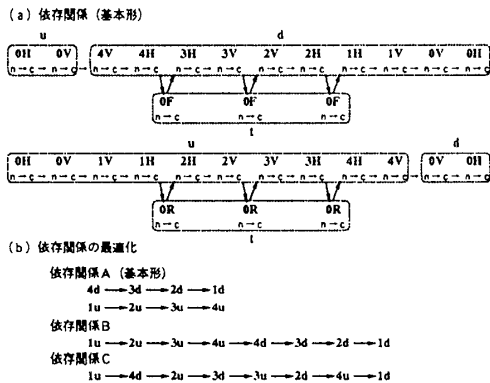


図 4: 仮想チャネル間の依存関係

ド時)。(b)の依存関係 A は (a) を簡略化して記述したもので、レベル 0 や、n、c の区別、H、V の区別は省略してある。(b)の依存関係 B、C のような依存関係を持たせ、この範囲でルーチングを行なうことにより、平均距離の増加を A より小さく抑えることができる。依存関係 B、C を用いた時、レベル間移動に用いる仮想チャネル 0 と間の依存関係にループが生じないことが保証される (依存関係 C に対する証明を図 5 に示す)。平均距離の増加は依存関係 B、C でそれぞれ 0.141、0.131 (16384 ノード時) となる。そこで依存関係 C を用いることにする。このとき平均距離は 16384 ノード構成で 6.95 となる。

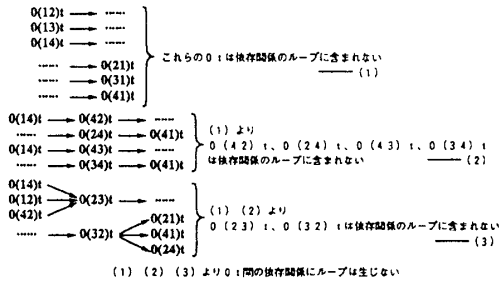


図 5: 0t 間の依存関係にループが生じないことの証明

5 適応フロー制御による実効バッファサイズ低下の抑制

デッドロックを回避するために複数の仮想チャネルを設けると、一般的に、仮想チャネルの使用頻度には偏りがあるため、実効的なバッファサイズが小さくなるのが問題である。仮想チャネルのサイズを使用頻度に合わせて選択する方法は、アクセスパターンによって使用頻度が異なる場合が多く、最適なサイズを選択するのが難しい。RSOT では次に示す適応的なフロー制御を行なうことにより、実効バッファサイズの減少を抑制する。ただし、Virtual Cut-Through を用いたパケット単位のフロー制御を前提としており、ここではフロー制御は仮想チャネルの割り当てという意味で用いる。

5.1 トーラス網における適応フロー制御

(a) 基本方式



(b) 適応フロー制御

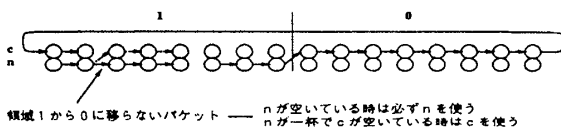


図 6: トーラス網における適応フロー制御

図 6 は図 3 の横 [縦] 方向の断面を表している。通常 (a) に示すようなフロー制御を行なうことによって、二次元トーラス網中の横 [縦] 方向物理チャネルのみによって構成されるループにより生じるデッ

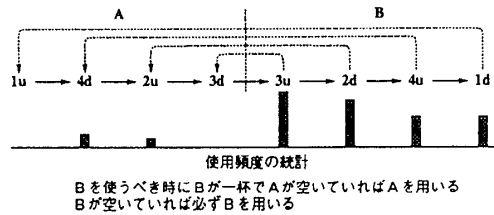


図 7: レベル間依存関係における適応フロー制御

ドロックを回避できることは 4.2 節で述べたが、このような方式をとると、仮想チャネル c の使用頻度が n よりも小さいため、c が最大限利用できず、実効バッファサイズが小さくなるのが問題である。(b) に示す適応フロー制御を行なうことにより、c、n をともに最大限利用することが可能となる。この方式では、領域 1 から 0 に移らないパケットは、もし n が空いていれば必ず n を使うが、n が一杯で c が空いている時は c を使う。一度 c を使っても、次のホップの時に n が空いていれば必ず n を用いる。n が一杯のために c に移されたパケットは次のホップでは n と c の双方に行く可能性があり、一見 c から n に依存関係があるように見えるが、そのパケットが進むための条件は、前方の n あるいは c のどちらかが空いていることであり、前方の c が空くことが保証されていれば、c から n への依存関係は無い。また、n が一杯の時 c へ移すことのできるパケットは、領域 1 から 0 に移らないパケットに限定しているため、仮想チャネル c の間で依存関係のループが生じることもない。したがって、デッドロックフリーであることが分かる。

5.2 レベル間依存関係における適応フロー制御

図 7 は図 4 (b) の依存関係 C を満足する 1 対 1 通信のルーチングを行なった場合の、各仮想チャネルの使用頻度を示している。図を見ると明らかにグループ A (1u、4d、2u、3d) よりもグループ B (3u、2d、4u、1d) の方が使用頻度が高い。あるノード対に対して、最適ルーチング (ホップ数が最小のルーチング) は一般に複数存在するが、その中から B の使用頻度が高いものを選択したためである。このような選択をした理由は後述する。このとき、次のような適応フロー制御を行なうことにより、バッファを最大限利用することが可能となる。すなわち、パケットが B に属するあるレベルの仮想チャネルを使用すべき時にその仮想チャネルが一杯で、A に属するそのレベルの仮想チャネルが空いている時は、A の方を用いる。B が空いている時は必ず B を用いるようにする。B が一杯であるために A に転送したパケットは、次のホップで B に空きがある場合は必ず B に戻す。このような操作を行なっても、B から A への依存関係は生じないため、デッドロックは起こらない。

次に頻度分布を B に偏らせた理由を述べる。B に使用頻度を偏らせるという操作を行なわなかった場合、B に属する仮想チャネルよりも A に属する仮想チャネルの方が頻度が大きいう状況が生じ、またその逆の状況も生じる (レベルによって異なる)。前者の場合、A を使うべき時に A が一杯で B が空いていれば B を用いるというような適応フロー制御も可能だが、常に B を用いて良い訳ではなく、次のような条件を満たしているパケットについてのみ、B を用いることができる。すなわち、例えば 4d、2u の順に進むべきとき、4d が一杯であるため 4u を用いるということを行なうと、4u から 2u に依存関係が生じるため、デッドロックが起こる可能性が生じる。したがって、4u を使うためには、その後に 4u より左の仮想チャネルを用いることがあるかどうかを調べなければならない。このようなコストをかけないために、あらかじめ B の使用頻度が高くなるようにルーチングテーブルを設定し、A が一杯のとき B が空ならば B を用いるというような操作を必要としない状況を作り出す。

6 おわりに

本稿では RSOT においてデッドロックを回避するためのルーチング、仮想チャネルの構成を示した。デッドロックを回避するために仮想チャネルを設けると、実効バッファサイズが小さくなるが問題であるが、その問題を解決する適応フロー制御について述べた。

参考文献

- [1] 秋山知之, 小池帆平, 田中英彦. 分散共有メモリ型並列計算機におけるディレクトリ方式を相互結合網について. 信学技報 CPSY94-49, 65-72.