

動的な情報を用いたルーティングアルゴリズム Cross-Lineの検討

西谷 雅史[†] 絵面 聡[†] 横田 隆史[†]
古川 文人^{††} 大津 金光[†] 馬場 敬信[†]

相互結合網のメッセージ転送方式は、並列計算機の性能に大きな影響を与える。ルーティング方式の検討は今まで様々行われてきたが、プロセッサの性能も、通信のバンド幅も、LSIの集積度も向上した現在の相互結合網を取り巻く状況を考えると、それらを有効に活用できる新しいルーティング方式を考える必要がある。我々は、相互結合網向けの新たなルーティングアルゴリズムとして、適応ルーティング Cross-Line を検討する。本アルゴリズムは、バーチャルチャネル (Virtual Channel) の状態を示す動的な情報を用いてルーティングを行うことを特徴とする。我々はこのアルゴリズム実現のために、相互結合網シミュレータ Chimera を作成し、Cross-Line についての詳細な検討を行った。本稿では、Cross-Line 実現に向けての詳細な検討結果について報告する。

Preliminary Research of a Novel Routing Algorithm Cross-Line Using Dynamic Information

MASASHI NISHITANI,[†] SATOSHI EZURA,[†] TAKASHI YOKOTA,[†]
FUMIHITO FURUKAWA,^{††} KANEMITSU OOTSU[†] and TAKANOBU BABA[†]

Routing methods for interconnection networks affect the performance of parallel computers considerably. Thus they have been studied and many methods are proposed and implemented. Advances in VLSI integration, network bandwidth, and processor performance, however, drive us to consider the novel routing methods. We are discussing a novel adaptive routing algorithm, called "Cross-Line". This routing algorithm has a unique feature using dynamic information. We have developed an interconnection networks simulation framework called "Chimera" so that we can evaluate the Cross-Line method quantitatively.

1. はじめに

相互結合網のメッセージ転送方式は、並列計算機の性能に大きな影響を与える。ルーティング方式の検討は今まで様々行われてきたが、プロセッサの性能、バンド幅、および LSI 集積度が大幅に向上した現在の状況を考えると、再検討を行う必要がある。既存の方式では、いかにハードウェア資源を抑えつつ性能を上げられるかというところに焦点があった。だが、現在の状況を考えると、今までよりも豊富なハードウェア資源を有効に活用し、バンド幅の向上に並行してより効率的なルーティング方式を模索する必要がある。

我々はこの問題に対し、ネットワークの状態を表す情報としてバーチャルチャネル (Virtual Channel: VC) がデータパケットを受信可能か否かという情報 (VC 情報) を用いて適応ルーティングを行うアルゴリズム "Cross-Line" の実現を考えている。動的な情報を用いることにより、ネットワーク状況に合わせた最適なルーティングを行うことで、ネットワークスループットの向上を促す

と同時に、レイテンシの削減ができると考えている。しかし、動的な情報を用いているがゆえに、Cross-Line の実現に向けてのモデルは不明瞭な要素が多い。情報の扱い方やその量など、実際に動かして検討してみなければ最適なモデルを確定できない。

そこで我々は、Cross-Line の最適なモデルを検討するために、相互結合網シミュレータ Chimera¹⁾ を作成した。Chimera は 2 つの特徴を持っている。1 つは容易なアルゴリズムの導入を可能にするための記述構造であり、もう 1 つはネットワークの構成に関する多数のパラメータを設定できることにより、柔軟なシミュレーションが可能なことである。我々はこの Chimera に、基本的なルーティングアルゴリズムである Dimension-order と Cross-Line を実装した。

Chimera によって Cross-Line の性能とオーバーヘッド、動的な情報の必要量と正確性、トラフィックの分散状況と渋滞状況などを確認し、Cross-Line の最適なモデルを検討した。

2. 相互結合網シミュレータ Chimera

2.1 設計思想

Chimera 開発の目的は、新たに提案するルーティングアルゴリズムの導入、検討、評価を容易に行えるようにすることである。Chimera を開発する上で、ルーティングアルゴリズムの導入と評価を考慮した 2 つの特徴を持た

[†] 宇都宮大学工学部情報工学科
Department of Information Science, Faculty of Engineering,
Utsunomiya University
^{††} 宇都宮大学サテライト・ベンチャー・ビジネス・ラボラトリー
Satellite Venture Business Laboratory, Utsunomiya
University

せ、検討を円滑に行えるように検討支援機能を持たせた。

2.2 特徴

特徴の1つは、新しいルーティングアルゴリズムの導入を容易に可能にする記述構造である。どのようなルーティングアルゴリズムを導入する際も、データパケットの生成、ネットワークトポロジの形成、フロー制御、データパケットの送受信などの部分は必ず用いられる。Chimeraではこれらを基幹部分として記述することにより、新しいルーティングアルゴリズムを導入する際に、基幹部分をアルゴリズムの動作の中でどのように連携させるかを書けばよい。これにより、新しいアルゴリズムの導入が容易に行えるようになっている。

もう1つは、多数のパラメータを設定可能にすることで、柔軟なシミュレーションを可能にしている点である。シミュレーション実行に際し、ネットワークサイズ、フリット数、コミュニケーションパターンなど、多くのパラメータが設定可能である。様々なネットワーク環境でルーティングアルゴリズムを評価でき、シミュレーション条件を合わせることでChimeraに実装していないアルゴリズムとの比較が可能となる。

2.3 支援機能

Chimeraは、ルーティングアルゴリズム検討のための2つの支援機能を備えている。

1つは、定常状態検出支援機能である。シミュレーション開始直後のネットワークではデータパケットの数が少なく、データパケットは転送をブロックされること無く目的地のノードに到着することができる。

シミュレーション開始から定常状態になるまでの過渡状態の影響を排し、定常状態での評価を可能にするための機能が定常状態検出支援機能である。検出方法は、一定のクロック毎に到着したデータパケットの平均レイテンシを計測することによりその様子をグラフ化し、ネットワークの混雑状況が安定する様子を、視覚的に確認することで検出する。

もう1つは、トラフィック可視化機能である(4.2節)。各ノードのVCで、データパケットがブロックされている時間を計測することにより渋滞の広がりを見視化する機能と、各ノードへの入力リンクの使用率を計測することでトラフィックの分散を見視化する機能を備えている。

3. Cross-Line

3.1 設計思想

議論の単純化のために、トポロジが2次元メッシュ構造(2D-Mesh)、2次元トーラス構造(2D-Torus)の場合を例にしてCross-Lineについて解説していく。

Cross-Lineは、図1に示すように、ノードから出るリンクの進行方向に沿った位置にあるVCの状態情報を用いる適応ルーティングである。進行方向の先にあるVCの状態情報に基づいて、最短経路を守ったルーティングを行う。動的な情報を利用することで、ネットワークの状況に合ったトラフィックの分散を行う。これによりネットワークスループットの向上を図るとともに、転送レイテンシの削減を目指している。

以下、Cross-Lineについて詳しく解説する。

3.2 動的な情報

Cross-Lineで用いる動的な情報は、各VCがデータパケットを受信できる空き容量があるかどうかという1ビットの情報が単位となる。受信可能なときは0 (Available) という情報で表され、受信不可能なときは1 (Not Avail-

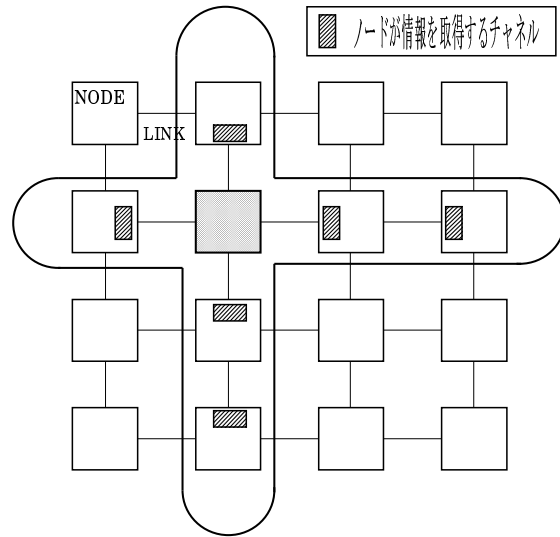


図1 ノードがVC情報をあつめる範囲

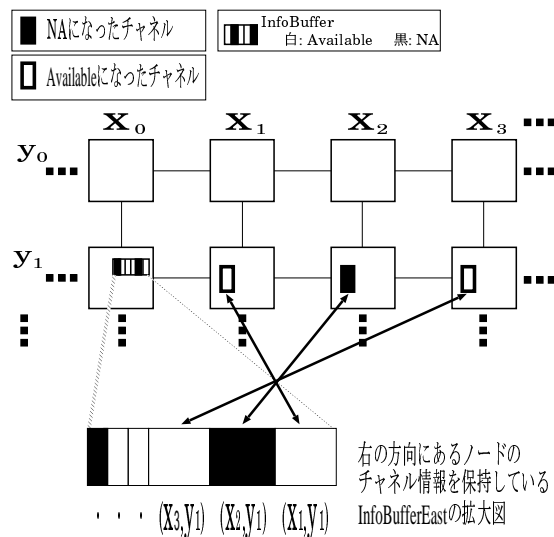


図2 InfoBufferによるVC情報の保持

able:NA)という情報で表される。各ノードの持つ情報は、図1に示すような、そのノードが属する縦横2本の通信経路上に存在するVCの情報である。

3.3 VC情報の保持

他ノードから来たVC情報を保持するために、ノードは各VCごとにInfoBufferを持つ。図2にInfoBufferの様子を示した。図のInfoBufferはInfoBufferEastである。InfoBufferEastは、所属するノードからEast方向にあるノードの、Westバッファに所属するVCの情報を保持する。InfoBufferはレジスタで実装される。最下位ビットは1 hopだけEast方向にあるノードの、最上位ビットは最も離れたノードのWestバッファに属するVCの情報を示している。VC情報の送信元を示す情報を扱わなくても、ビット列で保持することにより自分から何hop離れたノードのVC情報なのかを知ることができる。これにより、Cross-Lineが扱う動的な情報量の低減を実現している。

3.4 VC情報の転送

VC情報の転送には、データパケットと同じ物理リン

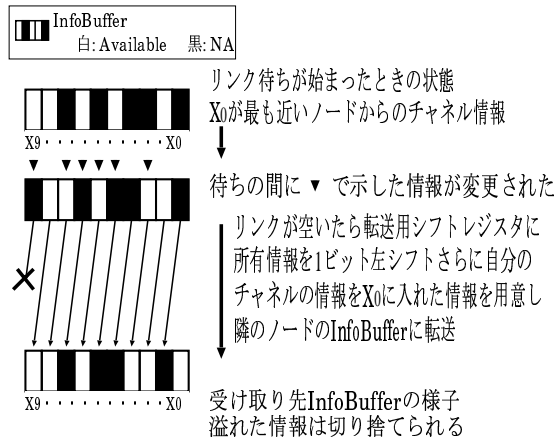


図3 VC情報のビット列とその転送

クを用いる。これによりVC情報が転送のために物理リンクを使用し、データパケットの転送を妨げることが起きる。これがCross-Lineのオーバーヘッドになる。このため、扱う動的な情報を利用するVCの空き情報のみとすることで、少ないオーバーヘッドで動的な情報を利用することを可能にしている。

Cross-Lineで動的な情報を利用する目的は、トラフィックの分散である。トラフィックの分散が行われるなら、必ずしもVC情報をリアルタイムに転送する必要は無い。よって、VC情報の転送よりもデータパケットの転送の優先するように制御する。VC情報を転送する際に、ノード内に同じリンクを使用する予定のデータパケットが存在するときは、VC情報の転送を抑制する。完全にリンクがフリーのときだけVC情報の転送を行う。すなわち、データパケット転送の間隙を縫うかたちでVC情報は転送される。

図3に、VC情報転送の過程を示す。VC情報は各VCごとにInfoBufferに保持される。InfoBufferの内容を転送用のシフトレジスタにコピーする。コピーされた内容を左1ビットシフトし、最下位ビットに自ノードのVC情報を入れて転送する。

VC情報の転送は、InfoBufferの内容に変化があったときにのみ行われる。隣接ノードのVC情報は、転送制御のためのハンドシェイク信号によって知ることができる。このため、ハンドシェイク信号の変化を検出したら、対応するInfoBufferのビットを変更し、VC情報を上流のノードに転送する。上流のノードは、下流のノードから転送されてきたVC情報により、自InfoBufferの内容を変更する。

3.5 アルゴリズム

Cross-Lineは動的な情報によりトラフィックの分散を図り、最短経路を守ったルーティングを行うことで転送の効率化を行う適応ルーティングである。

まず、2D-Meshに関して説明する。2D-Meshでは、デッドロック防止のために図4に示すように1方向につき2D-Meshでは2本のVCを用意する(VC0, 1)。データパケットを生成したノードから目的地がどの方向にあるのかで場合分けを行い、使用するVCを限定する。目的地がNorth-Westにある場合とSouth-Eastにある場合は、VC0を利用する。North-Eastにある場合とSouth-Westにある場合は、VC1を利用する。目的地までの道のりがNorth, East, South, Westの直線的な動きだけに限定される場合はランダムに選択する。

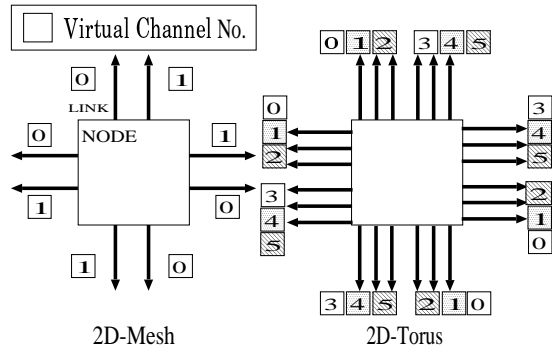


図4 Cross-LineのVC

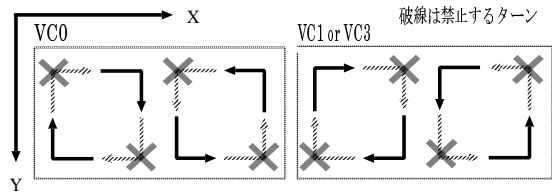


図5 Cross-LineのTurnmodel

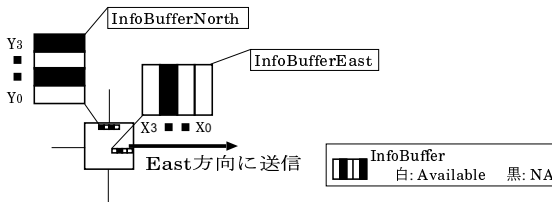


図6 Availableの連続数で決定する例

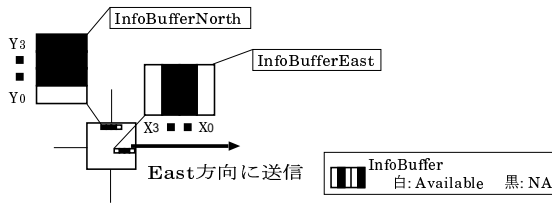


図7 NAの連続数で決定する例

同様に、2D-TorusではRoundTripを考慮して6本のVCを用意する(VC0~VC5)。2D-Meshと同様に、データパケットを生成したノードから目的地がどの方向にあるのかで場合分けを行い、使用するVCを限定する。目的地がNorth-Westにある場合とSouth-Eastにある場合は、VC0を利用する。North-Eastにある場合とSouth-Westにある場合は、VC3を利用する。RoundTripによるデッドロック防止のため、2D-TorusではX次元とY次元でVC番号を統一する。次元によらず、1度目のRoundTripで下位RoundTrip用VCを利用する。汎用VCとしてVC0を利用していたデータパケットはVC1を利用し、VC3を利用していたデータパケットはVC4を利用する。2度目のRoundTripでは上位RoundTripVCを利用する。下位RoundTrip用VCとしてVC1を利用していたデータパケットはVC2を利用し、VC4を利用していたデータパケットはVC5を利用する。

これによりブロックの循環が発生しなくなり、デッドロックは起こらない(図5)。

VCを決めると、次はルーティングを行う。ネットワー

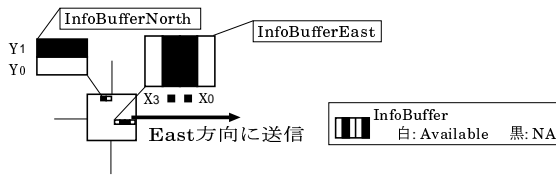


図8 InfoBufferのサイズで決定する例

クの混雑を表すVC情報は3.3, 3.4節で述べた情報の保持と転送の手法により, すでにVCごとに用意したInfoBufferに保持している. データパケットはその情報に基づきルーティングする.

目的地がノードのNorth-East方向に存在する場合を例として, アルゴリズムについて説明する.

一方向への出力がブロックされている場合

最短経路を保証したルーティングを行うため, パケットの進む方向をNorthかEastに限定する. ハンドシェイク制御によりVCへの転送がブロックされている場合, 1 hop先のNorthとEastノードのVCとその方向のリンクの状態を調査する. NorthかEastのどちらか一方の受信VC, 及びリンクが占有されていた場合, 情報によるルーティングは行わず, あいている方のノードにデータパケットを転送する.

両方向への出力が可能な場合

- (1) 両方向InfoBufferを最下位ビットから順に見て, Availableの値が連続している数をカウントする. この数が多い方がより空いていると期待できる. の例を図6に示す. InfoBufferEastのAvailable連続数が2, InfoBufferNorthが1であるから, 連続数の多いEast方向へデータパケットを転送する.
- (2) Available連続数が同じ場合, それ以降の上位ビットでNAの値が連続している数をカウントする. この場合, NA連続数が少ない方が空いていると期待できる. 図7に例を示す. この例では, InfoBufferEastの方がNA連続数が少ないため, East方向に転送する.
- (3) (1), (2)の手続きを踏んでも転送方向が決定しない場合, さらに(1), (2)の手順をInfoBufferの上位ビットに向かって繰り返す.
- (4) どちらかのInfoBufferの最上位ビットまで比較を行っても転送方向が決定しなかった場合, 各方向のVC情報量が多い方を優先して転送方向に選択する. 図8にその例を示す. この例では, North方向のVC情報量は2, East方向のVC情報量は4なので, East方向に転送する.
- (5) 上記のどれにも当てはまらなかった場合, すなわち, InfoBufferの情報量も内容も一致していた場合には, Chimeraに実装しているDimension-orderと同様に, X次元を優先的に転送方向に選択する.

4. ChimeraによるCross-Lineの検討

ChimeraにCross-LineとDimension-orderを実装した. 両者を比較しながら様々な視点から検討を行った結果について示す. 検討項目はレイテンシとスループット, トラフィックの可視化, Cross-Lineのオーバーヘッド, VC情報の正確性, VC情報の使用量の5項目である. 以下に特徴のある実験結果を示し, 各項目についての検討結果を述べる.

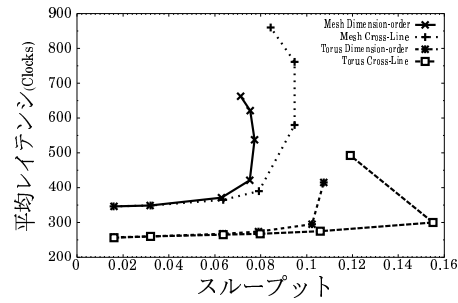


図9 ユニフォームランダム通信のレイテンシとスループット

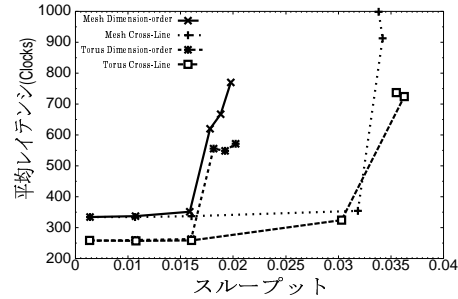


図10 ホットスポット通信のレイテンシとスループット

4.1 レイテンシとスループット

パケットの生成時間のインターバルを変化させてレイテンシとスループットを計測した.

レイテンシは, パケットが生成されてから目的地に到着するまでにかかった時間を平均したものである.

計測時間内での到着パケット数をA, ノード数をN, 計測時間をT, フリット数をFとして, スループットを式1のように定義する.

$$\text{スループット} = \frac{A}{N \times T} \quad \dots \quad (1)$$

分母が計測時間内の最大処理能力を表し, 分子がシミュレーション上の総処理量を表している.

図9, 10に, 32 × 32 (1024ノード), ストアアンドフォワード (パケットサイズ64Bytes), での転送特性のグラフを示す. 横軸がスループット, 縦軸が平均レイテンシである. 図9はユニフォームランダム通信, 図10は10%ホットスポット通信の場合である.

両グラフから, Cross-Lineはユニフォームランダム通信よりも, ホットスポット通信のようなデータパケットの集中する通信でより大きな効果を発揮することがわかる.

4.2 トラフィックの可視化

各ノードのVCでパケットがブロックされている時間を計測することで渋滞の広がりを可視化し, 各ノードへの入力リンクの使用率を計測することでトラフィックの分散状況を可視化する. 図11, 12は, ともに2D-Mesh, ワームホール方式でのホットスポット通信を, Dimension-orderとCross-Lineでシミュレーションしたときのトラフィックの様子である. ホットスポットをネットワークの中心に置き, 全パケットの10%がホットスポットに集中するようにした.

図11は, ネットワーク渋滞状況を示している. (1)がDimension-orderでの渋滞状況, (2)がCross-Lineでの渋滞状況である. 1ドットが1ノードを表し, 色が白に近

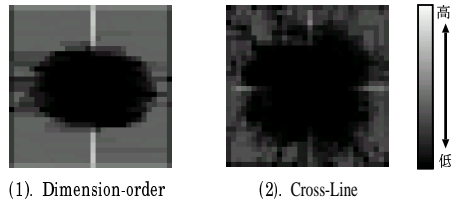


図 11 トラフィックの渋滞状況

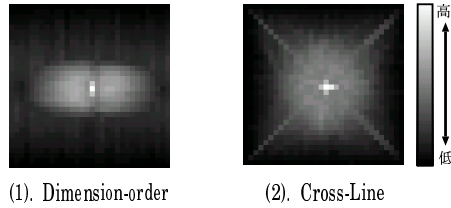


図 12 トラフィックの分散状況

い程ノード内でパケットが長くブロックされていることを表している。また、図12は、トラフィックの分散状況を示している。渋滞状況と同様、(1)がDimension-orderでの渋滞状況、(2)がCross-Lineでの渋滞状況である。1ドットが1ノードを表し、最もパケット流入量が多いノードを白で表すことで他のノードを相対的な値で色分けしている。

図11の(1)では、横次元優先Dimension-orderのために、パケットがHotSpotを中心にした縦次元に集中していることがわかる。中心部のブロック時間が短いことから、中心部に近い程パケットの転送が活発に行われていることがわかる。それに対し、(2)ではその集中を縦横の両次元に分散し、ネットワーク全体への渋滞の広がりも緩和していることがわかる。

また、図11の(1)が示すように、ブロックがHotSpotを中心とした縦次元に集中している分、図12の(1)では横次元のリンクが高使用率を示している。(2)ではその集中をネットワークの広範囲に分散していることがわかる。分散の度合は一定で、Cross-Lineが高いトラフィック分散能力を持っていることを示している。

4.3 オーバーヘッド

Cross-Lineは動的な情報を扱うため、情報の転送がパケットの転送を妨げることがある。これがCross-Lineのオーバーヘッドになる。情報の送信がパケット送信を妨げている時間の合計をB、全パケットのレイテンシの合計をLとすると、オーバーヘッドは式2で定義される。

$$\text{オーバーヘッド} = \frac{B}{L - B} \quad \dots \quad (2)$$

2D-Mesh, 2D-Torusにおいて、ストアアンドフォワード方式でのユニフォームランダム通信とホットスポット通信をシミュレーションしたときの、パケット生成インターバルの変化に応じたオーバーヘッドのグラフである。ホットスポット通信では、ホットスポットをネットワークの中心に置き、全パケットの10%がホットスポットに集中するようにした。

図13にオーバーヘッドの計測結果を示す。

オーバーヘッドは2D-Torusのストアアンドフォワード、ユニフォームランダム通信において、パケット生成インターバルが400(clocks)のときに最大値で6.74%を示した。また、インターバルが、ネットワークが飽和状態でなくなるインターバル以上になると、オーバー

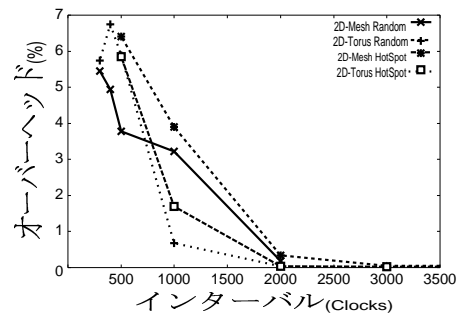


図 13 ストアアンドフォワードでのオーバーヘッド

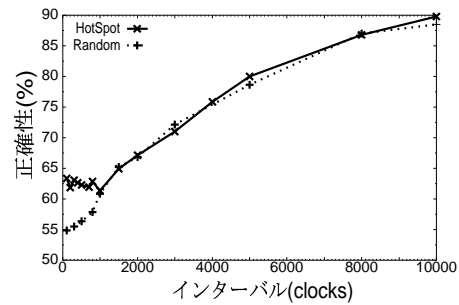


図 14 VC情報の正確性

ヘッドは急激に減少し、インターバルが2000(clocks)をこえると、全ての結果でオーバーヘッド1%を下回る結果になっている。

このように、Cross-Lineは低オーバーヘッドでの実現が可能である。

4.4 情報の正確性

3.4節で説明したように、送信時の優先度はVC情報よりもパケットのほうが高い。このため各ノードのInfoBufferにあるVC情報は、実際のVCの状態変化に遅れて更新される。Cross-Lineにおいて、ルーティング方向の決定に用いるVC情報(InfoBufferの内容)と、実際のVCの状態との差を測定した。全InfoBufferのうち、一致しているVC情報のビット数の割合を求めている。

図14は、2D-Meshにおいて32×32(1024ノード)ストアアンドフォワード、ユニフォームランダム通信と10%ホットスポット通信で、インターバルを変化することで得たシミュレーション結果である。

横軸は各ノードのパケット生成インターバルを表し、縦軸はVC情報の正確性を表している。空いている場合は両コミュニケーションパターンで90%近い高い正確性を示すが、ネットワークが次第に渋滞していくと徐々に低くなる。ホットスポット通信ネットワークが飽和している状態では、50~60%の値に安定する。ユニフォームランダム通信ではネットワークが飽和状態になっても正確性は安定せずに低下する。

4.5 VC情報の使用量

Cross-Lineでは、図1に示したように自ノードが属する縦横の2本の直線上に存在する全てのノードのVC情報を扱う。

現在のモデルでは収集可能な全てのVC情報を収集しているが、全てのVC情報を利用しているとは限らない。実際にルーティングに用いるのは何hop先までのVC情報を調査することで、Cross-Lineの最適なチャンネル情報

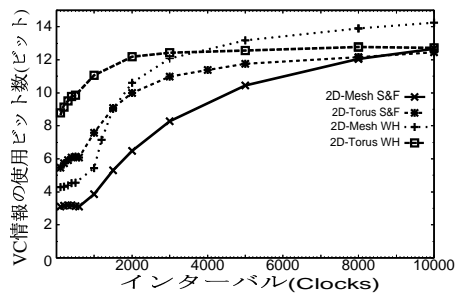


図 15 VC 情報の使用量

の量について検討を行った。

ネットワークサイズ、フリット数、パケット生成インターバルなどを変化させ、VC 情報の使用量がどのネットワークパラメータに依存するのかを調べた。図 15 は、 32×32 (1024 ノード) 構成でのホットスポット通信において、パケット生成インターバルを変化させたときの VC 情報の使用量を表している。インターバルが小さいとき、VC 情報の使用量は比較的小さい値を示している。全てのグラフでインターバルが約 800(clocks) 以上になると、VC 情報の使用量は増加していき、シミュレーション条件に依らず VC 使用量は 12~15 ビット程度に落ち着く。VC 使用量が増加し始める 800(clocks) というインターバルは、ホットスポット通信においてパケットの消費量を生成量が上回りはじめるインターバルであることが図 9, 10 より分かっている。つまり、VC 情報の使用量はネットワークの処理能力に依存することが分かる。

5. 関連研究

相互結合網のルーティングアルゴリズムはいままで様々なものが提案されてきた。VC を North ポートと South ポートにだけ備え、少ない VC 数でデッドロックフリーな完全適応ルーティングを実現する Opt- y^3 や、2 本の VC を用意し、適応ルーティング用 VC と Dimension-order 用 VC にわけることによってサブ適応ルーティングを行う Duato's Protocol³⁾ などがあるが、現在のネットワークを取り巻く状況を考え、適応ルーティングの特徴である動的な情報による判断を、さらに効率的に考えたルーティング方式が考えられている。

Cross-Line と同様に、ネットワークの動的な混雑情報を用いるアルゴリズム SSR (Speculative Selection Routing)²⁾ が提案されている。SSR ではルーティングの手続きを二段階に区別する。チャンネルにパケットが送れるかを判断し、ルーティングでデッドロックやライブロックの発生を許容するかを決定するルーティングファンクションと、パケットを送るチャンネルを決定するセクションファンクションである。SSR の役割はセクションファンクションであり、ルーティングファンクションとして Opt- y^3 や Duato's Protocol³⁾ などと組み合わせることによってルーティングを行う。

SSR は定期的にパケットの送信を中断し、各ノードが 2 hop 以内の直線上のノードから様々な情報を得る。情報はリンクの可用性 (Free/Busy)、リンクのディメンション、リンクのリクエスト数、ノードに存在するパケットの数という多くの情報から成る。これらを各ノードの情報用バッファに記憶し、ルーティングの際に情報用バッファの値をもとにルーティング方向を選択する。

Cross-Line は、SSR と比べるとシンプルな手法を用いている。各ノードが提供する情報は、VC が利用可能か

否かの 1 ビットの情報のみであり、リンクが完全に未使用となるときに VC 情報を転送する。これにより、4.3 節で示したように、低オーバーヘッドで情報を扱うことができる。シミュレーションの結果から、VC 情報の転送によるパケット転送の遅延は実質的に問題とならないことがわかっている。

また、SSR ではその情報量の多さから、3 hop 先の情報はハードウェアの機構が複雑になりすぎるため扱えないという問題がある。Cross-Line では情報量とその扱いの機構がともシンプルのため、何 hop 先の情報を扱ってもハードウェアの複雑さが極端に増すことはない。また、VC の Available と NA という状態の変化だけが情報として扱われるため、VC のパケット許容量を大きくすることで情報の送信回数を削減できることが予想でき、更なるパフォーマンスの向上が期待できる。

6. おわりに

本稿では、相互結合網のパフォーマンス向上を目指し、動的な情報を用いた適応ルーティングである Cross-Line 実現のための詳細な検討結果を示した。検討結果から、Cross-Line は現在のモデル高いトラフィック分散能力を持ち、でも十分に効果があることがわかる。今回検討した結果を踏まえて最適なモデルで実装すれば、更なる性能向上が期待できる。

また、VC 情報の使用量はネットワークのスループットに依存することがわかった。 32×32 (1024 nodes) 構成では、混み具合により 3~15 ビットの使用量を示す。VC 情報量を変化させると、情報の転送回数が変化することになる。実際に最適な VC 情報量を求めるには、情報量の変化による転送への影響を調査する必要がある。

VC 情報の正確性の変化は渋滞状況に依存し、混んでいると全体としての正確性は低減する。検討結果から、VC 情報量を変化させた場合、正確性は向上すると考えられる。

今後は、検討により導き出されたモデルを相互結合網シミュレータ Chimera に実装し、SSR をはじめとした他の適応ルーティングとの比較を行う予定である。また、サイズの大きい VC を備えたネットワークを Chimera でシミュレーション可能にし、今回 5 章で可能性を示すに留めた、サイズの大きい VC を備えたネットワークでの Cross-Line の性能についても評価と比較を行う予定である。

謝辞 本研究は、一部日本学術振興会科学研究費補助金 (基礎研究 (B)14380135, 同 (C)14580362, 若手研究 14780186) の援助による。

参考文献

- 1) 絵面聡, 西谷雅史, 横田隆史, 大津金光, 馬場敬信 “ルーティングアルゴリズム開発のための環境構築とその初期評価”, 情報処理学会第 66 回全国大会, 6T-5 (2004・3)
- 2) Tran Cong So, Shigeru Oyanagi, and Katsuhiko Yamazaki “Speculative Selection in Adaptive Routing on Interconnection Networks”, 先進的計算基盤システムシンポジウム SACSIS2003, pp.29-36(2003)
- 3) José Duato, Sudhakar Yalamanchili and Lionel Ni “Interconnection Networks –An Engineering Approach–”, Morgan Kaufmann Publishers(2003)