

故障ノードの通過に基づく耐故障ルーティング法の NAS Parallel Benchmarks を用いた性能評価

黒川 陽太[†] 福士 将[†]

山口大学 大学院創成科学研究科[†]

1. はじめに

メニーコアプロセッサにおける CPU コア間の接続方式として、NoC (Network-on-Chip) が注目されている。NoC は、CPU コアとルータからなるノードをネットワークで接続し、パケットルーティングによりノード間の通信を行う。このため、NoC では、製造時や稼働時に発生する故障に対処するための耐故障ルーティングが必要不可欠となる。

これまで、様々な耐故障ルーティング法が研究されてきた [1-4]。中でも、[1, 2] の手法は、故障ノードを迂回するだけでなく通過することを可能にしたものであり、既存手法よりも通信性能が優れている。しかし、これらの手法を含む多くの手法では、シミュレーションによる通信遅延やスループットの評価にとどまっている。ベンチマークを用いて実行時間を評価している例もあるが、通信パターンのみを反映したものが多く、十分な評価はなされていない。

本報告では、実際の並列プログラムを用いた実行時間の評価を行うことを目的に、実行時間の見積り方法を提案する。並列計算機のベンチマークとして一般的に用いられる NAS Parallel Benchmarks (NPB) を用いた評価を行い、耐故障ルーティング法が並列プログラムの実行時間に及ぼす影響を評価する。

2. 耐故障ルーティング法と評価方法

2.1. 耐故障ルーティング法

本報告の性能評価で対象とする耐故障ルーティング法を以下で簡単に説明する。

- Fcube4 [3]

本手法は、故障ノードを含む最小サイズの矩形の故障領域を作成し、パケットに迂回させる手法である。迂回のルールを定義するために、ネットワークを仮想的に多重化する仮想チャンネル (VC) を 4 個使用する。図 1 (a) に、出発ノ

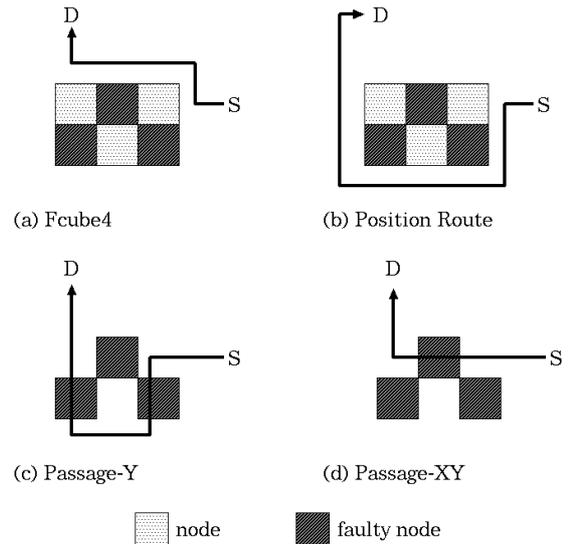


図 1 ルーティング例

ド (S) から目的ノード (D) までのルーティング例を示す。この手法では、D に近づく方向で故障領域を迂回する。

- Position Route [4]

本手法は、Fcube4 と同様に矩形の故障領域を作成するが、迂回方向が一意に定義されているため、VC を必要としない。図 1 (b) に、ルーティング例を示す。この手法では、現在のノードと D の位置関係から迂回方向が一意に決定されるため、経路長が長くなる場合もある。

- Passage-Y [1]

本手法は、ルータの周辺にスイッチとリンクを追加して、故障ノードの通過を可能にした手法である。このため、故障領域は作成せず、VC も必要としない。図 1 (c) に、ルーティング例を示す。この手法では、行方向の移動においては故障ノードを迂回し、列方向の移動においては故障ノードを通過させる。

- Passage-XY [2]

本手法は、Passage-Y を拡張したものであり、2 個の VC を用いて、故障ノードを行方向と列方向の両移動において通過させるものである。図 1

(d) に、ルーティング例を示す。この手法では、最短経路に近いルーティングが可能である。

2.2. 評価方法

ベンチマークを用いた評価を行うために、対象とする並列プログラム P から計算時間と通信情報を取得し、実行時間を見積もる方法を提案する。提案手法を以下に示す。

1. 計算時間の取得
プロセス間通信を除く計算処理に要する時間 R_t を取得するために、 P のソースコードの該当箇所にタイム関数を挿入する。
2. 通信情報の取得
プロセス間の通信パターンと通信量（通信情報 CI ）を取得するために、ソースコードの該当箇所に送受信ノードとデータサイズを出力する関数を挿入する。これらは、通信関数の引数から取得可能である。
3. P の実行
 P を実行し、 R_t と CI を取得する。
4. 実行時間の見積もり
 - 計算時間
NoC における計算時間 C_p を次式で算出する。

$$C_p = R_t \times F_{CPU} \times \frac{F_{router}}{F_{core}}$$

ここで、 F_{CPU} は P を実行した CPU の動作周波数、 F_{core} と F_{router} は、それぞれ、NoC で想定する CPU コアとルータの動作周波数を表す。 C_p は、実測で得られた計算時間をルータの通信時間に合わせたサイクル数を表す。

- 通信時間
 CI を用いて耐故障ルーティングのシミュレーションを行い、通信時間 C_c のサイクル数を計測する。
5. P の実行時間の算出
 $C_p + C_c$ により総サイクル数を算出する。1 サイクルは 1Hz に相当するものであることから、 P の実行時間の見積値が得られる。

3. 評価

2 節で説明した各手法に対し、ベンチマークを用いた場合の実行時間を評価する。本評価では、ベンチマークとして、NPB の整数ソート (IS) と高速フーリエ変換 (FT) を用い、各プログラムのクラスを W 、並列数を 64 とした。NoC のノード数を 10×10 、各ノードの故障率 f を 2% または 10% とし、ルータ内でのルーティング処理には、VC を用いる場合は 4 サイクル、用いない場合は 3 サイクルかかるものとする [5]。また、 F_{core} を 2GHz、 F_{router} を 200MHz とし、各プロセスはノード番号

表 1 各手法の $C_p + C_c$ ($\times 10^3$)

f	NPB	Fcube4	Position Route	Passage-Y	Passage-XY
2%	IS	1,293	1,795	1,167	1,062
	FT	445	470	441	433
10%	IS	1,677	2,347	1,383	1,033
	FT	465	494	447	431

の昇順に正常ノードのみに割り当てるものとする。

表 1 に、各手法の IS と FT の実行時間を示す。いずれの場合においても、Passage-XY の実行時間が一番短く、続いて、Passage-Y、Fcube4、Position Route の順になった。 $f = 10\%$ のとき、Passage-XY の Fcube4、Position Route、Passage-Y に対する削減率を求めると、IS の場合は、それぞれ、約 38%、約 56%、約 25%、FT の場合は、約 7%、約 13%、約 4% となる。IS のほうが FT に比べて通信量が多いため削減率も大きく、用いる耐故障ルーティング法の優劣により、並列プログラムの実行時間に大きく差が出ることが分かった。

4. おわり

本報告では、NoC 型のメニーコアプロセッサ向けに開発された耐故障ルーティング法に対して、実際の並列プログラムを用いた実行時間の評価を行うために、実行時間の見積り方法を提案した。今後は、NoC 上でより高速にベンチマークを実行するためのタスクのマッピング方法を考案する。

参考文献

- [1] Y. Kurokawa and M. Fukushi, "Passage of Faulty Nodes: A Novel Approach for Fault-tolerant Routing on NoCs," IEICE Trans. Fundamentals, Vol. E102. A, No. 12, pp. 1702–1710, Dec. 2019.
- [2] 黒川陽太, 福士将, "2次元メッシュ NoC における故障ノードの通過に基づく決定的・適応的耐故障ルーティング法," 信学論, Vol. J104-D, No.7, pp.574-585, July 2021.
- [3] R.V.Boppana and S.Chalasanani, "Fault-Tolerant Wormhole Routing Algorithms for Mesh Networks," IEEE Trans. Comput., Vol. 44, No. 7, pp. 848–864, July 1995.
- [4] Y. Fukushima et al., "A Region-based Fault-tolerant Routing Algorithm for 2D Irregular Mesh Network-on-Chip," J. Electron. Test.: Theory Appl., Vol. 29, Issue 3, pp. 415–429, 2013.
- [5] W. J. Dally and B. Towles, "Principles and Practices of Inter-connection Networks," Morgan Kaufman Publishers, 2004.