社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

# ホストから複数リンクを用いた低遅延ネットワークトポロジ

河野　隆太†　藤原　一毅††　松谷　宏紀†　天野　英晴†　鯉渕　道紘††

† 慶應義塾大学　〒 223–8522 神奈川県横浜市港北区日吉 3–14–1
†† 国立情報学研究所 / 総合研究大学院大学 / JST　〒 101–8430 東京都千代田区一ツ橋 2–1–2
E-mail: †{kawano,hunga}@am.ics.keio.ac.jp, ††{ikki,koibuchi}@nii.ac.jp, †††matutani@ny.ics.keio.ac.jp

**あらまし**　ハイパフォーマンスコンピューティング (HPC) システム上での並列アプリケーションに対して，ホスト間の通信遅延が重大な問題となっている．そのため，高次元スイッチを用いた高基数のトポロジとして HPC システムを構築することが必要である．これまでの我々の研究では，高次元スイッチを用いたトポロジを構成する際に，規則的なトポロジにランダムなスイッチ間リンクを付加する方法を提案した．本研究では従来の提案を拡張し，複数リンクを単一ホストと複数のスイッチとの間に付加した．フリットレベルシミュレーションにより，ランダムホストリンクを用いた我々のトポロジは，リンク集約を用いた従来のトポロジに比べ，スループットを同程度維持しつつ，レイテンシを最大 51 %減少させた．

**キーワード**　トポロジ，相互接続網，ハイパフォーマンスコンピュティング，高次元スイッチ．

# Low latency network topology using multiple links at each host

Ryuta KAWANO†, Ikki FUJIWARA††, Hiroki MATSUTANI†, Hideharu AMANO†, and Michihiro KOIBUCHI††

† Keio University　Hiyoshi 3–14–1, Kohoku-ku, Yokohama, Kanagawa, 223–8522 Japan
†† National Institute of Informatics / The Graduate University for Advanced Studies / JST　Hitotsubashi 2–1–2, Chiyoda-ku, Tokyo, 101–8430 Japan
E-mail: †{kawano,hunga}@am.ics.keio.ac.jp, ††{ikki,koibuchi}@nii.ac.jp, †††matutani@ny.ics.keio.ac.jp

**Abstract**　End-to-end network latency has become an important issue for parallel application on large-scale High Performance Computing (HPC) systems. It is thus necessary to build HPC systems as high-degree topologies by using high-radix switches. We have recently proposed a method to build topologies with such switches by adding random switch-to-switch links on a base regular topology. In this report, we extend the method by adding multiple links between a single host and multiple switches. Results obtained with flit-level discrete event simulation show that our random host-link topologies achieved comparable throughput with latency up to 51% lower than that of baseline topologies with link aggregation.

**Key words**　Topology, interconnection networks, high performance computing, high-radix switches

## 1. はじめに

次世代の高性能システムにおける多くのマルチコア並列アプリケーションは，ストロング/ウィーク・スケーリングを問わず，数百ナノ秒〜1 マイクロ秒の低 MPI 通信遅延が必要となることが予測されている [1] [2] ．したがって，これらの高性能計算システムに向けた低遅延ネットワークの研究開発が今後，重要となる．ネットワーク内では Infiniband QDR 1 台のスイッチ遅延が約 100 ナノ秒などスイッチ遅延が支配的である．一方，フリットの注入遅延，リンク遅延などは相対的に小さい．したがって，低直径，短い平均距離 (ホップ数) のトポロジを用いることがネットワーク内の低遅延化につながる．

現在，数十ポート以上の高次元スイッチが利用可能であるため，高次元トポロジの採用により低遅延化を探求することが可能である．高次元スイッチのその他の利用法としては，同一のノード (スイッチまたはホスト) 間で複数のリンクを集約してつなぎ，高帯域化や耐故障性の強化を狙うことが挙げられる．

ホスト・スイッチ間でのリンクの追加は，従来リンク集約を目的として行われていた．しかし，本研究では，ホスト間の遅延削減を目的として行う．すなわち，異なるホスト・スイッチ

間で複数のリンクを” ショートカット” としてランダムに繋ぐ.

最近の研究で，従来のトポロジにランダムショートカットを付加したネットワークトポロジが，ホップ数を劇的に削減でき．それらが HPC やデータセンターネットワーク (DCN) に適用可能であることが示されている [3] [4] [5] .

そこで，本研究では，従来の研究の適用範囲をスイッチとホストを含めたネットワークトポロジに拡張し，単一ホストから複数のスイッチにランダムホストリンクを付加することで，従来手法であるリンク集約を用いる場合に比べた場合のホップ数削減・遅延低減を目指す.

我々の目標は，ホスト・スイッチ間のランダムショートカットの有効な利用方法を追求し，一方で費用の増加やパッケージングの複雑化，通信遅延につながる総配線延長を小さくするよう最適化を行うことである.

本論文の構成は以下である． 2. 章で関連研究を述べ， 3. 章において，グラフ解析からランダムホストリンクの有用性を評価する． 4. 章では，配線長削減のための最適化手法を適用し，トポロジの配線長を評価する． 5. 章では，フリットレベルシミュレーションにより，ランダムホストリンクを用いたトポロジを評価する．最後に， 6. 章において結論を述べる.

## 2. 関 連 研 究

### 2.1 スイッチ間トポロジ

直径の小さい大規模なネットワークトポロジについては従来から研究されている．その中には，規則的な直接網として，De Bruijn [6] やスターグラフ [7] などがある．これらの多様なトポロジの存在は，相互接続トポロジにおけるデザインスペースが大きいことを示している.

一方，ランダムグラフを用いたネットワークが，そのスモールワールド性により，従来の規則的なトポロジに比べて直径や平均最短距離を小さくできることが注目されている．このようなスモールワールドグラフを用いたデータセンター向けネットワークは，拡張性や耐故障性，帯域に関して優れていることが報告されている [3], [5] .

また，最近の HPC ネットワークトポロジについては，性能・消費電力・費用・配線の複雑性・耐故障性などのトレードオフについて，包括的な議論がなされている．その中で，費用対性能の優れた HPC 向けトポロジとして Flattened Butterfly [8] が提案されている他，高基数のトポロジにおける配線長の削減を目的とした Dragonfly トポロジ [9] が提案されている.

本研究では，スイッチ間トポロジとして，規則的なトポロジである hypercube と，完全なランダムトポロジを取り上げ，ランダムホストリンクを適用した際の効果を定量的に評価した.

### 2.2 複数ホストリンク

特に高基数スイッチを用いて HPC システムのトポロジを実装する際は，リンク・アグリゲーション (リンク集約) と呼ばれる，同じホスト・スイッチ間で複数のケーブルを束ねて接続し，帯域幅のボトルネックを回避するのが実用的な方法である.

また，複数のホストリンクを使用するもう一つのアプローチは，単一の HPC システムで異なる相互接続網をサポートすることである．例えば，SGI Altrix3000 [10] には 2 つの同一のトポロジが存在し，各ホストがそれらへの入力を行うために 2 つのネットワークインターフェースを持つ.

HPC システム向けに，使用デバイスの異なる 2 種類のネットワークが提案されている．1 つは，長いバルクデータ転送のための光回路スイッチングネットワークであり，もう 1 つは低帯域幅の電気パケットスイッチングである [11] ．このようなネットワークは，高帯域幅のためだけでなく，低遅延のためにも使用することができる.

しかしながら，我々の知る限り，単一のトポロジ上で経路ホップ数を減らすために複数ホストリンクをショートカットとして利用する研究はこれまで行われていない.

## 3. グラフ解析

### 3.1 トポロジ生成

本章ではグラフ解析を用いることにより，リンク集約を用いた従来のトポロジと，複数ランダムリンクを用いたトポロジの評価を行う．具体的には，直径と平均最短距離の評価を，以下に示すトポロジについて行う.

- HYPERCUBE-$x$-i: 各ホストが $x$ 本のランダムリンクを，同一キャビネット内の異なるスイッチに繋いだ hypercube トポロジである.
- HYPERCUBE-$x$-o: ランダムリンクの接続範囲が異なるキャビネットのスイッチを含み，それ以外の条件は HYPERCUBE-$x$-i と同じである.
- RANDOM-$x$-i: スイッチ間のトポロジは，各スイッチの次数が等しい完全なランダムトポロジである．各ホストは $x$ 本のランダムリンクを，同一キャビネット内の異なるスイッチに繋ぐ.
- RANDOM-$x$-o: ランダムリンクの接続範囲が異なるキャビネットのスイッチを含み，それ以外の条件は RANDOM-$x$-i と同じである.

この解析では，平等な比較のため，スイッチ間トポロジにおける各スイッチの次数を $Log_2N$(hypercube の次数) に合わせる ($N$ はネットワークサイズ．)．$p$ を 1 キャビネットに格納するスイッチ数の上限とし，$p = 16$ をデフォルト値とする．ホスト数は 1 スイッチあたり 8 台とする.

また，通常の解析では，中間ホストによるパケットフォワーディングは行わないものとする．これは，ホストでのパケット転送による性能の向上がとても小さいためである．(このことについての定量的な評価を 3.4 章で行う．)

### 3.2 ランダムホストリンク数

まず，各ホストにおけるランダムリンクの数を増加させた場合の直径と平均最短距離の変化について解析する.

図 1 と図 2 に，1024 スイッチにおける各トポロジの直径，平均最短距離を示す．図 1 と図 2 より，HYPERCUBE トポロジにおいては，ランダムホストリンクの存在範囲をキャビネット外まで許した場合，存在範囲がキャビネット内に限定されている場合と比べて，直径・最短平均距離を減らすことができる．これは，スイッチ間のトポロジについては規則的となっており，

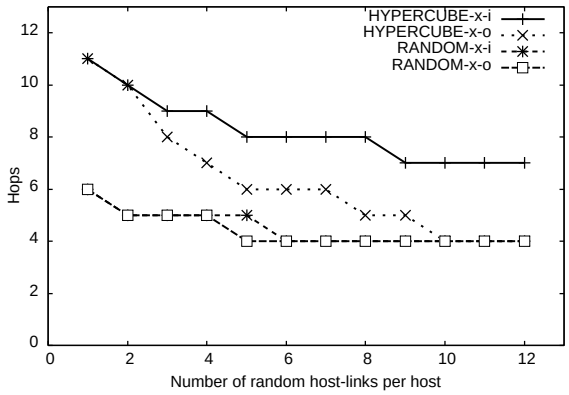


図 1　HYPERCUBE と RANDOM トポロジにおけるランダムホストリンクの数と直径 ($N$ =1,024 スイッチ)

Fig. 1　Diameter versus number of random host-links for HYPERCUBE and RANDOM topologies (1,024 switches).

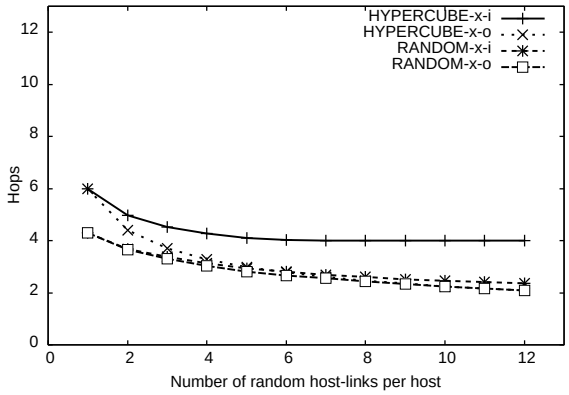


図 2　HYPERCUBE と RANDOM トポロジにおけるランダムホストリンクの数と平均最短距離 ($N$ =1,024 スイッチ)

Fig. 2　Average shortest path length versus number of random host-links for HYPERCUBE and RANDOM topologies (1,024 switches).

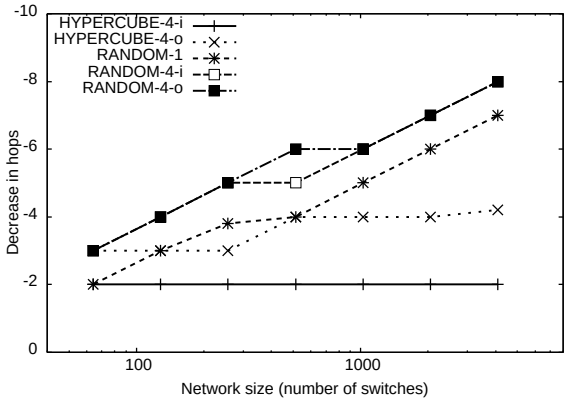


図 3　HYPERCUBE-1 を基準とした HYPERCUBE-4-{i,o}, RANDOM-1, RANDOM-4-{i,o} の直径の削減量

Fig. 3　Diameter decrease over HYPERCUBE-1 for HYPERCUBE-4-{i,o} and RANDOM-{1,4}-{i,o} topologies.

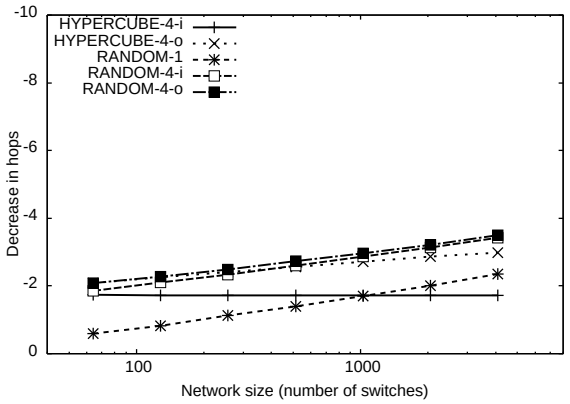


図 4　HYPERCUBE-1 を基準とした HYPERCUBE-4-{i,o}, RANDOM-1, RANDOM-4-{i,o} の平均最短距離の削減量

Fig. 4　Average shortest path length decrease over HYPERCUBE-1 for HYPERCUBE-4-{i,o} and RANDOM-{1,4}-{i,o} topologies.

キャビネット外へのランダムホストリンクを導入することにより，ネットワーク全体でランダム効果を得られるためである．一方，RANDOM トポロジについては，ランダムホストリンクの存在範囲がキャビネットの内外のいずれであるかは，経路ホップ数にほとんど影響を与えていないことがわかる．これは，スイッチ間のランダムトポロジが既にランダム効果を得ているためである．

### 3.3　スケーラビリティ

この章では，スケーラビリティの評価として，ネットワークサイズ $N$ が大きくなるにつれて，ランダムホストリンクによる性能改善がどの程度大きくなるかを調べる．ここでは，HYPERCUBE-1 と比べたホップ数の変化を HYPERCUBE-4-{i,o}, RANDOM-1, RANDOM-4-{i,o} のトポロジ間で比較した．

図 3 と図 4 は $N$ が $2^6$ から $2^{12}$ に変化した際の，HYPERCUBE-1 からの直径，平均最短距離の削減数を示している．(値の変化をマイナスで表示している)

RANDOM-1 が HYPERCUBE-1 に比べてスケーラビリティが高いのは我々の先行研究 [4] でも明らかな通りである．また，HYPERCUBE-4-i をのぞく全てのトポロジにおいて，$N$ が大きくなるにつれて，経路ホップ数の削減数が増大している．これは，トポロジ規模が大きくなるにつれて，ランダムリンクを使うことによる性能改善の効果が大きくなることを意味する．

さらに，HYPERCUBE-1 と RANDOM-1 の平均最短距離がそれぞれ 6.0, 4.3 と差があるのに対し，それらにランダムホストリンクを付加した HYPERCUBE-4-o と RANDOM-4-o がほぼ同じ平均経路長を持つ．このことから，スイッチ間のトポロジに関係なく，ランダムホストリンクの適用によりランダム効果が発揮され，経路ホップ数を削減し，スケーラビリティを得られることがわかる．

### 3.4　ホストによるパケットフォワーディング

ここまでの評価は，中間ホストをパケットが経由しないという条件で評価したが，この章では，中間ホストを経由する経路を許した場合の性能の変化を調査する．

図 5 と図 6 にホストによるパケットフォワーディングの有無による直径，平均最短距離の比較を示す．ここで，ホストによるパケットフォワーディングを許した場合の凡例の末尾を”-hrt”としている．これらの図より，HYPERCUBE-4-o をのぞく全てのトポロジにおいて，ホストを経由しない場合に対する性能の改善がほとんど見られないことが分かる．

従って，ホストによるパケットフォワーディングは，直径や平均最短距離を大きく改善するためにはあまり有用でないこと

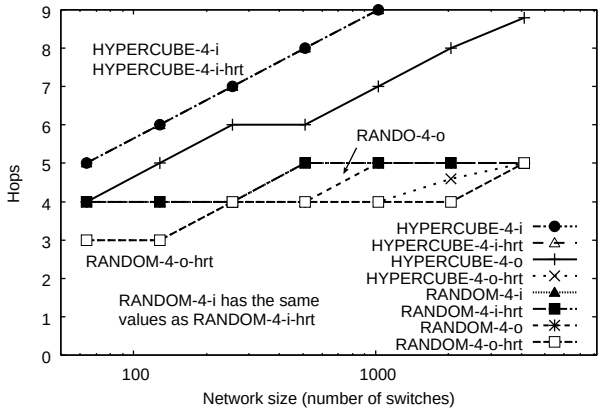


図 5　ホストによるパケットフォワーディングが有る場合と無い場合でのネットワークサイズと直径

Fig. 5　Diameter versus network size for host packet forwarding and for switch packet forwarding.

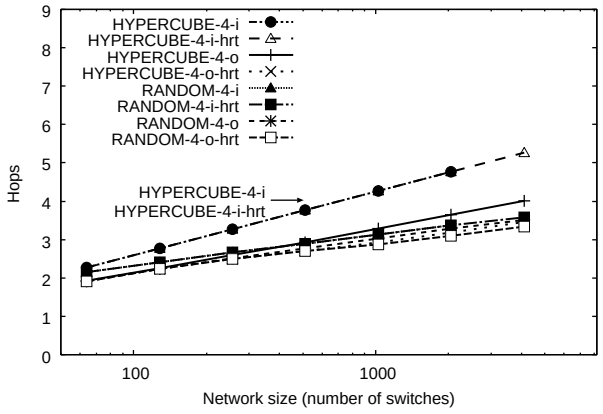


図 6　ホストによるパケットフォワーディングが有る場合と無い場合でのネットワークサイズと平均最短距離

Fig. 6　Average shortest path length versus network size for host packet forwarding and for switch packet forwarding.

が分かった．

## 4. 配線長の削減

ランダムホストリンクトポロジは，リンク集約を用いる場合と比較して，総配線延長を増大させる傾向にある．

そこで，この章では，トポロジの物理レイアウトを最適化する方法を用いることで，フロア内でのキャビネット配置において，キャビネット内・キャビネット間の総配線延長を減らすことを目指す．

### 4.1 方　　法

最適化手順の詳細については先行研究 [12] に示されており，次の 2 つの最適化ステップからなる: (1) スイッチおよびホストのクラスタリング, (2) キャビネットの配置.

以下に我々の実装を示す.

#### 4.1.1 クラスタリング・アルゴリズム

このステップでは，スイッチとホストを各キャビネットへグループ化するため，スイッチおよびホストを頂点とするグラフを，キャビネットを頂点とし，格納されたスイッチ及びホストの数を重みとする，重み付き単純無向グラフに変換する．密に接続された頂点群を 1 つにグループ化するクラスタリング手法を用いて，キャビネット間の配線数を減らすことができる．

先行研究 [12] では，様々なトポロジに対して複数のクラスタリング手法を適用し評価している．その中で Ward 法 [13] が実験的に最善手法の 1 つと示されている．そこで本研究では Ward 法をトポロジのクラスタリングに用いることとした．

なお，TOPOLOGY-$x$-i トポロジについて，あるキャビネット内のスイッチの数が $x$ よりも小さくクラスタリングされた場合には，そのキャビネットでは，各ホストがキャビネット内の全てのスイッチに対してホストリンクを接続する．この場合，各ホストのリンク数は $x$ を下回ることとなる．

#### 4.1.2 マッピング・アルゴリズム

このステップでは，フロアプランにおけるキャビネットの物理レイアウトを算出する問題を，施設配置問題としてモデル化し，最適化する．

我々は，2 次割当問題に対して適用実績のあるメタヒューリスティックスな方法として知られる，Simulated Annealing 法 (SA 法) [14] をマッピング手法として採用した．SA 法の反復回数は 1 億回，試行数は 5 回とした．

### 4.2 平均配線長

本報告の評価では，2 次元グリッド上に全てのキャビネットを配置するため，十分に大きな物理フロアプランを前提とする．厳密には，キャビネットの数を $m$ としたとき, キャビネットの列数は $q = \lceil \sqrt{m} \rceil$ であり，1 列あたりのキャビネット数は $p = \lceil m/q \rceil$ である．過去の推奨 [15] に従い，通路幅を含めたラックの占有面積は幅 0.6[m] × 奥行 2.1[m] とする．キャビネット間の距離はマンハッタン距離である．配線のオーバーヘッドについては先行研究 [8] に基づき，キャビネット内の配線を 2[m] とし，キャビネット間の配線オーバーヘッドをキャビネット当たり 2[m] とした．

各キャビネットはスイッチとホストを合わせて最大 144 台格納し，ネットワーク全体において，ホストの数はスイッチ数の 8 倍となっている．

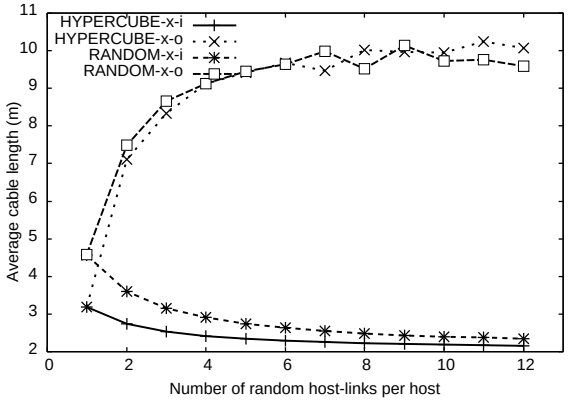


図 7　1024 スイッチにおける HYPERCUBE-$x$-$\{i, o\}$ と RANDOM-$x$-$\{i, o\}$ のランダムリンク数と平均配線長

Fig. 7　Average cable length versus number of random links at hosts for HYPERCUBE-$x$-$\{i, o\}$ and RANDOM-$x$-$\{i, o\}$ (1,024 switches).

図 7 は 1024 スイッチのネットワークにおける 1 ホストあたりのランダムホストリンクの本数と平均配線長 (1 本あたり)

の関係を示している．キャビネット外のスイッチへのホストリンクを許した場合，平均配線長は急激に増加する．また，$x$ が 4 以上の場合において，HYPERCUBE-$x$-o の平均配線長は RANDOM-$x$-o の場合に近くなり，HYPERCUBE-$x$-i に比べて 3.7 倍以上大きくなっている．これは，1024 スイッチでの RANDOM-4-o において，ランダムホストリンクが総配線延長の 86%を占めているためである．反対に，HYPERCUBE-$x$-i と RANDOM-$x$-i の場合は，ランダムホストリンクの長さは全て 2m であるため，平均配線長が減少する．実際，ランダムホストリンクをキャビネット内に限定することにより配線長を最大 73%減らせる．

## 5. シミュレーション評価

本章では，ランダムホストリンクのネットワーク性能への影響を調べるため，遅延を評価した．

### 5.1 シミュレーション環境

評価には C++で記述されたフリットレベルシミュレータ [4] を用いた．このモデルでは，スイッチングファブリックはチャネルバッファ，クロスバ，リンクコントローラ，制御回路で構成される．また，各スイッチは，同数の次数を持ち，スイッチング技術としてバーチャルカットスルーを用いた．

ヘッダフリットがスイッチを通過する遅延は最低 100[ns] とした．この中にはルーティング計算，仮想チャネルアロケーション，スイッチアロケーション，入力ポートから出力ポートへのクロスバを経由したフリット転送遅延が含まれる．また，リンク遅延は 5[ns/m] とした．各リンク長は前章の結果から得られたものを用いる．

各ホストは独立してネットワークにパケットを入力するものとした．フリットサイズは 256 ビットとし，リンクの実効バンド幅は 96Gbps とした．また，低遅延が必要となる通信粒度は細かく，さらにその細粒度通信でのスループットが重要であることが報告されているため [1] ，パケット長は 33 フリット (ヘッダ含む) とした．

すべてのトポロジにおいて仮想チャネルは 4 本とした．また，デッドロックフリーな topology-agnostic ルーティングである Duato のアプローチを採用し，その逃げ道として up*/down* routing を用いた [16]. 逃げ道を選択する際はなるべく” 遅延が最も低い経路” を選ぶこととした．この時，各ホップにおける配線長の違いから，最小ホップ数の経路 (最短経路) が選ばれるとは限らない．

前章と同じく，1 キャビネットに格納されるスイッチ・ホスト数は最大 144 とし，ホスト数はスイッチ数の 8 倍とする．

### 5.2 ネットワーク性能の評価

図 8-図 11 は，ランダムに宛先を選択する Uniform トラフィックと，合成トラフィックパターンである Matrix Transpose トラフィックを，スイッチ数 256・ホスト数 2048 のネットワークに注入した場合のシミュレーション結果である．縦軸はパケットが生成されてから宛先ホストに到達するまでのホスト間の遅延を，横軸は各ホストの受信フリットレートである accepted traffic を示す．

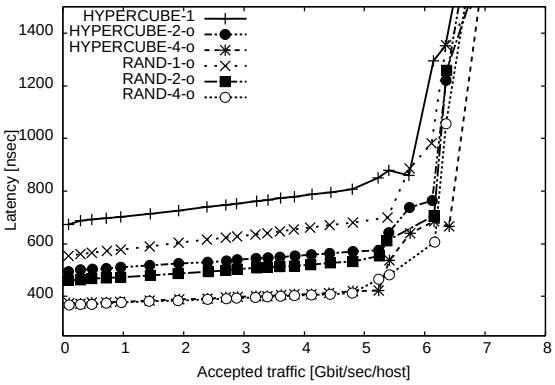


図 8 HYPERCUBE-$x$-o と RANDOM-$x$-o におけるネットワーク性能 (256 スイッチ，2,048 ホスト，Uniform トラフィック)

Fig. 8 Latency vs. accepted traffic for HYPERCUBE-$x$-o and RANDOM-$x$-o topologies (256 switches, 2,048 hosts, Uniform traffic).

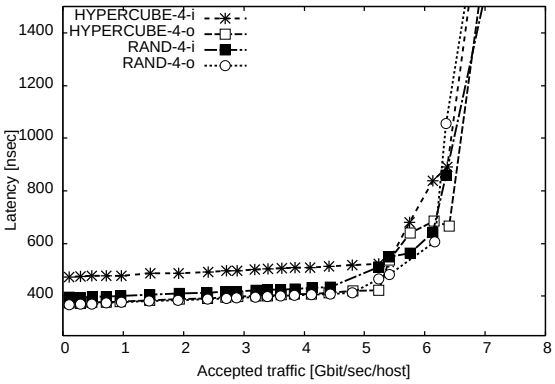


図 9 HYPERCUBE-$x$-{i,o} と RANDOM-$x$-{i,o} におけるネットワーク性能 (256 スイッチ，2,048 ホスト，Uniform トラフィック)

Fig. 9 Latency vs. accepted traffic for HYPERCUBE-$x$-{i,o} and RANDOM-$x$-{i,o} topologies (256 switches, 2,048 hosts, Uniform traffic).

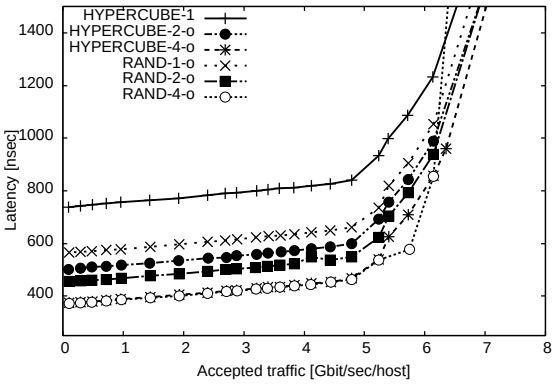


図 10 HYPERCUBE-$x$-o と RANDOM-$x$-o におけるネットワーク性能 (256 スイッチ，2,048 ホスト，Matrix transpose トラフィック)

Fig. 10 Latency vs. accepted traffic for HYPERCUBE-$x$-o and RANDOM-$x$-o topologies (256 switches, 2,048 hosts, Matrix transpose traffic).

図 8 と図 10 から，RANDOM-o はリンク数の増加に伴い遅延を減らしている．また，図 9 と図 11 から，RANDOM-4-i と RANDOM-4-o の遅延の値はほぼ同じであり，最も良い値を取る．このことから，スイッチ間のトポロジがランダムの場合は，ランダムホストリンクの存在範囲がキャビネット内に限られる

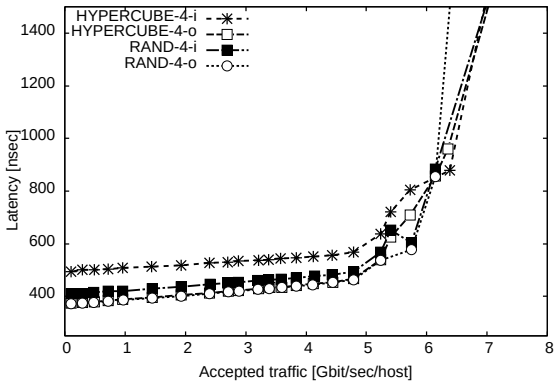


図 11 HYPERCUBE-$x$-{i,o} と RANDOM-$x$-{i,o} におけるネットワーク性能 (256 スイッチ，2,048 ホスト，Matrix transpose トラフィック)

Fig. 11 Latency vs. accepted traffic for HYPERCUBE-$x$-{i,o} and RANDOM-$x$-{i,o} topologies (256 switches, 2,048 hosts, Matrix transpose traffic).

場合でも，リンク数の増加に伴い遅延が減少する．

一方，HYPERCUBE の場合は，図 9 と図 11 から，HYPERCUBE-i トポロジでは，他のトポロジと比べて，ランダムリンクにより遅延を効果的に削減することは出来ない．また，図 8 と図 10 から，HYPERCUBE-1 に対する HYPERCUBE-4-o の遅延性能は，Uniform トラフィックにおいて 44%減少し，Matrix Transpose トラフィックにおいて 49%減少している．この遅延の改善度合は，ランダムリンク数の増加に伴い大きくなる．よって，HYPERCUBE では，ランダムリンクの存在範囲がキャビネット外の場合は，ランダムリンクにより効果的に遅延を削減できる．

また，ランダムホストリンクはトラフィックパターンに関わらず有益であることがわかる．

以上の結果は，遅延が経由ホップ数と相関することから，第 3. 章でのグラフ解析の結果を裏付けるものとなっている．

## 6. 結　　論

本研究では，単一ホストと単一スイッチとの間でリンクを集約してつなぐ従来の方法と異なり，単一ホストと複数スイッチとの間で複数のランダムなショートカットリンクを張ることにより，ネットワーク遅延の削減を目指した．

ランダムホストリンクの付加により，ホスト間の直径や平均最短距離が劇的に改善する．特に，スイッチ間が規則的なトポロジである hypercube の場合，ランダムホストリンクの追加によって直径や平均最短距離がランダムトポロジのそれらに近づく．

また，スイッチ間がランダムトポロジの場合は，ランダムなホストリンクの存在範囲をキャビネット内に限定することにより，直径や平均最短距離を増大させることなく，平均配線長を 73%減少させることが可能である．これはランダムホストリンクがネットワーク全体の配線長に対して支配的であることに起因する．

さらに，フリットレベルシミュレーションの結果より，ランダムホストリンクを用いたトポロジは，リンク集約を用いた従来のトポロジに比べ同等のスループットを達成しながら遅延を最大 51%削減することができる．

## 文　　献

[1] K. Scott Hemmert et al, “Report on Institute for Advanced Architectures and Algorithms, Interconnection Networks Workshop 2008,” http://ft.ornl.gov/pubs-archive/iaa-ic-2008-workshop-report-final.pdf.

[2] J. Tomkins, “Interconnects: A Buyers Point of View,” ACS Workshop, 2007.

[3] J.Y. Shin, B. Wong, and E.G. Sirer, “Small-World Data Centers,” Proc. of the Symposium on Cloud Computing, Oct. 2011.

[4] M. Koibuchi, H. Matsutani, H. Amano, D.F. Hsu, and H. Casanova, “A Case for Random Shortcut Topologies for HPC Interconnects,” Proc. of the International Symposium on Computer Architecture (ISCA), pp.177-188, 2012.

[5] A. Singla, C.Y. Hong, L. Popa, and P.B. Godfrey, “Jellyfish: Networking Data Centers Randomly,” Proc. of USENIX Symposium on Network Design and Implementation (NSDI), 2012.

[6] M.R. Samatham, and D.K. Pradhan, “The De Bruijn Multiprocessor Network: A Versatile Parallel Processing and Sorting Network for VLSI,” IEEE Trans. on Computers, vol.38, no.4, pp.567-581, 1989.

[7] S.B. Akers, B. Krishnamurthy, and D. Harel, “The Star Graph: An Attractive Alternative to the n-Cube,” Proc. of the International Conference on Parallel Processing (ICPP), pp.393-400, 1987.

[8] John Kim and William J. Dally and Dennis Abts, “Flattened Butterfly: A Cost-Efficient Topology for High-Radix Networks,” ISCA, pp.126-137, 2007.

[9] J. Kim, W.J. Dally, S. Scott, and D. Abts, “Technology-driven, highly-scalable dragonfly topology,” ISCA, pp.77–88, 2008.

[10] M. Woodacre, D. Robb, D. Roe, and K. Feind, “The sgin altixtm 3000 global shared-memory architectur,” SGI white paper.

[11] K.J. Barker, A.F. Benner, R.R. Hoare, A. Hoisie, A.K. Jones, D.J. Kerbyson, D. Li, R.G. Melhem, R. Rajamony, E. Schenfeld, S. Shao, C.B. Stunkel, and P. Walker, “On the Feasibility of Optical Circuit Switching for High Performance Computing Systems,” Proc. of SC, p.16, 2005.

[12] https://www.dropbox.com/s/c3x0lxhsnhxnclt/anonymous.pdf.Anonymized version of preliminary version of an article currently under review.

[13] P. Pons, and M. Latapy, “Computing communities in large networks using random walks,” Computer and Information Sciences ISCIS, pp.284–293, 2005.

[14] D.T. Connolly, “An improved annealing scheme for the QAP,” European Journal of Operational Research, vol.46, no.1, pp.93–100, May 1990.

[15] HP, “Optimizing facility operation in high density data center environments , technoloogy brief,” , 2007.

[16] F. Silla, and J. Duato, “High-Performance Routing in Networks of Workstations with Irregular Topology,” IEEE Trans on parallel and distributed systems, vol.11, no.7, pp.699–719, 2000.