

# 光グリッドにおけるホスト間情報共有のためのパス構成法

## Path Establishment in Optical Grid Networks for Information Sharing

原井 洋明\*<sup>1</sup>  
Hiroaki Harai

村田 正幸\*<sup>2</sup>  
Masayuki Murata

\*<sup>1</sup> 情報通信研究機構  
NICT

\*<sup>2</sup> 大阪大学  
Osaka University

### 1. はじめに

近年、インターネットを介した広域の分散計算環境（グリッド）が多く構築されている。しかし、インターネットでは、遅延や帯域の保証が難しく、波長パスネットワークを用いたグリッド（光グリッド）の研究開発も行なわれている[1-3]。本稿では、エンドエンドに帯域が保証されている波長パスネットワークへの光グリッド構築を検討する。本稿では、複数ホストからなるグループで情報共有をするために、グループ内に多対多通信の光パスを設定する方法を提案する。本方法は、(1) 光パス設定に用いる資源を減らし、ネットワーク内資源の効率的利用に寄与すること、(2) GMPLS ネットワークなど、WDM ネットワークで考えられてきた一対一のルーティング（経路制御）方式をそのまま使うこと、が可能である。

### 2. 多対多接続とルーティング

低コストで多対多接続する手段に、以下の二方法がある。

1. 双方向マルチキャスト木（Light Tree）を設定する。MST（Minimum Steiner Tree）問題を解き、コスト最小木を発見する。
2. 隣接ホスト間に片方向の光パス(Lightpaths)を順次設定する（図 1）。巡回セールスマン問題を解き、コスト最小パス集合を発見する。

なお、いずれも、一ホストがデータ送出に利用できる帯域の平均は、グループのホスト数に逆比例する。帯域がそれ以上必要なら、新たなパスを設定する等の処理を要する。

本稿では方法 2 を用いる。2 ではパスを片方向のみに設定すればよく、双方向パスを要する 1 より資源の有効利用に繋がると期待できるからである。方法 2 の最適化は NP 完全問題であり、Closest-point heuristic 等、様々なヒューリスティックアプローチが提案されている[4]。

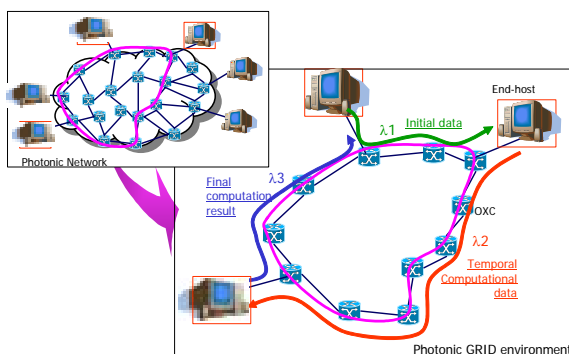


図 1. 隣接ノードにパスをつないだ多対多通信。

ただし、光グリッドへ適用する場合、単にコスト最小パス集合を発見するだけではなく、波長パスネットワークのルーティングとの親和性を考慮する必要がある。多対多通信に最適な経路は、グループ数やグループ内ホスト集合ごとに異なる。ホスト数  $N$  に対し最適な多対多通信の経路は  $\sum_{k=2}^N C_k$  とおりある。一方、ホスト数  $N$  に対し、一対一通信の最適な経路数は  $N(N-1)$  である。通常、ネットワークのルーティングは経路数の少ない一対一通信に特化する。したがって、ネットワーク内ノード（ルータ）に多対多通信のすべての最適経路を保持し、複数パスの接続をすべてルータに任せるのは実質的に困難である。本稿で対象とするネットワークでも、ルーティングは、多対多通信でなく一対一通信に最適な経路情報を保持する。

ホストは隣接するルータから定期的に経路情報を受け取り、自身から相手までの経路情報のみを持つ。例えば、ルータの経路制御が BGP で、隣接ルータから経路情報が広告される場合があてはまる。ホストは、それを元に多対多通信の接続パス最適化を図る。このとき、ホストは、隣接するノードへの経路情報は持つが、自身を含まない 2 ホスト間の経路情報は持たないという制約のもと、多対多接続のパスを探す必要がある。

### 3. 提案アルゴリズムとシグナリング

以下に、ホスト数  $G$  からなるグループにおける多対多接続のために、光パスをつなぐ送受信ホストの組を求める提案アルゴリズムを記述する。アルゴリズムにおいて、親ホストとは、多対多通信のトリガーを引くホストを示し、子ホストとは、残りのホストを示す。

1. 【親ホスト】情報共有をするホスト集合  $S=\{h_s, h_1, h_2, \dots, h_{N-1}\}$  に対し、情報を初期化する。既共有ノードリスト  $L$ 、未共有リスト  $U$  を作成する。すなわち、 $L=\{\phi\}$ ,  $U=\{h_s, h_1, h_2, \dots, h_{G-1}\}$  とする。
2. 【親ホスト】ホスト  $h_s$  を  $U$  から除外し、 $L$  に加える。すなわち、 $L=\{h_s\}$ ,  $U=\{h_1, h_2, \dots, h_{G-1}\}$ 。  $h_s$  から、 $U$  に含まれ経路のコストが最小となるホスト  $h$  に対して光パスを設定する。
3. 【子ホスト】光パスの宛先となるホスト  $h$  をリスト  $U$  から除外し、 $L$  に加える。例えば  $h=h_{G-1}$  であれば、 $L=\{h_s, h_{G-1}\}$ ,  $U=\{h_1, h_2, \dots, h_{G-2}\}$  とする。この時、 $U=\{\phi\}$  でなければ、 $h$  からリスト  $U$  に含まれず、かつ、経路のコストが最小となるホストに対して光パスを設定する。これを  $U=\{\phi\}$  となるまで繰り返し、次ステップへ。
4. 【子ホスト】最近リストに含まれたホストから、 $h_s$  に対してパスを設定する。

上述のアルゴリズムでは、グループ内のホストが、順次、自身から転送するデータの受信ホストを探す。したがって、2章の最後で述べたパス接続の制約は満たされる。

提案アルゴリズムのステップ 2~4 において、光パスを設定するために、シグナリングを行なう。図 2 に、親ホスト a、ホスト b、c がグループを構成するときのホスト選択とシグナリング例を示す。ここでは、a から c への経路よりも、a から b の経路の方が、コストが小さいとする。図に示すように、パス予約はフォワード型、すなわち、送信（上流）ホストから受信（下流）ホストに向けて行なう。パスが設定できることがわかれば、受信ホストは、送信ホストに対して ACK（受信通知メッセージ）を送付し、親ホストに対して P-ACK（親ホストへの受信通知メッセージ）を送付する。同時に、受信ホスト自身が次のパス設定のために送信ホストとなり、新たに探しだした受信ホストに対して、パス設定要求を送る（アルゴリズムステップ 3 参照）。こうして、すべてのパスが設定され、最終的に親ホストがすべての P-ACK を受け取ると、データ送信を開始する。

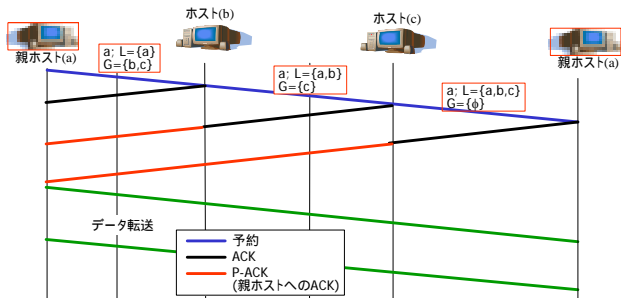


図 2. シグナリング (フォワード型) .

#### 4. 評価結果

本章では、提案方式の効果を示す。そのために、シミュレーションを実行した。ネットワークは 16 ノードの  $4 \times 4$  メッシュ構成とする。各ノードにホストが接続する。グループ内ホスト数は  $G$  固定とし、その親ホスト、子ホストはランダムに選ばれる。性能比較対象に、(1) 双方向のマルチキャスト木、(2) 集合のホストをランダムに選んだ順に、パスを順次接続する方法を選んだ。以降に示す図 (グラフ) では、それぞれ、(1) Tree Policy (2) Non-Shortest Policy と表示している。提案手法は Shortest Policy である。波長は各ホストでランダムに一つ選ぶ。

図 3 に、グループ内ホスト数に対する、その多対多通信に必要な光パスが要する物理リンク数の平均を示す。双方

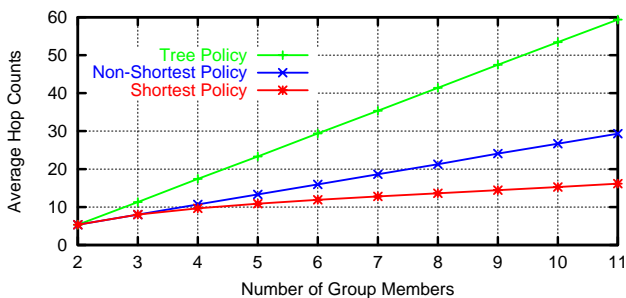


図 3. グループ内メンバー数( $G$ )と平均必要ホップ数の関係 .

向の木を接続するよりも、順次片方向の光パスを接続する方法が使用資源を抑えられること、また、提案方式を用いると、よりその効果が表われることがわかる。図 4 は、 $G=5, 10$  における多対多接続の第  $k$  番目 (横軸の値) の光パス設定に要する平均リンク数を示す。図からわかるように、提案方式では、後に接続するパスのホップ数は、(2) より若干長くなるが、先に接続するパスのホップ数が (2) より大幅に短い。したがって、先述した図 3 に示すように全体の平均リンク数が小さくなる。

次に、多対多通信用のパス接続要求がある場合の性能評価を行なった結果を図 5 に示す。グループによる多対多通信要求はポアソン到着、保留時間は平均 1 の指数分布に従う。リンクの波長数はすべて 8 である。性能指標を多対多接続のパス損失率とする。

図 5 より、グループのホスト数  $G$  が 5, 10 の場合とも提案方式の損失率の方が小さく、良好な性能を示すことがわかる。 $G=10$  の方が、若干性能の差が大きいため、よりパス接続に必要なリンク数を抑えると、性能に好影響を与えることがわかる。

#### 5. まとめ

光グリッドにおいて、グループで情報共有する場合の多対多通信実現方式を提案した。本提案により、一対一通信のルーティングを用いて、多対多通信のためのパス設定を省資源で行なえる。シミュレーション評価により、定量的にその有効性を示した。

#### 文献

- [1] CA\*net4, <http://www.canarie.ca/canet4/>.
- [2] OptIPuter, <http://www.optiputer.net/>.
- [3] H. Nakamoto, K. Baba, and M. Murata, "Proposal of a shared memory access method for Lambda computing environment," in Proc. OpNeTec, Oct. 2004.
- [4] T. H. Cormen, C. E. Leiserson, R. H. Rivest, and C. Stein, "Introduction to Algorithms (2<sup>nd</sup> Edition)," MIT Press, 2001.

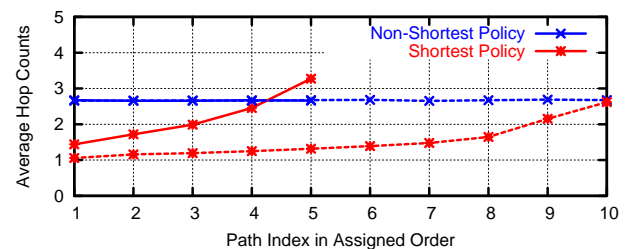


図 4. 各方式における第  $k$  (横軸) 番目に接続するパスの平均ホップ数 (実線  $G=5$ 、破線  $G=10$ ) .

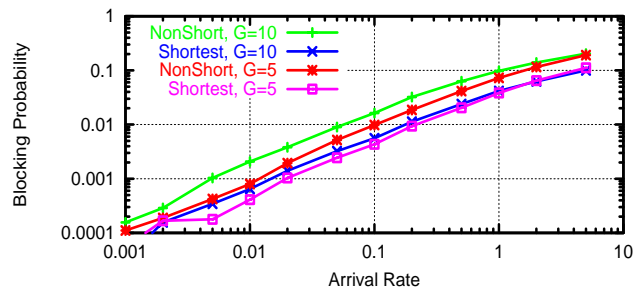


図 5. グループ内ホスト数と損失率の関係 .