大容量メモリネットワークの高速化

代表研究者　　安戸　僚汰　　広島大学　情報科学部　特任助教

1 研究の目的

ビッグデータ解析やディープラーニングといった近年重要なアプリケーションは莫大なデータ量を扱うため，メモリの大容量化が要求されている。たとえば図 1 は Amazon の商用 Web サービス Amazon Web Services (AWS)で提供されているクラウドサーバ EC2 および Microsoft のクラウドプラットフォーム Azure において要求されるメモリ容量を示したものである。2015 年ごろから急速に要求が増加して，2018 年では 6TB もの容量が必要になっていることがわかる。スマートフォンのメモリ容量は 4GB 程度なので，その 1500 倍にもなる大容量であることがわかる。

通常メモリは計算機のソケットと呼ばれるスロットに搭載するが，ソケットを増やすとコストが飛躍的に増加してしまうという問題があり，それがメモリの大容量化の障害となっている。ソケット数が 1 つの場合 2.1GHz の周波数で動作するが，3 つにすると 1.9GHz に低下すると報告されている。このように，半導体の集積化によりプロセッサの性能が指数関数的に増加するムーアの法則が成り立たないポスト･ムーア時代には，従来のメモリを継続して大容量化することは極めて難しい。

そこで次世代メモリ技術として，三次元積層を使った「メモリキューブ」が提案されている。メモリキューブはメモリに加えてスイッチ回路を内蔵しているためネットワークを構成することができ，低コストで大量のメモリを接続することが可能となる。これは従来の並列計算機の相互結合網におけるパケット通信の技術をメモリに応用したものといえる。しかしこのメモリネットワークは従来のネットワークにはない特徴を有しており，その最適な設計手法は明らかになっていない。特徴の一つは，電力・コストの制限によってメモリキューブのポート数が少ない点であり，結果としてネットワークの経路長が大きくなってしまう。これはポート数のたくさんあるルータを使用した並列計算機の相互結合網と対照的な点である。他方でプロセッサが複数のリンクを持つという利点もあり，それをどのように活用するかを探究する必要がある。また，複数のプロセッサを持つ共有メモリ型の計算機ではデータの内容に一貫性を持たせるためにキャッシュコヒーレンスプロトコルと呼ばれる仕組みが必要で，多くの場合メモリのデータを管理するディレクトリと呼ばれる機構をネットワークの中に持つ。ディレクトリの配置によってメモリ管理の遅延が大きく変わるが，これをどこに配置すべきか明らかになっていない。

そこで本研究では，実際のアプリケーションに適したメモリネットワークの最適な設計手法を明らかにすることを図る。特に，相互結合網の研究で提案されたホスト-スイッチグラフによる設計アプローチを発展させる。またプロセッサが複数のリンクを持つ利点を活かして，従来のネットワークとは異なるトポロジ（ネットワークのグラフ構造）を探究し，大容量メモリネットワークを高速化する。

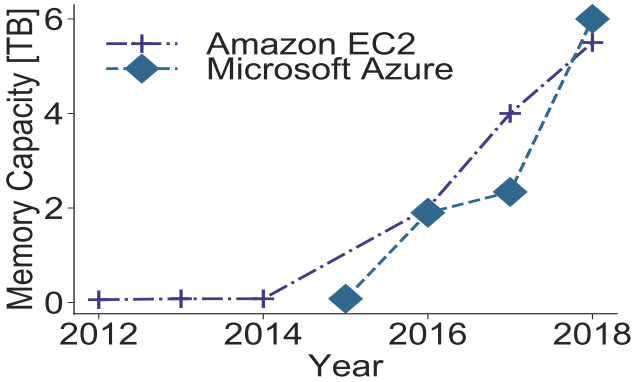

図 1. 要求されるメモリ容量の増加

公益財団法人電気通信普及財団
研究調査助成報告書　第36号　2021年度

2 背景技術

2-1 並列計算機

並列計算機において計算機をつなぐネットワークの相互結合網をグラフとみなして数学的に議論するアプローチはこれまでに多く研究されてきた。最も単純なのはネットワークを無向グラフ$G = (V, E)$として考えるものである。計算機を頂点とみなすと，頂点間の距離（ホップ数）が小さければ無負荷時の通信遅延が小さくなる。そこでグラフの直径や平均最短距離を最小化する方法が研究されてきた。具体的には，次に示すorder/degree problemが考えられている。

問題1（Order/degree problem）．頂点数n，最大次数dを満たす無向グラフの中で直径が最小となるものを求めよ。

この問題はdポートを持つ計算機ノードの数がnである相互結合網のうち無負荷時の遅延が最小になるものを見つけることに相当する。これまでの研究成果として，ランダム性を持ち込んだアプローチなどが提案されている。[10]

より詳細な相互結合網のモデルがホスト-スイッチグラフ[1]であり，計算機ノードとスイッチノードを明確に区別したものである。通常の無向グラフでは頂点が一種類だが，ホスト-スイッチグラフでは頂点としてホストとスイッチの二種類が存在する。ホストはすべて次数1であり，スイッチの最大次数は固定である。ホスト-スイッチグラフに対して次のような問題が定式化できる。

問題2（Order/radix problem）．ホスト数n，スイッチ数m，スイッチ最大次数（radix）rを満たすホスト-スイッチグラフの中でホスト間平均距離が最小となるものを求めよ。

スイッチを端点とした通信は行わないため，最小化するべき距離はホスト間の直径あるいは平均距離となる。さらに，通常固定されるのはホストの数のみであり，スイッチ数が固定されないところが特徴である。先行研究ではホスト-スイッチグラフによって相互結合網のトポロジについて既存トポロジの最適性や間接網と直接網の違い，最適なスイッチ数の予測など様々な知見が得られている[2]。本研究ではメモリネットワークについて同様にモデル化を行うことを考えていく。

2-2 Hybrid Memory Cube 等にみられる三次元積層メモリキューブ

メモリキューブは三次元積層された次世代メモリである。複数のメモリチップを積層し，その一番下の層にメモリコントローラやスイッチの機能を持つロジック層を置く。このロジック層によってパケットスイッチングが可能となる。シリコン貫通電極を用いた三次元積層はポスト・ムーア時代に集積度を高めるための有望技術であり，現在までにマイクロン・テクノロジ社によってHybrid Memory Cube（HMC）として規格化されている。本研究はこの技術を用いたメモリを一般的にメモリキューブと呼ぶ。メモリキューブでは従来のようにソケットによってプロセッサと接続するのではなく（図 2a），ロジック層の機能によってメモリキューブ間でネットワークを構成することができる（図 2b/c）。よって，ソケット数を増やさずにメモリ容量と性能を高めることができる。

2-3 メモリネットワーク

メモリキューブをネットワークとして構成することは文献[3]で提案された。この文献では特に従来のようなメモリがネットワークを構成せずプロセッサがネットワークを構成する方式をプロセッサセントリックネットワークと呼び，それと対照的にメモリがネットワークを構成するネットワークをメモリセントリックネットワークと名付けている。さらにそれらの両方の要素を取り入れたハイブリッドネットワークを提案している。文献[4]ではデータ圧縮や電力効率を上げるパワーゲーティングの活用，文献[5]では不揮発性メモリの活用が提案された。最近の文献[6]ではスケーラビリティを上げるためにランダム性を取り入れた複雑ネットワークの応用の応用が提案されたが，設計の困難さや配線長増大の問題が解決されていない。

本研究ではこれらのいずれとも異なる新しいネットワークを提案する。

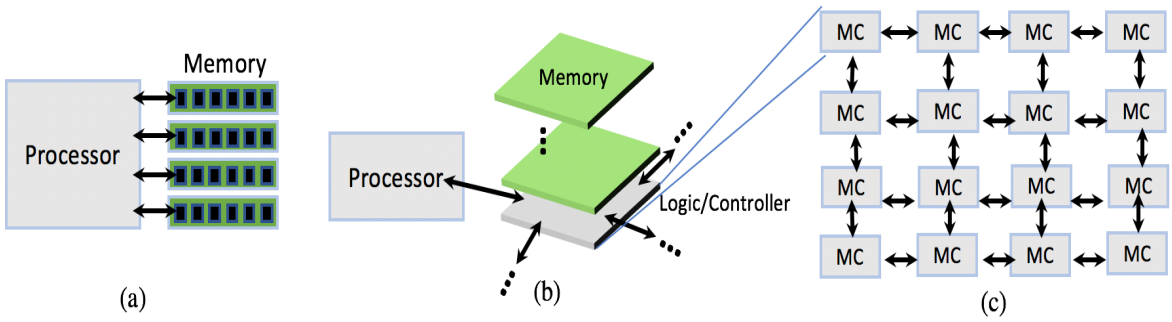

図 2. (a) 従来のメモリ，(b) メモリ・キューブ (MC)，(c) メモリネットワーク

3 方法

3-1 ホスト-メモリグラフ

まずメモリネットワークのモデル化について述べる。ホスト-スイッチグラフと同様に，ネットワークをグラフとして記述する。

ホスト-メモリグラフを三つ組$G = (H, C, E)$で定義する。$H = \{h_0, h_1, \ldots, h_{n-1}\}$は$n$個のホスト，$C = \{c_0, c_1, \ldots, c_{m-1}\}$は$m$個のメモリキューブをそれぞれ表す頂点であり，$E$は二つの頂点（ホスト同士，メモリキューブ同士，ホストとメモリキューブすべてを含む）を接合する辺を表す。ホストの次数は最大でd_h，メモリキューブの次数は最大でd_cとする。

図 3 に例を示す。ここではホストを円，メモリを四角形で示すことにする。この例のように，ホストの次数とメモリの次数が同程度で小さいことが多い。

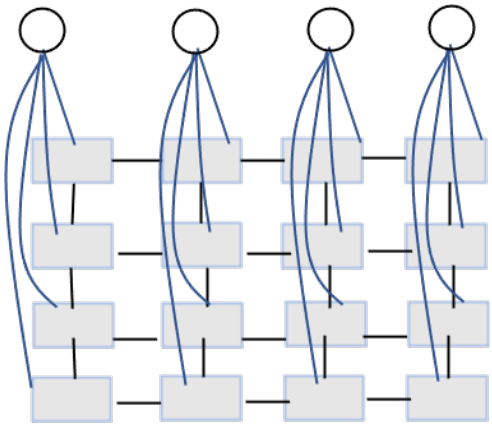

図 3. ホスト-メモリグラフの例。ホスト数 4，メモリ数 16，ホスト次数 4，メモリ次数 5。

メモリネットワークを設計する際はホストプロセッサの数とメモリキューブの数，それぞれのポート数が与えられる。最小化したいものはホストからメモリアクセスしたときの遅延である。これをホスト-メモリグラフで考えると，次のような単純な最適化問題を設定できる。

問題 3. 与えられたn, m, d_h, d_cを満たすホスト-メモリグラフの中で，ホストとメモリキューブ間の平均距離を最小になるものを求めよ。

目的関数がホストとメモリキューブ間の平均距離になっている点が特徴であり，ホスト同士とメモリキューブ間の距離は無関係であることに注意されたい。つまり，距離がいくら長くても良いし，非連結であっても構わない。この特徴によって，従来の order/degree problem や order/radix problem とは異なる考え方が必要になる。

メモリキューブネットワークを提案した最初の文献[3]で議論されている，$d_h = 4, d_c = 5$の場合を考える。この場合，メモリキューブ同士の接合の自由度はほとんどなく，図 3 に示すようなメッシュトポロジが代表的なトポロジとなる。ここで重要となるのがホストとメモリキューブの接合となる。たとえば図 4 に示すような三パターンが考えられるが，このうち(a)と(b)はホスト-メモリ間の平均距離が 5 であるのに対して(c)はホスト-メモリ間の平均距離が 4 であり，(c)の構成が最も良いことがわかる。(c)の構成は左上・左下・右上・右下の四つのメモリをグループとしてまとめることができ，グループ間のリンクはなくても平均距離はまったく変わらない。リンクを削除した構成は Network Duplication の手法[7][8]と一致する。

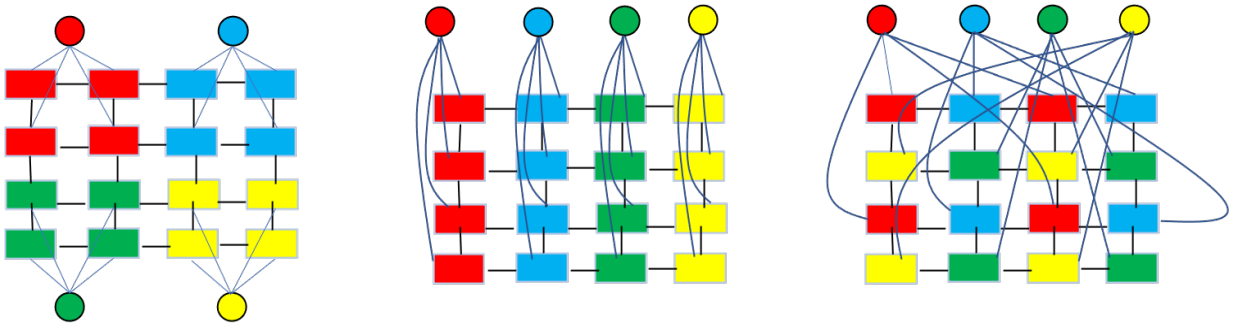

図 4. 三パターンのホスト-メモリグラフ。ローカルメモリとリモートメモリが区別できるよう色分けした。以後，ホストとメモリ間の辺が省略されている場合は同じ色のホストとメモリ同士は接続されているとする。

実システムでは，より詳細なモデル化が必要である。より詳細なモデル化として，共有メモリ型マルチプロセッサにおいてデータの一貫性を保証するためのキャッシュコヒーレンスプロトコルを考慮する。通常，メモリキューブごとに所有者（いずれかのホスト）が存在し，所有者からのアクセスはローカルアクセス，所有者以外からのアクセスはリモートアクセスと区別される。ホストからみたときには，自分が所有するメモリキューブはローカルメモリ，それ以外のメモリキューブをリモートメモリと呼ぶ。

ローカルアクセスは，単純にあるホストからローカルメモリへのアクセスによって直接ホストとメモリキューブの間でトラフィックが発生する最も単純なものである。しかしリモートアクセスでは，まずアクセス元のホスト A から，アクセスしたいメモリキューブを所有するホスト B へのトラフィックが発生し，次にホスト B からメモリキューブへのトラフィックが発生する。最後に，メモリからホスト A にデータが渡される。ここで最後のメモリからホスト A への通信は，ホスト B を経由する必要がない。以上のローカルアクセスとリモートアクセスを区別して，このアクセス遅延が小さくなるようなメモリネットワークを設計する必要がある。これらを低遅延で行うために経路長を小さくすることを考えると，従来のネットワークの研究のように経路長は最大経路長（直径）や平均経路長を最小化では，メモリネットワークには単純すぎるといえる。そこでトラフィックパターンに応じて，ノード間の通信に重みをつけた加重平均経路長を導入する。

ローカルアクセスとリモートアクセスの比率から，頂点間のトラフィックの全体に占める割合を求めることができる。その割合を重みとした距離の加重平均を，加重平均経路長として定義する。最終的に，次のような問題を考えることができる。

問題 4. 与えられたn, m, d_h, d_cを満たすホスト-メモリグラフの中で，ローカルアクセスとリモートアクセスの比率から求められる加重平均経路長を最小となるものを求めよ。

3-2 トポロジ

本研究で考案したホスト-メモリグラフを図 5 に示す。ホスト同士にリンクがある点と，メモリキューブのネットワークが非連結になっておりその数が多いことが特徴である。一つ一つのネットワークが小さくなるため，そのネットワークの直径を減らすことができる。このアプローチを極限まで押し進めると，理論的に直径を最小にすることもできる[9]。

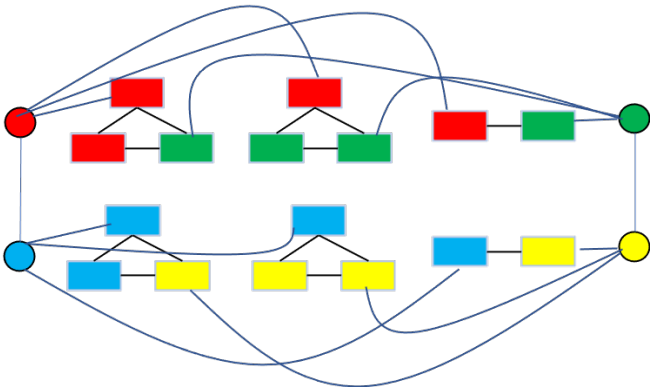

図 5. 提案するホスト-メモリグラフのトポロジ

4 結果

4-1 ローカルメモリアクセスとリモートメモリアクセス

トラフィック解析によってローカルメモリへのアクセスとリモートメモリへのアクセスの比率をアプリケーションごとに求める。並列計算機の研究によく使われる NAS 並列ベンチマーク[10]の 9 つのベンチマークで計測した。結果，いくつかのアプリケーションでリモートメモリへのアクセスが大部分を占めていることが明らかになった。表 1 にリモートメモリへのアクセスが少ない順に 1 から 9 まで番号をつけてベンチマークごとのローカルメモリへのアクセス比率とリモートメモリへのアクセス比率を示す。

表 1. ベンチマークごとのローカルメモリアクセス，リモートメモリアクセスの比率

Benchmark ID	ローカルメモリアクセス比率 (%)	リモートメモリアクセス比率 (%)
1	78.0	22.0
2	66.1	33.9
3	61.1	38.9
4	51.7	48.3
5	42.4	57.6
6	42.1	57.9
7	40.1	59.9
8	29.3	70.7
9	28.3	71.7

これに基づいて各アプリケーションに対して Baseline トポロジ（図 4）と Proposed トポロジ（図 5）を比較する。ここで，ローカルメモリ/リモートメモリの中で各メモリキューブは均等にアクセスされていると仮定する。

4-1 トポロジの比較

結果を表 2 に示す。リモートアクセスの比率が上がるごとに経路長が増えていくことがわかるが，Proposed トポロジはその上昇がゆるやかであり効率的にリモートアクセスができていることがわかる。ローカルメモリアクセスが支配的なベンチマークでは Baseline トポロジの方が経路長が短いが，ベンチマーク 4，つまりリモートアクセスが 48.3%であるベンチマークにおいて Proposed トポロジが逆転し経路長が短くなっている。全体として 9 ベンチマークのうち 6 ベンチマークで Proposed トポロジが優位であることがわかった。

今回の実験で用いたメモリキューブ数 16 の規模においてこの結果は顕著である。さらに大規模になっていくとリモートアクセスが増加し，この差は開いていくといえる。

表 2. ベンチマークごとの加重平均経路長。ベンチマーク ID が大きくなるほどリモートアクセスの比率が大きくなっている。

Benchmark ID	Baseline トポロジ	Proposed トポロジ
1	3.10	3.32
2	3.70	3.76
3	3.95	3.95
4	4.42	4.30
5	4.88	4.64
6	4.90	4.66
7	5.00	4.73
8	5.54	5.13
9	5.59	5.17

5 まとめ

本研究では，メモリキューブとホストからなるネットワークであるメモリネットワークの研究を行った。相互結合網研究におけるグラフを用いたアプローチで，ホスト-メモリグラフと加重平均経路長を定義して議論を行った。リモートメモリアクセスの影響を抑えるために，プロセッサ同士の接続とメモリキューブによる複数の小さいネットワークを組み合わせた新しいトポロジを提案した。実験の結果，リモートメモリアクセスの遅延を抑えることができ，9 つのベンチマークのうち 7 つで従来のメモリネットワーク構成の最善のものよりも全体の遅延を抑えることができることを示した。

【参考文献】

[1] Yasudo, Ryota, Michihiro Koibuchi, Koji Nakano, Hiroki Matsutani, and Hideharu Amano. "Order/radix problem: Towards low end-to-end latency interconnection networks." In *2017 46th International Conference on Parallel Processing (ICPP)*, pp. 322-331. IEEE, 2017.

[2] Yasudo, Ryota, Michihiro Koibuchi, Koji Nakano, Hiroki Matsutani, and Hideharu Amano. "Order/radix problem: Towards low end-to-end latency interconnection networks." In *2017 46th International Conference on Parallel Processing (ICPP)*, pp. 322-331. IEEE, 2017.

[3] Kim, Gwangsun, John Kim, Jung Ho Ahn, and Jaeha Kim. "Memory-centric system interconnect design with hybrid memory cubes." In *Proceedings of the 22nd international conference on Parallel architectures and compilation techniques*, pp. 145-155. IEEE, 2013.

[4] Zhan, Jia, Itir Akgun, Jishen Zhao, Al Davis, Paolo Faraboschi, Yuangang Wang, and Yuan Xie. "A unified memory network architecture for in-memory computing in commodity servers." In *2016 49th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, pp. 1-14. IEEE, 2016.

[5] Poremba, Matthew, Itir Akgun, Jieming Yin, Onur Kayiran, Yuan Xie, and Gabriel H. Loh. "There and back again: Optimizing the interconnect in networks of memory cubes." *ACM SIGARCH Computer Architecture News* 45, no. 2 (2017): 678-690.

[6] Ogleari, Matheus, Ye Yu, Chen Qian, Ethan Miller, and Jishen Zhao. "String figure: A scalable and elastic memory network architecture." In *2019 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, pp. 647-660. IEEE, 2019.

[7] Yasudo, Ryota, Michihiro Koibuchi, Koji Nakano, Hiroki Matsutani, and Hideharu Amano. "Designing high-performance interconnection networks with host-switch graphs." *IEEE Transactions on Parallel and Distributed Systems* 30, no. 2 (2018): 315-330.

[8] Hosomi, Takeo, Ryota Yasudo, Michihiro Koibuchi, and Shinji Shimojo. "Dual-Plane Isomorphic Hypercube Network." In *Proceedings of the International Conference on High Performance Computing in Asia-Pacific Region*, pp. 73-80. 2020.

[9] Yasudo, Ryota, Koji Nakano, Michihiro Koibuchi, Hiroki Matsutani, and Hideharu Amano. "Designing low-diameter interconnection networks with multi-ported host-switch graphs." *Concurrency and Computation: Practice and Experience* (2020): e6115.

[10] Bailey, David H., Eric Barszcz, John T. Barton, David S. Browning, Robert L. Carter, Leonardo Dagum, Rod A. Fatoohi et al. "The NAS parallel benchmarks summary and preliminary results." In *Supercomputing'91: Proceedings of the 1991 ACM/IEEE conference on Supercomputing*, pp. 158-165. IEEE, 1991.

[11] Nakano, Koji, Daisuke Takafuji, Satoshi Fujita, Hiroki Matsutani, Ikki Fujiwara, and Michihiro Koibuchi. "Randomly optimized grid graph for low-latency interconnection networks." In *2016 45th International Conference on Parallel Processing (ICPP)*, pp. 340-349. IEEE, 2016.

〈発　表　資　料〉

題　　名	掲載誌・学会名等	発表年月
Designing low-diameter interconnection networks with multi-ported host-switch graphs	Concurrency and computation: Practice and Experience	2020 年 12 月