高性能コンピューティング特論
講義メモ(7-1)
並列計算機の相互結合網

スーパーコンピュータの変遷

- 1970年代
  - 単体ベクトル計算機：Cray-1, Star-100, APU, IAP, ...
- 1980年代前半
  - 単体～少数並列ベクトル計算機：Cray-2, Cray-XMP, Cyber20X, S810, VP-200, SX-2
- 1980年代後半
  - 超並列計算機：iPSC, nCUBE, CS-1, T-series, CM-1, ...
- 1990年代前半
  - 高並列ベクトル計算機：NWT(VPP500), SX-4
- 1990年代後半
  - （本格的）超並列計算機：AP1000-3000, CP-PACS(SR2201), T3E, ...
  - 高並列ベクトル計算機：VPP5000, Cray-SV1, ASCI-xxx, ...
  - クラスタ計算機：x86 or Alphaベース、Storage Area Network (Myrinet, Quadrix)
- 2000年代～：
  - 高並列ベクトル計算機：Earth Simulator
  - 超並列スカラー計算機：BlueGene/L
  - クラスタ計算機：x86ベース、Storage Area Network (Infiniband, Myrinet)

（超）並列計算機における相互結合網の様々

- バス結合システム
  - Balance8000, Encore
- クロスバ網・多次元クロスバ網
  - 小規模ベクトル機, SX-xx, EarthSimulator, CP-PACS, SR11000
- 超並列：Multi-Stage Network
  - NYU Ultracomputer, IBM RP3, Cedar, GF-11
- 超並列：Mesh/Torus
  - Intel iPSC, 筑波PAX/PACS (PAX-128, QCD-PAX), Cray T3E, ASCI-Red, ASCI-RedStorm, IBM BG/L
- 超並列：Clos網、FatTree等
  - IBM ASCI-xx, SANに基づくPCクラスタ

超並列システムの相互結合網の変遷

- 単一バスシステム
  - 共有メモリ（cc-SMP等）のための小規模システム
⇒規模の拡大と共にクロスバ網に移行
⇒高速スイッチに基づくSMPへ
- Mesh/Torus
  - O(N)のハードウェアコストにより黎明期より注目
⇒PAX, T3E, BG/Lと、ベーステクノロジは変わっても継続的に利用
⇒現在も超並列の本命
- Hyper-Cube等
  - 数学的美しさ等で黎明期に活躍
⇒実装の難しさ等のため消滅
- FatTree
  - CM-5で用いられるがその後低迷
⇒SAN (Storage Area Network)がPCクラスタで用いられはじめ、現在の高性能クラスタでは標準的

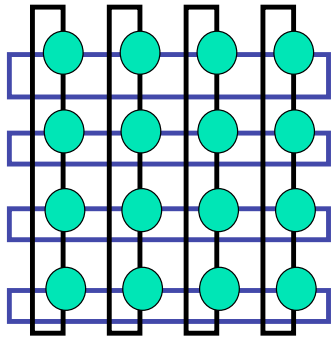
相互結合網の特性／分類

- Topology
  - Static (Direct) / Dynamic (Indirect)
  - Diameter (Distance)
  - Degree (Number of Links)
- Routing Algorithm
  - Route Decision
  - Buffering
- Performance Metrics
  - Throughput
  - Latency

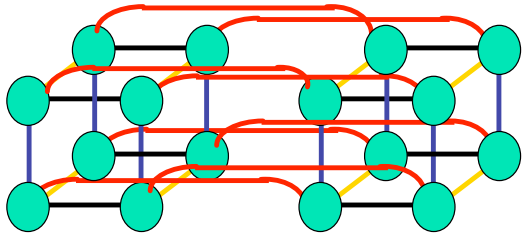
静的(直接)網

- 2-D/3-D Mesh/Torus
- Hypercube
- Direct Tree
- RDT (Recursive Diagonal Torus)

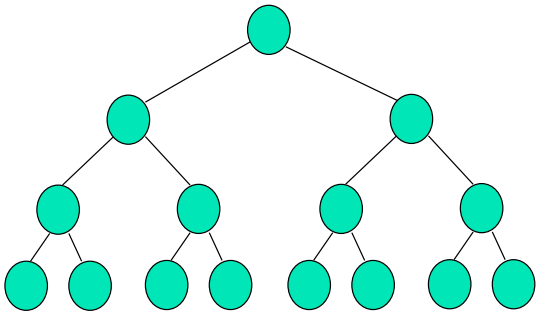
Mesh/Torus (k-ary n-cube)



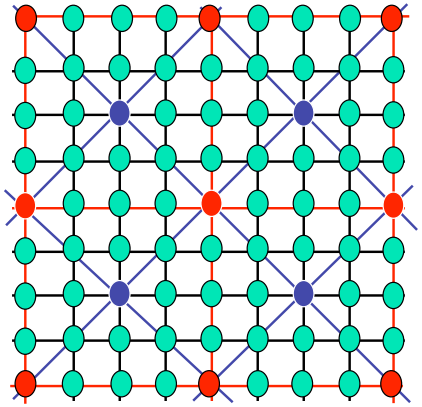
Hypercube (n-cube)



Direct Tree



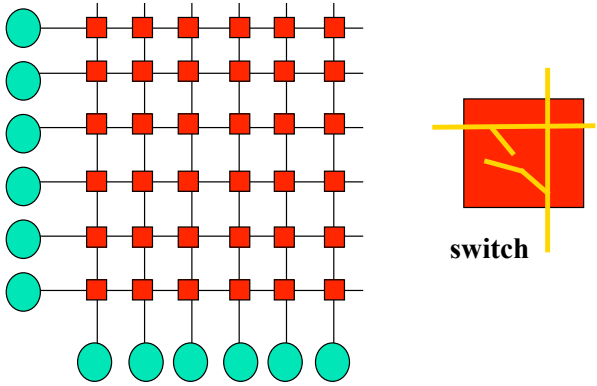
RDT (Recursive Diagonal Torus)



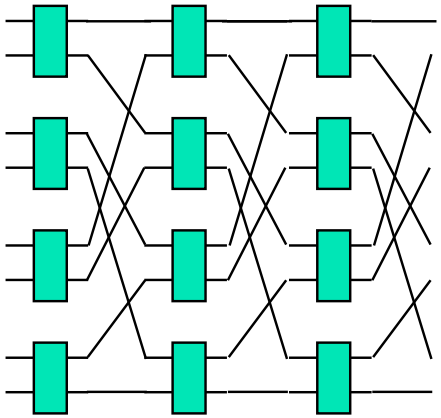
動的(間接)網

- Crossbar
- MIN (Multistage Interconnection Network)
- HXB (Hyper-Crossbar)
- Tree (Indirect)
- Fat Tree

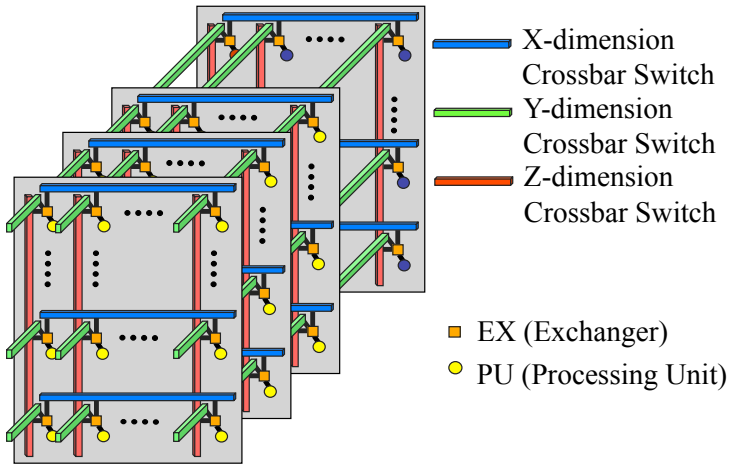
Crossbar



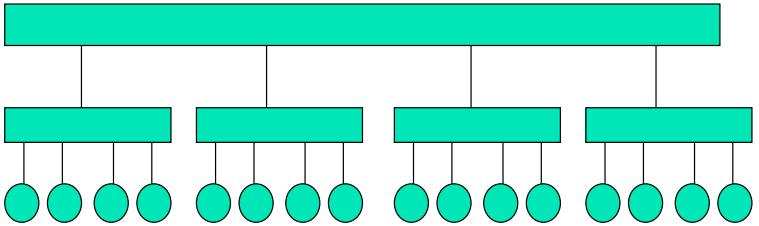
MIN (Multi-stage Interconnection Network)



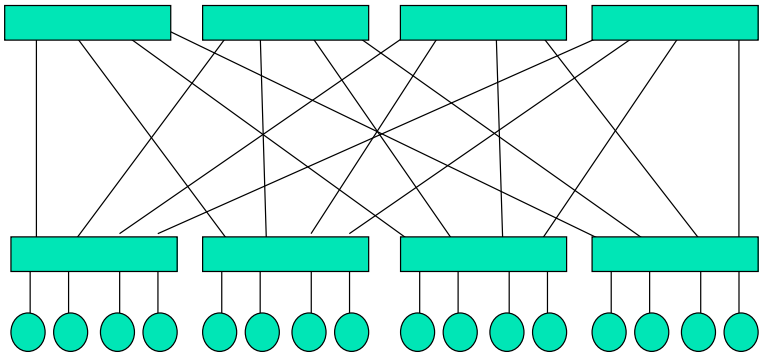
MDX (Multi-Dimensional Crossbar)... HXB



Tree



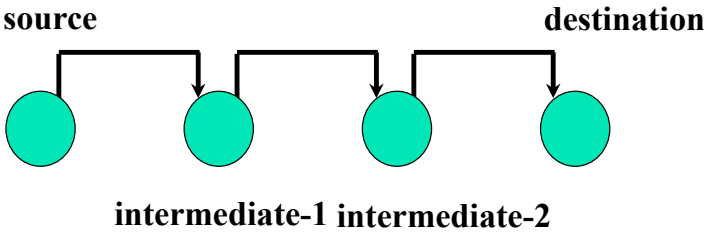
Fat Tree



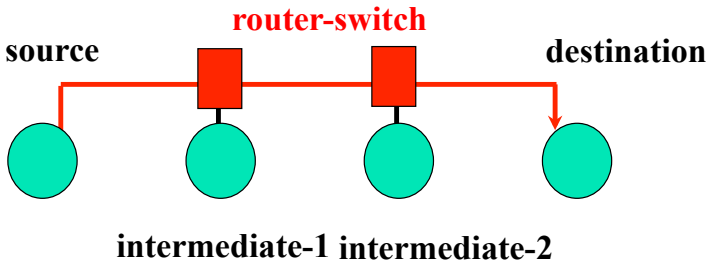
ルーティングアルゴリズム

- 経路決定
  - Fixed Routing （固定ルーティング）
  - Adaptive Routing （適応ルーティング）
  - Fault Tolerance, Detour （耐故障・迂回）
  - Deadlock problem （デッドロック問題の回避）
- バッファリング
  - Store & Forward
  - Wormhole
  - Virtual Cut-Through

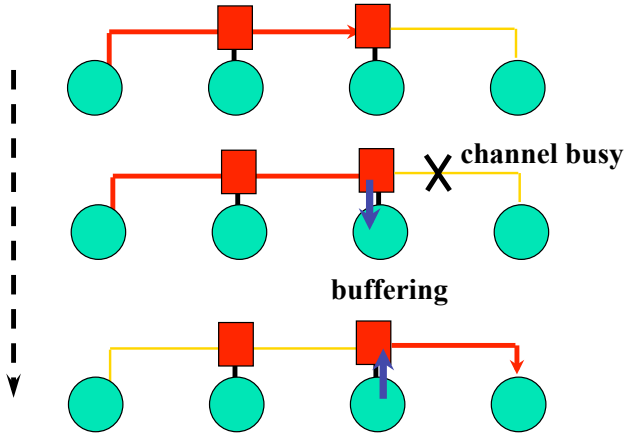
Store & Forward Routing



Wormhole Routing

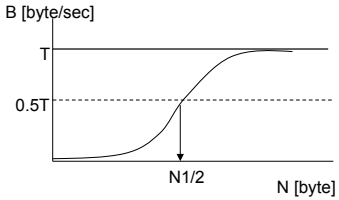


Virtual Cut-Through Routing



相互結合網と並列システム性能

- ネットワークの性能メトリック
  - バンド幅 [Byte/sec]
  - レイテンシ [sec]
- 半性能長
  - N1/2: "理論ピーク性能の半分の性能を達成するメッセージ長"
  - バンド幅=a[Byte/sec], レイテンシ=b[sec]の時
    N1/2 = ab [Byte]
  - ex) Infiniband 4xDDR 2[GB/s], 1.5[μs]
    ⇒ 3[KB]
    GbEthernet 125[MB/s], 10[μs]
    ⇒ 1.25[KB]
- 「バンド幅は金(リンク幅)で買えるがレイテンシは買えない」
  - 大規模システムで通信距離が増えた場合、レイテンシが耐えられるか？



転送時間・遅延時間・バンド幅、アプリケーション性能

- (補助資料参照)

ノード性能とネットワーク性能

- 演算性能[Flops]:メモリ性能[B/s]:ネットワーク性能[B/s]
  - **"Trinity on Parallel Performance"**
    - 理想:メモリ～4Byte/flop, ネットワーク～1Byte/flop程度
  - 超並列計算機(MPP)
    - 例:CP-PACS 0.3[GFlops] : 1.2[GB/s]:0.3[GB/s]
      = 1 : 4 : 1 ⇒ 理想的(*「古き良き時代」*)
      BG/L 5.6[Gflops]:0.4[GB/s]:1[GB/s]
      = 1 : 0.07 : 0.18 ⇒ かなり悪い
  - ベクトル機は特にメモリ性能が命
    - 例:ES 64[GFlops]:256[GB/s]:12.5[GB/s]
      = 1 : 4 : 0.2 ⇒ メモリは良いがネットワークがやや悪い

ノード性能とネットワーク性能(続き)

- クラスタ、特に fat node なシステムでは厳しい
  - 例: 3GHz dual-core dual Xeon, DDR2, IB SDR
    = 24[GFlops]:6.4[GB/s]:1[GB/s]
    = 1 : 0.27 : 0.083 ⇒ かなり悪い
    PACS-CS (thin node)
    = 5.6[Gflops]:6.4[GB/s]:0.75[GB/s]
    = 1 : 1.1 : 0.13 ⇒クラスタよりMPPに近い
- これらの性能はLinpackではほとんど影響しない
  ⇒これが問題
  "Linpack-awareなマシンでは実効性能が期待できない"
- いずれの例でもネットワーク性能は全体的に下がってきている

ノード実効性能とネットワーク性能

- Fat-node PCクラスタと並列ベクトル機のネットワーク性能
  - Fat-node PC clusterは、見かけ上のピーク性能は高いが実効性能は低い
    ⇒ネットワークに求められる性能は意外に低い？
  - 並列ベクトル機は実効性能が高い分、実はネットワーク性能に求められる値も高くなる
- 並列システムにおける実効性能の観点から
  - ネットワークに求められる性能は、ピーク演算性能よりも、むしろメモリ性能との比較から考えるべきかもしれない
- I/Oの中でネットワークは大幅な性能向上を達成してきている
  ⇒I/O busのボトルネックが顕著
  ⇒今後、他のI/Oとは独立させるべき

アプリケーション特性による違い

- EP
  - parameter search, monte-carlo simulation等
  - 通信性能の影響をほとんど受けない
- 近接通信がドミナント
  - 物理空間分割（domain decomposition）：PDE差分, QCD等
  - 局所空間での高性能が要求されるが大域通信は甘い
- 遠距離・規則的通信がドミナント
  - FFT等
  - All-to-all通信が基本、遠距離通信が求められる
- ランダム通信がドミナント
  - （cell法でない）particle simulation等
  - 通信よりも演算の負荷分散を重視

大規模化に伴い性能維持が困難

並列計算機専用ネットワークの現状

- かつては様々な並列ネットワークが提案されたが生き残ったものはごく僅か
- 専用システム（proprietary network）として
  - BG/L, ASCI RedStorm 等
    - ではプロセッサノードに組み込まれたルータを利用して相対的に高バンド幅・低レイテンシなネットワークを提供
    - システムの拡張性のためにMesh/Torus型ネットワークを利用
  - IBM Powerシステム、富士通 HPC2500等
    - 100ノード規模のSMPを構築するためのfat node内ネットワーク＋ノード間ネットワーク
    - システムのノード規模は中規模、ノード間ネットワークのバンド幅はノード性能を支えるには十分ではない（ノード内並列が基本）
  - いずれのシステムでも、ノード性能の向上に対するネットワークバンド幅向上が課題、超並列システムでは特にレイテンシが厳しい

コモディティネットワークの現状

- クラスタ向けネットワーク（commodity network）
  - SAN：Infiniband, Myrinet
    - 高性能PCクラスタ構築の主流で、最近では1000ノードクラスまでのシステムを構築可能
    - SANのcommodity化により、NIC及びスイッチの低価格化が加速
      ⇒ Full-bisectionバンド幅のFat Treeも構築可能に
    - Infinibandの勢いが加速中。4xDDRが現在の主流で、今後8xQDR製品も市場に出てくる
  - 10G Ethernet
    - クラスタ向けよりも汎用ネットワーク／長距離ネットワークとしての利用が主流
    - スイッチの低価格化は進んでいるが、NICは必ずしもクラスタ向けではない
- 並列ネットワークとしてだけでなく、Storage用ネットワークとしての利用も進んでいる
  - 例：Lustre over Infiniband
- クラスタにおけるcommodity networkの利用では、今後の fat node化に対するバンド幅向上が最大の問題

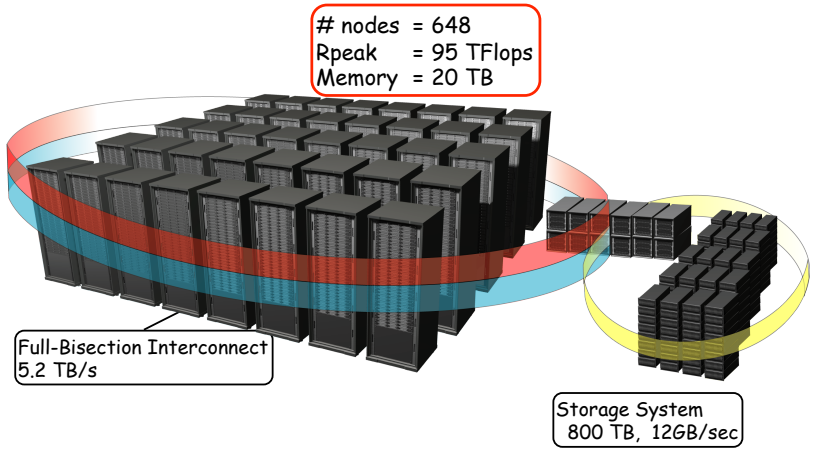
バンド幅向上のためのトランク技術

- commodity networkによるクラスタ構築の手段として、trunking技術が向上中
  - OpenMPI, MVAPICH等ではmulti-link利用が標準的に用いられている(Ethernet, Infiniband等)
  - 計算ノードのfat化(multi-core & multi-socket)に対応するために最も有効な手段
    例) 現在のhigh-end CPU (AMD Barcelona, Intel Clovertown)
    - quad core & 命令レベル並列性向上(dual SSE issue)
      ⇒ 32 ~ 48 GFLOPS / socket 程度、さらに dual ~ quad socket によりノード性能は100+ GFLOPS
  - 現在の高性能クラスタの主流はsingle-link SANだが、今後はmulti-linkが主流となっていくと思われる

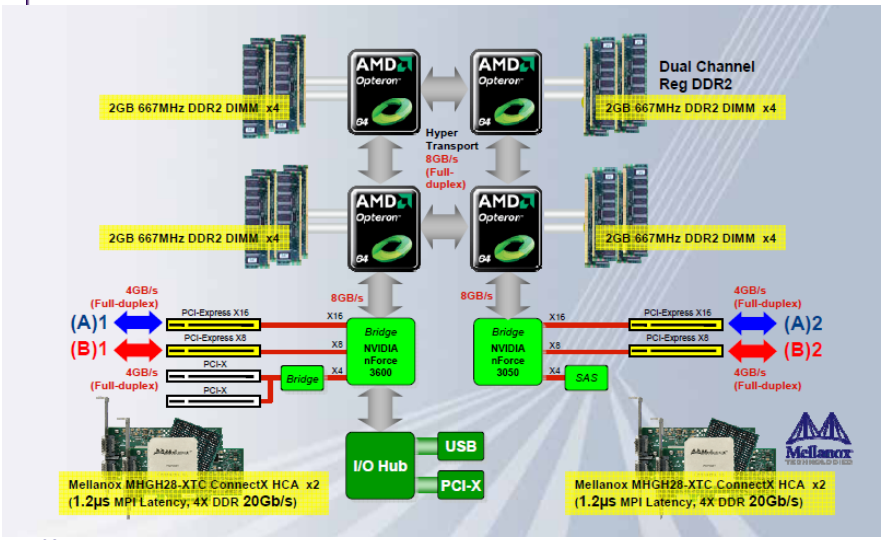
最先端commodity network技術の例

- T2K Open Supercomputer Alliance
  - 筑波大・東大・京大の連合によるスパコン調達
  - commodity processor + commodity network によるハイエンド・クラスタシステム
  - quad rail (multi-link trunking) による InfinibandまたはMyrinet結合
    = 5 ~ 8 GB/s network bandwidth / node
  - 基本的に full-bisection バンド幅での fat tree 結合
  - AMD Barcelona quad-core processor x 4 socket
    = 147GFLOPS/node, 40 GB/s memory

T2K Open Supercomputer
筑波大学構成 (Appro)

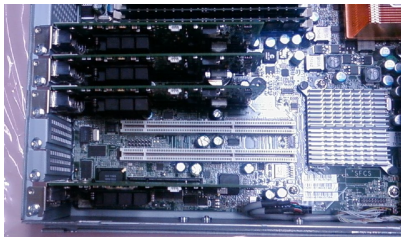


T2K筑波大システムのノード構成



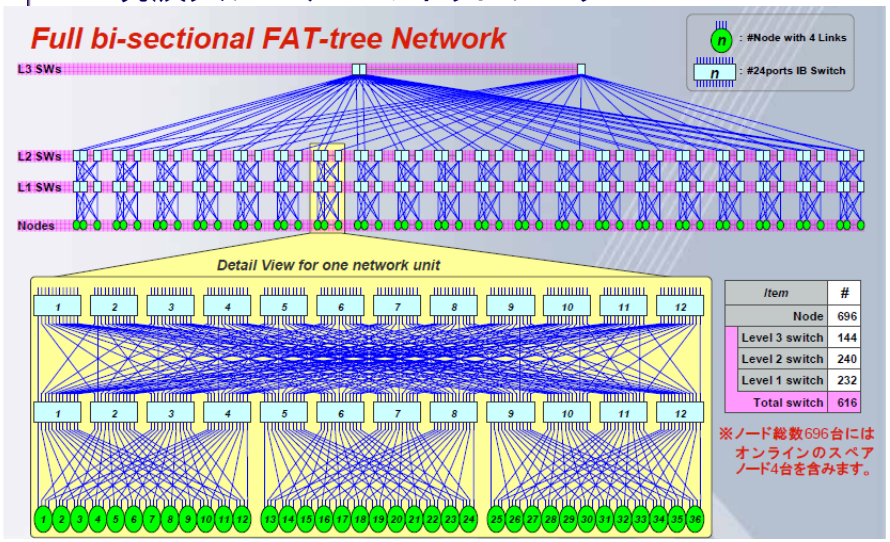
T2K筑波大システムの計算ノード

ノードシャーシ内部



Infiniband ConnectX x 4
(各PCI-Express x 8lane)

T2K筑波大システムのネットワーク



T2K-Tsukubaのネットワークの特徴

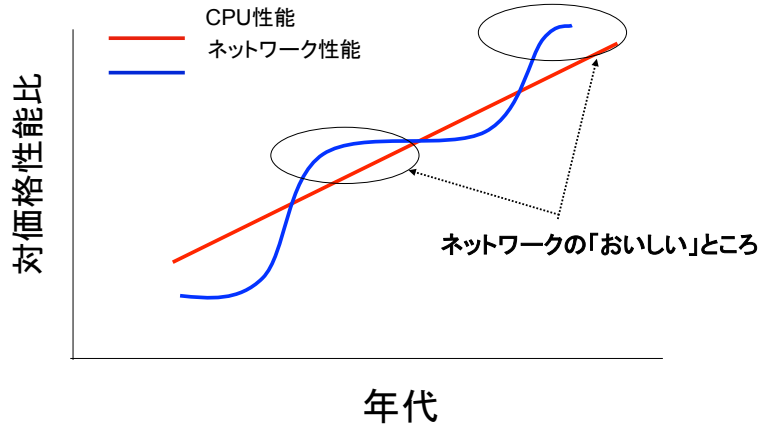
- Infiniband
  - 現在の高性能PCクラスタ向け相互結合網の代表
  - Ethernetに比べ性能当たり単価が安い（ポート当たりQDRの場合は4GB/secの片方向性能で20万円程度(2010現在)）
- Full-bisection bandwidth Fat-Tree Network
  - Fat-Tree構成で上位層スイッチの通信容量が下位層のそれと同等⇒上位層でのボトルネックがない
- 4-rail Infiniband
  - 4本の独立なInfiniband networkを平行結線してバンド幅を最大で4倍に増強
- ノード間並列ネットワークとI/Oネットワークを兼用
  - MPI等の通信以外にも、ファイルシステムのトラフィックが流れる

⇒これらの特性は最近の大規模PCクラスタの主流になっている

コモディティネットワークの今後

- HPCクラスタの成長をバックグラウンドに、対価格性能比が大幅に向上
- Storage & Parallel Network は非常に有効
- Infinibandは copper と optical の混在を許す（ケーブル側で選択する）
- しかし、クラスタの計算ノードのfat化に伴い、相対的な性能は低下
- Commodityの落とし穴⇒I/Oバス
  - 現状のベストはPCI-Expressだが、今後、仮にSANのバンド幅が数倍～十倍程度になった時にI/Oバスがボトルネックになる
  - これは、現在論じられている100G Ethernet等についても同様
- Multi-link & Multi-I/O busが成功の鍵
- いずれにしてもクラスタによる超並列化には電力性能の限界が見てくるのでは？
- スペース効率としては、今後のプロセッサのmany-core化は一つのソリューションだが Trinity Balance はますます崩れていく
  ⇒クラスタの実効性能向上のためには抜本的なアルゴリズム改良が必要になってくる

これまでのCPUとネットワークの対価格性能比の変化



CPUのmulti-core化の影響