Advanced Network Architecture Research

次世代高速・高品質インターネットのための技術課題

村田 正幸
大阪大学 大学院基礎工学研究科 情報数理系専攻
先進ネットワークアーキテクチャ研究室
e-mail: murata@ics.es.osaka-u.ajp
http://www-ana.ics.es.osaka-u.ac.jp/~ murata/

Advanced Network Architecture Research

インターネットの高速・高品質化に向けた技術課題；概観

- ネットワークに対する２つの見方
- 究極のアーキテクチャは存在するか?
- ATMの反省
- インターネットQoS；よくある間違い

Advanced Network Architecture Research

テレコムコミュニティの見方

- 「ネットワークが提供すべきは信頼性のあるコネクション型」
  - ✗ データを送る前にコネクション設定を行い、識別子を端末に与える
  - ✗ コネクションを設定した後、パラメータ、品質、コストに対する交渉を行う
  - ✗ 双方向通信、順番を保証したパケット転送を行う
  - ✗ 輻輳制御機能を提供する
- 主人公はネットワーク
- アプリケーション
  - ✗ 電話、動画像（テレビ会議）
  - ✗ リアルタイム（人と人とのコミュニケーション）

Advanced Network Architecture Research

インターネットコミュニティの見方

- 「ネットワークの仕事はビットを運ぶこと」
  - ✗いくらがんばってもネットワークが信頼性を確保することは難しい
  - ✗ホストはそれを受け入れてエラー制御をおこなう
  - ✗フロー制御は自分でする
- 主人公は*コンピュータ*
- アプリケーション
  - ✗telnet、ftp、WWW (http)
  - ✗バースト的、リクエスト／レスポンス

Advanced Network Architecture Research

複雑な制御をどこにおくか？

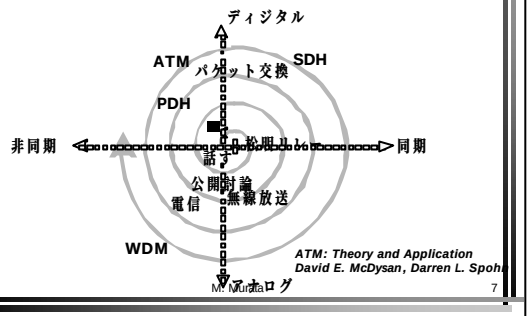
- コネクションレス型→トランスポート層（ホスト）
  - ✗ホストの処理能力の向上
  - ✗信頼性より高速な転送が必要なアプリケーションもある
- コネクション型→ネットワーク層（ノード）
  - ✗ユーザが希望するのは信頼性の高いトラブルのないサービス
  - ✗実時間音声や動画はコネクション型のほうが簡単

Advanced Network Architecture Research

究極の通信技術は存在するか？

- 「ATMはマルチメディア情報を統一的に扱う究極の統合通信網である」
  - ✗ マルチメディア情報とは何か？
  - ✗ 統合通信網は必要か？
  - ✗ エンドシステムは「端末」か？
- ネットワークの目的
  - ✗ 情報共有の手段（含：マルチメディア情報）
    - ✔情報発信、情報共有、情報流通、新しい（仮想）コミュニティの形成
  - ✗ コミュニケーションの手段
    - ✔音声から動画へ、メディアの多様化

通信の輪廻転生



ATM: Theory and Application
David E. McDysan, Darren L. Spoh

ATM (=Telecom)の反省

- ❑統計多重効果をはやくあきらめるべきだった
- ❑「何もかも扱う」という命題を背負った（背負わされた？）ところに問題がある
- ❑データ通信には原理的に向かない（次ページ参照）
  - ✗ データ端末はパケット交換か？
- ❑エンドシステムの軽視
  - ✗ コンピュータは端末ではない
- ❑ATMのAPIは解放されていたか？
  - ✗ インターネットが目の前にあったからこそ、それに適したWebというアプリケーションが生まれた
    - ✔背景：画像圧縮技術、GUI、画像表示能力
    - ✔にわとりと卵（？）

ネットワークの分類

	ベストエフォート型	品質保証型
コネクションレス型	*資源割当なし* インターネットIP	
コネクション型	*論理資源のみ割当* ATM UBRサービス ATM ABRサービス	*物理資源の割当* ハード保証型 ATM CBRサービス インターネットRSVP ソフト保証型 ATM VBRサービス

アプリケーションは重要か？

- ❑ネットワークインフラにとってアプリケーションは
  - ✗ ショーウインドウ
  - ✗ 画像アプリケーションが回線を埋め尽くすわけではない
- ❑重要なのはサービスアーキテクチャ
  - ✗ エンドシステム・アクセス系・バックボーンのどこで実現するかが問題
- ❑一億総プログラマの時代
  - ✗ 趣味・嗜好の細分化 → ネットワーク提供者がアプリケーションを考えるのは所詮無理
  - ✗ ユーザが自由にアプリケーションを作れるような環境を提供することが重要
  - ✗ 何が登場するかわからない → 単純なネットワーク構造が必要

データ系QoS：よくある間違い

- ❑ データ系のQoSはパケット棄却率・遅延
- ❑ データ系のQoS保証項目はアプリケーションレベルでの遅延
- ❑ アーラン呼損式に相当するのは待ち行列（網）理論に基づく結果
- ❑ TCPがあればネットワーク内部に輻輳制御メカニズムはいらない
- ❑ TCPはプロトコルとして同じ振る舞いが規定されているので、公平な通信サービスが提供される
- ❑ TCPはもはや古いので、新しい軽量・高性能プロトコルが必要
- ❑ WDMを導入すれば、ネットワーク速度は波長数分向上する

Advanced Network Architecture Research

高速・高品質化に向けた技術課題

- ❑ ~~課題１：実時間系QoS保証~~
- ❑ 課題２：バックボーンの高速化
- ❑ 課題３：プロトコルの高速化
- ❑ 課題４：エンドホストの高速化
- ❑ 課題５：ネットワーク機能の再配分
- ❑ 課題６：公平性の問題
- ❑ 課題７：ネットワークプロビジョニング
- ❑ 課題８：基礎理論はあるか？

データ系QoSとは？

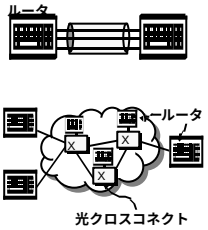
- ユーザの我慢の上に成り立っている
- データ系の通信保証は原理的に不可能
  - ✗「64Kbpsを保証する」＝
    「アクセス回線のみの保証」 or
    「呼損の発生」or
    「64Kbps以上は許さない」
- データ系サービスのQoS
  - ✗ 少なくともパケット遅延・棄却率ではない
  - ✗ ユーザレベルの遅延を「高速化」する
    例：Webのドキュメントダウンロード遅延
- 「インターネットサービス」を受け入れるには世代交代が必要（？）

課題2：バックボーンの高速化

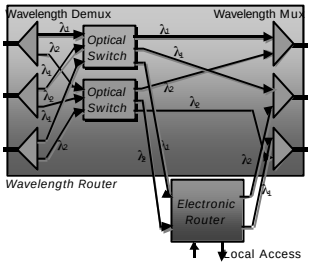
- WDMネットワークにすべてのネットワーク機能を取り込むのには無理がある
  - ✗ 例：コネクション設定、輻輳制御、経路制御
- ただし、IPの機能を一部取り込むことによって高信頼化・高性能化を実現することは可能
  - ✗ 高信頼化：代替経路を設定
    - ✔link protection
    - ✔dedicated-path protection
    - ✔shared-path protection
  - ✗ 高性能化：ルータによるパケットフォワード
    → 直接パスの設定
  - ✗ 前提はネットワークディメンジョニングがうまく機能すること
    - ✔波長ルーティングによる容量変更

フォトニックインターネットアーキテクチャ

- 4つのフォトニックネットワークアーキテクチャ
  1. WDMリンクネットワーク
     - ✔ルータ間をWDMリンクで接続
  2. WDMパスネットワーク
     - ✔ルータ接続（フォトニックパスによる論理ネットワークの構築）

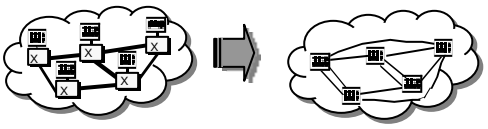


波長ルータの構成



波長パスによる論理トポロジーの構成

- 物理トポロジー
- 論理トポロジー
  - ✔ ルータから見ると冗長度の高いネットワークが構築可能

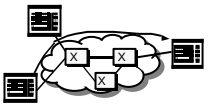


- 機能分担（信頼性技術）をどうするか？
  - ✗ IP層；経路制御
  - ✗ WDM層；パスプロテクション

フォトニックインターネットアーキテクチャ（続）

- 4つのフォトニックネットワークアーキテクチャ
  3. WDMパスネットワーク
     - ✔ラベルスイッチング（波長をラベルと見たMPLS）

  4. WDMパケットネットワーク
     - ✔例：バーストスイッチング（オンデマンド波長・経路選択）



1000波長WDMの実現可能性とその効果

❑ 与条件
- ✗ 物理トポロジー
- ✗ トラヒック量、トラヒック分布
  基本：電話網
  スケールファクタにより調整

❑ 物理トポロジーから論理トポロジーを生成
- ➥必要な波長数、ルータのパケット処理能力

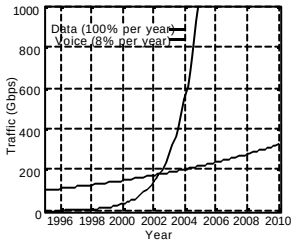

http://www.nttco.jp/databook/setubi/

❑ 村田正幸、北山研一、宮原秀夫（大阪大学）「1000波長WDMによるインターネットにおけるネットワークボトルネックの解消」発表予定

トラヒックマトリックス

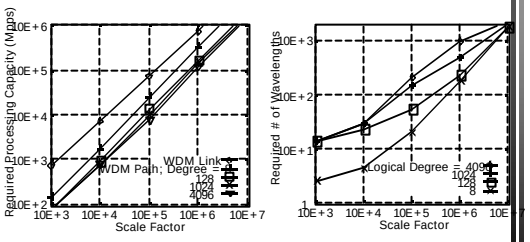
❑ NTTの電話網を基本とする
- ✗ NTT情報Webステーション
  http://www.ntt-east.co.jp/info-st/network/traffic/index.html
- ✗ 特徴
  - ✔近隣県間のトラヒック量≫遠距離県間のトラヒック量
    - ➥Zipfの法則（？）
  - ✔東京への一極集中
    - ○大阪→兵庫：大阪→東京：大阪→岩手=300：100：1
  - ✔トータル 30Gbps（1アーランを64Kbpsで換算）
    - ➥スケールファクタα を導入
- ✗ データ系トラヒック？

インターネットトラヒックの成長



❑ USの例
- ✗ 音声；年8%
  データ；年100%
  3年で1桁上昇
- ✗ K.G. Coffman and A.M. Odlyzko, "The size and growth rate of the Internet," http://www.researc.att.com/~amo

1000波長WDMによる効果



❑ ルータボトルネック！

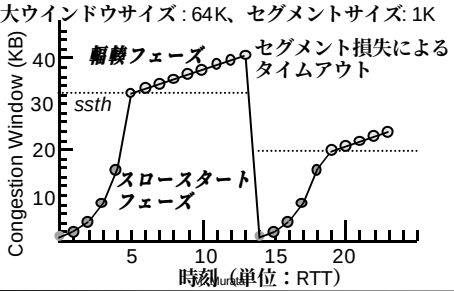
課題3：プロトコルの高速化

❑ 従来の議論
- ✗プロトコル処理のハードウェア化、パラレル化
- ✗軽量プロトコルの実現
- ✗エラー制御と輻輳制御の分離
- ➥Proprietaryなプロトコルはもはや通用しない

❑ TCPの性能向上？
- ✗TCP Tahoe → TCP Reno → TCP NewReno → TCP SACK → TCP Vegas (?)
- ✗プロトコルマイグレーションの確保
  - ✔送信側の変更は可能
  - ✔バージョンの異なるTCP混在時の性能？

TCPのウィンドウサイズの動き

最大ウインドウサイズ : 64K、セグメントサイズ: 1K



TCPのバージョン

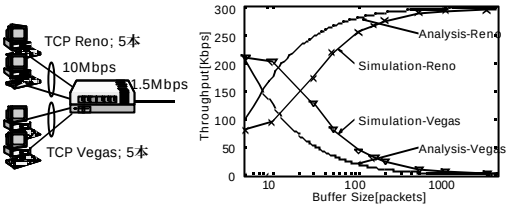
- ❑たまたま１個だけセグメントを落としただけならセグメントをすぐに再送
  fast retransmit; TCP Tahoe
- ❑ウィンドウサイズを半分にする
  fast recovery; TCP Reno
- ❑Go-Back-NからSelective Repeatへ；TCP SACK
- ❑RTTに基づいてウインドウサイズを調整する：TCP Vegas

$$diff = cwnd(t) / basertt - cwnd(t) / observed\_rtt$$

$$cwnd(t + t_A) = \begin{cases} cwnd(t) + 1, & \text{if } diff < a / base\_rtt \\ cwnd(t), & \text{if } a / base\_rtt \leqq diff \leqq b / base\_rtt \\ cwnd(t) - 1, & \text{if } b / base\_rtt < diff \end{cases}$$

TCP VegasとTCP Reno混在時のスループット比較

- ❑ TCP Vegasだけなら～40%のスループット向上



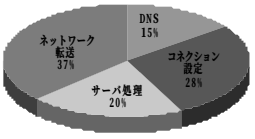
課題4：エンドシステムを含めた性能？

- ❑ パケット伝送時間がネットワーク性能を決めるわけではない
  - ✗回線容量
  - ✗ルータ処理能力
  - ✗ホスト（サーバ・クライアント）処理能力
  - ✗アプリケーション

アプリケーションの遅延配分

Webページのダウンロードの場合

- ❑ エンドシステムの重要性
- ❑ ネットワーク高速化の限界
- ❑ 重要なのはバランスのとれた資源配分

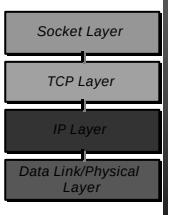


Produced from ftp://www.telcordia.com/pub/huitema/stats

参考文献：藤田 靖征, 村田 正幸, 宮原 秀夫, "Webサーバシステムのモデル化と性能評価," 電子情報通信学会論文誌, 1998.

高速プロトコル処理システム

- ❑ソケット層プロトコルの高速化（バッファ管理）
- ❑エンドシステムのプロトコル処理高速化（ゼロコピー）
- ❑ネットワークにおけるプロトコル高速化（TCPの輻輳制御メカニズム）
- ❑ネットワークの高速化（IP over WDM）



課題5：通信処理機能の再配分

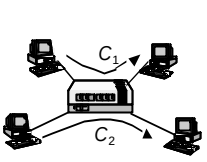
- ❑エンドホストに頼りすぎ
  - ✗ TCPによる輻輳制御
    - ✔輻輳制御は本来ネットワーク機能
  - ✗「公平なサービス」を阻害する
    - ✔輻輳制御を行わないホスト（バグ、コードの改変）
    - ✔「サービスの有料化」に対する障害
- ❑通信処理機能の再配分
  - ✗ フロー制御、誤り制御、輻輳制御、経路制御
  - ✗ RED、DRR、ECN、diff-serv、int-serv (RSVP)、ポリシールーティングは輻輳制御のネットワークへの回帰
  - ✗ ただし、過度の回帰はインターネットのメリットをなくす

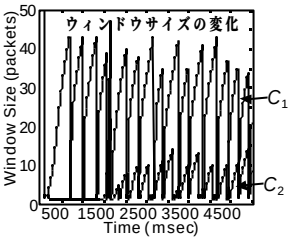
課題6：公平性の問題

- 資源が無限大になる時代がやってくることはない！
  - ✗帯域を埋めるようなアプリケーションがあれば、ネットワークが機能しないのは自明
  - ✗データ系は帯域に合わせて送信する
- 帯域をいかに公平に分配するか？
  - ✗TCPの輻輳制御は公平ではない
    - ✔いったんウィンドウサイズを下げると（短期的に）大きくならない
    - ✔RTT、帯域による不公平性
    - ➥ルータによる処理の必要性；RED、DRR
  - ✗実時間アプリケーションとデータ系アプリケーションの帯域の配分？

TCPコネクション間の公平性：伝播遅延時間が異なる場合

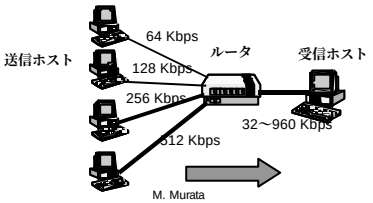
- $2\tau_1 = \tau_2$の場合





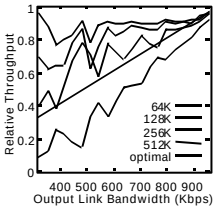
TCPコネクション間の公平性：回線容量が異なる場合

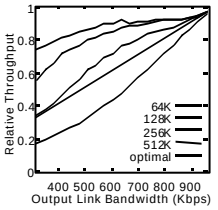
- 出力回線容量に対して、接続回線容量に応じたスループットが得られるか？
  - ✗ Relative Throughput = 回線容量を1とした場合のスループット



TCPコネクション間の公平性：回線容量が異なる場合の結果例

- ルータがDrop Tailの場合
- ルータがREDの場合





課題7：ネットワークプロビジョニング

- 現状はネットワークトラヒックの振る舞いの把握と分析
  - ✗ “Cooperative Association for Internet Data Analysis,” http://www.caida.org/
  - ✗ “Internet Performance Measurement and Analysis Project,” http://www.merit.edu/ipma/
- 意味のあるデータを拾い出せるか？
  - ✗ 経路制御による非安定性
  - ✗ TCPのセグメント再送
  - ✗ 実時間アプリケーションのレート適応・遅延適応制御
  - ✗ 低利用率は
    - ✔輻輳制御のため？
    - ✔低速アクセス回線のため？
    - ✔低速エンドホストのため？

アーラン呼損式はなぜ信用できるのか

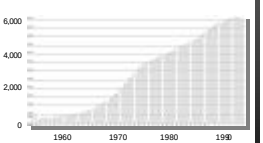
目標呼損率→回線容量
トラヒック測定→回線増強

① 呼損率＝ユーザ品質
② 安定した成長：統計データに関する過去の蓄積
③ アーラン呼損式

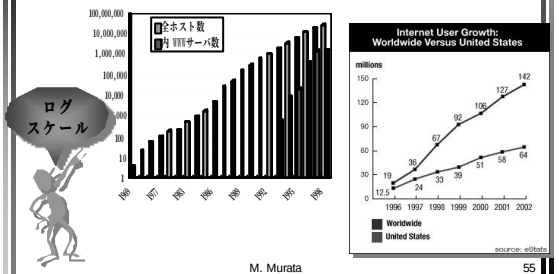
$$B = \frac{\rho^s/s!}{\sum_{i=0}^{s} \rho^i/i!}$$

i. サービス時間は一般分布→ローが妥当
ii. ポアソン到着は普遍的な事実

④ 品質測定＝呼損率を測る



インターネットの伸び

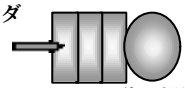


ネットワークプロビジョニングの課題

- （少なくとも）ネットワークプロビジョニングレベルでの品質予測
- 回線交換にない新たな問題
  - ✗ 品質とは何か？
  - ✗ 品質を測定できるか？
  - ✗ サービス品質の課金への反映？
  - ✗ マルチメディアトラヒックの予測の困難性
  - ✗ ネットワーク測定では不十分
- ネットワークトラヒック測定→分析→回線容量設計
  - ✗ フィードバックループを前提としたネットワーク設計論の確立
  - ✗ 柔軟な帯域設定を持つネットワークが大前提（ATM、フォトニックネットワーク）
  - ✗ Webトラヒック、Webサーバ、ネットワーク遅延 (RTT) のモデル化

課題8：パケット交換網の基礎理論？

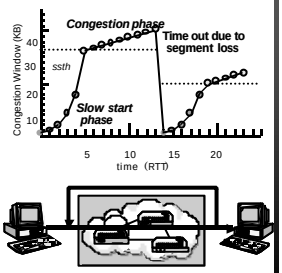
- M/M/1（待ち行列網理論）パラダイムは役に立つのか？
  - ✗ わかるのはノードにおけるパケット待ち時間、棄却率
    - ➥しかし、データ系のQoSはノードにおけるパケット待ち時間ではない
  - ✗ アーラン呼損式（＝電話網）では
    呼損率 ＝ ユーザレベルQoS



- ルータでの振る舞い？
  - ✗ フィードバック系が上位レベルにある時の振る舞いが重要
- ユーザレベルのQoS？
  - ✗ アプリケーションレベルの性能指標が重要
    - ✔e.g. Webドキュメント応答時間

インターネットはフィードバック系

- データ系
  - ✗TCPによる輻輳制御；輻輳に応じたウインドウサイズの制御
- 実時間系
  - ✗UDPを用いた遅延適応、レート適応制御
  - ✗RSVPベースの予約型；アーラン呼損式
- ➥制御理論
  - ✗安定性、収束性



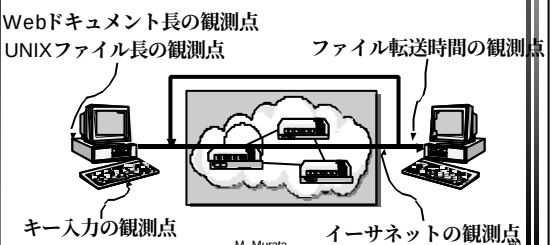
ネットワークの基礎理論の構築（１）

- 輻輳制御
  - ✗ 時間に依存したふるまい
  - ✗ フィードバックシステム：制御理論
- フィードバック系を前提にしたトラヒック測定・分析・ネットワークディメンジョニング → ネットワーク回線容量設計
- エンドシステムを包含したQoSアーキテクチャの構築

- 回線容量設計をするには
  - ➠いかにアプリケーション、エンドシステムの影響を除いた測定、モデル化を実現するか？
- 高品質なネットワークを構築するには
  - ➠アプリケーション、エンドシステムを含めたQoS設計

ネットワークの基礎理論の構築（２）

- トラヒックの長期依存性・自己相似性



ネットワークの基礎理論の構築（３）

- 複雑化・巨大化したシステムの評価手法
  - ✗ 理論、シミュレーション
  - ✗ 待時系・即時系混合待ち行列網
    - ✔ 即時系コネクションは各ノードの資源を同時に使用
    - ✔ 残りを待時系パケットが使用
  - ✗ 評価モデル？
    - ✔ シミュレーション、解析

    例：バックボーンにおけるTCPのふるまい？