ネットワークモデリングと
その応用によるネットワークの進化論的解析

無断複写

報告番号 甲第 4285 号

名古屋大学大学院工学研究科
博士学位論文

ネットワークモデリングと
それを用いたネットワークの通信量評価

石原 進

Doctoral Dissertation of Graduate School of Engineering, Nagoya University

Simulation Modeling and Performance Evaluation of Local Area Networks

Susumu ISHIHARA

目 次

第1章 序 論 **5**

1.1 ネットワーク性能評価とその必要性 5

1.2 教育用計算機環境のネットワーク 6

1.3 ネットワークシミュレーションの有用性 8

1.4 ネットワーク性能指標 11

1.5 教育用システムに適した LAN 構成 11

第2章 過渡的な負荷を持つネットワークのモデル化 **13**

2.1 はじめに 13

2.2 ネットワークプロトコルの概説 16

2.2.1 Ethernet 17

2.2.2 TCP/IP 22

2.3 シミュレーションモデル 27

2.3.1 多階層モデリングの手法 27

2.3.2 Ethernet サブモデル 29

2.3.3 IP サブモデル 39

2.3.4 TCP サブモデル 39

2.4 シミュレーションモデルの検証 41

2.4.1 過渡的な負荷を持つ LAN を想定した条件 41

2.4.2 アプリケーションモデル 42

2.4.3 シミュレーション結果 43

2.5 まとめ 52

第3章 可視化に適したネットワーク性能指標の提案 **53**

3.1 はじめに 53

3.2 ストール率 55
3.2.1 CSMA/CD 方式における送信遅延 55
3.2.2 ストール率の定義 55
3.2.3 ストール率とトラヒック特性 56
3.3 アプリケーション利用時の Ethernet 特性評価 65
3.3.1 場面設定 65
3.3.2 シミュレーション結果 67
3.3.3 ユーザストール率 78
3.4 むすび 85

第4章 教育用 LAN の性能評価 **87**
4.1 はじめに 87
4.2 教育用 LAN の負荷集中のモデル化 88
4.2.1 シミュレーションの場面設定 88
4.2.2 シミュレーションモデル 89
4.2.3 ユーザ挙動モデル 89
4.2.4 アプリケーションモデル 90
4.2.5 接続方式 90
4.3 シミュレーション結果 93
4.3.1 応答時間 93
4.3.2 教育用 LAN の適切な設計指針 108
4.4 まとめ 109

第5章 結 論 **111**
5.1 本研究のまとめ 111
5.2 今後の展望と課題 112

参考文献 **115**

論文目録 **121**

図目次

2.1 Ethernet CSMA/CD 19
2.2 ギガビット Ethernet における CSMA/CD の拡張 22
2.3 TCP のスライディングウインドウ機構 25
2.4 TCP セグメントの送信 25
2.5 TCP セグメントの再送信 26
2.6 シミュレーションモデルの全体構成 28
2.7 Ethernet サブモデルの状態遷移図 30
2.8 Ethernet サブモデルの制限 37
2.9 シミュレーションしたネットワーク構成 41
2.10 サーバプログラムの送信バッファサイズに対する平均サーバ応答時間の変化 44
2.11 CWS 台数に対する平均サーバ応答時間の変化 47
2.12 CWS 台数に対する平均衝突率と TCP 再送回数の変化 50

3.1 連続衝突回数に対する T_{CS}, T_{WT} の変化 57
3.2 データトラヒック発生時の平均ストール率 60
3.3 ユーザ挙動モデル 66
3.4 アプリケーション利用時の平均サーバ応答時間とネットワーク平均ストール率 68
3.5 クライアントとサーバのストール率と衝突率 72
3.6 NFS の平均応答時間と平均要求再送回数 78
3.7 ユーザストール率 80

4.1 ユーザ挙動モデル 90
4.2 検討対象とした接続方式 92
4.3 サーバ応答時間 (Shared-10M) 94

4.4 サーバ応答時間 (Shared-100M) 95
4.5 サーバ応答時間 (Shared-1G) 96
4.6 サーバ応答時間 (Switched-10M) 97
4.7 サーバ応答時間 (Switched-100M) 98
4.8 サーバ応答時間 (BigPipe-10M/100M) 99
4.9 媒体共有型ギガビット Ethernet と媒体共有型 100Mbps Ethernet との比較 107
4.10 教育用 LAN における Ethernet のコストと性能の関係 109

表目次

2.1 OSI 7 階層参照モデル 23
2.2 モデルで仮定したプロトコル 27
2.3 Ethernet サブモデルの状態遷移 30
2.4 ユーザ挙動モデルのパラメータ 42
2.5 サーバプログラムの送信バッファサイズに対する平均サーバ応答時間の変化 45
2.6 CWS 台数に対する平均サーバ応答時間の変化 (Model M) 48
2.7 CWS 台数に対する平均サーバ応答時間の変化 (Model K) 49
2.8 CWS 台数に対する平均サーバ応答時間の変化 (Model O) 49
2.9 シミュレーションの実行回数 50
2.10 CWS 台数に対する平均衝突率と TCP 再送回数の変化 51

3.1 連続衝突回数に対する $T_{\mathrm{CS}}, T_{\mathrm{WT}}$ の変化 58
3.2 パケット長分布 58
3.3 データトラヒック発生時の平均ストール率 (端末数 = 5) 61
3.4 データトラヒック発生時の平均ストール率 (端末数 = 10) 62
3.5 データトラヒック発生時の平均ストール率 (端末数 = 20) 63
3.6 データトラヒック発生時の平均ストール率 (端末数 = 40) 64
3.7 アプリケーション利用時の平均サーバ応答時間とネットワーク平均ストール率 (FTP) 69
3.8 アプリケーション利用時の平均サーバ応答時間とネットワーク平均ストール率 (NFS) 70
3.9 FTP クライアントのストール率と衝突率 73
3.10 FTP サーバ のストール率と衝突率 74
3.11 NFS クライアントのストール率と衝突率 75

3.12 NFS サーバのストール率と衝突率 76
3.13 NFS リクエスト再送率と応答時間 77
3.14 FTP クライアントのユーザストール率 81
3.15 FTP サーバ のユーザストール率 82
3.16 NFS クライアント のユーザストール率 83
3.17 NFS サーバ のユーザストール率 84

4.1 検討対象とした接続方式 91
4.2 平均サーバ応答時間 (Shared-10M) 100
4.3 平均サーバ応答時間 (Shared-100M) 101
4.4 平均サーバ応答時間 (Shared-1G) 102
4.5 平均サーバ応答時間 (Switched-10M) 103
4.6 平均サーバ応答時間 (Switched-100M) 104
4.7 平均サーバ応答時間 (BigPipe-10M/100M) 105
4.8 平均サーバ応答時間 (Shared-1G, フレームバースティングを行わない場合)106
4.9 適切なクライアント台数のめやす 108

あらまし

本論文は LAN のシミュレーションモデルの設計と，過渡的な変動負荷および高負荷なローカルエリアネットワーク（LAN）の性能評価に関する研究についてまとめたものである。

LAN の普及，小型ワークステーションの高速化，分散コンピューティング・パラダイムにより，現在，大学等の情報処理教育センターのような集合型情報処理教育施設の多くは分散型システムによって構築されるようになった。分散環境を導入した教育用システムの多くでは，分散ファイルシステムを用いてユーザの個人データおよび共用データを 1 台ないしは複数のファイルサーバによって一元管理するファイル管理手法が用いられている。これにより，利用者は多数台のユーザ用クライアント計算機のどれを利用してもつねに同一の個人環境が利用可能になる。また，ファイルサーバでデータを一元管理するために，システム管理が容易になるという利点がある。しかしながら教育用環境では授業利用時に多数のユーザが同時にシステムを利用するという特性がある。例えば，教官の合図のもとに一斉に計算機にログインしたり，ファイルサーバ上の同一ファイルを読み込むことが日常的に行われる。このような利用状況ではファイルサーバおよびネットワークに過渡的な負荷変動，短時間に集中した高負荷が発生することになる。

教育用システムにおいては，上記の要因によるユーザ操作に対する応答時間の増大を防ぎ，かつ効率的なファイル管理を行うため，教育用システムに対応した分散ファイルシステムが必要であろう。筆者らの研究室では，階層化構造をもつ分散ファイルシステムを基本とした教育用システム専用の分散ファイルシステムの構築を目指して研究を進めてきた。本研究はそのシステム設計の基礎検討の一部として位置付けられ，教育用システムに適した LAN の構成方式の検討，設計手法の確立を目的としている。

教育用システムに限らず，より効率的なネットワーク設計を目的としたネットワーク性能評価の重要性はますます高まっている。今日，イントラネットの普及により，企業，大学等では全組織にわたり LAN が広がり，多くの一般利用者用計算機，サーバ計算機が

LANに接続されている。イントラネットはネットワークサービスの利用による作業全般の効率化を目的としている。イントラネットの導入課程における初期段階では計算機をネットワークに接続し，サービスを可能にすることが第一の目的とされた。しかしながら，イントラネットが普及し成熟しつつある現在，多くの組織でネットワーク自体の価格性能費を再検討し，より効率的なネットワーク構成を模索する動きが高まっている。

ネットワーク性能評価の要求が高まる一方で，現実の計算機利用条件を考慮したネットワークの性能評価手法は確立されたとは言いがたい。電話網の性能評価以来ネットワークの性能評価に用いられてきた待ち行列理論による解析的手法は，複数階層のプロトコルによって制御される現在の計算機ネットワークをモデル化する能力は十分ではない。また，大規模ネットワークを試作し，実システムによる性能評価を行うことはコスト的に現実的ではない。そこで，現在シミュレーションによる性能評価が脚光を浴びつつある。シミュレーションでは，対象をいくらでも詳細にモデル化可能であるが，その分計算コストがかかるので，評価する対象によってモデル化の手法および詳細度を調節する必要がある。

本文ではまず，教育用システムのような過渡的な負荷変動および高負荷なトラヒック特性を持つ Ethernet LAN の性能評価を指向したイベント駆動型のシミュレーション手法を提案する。本手法では LAN 上のノードは階層化されたプロトコル群の詳細なモデルの集合として表される。各ノードのプロトコルモデルのイベント処理を平行に行うことにより，シミュレーションを行う。教育用システムにおける LAN では極めて高い時間的負荷集中および過渡的な変動負荷が起こり得る。従来のアプリケーションの挙動を含めたシミュレーションのためのモデルでは Ethernet などのデータリンク層以下のプロトコルを簡略化していたのに対し，提案モデルでは Ethernet のモデルを詳細に記述する一方で，Ethernet のフレーム衝突判定における処理の高速化を行っており，高精度かつ高速なシミュレーションが可能である。提案手法はこのような教育システムにおける LAN に限らず過渡的な負荷変動が起きるそのほかの LAN の性能評価に有用である。本モデルの妥当性は，現実の LAN における性能測定結果との比較によって行われ，良好な結果が得られた。

さて，ネットワークの性能を数値的に判断するための道具として，従来スループットや帯域利用率，エンド・トゥ・エンド遅延などの性能評価指標が用いられてきた。これらの指標はあくまでネットワークの性能そのものを表すが，それらが示すネットワーク性能上の問題点を発見するには熟練した専門家が複数の性能指標を総合的に判断する必要がある。しかしながら，低価格な LAN の普及に伴い，ネットワークの設計，管理は専門家の

みの仕事ではなくなってきている。彼らにとって複数の性能指標からネットワークの性能上の問題点を推測することは困難であるため，より直観的に問題点を発見できるような性能指標および支援システムが必要であると筆者は考える。

本論文では，専門家以外が直観的にネットワークシステムの問題点を発見することを容易にする性能指標として，可視化に適した CSMA/CD LAN のための性能指標・ストール率を提案する。ストール率は Ethernet に代表される CSMA/CD LAN におけるパケット送信が滞る時間，すなわち伝送路の空きを待つ時間，衝突検出による JAM 信号の送出時間，衝突検出後のバックオフによる待ち時間がパケット送信時間中に示す割合を示したものである。これらの割合は同一のグラフ中に表すことが容易であり，各割合の比からパケットの衝突の大量発生や，単一ノードの伝送路の占有などの CSMA/CD LAN における問題の発見，理解が可能である。また，ストール率はパケット送信時間を MAC(Media Access Control) 層におけるパケット送信開始からパケット送信の終了までの時間とする単純なストール率と，パケット送信時間を MAC 層へのパケット入力から MAC 層におけるパケット送信完了までの時間とするユーザストール率の二つの定義がある。これらを用いることにより，それぞれシステム管理者の視点，ユーザの視点から性能評価を行うことが可能である。

本文では従来の性能指標との比較，および教育用システムの LAN の性能評価への適用によってストール率の特質を明らかにした。ストール率の利点は直観的なプロトコル性能および問題点の理解の容易さにあるが，システム性能の定量的評価は困難である。定量的評価を行うためには従来の性能指標を組み合わせて使う必要があるだろう。また，ストール率が示すシステムの問題点は CSMA/CD LAN で発生する問題の一側面を表しているに過ぎない。キャプチャー効果に代表される Ethernet CSMA/CD における問題や，動的なシステム性能の変化に対応した簡便な性能指標が今後必要となろう。

本論文の最後では，教育用システムを指向した分散型ファイルシステム設計のための基礎検討として，最初に述べたシミュレーションモデルを用いて教育用システムに適した LAN 構成の検討を行う。検討対象として従来の CSMA/CD による媒体共有型とスイッチング型の Ethernet に基づく 6 種類のネットワークトポロジを取り上げた。教育用システムの特徴である過渡的な負荷集中のモデルを，現実の教育用 LAN におけるユーザ挙動の測定に基づいて設計し，これに基づいて 1 台のファイルサーバと 1 ～ 150 台のクライアント・ワークステーション間での NFS および FTP によるファイル転送のシミュレーションを行った。シミュレーション結果より，教育用システムでの負荷集中条件に適した

クライアント・ワークステーション台数を得た。また，性能・コスト両面からの評価を行った結果，教育用システムにおける Ethernet による LAN 構成では，100 Mbps 以上の媒体共有型または スイッチングハブによる 10 Mbps と 100 Mbps の組合せによるビッグパイプ型のネットワーク構成が価格性能比に優れているという結果が得られた。

本研究では，教育用システムを指向した分散型ファイルシステム設計のための基礎研究として，(i) シミュレーションモデルの提案，(ii) 可視化に適したネットワーク性能指標の提案，(iii) 教育用システムに適した LAN 構成の検討を行った。本研究の成果は分散型ファイルシステムの利用を想定した教育用システムでの LAN 構成の指針を与えるものとして意義があるものである。しかしながら，現在加速度的に進みつつある LAN 構成の多様化，マルチメディア通信によるトラヒックに対する対処，およびモーバイル環境を統合したネットワーク構成の検討などには，さらなる研究の必要があろう。

第1章　序　論

本章では，まず本研究の背景となるローカルエリアネットワークの現状について概観し，ネットワーク性能評価の必要性について述べる。また，本研究の動機となった教育用計算機システムにおけるネットワークの問題について述べ，本研究の学問的位置づけを明らかにする。次にネットワークの性能評価手法における歴史的背景および問題点を明らかにし，本研究の各論であるネットワークシミュレーションモデルおよびユーザ志向のネットワーク性能評価指標の概要に触れる。最後に本研究の主題である教育用計算機システムにおけるネットワークの設計手法の概要について述べる。

1.1　ネットワーク性能評価とその必要性

コンピュータネットワークの普及は目覚しい。30 年前にコンピュータネットワークに接続されていたコンピュータは極めて少数であった。しかし，現在ではほとんどの大学，多くの企業のコンピュータはローカルエリアネットワーク (LAN) に接続されており，またその LAN はインターネットに接続されている。さらに，家庭用コンピュータや携帯用コンピュータなど，あらゆるコンピュータがインターネットに接続されている。いまやコンピュータネットワークは，道路や水道のように現代社会に不可欠な社会的基盤であると言っても過言ではないだろう。この爆発的なコンピュータネットワークおよびコンピュータの普及は数多くの新技術を生み，新たな学問領域を切り開くことになった。CERN（欧州素粒子研究所：Conseil Europen pour la Recherche Nuclaire）によって開発された World-Wide Web (WWW) が現在におけるインターネットの爆発的な普及の引金となり，

数多くの応用技術が生まれたことはよく知られていることである。

さて，コンピュータネットワークといってもその構成形態は2台のコンピュータを接続した2点間通信から，Ethernetに代表されるLAN，そしてインターネットまで多岐にわたる。本論文で対象としているLANに限ってみても，良く知られているEthernetだけでなく，Token Ring，ATM LANなど様々な方式がある。さらに，Ethernetにも1 Mbps，10 Mbps，100 Mbps，1Gbpsと異なる伝送速度をもつものがあり，接続方式も従来のCSMA/CDによる媒体共有型とスイッチング型の2つの方式がある。これらの方式を組み合わせて，現実にネットワークを構成するとなると無数の実現方式が考えられる。コンピュータネットワークに限らず，あらゆるシステムを設計・構築する場合にはそのシステムの性能を明らかにする必要がある。無数の選択肢の中から高性能なネットワーク実現手法を選択するために，性能評価の重要性は非常に高い。また，一度設計したシステムのチューニングを行い，システムの利用条件に合わせてシステムの性能を高く保つためにも性能評価は欠かせない。

近年のイントラネットの普及により，特にLANの性能評価の必要性が高まってきた。ツイストペアケーブルによる接続が可能となったEthernet 10BASE-Tネットワークにより，LANへの接続性は従来のものに較べて格段に向上した。このため，ネットワークの専門家以外でも容易にLANを構成することが可能となり，LANに接続される計算機の数は飛躍的に増大した。その一方で増え続けるネットワークトラヒックをいかにして分散させるのか，サーバ計算機を適切に配置するにはどうすればよいのか，対投資効果を最大にするにはどのような構成が良いのかといった問題が浮上した。これらの諸問題は，クライアント・サーバシステムの性能試験システムや，種々のネットワークシミュレータの誕生を促した。

1.2 教育用計算機環境のネットワーク

LANにおけるトラヒックの発生パターンはLANが利用される環境によって異なると考えることができるが，そのなかでも興味深いのは集合型情報処理教育環境におけるLANである。

教育用システムにおけるLANの大きな特色として，短時間に極端な負荷集中が発生することが挙げられる。多数の学生を一度に指導する日本で良く見られる指導形態では，教官の指示のもとに多くの学生が同じネットワークに接続されたコンピュータの操作を同時

に行う。この結果として短時間に大量のネットワークアクセスが多くの端末から発生することになるが，このような状況は一般の業務用・研究用のシステムのように各端末のユーザが独立して操作を行う状況とは大きく異なる。

情報処理教育の重要性が高まった1980年代初頭に，日本全国の大学に情報処理教育センターなどの集合型情報処理教育施設が設置されるようになった。これらの施設におけるコンピュータシステム(教育用システム)は，当初，大型汎用計算機とそれにLAN接続された多数の端末により構成されていた。このような形態のシステムでは，授業利用時には百数十人のユーザが同一の計算機を利用して計算を行っていた。当然ながら汎用計算機における計算負荷は集中し，ユーザ操作に対する応答時間は長かった。この時点では教育用システムにおけるLANが応答時間のボトルネックとなることはなかった。

低価格のPC，ワークステーションの能力の向上および分散化コンピューティングパラダイムの普及により，現在の教育用システムは分散型のシステムで構成されるようになった。すなわち，ユーザの要求する計算の大半はユーザが利用するクライアントワークステーションで処理され，ファイルアクセス，プリント要求などが専用のサーバで処理されるようになった。これによって，ユーザの計算要求が単一の計算機に集中しなくなり，ユーザ操作に対する応答時間におけるボトルネックが計算時間となることはなくなった。その一方で新たなボトルネックとなったのが，LANにおける通信時間である。特に，広く普及しているLANの構成方式の一つであるEthernetは媒体共有型のLANであり，多数台の端末からの同時アクセスに対しては十分に高い性能を発揮できないという性質を持つ。

筆者の所属する研究室では，上記のような教育用システムにおけるLANへの負荷集中を避けるため，教育用システム専用の分散ファイルシステムの構築を計画している。分散システムを用いた教育用システムでは，Network File System (NFS)などの分散ファイルシステムを用いて1台ないしは複数台のファイルサーバで利用者の個人データや共有データを一元管理する手法が用いられる。これにより，利用者は多数台ある利用者用クライアントワークステーションのどれを利用しても，常に自分専用の環境を利用でき，また管理者はデータを1箇所で管理できるので管理が容易に行える。しかしながら，前述のLANへの負荷集中を考慮した場合，個人用データや共用データを一箇所にまとめて管理することは，それを管理するサーバやそれが属するネットワークへのアクセス集中を招くことになる。教育用システム専用の分散ファイルシステムの基本的なアイデアは，ネットワークを階層的に分割し，分割されたネットワーク毎にファイルサーバを分散配置する

ことである。分散配置されたファイルサーバは互いに連係してファイルの一貫性を保ち，ユーザ移動性に対応する。同様のアイデアはカーネギーメロン大学によって開発されたAndrew File System でも用いられている。

さて，このような階層型分散型ファイルシステムを設計するにあたって必要となる技術として，i) 分散して配置されたファイルサーバおよびそれが管理するファイルの管理手法，ii) 効率的なネットワーク分割手法，が挙げられる。本研究はこの教育システム専用の分散ファイルシステムの構築のための基礎検討として，ii) の部分を担うものである。効率的なネットワーク分割手法を検討するため，まず本研究では教育用システムのような過渡的な変動負荷，高負荷をもつネットワーク性能評価手法の提案を第 2 章，3 章で行い，それらを用いた各種のネットワーク構成における性能評価および効率的なネットワーク構成の検討を第 4 章で行う。

1.3 ネットワークシミュレーションの有用性

本研究で扱う教育用システムにおける LAN（教育用 LAN）の性能評価を行うためには以下の問題を考慮する必要がある。

- 過渡的な負荷変動，高負荷が起きた場合でも正確な評価が可能であること。
- 現実的なアプリケーション利用環境を想定した評価が可能であること。
  - ユーザの挙動
  - アプリケーションの挙動
  - 各階層のアプリケーションの挙動

これらの問題に対処し，正確な性能評価を行うため，本研究では計算機シミュレーションによる手法を用いた。以下，シミュレーション以外の評価手法の問題点と，シミュレーション手法に関する歴史的背景およびその必要性について概説する。

ネットワークの性能評価の手段には大きく次の 3 つの手法がある [村田 94]。

- 待ち行列理論などによる解析的手法
- 実システムによる性能測定
- 計算機シミュレーション

ネットワーク性能評価の手段としては従来から待ち行列理論が広く利用されてきた。待ち行列理論は電話交換システム構築のための理論的支柱として 20 世紀初頭にアーランによって創始された。待ち行列理論は電話交換網の設計問題をはじめ，交通問題，窓口問題，工程管理などの分野で広く利用されてきたが，コンピュータネットワークの分野では 1969 年の ARPANET の登場に始まるパケット交換ネットワークの設計問題への適用から利用されるようになった。しかしながら，待ち行列理論ではモデル化可能なプロトコルおよびトラヒックパターンが制限される。ペトリネットなどによる他の解析的手法でも程度の大小はあれ同様のことが言える。このため，今日の多階層のプロトコルの組合せによって実現される計算機ネットワークの性能評価を解析的手法で行うのは事実上困難である。もちろん，問題を簡単化することによって解決可能な問題 [DM92][MT88][ICT93][BSP90] はあるが，本論文で扱う利用者の挙動やアプリケーションプログラムの挙動を含むモデルを解析的手法で扱える形にモデル化することは困難である。また，解析的評価では，定常負荷における平均的なシステム性能を求めることしかできないので，教育用 LAN のような過渡的な負荷変動が起きる条件でのシステム評価はできない。

実システムによる性能測定は測定される性能値の正しさという点からは，他の手法よりも明らかに優れている。しかしながら，大規模システムを性能評価のために試作することはコスト的に困難である場合が多く，現実的な性能評価手法とはなり得ない。

解析的手法ではモデル化可能なシステムが制限されるのに対し，計算機シミュレーションによる評価では，モデル化は理論上際限なく精密に行うことができる。例えば，大規模な通信網の特性解明，複雑なスケジューリングを含む交換機制御の性能評価，輻輳制御やルーティグ制御の特性評価には，待ち行列理論などの解析的手法の適用は限られている [秋丸 90]。シミュレーションによる評価手法はこれら解析的手法の適用が困難なシステムの特性評価に用いられる。本論文で扱う教育用 LAN での負荷集中時における性能評価もこれに含まれる。

基本的にネットワークシステムは以下のような抽象化が行える。すなわち，各ユーザはデマンド（例えばパケット）を発生する。デマンドは資源（通信回線）を定められた順番に，あるいは複数の資源から同時に，サービス（パケット転送）を受けながら渡り歩く。また，各資源では定められた処理規律にしたがってサービスを受ける。統計的な振舞いを示す事象は疑似乱数によってその時刻を決定し，デマンドの発生・終了をイベントとして扱い，その時刻を管理することによってシミュレーションを行うことができる。クロックはイベントの開始・終了ごとに進めて行けば良いため，離散イベントシミュレーションと

なる [村田 95]。

離散イベントシミュレーションのためのツールとしては，GPSS, SIMSCRIPT, SLAN II などの汎用計算機言語がある [村田 95]。また，ネットワークシステムは解析的に解けないものも含めて，その多くが待ち行列モデルで記述可能であるため，待ち行列処理を容易に記述可能なネットワークシミュレーション言語や，汎用計算機言語用ライブラリなどが多く存在する。ネットワークシミュレーション言語の代表的なものとして，RESQ, QNAP 2 などがある。汎用計算機言語用ライブラリには C 言語用として，QSIM, SIMPACK, SMPL など，C++ 言語用としては SIMPACK++, SIM++, Awesime 2, CNCL, PARASOL, C++SIM などがある。

近年では GUI (Graphical User Interface) によるシミュレーションモデルの生成・編集・実行，評価結果の図示などが可能なネットワークシミュレーション支援環境が登場している。OPNET, BONes など，このような環境は商用のシステムとして提供され，非常に高価である。[LM94] また，これら商用シミュレータのパッケージには多くのプロトコルのためのシミュレーションモジュールが付属し，階層化されたプロトコル群のシミュレーションが可能である。しかしながら，それらのモジュールの仕様の多くは公開されていない。フリーなネットワークシミュレーション環境としては，VINT (Virtual InterNetwork Testbed) プロジェクトによる Ns [Proa] がある。Ns は高いモデル記述性，スケーラビリティ，シミュレーション状況の可視化など優れた性能をもつ。しかしながら，インターネット上でのトランスポート層プロトコルおよびネットワーク層プロトコルの評価を主な目的としており，本論文で扱う LAN 内での負荷集中時のシミュレーションを高精度で行う能力には欠けている。

上記の様な多階層のプロトコルを扱うシミュレーションモデル以外に，単一のプロトコル階層のシミュレーションを行うためのモデルも存在する。これらのモデルはシミュレーションのスケーラビリティに大きな特徴をもつ。Ethernet LAN のモデルでは，O'Reilly ら [OH94] によるノードのグループ化とスロット化に基づく手法，Marino[MdR93] らによるイベント発生時刻を前もって計算しておく手法などがある。

本論文の第 2 章では，教育用 LAN のような過渡的な負荷変動および高負荷なトラヒック特性を持つ Ethernet LAN の性能評価を指向したイベント駆動型のシミュレーション手法を提案する。従来のアプリケーションの挙動を含めたシミュレーションのためのモデルでは Ethernet などのデータリンク層以下のプロトコルを簡略化していたのに対し，提案モデルでは Ethernet のモデルを詳細に記述する一方で，Ethernet のフレーム衝突判定

における処理の高速化を行っており，高精度かつ高速なシミュレーションが可能である。

1.4 ネットワーク性能指標

ネットワークの性能を数値化する手法として，従来多くの性能指標が提案され，広く用いられている。スループットや帯域利用率はもっとも良く使われる性能指標の一例である。これらの指標はあくまでネットワークの性能そのものを表すが，それらが示すネットワーク性能上の問題点を発見するには熟練した専門家が複数の性能指標を総合的に判断する必要がある。しかしながら，低価格な LAN の普及に伴い，ネットワークの設計，管理は専門家のみの仕事ではなくなってきている。彼らにとって複数の性能指標からネットワークの性能上の問題点を推測することは困難であるため，より直観的に問題点を発見できるような性能指標および支援システムが必要であろう。

従来の性能指標に加えて視点の異なった別の性能指標を用いることにより，従来明らかにできなかったネットワークの振舞いや特性を発見することも期待できる。例えば，Ethernet の仕様上の問題とされる Capture Effect は，送信遅延の分布の調査 [Mol94] によってその現象の詳細が示されている。

本論文の第 3 章では，専門家以外が直観的にネットワークシステムの問題点を発見することを容易にする性能指標として，可視化に適した CSMA/CD LAN のための性能指標・ストール率を提案する。

1.5 教育用システムに適した LAN 構成

1.2 でも述べたように，現在の教育用システムの構成は分散型システムが主流である。教育用システムにおける分散型システムの最も簡単な構成は，浮貝ら [浮貝 92] によって提案されたシステムに見ることができる。これは，ファイルサーバと数十台のクライアントワークステーションを Ethernet に単純に接続した構成である。しかしながら，同時利用者数をより多くするために Ethernet にクライアントワークステーションを単純に増やすと，LAN への負荷集中が問題となる。そこで数百名以上の同時利用者数をサポートし，かつ LAN への負荷集中を避けるための LAN 構成手法がいくつか提案されている。たとえば，リムーバブルディスクに利用者の個人用データを保持し，ファイルサーバへのアクセスを極力避ける構成や，利用者用端末として X 端末を使用し，X 端末が接続された計

算サーバとファイルサーバは X 端末が接続された LAN とは物理的に異なる LAN を介して接続する方法 [中山 94] が提案されている。また，岡田ら [岡田 94][松本 94] はファイルサーバに複数のネットワークインターフェースを備えたワークステーションを用いることによって，クライアントワークステーションを接続する物理ネットワークを分離する手法を提案している。

さて，教育用システムの実環境でのネットワーク性能評価は中山ら [中山 94] や岩田ら [岩田 98] によって行われている。これらの研究は現実の教育用システムの実態を把握する上で重要であるが，基本となっている LAN は従来型の 伝送速度 10 Mbps の CSMA/CD 半二重 Ethernet である。今後の教育用システムの LAN 構成を検討するにはこれら従来式の Ethernet だけでなく，このほかの多様な LAN 構成方式も視野に入れる必要がある。

本論文の第 4 章では，教育用システムを指向した分散型ファイルシステム設計のための基礎検討として，最初に述べたシミュレーションモデルを用いて教育用システムに適した LAN 構成の検討を行う。また，検討の対象として従来の CSMA/CD による媒体共有型半二重の Ethernet とスイッチング型全二重の Ethernet に基づく 6 種類のネットワークトポロジを取り上げ，教育用システムにおける性能面，価格面の両面から教育用システムへの適正を検討し，ネットワーク設計の指針を与える。

本研究と同じように現実の LAN におけるトラヒックパターンを考慮してネットワーク性能評価を行う試みは石田ら [石田 95] によって行われている。石田らは研究環境における LAN のパケット長分布の測定を行い，この分布に基づいたパケット生成を用いたシミュレーションにより Ethernet LAN の性能評価を行っている。しかしながら，上位層プロトコルの挙動はパケット長のみに代表されており，そのほかの特性，たとえば端末間メッセージ依存性や輻輳制御，再送制御などは考慮されていない。

第 2 章で提案したモデルを用いれば，各層の通信プロトコル，アプリケーションおよびユーザの挙動を考慮したシミュレーションが可能である。教育用システムにおけるユーザの挙動をモデルの入力とするためには，現実のシステムにおけるユーザの挙動を知る必要がある。本研究では名古屋大学情報処理教育センターにおけるユーザ挙動の実測にもとづいてユーザ挙動モデルを設計し，シミュレーションを行った。なお，このモデルは第 2 章におけるシミュレーションモデルの妥当性の検証，第 3 章におけるストール率の評価にも用いられる。

第2章 過渡的な負荷を持つネットワークのモデル化

本章では，高負荷かつ過渡的な負荷変動を持つ LAN のプロトコルベースのモデル化およびシミュレーション手法を提案する。本手法では LAN 上のノードは詳細な通信プロトコルのモデルの集合として表される。各ノードのプロトコルモデルのイベント処理を並行に行うことにより，高信頼性，細粒度のシミュレーションが可能である。教育用システムにおける LAN は極めて高い時間的負荷集中が起きる可能性がある。またその性能評価のために大規模ネットワークを実際に構築するのは困難である。提案手法はこのような教育システムにおける LAN および過渡的な負荷変動が起きるそのほかの LAN の性能評価に有用である。

2.1 はじめに

近年，計算機ネットワークの性能評価の重要性が高まりつつある。今日，計算機ネットワークのトラヒックは FTP，電子メールなどの従来のプロトコルが発生するトラヒックのみならず，WWW などのハイパーテキスト，マルチメディアデータなど多岐にわたっている。また，キャンパス LAN，オフィス LAN は Local の名に反して極めて大規模になりつつあり，その性能を維持，効率を高めるために LAN の挙動・性能の予測は重要である。

LAN の性能を左右する要因は大きくわけて，ネットワークトポロジとそのネットワー

クの各ノードから発生するトラヒックの特性にあると考えられる。各ノードで発生するトラヒックの特性は，そのノードが通信に使用するプロトコル構成，それらのプロトコルを使用するアプリケーションの動作に依存する。さらにアプリケーションが対話的操作を提供する場合，そのアプリケーションが発生する通信要求の発生タイミングは，それを操作するユーザの挙動に依存する。

オフィス，研究施設などの一般的な環境の LAN では各端末を利用するユーザの動作は独立であるとみなして良いだろう。このため，互いに通信している 2 つのノードの間で依存性があることを除けば，多数の端末から構成される大規模 LAN 全体では各端末のパケット発生率は互いに独立であるとみなせる。ネットワーク負荷の時間的な変動は少ない。

一方，大学等の集合型情報処理教育システムでは，ユーザの操作は指導者の指示によって開始されることが多い。たとえば，教官の指示にしたがって同一のアプリケーションを起動したり，特定のファイルをコピーしたりすることが考えられる。このような環境では，先に見た一般的な環境に比べて，各端末でのパケット発生率には相関があり，ある時刻に複数の端末が同時にパケットを発生する確率が高くなる。ネットワーク負荷は過渡的に変動すると考えて良いだろう。

このようなユーザの集団的な動作の結果として過渡的な負荷集中が起こり得る LAN の性能評価を行うためには，ユーザの挙動を考慮する必要がある。ユーザの挙動を考えた場合，それによって動作するアプリケーション，およびその下位で動作する通信プロトコルの挙動までをも考慮する必要がある。

さて，従来多くの研究者らによって LAN の性能評価が行われてきたが，その手法は大きくわけて 3 つある。

1. 現実の LAN における計測
2. 解析的モデリング
3. シミュレーション

現実の LAN における計測は，現に存在する LAN の性能評価を行うには有効な手段である。しかしながら，大規模システムなどの評価のためにシステムを試作して計測を行うのは，大きなコストがかかり，現実的ではない。

解析的モデルは，過渡的な負荷変動が少ない場合，また負荷が極端に高くない場合には有効である。また計算量も比較的小さくて済む。しかしながら，解析的モデルを構築できるのは限られたトラヒックモデル，およびプロトコルに限られるため，過渡的な負荷変動が起きる場合や，動的な輻輳制御を行うようなプロトコルの評価は困難である。現実の LAN では (i) 複数階層のプロトコル上で通信が行われている，(ii) クライアント・サーバシステムや確認応答により信頼性保証を行うプロトコルの場合には送信者・受信者間でのメッセージ依存性がある，ことからアプリケーション層で発生する負荷の過渡的な変動があるような状況における LAN の性能評価を行うのは困難と言えよう。

Murata ら [MT88] は Token Ring と上位層のプロトコルからなるクライアントサーバモデルの階層化モデルを待ち行列による解析適手法により解く手法を提案している。Dorakopoulos ら [DM92] は Token Ring，Ethernet および Channel to Channel ネットワークにより構成されるネットワーク上の分散ファイルシステムの性能評価を待ち行列モデルとファイル移動を表す状態遷移モデルにより解析的に行っている。また，Ibe ら [ICT93] はペトリネットの変形である統計的 reward ネットを用いることにより CSMA/CD および Token Ring LAN 上のクライアントサーバシステムの性能評価を行っている。これら解析的な手法は一定負荷における平均的な LAN 性能を評価するものであるため，本論文で仮定する教育用 LAN のように高負荷かつ過渡的な負荷変動が起こり得る LAN の性能評価には適用困難である。また，解析的手法でモデル化可能な現象には限りがあり，TCP や NFS [Sun89] における動的な輻輳制御，再送処理をモデル化するのは困難である。

シミュレーションによる性能評価手法では，詳細なモデル化がいくらでも可能であり，解析的手法の適用が困難な場合に有用である。O'Reilly ら [OH94] は端末のグループ化とシミュレーション時間のスロット化に基づくスケーラビリティに優れた Ethernet のシミュレーション手法を提案している。また，Marino ら [MdR93] はイベント発生時刻をあらかじめ計算しておくことにより，Ethernet CSMA/CD のシミュレーションを高速化する手法を提案している。Ethernet よりも上位のプロトコルのシミュレーションが可能な環境として，VINT (Virtual InterNetwork Testbed) プロジェクトによる Ns [Proa] がある。Ns は高いモデル記述性，スケーラビリティ，シミュレーション状況の可視化など優れた性能をもつ。しかしながら，インターネット上でのトランスポート層プロトコルおよびネットワーク層プロトコルの評価を主な目的としており，本論文で扱う LAN 内での負荷集中時のシミュレーションを高精度で行う能力には欠けている。また，近年では OPNET，BONeS など，多くのプロトコル群のパッケージをあらかじめ備えた商用ネットワークシミュレーション支援環境が登場している [LM94]。しかしながら，それらのプロトコルモデルの仕様の多くは公開されていない。

本章では，高負荷かつ過渡変動がおこり得る LAN に適した効率的な LAN モデリング・シミュレーションのための手法を提案する [石原 95][IO97a]。提案モデルでは各階層のプロトコルは，それぞれプロトコルサブモデルにより構成され，各層のプロトコルはスロット化を行わないイベント駆動方式によってシミュレートされる。このため，提案手法は高い時間的粒度を持ち，高負荷，負荷変動のある LAN の評価においても高いシミュレーション精度を期待できる。また，端末間のプロトコルの相互作用による影響も評価可能である。

シミュレーションモデルを設計するにあたり，物理層，データリンク層プロトコルとして CSMA/CD によるアクセス制御を行う Ethernet 10BASE-T（伝送速度 10 Mbps，より対線接続，星型トポロジ）[ANS93]，トランスポート層およびネットワーク層プロトコルとして TCP/IP を仮定した。しかしながら，本手法におけるシミュレーションの枠組はこれら以外のプロトコル構成の評価にも適用可能である。

以下，まず 2 節で提案モデルで仮定するネットワークプロトコルについて概説し，3 節で多階層モデリングとシミュレーション手法について述べる。さらに，CSMA/CD プロトコルにおけるシミュレーションの高速化手法を提案する。3 節で本手法の応用として高負荷かつ高い負荷変動を持つ LAN におけるクライアントサーバシステムの評価を行った結果を示す。

2.2　ネットワークプロトコルの概説

本節では，以降のシミュレーションモデルでモデル化するネットワークプロトコルについて，基本的な事項を述べる。

一般に計算機間通信では，計算機間通信での一かたまりのデータ転送単位をパケットという言葉で表すが，以下の説明ではプロトコルごとにそれらを違った名前で呼ぶので，最初に断っておく。Ethernet でのデータ転送の一単位をフレーム，IP ではデータグラム，TCP ではセグメントと呼ぶ。これらは各プロトコルの仕様書等で用いられているものであり，また各プロトコルに範囲を限定した議論では一般に用いられるものであるので，本論文でもそれにならうことにする。

2.2.1　Ethernet

■ Ethernet の概要

イーサネット (Ethernet) は Xerox PARC で 1970 年代初めに発明された一般的なローカルエリアネットワークの技術に与えられた名前である。イーサネットは 1978 年，Xerox, Intel, DEC の 3 社によって標準化された。この標準を DIX 標準と呼ぶ。初期のイーサネットは通信媒体を直径 1/2 インチの同軸ケーブルとし，伝送速度は 10Mbps であった。こののち IEEE 802.3 [ANS93] による標準化が行われた。その後，高速化，さまざまな種類の媒体のサポートが行われ現在に至っている。

■ CSMA/CD アルゴリズム

Ethernet は分散アクセス制御をもったブロードキャストバスの技術である。すべての端末が一つの通信チャネルを共有しており，すべての端末のトランシーバは各転送を受信することが可能である。トランシーバが受信したフレームはすべてその端末のホストインタフェースに渡され。ホストインタフェースによって受け取るべきフレームの選別が行われる。これがブロードキャストバスと呼ばれる理由である。

Ethernet のアクセス制御機構は CSMA/CD (Carrier Sense Multiple Access with Collision Detection) と呼ばれている。各複数の端末はチャネルが他の端末によって占有されていないことを確認するため，送信を開始するためにキャリアの検出を行う (CSMA)。キャリアが検出されれば，キャリアが検出されなくなるまで送信を延期する。キャリアが検出されないことをうけてトランシーバが送信を開始しても，信号はすべての端末に同時に到達するわけではない。信号は同軸ケーブルの媒体上を光の真空中速度の約 80 % で伝播する。この伝播遅延のため，他の端末が送信を開始しているにも関らずそのキャリアを検出することができず，送信を開始することがありうる。この結果として複数端末が同時に一つの通信チャネルに信号を送ることになるが，これは信号の乱れをひき起こし，正しい送信とはならない。この信号の乱れを衝突と呼ぶ。衝突の発生はフレームの送信開始から，そのフレームの信号が伝送路全体に伝播されるまでの時間に起きる可能性がある。トランシーバは常に衝突を監視し (Collision Detection)，衝突が発生したならばフレームの再送を試みる。

衝突後の処理　再送は以下の手順で行われる。衝突を検出するとトランシーバは送信を中止する。フレームの送信中止はフレーム先頭の 64 ビットのプリアンブルとよばれるパターン信号の送信終了後に行われる。プリアンブルの送信が完了していれば送信は直ちに中止される。フレームの送信中止後，トランシーバは他の端末に衝突が起きたことを知らせるべくジャム信号を送信する。ジャム信号の送信終了後すぐにフレームを再送してしまうと，再び衝突がおきることになるので，衝突した複数の端末が繰り返し衝突を起こさないように互いの再送開始時刻をずらす必要がある。このために自乗バックオフ (binary exponential back-off) という考え方が採用されている。送信者は連続した衝突の回数 n に対して 0 から $2^{\min\{n,10\}}$ までの乱数にスロット時間（最小フレームの送信時間に等しい）をかけた時間だけ次の再送を延期する。同一フレームの送信を 16 回試みても成功しなかった場合（$n = 16$），ジャム信号の送信後そのフレームの送信は送信されず，破棄される。

スロット時間とフレームサイズ　前述のとおり，フレームの送信開始から信号が伝送路全体に伝播するまでの時間帯には衝突が起こり得る。最悪の条件は，最大端末間距離に位置する 2 つの端末 a, b があるときに，a からの信号が b に伝播する直前に衝突が発生する場合である。ここで a，b 間の伝播遅延を $\tau^{(\mathrm{MAX})}$ とすると，a が衝突を検出するのは，a が送信を開始してから $2\tau^{(\mathrm{MAX})}$ 後である。このときに a がフレームの送信を終了していると，衝突による信号の乱れを検出できない。従って，端末は最低でも $\tau^{(\mathrm{MAX})}$ は信号を送信し続けなければならない。この時間をスロット時間と呼ぶ。Ethernet の最小フレーム長はスロット時間に等しくなるように選択されている。伝送速度 10 Mbps, 100 Mbps の Ethernet のスロット時間は 512 ビット時間（= 512 ビットの送信に要する時間）となっている。なお，最大フレーム長は 1518 オクテット（= 12144 ビット）に定められている。

Ethernet では単一の端末がキャリア信号を途切らせることなく連続してフレームを送信することは許されていない。送信するフレームの間は最低でもフレーム間ギャップ時間（=96 ビット時間）だけ空ける必要がある。また，新たにフレームの送信を開始する場合には，最後にキャリア信号を受信してからフレーム間ギャップ時間経ってから送信を始める必要がある。

図 2.1 に Ethernet の CSMA/CD 方式の送信動作をまとめたものを示す。

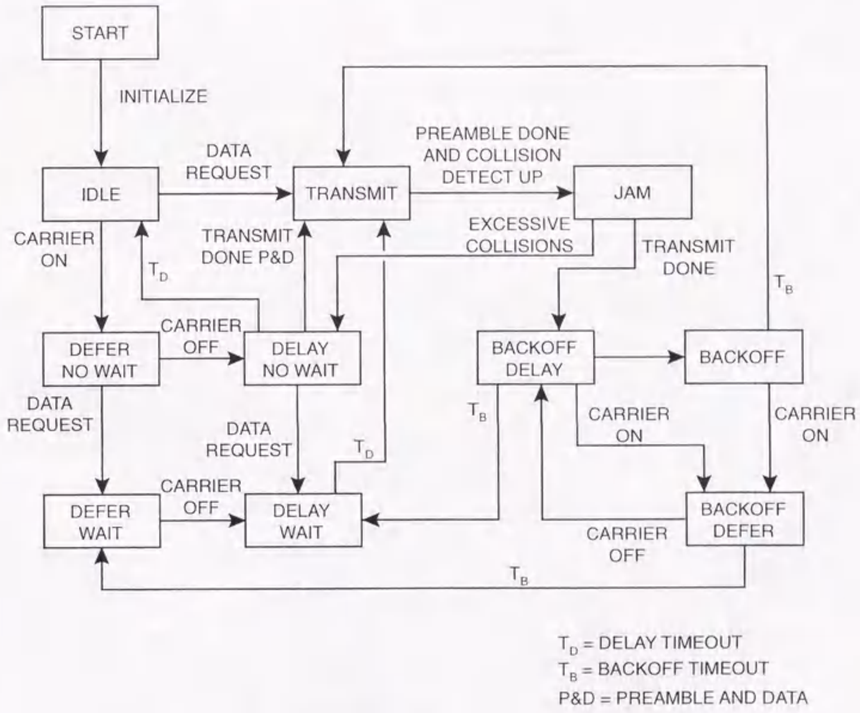


図 2.1: Ethernet CSMA/CD

Fig. 2.1 Ethernet CSMA/CD.

■ スター型接続 Ethernet

10BASE-5 と呼ばれる初期の Ethernet では伝送媒体として同軸ケーブルが用いられた。最大長は 500m であり，そこに端末をバス型接続する方式が採られた。媒体はリピータと呼ばれる接続機器で延長可能であった。現在広く普及しているのは，物理媒体として，UTP (Unshielded Twisted Pair) ケーブルを用いる伝送速度 10Mbps 10BASE-T と 伝送速度 100 Mbps 100BASE-TX [ANS95] である。10BASE-T と 100BASE-TX では，端末は複数のポートをもつリピータ（リピータハブ）にスター型接続される。しかしながら，理論的な接続形態はバス型接続であり，従来の Ethernet と同様に CSMA/CD によるアクセス制御が行われる。

■ スイッチング Ethernet

リピータのあるポートで受信したフレームはすべてのポートに複写（リピート）される。このため，複数のポートをもつリピータ（リピータハブ）では，同時に複数のポートからフレームが受信されると衝突が起こる。すなわち，リピータは Ethernet 上で透過的に働き，端末からはリピータで接続された Ethernet は同一の伝送媒体を共有しているように見える。このようにリピータで接続された伝送媒体の集合をコリジョンドメインと呼ぶ。CSMA/CD によるアクセス制御はコリジョンドメイン内で行われる。

これに対し，Ethernet スイッチは，リピータハブと同様に複数のポートを持つ Ethernet の接続装置として働くが，CSMA/CD によるアクセス制御は行わない。すなわち，Ethernet スイッチの入力側と出力側ではコリジョンドメインは異なる。2 つのポートで Ethernet コリジョンドメインを接続する機器はブリッジと呼ばれる。Ethernet スイッチはマルチポートのブリッジであると考えても良い。さて，Ethernet スイッチは，リピータハブとは異なり，受信したフレームを宛先のポートにのみフレームを転送する。宛先のポートが異なれば，平行して処理が可能であり，複数のポートからフレームを受信しても衝突が発生することはない。複数のポートを持つリピータがリピータハブと呼ばれるのに対し，Ethernet スイッチはスイッチングハブとも呼ばれる。

全二重通信　通信媒体として同軸ケーブルではなく，UTP ケーブルや光ファイバケーブルを用いた Ethernet では，端末とリピータハブまたは Ethernet スイッチを接続する伝送路は送信用，受信用で分かれている。リピータハブに接続した場合，送受信に用いる物

理的な伝送路が共有された同軸ケーブルによる Ethernet と同様に，CSMA/CD によるアクセス制御が行われるため，半二重通信しかできない。ところが，Ethernet スイッチに接続した場合，2 つの端末間を直接接続した場合には CSMA/CD によるアクセス制御の必要はない。このため，送信路と受信路を同時に使って全二重通信が可能である。Ethernet における全二重通信の仕様は IEEE802.3x [ANS97] で標準化されている。

本論文ではスイッチングハブのモデル化については扱わないが，第 4 章で本章で提案するシミュレーションモデルと小島 [小島 98] による全二重スイッチングハブのモデルを用いた性能評価を行っている。

■ ギガビット Ethernet

1998 年 6 月に，伝送速度 1 Gbps のギガビット Ethernet 1000BASE-X が IEEE802.3z として標準化された。1000BASE-X では伝送媒体として光ファイバケーブルあるいは平衡型シールドケーブルが用いられる。ギガビット Ethernet では従来の伝送速度 10 Mbps, 100 Mbps の Ethernet とフレーム互換性を保つ。ギガビット Ethernet では，スイッチングによる接続方式と CSMA/CD アクセス制御方式に基づく接続が可能である。

伝送速度を 10 倍に上げても信号の伝播速度は変わらない。従って，CSMA/CD によるアクセス制御を行う場合，フレーム形式を変更しなければ，スロット時間によって制限されるネットワークの規模は 1/10 に制限されることになる。しかしながら，100 Mbps Ethernet の 1/10 に制限されたネットワーク規模は実用的ではないので[1]，ギガビット Ethernet では従来の Ethernet の CSMA/CD 方式に 2 つの拡張，すなわちキャリア拡張とフレームバースティングが行われている（図 2.2）[MKk97]。

キャリア拡張　キャリア拡張ではフレーム形式を変更することなくスロット時間を 8 倍の 4096 ビット時間にする。CSMA/CD ではフレームの送信開始後，最低でもスロット時間の間は信号を送信し続けなければならないので，フレーム送信時間がスロット時間よりも短い場合には，フレームの信号として認識されない信号をスロット時間に達するまで送信し続ける。

フレームバースティング　キャリア拡張を行うと，スロット時間とそれより短いフレーム長との差の分だけ実効帯域が狭くなる。この帯域の損失を補うためのもう一つの拡張がフ

[1] 100 BASE-TX でのリピータを介した最大ノード間距離は 205 m である。

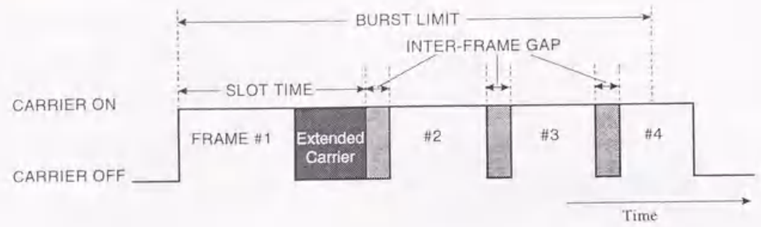


図 2.2: ギガビット Ethernet における CSMA/CD の拡張
Fig. 2.2 Extension of CSMA/CD for Gigabit Ethernet.

レームバースティングである。フレームバースティングでは，一つの端末がキャリア信号の切れ間なく連続してフレームを送信することを許す。すなわち，最初のフレームはキャリア拡張を用いて送信を行う。この最初のフレームの送信中に次に送るべきフレームが用意されていれば，最初のフレームの送信後に連続してフレームを送信する。最初のフレームの送信でスロット時間よりも長く信号を送信し続けているので，2 番目以降に送信するフレームの送信にはキャリア拡張の必要はない。この連続したフレーム送信をバーストと呼ぶ。バーストの長さはバーストリミットと呼ばれる時間で制限される。最初のフレームの送信開始からバーストリミットに達するまでは，連続したフレームの送信を開始することができる。すなわちバーストリミットはバースト全体の長さそのものを制限するパラメータではなく，バーストの先頭からバーストの末尾から 2 番目のフレームの末尾までの長さを制限するものである。1000 BASE-X ではバーストリミットは 65536 ビット時間とされている。なお，バースト中におけるフレームの間にはフレーム間ギャップの代わりとして，フレームとして認識されないフレームギャップと同じ長さの信号を送信する。

2.2.2　TCP/IP

■ TCP/IP プロトコルの概要

TCP/IP プロトコルはインターネットにおける標準のプロトコル体系である。TCP/IP という呼び方は，その体系の根幹をなす TCP と IP という二つのプロトコルの名前に由来している。

計算機間通信において問題をいくつかの部分に分割し，プロトコル設計を容易にするために，ISO (International Organization for Standardization) によって OSI (Open Systems

表 2.1: OSI 7 階層参照モデル

	名前	内容	プロトコルの例	
7	アプリケーション層	応用プログラム	Telnet, FTP, NNTP, HTTP, SMTP	NFS
6	プレゼンテーション層	データの表現方法		XDR
5	セッション層	抽象度度の高いインタフェース，認証		RPC
4	トランスポート層	通信主体（プロセス）間の通信，高信頼の転送	TCP, UDP	
3	ネットワーク層	計算機間での通信，経路制御	IP	
2	データリンク層	物理層での具体的なフレーム伝送形式	Ethernet，Token Ring	
1	物理層	物理伝送媒体	UTP ケーブル，同軸ケーブル	

Interconnection) 7 階層参照モデルが定義されている。各階層のプロトコルはそれぞれその層で要求される機能を担う。表 2.1 に OSI 7 階層参照モデルの各層で定められる内容と対応するプロトコルを示す。

TCP [Pos81b] はトランスポート層に対応し，信頼性のある転送を行う。UDP [Pos81c] も TCP/IP プロトコル体系では TCP と同一の階層のプロトコルとして動作するが，信頼性の確保は行わない。すなわち，UDP は信頼性の保証を行わないが，TCP と同様の上位インタフェースをもつ。このインタフェースはソケットと呼ばれる。UDP は NFS などの信頼性を必ずしも要求しない上位プロトコルによって使用される。IP はネットワーク層に対応し，複数の物理ネットワークを接続したインターネットワーク上での経路制御を行う。また，前述した Ethernet はデータリンク層および物理層に対応する。なお，TCP/IP プロトコル体系は ISO 7 階層モデルよりも後からできあがったものであり，その時点では ISO 7 階層モデルが定義する機能は古くなりつつあったので，それらは必ずしも一致しない。

■ IP

IP (Internet Protocol) [Pos81a] はネットワーク層プロトコルとして，複数の物理ネットワークを接続したインターネット上での経路制御を担う。IP ではそれより下の階層のプロトコルを限定しない。すなわち，物理ネットワークは Ethernet だけでなく，Token Ring や電話回線でもよい。ルータと呼ばれる複数のネットワークインタフェースを持つ

接続機器は，少なくとも IP 以下のプロトコルをサポートし，異なる物理ネットワークを相互に接続する。このように IP により複数の物理ネットワークを相互に接続して構成されたネットワークがインターネットである。

経路制御　IP でのデータ転送単位はデータグラムと呼ばれる。インターネットに接続された 2 つの計算機間には複数の経路が存在し得るが，IP はその中から適切な経路を使用して，送信元の計算機から宛先の計算機までデータグラムを転送する。この機能はルータ間のバケツリレー方式で実現される。すなわち IP におけるデータグラムの転送は，目的の経路上に含まれる同一の物理ネットワークで通信可能なルータにデータグラムを送ることを繰り返すことによって行われる。

信頼性の非保証　IP はデータグラムの転送において信頼性を保証しない。すなわち，IP 以下のプロトコルでデータグラムが破棄されたり，IP 内部における問題でデータグラムが送信不能になったとしても，その補償を行わない。

フラグメント化　IP では異なる物理ネットワーク間でデータグラムを転送する必要があるので，ある物理ネットワークから受け取ったデータグラムや上位層から要求された送信データの大きさよりも，転送先の物理ネットワークでサポートする最大フレームサイズが小さい場合には，転送するデータグラムを分割する。この操作をフラグメント化と呼び，分割されたデータグラムのそれぞれをフラグメントと呼ぶ。一度フラグメント化が行われたデータグラムは，最終的な宛先の計算機の IP でフラグメントから再構成される。

■ TCP

TCP [Pos81b] はトランスポート層のプロトコルとして，信頼性を保証しないプロトコル上で信頼性を保証した全二重通信路上でのストリーム通信を提供する。すなわち，データは一つのデータの固まりとして送られるのではなく，連続したデータ列として送られる。

信頼性を保証し，かつ効率の良い通信を行うため，TCP ではスライディングウインドウ機構を使用している。この機構は，データの順序番号，確認応答メッセージ，送信ウインドウ，受信ウインドウを使用することにより実現される。

図 2.3 に TCP のスライディングウインドウ機構の概略を示す。送信側は送信データをセグメントに分割し，セグメントにデータストリームの先頭の順序番号を付加して送信す

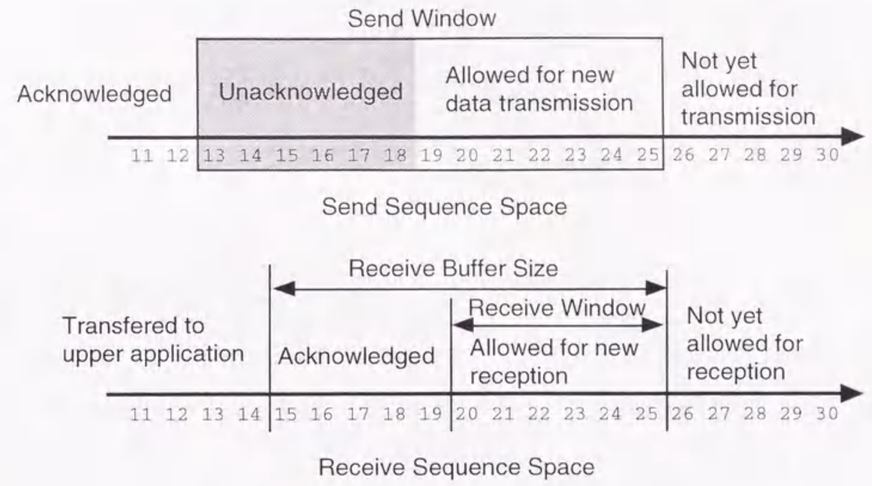


図 2.3: TCP のスライディングウインドウ機構

Fig. 2.3. Sliding window mechanism of TCP.

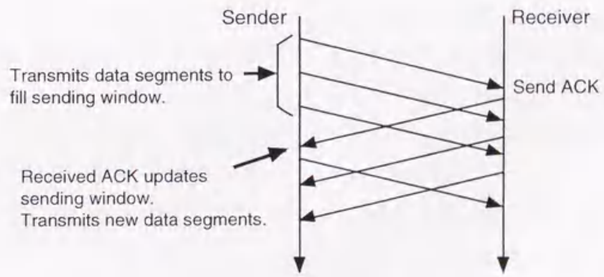


図 2.4: TCP セグメントの送信

Fig 2.4 Transmission of TCP segments.

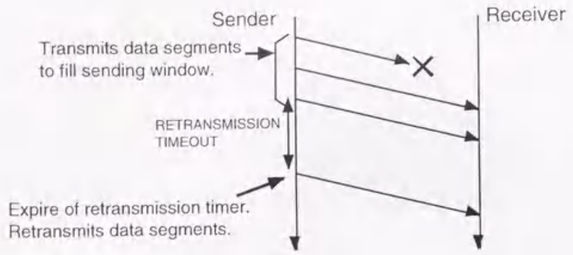


図 2.5: TCP セグメントの再送信

Fig. 2.5. Retransmission of TCP segments.

る。このとき，受信側から知らされている受信側 TCP の受信バッファの空き容量（送信ウインドウ）を満たすまでデータストリームを連続して送信できる。送信データは受信側からの確認応答（ACK）が得られるまで保持される（図 2.4）。

受信側はデータを含むセグメントを受けとると，ACK セグメントを返送する。ACK セグメントには，セグメントの順序番号と正しく受信できた受信データの順序番号，および受信バッファの空き（受信ウインドウサイズ）の情報が含まれる。この ACK セグメントは受信側から送信するデータセグメントと同一のものでもよい。また，ACK は受信したセグメント一つに対して必ずしも一つ送る必要はなく，正しく受信できたデータに対してまとめて確認応答すれば良い。

受信側から ACK を受け取ると，送信側の TCP は確認応答のあった送信データを送信バッファから解放する。受信したセグメントには受信側 TCP の受信ウインドウサイズが含まれているから，送信側の送信ウインドウが更新される。この結果，送信側の送信ウインドウには空きが生じ，再び送信ウインドウを満たすまで新しいセグメントを送信することができる。

もし送信時に設定した再送待ち時間以内に ACK が得られない場合には，確認応答の得られていないデータを再送信する。再送信時には確認応答を受けていないすべてのデータに対して ACK を受けとるまで，セグメントを一つずつ送信する。(図 2.5)

表 2.2: モデルで仮定したプロトコル

層	プロトコル名
トランスポート層	TCP および UDP
ネットワーク層	IP
データリンク層	IEEE 802.3 CSMA/CD
物理層	UTP ケーブル

2.3　シミュレーションモデル

2.3.1　多階層モデリングの手法

本手法では，ネットワークプロトコルの振舞いの詳細をシミュレーションモデルで記述する。この手法によるシミュレーションの実現の一つとして，本研究では表 2.2 で示すプロトコル構成のシミュレーションを行うことにした。図 2.6 にクライアントサーバ・ネットワークシステムを想定したネットワークモデルの構成を示す。各端末のモデルは，Ethernet サブモデル，TCP/UDP サブモデル，IP サブモデル，アプリケーションサブモデルおよびユーザ挙動モデルから構成される。このモデルにおいて，TCP/UDP サブモデルはソケットに対応している。

シミュレーションはイベント駆動方式で行う。各サブモデルの動作はイベントによって開始される。このイベントはそのサブモデル自身およびそのほかのサブモデルによってスケジュールされる。イベントにはスケジュールされたイベントと，即時イベントの二種類がある。

各サブモデルにおける処理が終了すると，サブモデルが次に動作するためのイベントがスケジュールされ，シミュレーション上の時間が更新される。即時イベントが発生した場合，そのイベントがサブモデル内部で処理されるべきものであれば，サブモデルの内部で連続して処理が行われる。ほかのサブモデルによって処理されるべきものである場合，内部での処理は終了し，同一時刻に処理されるべきイベントとして他のサブモデルに通知する。この処理に当てはまるイベントは，パケットの受信と送信である。

イベントのスケジュール方法は Ethernet サブモデルとその他のサブモデルとは若干異なる。Ethernet サブモデル以外のモデルでは，イベントのスケジュールは単にそのサブモデル内部でのみ行われる。外部のサブモデルとの通信は即時イベントのみで行われる。

Ethernet サブモデルにおけるイベントのスケジュールは，各ノード内のみではなく，Ethernet のコリジョンドメイン全体を考慮して行われる。すなわち，あるノード内部に

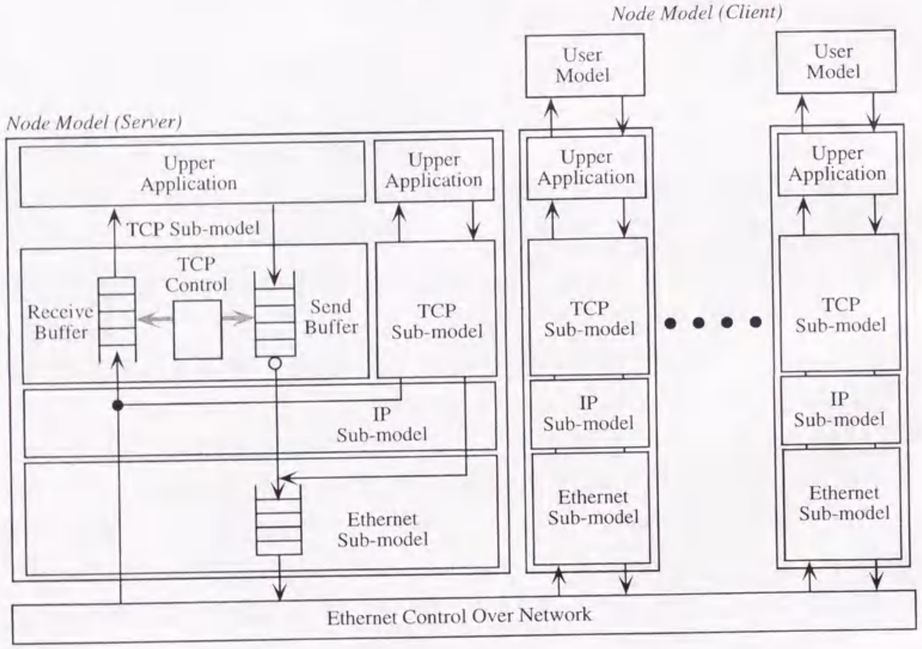


図 2.6: シミュレーションモデルの全体構成

Fig. 2.6 Overview of the simulation model.

おける状態の変化はそのノード自身のみではなく，他のノードにも影響を及ぼし，他のノードのイベントスケジュールを変更することがある。なぜなら，Ethernet における各ノードの動作はコリジョンドメイン全体の状態に依存しているためである。これについての詳細は 2.3.2 で述べる。

2.3.2　Ethernet サブモデル

■ 仮定するネットワークトポロジとその表現

以下の議論では Ethernet サブモデルで扱うネットワークは単一のコリジョンドメインのみから構成されるとする。すなわち，Ethernet の物理セグメントはリピータによってのみ接続され，ブリッジ，ルータを含まない。ネットワークトポロジはネットワークに接続された端末間の信号伝播遅延のみで表現されるとする。リピータはモデル上では単純に伝播遅延としてのみ表現される。

■ 状態遷移

Ethernet サブモデルは状態遷移モデルとして表される。IEEE802.3 における定義では CSMA/CD アルゴリズムは 11 状態からなる状態遷移モデルとして定義してある。本手法では効率化のため状態数を 11 から 6 個に削減した。この削減によってモデルの正確さは損なわれてはいない。図 2.7 に状態遷移図，表 2.3 に状態遷移表を示す。

端末 $i(=1,2,\ldots,N)$ における Ethernet サブモデルは送信状態 $S^{(i)} \in \{\mathrm{SLP}, \mathrm{CS}, \mathrm{CR}, \mathrm{TR}, \mathrm{CD}, \mathrm{WT}\}$，最後にキャリアの送信を開始した時刻 $t_s^{(i)}$，最後にキャリアの送信を終了した時刻とフレーム間ギャップの和 $t_e^{(i)}$ 現在処理中のフレーム送信処理の間に起きた衝突の回数 $n_c^{(i)}$ を持つ。ここで，$S^{(i)}$ の 6 個の状態はこのモデルで定義された状態であり，IEEE 802.3 で定義されたものとは異なる。端末 i の状態 $M^{(i)}$ は $M^{(i)} = \{S^{(i)}, t_s^{(i)}, t_e^{(i)}, n_c^{(i)}\}$ で表される。また，送信状態 $S^{(i)} = \mathrm{CS}$ となる端末数 N_c，端末 i, j 間の遅延 $\tau^{(i,j)}$ を定義する。

端末 i における次のイベント発生時刻 $t_n^{(i)}$ は $M^{(i)}$ および i 以外の端末 j の状態 $M^{(j)}$ によって決定される。イベントが発生すると端末 i の転送状態 $S^{(i)}$ は変化し，$t_n^{(i)}$ が新しく計算される。

転送状態 $S^{(i)}$ の各状態はそれぞれ以下のような意味を持つ。

表 2.3: Ethernet サブモデルの状態遷移

現在の状態	イベント	動作	次の状態
SLP	データが送信バッファに存在 伝送路空き	$n_c^{(i)} \leftarrow 0$	CR
	データが送信バッファに存在 伝送路ビジィ	$n_c^{(i)} \leftarrow 0$	CS
CS	伝送路空き	$t_s^{(i)} \leftarrow t$ $N_c \leftarrow N_c + 1$	CR
CR	衝突危険期間終了 衝突なし	$N_c \leftarrow N_c - 1$	TR
	衝突危険期間終了 衝突あり	$N_c \leftarrow N_c - 1$ $n_c^{(i)} \leftarrow n_c^{(i)} + 1$	CD
TR	フレーム送信終了	$t_e^{(i)} \leftarrow t$	SLP
CD	JAM 信号送信完了 $n_c^{(i)} = 16$	$t_e^{(i)} \leftarrow t$	SLP
	JAM 信号送信完了 $n_c^{(i)} < 16$	$t_e^{(i)} \leftarrow t$ $T_b^{(i)}$ を計算	WT
WT	送信延期期間終了。 伝送路空き。	—	CR
	送信延期期間終了。 伝送路ビジィ	—	CS

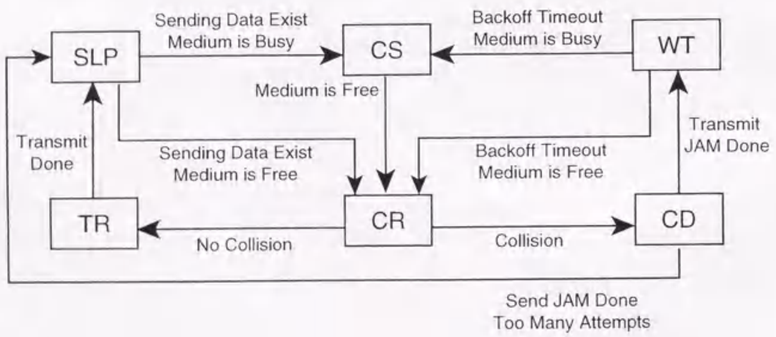


図 2.7: Ethernet サブモデルの状態遷移図

Fig. 2.7 State transition diagram of the Ethernet sub-model.

SLP (SLeeP)：端末が何も送信していない状態。送信要求が Ethernet サブモデルに到着すると，次の状態に移る。伝送路がビジィの場合には CS へ，そうでなければ CR に遷移する。

CS (Carrier Sense)：伝送路がビジィまたはフレーム間ギャップのために，端末がフレームを送信できない状態。伝送路からキャリア信号が消え，さらにそれからフレーム間ギャップが経過すると次の状態 CR に遷移する。

CR (CRitical)：状態 CR と TR はどちらもその端末がフレームを送信中であることを意味する。状態 CR と TR の違いは，TR ではフレームが衝突する可能性がないのに対し，CR ではそのノードが送信中のフレームが衝突する可能性があることである。IEEE 802.3 の CSMA/CD の仕様では，端末がフレームを送信中の状態は単一の状態として定義してあるが，このモデルではシミュレーションにおける衝突検出処理の効率化のために状態を 2 つに分けている。

$S^{(i)} =$ CR のとき，端末 i はフレームを送信してはいるものの，それが送信するキャリア信号はコリジョンドメイン内の全ての端末にまでは到達していない。すなわち，その時刻に i 以外の端末 j が送信を開始しようとした場合，j が i の送信するキャリア信号を検出しない可能性がある。キャリア信号を検出しなければ，j は送信を開始することになり，いずれ i, j が送信する信号が互いに衝突することになる。この様な時間帯を衝突危険期間と呼ぶことにする。この衝突危険期間は各端末で独立なものではなく，コリジョンドメイン全体で共有する期間である。言い換えれば，$S^{(i)} =$ CR であるということは，端末 i が衝突危険期間内にフレームの送信処理を行っていることを意味している。衝突危険期間の詳しい定義については後述する。

状態 CR は衝突危険期間の終了とともに終了する。このとき，後で述べる衝突判定を行い，衝突が発生していれば CD へ，そうでなければ TR に遷移する。

TR (TRansmit)：端末がフレームを送信しており，衝突が発生しないことが明らかである状態。フレームの送信完了により状態は SLP に遷移する。またこのとき，送信バッファの先頭のエントリを取り出し，宛先の端末の IP サブモデルへの到着イベントをスケジュールする。

CD (Collision Detect)：端末が衝突検出後に JAM 信号を送信している状態。JAM 信号の送信が終了すると，現在のフレーム処理中に起きた衝突の数 $n_c^{(i)}$ に従って次の

状態に移る。$n_c^{(i)} < 16$ すなわち IEEE 802.3 で規定された最大衝突数よりも少ない場合には，バックオフ遅延が計算され WT に遷移する。$n_c^{(i)} = 16$ の場合にはフレームの送信は失敗になるので，SLP に遷移する。またこのとき，送信バッファから送信データの情報を取り除く。

WT (WaiT)： 端末がフレーム再送信の前にバックオフ遅延分だけ待っている状態。待ち時間が終了すると，伝送路状態を調べ，伝送路がビジィならば CS へ，そうでなければ CR へ遷移する。

■ 伝送路状態の判定

前述したように，伝送路状態の判定は SLP または WT から他の状態に遷移する時に行われる。以下の説明では，端末 i が他の端末 j が送信するキャリア信号を検出するかどうかを判定するとする。

$S^{(j)} \in \{\mathrm{TR}, \mathrm{CD}\}$ のとき，端末 j はキャリア信号を送信しており，それはコリジョンドメイン全体で受信されている。このため少なくとも 1 台 $S^{(j)} \in \{\mathrm{TR}, \mathrm{CD}\}$ となる端末 j が存在するとき，端末 i での伝送路状態はビジィである。

$S^{(j)} \in \{\mathrm{SLP}, \mathrm{CS}, \mathrm{CR}, \mathrm{WT}\}$ の場合，端末 i での伝送路の状態は $t_s^{(j)}$，$t_e^{(j)}$ および $\tau^{(i,j)}$ に依存する。$S^{(j)} = \mathrm{CR}$ のとき，端末 j は $t_s^{(j)}$ に送信を開始しているが，端末 i,j 間の伝播遅延 $\tau^{(i,j)}$ のため，$t_s^{(j)} + \tau^{(i,j)}$ まではその信号は端末 i まで到達しない。一方，$S^{(j)} \in \{\mathrm{SLP}, \mathrm{CS}, \mathrm{WT}\}$ のとき，端末 j は何も信号を送信していないが，伝播遅延による遅れのため $t_e^{(j)}$ までは端末 i は端末 j からの信号を受信することになる。

以上をまとめると，時刻 t における端末 i での伝送路状態は以下のように判定できる。

$$
\text{端末 } i \text{ で伝送路状態がビジィ} = \begin{cases} \left[S^{(j)} \in \{\mathrm{TR}, \mathrm{CD}\}\right] \text{ となる端末 } j(\neq i) \text{ が存在する。} \\ \quad \text{または} \\ \left[S^{(j)} = \mathrm{CR} \ \wedge \ t_s^{(j)} + \tau^{(i,j)} \leq t\right] \text{ となる端末 } j(\neq i) \text{ が存在する。} \\ \quad \text{または} \\ \left[S^{(j)} \in \{\mathrm{SLP}, \mathrm{CS}, \mathrm{WT}\} \ \wedge \ t \leq t_e^{(i)} + \tau^{(i,j)}\right] \text{ となる端末 } j(\neq i) \text{ が存在する。} \end{cases} \tag{2.1}
$$

■ 衝突発生の判定

CSMA/CD アルゴリズムでの衝突は，ある端末 i が送信中に，他の端末 j が i が送信するキャリア信号を検出しないうちに送信を開始することによって発生する。端末が送信するキャリア信号がコリジョンドメイン内の全ての端末で検出されれば，キャリア信号を検出した端末は送信開始を延期するので衝突は起こらない。ここでコリジョンドメイン内のすべての端末が送信していないとき（すなわち，すべての端末 j に関して $S^{(j)} \in \{\mathrm{SLP}, \mathrm{CS}, \mathrm{WT}\}$）に最も早く送信を開始した端末を i とする。端末 i が送信を開始した時刻 $t_s^{(i)}$ から，$t = t_s^{(i)}$ 以後に送信を開始した端末（i を含む）からのキャリア信号がコリジョンドメイン内の全ての端末で受信される時刻までの期間に，複数の端末が送信を開始すると衝突が起こる。この期間を衝突危険期間と呼ぶことにする。

ある時刻 t で $S^{(i)} = \mathrm{CR}$ となる端末 i が存在するとき，衝突期間は以下の式で表される。

$$
\text{衝突危険期間の開始時刻} \qquad T_s = \min_{i|S^{(i)}=\mathrm{CR}} t_s^{(i)} \tag{2.2}
$$

$$
\text{衝突危険期間の終了時刻} \qquad T_e = \min_{i|S^{(i)}=\mathrm{CR}} \left\{ \max_{j \neq i} (\tau^{(i,j)}) + t_s^{(i)} \right\} \tag{2.3}
$$

衝突の判定は，T_e で $S^{(i)} = \mathrm{CR}$ となる端末の数 N_c を調べることによって行う。

$$
\begin{cases} \text{If} \quad N_c > 1 \quad \text{then} & \text{衝突発生} \\ \qquad\qquad \text{otherwise} & \text{衝突なし} \end{cases} \tag{2.4}
$$

以上の判定はシミュレーション上で衝突が発生したか否かを判定するものである。また，衝突を起こした端末 i が衝突の発生を検知する時刻 t_c は，端末 i が送信中に最も早く到着する他の端末 j からの信号到着時刻として計算される。

$$
t_c^{(i)} := \min_{j \neq i, S^{(j)}=\mathrm{CR}} \{ t_s^{(j)} + \tau^{(i,j)} \} \tag{2.5}
$$

■ イベントのスケジュール

コリジョンドメイン内のいずれかの端末でイベントが発生すると，その端末における次の新しいイベントがスケジュールされる。また，場合によっては，他の端末でのイベント

もスケジュールされる。以下では，端末 i が送信状態 $S^{(i)}$ にあるときに，新しいイベントがスケジュールされる時刻を $t_n^{(i)}$ として，イベントのスケジュール方法を説明する。

SLP： 次に発生するべきイベントは新しい送信要求の発生である。上位層プロトコルの動作の結果，新しい送信要求が発生すると，それは Ethernet サブモデルの FIFO 待ち行列である送信バッファに入れられる。このとき送信バッファが空の場合は，送信要求はイベントは即時イベントとして処理される。また，SLP 以外の状態から SLP に遷移した場合に送信バッファが空ではない場合にも送信要求イベントが即時イベントとして処理される。

また，端末 i が他の状態から SLP に遷移したときは $t_n^{(i)}$ は未定であるので，

$$t_n^{(i)} := \infty \tag{2.6}$$

としてスケジュールされる。

CS： 次に発生するべきイベントは，伝送路状態が変化して送信可能になることである。

$S^{(i)} = \mathrm{CS}$ のとき，端末 i が他の送信中の端末 j（$S^{(j)} \in \{\mathrm{CR}, \mathrm{TR}, \mathrm{CD}\}$）のキャリア信号を受信しているとすると，端末 j の状態が変化して，j が送信を終了しない限り，端末 i での伝送路状態は変化しない。

一方，端末 i が送信をしていな端末 j（$S^{(j)} \in \{\mathrm{SLP}, \mathrm{CS}, \mathrm{WT}\}$）からのキャリア信号を受信している場合，端末 i は $t = t_e^{(j)} + \tau^{(i,j)}$ に j からのキャリア信号の検出の終了およびフレームギャップ時間の経過が完了する。

したがって，$S^{(i)} = \mathrm{CS}$ となる端末 i に対するイベントのスケジュールは，$S^{(j)} \in \{\mathrm{TR}, \mathrm{CD}\}$ となる端末 j が 1 台しかないときに，$S^{(j)}$ が SLP, CS または WT に遷移するときに行う。イベント発生時刻 $t_n^{(i)}$ は以下のようにして計算される。

$$t_n^{(i)} := \max_{j \neq i} \{t_e^{(j)} + \tau^{(i,j)}\} \tag{2.7}$$

また，端末 i が CS 以外の送信状態から CS に遷移したときには以下の式に従ってイベントをスケジュールする。

$$t_n^{(i)} := \begin{cases} \infty & \left(\exists j \quad \text{s.t.} \quad \left[S^{(j)} \in \{\mathrm{TR}, \mathrm{CD}\}\right] \vee \left[S^{(j)} = \mathrm{CR} \wedge t < t_s^{(j)} + \tau^{(i,j)}\right]\right) \\ \max\limits_{j \neq i} \{t_e^{(j)} + \tau^{(i,j)}\} & (\text{otherwise}) \end{cases} \tag{2.8}$$

CR： 次に起こるべきイベントは，衝突危険期間の終了である。したがって，このイベントの発生する時刻は T_e である。T_e は状態 CR にある端末が遷移する度に，式 2.3 にしたがって更新される。このとき，$t_n^{(i)}$ も新しい T_e に更新される。

$$t_n^{(i)} := T_e \tag{2.9}$$

TR： 次に起こるべきイベントは，フレームの送信が終了し，フレームギャップ時間が経過することである。ここでフレーム送信時間を $\tau_f^{(i)}$，フレームギャップ時間を τ_g とすると，$t_n^{(i)}$ は以下のように計算される。

$$t_n^{(i)} := t_s^{(i)} + \tau_f^{(i)} + \tau_g \tag{2.10}$$

このイベントは，CR から TR への状態遷移の時点でスケジュールされるだけである。このイベントがスケジュールされた後は，他の端末の動作の影響は受けない。

CD： 次に起こるべきイベントは，JAM 信号の送信終了後のフレームギャップ時間の経過である。IEEE 802.3 では，JAM 信号の送信を開始する前に少なくとも 64 bit のプリアンブルを送信していなければならないと規定している。したがって，プリアンブルの送信に要する時間を τ_p とすると JAM 信号の送信を開始するのは，$\max\{t_s^{(i)} + \tau_p, t_c^{(i)}\}$ である。ここで τ_j をジャム信号の送信に要する時間とすると，$t_n^{(i)}$ は以下のように計算される。

$$t_n^{(i)} := \max\{t_s^{(i)} + \tau_p, t_c^{(i)}\} + \tau_j \tag{2.11}$$

t_c は式 2.5 で計算される，端末 i が衝突を検出した時刻である。

WT： 次に起こるべきイベントは，バックオフによる待ち時間の終了である。バックオフ時間による待ち時間 $\tau_b^{(i)}$ は，フレーム送信中の衝突検出回数 $n_c^{(i)}$ にしたがって計算される。この計算は CR から CD への遷移または CD から WT への遷移の時に計算すればよい。t_n は以下のように計算される。

$$t_n^{(i)} := t_c^{(i)} + \tau_j + \tau_b^{(i)} \tag{2.12}$$

■ **Ethernet サブモデルの制限**

ここでは，Ethernet サブモデルの構築における仮定の妥当性に付いて検討する。以下の説明では，コリジョンドメイン間の最大端末間遅延 $\max_{i\neq j}\tau^{(i,j)}$ を $\tau^{(\mathrm{MAX})}$ で表し，最小端末間遅延 $\min_{i\neq j}\tau^{(i,j)}$ を $\tau^{(\mathrm{MIN})}$ で表すことにする。

まず，送信状態 CR の定義について考察する。端末 i が衝突し，このとき $t_c^{(i)} < T_e$ となった場合，端末 i は $S^{(i)} = \mathrm{CR}$ である間にプリアンブルの一部および JAM 信号の送信を開始することになる。しかしながら，

$$\max\{(t_s^{(i)} + \tau_p), t_c^{(i)}\} + \tau_j + \tau_g \leq T_e \tag{2.13}$$

となる場合，端末 i は $S^{(i)} = \mathrm{CR}$ であるうちに JAM 信号の送信を完了しフレームギャップ時間が経過してしまう。この状況は送信状態 CR が前提としている端末がなんらかの信号を送信している，もしくはフレームギャップにより他の端末の送信を妨げている状況とは矛盾する。したがって，本モデルでシミュレーションするネットワークの規模は次の式の条件を満たすものでなければならない。

$$\max\{(t_s^{(i)} + \tau_p), t_c^{(i)}\} + \tau_j + \tau_g > T_e \tag{2.14}$$

ここで式 2.13 の最悪の場合として以下のような状況を考える（図 2.8）。

- 3 台の端末 a，b および c があり，$\tau^{(\mathrm{MIN})} = \tau^{(a,b)}$，$\tau^{(\mathrm{MAX})} = \tau^{(a,c)} = \tau^{(b,c)}$ である。
- a と b が $t_s^{(a)} = t_s^{(b)}$ に同時に送信を開始した結果，それらの送信するフレームが互いに衝突し，JAM 信号を送信する。

このとき，式 2.13 の左辺の最小値は，式 2.3 から次の式で与えられる。

$$\max\{(t_s^{(a)} + \tau^{(a,b)}), (t_s^{(a)} + \tau_p)\} + \tau_j + \tau_g \tag{2.15}$$

また，式 2.5 より，式 2.13 の右辺の最大値は次の式で与えられる。

$$\tau^{(a,c)} + t_s^{(a)} \tag{2.16}$$

したがって，式 2.13 は次のように書き換えられる。

$$\max\{\tau_p, \tau^{(a,b)}\} + \tau_j + \tau_g < \tau^{(a,c)} \tag{2.17}$$

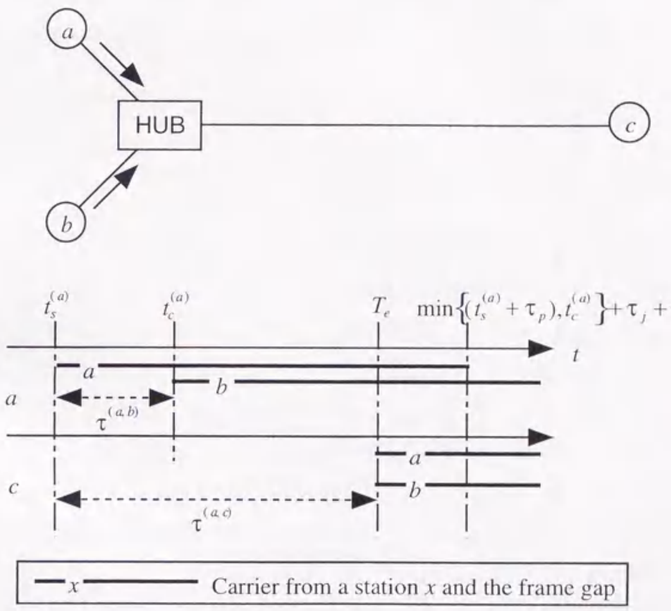


図 2.8: Ethernet サブモデルの制限

Fig. 2.8 Restriction of the Ethernet sub-model.

上式の $\tau^{(a,b)}$ と $\tau^{(a,c)}$ をそれぞれ $\tau^{(\mathrm{MIN})}$, $\tau^{(\mathrm{MAX})}$ に書き換え，整理すると，Ethernet サブモデルの制限は次の式のようにまとめられる。

$$\tau^{(\mathrm{MAX})} - \max\{\tau^{(\mathrm{MIN})}, \tau_p\} < \tau_j + \tau_g \tag{2.18}$$

次に送信状態 CD の定義について考察する。本モデルでは，$S^{(i)} = \mathrm{CD}$ となる端末が少なくとも 1 台存在する時にすべての端末がキャリア信号を受信していることを仮定している。すなわち，ある端末が，最後に JAM 信号を送出した端末からの JAM 信号を受け取る以前に，他の端末からの JAM 信号の受信を完了し，その端末の位置で伝送路が空きになることを許していない。

以下に起こり得る最悪の状況を示す。

- 3 台の端末 a, b および c があり，$\tau^{(\mathrm{MIN})} = \tau^{(a,b)}$, $\tau^{(\mathrm{MAX})} = \tau^{(a,c)} = \tau^{(b,c)}$ である。
- a と b が $t_s^{(a)} = t_s^{(b)}$ に同時に送信を開始した結果，それらの送信するフレームが互いに衝突し，JAM 信号を送信する。
- a または b からのキャリア信号が c に達する寸前 $t_s^{(a)} + \tau^{(a,c)} - \delta$ にフレームの送信を開始する。

a と b はコリジョンドメインの中で最も早く $t_s^{(a)} + \tau^{(a,b)} + \max\{\tau_p, \tau^{(a,b)}\}$ に a または b が送信する JAM 信号を受信する。a は $t_s^{(a)} + \tau^{(a,b)} + \max\{\tau_p, \tau^{(a,b)}\} + \tau_j + \tau_g$ に b からの JAM 信号の受信を終了し，フレームギャップ時間が経過した状態になる。バックオフ遅延が 0 であれば，この時点で a, b は次の送信が可能な状態にある。さて，c から送信された信号は送信直後に a および b からの信号と衝突することになるので，c は時刻 $t_s^{(a)} + \tau^{(a,c)}$ に衝突を検出し，プリアンブルの残りと JAM 信号を送信する。c が送信する信号が a に到着するのは $t_s^{(a)} + 2\tau^{(a,c)}$ である。この時点で，a および b が送信可能な状態であってはならないから，以下の条件が満たされていなければならない。

$$t_s^{(a)} + \tau^{(a,b)} + \max\{\tau_p, \tau^{(a,b)}\} + \tau_j + \tau_g > t_s^{(a)} + 2\tau^{(a,c)} \tag{2.19}$$

この式で，$\tau^{(a,b)}$, $\tau^{(a,c)}$ をそれぞれ $\tau^{(\mathrm{MIN})}$, $\tau^{(\mathrm{MAX})}$ と書き直し，整理すると，本モデルで要求するネットワーク規模の第 2 の条件として以下の式が得られる。

$$2\tau^{(\mathrm{MAX})} - \left[\tau^{(\mathrm{MIN})} + \max\left\{\tau_p, \tau^{(\mathrm{MIN})}\right\}\right] > \tau_j + \tau_g \tag{2.20}$$

■ ギガビット Ethernet のための拡張

本モデルをギガビット Ethernet のキャリア拡張，フレームバースティングに対応させるのは容易である。これには，式 2.10 で用いられるフレーム送信時間として，i) バーストの先頭のフレーム送信時にはフレーム長とスロット時間の小さい方の値を用い，ii) バーストの先頭以外のフレーム送信時にはフレーム長のみを用いれば良い。また，バーストを続けるか否かの判定には，送信バッファ内のデータ有無と，t_s と 現在時刻 t から得られるバースト長 $t_s - t$ とバーストリミットとの比較により行える。バーストが続く場合には 状態 TR からの遷移先は TR となる。

2.3.3 IP サブモデル

IP の機能はルーティング，IP データグラムのフラグメント化と再構成である。本モデルではシミュレートされるネットワークは単一の論理ネットワークから構成されると仮定している。従って IP によるルーティングに関する制御は無視できる。また，もう一つの仮定として TCP が生成するセグメントが IP によるフラグメント化を必要としない程度に小さいとする。このため，IP がメッセージの送信時間に及ぼす影響は IP データグラムのフォーマットのみとなる。そこで IP のモデルでは単純に，送信する TCP セグメントにたいして IP ヘッダ長分のデータの追加し，Ethernet サブモデルから受信するメッセージから IP ヘッダ長分のデータを取り除く機能のみをもつこととした。

2.3.4 TCP サブモデル

TCP の性能は TCP の各機能 — 送信，再送信，再送タイムアウトの制御，ウインドウサイズ — の実装方式に依存している。TCP の仕様 [Pos81b] では TCP のメッセージフォーマットや外から見える動作の規定は行われているが，その性能を左右する実装方式については定められていない。このため以下のような事項に関してな実装方針が異なる様々な TCP の実装が存在する。

- 再送タイムアウトの決定手法（固定タイムアウト，動的制御およびその手法）
- 再送信時の再送タイムアウトの決定手法（Karn のアルゴリズムを使うか否か）
- ウインドウサイズの変更手法（スロースタート方式を行うか否か）

- 送信ウインドウの補助的な機構である輻輳ウインドウの使用の有無

そのなかから本研究では UNIX 4.3 BSD Tahoe Release [LM+89] の実装に基づいてモデルを設計することにした。この理由は，この実装の TCP の各機能における戦略は，現在多くの OS で実装された TCP の模範となっていることである。

TCP のパラメータを以下に示す。

送信バッファサイズ: 4,096 octet.

受信バッファサイズ: 4,096 octet.

最大セグメント長: 1,460 octet. この値は Ethernet 上で動作する IP でセグメントのフラグメント化を必要としない。

再送タイムアウト制御方式: まず最初になめらかな RTT(Round-Trip Time) T_{srtt_n} を最期に測定した RTT T_{rtt} から計算する。

$$T_{srtt_n} := 0.9T_{srtt_{n-1}} + 0.1T_{rtt} \tag{2.21}$$

次回の送信に用いられる再送タイムアウト T_{rto} は以下のようにして計算される。

$$T_{rto} := 2T_{srtt_n} \tag{2.22}$$

T_{rto} [sec] は $1 \leq T_{rto} \leq 64.0$ に制限されている。TCP ソケットがセグメントを再送信する場合，T_{srtt_n} の評価は行われない。再送セグメントに対する再送タイムアウトの計算は Karn のアルゴリズム [CS91] に従って行われる。

最大再送回数: 無限大

本モデルでは TCP の全二重通進路が確立した状態における通信のみをシミュレートする。通信路の確立に伴う処理や，緊急データ送信処理は扱わない。したがって，本モデルでモデル化するのは TCP のスライディングウインドウによるデータ送受信のみである。これを実現するため，本モデルでは，TCP セグメントヘッダ情報（送信元，宛先アドレス，セグメントのシーケンス番号，ACK 番号，ウインドウサイズおよびデータ長）が TCP サブモデル間で交換される。

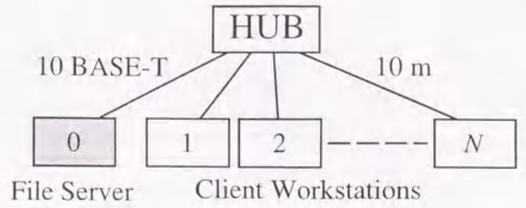


図 2.9: シミュレーションしたネットワーク構成

Fig. 2.9 Simulated network

TCP サブモデルで扱うイベントは，送信要求の発生，セグメントの受信，待機タイマ，再送タイマ，およびアプリケーション受信バッファの空き領域の増大である。これらのイベントの処理はそれぞれ UNIX 4.3 BSD の実装方式に従って処理される。

TCP サブモデルによりセグメントの送信イベントが発生すると，それは即時イベントとして IP サブモデルにより処理される。IP サブモデルが行う処理は，前述のようにセグメントに IP ヘッダ長の情報を付加して，Ethernet サブモデルに送信イベントを渡すだけである。TCP および IP サブモデルにおける CPU 処理時間は無視されている。また同様に IP サブモデルで受信イベントが発生すると IP サブモデルで IP ヘッダ長の情報が取り除かれ，TCP サブモデルでの受信イベントが即時イベントとして発生する。

2.4　シミュレーションモデルの検証

本節では，提案モデルによるシミュレーションと実システムにおける測定との比較により，提案モデルの確かさの検証を行う。

2.4.1　過渡的な負荷を持つ LAN を想定した条件

過渡的な変動負荷を持つ LAN の例として，教育用 LAN におけるクライアント・サーバシステムを取り上げる。このようなシステムでは授業利用時に教官の指示にしたがって利用者がほぼ同時に同じ処理を行う。このため，その処理で使用するファイルが LAN 上の異なる計算機にある場合には過渡的に高い負荷が発生することになる。

ここでは N 台のクライアントワークステーション (CWS) と 1 台のファイルサーバ

表 2.4: ユーザ挙動モデルのパラメータ

Model	μ [sec]	σ [sec]
Model M (mouse operation)	2.2	1.6
Model K (keyboard operation)	6.3	10.6
Model O (at-a-time)	1.0	0.0

(FS) から構成されるクライアントサーバシステムにおいて，FTP [Pos85] による FS から CWS のファイル転送要求が，すべての CWS からほぼ同時に発生する状況を想定する。FS と CWS は伝送速度 10 Mbps の Ethernet 10BASE-T で 1 台のリピータハブを介して接続されているとする（図 2.9）。各 CWS は以下で説明するタイミングにしたがって，FS にファイル転送要求を送信する。各 CWS i におけるファイル転送要求の発生タイミング $t_{start}^{(i)}$ は次に示す分布に従った疑似乱数によって決定される。

$$f(t_{start}) = \begin{cases} \dfrac{\exp\left(\dfrac{-(t_{start}-\mu)^2}{2\sigma^2}\right)}{\displaystyle\int_0^\infty \exp\left(\frac{-(\xi-\mu)^2}{2\sigma^2}\right) d\xi} & (t_{start} \geq 0) \\ 0 & (t_{start} < 0) \end{cases} \tag{2.23}$$

式 2.23 は乱数 $t_{start} < 0$ となるように正規分布 $N(\mu, \sigma)$ に 手を加えたものである。パラメータ μ と σ は教育用システムにおける利用者の応答時間の実測に基づいて決定した。利用者の応答時間の分布は計算機のユーザインタフェースによって異なると考えられる。そこで μ, σ の組としてマウス操作による応答を仮定したモデル M (Mouse)，キーボードによる応答を仮定したモデル K (Keyboard) および参照用に要求発生が全ての CWS で完全に同時に発生したと仮定するモデル O を用意した。これらのモデルをユーザ挙動モデルと呼ぶことにする。各ユーザ挙動モデルのパラメータを表 2.4 に示す。これらのモデルに関する詳細は第 4 章で改めて述べる。

各モデルで得られる t_{start} は，次に述べる各 CWS モデル上のアプリケーションモデルでの動作開始イベントの発生時刻として処理される。

2.4.2　アプリケーションモデル

TCP を利用するクライアント・サーバ型アプリケーションの例として，FTP (File Transfer Protocol) を仮定する。モデルでは FTP の動作を以下のように簡略化した。

1. CWS におけるクライアントプログラムは，ユーザ挙動モデルにより発生した動作開始イベントを受けると，FS 上のサーバプログラムに 20 bytes のファイル転送要求メッセージを送信する。
2. サーバプログラムは，CWS からの要求メッセージを受信すると，TCP を介して要求されたファイルのデータを CWS に送信する。サーバプログラムの送信バッファの容量は S_{server} とする。
3. CWS における TCP は受信データを容量が R_{client} bytes の TCP 受信バッファに格納する。クライアントプログラムは，平均 0.7 [sec] の指数乱数で決められる読みとり間隔で TCP の受信バッファからデータを 1024 bytes ずつ読み取る。
   読み込み時間の決定に用いる乱数の分布は実システムでの測定に基づき決定した。

本モデルの TCP では全二重通信路の確立処理は除いてあるので，アプリケーションのモデルでもその動作に全二重通信路の確立処理は含まれないものとしている。また，アプリケーションのモデルでは，バッファ読み込みに要する時間を除く CPU 処理時間およびファイルアクセス時間は無視している。

TCP，IP および Ethernet のヘッダを含む各メッセージの Ethernet のフレームサイズは以下のようになる。

- クライアントからの要求メッセージ： 74 bytes。
- サーバからの送信データ列： min{Maximum segment size of TCP, S} + 58 bytes 以下。最小値は 64 bytes。この値は受信側である CWS 上の TCP 受信バッファの空き容量によって変化する。
- クライアントからの TCP 確認応答メッセージ： 64 bytes。

2.4.3　シミュレーション結果

以下シミュレーション結果と実システムにおける実測結果との比較を示す。

■ 1 対 1 通信

最初に，$N = 1$ のときのシミュレーション結果を示す。図 2.10 および表 2.5 は，サーバプログラムにおける送信バッファサイズ S_{server} に対する，サーバ応答時間 $T_{trs}^{(1)} = t_{end}^{(1)} - t_{start}^{(i)}$

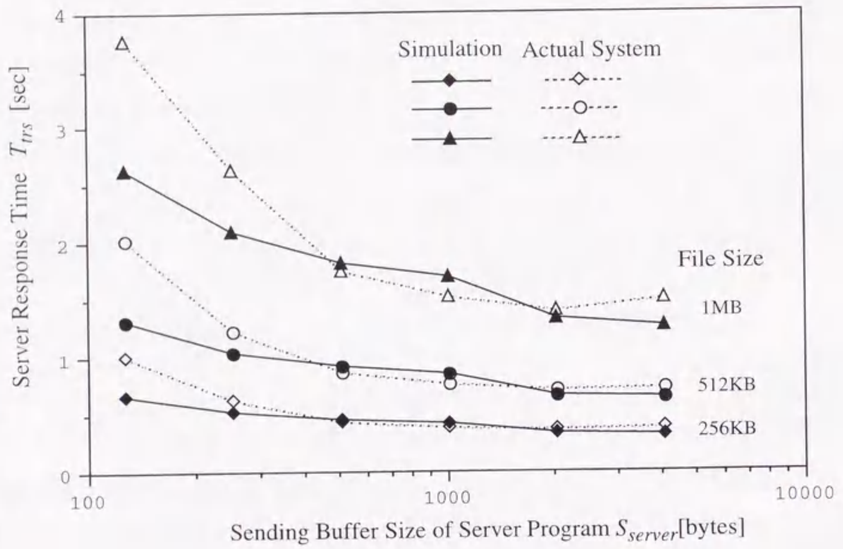


図 2.10: サーバプログラムの送信バッファサイズに対する平均サーバ応答時間の変化

Fig. 2.10 The mean server response time versus the sending buffer size of the server program.

表 2.5: サーバプログラムの送信バッファサイズに対する平均サーバ応答時間の変化

(a) ファイルサイズ = 256 Kbytes

送信バッファ	平均サーバ応答時間 [sec]			
サイズ [bytes]	シミュレーション	95 % 信頼区間	実測値	95 % 信頼区間
128	0.655	0.002	1.011	0.026
256	0.523	0.002	0.625	0.014
512	0.457	0.003	0.440	0.010
1024	0.427	0.005	0.386	0.011
2048	0.332	0.005	0.360	0.004
4096	0.319	0.005	0.386	0.010

(b) ファイルサイズ = 512 Kbytes

送信バッファ	平均サーバ応答時間 [sec]			
サイズ [bytes]	シミュレーション	95 % 信頼区間	実測値	95 % 信頼区間
128	1.311	0.003	2.025	0.047
256	1.043	0.003	1.222	0.015
512	0.912	0.004	0.861	0.014
1024	0.847	0.007	0.768	0.011
2048	0.668	0.007	0.705	0.012
4096	0.635	0.006	0.737	0.010

(c) ファイルサイズ = 1024 Kbytes

送信バッファ	平均サーバ応答時間 [sec]			
サイズ [bytes]	シミュレーション	95 % 信頼区間	実測値	95 % 信頼区間
128	2.621	0.005	3.752	0.055
256	2.089	0.006	2.635	0.129
512	1.820	0.006	1.753	0.022
1024	1.690	0.008	1.528	0.017
2048	1.333	0.006	1.404	0.016
4096	1.263	0.016	1.504	0.020

の変化を示している。ここで $t_{end}^{(i)}$ は，CWS i のアプリケーションモデルが要求したファイルをすべて受信した時刻である。クライアントの TCP における受信バッファサイズは 4096 bytes に固定してある。

また，伝送速度が同じ Ethernet 10BASE-5（同軸ケーブルによるバス型接続）によるネットワーク構成における実システムでの測定結果も図 2.10 に併せて示す。この実測結果は TCP における全二重通信路の確立処理に伴うオーバヘッドを含んでいる。実システムの物理ネットワークの構成はシミュレーションで用いたものとは異なるが，論理的な構成はシミュレーションで想定した条件に等しく，2.3.2 で示したネットワーク規模の条件を満たす。

$S_{server} \geq 512$ [bytes] のとき，シミュレーション結果は実測結果と良く一致している。一方，S_{server} が小さい時，シミュレーション誤差が増加している。送信バッファが小さい場合，TCP が送信するセグメントのサイズも小さくなる。このため TCP およびアプリケーションでの送受信処理に伴う CPU 時間がデータ送信時間に対して相対的に大きくなる。シミュレーションモデルでは CPU 時間を考慮していないので，送信バッファサイズが小さい時に誤差が大きくなったと考えられる。ただし，現実的にはデータ送信を行うアプリケーションで 512 bytes より小さいバッファサイズを用いることはほとんどないので，現実的には上記の誤差は問題にはならないだろう。

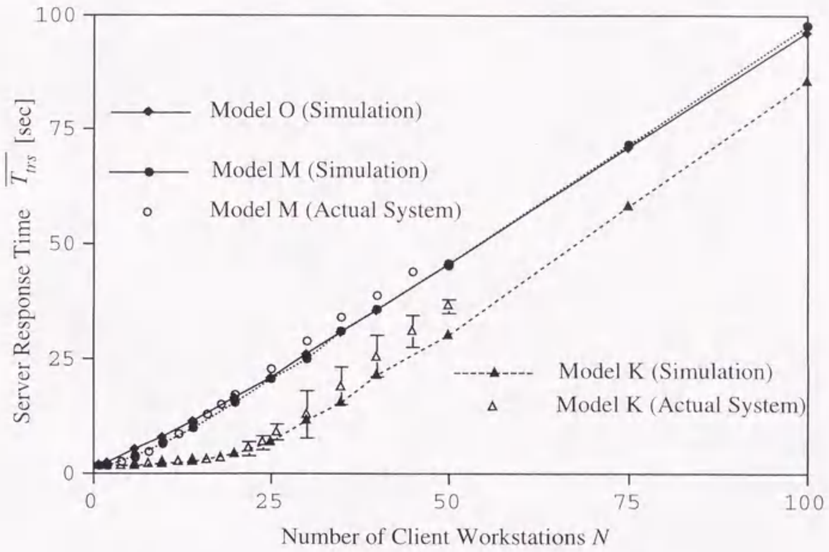
■ 1 対 N 通信

次に，$N > 1$ の場合の平均サーバ応答時間 $\overline{T_{trs}}$ の CWS 数 N に対する変化を実システムにおける測定値（ただし $N \leq 50$）とともに図 2.11 および表 2.6，2.7，2.8 に示す。$\overline{T_{trs}}$ は以下の式で与えられる。

$$\overline{T_{trs}} = \overline{t_{end}^{(i)} - t_{start}^{(i)}} = \frac{1}{N}\sum_{i=1}^{N} T_{trs}^{(i)} \tag{2.24}$$

転送ファイルサイズは 1 Mbytes とし，R_{server} と R_{client} はそれぞれ 1024 bytes に固定した。図示した値は複数回のシミュレーション試行結果の平均値である。シミュレーション試行回数は N に応じて異なる（表 2.9）。なお，実システム即定時のファイル転送要求発生時刻 $t_{start}^{(i)}$ は，シミュレーションプログラムと同じプログラムを用いて発生させた。

ユーザ挙動モデル M，O での サーバ応答時間 $\overline{T_{trs}}$ は，モデル M で $N < 10$ の領域でその増加率が若干小さいものの，ほぼ N に比例して増加する。一方，モデル K では



Error-bar: 95 % Confidence interval of the measured value

図 2.11: CWS 台数に対する平均サーバ応答時間の変化

Fig. 2.11 The mean server response time versus the number of clients.

表 2.6: CWS 台数に対する平均サーバ応答時間の変化 (Model M)

	平均サーバ応答時間 [sec]			
N	シミュレーション	95%信頼区間	実測値	95%信頼区間
1	1.686	0.007	1.738	0.114
2	1.806	0.069	1.916	0.133
4	—	—	2.562	0.326
6	3.938	0.357	3.549	0.683
8	—	—	5.066	0.622
10	6.517	0.336	6.935	0.838
12	—	—	8.717	0.568
14	10.173	0.394	11.219	0.526
16	—	—	13.206	0.56
18	—	—	15.426	0.475
20	15.821	0.293	17.595	0.392
25	20.798	0.342	23.131	0.528
30	25.229	0.417	28.841	0.847
35	31.015	0.299	34.044	0.816
40	35.789	0.421	38.843	0.465
45	—	—	44.071	0.574
50	45.77	0.569	45.49	0.874
75	71.676	0.742	—	—
100	97.469	1.616	—	—

$N < 20$ のとき，$\overline{T_{trs}}$ は小さい値を保つが，$N = 20$ 付近で急激に増加し，その後モデル M の場合とほぼ同じ増加率で増加する。シミュレーション結果は $N < 50$ の N に対して実測値の 90 本シミュレーションモデルによる結果は高負荷かつ，過渡的な負荷変動がある LAN の特性を調べるためのツールとして十分であるといえよう。

しかしながら，モデル K における $25 < N < 50$ の領域では実測値とシミュレーション結果の平均値との差は大きくなっている。また，この領域では実測値の信頼区間幅も広い。この理由は以下のように考えられる。モデル K で N が小さい場合に T_{trs} がほとんど増加しないのは，その条件では CWS からの要求発生がまばらであり，複数のクライアント・サーバ間通信が並行して行われていることがほとんどないからだと考えられる。一方，$N > 25$ では，徐々に平行して行われるクライアント・サーバ間通信が増えるため，T_{trs} の増加率はモデル M，O の場合に近づく。クライアント・サーバ間通信が平行して行われる状況は乱数によって与えられる $t_{start}^{(i)}$ に依存する。このため，測定回数およびシミュレーション試行回数が少ない場合には信頼区間幅は広くなり，実測値とシミュレーション結果の相違が生まれる。

図 2.12 および表 2.10 はシミュレーションにおけるネットワーク全体の Ethernet 平均

表 2.7: CWS 台数に対する平均サーバ応答時間の変化 (Model K)

	平均サーバ応答時間 [sec]			
N	シミュレーション	95%信頼区間	実測値	95%信頼区間
1	1.686	0.007	1.723	0.061
2	1.712	0.048	1.851	0.156
4	—	—	1.838	0.136
6	1.866	0.097	2.118	0.281
8	—	—	2.221	0.397
10	2.295	0.212	2.546	0.388
12	—	—	2.619	0.373
14	2.652	0.293	2.957	0.464
16	—	—	3.19	0.604
18	—	—	3.593	0.755
20	4.624	0.615	4.412	1.137
22	—	—	5.533	1.433
24	—	—	6.816	1.581
25	7.066	0.748	—	—
26	—	—	9.19	1.792
30	11.687	1.389	13.134	5.31
35	15.591	1.633	19.314	4.291
40	21.649	1.489	25.745	4.61
45	—	—	31.223	3.275
50	30.531	1.273	36.627	1.494
75	58.48	1.531	—	—
100	85.376	1.943	—	—

表 2.8: CWS 台数に対する平均サーバ応答時間の変化 (Model O)

	平均サーバ応答時間 [sec]	
N	シミュレーション	95%信頼区間
1	1.687	0.006
2	2.157	0.004
6	5.388	0.146
10	7.950	0.212
14	11.348	0.284
20	16.318	0.252
25	21.045	0.387
30	25.815	0.364
35	30.592	0.459
40	35.645	0.582
50	45.543	0.846
75	71.089	0.649
100	95.765	2. 100

表 2.9: シミュレーションの実行回数

N	Number of trial
1...25	20
30, 35	15
40...75	10
100	5

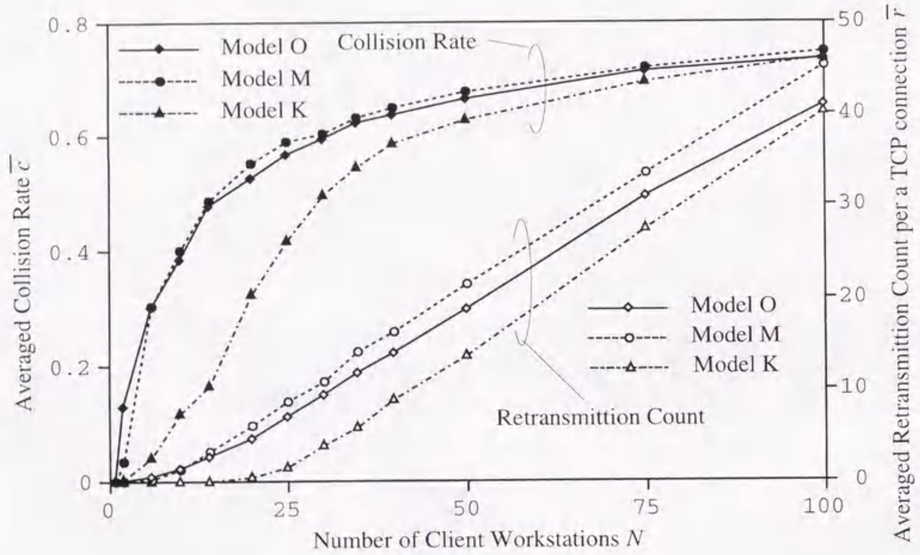


図 2.12: CWS 台数に対する平均衝突率と TCP 再送回数の変化

Fig. 2.12 The TCP retransmission count and the Ethernet collision rate vs. the number of clients.

表 2.10: CWS 台数に対する平均衝突率と TCP 再送回数の変化

N	平均衝突率			平均 TCP 再送回数		
	Model M	**Model K**	**Model O**	**Model M**	**Model K**	**Model O**
1	0.000	0.000	0.000	0.000	0.000	0.000
2	0.034	0.006	0.129	0.000	0.000	0.000
6	0.330	0.044	0.303	0.200	0.000	0.450
10	0.422	0.120	0.384	1.300	0.000	1.300
14	0.506	0.166	0.476	3.200	0.000	2.550
20	0.570	0.326	0.525	6.000	0.400	4.550
25	0.605	0.418	0.566	8.750	1.650	7.050
30	0.617	0.499	0.594	10.933	3.867	9.267
35	0.644	0.546	0.621	14.067	5.867	11.800
40	0.659	0.588	0.636	16.200	9.000	13.900
50	0.686	0.629	0.663	21.500	13.700	18.700
75	0.724	0.694	0.711	33.600	27.500	31.000
100	0.750	0.735	0.733	45.200	40.200	40.800

衝突率 $\bar{c}$ の N に対する変化を示したものである。ここで $\bar{c}$ は以下のように定義される。

$$\bar{c} = \frac{1}{N+1} \sum_{i=0}^{N} \frac{n_c^{(i)}}{n_a^{(i)}} \tag{2.25}$$

$n_a^{(i)}$ と $n_c^{(i)}$ はそれぞれ端末 i の Ethernet サブモデルにおけるフレーム送信試行回数と衝突回数である。端末 $i = 0$ は FS を指す。図 2.12 ではまた，各クライアント・サーバ間の TCP 接続における平均再送回数 $\bar{r}$ も示す。$\bar{c}$ は N が小さい時に急激に増加し，その後緩やかな上昇を見せる。一方，$\bar{r}$ は $\bar{c} > 0.3$ となる N 付近から急激に増加している。また，$\bar{r}$ は N に対して図 2.11 で示した T_{trs} とほぼ同じ特性を示す。

注目するべきは，モデル M における $\bar{r}$ がモデル O の場合よりも大きくなっていることである。このことから TCP の再送タイムアウトの動的制御が t_{start} が分散しているために，TCP の各通信路における送信開始後の RTT の変化の急速な変化に適切に作用していないことが推測できる。しかしながら，$\bar{c} > 0.3$ における $\overline{T_{trs}}$ と $\bar{r}$ の増加率はほぼ一定であり，Ethernet における衝突と TCP の再送回数の増加によるネットワークシステムの破綻は観測されていない。

このシミュレーション結果は教育用システムにおける理想的な LAN 構成の指針を与える。すなわち，サーバ応答時間および Ethernet 上での衝突率を低く保つ理想的な状況をたもつには，伝送速度 10 Mbps の Ethernet での 1 台の FS に対する CWS の台数は最大 25 台程度が適切であるといえる。

最後に本モデルでのシミュレーション時間について記す。シミュレーションに要する時

間は N，アクセス集中モデル，伝送速度，転送ファイルサイズに依存する。今回のシミュレーションには Sun Microsystems 社の SPARC center 2000 を用いたが，$N = 20$，アクセス集中モデル M，伝送速度 10 Mbps 転送ファイルサイズ 1 MB のとき，1 回のシミュレーションに要した時間は 39 秒であった。

2.5　まとめ

本章では過渡的な負荷変動，高負荷という性質をもつ LAN の正確な性能評価のための効率的なシミュレーション手法を提案した。本手法ではネットワーク上の端末のモデルは多階層のプロトコルサブモデルにより構成され，それらは並列にイベント駆動で動作する。プロトコル挙動に関する省略，近似は行われていないので，高負荷および高い負荷変動がある条件においても正確な性能予測が可能である。効率的なシミュレーションのため，CSMA/CD における衝突判定をこの手法ではシミュレーションの正確さを損なうことなく高速化する手法を用いた。シミュレーションモデルの妥当性は，現実の LAN における測定値との比較によって確認された。

本手法の応用として，集合型情報処理教育施設における高負荷かつ負荷変動の高い LAN の性能評価を行った。この評価により，単純な 1 階層のシミュレーションモデルや解析的手法では評価困難な LAN の挙動の詳細を観察することができた。なお，今後の課題として，ブリッジやルータを含む複雑なネットワーク構成のモデル化とさまざまなネットワークアプリケーションのモデル化が必要である。

第3章　可視化に適したネットワーク性能指標の提案

本章では CSMA/CD LAN のための一評価指標としてストール率を提案する。ストール率はパケット送信遅延に占める送信待機時間の割合をその原因別に示した値である。CSMA/CD LAN の送信遅延を増大させる要因は，伝送路の空きを待つ時間と衝突後のバックオフによる送信待機時間に大別される。ストール率ではこれらによるネットワークの性能低下とその原因を直観的，視覚的に把握することが容易であり，CSMA/CD LAN の詳細についての専門知識を持たないプロトコル上位層の設計者がシステム評価を行う上で有用である。本章ではストール率の定義とその一般的性質について明らかする。またストール率を用いて一般アプリケーション利用時の Ethernet LAN の性能について評価し，本指標の有効性を示す。

3.1　はじめに

近年の LAN の普及およびそれに伴うシステムの複雑化により，ネットワーク管理者および設計者の負担が増加する傾向にある。彼らの業務にとって，ネットワーク動作状況のモニタリングおよび予測は，重要な位置を占めるが，ネットワークアナライザ，シミュレータによって示される性能値から，システムの問題追求を行うためには，高度な専門知識を要する。

スループットや送信遅延は，ネットワークの性能評価指標として広く用いられている

が，それらが示す値はあくまでもネットワークの性能そのものである。システム設計者の立場からは，それらのような結果そのものを表す性能指標はもちろんのこと，その性能が得られた原因を明らかに示すことのできる評価指標が求められる。また，複数の異なる指標を用いることにより，従来明らかにできなかったネットワークの特性を発見できることも期待される。

Ethernet に代表される CSMA/CD ネットワークの性能変化の原因を追求するための手段としては，従来より衝突数，送信遅延の分布およびネットワーク利用率等が用いられてきた [BMK88][Mol94][MKk97]。Ethernet の仕様上の問題とされる Capture Effect [1][WSF94] は Molle による送信遅延の分布の調査 [Mol94] によってその現象の詳細が示されている。しかし，彼らの評価で行われたシミュレーションでは，上位層プロトコルのモデルは単純化されており，アプリケーション利用時における Ethernet のトラヒック特性については十分な評価が行われていなかった。一方，石田らは実際に運用中の LAN 上での特定のアプリケーション利用時におけるパケット長およびパケット発生間隔を測定し，この結果に基づいて Ethernet 単体レベルでのシミュレーション [石田 95] を行っている。しかし，性能評価はスループットおよび送信遅延特性のみにとどまっており，アプリケーションに依存するトラヒック特性の詳細を評価するには至っていない。

本章では，CSMA/CD ネットワークの性能評価指標の一つとしてストール率を提案する [IO97b][石原 99]。ストール率は概念的には，フレーム送信遅延に対する，伝送路が使用中のために待機していた時間および，衝突発生後のバックオフによる待ち時間の割合と要約される。ストール率によればネットワーク特性の直観的な理解が容易であり，アプリケーションレベルでのネットワーク特性評価を行なう際には，ネットワーク管理者等にとって有用な指標であると考える。

以下，まず **3.2** でストール率の定義およびそれによって表されるネットワークの特性について説明する。次に **3.3** で高負荷ネットワーク上でのアプリケーション利用時を想定したシミュレーションの概略を示し，ストール率を用いた評価結果を上位層プロトコルの特性と照らし合わせて示す。

[1]複数端末の送信の結果衝突が起きた場合，そのフレームにおける衝突回数が少ない端末の方が，常に次に伝送路を占有できる確率が高くなる現象

3.2　ストール率

3.2.1　CSMA/CD 方式における送信遅延

まず CSMA/CD 方式の概略について Ethernet の例で簡単に説明する [ANS93]。Ethernet では，各端末は共有伝送路の状態を確認後，伝送路が空きならばパケット送信を開始し，伝送路が使用中であれば伝送路が空くまで待ってから送信する。送信開始後，他の端末との同時送信開始の結果パケットの衝突が起きると，ジャム信号を送信後，二進指数バックオフによりランダム時間だけ待ってから送信を再試行する。衝突が規定回数（Ethernet では 16 回）に達した場合は再送信を行わない。CSMA/CD LAN においては送信遅延を増す大きな要因は，(i) 伝送路使用中による送信待ち時間 T_{CS}，(ii) 衝突後のバックオフによる待ち時間 T_{WT}，の 2 つである。

CSMA/CD LAN の性能指標のひとつとして広く用いられているスループットは T_{CS} と T_{WT} を区別なく総合的に評価するものであり，衝突数または衝突率は T_{WT} について評価するものである。これらの指標はネットワークアナライザで容易に取得可能であるが，これらのみから LAN 上での現象を直観的に把握するのは困難である。一方，詳細なシミュレーションによる評価を行う場合，T_{CS} と T_{WT} はそれぞれ容易に取得可能である。T_{CS}, T_{WT} それぞれについて送信遅延との比較を行うことで，これらの直観的かつ総合的な評価が可能であると考える。

3.2.2　ストール率の定義

ストール率は，パケットの送信を開始してから送信が成功または 16 回の衝突によって終了するまでの時間（パケット送信遅延）に対する伝送路使用中による送信待ち時間の比率，および衝突後のバックオフによる待ち時間の比率を示したものである。ここで，端末 i における測定期間内のパケット送信遅延の和を $T_{\mathrm{XMT}}^{(i)}$，衝突後のバックオフによる待ち時間の和を $T_{\mathrm{WT}}^{(i)}$，伝送路使用中による送信待ち時間の和を $T_{\mathrm{CS}}^{(i)}$ とする。これより，N 台の端末における平均ストール率は以下のように定義される。

平均 CS ストール率

$$R_{\mathrm{CS}} = \frac{\sum_{i=1}^{N} T_{\mathrm{CS}}^{(i)}}{\sum_{i=1}^{N} T_{\mathrm{XMT}}^{(i)}} \tag{3.1}$$

平均 WT ストール率

$$R_{\mathrm{WT}} = \frac{\sum_{i=1}^{N} T_{\mathrm{WT}}^{(i)}}{\sum_{i=1}^{N} T_{\mathrm{XMT}}^{(i)}} \tag{3.2}$$

平均トータルストール率

$$R_{\mathrm{T}} = R_{\mathrm{CS}} + R_{\mathrm{WT}} \tag{3.3}$$

CS (Carrier Sensing) ストール率は送信遅延に対する伝送路の空き待ちのために送信できなかった割合を示し，WT (WaiTing) ストール率は衝突後のバックオフによる送信再試行の待ち時間の割合を示す。

3.2.3　ストール率とトラヒック特性

ここでは，ストール率と CSMA/CD LAN の特性との関連について説明する。

■ CS ストール率

伝送路の空きを待っている時間 T_{CS} は，ネットワークの負荷およびネットワーク全体の平均パケット長に依存する。ネットワークの負荷が高いほど，パケット送信時に伝送路が占有されている確率が高い。また，伝送路が占有されている場合の待ち時間は平均パケット長に比例する。したがって，

- R_{CS} はネットワークの負荷が高いほど大きな値を示す。
- ネットワーク内の一部の端末について R_{CS} を評価した場合，それら端末における平均パケット長に対してネットワーク全体の平均パケット長が長ければ R_{CS} は大きくなる。

■ WT ストール率

Ethernet における衝突後の待ち時間 T_{WT} は，二進指数バックオフ調停アルゴリズム [ANS93] によって決定されるので，1 つのパケット送信あたりの衝突回数に対して指数的

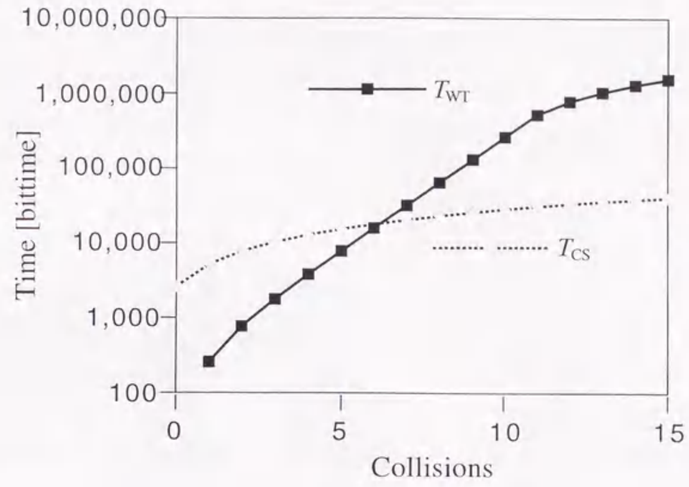


図 3.1: 連続衝突回数に対する T_{CS}, T_{WT} の変化

Fig. 3.1 T_{CS} and T_{WT} versus the number of Ethernet collisions.

に増加する。一方，1 回の送信試行あたりの T_{CS} は最大で最大パケット長と等しく，全送信試行において伝送路が他の端末に専有されていた場合においても，T_{CS} の平均はネットワーク全体の平均パケット長の半分である。

図 3.1 表 3.1 に連続衝突回数に対する 1 つのパケット送信における T_{WT}, T_{CS} を示す。ここでは平均パケット長を 649 バイトとした。この平均パケット長は，Gusella [Gus90] による LAN 上のパケット長分布の実測値に基づく。この図より，連続衝突回数が少ない時には T_{CS} が T_{WT} よりも大きいが，連続衝突回数が増加すれば T_{WT} が劇的に増加し T_{CS} を上回ることが明らかである。したがって，

- R_{WT} はネットワーク上で衝突が多く発生すればするほど，衝突が連続して発生すればするほど大きくなる。
- 連続した衝突の発生により，R_{WT} の値は R_{CS} を上回る。
- 大きな R_{CS} はネットワーク負荷の増大を意味するが，R_{WT} が小さい場合，連続した衝突の発生は少ない。

表 3.1: 連続衝突回数に対する $T_{\mathrm{CS}}, T_{\mathrm{WT}}$ の変化

衝突回数	平均バックオフ時間 [bt]	平均 T_{wt} [bt]	平均 T_{cs} [bt]
0	0	0	2596
1	256	256	5192
2	512	768	7788
3	1024	1792	10384
4	2048	3840	12980
5	4096	7936	15576
6	8192	16128	18172
7	16384	32512	20768
8	32768	65280	23364
9	65536	130816	25960
10	131072	261888	28556
11	262144	524032	31152
12	262144	786176	33748
13	262144	1048320	36344
14	262144	1310464	38940
15	262144	1572608	4153

bt = ビット時間（10Mbps の場合 1 bt = 0.1 μsec）

表 3.2: パケット長分布

Length [byte]	Probability
64	0.304
144	0.083
220	0.080
576	0.100
1072	0.250
1500	0.183
Average	649.1 bytes

■ トータルストール率

トータルストール率 R_{T} は R_{CS} と R_{WT} の和であるから，

- R_{T} はネットワークへの送信が滞った割合を示す。
- $1 - R_{\mathrm{T}}$ は送信ノードから見た実効スループットを示す。
- R_{T} に対する $R_{\mathrm{CS}}, R_{\mathrm{WT}}$ の内訳はネットワークへの送信が滞った原因を示している。

表 3.2 で示したパケット長分布 [Gus90] のデータトラヒック発生を第 2 章のシミュレーションモデルでシミュレーションした。ネットワークは伝送速度 10 Mbps の Ethernet

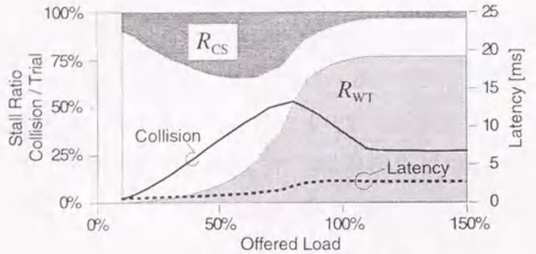
10BASE-T で 1 台のリピータハブにすべての端末が 10 m のケーブルで接続されているとした。送信要求の発生間隔は指数分布とした。

ストール率を図 3.2 および表 3.3 ～3.6 に示す。同図には参考のため衝突率（パケット送信試行回数に対する衝突回数の比）とパケット送信遅延を併せて示した。

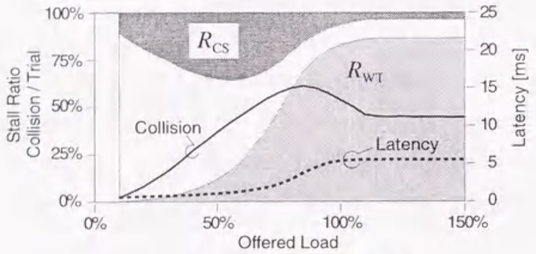
図 3.2 中で白く塗られていない部分が $R_{\mathrm{T}} = R_{\mathrm{CS}} + R_{\mathrm{WT}}$ を示している。ネットワークの負荷が増大するにつれて白い部分が減少し，ネットワークインタフェースが送信処理を行っていても実際には信号が送信されなくなることがわかる。また $R_{\mathrm{CS}}, R_{\mathrm{WT}}$ の値から送信効率を減少させる要因が直観的に理解できる。

負荷 70 % 程度までは R_{CS} が大きな値を占め，R_{WT} は小さい。これはある端末が送信を開始しようとした場合に伝送路が占有されていた場合，伝送路が空いた時点で送信を開始すればすぐに伝送路を占有できる確率が高いことを示す。言い換えれば，伝送路の空きを待っている端末がたかだか 1 台であり，伝送路が空いた時点で送信を開始しても他の端末の送信と衝突することがないことを示している。

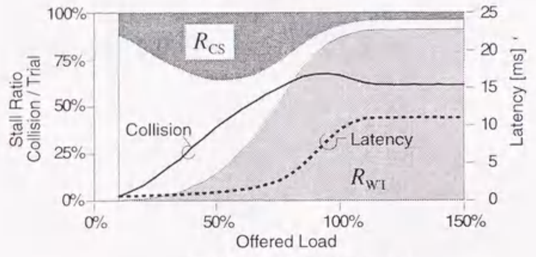
R_{WT} は負荷が増大するにつれて衝突率ともに増大し，負荷 70 % 程度で R_{CS} 送信遅延の大半を占める様子がよくわかる。端末数が少ない場合には，負荷 70 % を越えると衝突率は低下しているが，依然として衝突後の待ち時間により送信効率が低下していることがはっきりと読みとれる。



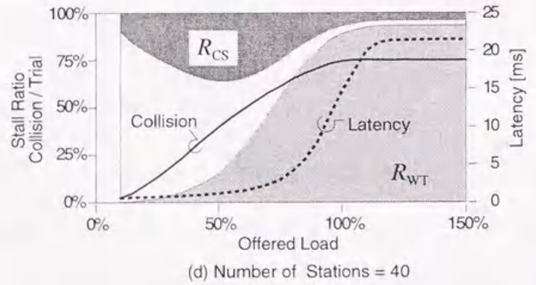
(a) Number of Stations = 5



(b) Number of Stations = 10



(b) Number of Stations = 20



(d) Number of Stations = 40

図 3.2: データトラヒック発生時の平均ストール率

Fig. 3.2 The mean stall ratio at Ethernet with general data traffic.

表 3.3: データトラヒック発生時の平均ストール率 (端末数 = 5)

入力負荷	送信遅延 [msec]	衝突率	**CS** ストール率 R_{CS}	**WT** ストール率 R_{WT}	トータルストール率 R_T
0.10	0.614	0.021	0.098	0.002	0.100
0.15	0.636	0.043	0.125	0.005	0.130
0.20	0.681	0.074	0.176	0.009	0.185
0.25	0.711	0.111	0.210	0.015	0.225
0.30	0.760	0.149	0.249	0.022	0.271
0.35	0.796	0.194	0.268	0.033	0.301
0.40	0.852	0.237	0.298	0.049	0.347
0.45	0.913	0.286	0.320	0.068	0.388
0.50	0.979	0.331	0.337	0.090	0.427
0.55	1.055	0.374	0.345	0.122	0.467
0.60	1.151	0.419	0.346	0.166	0.511
0.65	1.291	0.457	0.342	0.219	0.561
0.70	1.513	0.499	0.309	0.313	0.622
0.75	1.672	0.515	0.286	0.374	0.660
0.80	2.194	0.532	0.209	0.528	0.737
0.85	2.581	0.500	0.143	0.636	0.779
0.90	2.757	0.464	0.107	0.687	0.794
0.95	2.851	0.418	0.081	0.722	0.803
1.00	2.843	0.372	0.061	0.743	0.804
1.05	2.817	0.328	0.047	0.757	0.804
1.10	2.789	0.282	0.036	0.766	0.802
1.15	2.772	0.275	0.034	0.767	0.801
1.20	2.768	0.273	0.034	0.767	0.801
1.25	2.768	0.272	0.034	0.768	0.801
1.30	2.766	0.272	0.033	0.768	0.801
1.35	2.762	0.270	0.033	0.768	0.801
1.40	2.765	0.269	0.033	0.768	0.801
1.45	2.762	0.271	0.033	0.768	0.801
1.50	2.762	0.271	0.034	0.768	0.800

表 3.4: データトラヒック発生時の平均ストール率 (端末数 = 10)

入力負荷	送信遅延 [msec]	衝突率	**CS** ストール率 R_{CS}	**WT** ストール率 R_{WT}	トータルストール率 R_{T}
0.10	0.621	0.022	0.110	0.002	0.113
0.15	0.652	0.051	0.154	0.006	0.160
0.20	0.693	0.082	0.195	0.010	0.205
0.25	0.733	0.123	0.227	0.018	0.245
0.30	0.774	0.166	0.257	0.027	0.284
0.35	0.836	0.216	0.292	0.042	0.334
0.40	0.904	0.272	0.319	0.065	0.383
0.45	0.972	0.318	0.337	0.087	0.423
0.50	1.063	0.369	0.352	0.120	0.472
0.55	1.179	0.417	0.359	0.160	0.519
0.60	1.332	0.460	0.355	0.219	0.573
0.65	1.566	0.504	0.334	0.306	0.635
0.70	1.855	0.544	0.307	0.383	0.690
0.75	2.307	0.577	0.264	0.485	0.749
0.80	2.970	0.600	0.208	0.596	0.805
0.85	3.832	0.611	0.157	0.692	0.848
0.90	4.524	0.600	0.118	0.755	0.872
0.95	5.094	0.572	0.086	0.802	0.888
1.00	5.372	0.537	0.066	0.829	0.895
1.05	5.484	0.503	0.053	0.846	0.899
1.10	5.541	0.459	0.040	0.861	0.901
1.15	5.540	0.449	0.038	0.862	0.901
1.20	5.537	0.447	0.038	0.863	0.901
1.25	5.548	0.449	0.038	0.863	0.901
1.30	5.536	0.446	0.038	0.863	0.901
1.35	5.547	0.447	0.038	0.863	0.901
1.40	5.543	0.447	0.038	0.863	0.901
1.45	5.541	0.444	0.037	0.864	0.901
1.50	5.530	0.445	0.037	0.864	0.900

表 3.5: データトラヒック発生時の平均ストール率 (端末数 = 20)

入力負荷	送信遅延 [msec]	衝突率	**CS** ストール率 R_{CS}	**WT** ストール率 R_{WT}	トータルストール率 R_{T}
0.10	0.628	0.024	0.119	0.002	0.121
0.15	0.646	0.052	0.148	0.007	0.155
0.20	0.691	0.082	0.190	0.010	0.200
0.25	0.736	0.127	0.233	0.020	0.252
0.30	0.798	0.179	0.268	0.033	0.301
0.35	0.843	0.222	0.297	0.047	0.343
0.40	0.937	0.285	0.326	0.078	0.404
0.45	1.021	0.335	0.348	0.102	0.450
0.50	1.139	0.391	0.361	0.144	0.505
0.55	1.263	0.436	0.359	0.194	0.552
0.60	1.479	0.484	0.350	0.265	0.615
0.65	1.704	0.523	0.334	0.330	0.664
0.70	2.127	0.565	0.300	0.429	0.728
0.75	2.606	0.596	0.262	0.515	0.778
0.80	3.493	0.631	0.214	0.618	0.833
0.85	4.761	0.659	0.166	0.710	0.877
0.90	6.357	0.673	0.123	0.785	0.908
0.95	8.000	0.675	0.093	0.835	0.927
1.00	9.422	0.666	0.070	0.869	0.939
1.05	10.331	0.647	0.054	0.891	0.946
1.10	10.912	0.627	0.044	0.905	0.949
1.15	10.989	0.617	0.041	0.909	0.950
1.20	11.001	0.617	0.041	0.910	0.950
1.25	10.999	0.615	0.040	0.910	0.950
1.30	11.051	0.617	0.040	0.910	0.950
1.35	11.026	0.615	0.040	0.911	0.951
1.40	11.043	0.616	0.040	0.910	0.950
1.45	11.004	0.615	0.040	0.910	0.950
1.50	11.030	0.615	0.040	0.910	0.95

表 3.6: データトラヒック発生時の平均ストール率 (端末数 = 40)

入力負荷	送信遅延 [msec]	衝突率	CS ストール率 R_{CS}	WT ストール率 R_{WT}	トータルストール率 R_T
0.10	0.610	0.024	0.098	0.003	0.100
0.15	0.659	0.050	0.161	0.006	0.167
0.20	0.704	0.090	0.203	0.012	0.215
0.25	0.751	0.133	0.242	0.020	0.262
0.30	0.806	0.184	0.275	0.036	0.311
0.35	0.866	0.234	0.306	0.053	0.359
0.40	0.953	0.290	0.332	0.082	0.413
0.45	1.039	0.343	0.351	0.109	0.460
0.50	1.154	0.395	0.361	0.152	0.513
0.55	1.321	0.444	0.359	0.213	0.572
0.60	1.545	0.490	0.347	0.285	0.632
0.65	1.866	0.535	0.323	0.370	0.693
0.70	2.354	0.578	0.289	0.465	0.754
0.75	2.980	0.612	0.248	0.557	0.806
0.80	4.122	0.651	0.203	0.655	0.858
0.85	5.680	0.681	0.160	0.736	0.896
0.90	7.882	0.706	0.123	0.802	0.925
0.95	11.415	0.732	0.090	0.858	0.948
1.00	14.601	0.744	0.071	0.890	0.960
1.05	17.316	0.751	0.058	0.909	0.967
1.10	19.993	0.753	0.047	0.925	0.972
1.15	21.089	0.752	0.043	0.931	0.974
1.20	21.359	0.751	0.041	0.933	0.975
1.25	21.404	0.751	0.041	0.933	0.975
1.30	21.443	0.750	0.041	0.933	0.975
1.35	21.463	0.750	0.041	0.934	0.975
1.40	21.457	0.750	0.041	0.934	0.975
1.45	21.500	0.750	0.041	0.934	0.975
1.50	21.482	0.750	0.041	0.934	0.970

3.3 アプリケーション利用時の Ethernet 特性評価

本節では現実的な高負荷 LAN 環境の一例として，集合情報処理教育環境におけるクライアントサーバシステムをとりあげ，この環境下でネットワークアプリケーションを利用した場合の CSMA/CD LAN の特性をシミュレーションにより評価する。

3.3.1 場面設定

■ ネットワーク

1 台のファイルサーバと N 台のクライアントワークステーション WS が 1 台の HUB に 10 m のケーブルを介して接続されている Ethernet 10 BASE-T (10 Mbps) LAN を仮定する。

■ トラヒック発生源

集合情報処理教育用システムで最もネットワークアクセスが集中するのは，授業利用時に教官の指示のもとに学生が一斉にネットワークアクセスを伴う操作を行うときである。教官の指示後の学生の反応時間をモデル化するため，各クライアント WS がサーバ WS に対してリクエストを送信する時刻 t_s の分布は次式により与えられる密度関数 $f(t_s)$ に従うとした。

$$f(t_{start}) = \begin{cases} \dfrac{\exp\left(\dfrac{-(t_{start}-\mu)^2}{2\sigma^2}\right)}{\displaystyle\int_0^\infty \exp\left(\frac{-(\xi-\mu)^2}{2\sigma^2}\right) d\xi} & (t_{start} \geq 0) \\ 0 & (t_{start} < 0) \end{cases} \tag{3.4}$$

式 3.4 は正規分布において $t_{start} < 0$ のとき $f(t_{start}) = 0$ とした分布である。パラメータ μ, σ の決定には，情報処理教育環境 [岡田 96] におけるユーザの応答時間の実測結果 [石原 96] を用いた。この測定では学生が教官の指示に対してマウスによるコマンド入力で応答するまでの時間を計測している。この結果より $\mu = 2.2$ [sec]，$\sigma = 1.6$ [sec] とした。図 3.3 に，測定結果と本シミュレーションで仮定したモデルにおける t_{start} の分布を示す。

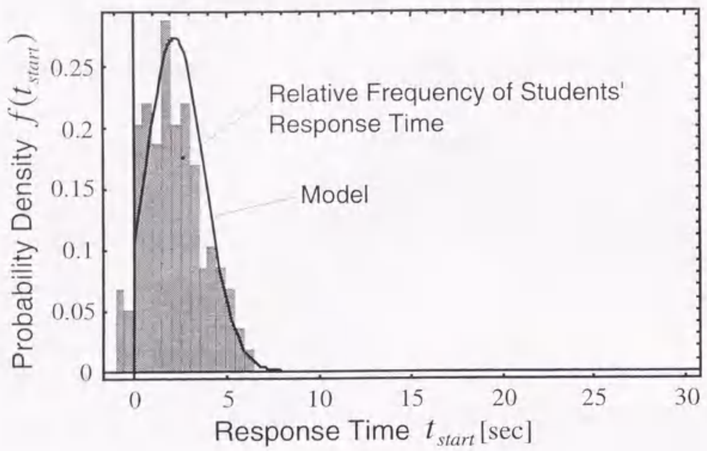


図 3.3: ユーザ挙動モデル

Fig. 3.3 User behavior models.

■ アプリケーション

クライアント WS がサーバ WS に対して FTP および NFS (version 2) [Sun89] によるファイル転送要求を行う場合を想定する。

FTP は TCP 上で動作するので，障害処理は TCP によって行われる。モデルでの動作の流れは以下のようになる。クライアント WS 上のクライアントプログラムがユーザ操作に従ってファイル転送要求を送信し，それを受信したサーバ WS 上のサーバプログラムが連続して要求されたファイルを TCP を介して送信する。この FTP のモデルは第 2 章で用いたものと同じであり，TCP の二重通信路の確立に伴う処理は省略されている。

一方，NFS は TCP 上ではなく UDP 上で動作する。UDP は信頼性の保証を行わないトランスポート層プロトコルであるので，障害処理は NFS 自身によりタイムアウトアンドリトライ方式で行われる。NFS ではクライアントからの 1 回のメッセージで要求できる転送データの最大サイズは 8 Kbytes と制限されている。従って，クライアントは 8 Kbytes を受信するたびに新しい転送要求を送信することになる。要求に対する応答がタイムアウト以内に得られないと，NFS クライアントは再度要求メッセージをサーバに送信する。NFS のモデルでは，再送タイムアウト等の NFS のパラメータには Sun OS 4.1.3 のデフォルト値を使用した。また，NFS では 8 Kbytes までのデータ送信要求を一度に

IP に要求するので，IP ではデータグラムのセグメント化が行われる。この処理のために 2 章で述べた IP のモデルに拡張を施している。

クライアントにおける転送要求発生時刻の分布は先に示した学生の反応時間のモデルに従う。転送ファイルサイズは，中規模のアプリケーションをファイルサーバから読みとることを想定し，すべて 1 Mbytes とした。

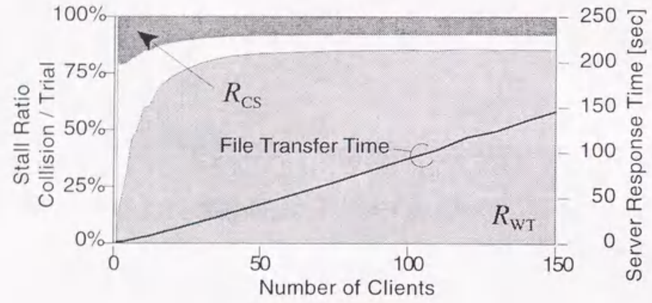
3.3.2　シミュレーション結果

■ ネットワーク全体の特性

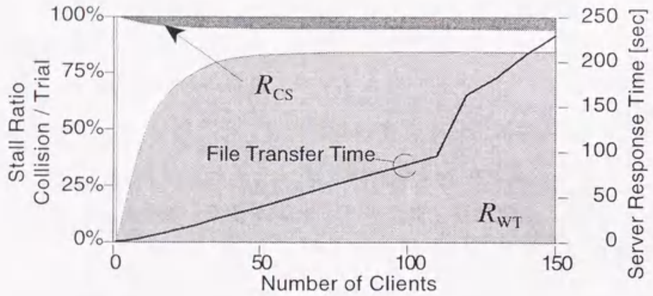
図 3.4 および表 3.7，3.8 に FTP および NFS によるファイル転送時における，全クライアントでのファイル転送に対する平均サーバ応答時間，およびサーバを含むすべての端末におけるストール率を示す。

FTP, NFS どちらの場合でもクライアント数の増加に伴って R_{WT} が増加し，クライアント数 50 台以上で 80 % 程度で変化がなくなる。一方 R_{CS} は FTP の場合に少いクライアント数で大きな値を示しているが，NFS の場合には逆に R_{CS} は小さい値を示す。FTP はトランスポート層に TCP を用いているため，サーバからデータを受信すると確認応答を送信する必要がある。このため，クライアント数が少ない場合においても，サーバとクライアントとの間で同時に送信を行うことがあり，その結果 R_{CS} が大きな値を示す。一方，NFS ではタイムアウトが起きない限りクライアントはサーバに対して送信要求メッセージを送信しない。このため，クライアント数が少ない場合においては，サーバとクライアントが同時に送信を行うことがない。

しかしながら，ストール率からは，上記のようなプロトコルの違いによるトラヒック特性の差を知ることは出来るが，平均サーバ応答時間のようなアプリケーションレベルでの性能に及ぼす影響を読みとることは出来ない。



(a) FTP / Network



(b) NFS / Network

図 3.4: アプリケーション利用時の平均サーバ応答時間とネットワーク平均ストー ル率

Fig. 3.4 The mean server response time and the mean network stall ratio.

表 3.7: アプリケーション利用時の平均サーバ応答時間とネットワーク平均ストール率 (FTP)

N	平均サーバ応答時間 [sec]	CS ストール率 R_{CS}	WT ストール率 R_{WT}	トータルストール率 R_T
1	1.296	0.202	0.055	0.256
2	1.678	0.214	0.164	0.378
4	2.362	0.206	0.283	0.489
6	3.900	0.186	0.455	0.641
8	5.183	0.174	0.503	0.677
10	6.864	0.153	0.596	0.749
12	7.945	0.151	0.604	0.754
14	9.807	0.138	0.659	0.796
16	11.489	0.128	0.696	0.823
18	13.436	0.122	0.715	0.837
20	14.976	0.116	0.735	0.851
25	19.462	0.107	0.768	0.875
30	24.022	0.099	0.794	0.893
35	28.541	0.094	0.811	0.904
40	33.777	0.089	0.825	0.914
50	42.895	0.085	0.836	0.922
75	66.919	0.081	0.851	0.932
100	93.304	0.080	0.854	0.934
110	102.847	0.081	0.853	0.933
120	116.922	0.079	0.856	0.936
130	123.815	0.080	0.856	0.935
140	135.379	0.080	0.857	0.936
150	145.759	0.080	0.855	0.93

表 3.8: アプリケーション利用時の平均サーバ応答時間とネットワーク平均ストール率 (NFS)

N	平均サーバ応答時間 [sec]	CS ストール率 R_{CS}	WT ストール率 R_{WT}	トータルストール率 R_T
1	1.296	0.202	0.055	0.256
2	1.678	0.214	0.164	0.378
4	2.362	0.206	0.283	0.489
6	3.900	0.186	0.455	0.641
8	5.183	0.174	0.503	0.677
10	6.864	0.153	0.596	0.749
12	7.945	0.151	0.604	0.754
14	9.807	0.138	0.659	0.796
16	11.489	0.128	0.696	0.823
18	13.436	0.122	0.715	0.837
20	14.976	0.116	0.735	0.851
25	19.462	0.107	0.768	0.875
30	24.022	0.099	0.794	0.893
35	28.541	0.094	0.811	0.904
40	33.777	0.089	0.825	0.914
50	42.895	0.085	0.836	0.922
75	66.919	0.081	0.851	0.932
100	93.304	0.080	0.854	0.934
110	102.847	0.081	0.853	0.933
120	116.922	0.079	0.856	0.936
130	123.815	0.080	0.856	0.935
140	135.379	0.080	0.857	0.936
150	145.759	0.080	0.855	0.93

■ クライアント・サーバの送信特性

クライアントサーバシステムを考えた場合，サーバの送信するパケットは一般的に長く，その数は多い。またクライアントの送信するパケットは要求メッセージや TCP の確認応答がほとんどであり，パケット長が短く数も多くはない。さらに，クライアントサーバシステムではクライアントとサーバ間のメッセージ依存性があるので，両者における送信要求の発生時刻の分布は異なり，パケット送信時のネットワーク状態も異なるものになる。このため，アプリケーションを含んだ性能評価を行う上では，ネットワーク全体の単純な統計値のみによる評価では不十分である。

図 3.5 および表 3.9 ～3.12 にクライアントとサーバそれぞれの平均ストール率と衝突率および平均パケット送信遅延を示す。明らかにサーバの方がストール率が低く，高い送信効率を保っていることがわかる。サーバがほとんどのパケットを最大パケット長 (1518 byte) で送信しているため相対的にバックオフによる待ち時間および伝送路の空きを待っていた時間が短くなることが，高い送信効率の一因である。しかしそれだけでなく実際，サーバの衝突率はクライアントに比べて低い。このことから，サーバ送信時における衝突は少なく，サーバの送信効率の極端な低下は起きないが，クライアントは複数のクライアントが同時に送信しようとすることによって送信効率が悪くなっていることがわかる。

この傾向は NFS の場合に顕著である。TCP を用いる FTP の場合，再送信はデータ送信側であるサーバによって開始されるのに対し，NFS ではクライアントにより要求の再送信が行なわれる。従って，再送タイムアウトが発生した場合 NFS の方がより多くの端末からトラヒックが発生することになるのである。

しかし，ストール率は上記のような上位プロトコルの特性が LAN 挙動に及ぼす影響を示しはするものの，ストール率のみからアプリケーションの性能を判断するのは困難である。実際，図 3.4(b) で見られる NFS におけるファイル転送に対するサーバ応答時間の増加は，アプリケーションレベルの応答時間やリクエスト再送回数の評価によって初めて原因が特定可能である。図 3.6 および表 3.13 は NFS の各要求メッセージに対するサーバの平均応答時間（ファイル全体の送信要求に対する応答時間ではない）とクライアントでの NFS 要求の平均再送回数を示したものである。N の増加に伴い各 NFS 要求に対する応答時間が増加し，再送タイムアウトを越えることで再送回数が急増している。この再送回数の急増したときの N とファイル全体の転送要求に対する応答時間が急増する N が一致することは明らかである。

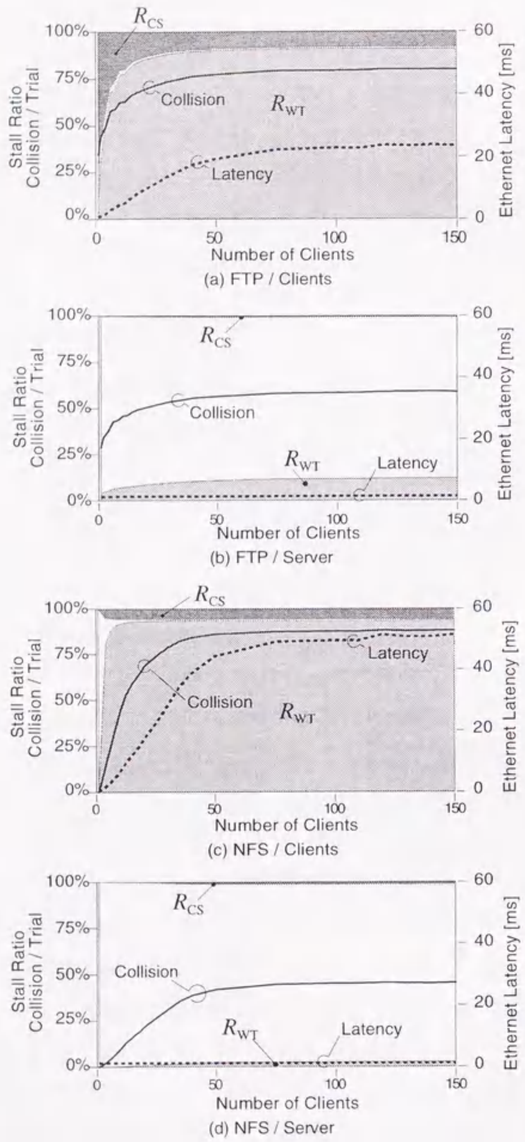


図 3.5: クライアントとサーバのストール率と衝突率

Fig. 3.5 The mean stall ratio and the Ethernet collision rate at clients and the server.

表 3.9: FTP クライアントのストール率と衝突率

N	送信遅延 [msec]	衝突率	CS ストール率 R_{CS}	WT ストール率 R_{WT}	トータルストール率 R_T
1	0.6	0.347	0.762	0.100	0.862
2	1.0	0.438	0.589	0.318	0.907
4	1.6	0.496	0.434	0.506	0.940
6	3.0	0.577	0.294	0.672	0.966
8	3.5	0.591	0.260	0.710	0.971
10	4.8	0.626	0.205	0.774	0.979
12	5.1	0.629	0.201	0.778	0.979
14	6.4	0.655	0.173	0.810	0.983
16	7.5	0.672	0.155	0.831	0.986
18	8.4	0.682	0.146	0.841	0.987
20	9.3	0.693	0.136	0.852	0.988
25	11.3	0.715	0.122	0.868	0.990
30	13.5	0.734	0.110	0.881	0.991
35	15.3	0.748	0.103	0.889	0.992
40	17.3	0.762	0.097	0.895	0.993
50	19.2	0.773	0.092	0.901	0.993
75	22.2	0.791	0.087	0.907	0.994
100	22.8	0.793	0.085	0.909	0.994
110	22.7	0.793	0.086	0.908	0.994
120	23.7	0.799	0.085	0.910	0.994
130	23.5	0.798	0.085	0.909	0.994
140	23.8	0.800	0.085	0.910	0.994
150	23.5	0.798	0.085	0.909	0.99

表 3.10: FTP サーバ のストール率と衝突率

N	送信遅延 [msec]	衝突率	**CS** ストール率 R_{CS}	**WT** ストール率 R_{WT}	トータルストール率 R_{T}
1	1.2	0.289	0.002	0.038	0.040
2	1.2	0.342	0.003	0.048	0.050
4	1.2	0.376	0.003	0.054	0.057
6	1.3	0.429	0.004	0.065	0.069
8	1.3	0.436	0.004	0.067	0.071
10	1.3	0.459	0.004	0.073	0.077
12	1.3	0.462	0.004	0.074	0.078
14	1.3	0.480	0.005	0.079	0.083
16	1.3	0.491	0.005	0.082	0.087
18	1.3	0.498	0.005	0.084	0.089
20	1.3	0.505	0.005	0.086	0.092
25	1.3	0.521	0.006	0.092	0.097
30	1.3	0.535	0.006	0.097	0.103
35	1.3	0.545	0.007	0.101	0.107
40	1.3	0.557	0.007	0.106	0.113
50	1.3	0.565	0.008	0.109	0.117
75	1.4	0.581	0.008	0.117	0.125
100	1.4	0.583	0.009	0.118	0.126
110	1.4	0.585	0.009	0.119	0.127
120	1.4	0.588	0.009	0.120	0.128
130	1.4	0.589	0.009	0.121	0.129
140	1.4	0.591	0.009	0.121	0.130
150	1.4	0.589	0.009	0.120	0.129

表 3.11: NFS クライアントのストール率と衝突率

N	送信遅延 [msec]	衝突率	**CS** ストール率 R_{CS}	**WT** ストール率 R_{WT}	トータルストール率 R_{T}
1	0.2	0.000	0.000	0.000	0.000
2	0.6	0.046	0.020	0.296	0.317
4	1.7	0.150	0.051	0.811	0.862
6	3.2	0.253	0.054	0.881	0.935
8	4.8	0.352	0.057	0.908	0.965
10	6.8	0.443	0.057	0.918	0.975
12	9.3	0.522	0.058	0.924	0.982
14	11.2	0.569	0.057	0.928	0.985
16	13.1	0.611	0.058	0.929	0.987
18	14.9	0.643	0.058	0.931	0.988
20	17.4	0.682	0.058	0.932	0.990
25	22.5	0.739	0.058	0.934	0.992
30	28.1	0.784	0.058	0.935	0.993
35	34.2	0.819	0.058	0.936	0.994
40	38.6	0.839	0.058	0.936	0.994
50	44.3	0.861	0.059	0.936	0.995
75	49.0	0.875	0.059	0.937	0.995
100	49.5	0.877	0.059	0.936	0.995
110	49.4	0.877	0.059	0.937	0.995
120	51.5	0.881	0.061	0.935	0.995
130	50.7	0.880	0.061	0.935	0.995
140	50.8	0.880	0.061	0.934	0.995
150	51.5	0.882	0.061	0.934	0.99

表 3.12: NFS サーバのストール率と衝突率

N	送信遅延 [msec]	衝突率	**CS** ストール率 R_{CS}	**WT** ストール率 R_{WT}	トータルストール率 R_{T}
1	1.2	0.000	0.000	0.000	0.000
2	1.2	0.008	0.000	0.000	0.000
4	1.2	0.026	0.001	0.001	0.001
6	1.2	0.047	0.001	0.001	0.002
8	1.2	0.072	0.001	0.002	0.003
10	1.2	0.100	0.002	0.003	0.005
12	1.2	0.131	0.003	0.004	0.007
14	1.2	0.151	0.003	0.005	0.008
16	1.2	0.174	0.004	0.006	0.010
18	1.2	0.193	0.004	0.007	0.011
20	1.2	0.220	0.005	0.008	0.013
25	1.2	0.266	0.006	0.011	0.017
30	1.2	0.313	0.008	0.014	0.022
35	1.2	0.356	0.010	0.017	0.027
40	1.2	0.386	0.011	0.020	0.031
50	1.2	0.418	0.013	0.023	0.036
75	1.2	0.446	0.014	0.026	0.040
100	1.2	0.451	0.015	0.026	0.041
110	1.2	0.451	0.015	0.027	0.041
120	1.2	0.454	0.015	0.028	0.042
130	1.2	0.452	0.015	0.027	0.042
140	1.2	0.451	0.015	0.027	0.042
150	1.2	0.454	0.015	0.028	0.042

表 3.13: NFS リクエスト再送率と応答時間

N	**NFS** リクエスト再送率	**NFS** 応答時間 [sec]
1	0.000	0.007
2	0.002	0.009
4	0.007	0.013
6	0.014	0.020
8	0.023	0.028
10	0.033	0.038
12	0.046	0.051
14	0.056	0.061
16	0.066	0.071
18	0.076	0.082
20	0.091	0.097
25	0.120	0.127
30	0.153	0.161
35	0.190	0.200
40	0.217	0.235
50	0.255	0.303
75	0.289	0.493
100	0.297	0.672
110	0.301	0.748
120	0.980	1.276
130	1.007	1.425
140	1.126	1.625
150	1.199	1.79

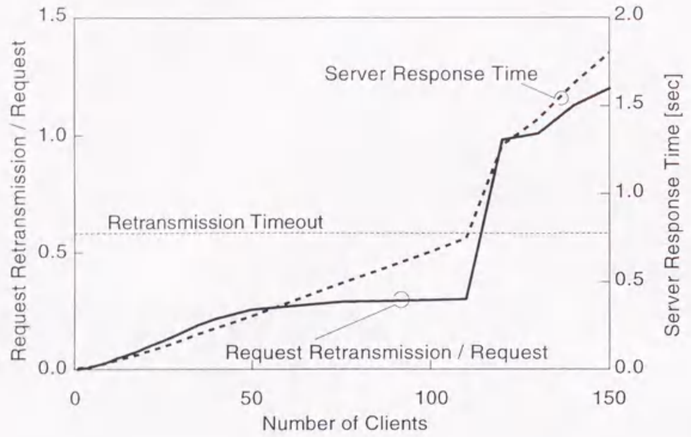


図 3.6: NFS の平均応答時間と平均要求再送回数

Fig. 3.6 The mean NFS response time and the mean request retransmission count par request.

3.3.3　ユーザストール率

ストール率の定義では MAC 層の性能に重点をおいている。しかしながら，ユーザアプリケーションレベルでは MAC 層におけるエンドツゥエンド遅延よりもアプリケーションの動作に直接影響を与える上位層におけるエンドツゥエンド遅延が問題になる。そこで，ユーザ側の視点に立ったストール率として，ストール率の定義（**式 3.1**，**式 3.2**）における分母 T_{XMT} に MAC 層における待ち行列遅延を含めて，以下のようにユーザストール率を定義する。

平均バッファリングストール率

$$\tilde{R}_{\mathrm{Q}} = \frac{\sum_{i=1}^{N} T_{\mathrm{Q}}^{(i)}}{\sum_{i=1}^{N} (T_{\mathrm{XMT}}^{(i)} + T_{\mathrm{Q}}^{(i)})} \tag{3.5}$$

平均 CS ストール率

$$\tilde{R}_{\mathrm{WT}} = \frac{\sum_{i=1}^{N} T_{\mathrm{WT}}^{(i)}}{\sum_{i=1}^{N} (T_{\mathrm{XMT}}^{(i)} + T_{\mathrm{Q}}^{(i)})} \tag{3.6}$$

平均 WT ストール率

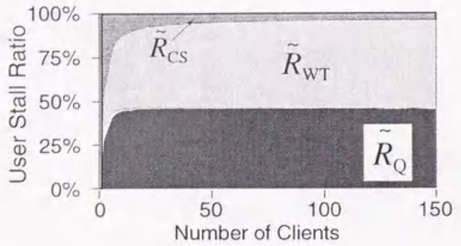
$$\tilde{R}_{\mathrm{CS}} = \frac{\sum_{i=1}^{N} T_{\mathrm{CS}}^{(i)}}{\sum_{i=1}^{N} (T_{\mathrm{XMT}}^{(i)} + T_{\mathrm{Q}}^{(i)})} \tag{3.7}$$

平均トータルストール率

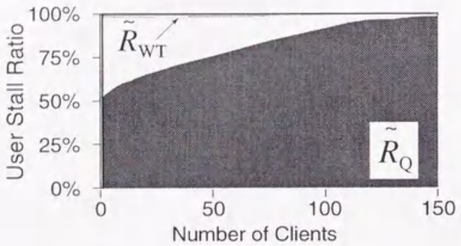
$$\tilde{R}_{\mathrm{T}} = \tilde{R}_{\mathrm{Q}} + \tilde{R}_{\mathrm{CS}} + \tilde{R}_{\mathrm{WT}} \tag{3.8}$$

上の定義で平均バッファリングストール率 $\tilde{R}_{\mathrm{Q}}$ は MAC 層における待ち行列遅延の割合を示す。この値は各端末において Ethernet の処理率に対して過剰な入力負荷がかかっている場合に大きくなる。

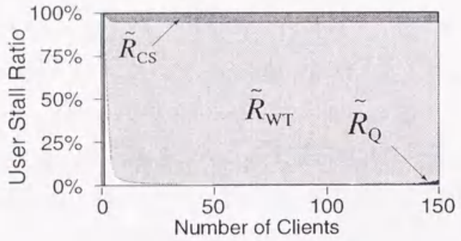
図 3.7 および表 3.14 ～3.17 にサーバ，クライアントのユーザストール率を示す。FTP, NFS ともにサーバにおける $\tilde{R}_{\mathrm{Q}}$ が高い値を示している。これはクライアントが TCP の確認応答パケットおよび NFS のリクエストメッセージしか送信しないのに対し，サーバが大量のデータを全クライアントに送信するためである。得に (d) の NFS サーバの場合には，クライアント数 100 台程度で $\tilde{R}_{\mathrm{Q}}$ がほぼ 100 % に達している。この結果としてサーバ応答遅延が増大し，クライアントにおける再送要求の発生，サーバ応答時間の急増に至ると推測できる。



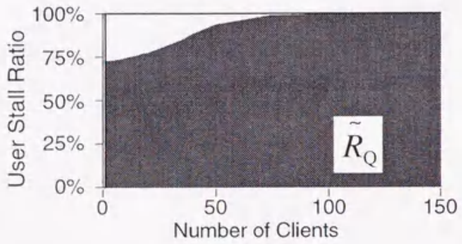
(a) FTP / Clients



(b) FTP/ Server



(c) NFS / Clients



(d) NFS / Server

図 3.7: ユーザストール率

Fig. 3.7 The user stall ratio.

表 3.14: FTP クライアントのユーザストール率

N	バッファリングストール率 $\tilde{R}_{Q}$	WT ストール率 $\tilde{R}_{WT}$	CS ストール率 $\tilde{R}_{CS}$	トータルストール率 $\tilde{R}_{T}$
1	0.127	0.048	0.700	0.875
2	0.275	0.193	0.459	0.926
4	0.348	0.285	0.322	0.955
6	0.412	0.368	0.197	0.976
8	0.431	0.398	0.154	0.983
10	0.444	0.414	0.128	0.986
12	0.448	0.426	0.115	0.988
14	0.446	0.439	0.105	0.989
16	0.454	0.445	0.093	0.991
18	0.456	0.450	0.086	0.992
20	0.456	0.457	0.080	0.993
25	0.458	0.464	0.071	0.994
30	0.458	0.473	0.063	0.995
35	0.458	0.478	0.059	0.995
40	0.457	0.482	0.056	0.995
50	0.456	0.489	0.052	0.996
75	0.456	0.496	0.045	0.997
100	0.453	0.502	0.042	0.997
110	0.455	0.502	0.040	0.997
120	0.455	0.503	0.040	0.997
130	0.450	0.507	0.040	0.997
140	0.453	0.505	0.039	0.997
150	0.447	0.510	0.040	0.997

表 3.15: FTP サーバ のユーザストール率

N	バッファリングストール率 $\tilde{R}_{\mathrm{Q}}$	**WT** ストール率 $\tilde{R}_{\mathrm{WT}}$	**CS** ストール率 $\tilde{R}_{\mathrm{CS}}$	トータルストール率 $\tilde{R}_{\mathrm{T}}$
1	0.508	0.010	0.001	0.520
2	0.535	0.012	0.002	0.548
4	0.551	0.013	0.002	0.566
6	0.573	0.014	0.002	0.589
8	0.587	0.014	0.002	0.604
10	0.602	0.015	0.002	0.619
12	0.610	0.015	0.002	0.628
14	0.617	0.015	0.002	0.635
16	0.631	0.015	0.003	0.649
18	0.639	0.015	0.003	0.657
20	0.650	0.016	0.003	0.668
25	0.669	0.015	0.003	0.687
30	0.690	0.015	0.003	0.708
35	0.710	0.015	0.002	0.727
40	0.726	0.014	0.002	0.742
50	0.758	0.013	0.002	0.773
75	0.845	0.009	0.002	0.856
100	0.916	0.005	0.001	0.923
110	0.947	0.004	0.001	0.951
120	0.966	0.002	0.000	0.969
130	0.966	0.002	0.000	0.969
140	0.980	0.001	0.000	0.981
150	0.984	0.001	0.000	0.985

表 3.16: NFS クライアント のユーザストール率

N	バッファリングストール率 $\tilde{R}_{\mathrm{Q}}$	**WT** ストール率 $\tilde{R}_{\mathrm{WT}}$	**CS** ストール率 $\tilde{R}_{\mathrm{CS}}$	トータルストール率 $\tilde{R}_{\mathrm{T}}$
1	0.000	0.000	0.000	0.000
2	0.000	0.342	0.021	0.362
4	0.000	0.800	0.046	0.846
6	0.000	0.894	0.051	0.945
8	0.000	0.911	0.052	0.963
10	0.000	0.920	0.053	0.973
12	0.000	0.926	0.053	0.979
14	0.000	0.928	0.054	0.982
16	0.000	0.931	0.054	0.985
18	0.000	0.934	0.054	0.987
20	0.000	0.934	0.054	0.988
25	0.000	0.937	0.054	0.991
30	0.000	0.938	0.055	0.993
35	0.000	0.938	0.055	0.994
40	0.000	0.939	0.056	0.994
50	0.000	0.939	0.056	0.995
75	0.000	0.939	0.057	0.996
100	0.000	0.939	0.057	0.996
110	0.000	0.938	0.057	0.996
120	0.006	0.933	0.057	0.996
130	0.006	0.931	0.059	0.996
140	0.004	0.933	0.059	0.996
150	0.022	0.916	0.058	0.996

表 3.17: NFS サーバのユーザストール率

N	バッファリングストール率 $\tilde{R}_{\mathrm{Q}}$	**WT** ストール率 $\tilde{R}_{\mathrm{WT}}$	**CS** ストール率 $\tilde{R}_{\mathrm{CS}}$	トータルストール率 $\tilde{R}_{\mathrm{T}}$
1	0.726	0.000	0.000	0.726
2	0.727	0.000	0.000	0.727
4	0.730	0.000	0.000	0.730
6	0.733	0.000	0.000	0.733
8	0.736	0.000	0.000	0.737
10	0.743	0.001	0.000	0.744
12	0.749	0.001	0.000	0.750
14	0.754	0.001	0.000	0.756
16	0.760	0.001	0.000	0.762
18	0.771	0.001	0.000	0.773
20	0.773	0.002	0.000	0.775
25	0.797	0.002	0.001	0.800
30	0.823	0.002	0.001	0.826
35	0.848	0.002	0.001	0.851
40	0.883	0.002	0.001	0.886
50	0.936	0.002	0.001	0.938
75	0.991	0.000	0.000	0.991
100	0.996	0.000	0.000	0.996
110	0.997	0.000	0.000	0.997
120	0.997	0.000	0.000	0.997
130	0.999	0.000	0.000	0.999
140	0.999	0.000	0.000	0.999
150	0.999	0.000	0.000	0.999

3.4　むすび

CSMA/CD LAN の性能評価のための新しい指標としてストール率を提案した。ストール率による評価では，可視化によって端末の送信効率とその低下の原因の直観的理解が容易である。ストール率による現実的な高負荷 LAN 環境の評価例として，集合情報処理教育環境におけるクライアントサーバシステムにおけるネットワークアプリケーション利用時のトラヒック特性をストール率を用いて評価し，ネットワーク全体のストール率と，役割の違う端末のストール率を比較することによってアプリケーションの動作が LAN 性能に及ぼす影響が明らかになることを示した。また，特にアプリケーション動作時の性能評価を行う場合には，Ethernet の送信待ち行列遅延を考慮したユーザストール率が有効であることがわかった。

ストール率による評価は，CSMA/CD LAN の挙動の直観的理解を助けるものの，それのみによってシステムの定量的評価を行うには不向きである。しかし，ストール率の特徴である視覚的·直観的理解のしやすさは従来からの性能指標の理解を促進し，設計システムの早期問題発見に有用であると考える。今後の課題として，Capture Effect [WSF94][Mol94] のようなネットワークの動的な挙動を視覚的に評価できる指標の検討がある。

第4章　教育用 LAN の性能評価

本章では，教育用 LAN に適したネットワーク構成の検討を第 2 章で提案したモデルを用いたシミュレーション評価によって行う。大学等の集合情報処理教育施設においては，NFS 等の分散ファイルシステムを用い，数千人以上のユーザデータや共有データを管理しているところが多い。このような環境においては，授業利用時のユーザによる一斉操作によるネットワーク，サーバへの負荷集中への対応がシステム設計において重要である。本研究では Ethernet を基本とした様々なネットワーク構成例に対して，現実のユーザの挙動の測定値に基づいたモデルを用い，ユーザ操作による負荷集中を想定したシミュレーションを行った。負荷集中時の性能とコスト面での考察を行い，教育用システムに適したネットワーク構成を示す。

4.1　はじめに

大学等の情報処理教育施設などのワークステーションを主体とした大規模集合型情報処理教育システムの多くでは，ユーザの個人データや共有データを NFS などの分散ファイルシステムによって管理している。このような施設における LAN （教育用 LAN）では，授業利用時におけるネットワークへの負荷集中がそのネットワーク設計において問題になる。

分散ファイルシステムを用いた場合，ユーザのホームディレクトリ，アプリケーションプログラムや演習資料などの共有データはファイルサーバ (FS) 上に保存され，クライアントワークステーション (CWS) のユーザがそれらを利用する時には FS・CWS 間での

ファイル読み込み，書き込み操作が行われる。これらの操作の多くは教官の指示に従った学生が CWS を操作した結果として行われることが多い。いわば「よーいどん」で一斉に操作を行う状況である。最悪の場合全ユーザが同時に操作を行うことになり，この場合 LAN, FS への負荷は多くの端末から大量に発生されることになる。この結果ユーザの操作に対する応答時間が長くなる。

十分に速いユーザ応答速度を保ちかつ低コストでネットワークを設計するためには，上記のような過渡的な負荷集中の発生についての考慮が重要である。ユーザの一斉操作によって発生する LAN への時間的負荷集中を避けるため，様々なネットワークトポロジが教育用 LAN 向けに提案されてきた [岡田 96] [中山 94]。Ethernet を用いたクライアント・サーバ型システムの最も単純な構成は，FS と CWS を同一の Ethernet コリジョンドメインに接続する方式であるが，この場合，コリジョンドメイン内の端末数が大きい場合には Ethernet 上でのパケットの衝突が頻発することが予想される。岡田ら [岡田 96] は数十台の CWS ごとにコリジョンドメインを分け，各ドメインをそれぞれ複数のネットワークインタフェースを持つ FS に接続する方式を提案している。この方式を基本にした構成の場合，各コリジョンドメインに接続する CWS の台数，および接続する Ethernet の方式の選択がその性能を左右する要因となろう。

本章ではスイッチング型，媒体共有型およびギガビット Ethernet による 6 種の構成方式における性能比較を行い教育環境に適切な LAN 構成の指針を提案する [IKO98a] [IKO98b] [小島 98]。以下，2 節で教育用 LAN における負荷集中時の性能評価を行うためのモデリング方および負荷集中モデルについて述べ，3 節ではシミュレーション結果とそれにより得られた教育用 LAN の設計指針についてまとめる。

4.2　教育用 LAN の負荷集中のモデル化

4.2.1　シミュレーションの場面設定

シミュレーションを行う条件として以下のような状況を想定した。

- 1 台の FS と $N(=1 \sim 150)$ の CWS から構成される Ethernet LAN。
- NFS version 2 [Sun89] および FTP による FS から CWS への 1 MB のファイル転送。
- 教官の合図に従ったユーザの一斉操作により，CWS でファイル転送要求が発生。

4.2.2　シミュレーションモデル

シミュレーションには第 2 章で述べたイベント駆動型ネットワークシミュレータ [石原 95] [IO97a] および小島らによるそのスイッチング LAN への拡張 [小島 98] を使用した。本シミュレータでは NFS および TCP/IP の再送制御およびフロー制御，Ethernet の CSMA/CD アルゴリズムを忠実にモデル化している。各端末のモデルは各階層のプロトコルモデル，アプリケーションモデルおよびユーザ挙動モデルから構成される。

シミュレーションは以下のような手順で行われる。まず，ユーザ挙動モデルが定義する分布に従い各 CWS のユーザの操作時刻が乱数により決定され，イベントが発生する。このイベントを受け取った CWS の端末モデル上のアプリケーションモデルでメッセージの生成が行われ，各階層のプロトコルモデルの動作を介して FS へのメッセージ送出が行れる。以後のメッセージ生成・送信はアプリケーションモデルおよび各階層のプロトコルモデルの動作に従って行われる。

なお，本シミュレーションでは Ethernet 送信部の待ち行列長に制限を設けていない。また，ディスクアクセスおよび CPU 動作による遅延は無視している。

4.2.3　ユーザ挙動モデル

ユーザ挙動モデルは CWS におけるユーザの操作が行われる時刻 t_{start} の分布を定義する。筆者らは集合型情報処理教育施設の現場で教官の指示に対して学生がマウスのダブルクリックで応答するまでの時間を測定し，この結果に基づいた 3 つのユーザ挙動モデルを作成した [石原 96]。

ユーザ挙動モデルでは以下の分布に従った疑似乱数で決定する。

$$f(t_{start}) = \begin{cases} \dfrac{\exp\left(\dfrac{-(t_{start}-\mu)^2}{2\sigma^2}\right)}{\displaystyle\int_0^\infty \exp\left(\frac{-(\xi-\mu)^2}{2\sigma^2}\right) d\xi} & (t_{start} \geq 0) \\ 0 & (t_{start} < 0) \end{cases} \tag{4.1}$$

上式は，正規分布において $t_{start} < 0$ のとき $f(t_{start}) = 0$ とした分布である。パラメータ μ，σ は図 4.1 は学生の応答時間の分布に従って決定した。各モデルはそれぞれ以下の条件を想定したものである。

モデル M ($\mu = 2.2, \sigma = 1.6$ [sec]) ユーザの応答がマウス操作で行われることを仮定。μ,

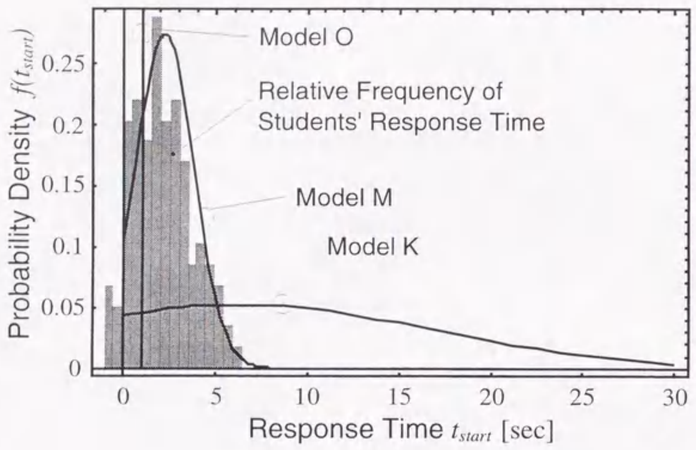


図 4.1: ユーザ挙動モデル

Fig. 4.1 User behavior models.

σ は実測値の分布に等しい。

モデル K ($\mu = 6.3, \sigma = 10.6$ [sec]) ユーザの応答がキーボード操作で行われることを仮定。モデル K では M よりも広い分布を持つ。

モデル O ($\mu = 1.0, \sigma = 0.0$ [sec]) ユーザ操作が完全に同時に行われたと仮定した理想モデル

4.2.4　アプリケーションモデル

FS, CWS 間のファイル転送を行うアプリケーションとして，FTP と NFS の二種類のシミュレーションを行った。これらのモデルは 3.3.1 で説明したものと同じである。

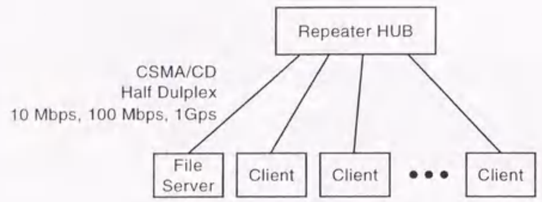
4.2.5　接続方式

教育用 LAN の適切な設計指針を得るため，CSMA/CD 方式の媒体共有型 Ethernet による 3 種類，スイッチング型の Ethernet による 3 種類，計 6 種の接続方式（図 4.2，表

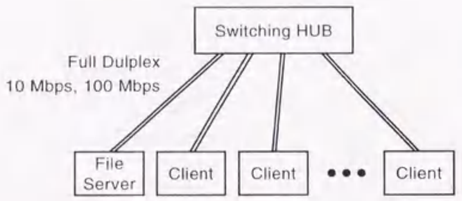
表 4.1: 検討対象とした接続方式

	方式	サーバ	クライアント
Shared-10M	CSMA/CD	10 Mbps	10 Mbps
Shared-100M	CSMA/CD	100 Mbps	100 Mbps
Shared-1G	CSMA/CD	1 Gbps	1 Gbps
Switched-10M	スイッチ（全二重）	10 Mbps	10 Mbps
Switched-100M	スイッチ（全二重）	100 Mbps	100 Mbps
BigPipe-10M/100M	スイッチ（全二重）	100 Mbps	10 Mbps

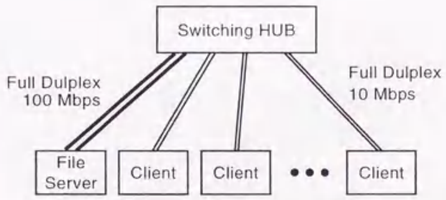
4.1）を想定した。全ての方式で FS と全ての CWS は 1 台のリピータハブまたはスイッチングハブに接続されているものとし，Ethernet の接続方式（媒体共有型・スイッチング型）と伝送速度を変化させた。Shared-1G を除いてハブと端末間のケーブル長は 10BASE T および 100BASE TX で定める最大長である 100 m とした。スイッチングハブの方式はストア・アンド・フォワード形式であるとし，スイッチング時間は 0 とした。



(a) Shared-10M, Shared-100M, Shared-1G



(b) Switched-10M, Switched-100M



(c) BigPipe-10M/100M

図 4.2: 検討対象とした接続方式

Fig. 4.2 Network topologies for experiments.

4.3 シミュレーション結果

4.3.1 応答時間

ここでは，CWS から最初のリクエストが送信されてから，CWS が要求したファイルの全体を受信するまでの時間を応答時間と定義する。図 4.3〜4.8 に CWS 数 N と全 CWS での平均応答時間 T の変化を示す。

媒体共有型　Shared-10M の NFS における T は，ユーザ挙動モデル M の場合，N にほぼ比例して増加している。モデル O との違いは小さい。モデル K の場合，$N > 20$ では他のモデルと同様の増加率を示すが，$N < 20$ では T の増加率は非常に小さい。Shared-100M のモデル M では，$N = 30$ を境として同様の傾向が見られる。これは N が小さい時には要求発生時刻のばらつきのために，複数台の CWS による競合がほとんど起きないためである。この領域が Ethernet の使用環境における理想状態といえよう。

また，Shared-10M では，N が 100 を越えると，すべてのモデルで T は急激に増大する。これは NFS クライアントからの要求の再送タイムアウト発生が原因である。NFS の各要求に対する平均サーバ応答時間は $N = 100$ 付近で再送タイムアウトを越えており，このため CWS からの要求の再送が頻発していることが確認されている [石原 97]。この状態ではパケットの消失がないにも関らず，要求の再送が行われることになる。要求の再送が発生すると，CWS からの再送要求メッセージ数，FS からの重複した応答数により，ネットワーク負荷は更に増大する。NFS のパラメータの調節，CWS 台数の削減によってこのような状況の発生を避けるべきである。

一方 FTP の場合，T の急激な増加はみられない。FTP は TCP 上で動作するが，TCP におけるデータの再送は送信側で動的に設定される再送タイムアウトによって制御される。このため，NFS で起きたように，パケットの消失が実際には起きていない場合に再送が行われることが少ない。また，再送が送信側である FS から行われるので，台数の多い CWS からの再送要求によるトラヒックの増加が少ない。このため，NFS の場合のような急激な T の増大は起きなかったと考えられる。

媒体共有型 ギガビット Ethernet　Shared-1G では，モデル M，K ともにアクセス集中による影響は全く見られなかった。図 4.9 に NFS における Shared-100M の場合の $T/10$ との比較を示す。Shared-100M の $T/10$ のモデル M では $N = 30$ 付近で増加率の上昇があるが，Shared-1G の場合にはそれがない。

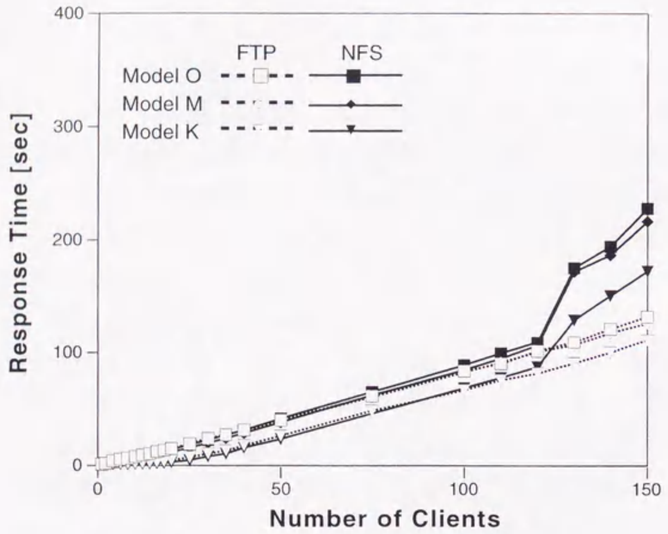


図 4.3: サーバ応答時間 (Shared-10M)

Fig. 4.3 Server response time (Shared-10M).

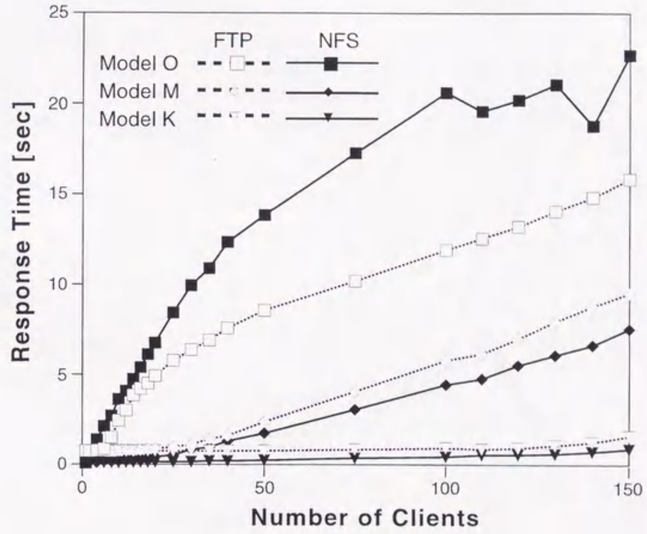


図 4.4: サーバ応答時間 (Shared-100M)

Fig. 4.4 Server response time (Shared-100M).

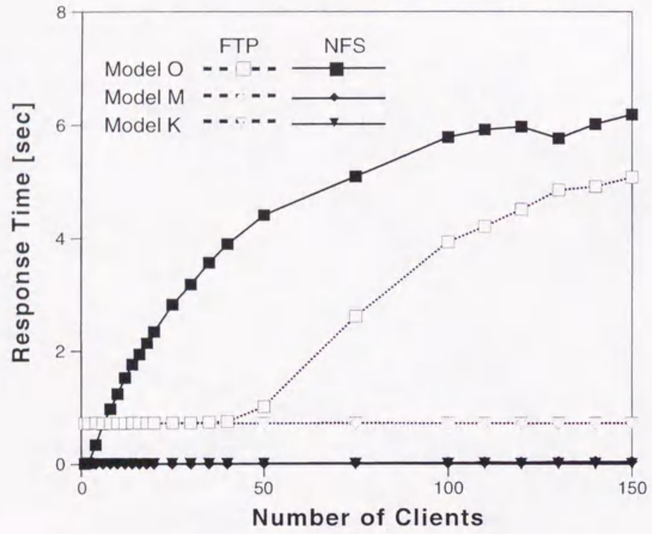


図 4.5: サーバ応答時間 (Shared-1G)

Fig. 4.5 Server response time (Shared-1G).

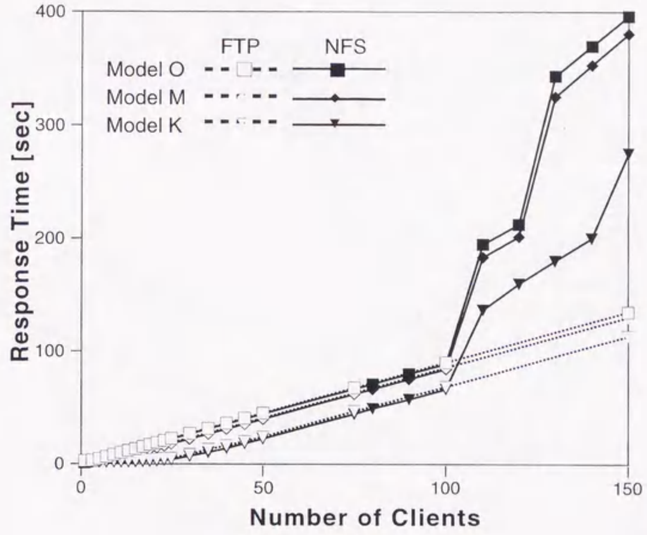


図 4.6: サーバ応答時間 (Switched-10M)

Fig. 4.6 Server response time (Switched-10M).

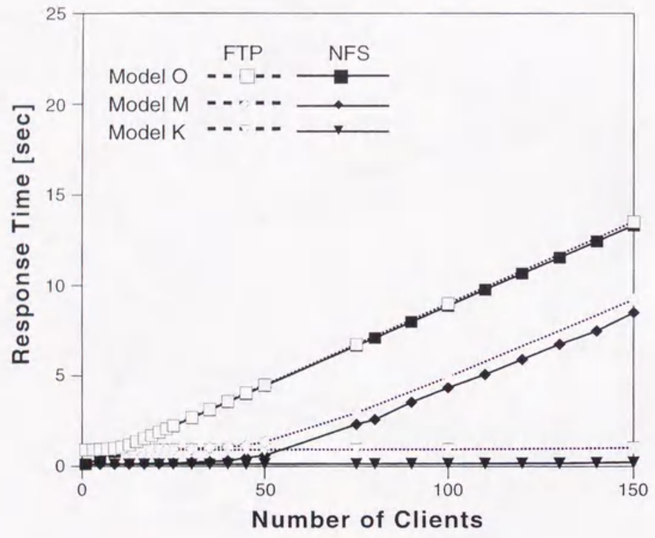


図 4.7: サーバ応答時間 (Switched-100M)

Fig. 4.7 Server response time (Switched-100M)

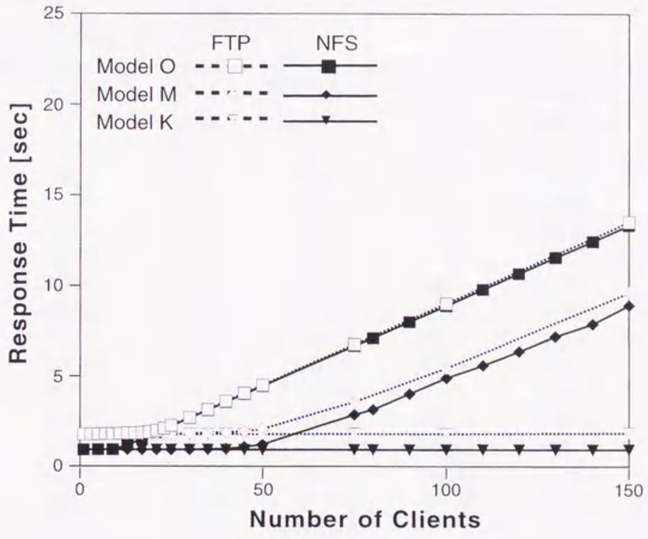


図 4.8: サーバ応答時間 (BigPipe-10M/100M)

Fig. 4.8 Server response time (BigPipe-10M/100M)

表 4.2: 平均サーバ応答時間 (Shared-10M)

	平均サーバ応答時間					
	FTP			NFS		
N	Model M	Model K	Model O	Model M	Model K	Model O
1	1.285	1.283	1.284	0.912	0.912	0.912
2	1.639	1.317	2.156	1.088	0.937	1.782
4	2.483	1.492	4.214	1.597	1.015	3.171
6	3.692	1.709	5.695	2.603	1.127	4.620
8	5.087	1.802	6.578	3.677	1.248	5.902
10	6.343	1.985	7.972	4.930	1.497	7.251
12	7.832	2.380	9.351	6.193	1.568	8.561
14	9.421	2.682	10.534	7.341	1.742	9.946
16	11.163	3.149	12.334	8.885	2.133	11.318
18	12.683	3.493	13.598	10.615	2.403	12.915
20	14.392	4.761	15.057	11.717	3.039	14.277
25	18.435	7.303	19.314	15.814	4.736	18.435
30	23.149	10.556	24.161	19.952	8.099	22.688
35	27.100	13.181	27.322	24.159	10.569	27.166
40	31.360	17.735	31.407	28.721	15.819	31.472
50	39.554	26.376	40.235	38.325	23.565	41.511
75	60.711	48.687	61.656	62.239	—	65.289
100	82.748	66.797	83.859	84.715	68.745	89.009
110	91.357	75.845	90.256	94.833	77.827	99.687
120	100.861	81.690	101.606	106.694	87.514	109.688
130	107.239	90.869	109.798	172.156	129.561	175.547
140	118.252	100.315	121.741	186.958	151.017	194.434
150	126.856	111.737	132.490	216.694	172.793	228.103

表 4.3: 平均サーバ応答時間 (Shared-100M)

	平均サーバ応答時間					
	FTP			NFS		
N	Model M	Model K	Model O	Model M	Model K	Model O
1	0.757	0.754	0.755	0.091	0.091	0.091
2	0.757	0.754	0.759	0.127	0.091	0.441
4	0.759	0.754	0.785	0.109	0.109	1.382
6	0.759	0.755	0.845	0.177	0.099	2.125
8	0.759	0.755	1.480	0.161	0.103	2.698
10	0.765	0.755	2.078	0.220	0.105	3.631
12	0.772	0.755	3.092	0.323	0.111	4.094
14	0.799	0.754	3.649	0.329	0.126	4.749
16	0.819	0.754	4.081	0.343	0.126	5.401
18	0.801	0.754	4.469	0.381	0.119	6.123
20	0.840	0.756	4.845	0.413	0.142	6.765
25	0.929	0.757	5.615	0.523	0.145	8.437
30	1.136	0.759	6.244	0.835	0.159	9.947
35	1.334	0.761	6.941	0.950	0.210	10.897
40	1.546	0.761	7.330	1.311	0.205	12.337
50	2.291	0.772	8.442	1.749	0.240	13.846
75	4.037	0.797	10.019	3.061	0.361	17.274
100	5.400	0.816	11.472	4.449	0.436	20.615
110	5.936	0.846	12.723	4.781	0.551	19.581
120	6.907	0.915	13.170	5.531	0.559	20.209
130	8.027	0.998	14.227	6.094	0.605	21.061
140	8.613	1.104	14.518	6.658	0.729	18.786
150	9.169	1.487	15.340	7.563	0.874	22.680

表 4.4: 平均サーバ応答時間 (Shared-1G)

	平均サーバ応答時間					
	FTP			NFS		
N	Model M	Model K	Model O	Model M	Model K	Model O
1	0.722	0.722	0.722	0.009	0.009	0.009
2	0.723	0.724	0.723	0.009	0.009	0.049
4	0.725	0.726	0.727	0.009	0.009	0.359
6	0.724	0.724	0.725	0.010	0.009	0.769
8	0.723	0.725	0.726	0.010	0.010	0.959
10	0.724	0.725	0.726	0.010	0.010	1.261
12	0.725	0.725	0.724	0.010	0.010	1.523
14	0.723	0.722	0.727	0.010	0.010	1.690
16	0.724	0.724	0.728	0.010	0.010	1.879
18	0.724	0.725	0.728	0.011	0.010	2.132
20	0.724	0.723	0.728	0.010	0.010	2.339
25	0.725	0.725	0.731	0.011	0.010	2.810
30	0.725	0.724	0.736	0.010	0.010	3.187
35	0.724	0.724	0.745	0.010	0.010	3.666
40	0.727	0.725	0.759	0.012	0.011	3.933
50	0.726	0.724	1.019	0.010	0.010	4.403
75	0.726	0.725	2.625	0.026	0.011	5.014
100	0.726	0.724	3.939	0.025	0.017	4.936
110	0.726	0.724	4.211	0.037	0.010	6.040
120	0.726	0.724	4.513	0.035	0.010	6.594
130	0.726	0.724	4.854	0.028	0.010	5.563
140	0.729	0.723	4.916	0.037	0.010	5.645
150	0.730	0.724	5.082	0.035	0.010	6.524

表 4.5: 平均サーバ応答時間 (Switched-10M)

	平均サーバ応答時間					
	FTP			NFS		
N	Model M	Model K	Model O	Model M	Model K	Model O
1	2.641	2.641	2.642	1.084	1.084	1.084
3	2.818	2.676	3.034	1.369	1.152	2.671
5	3.286	2.712	4.507	2.102	1.277	4.448
7	4.337	2.790	6.309	3.028	1.313	6.225
9	5.498	2.799	8.113	4.267	1.457	8.003
11	6.948	2.987	9.918	5.965	1.505	9.780
13	8.618	3.014	11.724	7.519	1.647	11.557
15	10.184	3.224	13.529	9.165	1.755	13.334
17	11.696	3.675	15.334	10.832	2.001	15.112
19	13.621	3.731	17.139	12.628	2.307	16.889
21	15.372	4.152	18.944	14.472	2.687	18.666
23	17.075	4.714	20.748	16.005	3.413	20.443
25	18.602	4.951	22.553	17.936	3.837	22.221
30	23.225	8.394	27.065	22.113	6.894	26.664
35	27.469	12.840	31.578	26.629	10.353	31.107
40	31.974	16.278	36.089	30.982	13.723	35.550
45	36.148	19.948	40.602	35.363	18.098	39.993
50	40.867	24.008	45.113	39.730	22.451	44.436
75	63.126	46.798	67.673	61.943	44.379	66.652
80	—	—	—	66.325	49.192	71.095
90	—	—	—	75.399	57.244	79.981
100	85.612	68.463	90.232	84.279	66.383	88.867
110	—	—	—	183.581	136.793	194.831
120	—	—	—	201.740	160.087	212.636
130	—	—	—	325.210	180.234	343.388
140	—	—	—	352.945	200.178	369.875
150	130.627	113.767	135.348	380.789	275.299	396.339

表 4.6: 平均サーバ応答時間 (Switched-100M)

	平均サーバ応答時間					
	FTP			NFS		
N	Model M	Model K	Model O	Model M	Model K	Model O
1	0.907	0.907	0.907	0.108	0.108	0.108
3	0.911	0.908	0.918	—	—	—
5	0.914	0.910	0.934	0.117	0.110	0.445
7	0.917	0.912	0.959	—	—	—
9	0.917	0.910	0.999	0.124	0.112	0.800
11	0.922	0.912	1.073	—	—	—
13	0.922	0.911	1.191	0.131	0.111	1.156
15	0.926	0.912	1.351	—	—	—
17	0.926	0.911	1.526	0.145	0.116	1.511
19	0.931	0.911	1.706	—	—	—
21	0.938	0.913	1.886	0.157	0.112	1.867
23	0.942	0.913	2.065	—	—	—
25	0.946	0.913	2.247	0.167	0.116	2.222
30	0.978	0.915	2.697	0.214	0.119	2.666
35	1.023	0.915	3.148	0.261	0.118	3.111
40	1.080	0.917	3.600	0.302	0.124	3.555
45	1.144	0.918	4.051	0.465	0.121	3.999
50	1.343	0.920	4.501	0.599	0.134	4.444
75	2.955	0.929	6.757	2.324	0.143	6.665
80	—	—	—	2.579	0.140	7.110
90	—	—	—	3.546	0.148	7.998
100	4.937	0.937	9.013	4.351	0.155	8.887
110	—	—	—	5.080	0.163	9.775
120	—	—	—	5.913	0.171	10.664
130	—	—	—	6.759	0.176	11.553
140	—	—	—	7.503	0.204	12.441
150	9.229	1.011	13.527	8.514	0.212	13.330

表 4.7: 平均サーバ応答時間 (BigPipe-10M/100M)

	平均サーバ応答時間					
	FTP			NFS		
N	Model M	Model K	Model O	Model M	Model K	Model O
1	1.774	1.774	1.775	0.927	0.927	0.927
3	1.777	1.776	1.781	—	—	—
5	1.781	1.778	1.788	0.927	0.927	0.928
7	1.783	1.779	1.796	—	—	—
9	1.784	1.778	1.803	0.927	0.927	0.930
11	1.787	1.779	1.816	—	—	—
13	1.788	1.778	1.829	0.927	0.927	1.162
15	1.791	1.779	1.848	—	—	—
17	1.792	1.778	1.875	0.927	0.927	1.518
19	1.795	1.779	1.915	—	—	—
21	1.801	1.780	1.984	0.927	0.927	1.873
23	1.804	1.780	2.098	—	—	—
25	1.807	1.780	2.255	0.928	0.927	2.229
30	1.825	1.782	2.701	0.936	0.927	2.673
35	1.853	1.783	3.151	0.950	0.927	3.117
40	1.892	1.784	3.602	0.971	0.927	3.561
45	1.944	1.785	4.053	1.072	0.927	4.006
50	2.074	1.786	4.503	1.214	0.927	4.450
75	3.579	1.794	6.759	2.861	0.927	6.672
80	—	—	—	3.144	0.927	7.116
90	—	—	—	4.019	0.927	8.005
100	5.452	1.802	9.014	4.896	0.927	8.893
110	—	—	—	5.607	0.929	9.782
120	—	—	—	6.385	0.929	10.670
130	—	—	—	7.203	0.931	11.559
140	—	—	—	7.886	0.939	12.448
150	9.628	1.855	13.528	8.917	0.946	13.336

表 4.8: 平均サーバ応答時間 (Shared-1G，フレームバースティングを行わない場合)

	平均サーバ応答時間		
	NFS		
N	Model M	Model K	Model O
1	0.010	0.010	0.010
2	0.010	0.010	0.028
4	0.010	0.010	0.360
6	0.010	0.010	0.718
8	0.010	0.010	0.981
10	0.011	0.010	1.218
12	0.010	0.010	1.449
14	0.011	0.010	1.682
16	0.011	0.010	1.869
18	0.011	0.010	2.049
20	0.011	0.010	2.228
25	0.012	0.010	2.531
30	0.012	0.010	2.781
35	0.013	0.010	2.948
40	0.014	0.010	3.000
50	0.016	0.010	3.168
75	0.022	0.010	3.473
100	0.034	0.010	3.825
110	0.028	0.011	3.730
120	0.041	0.011	3.964
130	0.039	0.010	3.910
140	0.051	0.012	3.834
150	0.046	0.015	3.98

媒体共有型のギガビット Ethernet では，従来の Ethernet CSMA/CD に対して，キャリア拡張とフレームバースティングの二つの拡張が行われている。フレームバースティングはキャリア拡張による有効帯域の減少を防ぐための技術だが，この機能が実装されない場合もある。表 4.8 にフレームバースティングを行わない場合の Shared-1G における平均サーバ応答時間を示す。また，図 4.9 に **4.2.5** で述べたフレームバースティングの有無による性能の比較を示す。フレームバースティングの使用による性能向上効果は見られなかった。今回想定した条件では，トラヒックの大半はサーバからの最大サイズの Ethernet パケットから構成される。このため，フレームバースティングによって連続送信されることになる短いパケットが少なかったため，このような結果になったと考えられる。

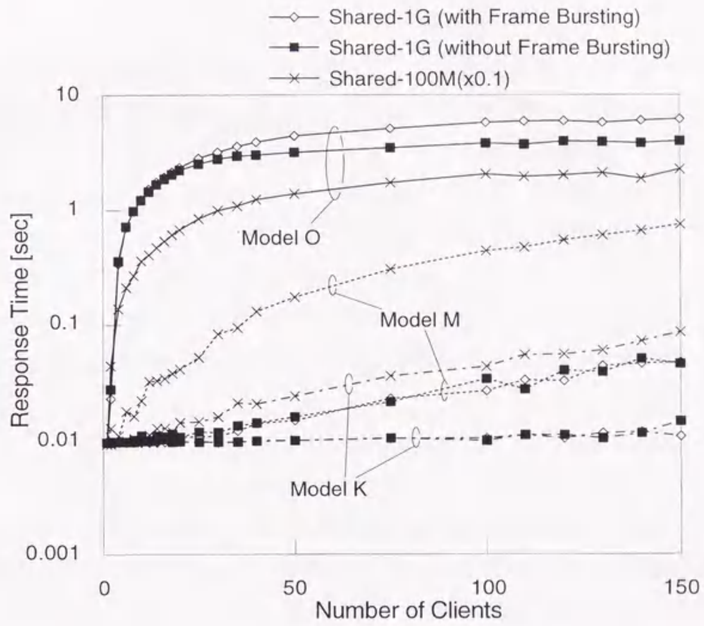


図 4.9: 媒体共有型ギガビット Ethernet と媒体共有型 100Mbps Ethernet との比較
Fig. 4.9 Comparison of the performance of shared gigabit Ethernet and shared 100 Mbps Ethernet.

表 4.9: 適切なクライアント台数のめやす

	GUI	CUI
Shared-10M	10 台以下	20 台以下
Shared-100M	30 台以下	150 台以下
Shared-1G	150 台以上	150 台以上
Switched-10M	10 台以下	20 台以下
Switched-100M	50 台以下	150 台以上
BigPipe-10M/100M	50 台以下	150 台以上

スイッチング型　スイッチング型では，媒体共有型で発生する Ethernet パケットの衝突が発生しないため，特に CWS 数が多い場合に性能の向上が期待される。伝送速度 10 Mbps の場合，スイッチング型と媒体共有型で T は大きく異ならなかった。一方，伝送速度 100 Mbps の場合には大きな違いが見られる。特にモデル O の場合，Shared-100M では N が小さい時に T が急激に増加するが，Switched-100M では，T は N に比例して増加している。モデル M における T の増加率の変化点は Shared-100M の $N = 30$ から $N = 50$ になっている。なお，媒体共有型とスイッチング型の詳細な比較では，CWS 間で応答時間のばらつきが少なく，システム全体として公平なデータ転送が可能であることがわかっている [小島 98]。

BigPipe-10M/100M では，Switched-100M とほぼ同様の結果が得られた。FS からの送信が主の場合，ハブ・CWS 間の伝送速度を 10 Mbps にしても，性能に影響はないといってよい。今回想定した条件とは逆に，CWS からのデータ転送が集中した場合には，CWS 側の小さい伝送速度が問題になるだろう。しかしながら教育用 LAN での使用条件では，FS 上のプログラム，資料などの大容量ファイルを CWS に送信することが多く，CWS からの大容量ファイルの転送が時間的に集中して行われるケースは少ない。

4.3.2　教育用 LAN の適切な設計指針

前節に示した結果より得られた，サーバの 1 つのネットワークインタフェースに接続する CWS の適切な台数のめやすを表 4.9 に示す。これらの台数の選択の基準は，アクセス集中時に 1 Mbytes のファイル転送要求に対する応答時間が 10 秒以内であり，N に対する増加率の変化がないこととした。

教育用施設の 1 教室に 30 台から 50 台程度の学生用 CWS を設置することを考慮すると，ビッグパイプ型を含む媒体共有型，スイッチング型の 100 Mbps Ethernet は適切な

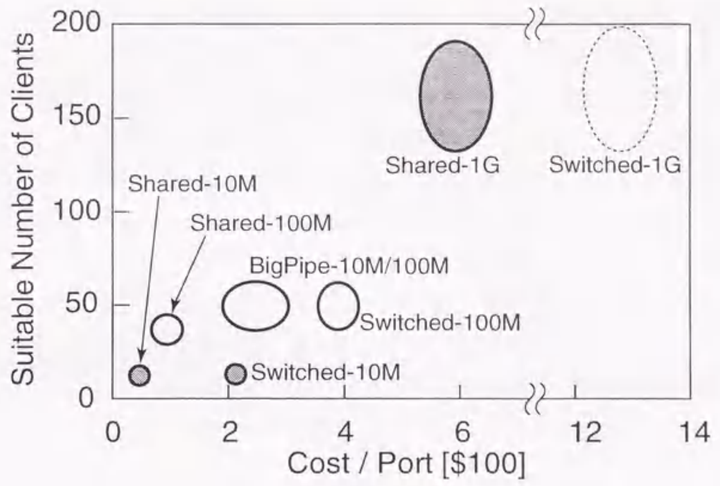


図 4.10: 教育用 LAN における Ethernet のコストと性能の関係

Fig. 4.10 Cost performance of Ethernet LANs at educational systems.

選択肢であるといえよう。しかしながら，コスト面を考慮すると，スイッチング型のネットワークを構成する場合にはビッグパイプ型を選択するのが望ましい。図 4.10 は今回評価したネットワーク構成の教育用 LAN における性能と 1999 年度におけるポートあたりの予想コストの関係を示している [Gig98]。全ての端末を 100 Mbps で接続するスイッチング型と比較して，ビッグパイプ方式および 100 Mbps の媒体共有型の性能に大きな違いはなかったが，コスト的にはこれらの方式が有利である。

また，今回比較した中では媒体共有型ギガビット Ethernet の性能が群を抜いているが，その価格はスイッチング型ギガビット Ethernet の半分，100 Mbps のスイッチング型の 2 倍以下である。また，アクセス集中時の性能は媒体共有型 100 Mbps Ethernet の 10 倍以上であった。媒体共有型ギガビット Ethernet の導入効果は高いといえよう。

4.4　まとめ

教育用 LAN 上の分散ファイルシステムにおける授業利用時のネットワーク負荷集中を防ぐための適切なネットワーク設計を得るため，ユーザの挙動モデルに基づくシミュレー

ションにより性能評価を行った。シミュレーションの結果，Ethernet を中心としたネットワーク構成における FS の 1 ネットワークインタフェースあたりの適切な CWS 数を得ることができた。設置コスト，負荷集中が起きた場合のユーザ操作に対する応答時間を考慮して現時点での教育用 LAN に適したネットワーク構成を選ぶとすると，30 から 50 台の学生用 CWS を保有する教室では，媒体共有型の 100 Mbps Ethernet もしくは，スイッチング型 Ethernet を用いた 10 Mbps / 100 Mbps のビッグパイプ方式が適切との結論を得た。

なお，本研究では FS の CPU 負荷，ディスクアクセス遅延などは考慮しなかったが，ギガビット Ethernet 環境の評価結果には少なからず影響を与えるであろう。サーバ負荷の考慮，および複数サーバ構成における性能予測は今後の課題である。

第5章　結　論

本章では本研究の内容とその成果について要約するとともに，提案したネットワークシミュレーション方式の応用，および LAN 設計と性能評価に関する今後の課題と展望について述べる。

5.1　本研究のまとめ

本論文では，詳細なモデル化に基づくネットワークシミュレーションモデルの提案とそれを用いた教育用 LAN におけるネットワークファイルシステムの評価，および可視化に適したネットワーク性能指標の提案を行った。

第 1 章では，研究に至る歴史的背景と本研究の意義と目的を述べた。

第 2 章では，イベント駆動方式に基づく多階層ネットワークシミュレーションモデルを提案した。本モデルは，過渡的な変動負荷や高負荷条件のように高精度のシミュレーションが要求される条件下で，ユーザおよびアプリケーションの挙動を考慮した現実の LAN 利用条件に即したネットワーク性能評価を指向して設計されている。従来のアプリケーションの挙動を含めたシミュレーションのためのモデルでは Ethernet などのデータリンク層以下のプロトコルを簡略化していたのに対し，提案モデルでは Ethernet の動作を忠実かつ高速に再現でき，高精度のシミュレーションが可能である。本モデルの妥当性は，現実の LAN における性能測定結果との比較によって行われ，良好な結果が得られた。

第 3 章では，可視化に適した CSMA/CD LAN のための性能評価指標としてストール率を提案した。従来のネットワーク性能指標はネットワークの性能そのものを指してお

り，ネットワークの性能上の問題を発見するためには，熟練したネットワーク技術者が複数の性能値から総合的に判断する必要があった。ストール率は専門家ではないネットワーク設計者，管理者が直観的にネットワーク性能上の問題点，原因を発見できることを可能にするための性能指標の提案の一つの試みである。ストール率は CSMA/CD LAN におけるパケット送信遅延時間に占める衝突処理・伝送路の空きの時間待ち時間などのパケット送信が滞った時間の割合を示す値である。本論文ではストール率の定義を示し，その性質を従来の性能指標との比較によって明らかにした。また，ストール率の評価を教育用 LAN の性能評価に応用した例を示し，有効性を明らかにした。

第 4 章では，教育用 LAN におけるネットワークファイルシステムの評価を行った。時間的に高い負荷集中というトラヒック特性をもつ教育用 LAN における分散ファイルシステムでのファイルサーバアクセスを想定した条件下で，第 2 章で提案したシミュレーションモデルを用いて評価した。これにあたり，教育用LAN におけるユーザ挙動をシミュレートするため，現実の教育用システムにおけるユーザ挙動の測定データに基づくユーザ挙動モデルを作成し，利用した。また，教育用 LAN に適したネットワーク構成を得るため，Ethernet を基本とした複数のネットワークトポロジにおける性能を評価し，価格性能比の観点から検証を行った。この結果，高い時間的負荷集中を持つクライアント・サーバシステムである教育用 LAN においては，媒体共有型の Ethernet およびビッグパイプ型の構成方式が有効であることがわかった。この結果は教育用のみならず，時間的に高い負荷集中がおき得る LAN においても適用可能である。

本研究では，教育用システムを指向した分散型ファイルシステム設計のための基礎研究として，(i) シミュレーションモデルの提案，(ii) 可視化に適したネットワーク性能指標の提案，(iii) 教育用システムに適した LAN 構成の検討を行った。本研究の成果は分散型ファイルシステムの利用を想定した教育用システムでの LAN 構成の指針を与えるものとして意義があるものである。

5.2 今後の展望と課題

本研究で提案したネットワークシミュレーション技法は，ネットワークの規模として単一コリジョンドメインのみを仮定しており，ルータおよびブリッジによって接続された複雑なトポロジのネットワークのシミュレーションはできない。このようなシミュレーションを可能にするには，提案モデルでは省略された IP のルーティング機能の実装，データ

リンク層におけるメッセージ経路の認識機構の実装が不可欠である。また，複雑なネットワークトポロジのデータをシミュレータに入力するための手段として，GUI または言語によるトポロジ記述方式が提供されるべきである。これらの機能はインターネット向けのシミュレータである Ns [Proa] や商用シミュレータではすでに実装されている。今後はそれらの技術との融合を図ることが必要とされよう。

本研究で提案した性能指標であるストール率は，専門家でないネットワーク管理者，設計者を対象とした直観的理解が可能なネットワーク性能評価指標を提案する試みの一つである。しかしながら，ストール率によって明らかになるネットワーク性能上の問題点は，多くある問題点の一側面を表しているに過ぎない。フレームバースティングのように，多くのパケットのデータの統計値から明らかにされる Ethernet 上の問題を発見するための直観的手法が望まれる所である。

教育用 LAN の性能評価の本来の目的は，教育用システムの利用条件に即した階層型分散ファイルシステム設計のための基礎データを得ることであった。残念ながら，その設計は未だ基礎データの検討段階であり，実現には程遠い。Andrew File System[HKM+88]，WWFS(World-Wide File System)[Prob]，およびその他の分散ファイルシステム [LS90] の既存技術と本研究による基礎データをもとに実現させたい。また，その設計には既存の技術だけでなく，今後のネットワーク技術の進歩も視野にいれる必要があろう。ネットワーク技術は今後，帯域を必要とするマルチメディア通信，長距離・高速ネットワーク，モーバイルコンピューティングを大きな核として進んでいくであろう。本研究で仮定した TCP/IP プロトコルも，長距離・高速ネットワーク時代を迎えて変わりつつある。従来その大きさが 64KB に制限されており，長距離高速ネットワークにおけるボトルネックとなっていた TCP のウインドウサイズに関しても，その制限を上回ることが可能な実装が広く利用されるようになってきている [J+92]。このような実装に基づく条件では本研究における性能評価で明らかとなった負荷集中条件においてもはるかに高い性能が得られることが予測される。

謝　辞

本論文は，筆者が名古屋大学大学院工学研究科博士課程後期課程（電子情報学専攻）在学中に行った研究成果をまとめたものである。

名古屋大学大学院工学研究科電子情報学専攻教授島田俊夫先生には，筆者の指導教官として研究の機会を与えて戴くとともに，ご指導，研究に対する姿勢へのご助言を賜わった。ここに深謝の意を表する。

名古屋大学大学院工学研究科情報工学専攻岡田稔助教授には直接の指導者としての綿密なご指導，ご助言を戴くとともに，日頃より研究者としての姿勢，在り方を教えて戴いた。また，名古屋大学大学院工学研究科計算理工学専攻教授内川嘉樹先生には，機会ある毎に叱咤激励を戴いた。ここに深謝の意を表する。

本研究の遂行には名古屋大学情報処理教育センター（現情報メディア教育センター）での実験が必要不可欠であった。同センター長（名古屋大学大学院工学研究科電気工学専攻教授）毛利佳年雄先生には，同センターでの研究の機会を与えて戴くとともに，また数多くのご助言，ご支援を戴いた。愛知県立大学情報科学部情報システム学科教授櫻井桂一先生，名古屋大学情報処理教育センター助手であった大同工業大学工学部応用電子工学科教授岩田晃先生，名古屋大学大学院工学研究科情報工学専攻助手松本哲也先生，ならびに名古屋大学大学院工学研究科電子情報学専攻助手池田幹男先生には日頃からきめ細かなご指導とご助言を戴いた。先生方およびセンター各位に深く感謝する。

名古屋工業大学工学部電気情報工学科教授田坂修二先生，同助教授石橋豊先生には，ネットワークシミュレーションに関するご助言ならびに温かい激励を戴いた。両氏に感謝する。

本研究の土台となった LAN のシミュレーションモデルの基礎モデルは名古屋大学工学部電気工学科卒研生岡崎香織氏（現 NTT データ通信）の手によるものである。本研究の出発点を築いた彼女に深謝の意を示す。また，教育用 LAN でのアクセス集中の計測実験は名古屋大学工学部電子情報学科卒研生川村尚央氏（現三井物産）の手によるものであり，本論文の第 4 章での教育用 LAN の性能評価実験に用いたプログラムおよび実験の一

部は名古屋大学大学院工学研究科電子情報学専攻大学院生（現 KDD 株式会社）小島英樹氏の手によるものである。両氏の協力に感謝の意を表する。

最後に日頃から活発な討論を戴くとともに，公私ともに良き相談相手になって戴いた，名古屋大学大学院工学研究科電子情報学専攻大学院生河合励氏，穐山譲太朗氏ほか元，現岡田研究室ならびに内川研究室各位に感謝する。

参考文献

[ANS93] ANSI/IEEE Standard 802.3. Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications, 1993.

[ANS95] ANSI/IEEE Standard 802.3. Carrier sense multiple access with collision detection (CSMA/CD) access method & physical layer specifications: MAC parameters, physical layer, medium attachment units and repeater for 100 Mb/s operation, March 1995.

[ANS97] ANSI/IEEE Standard 802.3x, y. Specification for 802.3 full duplex operation and physical layer specification for 100 Mbit/s operation on two pairs of category 3 or better balanced twisted pair cable (100BASE-T2), November 1997.

[BMK88] D. R. Boggs, J. C. Mogul, and C. A. Kent. Measured capacity of an ethernet: Myths and reality. In *Proc. of SIGCOMM '88*, pp. 222–234. ACM, 1988.

[BSP90] J. C. Bolot, A. U. Shankar, and B. D. Plateau. Performance analysis of transport protocols over congestive channels. *Performance Evaluation*, Vol. 11, No. 1, pp. 45–66, 1990.

[CS91] D. E. Comer and D. L. Stevens. *Internetworking with TCP/IP Vol. II: Design, Implementation, and Internals.* Prentice-Hall, Inc., 1991. (邦訳 村井, 楠本, TCP/IP によるネットワーク構築 Vol. II, 共立出版 (1994)).

[DM92] E. Drakopoulos and M. J. Merges. Performance analysis of client-server strage systems. *IEEE Trans. Comput.*, Vol. 41, No. 11, pp. 1442–1452, 1992.

[Gig98] Gigabit Ethernet Alliance. Gigabit ethernet overview. http://www.gigabit-ethernet.org/technology/whitepapers/gige_0698/geal1998v3.pdf, 1998.

[Gus90] R. Gusella. A measurement study of diskless workstation traffic on an ethernet. *IEEE Trans. Commun.*, Vol. 38, No. 9, pp. 1557–1507, 1990.

[HKM+88] J. H. Howard, M. L. Kazar, S. G. Manees, D. A. Nichols, M. Satyanarayanan, and R. N. Sidebothan. Scale and performance in a distributed file system. *ACM Trans. Comput. Syst.*, Vol. 6, No. 1, pp. 55–81, 1988.

[ICT93] O. C. Ibe, H. Choi, and K. S. Trivedi. Performance evaluation of client-server systems. *IEEE Trans. Parallel and Distributed Systems*, Vol. 4, No. 11, pp. 1217–1229, 1993.

[IKO98a] S. Ishihara, H. Kojima, and M. Okada. Performance evaluation of educational workstation LAN systems. In *Proc. of World Multiconference on Systemics, Cybernetics and Informatics '98*, Vol. 2, pp. 612–619, 1998.

[IKO98b] S. Ishihara, H. Kojima, and M. Okada. Performance evaluation of educational workstation network systems. In *Proc. of SoftCOM'98*, pp. 549–558, 1998.

[IO97a] S. Ishihara and M. Okada. A modeling and simulation method for transient traffic LAN. *Trans. IEICE*, Vol. E–80–B, No. 8, pp. 1239–1247, 1997.

[IO97b] S. Ishihara and M. Okada. A stall rate analysis of heavy and transient traffic LAN by simulation on multi-layer protocols. In *Proc. of Information Systems and Technologies for Network Society*, pp. 64–67. IPS Japan, 1997.

[J+92] V. Jacobson, et al. TCP extentions for high performance. RFC 1323, 1992.

[LM+89] S. J. Leffler, Marshall Kirk McKusick, *et al. The design and implementation of the 4.3BSD UNIX operating system.* Addison-Wesley Publishing Company, Inc., 1989. (邦訳 中村明 他, UNIX 4.3 BSD の設計と実装, 丸善 (1991)).

[LM94] A. M. Law and M. G. McComas. Simulation software for communications networks: The state of the art. *IEEE Commun. Mag.*, Vol. 32, No. 3, pp. 44–50, 1994.

[LS90] E. Levy and A. Silberschatz. Distributed file systems: Concepts and examples. *ACM Computing Serveys*, Vol. 22, No. 4, pp. 321–374, 1990. (邦訳 天海良治, 分散ファイルシステム：その概念と実例, bit 別冊コンピュータサイエンス, 共立出版, pp. 141–191, (1992)).

[MdR93] P. Marino and A. del Rio. An accurate and fast CSMA/CD simulator. *Microprocessing and Microprogramming*, Vol. 39, pp. 197–190, 1993.

[MKk97] M. Molle, M. H. Kalkunte, and J kadambi. Frame bursting: a technique for scaling CSMA/CD to gigabit speeds. *IEEE Mag. Network*, Vol. 11, No. 4, pp. 6–15, 1997.

[Mol94] M. Molle. A new binary logarithmic arbitation method for ethernet. Technical Report CSRI-398, Computer Systems Research Institute, University of Toronto, Toronto, Canada, 1994.

[MT88] M. Murata and H. Takagi. Two-layer modeling for local area networks. *IEEE Trans. Commun.*, Vol. 36, No. 9, pp. 1022–1034, 1988.

[OH94] P. J. P. O'Reilly and J. L. Hammond. An efficient simulation technique for peformance studies of CSMA/CD local networks. *IEEE J. Sel. Areas Commun.*, Vol. SAC-2, No. 1, pp. 238–249, 1994.

[Pos81a] J. Postel. Internet protocol. RFC 791, Information Science Institute/ University of Southern California, 1981.

[Pos81b] J. Postel. Transmission control protocol. RFC 793, Information Science Institute/ University of Southern California, 1981.

[Pos81c] J. Postel. User datagram protocol. RFC 768, Information Science Institute/ University of Southern California, 1981.

[Pos85] J. Postel. File transfer protocol. RFC 765, 1985.

[Proa] VINT Project. http://www-mash.cs.berkeley.edu/ns/ns.html.

[Prob] WIDE Project. http://www.wide.ad.jp/software/readme.wwfs-109.5.html.

[Sun89] Sun Microsystems, Inc. NFS: Network file system protocol specification. RFC 1094, 1989.

[WSF94] B. Whetten, S. Steinberg, and D. Ferrari. The packet starvation effect in CSMA/CD LANs and a solution. In *Proc. Conference on local computer networks*, Vol. 19, pp. 206–217, 1994.

[秋丸 90] 秋丸春夫, 川島幸之助. 情報通信トラヒック – 基礎と応用. オーム社, 1990.

[石田 95] 石田陽子, 高原幹男. イーサネット上のトラヒック特性によるパケット発生モデルについて. 電子情報通信学会論文誌, Vol. J78–B–I, No. 11, pp. 664–671, 1995.

[石原 95] 石原進, 岡田稔, 岩田晃, 櫻井桂一. イベント駆動方式による LAN 通信量解析モデル. 電子情報通信学会論文誌, Vol. J78–A, No. 8, pp. 961–964, 1995.

[石原 96] 石原進, 岡田稔, 岩田晃, 櫻井桂一. 教育用 LAN の通信量解析. 1996 年情報学シンポジウム講演論文集, pp. 17–23, 1996.

[石原 97] 石原進, 岡田稔. ストール率による高負荷 CSMA/CD LAN のトラヒック解析. 電子情報通信学会技術研究報告・情報ネットワーク研究会 IN97-106, 1997.

[石原 99] 石原進, 岡田稔. 可視化に適した CSMA/CD LAN のための性能指標：ストール率. 情報処理学会論文誌, Vol. 40, No. 1, pp. 95–102, 1999.

[岩田 98] 岩田晃, 石原進, 小島英樹, 岡田稔. 一般情報処理教育における大規模分散システムの利用実態. 情報処理学会研究報告・分散システム運用技術研究会 DSM98-10, Sep 1998.

[浮貝 92] 浮貝雅裕, 菅原研次, 三井田惇郎. 情報系学科新入生に対する導入教育とそのための演習教育環境. 情報処理学会論文誌, Vol. 33, No. 1, pp. 1–9, 1992.

[岡田 94] 岡田稔, 岩田晃, 松本哲也, 池田幹男. 教育用大規模分散型 WS システムの構成方法. 情報処理学会研究報告・分散システム運用技術研究会 DSM-9411020, 1994.

[岡田 96] 岡田稔, 櫻井桂一, 岩田晃. 教育用大規模分散型 WS システムの一構成方法. 情報処理学会論文誌, Vol. 37, No. 12, pp. 2447–2456, 1996.

[小島 98] 小島英樹, 石原進, 岩田晃, 岡田稔. イベント駆動シミュレーションによるイーサネットハブの特性解析. 電子情報通信学会技術研究報告・情報ネットワーク研究会 IN97–165, 1998.

[中山 94] 中山仁, 大西淑雅, 末永正, 有田五次郎. 工学系学生のための情報処理集合教育環境の設計と構築. 情報処理学会論文誌, Vol. 35, No. 11, pp. 1442–1452, 1994.

[松本 94] 松本哲也, 池田幹男, 岩田晃, 岡田稔. 教育用大規模分散型 WS システムの利用状況. 情報処理学会研究報告・分散システム運用技術研究会 DSM-9411021, 1994.

[村田 94] 村田正幸, 宮原秀夫. 通信トラヒック理論とその応用 (i) システムのモデル化と性能評価. 信学誌, Vol. 77, No. 9, pp. 968–975, 1994.

[村田 95] 村田正幸, 宮原秀夫. 通信トラヒック理論とその応用 (vi) シミュレーション技法と その適用例. 信学誌, Vol. 78, No. 3, pp. 264–270, 1995.

論文目録

[1] 石原進, 岡田稔, 岩田晃, 櫻井桂一. イベント駆動方式による LAN 通信量解析モデル. 電子情報通信学会論文誌, Vol.J78-A, No.8, pp.961–964, 1995.

[2] S. Ishihara and M. Okada. A Modeling and Simulation Method for Transient Traffic LAN. *Trans. IEICE Comm.*, Vol. E-80-B, No.8, pp.1239–1247, 1997.

[3] 石原進, 岡田稔. 可視化に適した CSMA/CD LAN のための性能評価指標: ストール率. 情報処理学会論文誌, Vol. 40, No. 1, pp. 95–101, 1999.

[4] S. Ishihara and M. Okada. A Stall Rate Analysis of Heavy and Transient Traffic LAN by Simulation on Multi-layer Protocols. *In Proc. of Information Systems and Technologies for Network Society*, pp. 64-67, Sep. 1997 (Fukuoka/Japan).

[5] S. Ishihara, H. Kojima and M. Okada. Performance Evaluation of Educational Workstation LAN Systems. *In Proc. of WSCI'98, World Multiconference on Systemics, Cybernetics and Informatics*, Vol. 2, pp. 612-619, Jul. 1998, (Orlando/U.S.A.).

[6] S. Ishihara, H. Kojima and M. Okada. Performance Evaluation of Educational Workstation Network Systems. *In Proc. of SoftCOM'98, International Conference on Software in Telecommunications and Networks*, pp. 549–558, Oct. 1998, (Bari/Italy).

[7] 石原進, 岡田稔, 岩田晃, 櫻井桂一. 教育用 LAN における通信量モデリングの一手法. 電子情報通信学会技術研究報告・回路とシステム研究会, CAS94-88, pp.29–36, 1995.

[8] 石原進, 岡田稔, 岩田晃, 桜井桂一. 教育用 LAN の通信量解析. 1996 年情報学シンポジウム講演論文集, pp.17–23, 1996.

[9] 石原進, 岡田稔. 高負荷多セグメントネットワークにおけるトラフィック評価. 情報処理学会研究報告・分散システム運用技術研究会, 97DSM5-4, pp. 19–24, 1997.

[10] 石原進, 岡田稔. 高負荷 LAN におけるクライアント・サーバシステムの性能評価. 電子情報通信学会技術研究報告・情報ネットワーク研究会, IN96-129, pp. 9–16, 1997.

[11] 小島英樹, 石原進, 岩田晃, 岡田稔. 大規模分散型教育用ワークステーションシステムの利用実態. 情報処理学会研究報告・分散システム運用技術研究会, 97-DSM-5-5, pp. 25–3, 1997.

[12] 石原進, 岡田稔. ストール率による高負荷 CSMA/CD LAN のトラヒック解析. 電子情報通信学会技術研究報告・情報ネットワーク研究会, IN97-106, pp.49-54, 1997.

[13] 小島英樹, 石原進, 岩田晃, 岡田稔. イベント駆動シミュレーションによるイーサネットハブの特性解析. 電子情報通信学会技術研究報告・情報ネットワーク研究会, IN97-165, pp. 37–44, 1998.

[14] 石原進, 小島英樹, 岡田稔. ユーザ挙動モデルに基づく脅威今日システムに適したネットワーク構成の検討. 情報処理学会研究報告・分散システム運用技術研究会, 98-DSM-11, pp. 7–12, 1998.

[15] 岩田晃, 石原進, 岡田稔. 一般情報処理における大規模分散システムの利用実態. 情報処理学会研究報告・分散システム運用技術研究会, 98-DSM-18, pp. 55–61, 1998.

[16] 石原進, 岡田稔, 櫻井桂一. 情報処理教育用 LAN のネットワーク負荷解析. 平成 6 年電気関係学会東海支部連合大会, p.401, 1994.

[17] 石原進, 岡田稔, 岩田晃, 櫻井桂一. シミュレーションモデルによる教育用 LAN の通信量解析. 平成 7 年前期情報処理学会全国大会, 1-147, 1995.

[18] 石原進, 岡田稔, 岩田晃, 桜井桂一. 教育用 LAN の通信量解析モデルの評価. 平成 7 年電気関係学会東海支部連合大会, 610, 1995.

[19] 石原進, 岡田稔, 岩田晃, 桜井桂一. 教育用 LAN 環境における高速 LAN の性能評価. 平成 8 年電子情報通信学会総合大会, B-840, 1996.

[20] 石原進, 岡田稔. 負荷集中 LAN におけるネットワークファイルシステムの性能評価. 平成 8 年電気系学会東海支部連合大会, 633, 1996.

[21] 石原進, 岡田稔. 高負荷多セグメントネットワークの性能評価モデル. 平成 9 年電子情報通信学会総合大会, B-7-179, 1997.

[22] 小島英樹, 石原進, 岩田晃, 岡田稔. 大規模分散型教育用計算機システムの利用実態, 平成 9 年電気系学会東海支部連合大会, 640, 1997.

[23] 石原進, 岡田稔. 教育用 LAN における共有ギガビット Ethernet の評価. 平成 10 年電子情報通信学会総合大会, B-7-169, 1998.

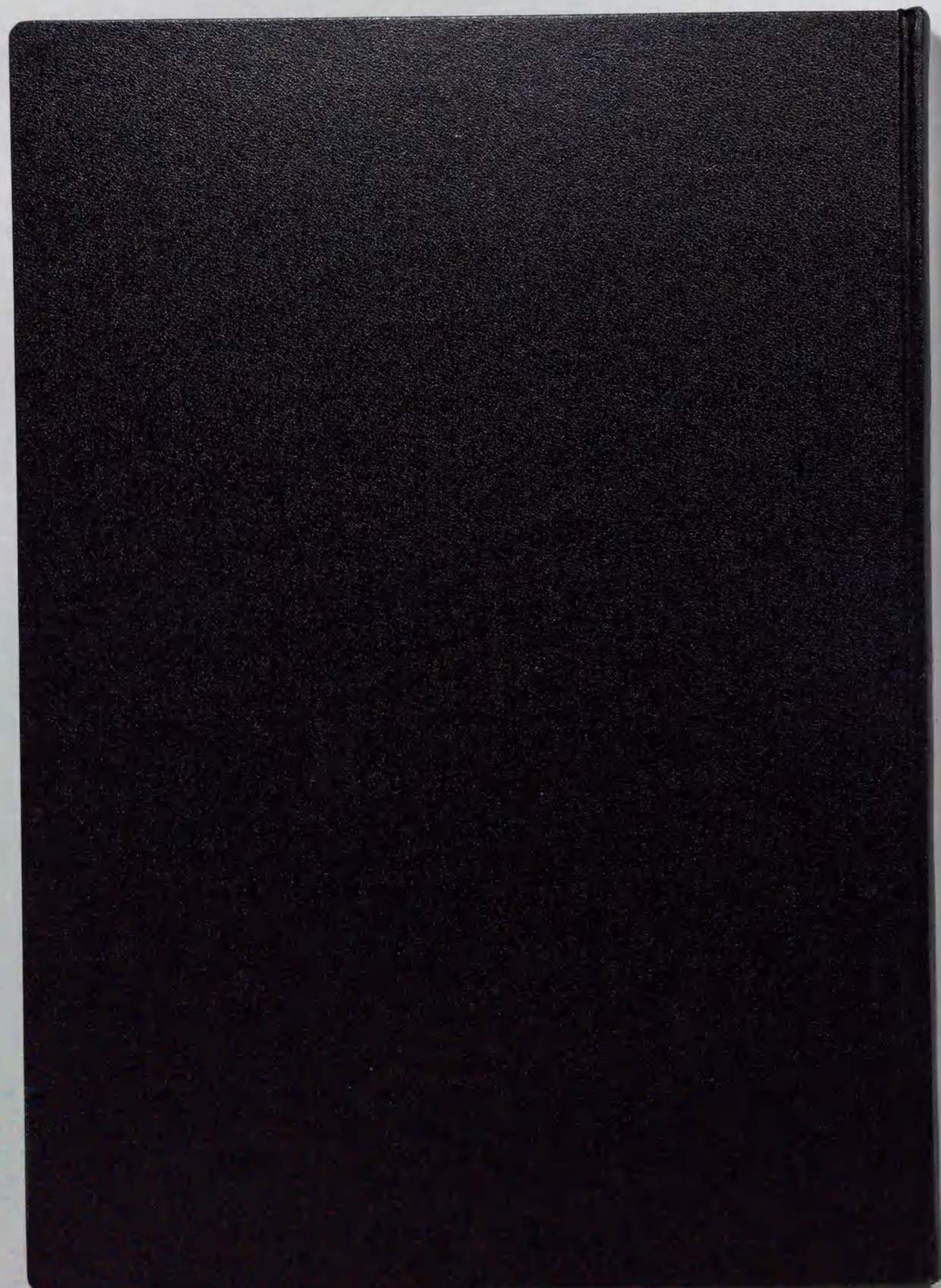
[24] 小島英樹, 石原進, 岩田晃, 岡田稔. クライアント・サーバシステムにおけるスイッチング HUB の性能評価. 平成 10 年電子情報通信学会総合大会, B-7-171, 1998

[25] 石原進, 岡田稔. 負荷集中条件下の NFS 運用時における媒体共有型ギガビット Ethernet の評価. 平成 10 年電気系学会東海支部連合大会, 617, 1998.

[26] 石原進, 岡田稔. 多階層モデルによるギガビットネットワークの評価. 平成 10 年後期情報処理学会全国大会, 3-438, 1998.

ネットワークモデリングとそれを用いたネットワークの性能評価

石原 進

（名古屋大学大学院工学研究科電子情報学専攻）

1999 年 1 月