高速 LAN に適したデータ転送プロトコルの設計と評価†

本村公太†† 坂口勝章††

LAN は光伝送技術の進展に伴い益々高速化している．それに対し，OSI トランスポート層でのプロトコル処理速度は遅く，アプリケーションへ高速データ転送サービスを提供する際のボトルネックとなっている．また，LAN のアプリケーションの中には，静止画データベースアクセスのように転送方向によって要求されるサービス品質が異なる通信や，RPC のようにコネクションレス型でかつ信頼性も重要な通信を必要とするものなど，OSI プロトコルではカバーしきれない通信を要求するものもある．これらの性能的・機能的な問題を解決するため，著者らは新しいデータ転送用プロトコルとして高速転送プロトコル (HTP) を提案する．HTP は，LLC 副層からトランスポート層までを一つの層に縮退させることによって重複・不要機能の排除と層間インタフェースの削減を図り，さらにフロー制御や誤り回復などの手順要素の修正や追加によって高速化と高機能化を図っている．HTP を実装した通信処理プログラムの試作から得られたダイナミックステップ等の性能評価データから，HTP の処理のみに着目した場合，6 MIPS 程度の CPU を使用することによって 100 Mbps の高速性，2 msec の低遅延性を得られることを示す．

1. はじめに

ローカルエリアネットワーク (LAN) は光伝送技術の進展に伴って高速化し，伝送速度 100 Mbps の LAN も FDDI (Fiber Distributed Data Interface) として標準化されている．一方，LAN のアプリケーションとしては静止画データベースアクセス，リモートファイルアクセス，CAD データの転送，実時間制御システム (ファクトリオートメーション) 等，数十 Mbps 程度の高速性，数十 msec 以下の低遅延性を必要とするものが多くなってきている．

これに対して OSI プロトコルの性能的な面を見ると，トランスポート層でのプロトコル処理速度は一般の環境ではせいぜい数 Mbps であり，20 Mbps を得るためには 18 MIPS の CPU が必要との計算もある[1]．プロトコル処理を高速化するために処理のハードウェア (LSI) 化を図る方法は，OSI プロトコル (特にトランスポートプロトコルクラス 4) の処理の複雑さや論理深度の深さから，困難であると言われている．最近になってトランスポート層までのデータ転送正常処理のみを専用ハードウェアで行うことにより，ソフトウェアに比べて処理時間を 1/10 程度にするというアプローチが発表された[2]．しかしながら，トランスポート層やネットワーク層で必須機能とされている分割・組立て機能が不要な通信への適用のみを対象としているため，現時点では OSI プロトコルのハードウェア処理に成功しているとは言い難い．分割・組立て機能を考慮しなければ，送信時はデータ部の先頭にヘッダを付加するだけで済み，受信時は除去していくだけで済むため，処理がかなり簡単にできる．以上のように，OSI プロトコルでは，LAN 自体が高速化してもアプリケーションが必要とする高速データ転送サービスを提供することが困難となっている．

さらに，機能的な面において，静止画データベースアクセスのようにコマンドを転送する方向と静止画データを転送する方向では要求される通信品質が異なってくるものもあり，OSI での双方向同一品質のコネクション (CO) 型通信ではその要求に応えられない．また，RPC (Remote Procedure Call) には CL 型の通信が適するが，同時に信頼性も重要であるため，OSI で規定されている転送保証のないコネクションレス (CL) 型通信を使用した場合には，上位で誤りが発生した場合に備える必要が生じる．

このような OSI プロトコルの性能的・機能的な問題に対処するために，新しいプロトコルを構築するさまざまな試みがなされている．例えば，大容量のデータ転送を指向して複数のパケットをまとめたブロック単位でフロー制御や再送制御を行う高速通信プロトコル[3]，パラメータ長を CPU のバス幅に合せたり，チェックサムパラメータを末尾にするなどハード処理を指向した XTP[4]，ページ単位のネットワークファイルアクセスをトランザクション処理で行うのに適した VMTP[5] などが検討されている．1990 年には，コンピュータ間通信に関する主だった会議において高速プロトコルに関するチュートリアル[6),7)] が実施されるなど，新しいプロトコルに対する関心が高まってい

† Design and Evaluation of the Data Transfer Protocol for High-speed LANs by KOTA MOTOMURA and KATSUAKI SAKAGUCHI (NTT Telecommunication Networks Laboratories).

†† NTT 通信網総合研究所

る．また，米国で3･4層の標準化を行っている ANSI X3S3.3 委員会は，1988年11月に高速プロトコルについて検討するワーキンググループを設置し，要求条件を中心に議論を進めている[8]．ANSI での動きは，ローカルプロトコルの乱立状態になってしまいそうな高速プロトコルを標準化ベースで考えていこうとするものであり，標準としての OSI プロトコルと高速プロトコルの共存の可能性を示唆するものである．また，U. S. Navy の SAFENET (Survivable Adaptable Fiber Optic Embedded Network) では，プロファイルとして，GOSIP (Government Open Systems Interconnect Profile) に準拠したフル OSI 版と簡易なプロトコルを採用した軽量版との二本立てとし，用途に応じて使い分けるという考え方になっている[9]．

前述した OSI プロトコルの性能的・機能的な問題に対処するため，筆者らは，層の縮退と手順要素の修正や追加によって高速化と高機能化を狙った新しいデータ用プロトコルとして高速転送プロトコル (HTP : High-speed Transfer Protocol) の検討を進めてきた[10),11]．HTP は，論理リンク制御 (LLC) 副層からトランスポート層までを一つの層に縮退させ，重複・不要機能の排除と層間インタフェースの削減によって高速化を図っている．さらに，明示的再送要求と選択再送方式の採用，最大サービスデータ単位 (SDU) 長に基づくフロー制御の最適化，送達保証 CL 型通信のサポートなどによって高速化・高機能化を図っている．従来検討されてきた新プロトコルは特定のアプリケーション（例えば，ファイル転送）を想定して設計されることが多かったが，HTP はできるだけ汎用性を保ったまま高速化・高機能化するように設計を進めた．

本論文では，HTP の考え方や規定内容について述べるとともに，到達可能性解析による検証結果，HTP を実装した通信処理プログラムの試作から得られたダイナミックステップ等の性能評価データについて報告する．

2. HTP の考え方と規定内容

2.1 HTP の設計目標

HTP は，LAN 内アプリケーションが要求する次のような多様な通信への適用をスコープとする．

- ファイル転送等の大容量（高速）通信
- ファクトリオートメーション等の実時間（低遅延）通信
- 静止画データベースアクセス等の転送方向ごとに異なるサービス品質 (QOS) を要求する通信
- コマンドとレスポンスの対で1回の通信が終了するトランザクション処理向け送達保証 CL 型通信

これらの多様な通信要求に応えるために，以下の点を目標として設計する．

- プロトコル全体の軽量化
- 効率的なフロー制御と誤り回復
- 非対称通信のサポート
- CL 型通信の信頼性向上 (CO と CL の一体化)

前提とするネットワークは，対象とするアプリケーションの利用形態から，ブリッジで接続された複数の LAN の範囲を基本として考える．

HTP の利用者としては，OSI プロトコルが提供するサービスでは機能的・性能的に満足できない利用者を想定し，他の LAN との公衆網を介した広範囲な相互通信を必要とする利用者は OSI プロトコルを使用するものと考える．すなわち，HTP と OSI プロトコルは適用域が異なるものであり，LAN 内で共存するものと位置付ける．

2.2 アプローチと設計のステップ

LAN の環境においては，通常，ネットワーク層まではコネクションレス型のプロトコルが使用されるため，トランスポートプロトコルとしては最も高機能なクラス4が使用される[12]．そのため，トランスポートプロトコルの処理がボトルネックとなり[1]，高速プロトコルの研究としてはトランスポートプロトコルの簡略化に主眼の置かれることが多い．しかしながら，階層化されたプロトコルの実装においては，各層でのプロトコル処理のみならず層間インタフェースのオーバヘッドが大きく，トランスポート層のプロトコルだけを簡略化しても，システム全体の高速化には不十分であると考える．そのため HTP では，OSI の7層モデルでの複数の層を一つの層に縮退させることを，プロトコル全体の軽量化のための基本的なアプローチとする．この層の縮退のステップにおいて，層間で重複する機能は一つにし，必要性の少ない機能は省略する．さらに，実装上のオーバヘッドの大きい層間インタフェースを削減する．

次に手順要素の修正・追加のステップとして，フロー制御や誤り回復（再送）などの手順要素を高速・高機能に適したものに変更する．従来規定されていた手順要素はそれらのプロトコルが設計されたときの環

境(高ビット誤り率,低伝送速度)を前提に考えられており,光 LAN の環境に適したものではないと考えるからである.例えば,ビット誤り率が小さくなると,大きなプロトコルデータ単位 (PDU) 長を使用して大きなウインドウサイズを使用しても伝送誤りが発生する可能性は小さくなるが,もし誤りが発生した場合,誤りのあった PDU 以降を全部再送する方式では,転送効率の急激な低下が生じる.このため,光 LAN の環境では,再送方式としては選択再送が望ましいと言える.この手順要素の修正・追加のステップにおいて,従来の OSI にはなかった転送方向別の QOS 制御,送達保証型 CL 型通信などの手順も追加し,高機能化を図る.

2.3 層の縮退

(1) 縮退の対象とする層

次の理由により,LLC 副層からトランスポート層までを対象とする.

① LAN 間をブリッジで接続したとき,LLC 以上はすべてエンドツーエンドのプロトコルになる.

② 分割と組立て,誤り検出など重複機能が LLC 副層からトランスポート層までに多い.

③ エンドツーエンドで確実にデータを転送する機能はトランスポート層までで満たされる.その後処理は,アプリケーションが必要に応じて行える.

④ サービスプリミティブの少ない層(使いやすい層)で切るのが望ましい.

高速転送プロトコルの階層モデルを図1に示す.

高速転送層のプロトコルである HTP は,MAC 副層以下が提供する高速の伝送サービスを利用して,高速・高信頼・高機能のデータ転送サービスを利用者に提供する.

(2) 統一する機能

統一の対象として考える各層のプロトコルは,LAN の環境で標準的に使用されるプロファイル[12]として LLC はタイプ1,ネットワークは CL 型ネットワークプロトコル(CLNP),トランスポートはコネクション型トランスポートプロトコルクラス4(TP 4)とする.

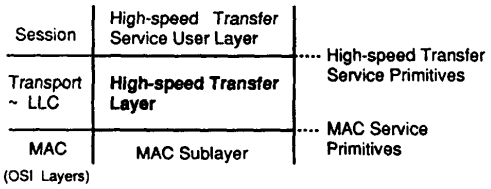


図1 高速転送プロトコルの階層モデル
Fig. 1 Layered model of the high-speed transfer protocol.

統一する機能は,各プロトコルに共通にある PDU の作成・分解機能,PDU 転送機能,通信相手識別機能,さらに CLNP と TP 4 にある分割と組立て機能,誤り検出機能である.

(3) 省略する機能

TP 4 の連結と分離の手順は,通常の実装においてはあまり使用しないうえ,処理の複雑化を招くので省略する.また,下位の伝送速度の向上によって待ちキューがたまるような状況が考え難いため,優先データ転送の手順は省略する.

CLNP の経路選択に関する機能や PDU の寿命制御に関する機能は,ブリッジで接続された範囲のネットワークを対象とするので省略する.

2.4 手順要素の修正と追加

(1) 明示的再送要求

TP 4 ではタイムアウトにより PDU を再送する方式が規定されている.この方式ではタイムアウトが発生するまで PDU の再送が行われないため,伝送誤り検出による受信側での PDU 廃棄あるいは下位層での PDU 廃棄に対する回復が遅れる.

HTP では,送達確認 PDU の中に再送要求を指示するフラグを設けることにより,再送要求を明示的に通知し,迅速な回復を可能とする.

(2) 選択再送方式

TP 4 でのタイムアウトによる再送は,一つのウインドウ内で連続して送信した複数の PDU に対する各再送タイマがほぼ同時にタイムアウトするため,実効的に,再送が必要な PDU 以降を連続的にすべて再送する全再送方式になっている.このため,正常に受信されていた PDU まで再送されることになり,いったん,PDU 廃棄が発生すると転送効率が急に低下する.

HTP では,この問題を防止するため,制御が若干複雑になるが効率のよい選択再送方式を採用する.選択再送を実現した場合に受信側でのバッファ管理を効率化するため,分割と組立ての手順で使用するパラメータとして TP 4 での終端識別子に加えて,SDU 識別子,セグメントオフセットを使用する.

(3) 最大 SDU 長に基づくフロー制御の最適化

高速データ転送のためには,データ転送中,特に一つの SDU 分のデータ転送中にはフローの停止が起こらないことが望ましい.しかしながら,TP 4 でのフロー制御は,SDU 長とは無関係に受信可能な PDU

数に基づいて行われるため，一つの SDU 分のデータ転送中にフローが停止したり，逆に必要以上のバッファを確保して資源の無駄が生じたりする．これは，TP 4 では，トランスポートエンティティ同士が相手の最大 SDU 長を知らないため，それに合せたバッファ確保の最適化が困難なためである．

HTP では，コネクション確立時のプリミティブおよび PDU に最大 SDU 長のパラメータを設け，それぞれの転送方向で使用する最大の SDU 長を利用者からの要求に基づいてコネクション確立時に HTP エンティティ間で交換する．このパラメータに基づいて送信/受信バッファの確保を行うことにより，バッファビジーを回避する．また，最大 SDU 長と最大 PDU 長とからウインドウサイズ（クレジット値）を決定し，一つの SDU 分のデータ転送中のフローの停止を回避する．この手順により，利用者および転送方向ごとのデータ量に応じた最適フロー制御と資源の効率的利用を図ることができる．

（4） 転送方向別 QOS 制御

TP 4 での QOS は，転送方向には関係なく，コネクション対応に定まる．しかしながら，静止画データベースアクセスの場合には，伝送誤り（ビット誤り）に対する許容率の厳しいコマンドを送る端末からセンタの方向と，許容率の緩い静止画データを送るセンタから端末の方向では要求される QOS が異なる．このような通信において，コマンドから要求される QOS に合せて伝送誤り検出（チェックサム）の手順を常に使用すると，データ量の多い静止画にも不必要な手順を使用することになり通信速度が遅くなる．

HTP では，このような非対称通信の要求条件に応えるため，コネクション確立時に転送方向別の QOS を指定し，それに基づいてデータ転送フェーズでの QOS の制御を行う．

（5） コネクション解放時の保留モード

TP 4 では，利用者からのコネクション解放要求プリミティブを受け取ると直ちに解放フェーズに移行し，それまでのデータ転送の保証をしない．このためダイアローグ管理の機能が必要なくともセションプロトコルを載せる必要がある．

HTP では，コネクション解放に際してそれまでのデータの転送が保証できるようにするため，送達が完了するまで解放フェーズに移行しない動作モードを設けることとする．このモードを使用するか否かは，コネクション解放要求プリミティブの切断モードパラメータで指定可能とする．このモードによるコネクションの解放を，本論文ではグレースフル解放（graceful release）と呼ぶ．この手順によって，未転送データに対して上位層で対処する必要がなくなる．

（6） 送信側からの送達確認要求

TP 4 で送達確認 PDU を返すタイミングは，受信側に任されており，コネクション確立時に交換する確認時間内という制限だけがある．そのため，送信側が性能確認のためにラウンドトリップ時間を測定しようとしても，送達確認 PDU がすぐに返ってこない場合もあり得る．また，送信側のバッファに余裕がないために保持している PDU を早く解放したい場合にも，なかなか解放できないこともあり得る．

HTP では，データ PDU に送達確認要求のビットを設け，送信側が自分の必要なタイミングで受信側からの送達確認 PDU を受け取ることを可能とする．

（7） 送達保証コネクションレス型通信

OSI におけるコネクションレス型通信は，ネットワーク層はもちろん，トランスポート層においてもデータ転送の保証はしない．分散ファイルシステム等に利用される RPC にはコネクションレス型通信が適しているが，信頼性も重要である．

HTP では，CL 型の PDU の構成やパラメータを CO 型の PDU のものとできるだけ共通化することにより，CL 型のデータ転送において CO 型と類似の処理ができるようにし，送達保証（再送による回復を行う）コネクションレス型通信を実現する．

ここまで述べてきた機能を含め，HTP において規定している機能と CLNS 上の TP 4 として規定されている機能との比較を**表 1** に示す．

2.5 サービス品質とサービスプリミティブ

（1） 提供するサービス品質

光ファイバを利用する高速 LAN では伝送誤り率が低いため，画像系データ等に対しては伝送誤り検出手順を使用しなくとも要求品質を満足できる場合もある．したがって，伝送誤り検出手順を使用するか否かを QOS として指定可能とする．

また，実時間制御やリモートセンシング等の場合を考えると，伝送誤りを検出しても再送を要求せずに次のデータを待った方が望ましい場合や，利用者がデータを受け取る前に次のデータがきたときフローを止めるよりもデータを上書きした方が望ましい場合もある．したがって，再送の要否，およびデータを上書き

表 1 HTP と TP 4 の機能比較
Table 1 Protocol mechanism comparison between HTP and TP4.

Protocol mechanism (Procedure)	TP4 over CLNS	HTP	
		CO type	CL type
PDU transfer	◎	◎	◎
Segmenting and reassembling	◎	◎	◎
Concatenation and separation	◎	×	×
Connection establishment	◎	◎	×
Connection refusal	◎	◎	×
Normal release	◎	◎	×
Graceful release	×	○	×
Error release	×	◎	×
Association of PDUs with Connections	◎	◎	×
One-way QOS control	×	◎	◎
Data PDU numbering	◎	◎	◎
Expedited data transfer	◎	×	×
Retention until acknowledgement of PDUs	◎	○	○
Acknowledgement request	×	◎	○
Explicit flow control	◎	◎	×
Checksum	○	○	○
Frozen references	◎	◎	○
Retransmission request	×	○	○
Selective retransmission	×	○	○
Retransmission of time-out	◎	○	○
Resequencing	◎	◎	×
Inactivity control	◎	×	×
Treatment of protocol errors	◎	◎	◎

◎: Mandatory, ○: Optional (depend on QOS), ×: Not applicable.

するか否かを QOS として指定可能とする.

さらに, MAC サービスに優先クラスがある場合に備えて, HTP にも QOS として優先度を設け, MAC での優先度にマッピングする.

(2) 高速転送サービスプリミティブ

サービスプリミティブは, OSI のトランスポートサービスプリミティブを基本とする. 追加するパラメータとしては, 前述のとおり, コネクション確立フェーズでの最大 SDU 長と, 解放フェーズでの切断モードとがあるが, これらのパラメータはオプションとする. これらのパラメータを使用しない場合, トランスポートサービスプリミティブと等価になり, 上位層に OSI プロトコルも利用可能となる.

2.6 PDUの構成とヘッダ削減・層間インタフェース削減の効果

(1) PDU の構成

HTP の PDU の構成は, 16 オクテットの固定長のヘッダを基本とし, チェックサムのフィールド位置は PDU の種類によらず一定である. さらに, データ転送関連の PDU は, CL 型でのアドレスフィールドの存在を除いて, CO 型 CL 型で共通である.

PDU の構成の例として, CO 型のデータ PDU の構成を図 2 に示す.

(2) ヘッド削減の効果

ヘッダ項目数とヘッダ長に関して, HTP と OSI プロトコルとを比較してみる. OSI プロトコルとしては 2.3 節(2)で述べた標準的なプロファイルとし, データ転送時のみ比較対象とする.

HTP のデータ転送時のヘッダ項目は図 2 から 10 項目 (Bit field には 2 項目ある) であり, 長さは 16 オクテットである. 一方, OSI プロトコルの場合は 23 項目, 60 オクテット程度となる. したがって, ヘッダ項目数は約 1/2, ヘッダ長は約 1/4 となる.

(3) 層間インタフェース削減の効果

階層モデルに従ったプロトコルを実装する場合, 各層のプロトコルの独立性を維持するため, 各層は独立に実装され, 明確な層間インタフェースが定義される. 層間では, 発生したイベントの通知とともに通常はデータのコピーが行われるため, 層縮退によるインタフェースの削減の効果は大きい.

例として, 8 K オクテットの利用者データを 8 個に分割して 1 K オクテットの PDU として転送する場合を考える. HTP では, 上位とのインタラクション

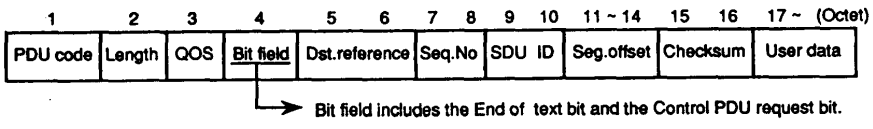


図 2 CO 型 HTP のデータ PDU の構成
Fig. 2 Structure of the data PDU of HTP CO type.

が1回と下位とのインタラクションが8回の計9回となる．これに対して，標準プロファイルでは，通常，分割は再送機能を持ったトランスポート層で行われることから，トランスポート層とネットワーク層の間およびネットワーク層と LLC 副層との間でそれぞれ8回のインタラクションが余計に必要となり，計25回となる．インタラクションが3倍近く増加する分，処理速度は低下する．

3. 評　　価

3.1 プロトコルの検証 (Validation)

プロトコル仕様の机上検証の方法として，到達可能性解析[13),14)]が知られている．到達可能性解析では，初期状態から各種イベント (PDU の送受信など) の発生に伴いどの状態に遷移していくかをツリー構造に展開していき，到達先がない場合や遷移の契機がない場合などを検出していく．HTP は，到達可能性解析により，デッドロック，受信動作未規定 (受信不能状態)，実行不能相互作用などの欠陥がないことを確認した．詳細は付録に示す．

また，実動作に基づく検証のために，**図3**に示す高速転送プロトコル評価プログラムを UNIX ワークステーション (SONY NWS 830) 上に作成した．MAC シミュレータ部分において，PDU の廃棄，遅延，重複，順序入替えなどを発生させることにより，設計どおりの動作をすることを確認した．

3.2 ダイナミックステップによる性能評価

(1) ダイナミックステップの一般式

評価プログラムを動作させ，HTP 処理におけるダイナミックステップ (DS) 数をカウントした．その結果をもとに DS の一般式を導出した．

CO 型データ送信時の DS, DS dt_send は SDU 長と最大 PDU 長の関数として，(a)式で与えられる．同様に，CO 型データ受信時 DS, DS dt_recv は(b)式で与えられる．

DS dt_send(SDU 長，最大 PDU 長)
$=K_0$ dt_send + BUF dt_send(最終 PDU 長)
+CS dt_send(最終 PDU 長) + (Mdt−1)
*(K_1 dt_send + BUF dt_send(最大 PDU 長)
+CS dt_send(最大 PDU 長)) (a)

DS dt_recv(SDU 長，最大 PDU 長)
$=K_0$ dt_recv + BUF dt_recv(最終 PDU 長)
+CS dt_recv(最終 PDU 長) + (Mdt−1)
*(K_1 dt_recv + BUF dt_recv(最大 PDU 長)
+CS dt_recv(最大 PDU 長)) (b)

ここで，K_0 dt_send, K_1 dt_send, K_0 dt_recv, K_1 dt_recv はヘッダ処理に係わる固定分で，300〜900 ステップ程度となる．

分割数 Mdt=ceil(SDU 長/(最大 PDU 長−H))

ceil ()は，() 内の値以上の最小の整数を意味する．H は，データ PDU のヘッダ長で16である．

最終 PDU 長=SDU 長−(Mdt−1)
*(最大 PDU 長−H)+H

バッファコピーに要するダイナミックステップ

BUF dt_send=2*B+2*(ceil((PDU 長−H)/4)
+ceil(PDU 長/4))

BUF dt_recv=B+2*ceil((PDU 長−H)/4)

チェックサムの計算に要するダイナミックステップ

CS dt_send=C_s+7*PDU 長

CS dt_recv=C_r+7*PDU 長

B, C_s, C_r はバッファコピーやチェックサムの計算に係わる固定分で数十ステップ程度である．

これらの式において，バッファコピーとチェックサムの計算に要する分を除く固定分がヘッダ処理に要するステップ数を意味する．この固定分には再送要 (送達保証) CL 型との場合分けのステップも含まれており，その分，若干増加している．

なお，送信時のバッファコピーは，SDU から PDU を作成する際に1回と，PDU を MAC へ渡すフォーマットにする際の1回の計2回行われている．

送達確認やフロー制御のために送受信されるデータ制御 PDU の送受信に要する DS, DS ct_send と DS ct_recv はともに数百ステップであり，その約半分はチェックサムの計算に要するステップである．

CL 型の場合の DS の一般式も同様であり，定数部分が増減するだけである．

(2) DS から計算したスループット

異なるワークステーション上の HTP 処理プロセス

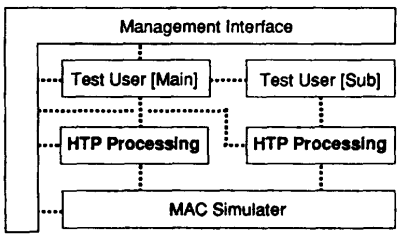


図 3 高速転送プロトコル評価プログラムの構成
Fig. 3 Structure of the HTP evaluation program.

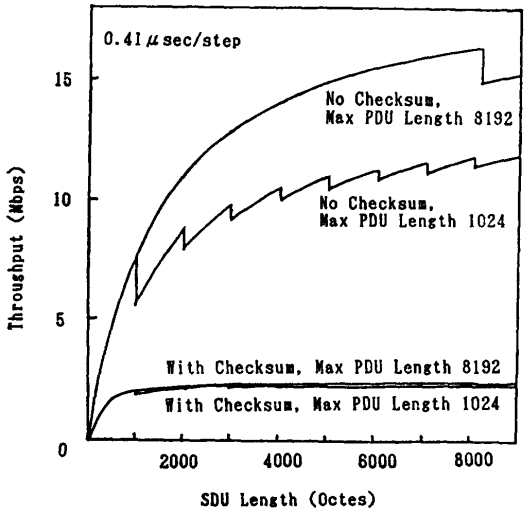


図 4 評価プログラムのダイナミックステップから計算した CO 型 HTP のスループット
Fig. 4 Throughput of HTP CO type calculated from the dynamic steps of the evaluation program.

同士が通信した場合を想定すると，送信側の DS の方が大きいので，送信側がネックとなる．スループットは次式で定義する．

スループット (bps)

＝SDU 長*8/(1 SDU 当りの処理 DS*P).

ここで，P は使用するワークステーションでの1ステップ当りの処理時間を表す．

DS から求めた CO 型のスループットを**図 4** に示す．

CO 型の場合，再送の要否によるスループットの差はほとんどない．CL 型の場合も，図 4 と同様のグラフとなるが，SDU 長の大きい領域で，再送不要の場合は CO 型に比べて 1.06 倍，再送要の場合は 0.83 倍のスループットとなる．

なお，今回の試作では QOS としてのデータを上書きするか否か，および優先度についてはパラメータを運ぶだけとなっているため，これらの QOS 指定による性能的な違いはない．

(3) DS の削減とその効果

DS はプログラムの作りそのものに依存する部分が多い．特に HTP 評価プログラムでは，仕様検証・動作アルゴリズム確認の目的もあったため，性能的な面を考えると不十分な作りとなっている点もある．

SDU から PDU を作成する際に MAC へのインタフェース制御情報分だけずらしてコピーすれば，2回のバッファコピーを1回とすることが可能である．また，バッファコピールーチンを現在のアセンブラからライブラリ (bcopy) に変更すると約4割 DS を削減できることが分かっている．これらによりバッファコピーに要する DS は7割減となる．

さらに，利用者からの要求のパラメータチェックをコンパイルオプションとし，安定化した状況ではチェックなしとすれば，固定分中の約 100 ステップを削減できる．

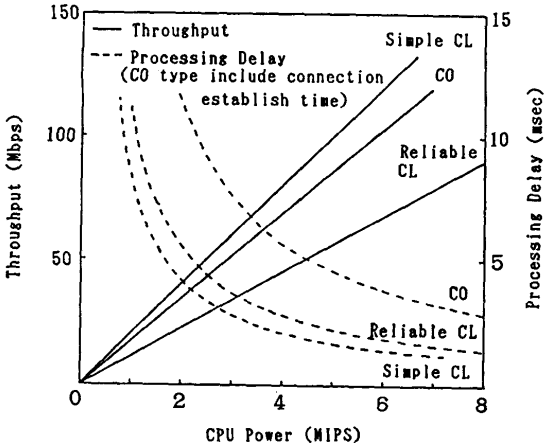
上記の改善を実施した場合，チェックサムなし，かつ SDU 長の大きい領域で，CO 型の場合は 2.5 倍，再送不要 CL 型の場合は 2.7 倍，再送要 CL 型の場合は 2.0 倍高速化できる．

図 5 に CPU の性能と DS 削減後の HTP 処理プロセスでのスループット，処理遅延の関係を示す．図 5 から HTP の処理のみであれば，6 MIPS の CPU を使用することにより 100 Mbps 程度の性能が得られることが分かる．また，遅延に関しては，コネクション確立遅延を除けば，6 MIPS で 2 msec 程度の処理遅延に抑えられることが分かる．

HTP の処理を汎用 CPU, I/O を専用プロセッサで行うフロントエンドプロセッサのボードという形態で HTP を実装すれば，図 5 に近いスループットが得られるものと期待される．

3.3 実環境における HTP と TCP の比較

実際に LAN を介した環境における動作確認・評価のために，図 3 に示した HTP 評価プログラムを改造し，静止画転送を行うプロトタイプシステムを構築した．HTP プロトタイプシステムの構成を**図 6** に示



Conditions : No Checksum, Max PDU Length=8192 octets, SDU Length=8000 octets

図 5 CPU 性能に対する HTP のスループットと処理遅延 (DS 削減後)
Fig. 5 Throughput and processing delay of HTP vs. CPU power (after DS reduced).

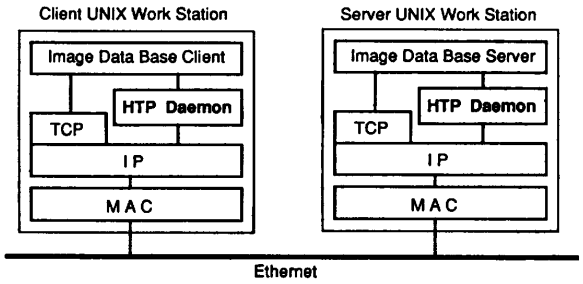


図 6 HTP プロトタイプシステムの構成
Fig. 6 Structure of the HTP prototype system.

す．使用した WS は，前述の UNIX ワークステーションである．HTP 処理は常駐プロセス（デーモン）とした．下位インタフェースは，MAC インタフェースが非公開のため，IP インタフェース（PF_INET の SOCK_RAW）を使用することとした．

プロトタイプシステムのユーザから見たスループットを次のように定義する．

ユーザから見たスループット（bps）
＝（転送された静止画のビット数）/
（静止画の要求から表示完了までの時間）

ここで，静止画の要求から表示完了までの時間とは，クライアント側から静止画の表示を要求するコマンドを送信してから，サーバ側からのデータを受信して表示を完了するまでの時間であり，コマンドの転送時間，静止画データのファイルからの読出し時間，表示のための時間を含んでいる．したがって，実際の HTP や TCP（OSI の TP 4 とほぼ同等の機能を持っている）の性能からはかなり落ちているが，逆に静止画転送を使用するユーザが実感する性能となっている．

TCP に合わせて HTP でもチェックサムを使用することとし，転送する静止画の枚数を 7 枚（約 7 M ビット）とした．ユーザから見たスループットは，HTP を利用した場合 0.75 Mbps となるのに対し，TCP を利用した場合 0.51 Mbps となり，HTP を利用した場合の方が約 1.5 倍大きくなる．この結果をプロトコル自身の性能差という観点から見ると，ソケットインタフェースでのデータ移動のオーバヘッドが大きいことや，TCP は OS レベルで組み込まれているのに対し HTP はユーザプロセスとなっていることなどから，HTP と TCP の性能差としては実際よりも小さく現れているものと考える．

4. おわりに

本論文では，高速 LAN を利用するアプリケーションからの多様な通信要求に応えるために HTP を提案した．HTP は，プロトコルの軽量化のためのアプローチとして LLC 副層からトランスポート層までを一つの層として扱うことにより，重複・不要機能の排除と層間インタフェースの削減を図った．また，効率的な誤り回復のために明示的要求と選択再送方式を採用し，効率的なフロー制御のために最大 SDU 長に基づくバッファ予約とフロー制御の最適化を実現した．最大 SDU 長に基づくフロー制御の最適化は，転送方向別 QOS 制御と同様に，非対称通信のサポートにもなっている．CL 型通信の信頼性向上のためには，CO 型と CL 型の手順や PDU 構成の共通化を図り，送達保証 CL 型通信を実現した．さらに，コネクション解放時の保留モード，送信側からの送達確認要求など，従来の OSI プロトコルにはない手順を実現して高機能化を図った．

以上のような高機能化にもかかわらず，層の縮退を行っているため，HTP のプロトコルヘッダは OSI の標準的なプロファイルのヘッダと比較してもヘッダ項目数で約 1/2，ヘッダ長で約 1/4 に抑えられる．

設計上の欠陥がないことを確認するために，到達可能性解析による机上検証のほか，ワークステーション上に評価プログラムを作成して動作検証も行った．また，評価プログラムから求めたダイナミックステップを分析し，HTP の処理のみに着目すると 6 MIPS 程度の CPU を使用することによってスループット 100 Mbps，処理遅延 2 msec の性能を得られることを示した．さらに，実際に LAN を介した環境で静止画転送を行った場合，ユーザから見たスループットは，HTP を利用すると TCP を利用した場合の 1.5 倍になることを述べた．

今後の課題としては，信頼性のある同報機能をいかに実現するかという点が挙げられる．現在の HTP においてもある程度は実現できるが，実用的にするためには送達確認 PDU の集中を回避するための機構が必要となる．また，プロトコル処理の面では，プロトコルそのものの性能が引き出せるように，バッファ管理やタイマ管理の工夫，チェックサム等の一部処理をハード化した実装等の検討を進める必要がある．

謝辞 本研究を進めるに際し有益なご意見をいただいた通信網総合研究所総合網研究部の山縣部長，木下

主幹研究員，斎藤主幹研究員に深謝いたします．また，プロトタイプシステムの構築にご協力いただいた荻原研究主任，池川研究主任に深謝いたします．

参考文献

1) Strauss, P.: OSI Throughput Performance: Breakthrough or Bottleneck?, *Data Communication*, May 1987, p. 53 (1987).
2) 横山，松井，平田，水谷，寺田：通信プロトコル高速処理プロセッサの方式提案，第41回情報処理学会全国大会論文集，4Q-7 (1990).
3) 谷，前原，明石：高速通信プロトコル，信学技報，SSE 88-72, p. 47 (1988).
4) Chesson, G. and Green, L.: XTP-Protocol Engine VLSI for Real-Time LANs, *Proceedings of EFOC/LAN 88*, p. 435 (1988).
5) Cheriton, D. R.: VMTP: A Transport Protocol for the Next Generation of Communication Systems, *Proceedings of SIGCOMM '86 ACM*, p. 406 (1986).
6) Rudin, H.: *Tutorials Part A: High-Performance, Transport-Level Protocols, SIGCOMM '90* (1990).
7) Doeringer, W. A.: *High Speed Protocols Tutorial #1, INFOCOM '90* (1990).
8) Green, L.: X 3 S 3. 3 HSP Activity, *Transfer*, Vol. 3, No. 3, p. 6 (1990).
9) Cohn, M.: A Lightweight Transfer Protocol for the U. S. Navy SAFENET Local Area Network Standard, *Proceedings of 13th Conference on Local Computer Networks*, p. 151 (1988).
10) 本村，坂口：高速転送プロトコル (HTP) の検討，第38回情報処理学会全国大会論文集，3 H-5, p. 1617 (1989).
11) 本村，坂口：高度転送プロトコルとその評価，信学技報，IN 90-23, p. 61 (1990).
12) ISO/IEC DISP 10608: International Standardized Profile TA-nnnn—Connection-mode Transport Service over connectionless-mode Network Service (1990).
13) Zafiropuloet, P. et al.: Towards Analyzing and Synthesizing Protocols, *IEEE Trans. Commun.*, Vol. 28, No. 4, p. 651 (1980).
14) 白鳥，郷原，野口：EXPA：パータベーション解析に基づく通信プロトコルの検証法，情報処理学会論文誌，Vol. 26, No. 3, pp. 446-453 (1985).

付録 到達可能性解析による HTP の検証

プロトコル検証手法の一つとして知られている到達可能性解析を HTP に適用した結果について述べる．ここでは，状態遷移の複雑な CO 型の場合についてのみ述べることとし，CL 型の場合は省略する．CL 型の場合にはコネクション確立/解放フェーズがなく，ウインドウの上限という概念もないため，CO 型よりもはるかに簡単に検証できる．

到達可能性解析では，プロトコルを，協調動作している複数プロセスからなるシステムにおけるプロセス間のメッセージ交換の規則と捉える．各プロセスは，FIFO のチャネルを介してメッセージを交換するものとする．検出できる欠陥は以下の四つである．

状態デッドロック (State Deadlocks)

すべてのチャネルが空で，かつ自律的遷移がないシステム状態．特別に意図された最終状態以外の場合は，誤りである．

受信動作未規定 (Unspecified Reception)

チャネルの中からメッセージを取り出して遷移する先の状態がないシステム状態．

実行不能相互作用 (Nonexecutable Interaction)

設計上はあるが，到達可能性ツリーには現れない状態．正常動作条件の元で実行されるのを想定して設計されていた場合は誤り，リカバリ動作として設計されていた場合には誤りではない．両者を区別するために，リカバリ動作を加える前に正常動作の検証をする．

あいまい状態 (State Ambiguities)

すべてのチャネルが空であるシステム状態（安定状態対）の中に，複数現れるプロセス状態．誤りかどう

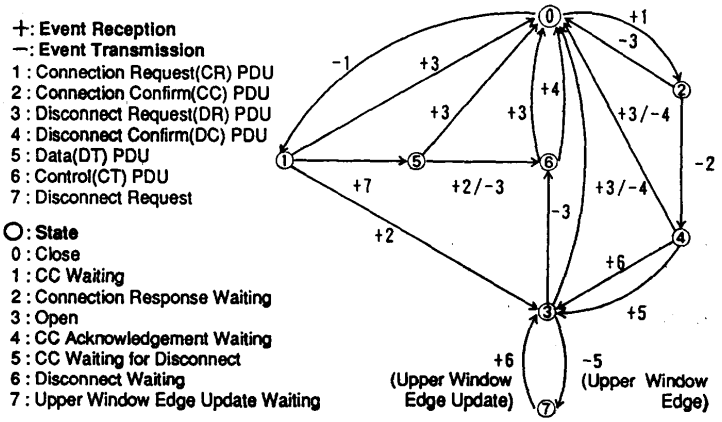


図 7 エンティティ間の相互作用に着目した状態遷移図
Fig. 7 State diagram of HTP CO type for two-entities interaction.

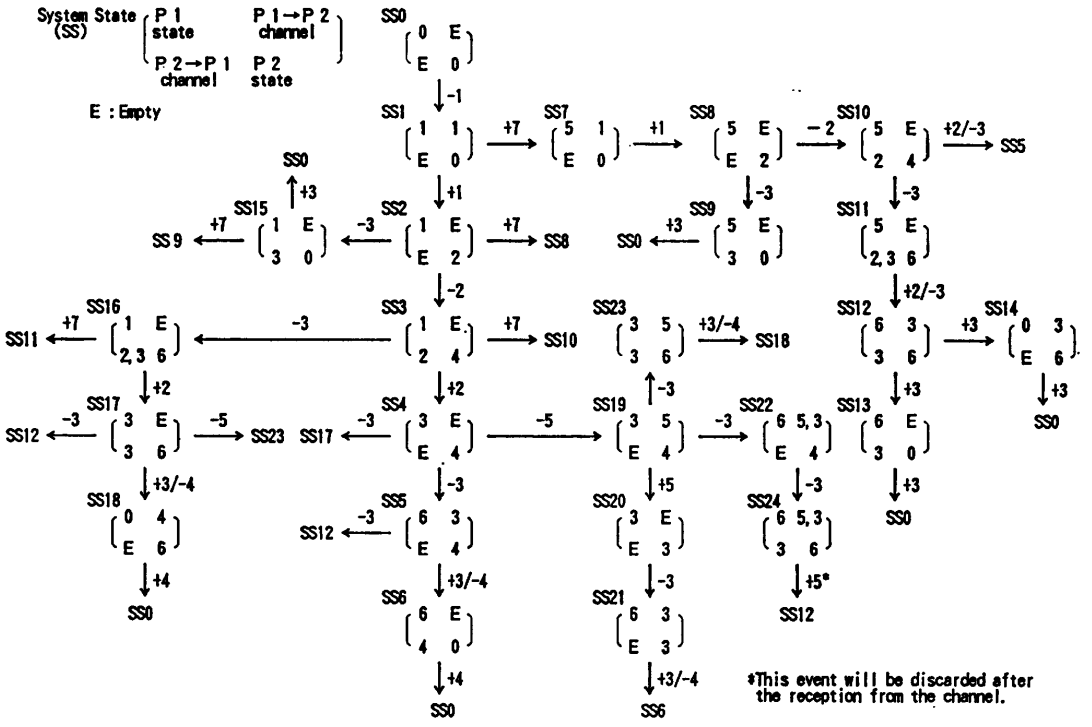


図 8(a) CO 型 HTP の到達可能性ツリー (コネクション確立/解放フェーズ)
Fig. 8(a) Reachability tree of HTP CO type (connection establishment/release phase).

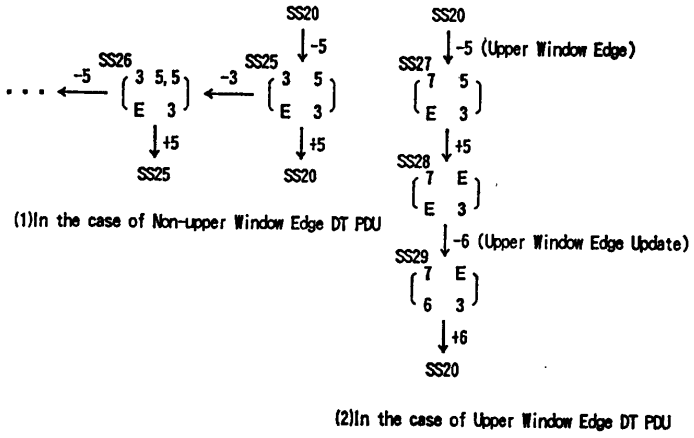


図 8(b) CO 型 HTP の到達可能性ツリー (データ転送フェーズ)
Fig. 8(b) Reachability tree of HTP CO type (data transfer phase).

かは，意味的に判断する．

図7にエンティティ間の相互作用に着目した状態遷移図を示す．図7では，PDU の交換によるエンティティ間の相互作用に関係のないメッセージ(イベント)として，サービスプリミティブ，タイムアウト，リトライアウトは省略している．CC 待ち状態からコネクション解放のための CC 待ち状態への遷移は PDU とは無関係に起こるが，その後の PDU 受信時の動作が異なるため，Disconnect Request プリミティブだけは残している．また，実行不能相互作用を正しく検出するためにリカバリ動作に相当する Error PDU の送受信は含めていない．

図7の状態遷移図から発呼側 (P 1)，着呼側 (P 2) それぞれの有限状態グラフを作成し，コネクション確立/解放フェーズのシステム状態を展開した (図8(a)参照)．図8(a)では，SS4 から発呼側が DT を送信

することによりP2がCC確認待ちからオープン状態へ遷移する場合について展開されているが，SS4から発呼側がCTを出す場合も全く同様に展開できる．データ転送フェーズにおける到達可能性ツリーの展開は，SS20を初期状態とみなして展開できる（図8(b)参照）．データ転送は転送方向ごとに独立であるため，片方向の到達可能性ツリーで十分であり，ここでは発呼側から着呼側に転送する場合について展開している．データ転送時の到達可能性ツリーはDTの連続的送信によってどんどん展開されていくが，ウインドウ上限（Upper Window Edge）のDT送信時とウインドウ上限を更新するCT受信時以外は，エンティティの状態遷移は起こらない．

図8に展開された到達可能性ツリーから，CO型HTPについては次のように検証できる．

- すべてのチャネルが空で，かつ自律的遷移がないシステム状態は存在しないので，状態デッドロックはない．
- チャネルの中からメッセージを取り出して遷移する先の状態がないシステム状態は存在しないので，受信動作未規定はない．
- 到達可能性ツリーに現れないプロセス状態は存在しないので，実行不能相互作用はない．
- 二つのチャネルが空になる安定状態対は，SS0, SS2, SS4, SS8, SS20, SS28の六つであり，あいまい状態は，発呼側と着呼側の状態3，および着呼側の状態2である．しかしながら，これらのあいまい状態は，CCの送達確認のために必要であったもの等，プロトコルの正常動作の規定上意図されていたものであるため問題はない．

（平成2年12月20日受付）
（平成3年7月8日採録）

本村 公太（正会員）

昭和56年大阪府立大学工学部電子卒業．昭和58年同大学院博士前期課程修了．同年日本電信電話公社（現日本電信電話(株)）横須賀電気通信研究所入所．以来，電子メールシステムの研究開発，高速データ通信用プロトコルの研究に従事．現在，同社通信網総合研究所ネットワークインテグレーション研究部主任研究員．SC 6/WG 2国内小委員会委員．CL型ルーチング交換プロトコルJIS原案作成委員会委員長．

坂口 勝章（正会員）

昭和46年熊本大学工学部電子卒業．昭和48年同大学院修士課程修了．同年日本電信電話公社（現日本電信電話(株)）横須賀電気通信研究所入所．以来，テレマティークプロトコル，データ通信用プロトコルの研究に従事．現在，同社通信網総合研究所ネットワークインテグレーション研究部主任研究員．SC 6/WG 4国内小委員会委員．電子情報通信学会会員．