計算力を持つネットワークのアーキテクチャ

宇夫 陽次朗 †　　篠田 陽一‡

北陸先端科学技術大学院大学 情報科学研究科§

概 要

インターネットの利用量の増大や通信の品質の向上の要求を解決するために、インターネットにおいて従来通信の端点からは制御不能であったネットワーク部の操作を可能とする新規ネットワークアーキテクチャのアイデアについて説明を行う。

ネットワークのノード部分に着目し、ストリームに対する情報処理能力を付加することで、ネットワーク特性の操作およびネットワークサービスの動的な生成を行うことを目指す。そのための、情報処理能力を提供する機構として、NNP(Network Node Processor)の設計を行った。また、情報処理エレメントの一実装法として NPE(Network Processing Element) について述べる。

1　はじめに

近年、インターネットは急激にその規模を拡大し、名実共に世界的な大規模ネットワークとなった。その発展に伴ってネットワーク自体の性格も変化している。従来は、研究ネットワークの名のもとに、接続性の保証を唯一の必要条件としており、情報の伝達機能を中心として利用されるネットワークとして構築していたが、多種多様のユーザの通信需要を包含するようなることで、それぞれの要求をサポートする付加価値を提供する機能を導入することが望まれている。すなわち、リアルタイム通信・帯域予約技術の導入や、セキュリティ的に強固な通信路の確立など、既存のインターネットでは利用できなかった技術が今後実際に利用されるようになると考えられる。

このような中で、インターネットのネットワークサービスのための機能を如何に拡大するかということを検討する必要がある。インターネットのネットワークアーキテクチャは端点同士アソシエーションを提供する際にも複数のトランジットノードを経由して通信を行っている。その場合、通信に関わる要素は複数存在するが、明示的に利用できるのは、お互いに通信を行う端点のみであり、他の通過ノードは単なる通信路として扱われる。そのため、端点の通信主体が通過ノードを含むネットワーク部でのなんらかの操作を行うことは不可能である。

このような制約があるため、従来のインターネットではネットワークサービスに対する拡張を行う際には、ネットワーク部に対して本質的な変更を加えずに端点同士での制御を行うことで対応するか、その機能をサポートする新しい機構をネットワーク部に組み込むことで解決を図っている。しかし、前者の方法による解決では対応できないことが存在し、また、後者による対応では、利用技術の標準化を行うなどして、あらかじめ経路上で利用可能でなければならないため非常にコストが高いという問題点が指摘される。このような要求に対する本質的な解として、ネットワーク全体を変更する無しに、新規技術を導入するための機能自体ををネットワークサービス

---

*An Archtecture of Distributed Processing Network

†Youjiro UO

‡Youichi Shinoda

§School of Info. Science,Japan Advanced Institute of Science and Techonology

として供給することを考える。そのためには、インターネットアーキテクチャを改良し、従来静的に決定されていたアーキテクチャ部分において自由度を持たせることが必要である。

本研究では、インターネットのネットワーク部に対してその部分を通過するネットワークストリームへの処理を行うモジュールを挿入する機構について考察を行う。従来不可能であった領域の制御を行う要素の導入により、ネットワーク全体のポテンシャルを高めることができる。ネットワーク上に存在するサービスとして情報処理能力を提供するサービスすなわちネットワークストリーム自体を処理する要素を導入し、経路上を流れるネットワークストリームをその要素にディスパッチする機構を通過ノードに導入することで前述の目的を達成する。

2 ネットワークアーキテクチャ

本研究で提案するネットワークモデルは、基本的には現在のインターネットで利用されている構成と同一である。すなわち、インターネットはルータ (もしくはゲートウェイ) を利用してネットワーク同士を相互接続することでネットワークを構成している。インターネットで基本的に利用されているプロトコルである。IP(Internet Protocol) を利用して通信を行う際には、発信側は相手先を識別するアドレスを付加した情報をネットワークに投入するだけであり、あとは通過ノードにある『ルータ』が配送制御を行うことで相手先への通信を確保する。

2.1 通信モデル

インターネットにおける通信経路は通過ノードの集合体として定義されている。このようなネットワークモデルにおいて、ネットワークストリームを操作するための方法として経由する通過ノードに変更を加えることを考える。

従来は通過ノードはルーティングを行う『ルータ』から構成されていたが、それを計算能力

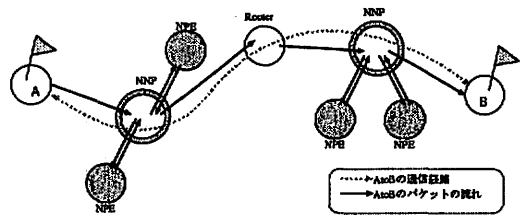

図 1: 通信モデル

をサポートする要素に置き換える。ここでいう計算能力とはネットワーク中を流れる任意の情報に対する処理能力全般をさす。つまり、現在ネットワーク上で提供されているサービスすべてが対象となる。

ネットワーク自体のパフォーマンスを考慮すると、定常的に通過する部分に高い付加をもたらす可能性のある要素を組み込むことはできないためネットワークストリームに対する処理は、通過ノードでは行われずに計算サービスを提供する要素で行われる。概念図を図 1 に示す。

概念的には、実際の処理を行うネットワーク上の要素に対して送り込む要素と、実際の処理を受け持つ要素に 2 分されることになる。前者を Network Node Processor (以下 NNP)、後者を Network Processing Element (以下 NPE) と呼ぶ。NNP は既存のルータを内包する概念であり、複合機能を持つルータと考えることができるが、混乱を避けるためにこのように呼ぶ。

また、計算リソースの必要量はネットワークの規模や通過ノードの処理流量などで変化するため、ネットワーク中に存在する NPE の数は変化する。そのため、NNP はパケットのディスパッチだけでなく、NPE の管理を行う。

2.2 経路上の処理

ネットワークの経路上で処理を行う場合をレイヤモデルに則った形で説明する。インターネット 5 層モデルを利用した図を図 2に示す。ネットワーク内で行う処理によって目的とす

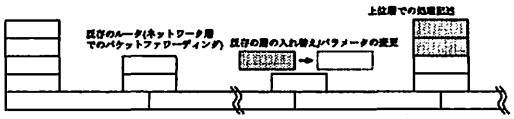

図 2: 経路上での処理

ることは、特定のネットワークストリームに対するネットワークの適合性の向上である。途中ノードを通過するネットワークストリームは1系統だけではなく多数である。よって、それぞれのネットワークストリームに対して独立な記述および処理をおこなわなければならない。

ネットワークストリームに対する処理として、ネットワークの経路上で行う基本操作は、

- 既存プロトコルの差し換え/部分的変更
- プロトコル変換スタックの導入
- 新規プロトコルスタックの導入
- アプリケーション層へのプログラム導入

である。

一般的に既存のルータはネットワーク層 (第3層) 場合によってはデータリンク層 (第2層) によるパケットフォワーディングを行っている。つまり第3層までがルータの守備範囲であったが、NNP はNPE を伴うことで第5層までの記述を許容する。TCP 以上のコネクションオリエンテッドなアプリケーションやパケットリレーサーバなどもここに含まれる。また、既存のルータが行っていた部分である第2層・第3層についての記述を行うことで、ネットワークの挙動制御を行うことも考える。ルーティングアルゴリズムの変更やIP 層のパラメータ決定戦略アルゴリズムの入れ替えなどの処理がこれにあたる。

ネットワークストリームをパケット単位ではなくストリームとして扱うことで、単純なパケットフィルタの実現から、上位層のプロトコルスタックを理解するモジュールを利用したアプリケーション層でのデータ処理を行うコンポーネントの記述が可能である。

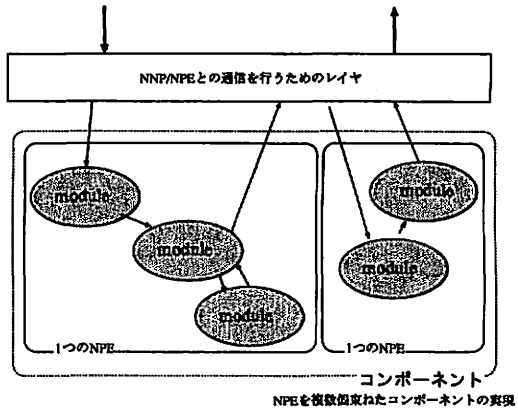

図 3: モジュールとコンポーネント

NPE が提供するサービスとして、処理ストリームとは独立したコネクションを張ることを考える。このことにより、ネットワークの経路上で得られた情報を副作用とするようなストリームに対する処理を記述することが可能である。

2.3 モジュールとコンポーネント

ネットワークストリームに対する処理を記述する要素がモジュールである。モジュールは外部から NPE に挿入されるか組み込みオブジェクトとして定義される。1つもしくは複数のモジュールを組み合わせることで、処理を記述する。1つの処理を記述するモジュール群を『コンポーネント』と呼ぶ。モジュールとコンポーネントの関係を図 3に示す。

モジュールの記述単位としては、IP, TCP, UDP などのプロトコル記述や、バッファリングや名前解決などのネットワークサービス、第5層に挿入されるアプリケーションモジュールがある。NPE のマシンアーキテクチャを規定することは出来ないため、モジュールはアーキテクチャ独立の記述言語によって記述される。これらのモジュールからコンポーネントを作成しサービスの提供を行う際のパフォーマンス向上のために動的コンパイルによる実

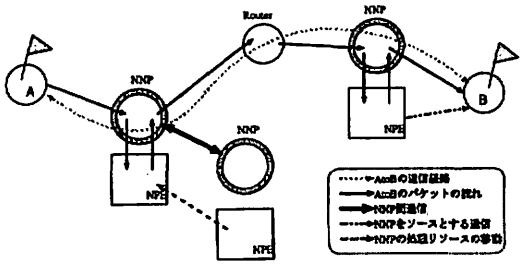

図 4: ネットワーク構成

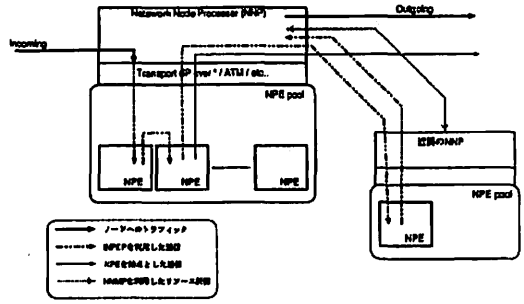

図 5: NNP と NPE の関係

行時最適化を行う。モジュール記述言語の仕様については、現在検討中である。

2.4 ノード構成

ノードを構成する要素である NNP と NPE に関係について述べる。

ネットワーク上に任意の場所から NNP 及び NPE に対してモジュールを挿入しコンポーネントを生成する機構を導入することで、特定のネットワークストリームに対して操作を行う。全体像を図 4に示す。

通常は、NNP を通過するトラフィックに対して NNP は既存のルータと同様の処理を行う。ある特定のストリームを識別すると、あらかじめ設定してあった NPE への経路にパケットをフォワードする。NNP での処理トラフィックの識別トリガにはフローラベルを利用する。NPE 群が処理を行った情報はまたもとの経路に戻されて NNP から配送される。

ネットワーク上に存在する NPE はかならず特定の NNP に帰属し、NNP を通じてネットワーク中にサービスを提供する。そのため、NNP は NPE を自身のリソースとして扱い、処理モジュールの挿入や管理は NNP を通じて行われる。NPE と NNP の関係を図 5に示す。

ここで、通信を行うホストと NNP、NNP と NPE 間の通信をサポートするプロトコル群についての定義を行う。

INPEP Inter NPE Protocol
NNP–NPE 間または NPE–NPE 間の通信に利用されるプロトコルである。NPE での処理を行うためのパケットのカプセル化およびパケット毎の処理指定を行う。

NPEMP NPE Management Protocol
NNP–NPE 間または NPE–NPE 間の通信に利用されるプロトコルである。NNP が NPE 群を管理運用するため利用する。

NNMP Network Node Manegement Protocol
通信を行うホストと NNP 間および NNP 間での協調を行うためのプロトコルである。NNP に対する処理のネゴシエーションを行ったり、各 NNP のリソースコントロールや状態制御を行う。ストリームに対するモジュールパスの設定、NPE に対するモジュールの挿入要求やモジュールの転送なども NNMP を通じて行われる。

NNPD NNP Discovery
経路上での処理を行うことを考えたとき、ネットワーク上の任意の点からリソースの提供が可能な NNP を捜す必要がある。そのための機構として (NNPD) を設計する。NNPD は ICMP のメッセージを拡張することで実現する。NTP リソースを要求するホストが要求条件を示したパケットを送出し、その条件を満たす NNP がリソース予約 ID を返す。予約 ID を得たあとは NNMP でのセッションに移行

する。

1つのNPEが提供する計算リソースは有限であるため、必要であればNNP間でリソースの移転操作を行う。NNP間で計算リソースの移動を行うことで、ネットワーク内に存在する計算リソースを有効に利用できる。NNP間での協調を行うためのプロトコルとしてNNMPを利用する。NPEの移転操作はNNPを経由して行うため遠隔地のNPEを扱うことができる。

2.5 通信シーケンス

あるストリームの処理の指示をネットワークに対して行うシーケンスを説明する。一般的には下記のようなシーケンスを通じてネットワーク側にサービスを登録する。

1. 必要であれば経路予約を行う。
2. NNPDを利用して利用可能なNNPの探索を行う
3. NNPDで得られた利用可能NNPのリストから、条件にあうものを選択する。
4. NNPとNNMPを利用して通信を行う。

   トリガー条件を設定し、必要であれば、NNPに対して処理モジュールを挿入する。

   NNPはNPEMPを利用して、NPE群の設定を行う。
5. 設定が完了する。

登録されたサービスはNNPごとのポリシーによって維持される。またNNPによっては永続的に存在するサービスもあるため複雑なシーケンスを実行しないこともある。

2.6 経路の揺らぎ

現在のインターネットではネットワークストリームを構成するパケットが通過する経路は一意に定まらないため、途中経路での処理

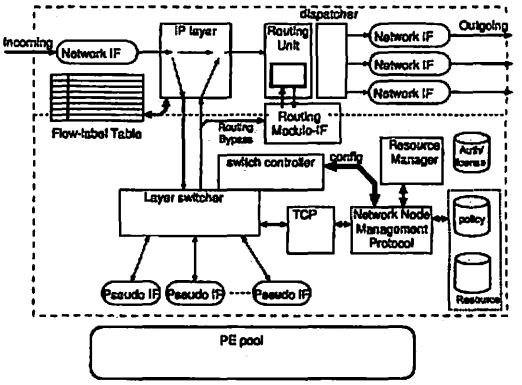

図 6: NNPの構成

を保証することはできない。すなわち、ネットワークストリームを構成するすべてのパケットに対して本方式を適用するためには特定経路の保証が必要である。経路保証を要求する様な経路上の処理を行う際には経路保証技術との併用を行う。インターネットにおける経路保証は、インターネットの特徴としてあげられる系としての冗長性がもたらす頑健性を損なう可能性がある。経路保証技術はネットワークのQoSファクタであり将来的には必要な技術だが、ここではスコープとはしない。現在のインターネットでは適当な経路保証技術は存在しないため、しばらくはNNPを含む経路をLSR (Loose Source Routing) を利用して選択することにする。

3 構成要素詳細

本ネットワークアーキテクチャを構成する要素であるNNPとNPEの詳細について説明する。

3.1 NNP

NNPの内部構成について説明を行う。NNPの構成を図6に示す。

NNPはNPEを含んだリソース全体の管理を行う。そのため、それぞれの管理情報を保持してる。管理情報の種類として

- 挿入されたモジュールの外部安全性を保証するためのもの
- NNP がもつリソース (NPE の数やそれぞれのロード、現在挿入されているモジュールのうち公開されているもののリストなど)
- リソース割り当てをおこなう際のポリシ

などがある。通常、NNP は NNPD でリソース問い合わせを受けたとき、リソースを提供できるかの判断を行うためにポリシ/ライセンスの適用を行う。ポリシは各 NNP に特有な情報であり NNP の管理者が設定する。

処理のトリガとしては IP の flow label を利用するのが妥当であるため、flow label を利用している場合の説明を行う。NNP に入力されたパケットは IP の flow label を利用してストリームとして認識される。IP 層の処理において、flow-label Table を参照し、もしそのストリームが処理を行うことが要請されているストリームであると認識された場合は、IP 層で NPE へのバイパス経路へスイッチする。

NPE への経路は、トラフィックが流れているネットワークに相乗りするだけでなく、専用のインターフェイスを利用することも考慮する必要がある。そのため、NPE 毎に仮想インターフェイス (PIF) を設定し、物理インターフェイスおよび物理層プロトコルから独立した層を設ける。その層が Layer Swithcer である。Layer Swithcer はストリームに応じた NPE 群の経路を設定し、処理を行う NPE 用の PIF に対してパケットのディスパッチを行う

PIF では NNP と NPE の物理インターフェイスの種類に応じて、パケットを INPEP でカプセル化を行い、物理インターフェイスを通じて送信する。

3.1.1 ルーティングモジュール

ルーティングに関係する操作を行うモジュールは他のモジュールとは性格が異なる。ルーティングは通信全体に密接に関係してため、容易にモジュール化することが不可能だからである。ルーティングを操作するモジュールは NNP に組み込まれているルーティングユニットから情報を得る必要があるが、実際のモジュールが組み込まれる部分は NPE なので、NPE 内のモジュールはそのままの構成ではそれらの情報を利用することができない。

そのため、ルーティングを行うモジュールのために『Routing Module-IF』を NNP に導入する。これは、ルーティングモジュールの一部の機能をこの部分で実行するもので、NPE 内のルーティングモジュール本体と通信を行うことで、ストリームごとのルーティングポリシ変更を実現する。

3.2 NPE

NPE は NNP となんらかの手段で接続されており、実際に NNP から渡されるパケットに対する処理を行う。NPEs(複数の NPE) は NNP を通して通信することで、協調動作を行う。同一セグメント上に配置されているなど、物理的に可能なら NPE 間の直接通信もサポートする。

モジュールには、NPE に標準的に含まれるモジュールと挿入モジュールが存在する。NPE は NPEMP を通じて複数のモジュールを挿入され、それらから生成されるコンポーネントによってストリームの処理を行う。既存モジュールには各種標準的プロトコル (IP スタック、TCP スタック) などの良く利用されるものが含まれており、それに必要なモジュールを合成してストリームの処理パスを動的に生成する。NPE から外部への通信をサポートする組込みモジュールが提供される。

NPE に要求される機能は、それらのオブジェクトからストリームごとに独立したコンポーネントを効率的に作成することである。また、モジュールについての付加情報の管理もある程度おこなう必要がある。

NPE の実現方法は多数存在しするため、本研究で提案するシステムでは具体的な NPE の実装手法についての限定は行わない。

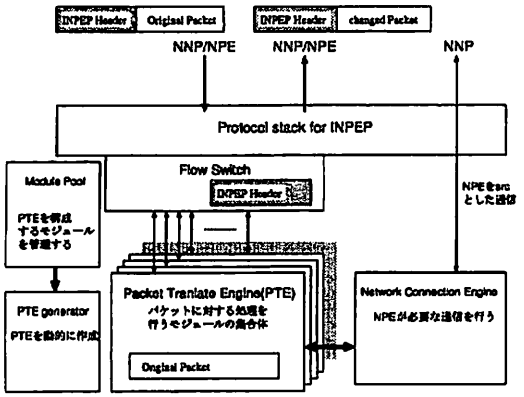

図 7: NPE の構成

INPEP および上記要求仕様のみを規定するものとする。

NPE の構成例を図 7に表す。

現在は、NPE を構成するためのオペレーティングシステムの検討を行っている最中である。

4 まとめ

従来、インターネットアーキテクチャでは通信経路上のノードを利用して、ネットワークストリームに対して処理を行うことは非常に困難であった。途中経路を操作する機構が存在しないため、ノードを明示的に処理要素として扱うことが不可能であったため、なんらかの処理を行うためには、あらかじめ通過ノードに対して、処理エレメントを挿入しておかなければならかったからである。

本研究では、経路上に存在するノードについて着目し、トラフィックに対する処理を途中経路で行うことのできるネットワークアーキテクチャを提案する。

アーキテクチャを実現するための要素として、ルーティング機能を内包した要素である NNP の設計を行った。既存インターネットではルータが存在していた部分を NNP で置換することで、ネットワーク上の任意の場所を限定された形で変化させることが可能となり従来のインターネットの自由度を向上させる。

通信経路上の NNP を設定することでネットワークストリームに対する処理を行うことができるため、既存のアプリケーション資産を有効に活用できる。

5 今後の課題

現段階ではネットワークアーキテクチャ全体を俯瞰しながら、詳細を詰めている最中である。

今後の課題としては、アーキテクチャ全体の検証を行う予定である。今年度中に、

- **各種プロトコルの設計**
  INPEP・NNMP・NPEMP のプロトコル詳細の設計を行う。また ICMP の拡張を行い NNPD を設計する。
- **NNP の実装**
  BSD 系の UNIX を利用してカーネルレベルで (一部実装はユーザ空間に置く) の拡張を行うことで、NNP 機能を実装する。
- **NPE の実装**
  疑似的に NPE の役目を果たすユーザプロセスを作成し、NNP のサンプル実装上で稼働させる。

し、評価を行うことを予定している。

さらに、モジュールの記述言語及びダイナミックコンパイルによる動的最適化について研究を行う。

参考文献

[1] J.M.Smith, D.J Faber, C.A Gunter, S.M.Nettles, D.C.Feldmeier and W.D.Sincokie "Switch-Ware:Accelerating Network Evolution"

[2] D.L Tennenhouse, D.J. Wetherall, "Towards an ACtive Network Architecture"