第Ⅴ部

ラベルスイッチ技術によるインターネット構築実験

第5部
ラベルスイッチ技術によるインターネット構築実験

第1章 概要

本報告書では、LAST (LAbel Swich Technology) WG での活動報告を行う。LAST WG では、MPLS (MultiProtocol Label Swiching) 技術の検討を行っている。本年度の活動では、大きく分けて 2 つのトピックがある。1 つは、MPLS を用いたマルチキャストの検討、もう 1 つは、MPLS の NetBSD 上への実装である。これらを順番に報告する。

第2章 MPLS を用いたマルチキャストに関する検討

MPLS 技術は既に様々な形でインターネット内で利用されつつあるが、MPLS 技術はいまだ研究段階の要素が強く、2000 年 4 月現在、IETF での議論の途中であり、今後様々な仕様の変更、発展が予想される。現時点では、主に MPLS におけるユニキャスト通信に関する仕様が議論の中心であり、MPLS のマルチキャスト対応化に関する議論は今だ理論的ものが多い。これらの議論は主に Internet-Draft[1] を通じて追うことができる。

MPLS および IP マルチキャストはそれぞれ第 3 層に深く依存しているうえ、それぞれ多くの要素技術によって複雑に構成されている。さらにマルチキャストはユニキャストとは本質的な部分で異なる技術であり、現在の MPLS の仕様をマルチキャストに対応させるためには現在の MPLS に対する多くの拡張が必要になる。そのため MPLS の IP マルチキャストへの対応化に関する議論や提案は非常に多岐にわたり、議論や提案の全体像を把握することが困難な状況となっている。

そこで、本章では MPLS のマルチキャスト対応化に関する議論や提案を調査し、MPLS マルチキャストに関する要求や今後の課題をまとめることで、本分野の研究の課題を明確化することを目的としている。

2.1 ラベルスイッチ技術

インターネットは IP (Internet Protocol) によるノード間のパケット伝送を基本としたネットワークである。それぞれのノード (ルータなど) が受け取ったパケットを次中継点へ中継することで端点間 (end-to-end) の接続が実現される。IP はインターネットのレイヤ構造的には第 3 層に位置しており、IP パケットを始点から終点へ到達させるための経路を解決する経路制御機構も第 3 層に置かれている。経路制御機構による次中継点の解決は各中継点が自律的におこなうため、中継点毎で

- 第 3 層でのパケット解析
- 次中継点決定のための経路検索処理

が必要である。

ラベルスイッチ技術は、中継点毎におこなわれているこれらの処理を、入口ノード (ingress node) で集約する技術である。入口ノードで、パケットの第 3 層情報を解析し、その解析結果を短い固定長の**ラベル**に割り当てる。このラベルと第 3 層情報の対応関係をラベルスイッチ雲内に伝播させることで、中間ノードでの第 3 層情報を解析を省略できる。ルータでの処理を軽減することができるため、高速化技術として注目されてきた。既存のパケットの伝達とラベルスイッチでのパケットの伝達の違いを図 2.1 に示す。

また、ラベルという別の情報での経路制御をおこなうため、第 3 層での経路制御とは別の形態の経路制御を実現できる可能性が高い。そのため、ネットワーク内のトラフィックを高度に制御するためのトラフィックエンジニアリング技術やポリシ経路制御などを実現するための要素技術としての位置づけを

[1] http://www.ietf.org/ID.html

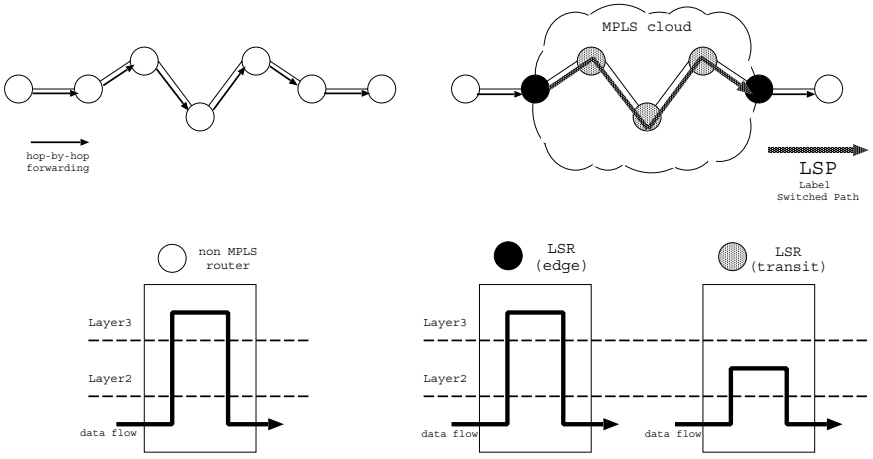

図 2.1.　既存伝送とラベルスイッチ伝送

確保しつつある。

インターネットの技術標準策定機関である IETF では、ラベルスイッチ技術の初期段階で提案されてきた様々なラベルスイッチ技術で得られた経験を集約し、ラベルスイッチ技術の標準化仕様として MPLS (Multi Protocol Label Switching) 技術を策定中である。以下、本文書ではラベルスイッチ技術として MPLS を扱う。

2.2 MPLS 技術

IETF における標準化がおこなれている MPLS(Multi Protocol Label Switching) 技術の基本アーキテクチャについて概説する。

2.2.1 MPLS 技術の特徴

一般にパケットの転送処理は、そのパケットの転送特性である FEC に基づいておこなわれる。MPLS では、FEC の認識は MPLS ドメインの入口 LSR のみでおこなわれ、その情報はラベルとしてパケットに付加される。ラベルと FEC の束縛情報 (FEC/ラベルマッピング情報) は、ラベル配布プロトコルによって生成/広告される。各中継 LSR はラベルのみを用いて、単純な『ラベル置換 (label swapping) 動作』のみを繰り返すだけである。パケットは、最終点に到達するか MPLS ドメインの出口から非 MPLS ドメインへ転送されるまだラベル置換操作のみによって転送される。

この特性はすなわち、

- 転送は本質的に第 3 層経路ドメインとは無関係である

ということを意味する。つまり、MPLS ドメインの経路制御技術は既存のインターネットのように第 3 層経路制御技術の束縛をうけないことを意味する。注意しておくが、このことから安易に『MPLS と第 3 層経路制御が一致しない』と考えてはいけない。第 3 層経路制御ドメインと MPLS による経路制御ドメインがお互い独立して生成しているだけで、第 3 層経路制御ドメインと同一の MPLS 経路制御ドメインを構成することは技術的に可能である。実際に、ラベル配布プロトコルに LDP を用いた MPLS の中継点毎転送モードでは、MPLS の経路制御ドメインは第 3 層経路制御ドメインと完全に一致する。

したがって、MPLS においては、

- 経路ドメインを生成する機構
- 経路ドメインに応じた転送を実現する機構

を完全に独立したものとして扱うことが可能である。ここでの『経路ドメインを生成する機構』とは、ラベル配布プロトコルと中心とした FEC/ラベルマッピングの生成系であり、『経路ドメインに応じた転送を実現する機構』とは LSR を中心とした MPLS の転送アーキテクチャを指す。

この特性は、転送がユニキャストであるかマルチキャストであるかに無関係で成立する。MPLS 技術では転送の特性に応じた経路ドメインの構成および、その実現をおこなうことで、任意の転送特性が実現可能である。

2.2.2 MPLS 経路ドメインの生成機構

MPLS 経路ドメインは FEC とラベルのマッピン

グ情報で構成される。各 LSR はラベル配布対となっている LSR と FEC/ラベルマッピング情報をラベル配布プロトコルを用いて交換する。

FEC/ラベルマッピング情報の交換によって、結果的にラベルスイッチパス (LSP:Label Switch Path) が構成される。LSP は一種の仮想経路 (VP: Virtual Path) である。ユニキャスト通信ではこのラベルスイッチパスは一本の連続した経路となる。マルチキャスト通信については次の章で述べる。

2.2.3 MPLS 経路ドメインに応じた転送の実現機構

MPLS 経路ドメイン生成系によって生成された経路ドメインを実際に実現するための要素が LSR である。LSR は FEC に対応するラベルから導出される LSP を通してパケットの転送処理をおこなう。

以下に MPLS ドメイン内のパケット転送処理を示す。

1. 入口 LSR での FEC 解析
   各パケットの FEC を上位層の情報を用いて解析し、MPLS 的な FEC として識別する。必要なら FEC に対応する LSP を形成する。
2. 入口 LSR での初期ラベル付加
   FEC に対応するラベルをパケットに付加し、そのパケットを MPLS ドメイン内に転送する。
3. 中間 LSR でのラベル置換処理
   パケットに付加されているラベルを、次中継点となる LSR と折衝して得られたラベルと付け変えて転送する。
4. 出口 LSR でのラベルの除去
   MPLS ラベルを取り除き、通常パケットとして転送する。

これらの操作と通常の第 3 層転送を比較を図 2.2 に示す。

FEC F を持つパケットの伝送手順を以下に示す。まず入口 LSR では以下のようにほぼ通常の転送処理と同様な処理をおこなう。

1. 第 3 層のヘッダを解析し、そのパケットの FEC を決定する。ここでは FEC を F とする。
2. 隣接 LSR と折衝してあった FEC F に対応するラベル (ここではラベル L) をパケットに付加する。
3. 次中継点となる隣接 LSR に転送する。

続く中継 LSR では、

1. ラベル L から FEC F であることを識別する。
2. FEC F に対応するラベルをラベル表から検索 (ここではラベル L' とする)。
3. パケットのラベルをラベル置換 ($L \rightarrow L'$) する。
4. 次中継点となる隣接 LSR に転送する。

のように、入口 LSR での解析結果がラベルとして伝達されているため、第 3 層情報を解析せずに単純なラベル置換処理のみで転送できる。最終的な MPLS 出口では、ラベルを取り除いてから通常 IP パケットとして第 3 層転送する。

2.3 MPLS におけるマルチキャストの実現

2.3.1 MPLS におけるマルチキャスト実現の構成要素

MPLS でマルチキャストを実現するために考察しなければならない要素は大きく以下の 2 つの分類できる。これらは直交した要素である。

- マルチキャスト経路ドメインの生成機構
- マルチキャスト経路ドメインを実現するマルチキャスト配送機構

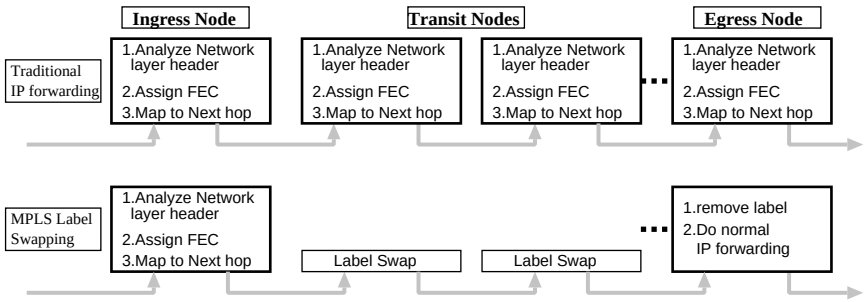

図 2.2. MPLS 転送と第 3 層転送の比較

前者はマルチキャスト配送をおこなうための経路情報を形成するためのアルゴリズム、経路プロトコルなどが含まれる。後者は、マルチキャスト経路ドメインで生成された配送木を実際に MPLS 的に実現するもので、LSR の実装などが含まれる。

2.3.2 マルチキャスト対応の現状

MPLS は本質的にはユニキャスト、マルチキャストの差異を問わない技術である。IETF での議論においても、MPLS においてはユニキャストだけでなマルチキャストを視野に入れた議論をおこなうべきだと言われている。しかしながら、実際のところ、現在の MPLS アーキテクチャ仕様 [140] では、マルチキャストは将来の課題であるとし、ユニキャストについてしか規定していない。MPLS は全般的にまだ開発途上の技術であり、ユニキャストについてすら標準化が終了していない現状を考えると、この状況はある意味、現実的な対処の結果とも取ることができる。

一方、MPLS のアーキテクチャ仕様とは別に、MPLS におけるマルチキャストを論じている文書が複数公開されている。2000 年 4 月現在では、以下の 5 本の I-D が公開されている。

- draft-ietf-mpls-multicast-00.txt[123]
- draft-farinacci-mpls-multicast-01.txt[44]
- draft-hummel-mpls-explicit-tree-01.txt[76]
- draft-wu-mpls-multicast-te-00.txt[168]
- draft-acharya-ipsofacto-mpls-mcast-00.txt[2]

これらの文書はそれぞれスコープとしている要素は異なっているが、なんらかの形で MPLS におけるマルチキャストの実現を論じている。表 2.1 に、各 I-D がどの部分を論じているかを示す。表中の用語について捕捉しておく。

フレームワーク
MPLS におけるマルチキャスト実現の全般的フレームワークの規定、問題点、将来の課題についての議論

第 2 層独自
CR-LDP を用いた独自マルチキャスト配送木の形成方法および問題点の議論

第 3 層準拠
第 3 層での IP マルチキャスト経路制御機構で生成された配送木を MPLS 的に実現するための方法および問題点の議論

LSR 拡張機構
マルチキャスト対応 LSR についての議論

図 2.3 に各 I-D の経路ドメインの生成機構に関する関連を示す。

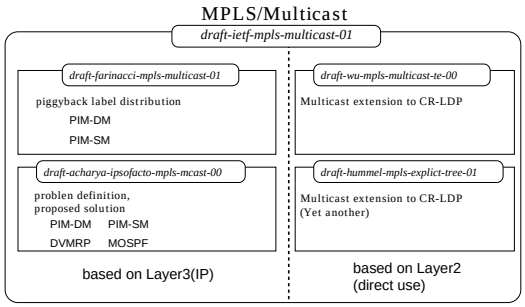

図 2.3.　各 I-D の相関図 (経路ドメイン生成機構関連)

2.3.3 MPLS/マルチキャスト対応フレームワーク

IETF の MPLS 分科会では、MPLS におけるマルチキャスト対応フレームワークを MPLS アーキテクチャ [140] とは独立して議論している [123]。このフレームワークは、最終的には MPLS でのマルチキャスト関連の仕様を一括して定義することを目的としており、

表 2.1.　MPLS/Multicast 関連 I-D の記述範囲

I-D 名	経路ドメインの生成機構	マルチキャスト配送機構
draft-ietf-mpls-multicast-00	フレームワーク	LSR 拡張機構
draft-farinacci-mpls-multicast-01	第 3 層準拠	無し
draft-hummel-mpls-explicit-tree-01	第 2 層独自	無し
draft-wu-mpls-multicast-te-00	第 2 層独自	無し
draft-acharya-ipsofacto-mpls-mcast-00	第 3 層準拠	無し

- マルチキャスト経路ドメインの構築
- マルチキャスト対応 LSR の実現

についてそれぞれ議論している。

現在の I-D の概要を以下に示す。

- MPLS で対象とする第 2 層の特徴のまとめ
- マルチキャスト対応 MPLS で扱うマルチキャストルーティングプロトコルの分類
- 複製されたパケットの第 2 層、第 3 層への同時出力の概要
- マルチキャスト対応 MPLS における LSP(Label Switched Path) 確立のトリガの分類
- ラベル配布プロトコルの他プロトコルへの相乗りに関する問題点の指摘
- 明示的経路指定に関する問題点の指摘
- QoS、CoS のサポートに関する考察
- マルチアクセスネットワークにおける問題の指摘
- その他の問題の指摘
  - TTL フィールドに関する問題
  - ラベル配布コントロールに関する問題
  - ラベル維持モードに関する問題
  - ラベル割り当てに関する問題
  - ラベル配布に関する問題

LSR の拡張機構、および、この I-D を含んだ各 I-D におけるマルチキャスト経路ドメインの構築については、それぞれ 2.4 章および 2.5 章でまとめる。

2.4 マルチキャスト対応 LSR 拡張

MPLS でマルチキャストを実現するための LSR の拡張については、draft-ietf-mpls-multicast-00[123] で議論されている。LSR 拡張は MPLS アーキテクチャへの拡張として将来的に MPLS の標準アーキテクチャである [140] に統合されると予想される。

本章では、MPLS 対応 LSR に要求される機構についてまとめる。

2.4.1 マルチキャスト配送機能の分類

MPLS でのマルチキャスト配送は以下の 2 種類に分類できる。

- 第 2 層から第 2 層へのマルチキャスト配送
- 第 2 層から第 2 層/第 3 層へのマルチキャスト配送

前者は MPLS ドメイン内で閉じた形式のマルチキャスト配送で、全配送木が MPLS ドメイン内にある場合がこれにあたる。一方、マルチキャスト対応の MPLS ルータによって構成されるネットワークとその外側のネットワークをまたがったマルチキャスト配送木 (図 2.4) を確立する場合も考えられるため、後者の配送機構も必要である。

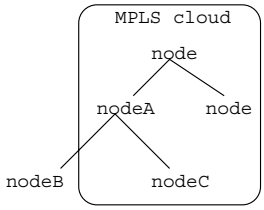

図 2.4. MPLS ネットワークの境界をまたいだ配送木

それぞれの用途を実現するためには、LSR において以下の条件を充足しなければならない。

1. 第 2 層からの入力に対して、第 2 層への複数の出力を行う
2. (1) に加えて第 3 層へも複数の出力を行う

図 2.4 の例では nodeA が (2) の機能を必要とする。nodeB は第 3 層でマルチキャスト配送を行うルータ、nodeC は第 2 層でマルチキャスト配送を行うルータである。nodeB は MPLS ネットワークの外側に存在するが、nodeC は MPLS ネットワークの内側に存在する。

2.4.2 第 2 層へのマルチキャスト出力手法

第 2 層からの入力に対する第 2 層への複数の出力をおこなう手法はいくつか存在する。現時点では、どの方法を利用すべきかについては議論の最中であり、実現可能なモデルを提示するにとどまっている [123]。それらのモデルを図 2.5 に示す。図の左側から、

- 2 層でのパケットコピー機構を利用
- 2/3 層での折衷案
- 3 層でのパケットコピー機構を利用

となっている。

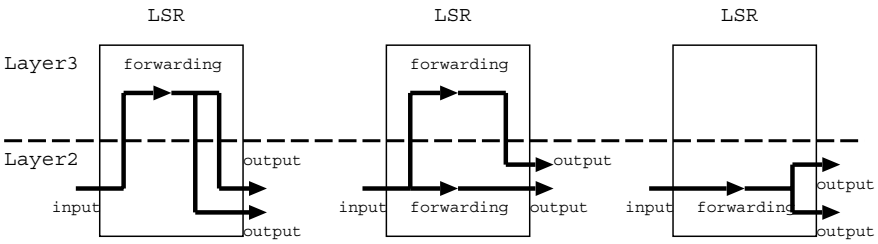

図 2.5.　第 2 層のマルチキャスト転送

2.4.3 第 2/3 層へのマルチキャスト出力手法

入力されたパケットを第 2 層および第 3 層へ出力する機能を実現するためには、図 2.6 に示す方法が考えられる。

[123] では図 2.6 に示す方法をサポートすることによる以下の利点を挙げている。

- MPLS ネットワークの内側と外側にまたがった配送木を確立できる。
- より低いサービス品質が要求された場合第 3 層へ出力することにより、より低いサービス品質が実現できる。これにより複数のサービス品質が定義できるので、区分化されたサービスの一種を提供できる。
- 後述の on Demand ラベル配布モードにおいてトラフィック駆動型のトリガが利用できる。

2.5 MPLS におけるマルチキャスト経路ドメインの構成

MPLS でのマルチキャストを実現するうえで現在盛んに議論されているのが、『MPLS 経路ドメインの生成手法およびその機構』についてである。2.3.2 節で論じたように、その機構は大きく

- 第 2 層で独自にマルチキャスト配送木を生成するもの
- 第 3 層の IP マルチキャスト配送木を MPLS 的に実現するもの

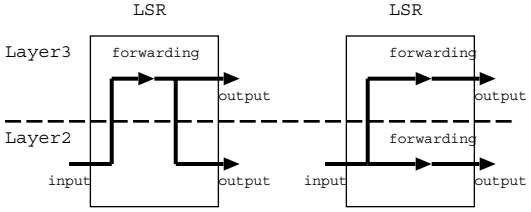

図 2.6.　第 2 層、第 3 層へのマルチキャスト転送

の 2 種類に分類できる。

現時点での議論の内容を以下にまとめる。

2.5.1 第 2 層マルチキャスト配送木の MPLS 的実現

第 2 層での独自マルチキャスト経路網の実現は、現時点ではほとんど議論されていない。[168] ではトラフィックエンジニアリングのマルチキャスト対応という視点から MPLS を捕らえる際に、CR-LDP を拡張した第 3 層とは独立したマルチキャスト配送木の実現を論じている。

2.5.2 第 3 層マルチキャスト配送木の MPLS 的実現

現在までに様々な IP マルチキャスト経路制御プロトコルが提案されている。現時点では PIM-DM および PIM-SM が今後の主力プロトコルになると考えられており、MPLS の IP マルチキャスト対応化に関しても、MPLS を PIM-DM および PIM-SM に対応させることを目的とした議論が活発である。

PIM-DM および PIM-SM を MPLS で実現する上で現在指摘されている問題点を以下にまとめる。

PIM-DM に関する問題点

[2] では MPLS のマルチキャスト対応化における PIM-DM に関する以下の問題点を指摘している。

- PIM-DM を用いてマルチキャスト経路制御をおこなうネットワークにおいて、現在の MPLS を利用する場合 PIM-DM の性質から以下の制約が発生する。
  - ラベル割り当て方法の制約
    - ホップ・バイ・ホップ (hop-by-hop) の転送
    - トラフィック駆動型の LSP の確立
  - ソース/グループのペア (以下 (S,G)) について入力ラベル、出力ラベルの統一ができない

PIM-SM に関する問題点

[44] ではラベル配布の一手法として、PIM-SM への相乗りを提案している。しかしながら、[123] では経路制御プロトコルへの相乗りに関して以下の欠点を指摘している。

- プロトコルの相乗りは既存のマルチキャスト経路制御プロトコルの拡張を必要とする。そのため、ラベル配布を特定のマルチキャスト経路制御プロトコルに相乗りさせることは複数の経路制御プロトコルの利用を制限する結果となる。

LSP の確立

マルチキャスト対応の MPLS における LSP 確立の駆動方法は現在以下の 3 種類の手法が提案されている [123]。

- 要求駆動
  制御メッセージの送受信をトリガとして LSP を確立する。
- トポロジー駆動
  トポロジーの変化 (経路表の変化) をトリガとして LSP を確立する。
- トラフィック駆動
  マルチキャストのデータの到着をトリガとして LSP を確立する。

IP マルチキャストでは複数の経路制御プロトコルが提案されており、経路制御プロトコルによって利用できる駆動方式がそれぞれ異なることが指摘されている。

ラベル配布

現在提案されている LDP では以下に示すモードやコントロールが定義されている [8]。

- ラベル配布モード
- ラベル配布コントロール
- ラベル維持モード

上記のモードやコントロールについて、それぞれ複数のモードやコントロールが具体的に定義されている。

マルチキャスト対応の MPLS においても基本的にはユニキャストの MPLS と同様のモードが用いられる。ただし、モードの組み合わせ方やマルチキャスト経路制御プロトコルの関連によっては利用できない組み合わせも存在する。

ラベル配布モード

- on Demand ラベル配布モード
  他 LSR からの明示的な要求があった場合のみ、自 LSR における FEC/ラベルマッピングを広告する。
- Unsolicited ラベル配布モード
  他 LSR からの明示的な要求がなくても、自 LSR における FEC/ラベルマッピングを広告する。ただし、他 LSR からの明示的な要求があった場合も自 LSR における FEC/ラベルマッピングを広告する。

ラベル配布コントロール

- Independent ラベル配布コントロール
  他 LSR の FEC/ラベルマッピングに依存せずに FEC/ラベルマッピングを配布する。
- Ordered ラベル配布コントロール
  他 LSR の FEC/ラベルマッピングに依存して FEC/ラベルマッピングを配布する。

MPLS のマルチキャスト対応化において、Independent コントロールの場合、前述の 3 種類のトリガは全て利用できるが、Ordered コントロールで第 2 層、第 3 層へのマルチキャストをサポートする場合、トラフィック駆動型の LSP の確立はできない。

ラベル維持モード

- Conservative ラベル維持モード
  必要なラベルだけ保持する。
- Liberal ラベル維持モード
  全てのラベルを保持する。

LDP[8] では上記の 2 種類のラベル維持モードが提案されている。

MPLS のマルチキャスト対応においては Liberal モードの利用は意味を持たない。その理由として [123] では以下の 2 点を挙げている。

- ユニキャストにおいては、全ての LSR は全ての FEC に対する経路を保持している。しかしマルチキャストにおいては、全ての LSR が全ての FEC に対する経路を保持しているわけではない。
- マルチキャストでは、LSR はどの隣接 LSR に対してラベル要求/ラベルマッピングメッセージを送信すべきか常に知っている。たとえば、ユニキャスト下流モードでは LSR はどこにラベルマッピングメッセージを送信すべきか知らず、全ての隣接 LSR に対してメッセージを送信する。このケースにおいて、Liberal モードをサポートすることは特別に新しいメッセージを生成することはなく、Liberal モードをサポートした場合のコストは低いと考えられる。

  ユニキャストの MPLS においては LSR1 から LSR2 への経路は必ずしも LSR2 から LSR1 への経路と一致するとは限らない。そのため、ある LSR は次ホップを知っている必要はあっても、前ホップを知っている必要はない。しかしながら、マルチキャスト対応の MPLS においては、LSR は次ホップおよび前ホップを知っている必要がある。

2.6 MPLS におけるマルチキャスト実現の課題

その他、現時点で論じられているトピックについてまとめる。

2.6.1 トラフィックエンジニアリングへの応用

ISP のバックボーンネットワークにおいて多数のマルチキャストグループのトラフィックが存在するときに異なるマルチキャストトラフィックに対してそれぞれ区分化されたサービスを提供することは非常に困難な課題となりつつある。複数のマルチキャストのトラフィックにおいてネットワークの資源を最適化し効率的に利用することも同様に非常に困難である。

[168] ではマルチキャストにおけるトラフィックエンジニアリングの一手法として MPLS および明示的な経路束縛を用いることを提案している。

2.6.2 TTL フィールドに関する問題

[123] はマルチキャストに対応した MPLS における TTL フィールドに関する問題を指摘している。

パケットの無限ループの防止を目的として IP ヘッダの TTL フィールドが用いられることがある。IP マルチキャストでは IP ヘッダの TTL フィールドの値をパケットの転送回数の上限とすることでパケットの無限ループを防ぐ。しかしながら、MPLS を用いたネットワークではパケットの転送時に IP ヘッダの TTL フィールドの値は減算されない。

ユニキャストの LSP では入口ノードおよび出口ノードはそれぞれ唯一であるので、LSP は必ず 1 本に決定することができ、LSP の長さ (LSP における LSR の経由回数) はあらかじめ判明している。そのため入口ノードにおいてその LSP における第 2 層でのパケットの転送回数が判明しており、TTL フィールドの値は入口ノードにおいて減算することができる。マルチキャストでは配送木の枝の長さはそれぞれ異なる可能性があるので TTL フィールドの値は入口ノードでは減算できない。そのため、TTL フィールドの値が本来はパケットの破棄を意味する場合であってもパケットの転送がおこなわれてしまう可能性がある。

2.6.3 明示的経路指定に関する問題

MPLS では [83] をサポートすることにより明示的にパケット転送の経路を指定することができる。[123] ではマルチキャスト対応 MPLS における明示的経路指定における以下の問題点を指摘している。

- 既存の MPLS において明示的に指定された経路は双方向である。マルチキャストにおいてはマルチキャストデータ (配送者から受信者) とマルチキャストルーティングメッセージ (受信者から配送者) が同一の経路を通ってしまう。
- RPF (Reverse Path Forwarding) の計算が明示的に指定された経路によっておこなわれてしまう。

2.6.4 QoS/CoS の実現

[123] ではマルチキャスト対応 MPLS における区分化サービスの実現は容易であると論じている。あるマルチキャストのソース/グループ (以下 (S,G)) の組み合わせに CoS の情報を付加した (S,G,CoS) ツリーを LSP に対応させることで実現できると論じている。

[123] は RSVP を用いた QoS の実現についても言及している。マルチキャストにおける重要な問題として、どのようにして 'heterogeneous receivers' の概念を第 2 層に適用するかという問題が挙げられる [20]。[123] ではこの問題に対し、以下の提案を行っている。

- 1 つの (S,G) の組み合わせに対してサービスクラスごとに木を構成する (たとえば best-effort 用の木、QoS 用の木)。これらの木ごとに LSP を確立する。
- 前述の第 2 層、第 3 層混在の出力を用いて、単一の配送木上で異なるサービスクラスを実現する。

2.7 まとめ

本章では MPLS のマルチキャスト対応化に関する議論や提案を調査し、まとめた。IETF における MPLS のマルチキャスト対応化に関する標準化動向についてもまとめた。

現状では、MPLS マルチキャストについては、将来的な方向性についてや問題点の指摘が主な議論対象となっており、仕様や細部に関する提案は少ない状況である。その大きな理由は、MPLS の仕様全体がまだ策定途中であり、ユニキャストを実現する部分さえ議論途中の部分が多いことであろう。

MPLS で実際にマルチキャストを実現するためには、以下の項目についての詳細化/標準化が必要である。

- MPLS マルチキャストフレームワークの標準化
- LSR 拡張機構の標準化
- 各種マルチキャストモデルに関するコンセンサスの確立
- FEC とマルチキャスト関係の詳細化
- マルチキャスト配送木の表現手法の詳細化
- 各種マルチキャスト配送木の構成法の検討
- その他

また、MPLS のマルチキャスト対応化に関して現在指摘されている問題は、現在の MPLS と既存の IP マルチキャスト経路制御プロトコルの不整合に起因するものが多い。また、マルチキャストにおける QoS、CoS のサポートに起因する問題も少なからず存在する。これらに関する議論も今後おこなっていかなければならない。

今後は現在指摘されている問題点に対する対策や、具体的な仕様の提案が増え、MPLS のマルチキャスト対応化に関する議論はこれまで指摘された問題点の具体的な対策方法や仕様の決定に徐々に移行すると思われる。

第 3 章 AYAME: A design and implementation of the CoS capable MPLS layer for BSD Network stacks

3.1 Introduction

With the growth of the Internet, many animated discussions have been appearing about to addition of Service Classes to traffic. The Differentiated Services (Diffserv) architecture [16] is one of the most common technologies to append these functions to the today's Internet.

Furthermore, since the Internet will become a more high-performance network in the near future, some improvements on the following area are being required:

- on the network layer routing performance
- on the scalability of the network layer
- on the variety and functionality of routing services

When ISPs wish to provide the differentiated services, they need additional functions that support more efficiently and well-operated traffic engineering, well-controlled QoS management and VPN services in their networks. There is the "label switching forwarding paradigm with net-

work layer routing" that satisfies these requirements. The Multi Protocol Label Switching (MPLS) [140] is one of technologies, which are studied in IETF MPLS working group now, to actualize that paradigm. This technology is attracting a great deal of public attention in order to add these functions to the network that want to get Differentiated Service capabilities.

The MPLS is a technology for flexible transfer of Layer 3 packets using the fixed short-length Label information created from the Layer 3 address information or other information to constrain the route to a specific path. In an MPLS domain, when a data stream traverses a common path, a Label Switched Path (LSP) can be established using MPLS signaling protocols. The ingress Label Switch Router (LSR) assigns a label to each packet and transmits it to downstream. Each LSR along the LSP decides the next hop according to the label in each packet.

Since using label approach of the MPLS allows flexible control to routing, each node can decide the next hop by only the label(s) rather than the network layer information. In result, new routing services can be introduced easily, being independent of the existing routing mechanism, nor with change in the forwarding paradigm [11].

If the LSRs have the capabilities to define a QoS/CoS characteristic, and to map Diffserv Per Hop Behaviors (PHBs) to each LSP on it, the Diffserv enabled MPLS networks can be constructed [45].

Because the MPLS related researches are developing, we need platforms as research environment for it. In other word, we need implementation that can be changed and extended freely for the purpose of researches, verifications and experimental operations.

To address this situation, we designed the architecture of MPLS Router system that well suited to the CoS functions, and implemented it to the BSD UNIX Network stack (The NetBSD, one of the derive of BSD system.). This system is named "AYAME."

This paper discusses the AYAME architecture mainly, and how to match the MPLS functions to the BSD network stack.

3.2 AYAME: A new-generation network layer on BSD network stack

This section describes the architecture of AYAME, its basic characteristics, design issues, and some advance feature such as its design structure modularity, hierarchical Label Switching Engine support, and exception processing.

3.2.1 Overview of AYAME

AYAME is the implementation of Multi Protocol Label Switching (MPLS) based Label Switching Router (LSR) on NetBSD system, with CoS

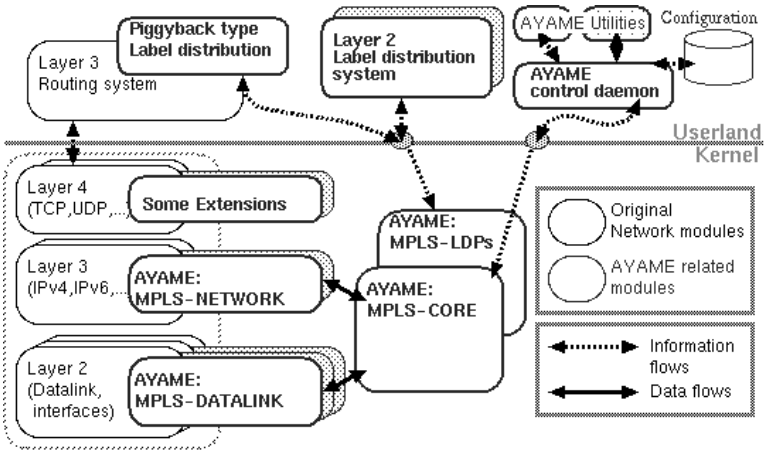

Fig. 3.1. AYAME system overview

related functions to support Diffserv capabilities. We implemented AYAME faithfulness to the policy of original BSD Network code. Because the purpose that the "packet forwarding" never changes itself before and after the insertion of MPLS functions into the system.

Most of the MPLS essential functions, such as the label swapping engine (LSE), Label information structures and any other mechanisms that would be needed when each packet is processed, are located in the kernel (details are below). But some entities, for example, a label distribution protocol(s), a mechanism for the interaction with layer 3 routing modules, and configuration utilities, are located in userland rather then kernel. 3.1 shows the brief architecture of AYAME.

Followings are main features and characteristics of AYAME:

- Characteristics:
  - Easily be able to support multiple Datalink layers and network layers according to a module structure.
  - Will be distributed under AS-IS license as same as BSD license.
- Basic features:
  - Basic MPLS label swapping, across different type interfaces
  - IPv4 support (IPv6 have not supported yet, but it will be support soon.)
  - Point-to-Point type link (LC-ATM [139], PPP (not yet)) and Broadcast type link (Ethernet [139]) support
  - A kernel API to support multiple label distribution protocol entities
  - Implementation of label distribution protocols, such as LDP [8], CR-LDP, and BGP piggyback label distribution
  - Hierarchical LSPs support
- Advance features:
  - CoS related functions (for Diffserv) using by ALTQ
  - Hierarchical Label Switch Engine (LSE) mechanism to support an ATM switch(es) as AYAME's LSE

3.3 Inside of AYAME kernel

This section describes AYAME kernel design policy and some remarkable characteristics of the implementation issue.

3.3.1 Design policy

The functions provided by MPLS is located between Layer 2 and Layer 3 in a network. Therefore, there are some requirements of changes in both layers to add the MPLS extensions to the existing network stack.

We must give much attention to introduce MPLS, which was not involved in the design of the existing network stack, without any violation of the current semantics. Because the total consistency might be destroyed if the thoughtless changes for the existing network cords affect some parts that have used those cords. In such cases, besides, it would be difficult to ensure the current semantics of network processing.

AYAME kernel implementation is using the "complete modularization" according to functions to minimize those impacts. We classified all function blocks that construct MPLS as follows:

- MPLS specific module (MPLS-CORE,MPLS-LDPs):
  Core module of MPLS label processing
- Layer 3 related module(s) (MPLS-NETWORK):
  Interface(s) between a particular Layer 3 and MPLS-CORE module, (e.g. MPLS-IPv4, MPLS-IPv6) to process the Layer 3 specific functions
- Layer 2 related module(s) (MPLS-DATALINK):
  Interface(s) between a particular Layer 2 and MPLS-CORE module, (e.g. MPLS-Ethernet, MPLS-ATM) to process the Layer 2 specific functions

Because this approach can separate the MPLS functions from the existing network cords as far as possible, it needs the minimum alteration to the existing network cords. Furthermore, the layering adapted to a current BSD UNIX Network stack layering. We inserted the MPLS layer between the Datalink layer (Layer 2) and the Network layer (Layer 3). Most MPLS related functions are integrated into this layer.

In this layering architecture, the Network layer can treat the MPLS layer as a kind of "intelligent" network interface. While the MPLS layer introduces some new functions, it does not destroy the current network semantics. We extended some APIs to manipulate the MPLS specific configurations and data structures (the routing table, and so forth) as well. Such design policies are familiar with the current network applications, and make effortless to handle new functions offered by the MPLS layer from the upper layers.

This modularization makes division and relation between the functions clearer, in result, the extendability is improved very much as well. As MPLS is the developing technique in the standardization process, it is supposed that more specifications will be suggested henceforth. The AYAME design policy will make it easy to expand these new specifications.

3.3.2 Module structure

This section describes the module structure of AYAME.

3.2 is the illustration of data flow in the extended network stack with AYAME. In the rest of this section, summaries of each module will be described.

3.3.3 MPLS-CORE/MPLS-LDPs

Only MPLS specific functions are provided by the MPLS-CORE and MPLS-LDPs modules, while all functions that need interaction with any other layer are located in other modules rather than here.

The MPLS-CORE mainly provides MPLS style packet forwarding, "Label swapping." Because this module is designed symmetrical for input to and output from other modules, it treats both MPLS-NETWORKS(s) and MPLS-DATALINK(s) with a same manner. This module consists of two sub-modules, the Label Swapping Engine (LSE) that manipulates the label swapping and the label stack processing for each labeled packet, and the Label Information Database (LID) that maintains the system-wide information related to MPLS label swap processing such as the Label-FEC binding.

The MPLS-LDPs provide some support functions to support Label Distribution Protocols, such as the functions to allocate labels and to preserve the label space consistency. Because the MPLS-LDPs would be marshaling the information from LDPs, the LID at MPLS-CORE keeps summarized information, which is enough to process a received packet. 3.3 shows detail of MPLS-CORE.

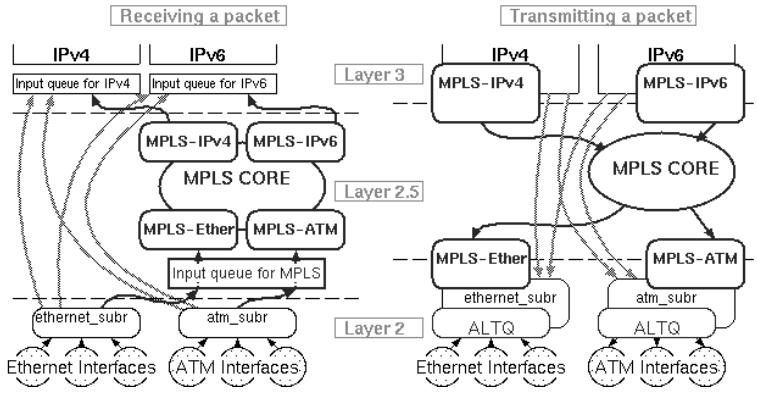

Fig. 3.2. Data flow of AYAME kernel

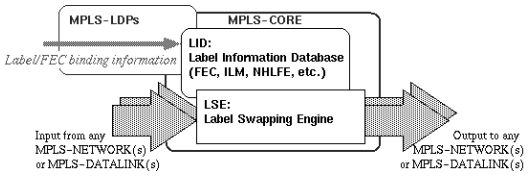

Fig. 3.3. MPLS-CORE and MPLS-LDPs

Detail of MPLS-LSPs' module internal structure is described below.

3.3.4 MPLS-NETWORK

AYAME provides one module per Network Layer contained in the system. The modules are generically called MPLS-NETWORK. (e.g. The MPLS-IPv4 manipulates IPv4 specific operations, and the MPLS-IPv6 performs IPv6 specific ones.) It provides a kind of buffer function that canonicalize a data structure between any network layer and MPLS-CORE, moreover it also provides the network layer specific functions to process MPLS label swapping, such as FEC analysis of each packet.

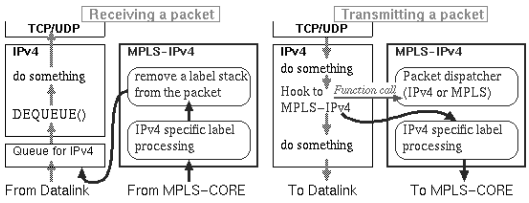

Fig. 3.4. MPLS-IPv4 (an example of MPLS-NETWORK)

3.4 shows the detail of MPLS-IPv4, which is one of MPLS-NETWORK modules to support the IPv4 network layer.

As an example, consider an operation that a packet, transmitted by any datagram from an upper layer or by the IP forwarding, and forwarded to a Next Hop node. In this operation, the IP packet is outputted from an interface connect with the Next hop node, by the ip_output() function. If the MPLS ingress node manipulates this process, to decide which it is needed to put the transferred packet as a labeled one in the MPLS domain, it can hook in the middle of the ip_output() function and perform the following sequence:

1. Analyze the network layer information of the packet and decide a FEC corresponding with it.
2. Lookup a FEC-to-NHLFE (Next Hop Label Forwarding Entry) mapping in the LID. If a corresponding entry is found, the packet should be passed to the MPLS module, or it should be sent back to the original IP stack.
3. Return the result.

If it does not have to deal with the packet as a labeled one, it can continue the process without any change of the existing semantics. The processing units that actually execute functions related to MPLS are located in the MPLS-IPv4. For this reason, the alternation of the existing IP structure can be minimized because this function requires no change of the codes but addition of a single hook.

3.3.5 MPLS-DATALINK

AYAME also provides one module per Datalink Layer contained in the system, as same as ones for the Network Layers. The modules are generically called MPLS-DATALINK. (e.g. The MPLS-Ethernet manipulates Ethernet specific operations, and the MPLS-ATM performs ATM specific ones.) Each module provides a kind of buffer function that canonicalizes a data structure between any network layers and MPLS-CORE, moreover it provides also the Datalink layer specific functions to process the MPLS label swapping, such as one to compose a packet from a packet payload and a MPLS label stack.

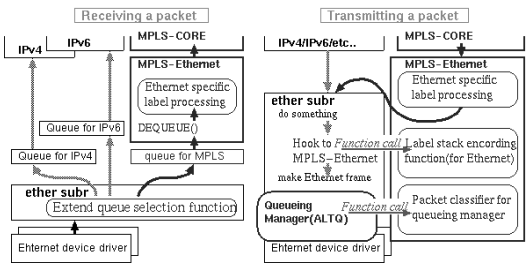

Fig. 3.5. MPLS-Ethernet (an example of MPLS-k DATALINK)

3.5 shows the detail of MPLS-Ethernet, which is one of MPLS-DATALINK modules to support Ethernet interfaces. This module provides some Ethernet specific label processing such as a label stack encoding and decoding for the Ethernet type media defined in [139] and queue control of output interfaces. This module also provides some of queue management support functions described below.

第 4 章　MPLS 実装 AYAME におけるパケット転送機構の設計および実装

4.1 MPLS 実装 AYAME

我々は、研究・開発用の MPLS スタックとして、**“AYAME”** の設計・実装を行った。

AYAME は、NetBSD に対する拡張として実装した。NetBSD は、4.4BSD の流れをくみ自由に利用可能で再配布可能な、UNIX-like なオペレーティングシステムである。我々は、AYAME の利用/配布が、基礎とするコードの利用条件/配布条件に束縛されることは研究開発目的に利用する上で好ましくないと考え、自由に利用可能で再配布可能な NetBSD を選択した。

AYAME の設計・実装においては、以下の点について特に留意した。

- **拡張性の高い FEC 決定手法**
  通常の IP ルータにおいては、各パケットに対する転送方針はネットワーク層の経路表のみで決定される。これはつまり、パケットの属する FEC は経路表のみから決定することが可能であると言える。しかし、この先のインターネットの高機能化を考慮すると、常にパケットの属する FEC が経路表のみから決定されるとは限らなくなると考え、より拡張性の高い FEC 決定手法を検討した。
- **モジュール構造を用いた実装**
  パケット転送機構において、特定のデータリンク層やネットワーク層に依存する部分と汎用部との切り分けを行った。これにより、現在対応していない新たな層への対応や新機能の追加を容易とした。
- **複数のラベル配布プロトコルの使用が可能**
  同一の FEC 群を取り扱う複数のラベル配布プロトコルの共存が可能な FEC/ラベル管理機構について検討を行った。
- **既存スタックへの最小限の影響**
  既存の NetBSD のネットワークスタックへの変更点が最小限に押さえられるよう設計実装した。これは、既存スタックへの変更は MPLS 拡張以外の新たな機能追加との親和性を低くし、また、MPLS 拡張自体の汎用性を低減する恐れが強いためである。

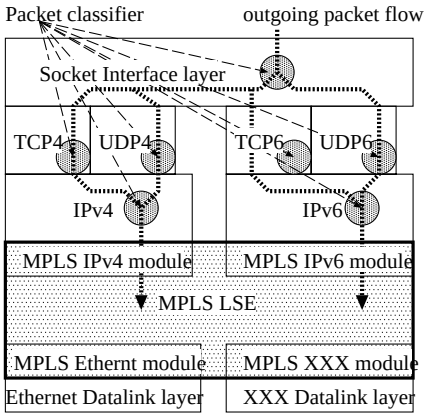

図 4.1.　LSE と FEC 決定機構

先にも、LSR を実装する上で拡張が必要となる機能は、パケット転送機構と FEC/ラベル管理機構であると述べたが、本章では特にパケット転送機構に着目している。AYAME のパケット転送機構について見ると、以下の 2 つの機構で構成される (図 4.1)。

- **LSE (Label Switching Engine)**
  ラベルの付いたパケットに対し、必要な処理を行ったうえで次ホップ LSR への転送を行う機構。
- **FEC 決定機構**
  MPLS ネットワークの入口点等において、ラベルの付いていないパケットに対しそのパケットが属する FEC を決定する機構。この決定に基づいて、パケットにラベルが付けられる。

本章では、以下に、AYAME におけるパケット転送機構の実装手法について述べる。

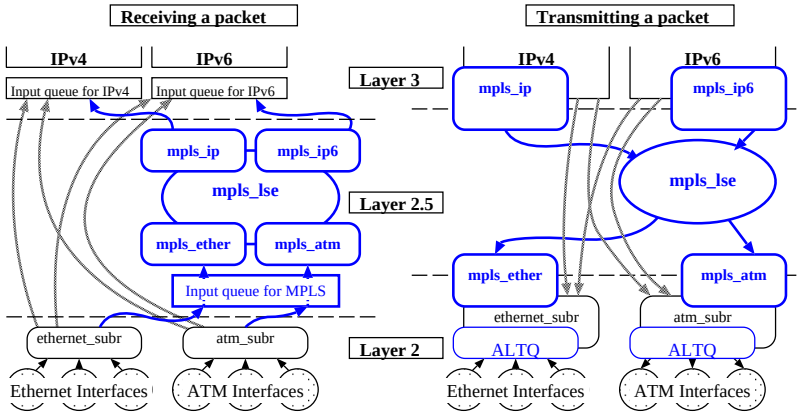

図 4.2. LSE モジュール構成

4.1.1 LSE の実現

AYAME における LSE は、NetBSD のネットワーク層スタックの 1 つとして実装した。その際、既存のネットワーク層/データリンク層スタックと LSE スタック間の API として、ネットワーク層-データリンク層間 API を用いている (図 4.2)。これにより、データリンク層およびネットワーク層の既存スタックへの変更が最小限に押さえられた。

また、LSE 内部構造は、複数のデータリンク層/ネットワーク層依存部分と汎用部を分離しモジュール化している。

汎用部 (*mpls_lse*) は、ラベルの付いたパケットに対し、ラベルから処理内容を検索し、pop/push/swap の操作を行った後、適当なインターフェースに出力するのみの単純なものである。複数回の pop/push/swap 動作は、再度 *mpls_lse* を通過することにより実現している。

データリンク層依存モジュール (*mpls_ether* など) は、ラベルのエンコーディング方法等データリンクに依存した処理を行う。

ネットワーク層依存モジュール (*mpls_ip* など) は、ネットワーク層より MPLS 層に送られてくるラベルの付いていないパケットについて、そのパケットの属する FEC に対応するラベルを付加した上で LSE に渡す等、特定のネットワーク層に依存した処理を行う。

このような構成のため、*mpls_lse* は、データリンク層からのパケットもネットワーク層からのパケットも同一視して扱うことができ、単純化されている。また、処理方針を記述する表は、ILM と NHLFE を組合わせた形で実現した。

4.1.2 FEC 決定機構の実現

先にも述べた通り、現状では、一般的にパケットの属する FEC はネットワーク層の情報で決まる。しかし、今後のインターネットの高機能化に伴い、パケットの転送により細かい粒度での制御が必要となると考えられる。ネットワークスタック内でパケットの持つ情報量は上位層ほど多く、下位層ほど少ない。例えば、TCP/IP においては、IP 層では { 始点アドレス, 宛先アドレス, 上位プロトコル種別 } しか知ることができないが、TCP 層では、これに加え「サービス種別 (ポート番号)」を知ることができる。

そこで、AYAME では、パケットの属する FEC の決定戦略として、パケットが上位層から下位層に流れる過程において、各層でそのパケットが属する FEC が決定できる場合はその FEC を採用し、決定できない場合はその決定を下位層に委ねるとした (図 4.3)。また、IP パケットの場合は上位層で FEC が決定できなかった際は、既存の IP ルータと同じく IP 層で経路表を検索した際に決定することとなる。

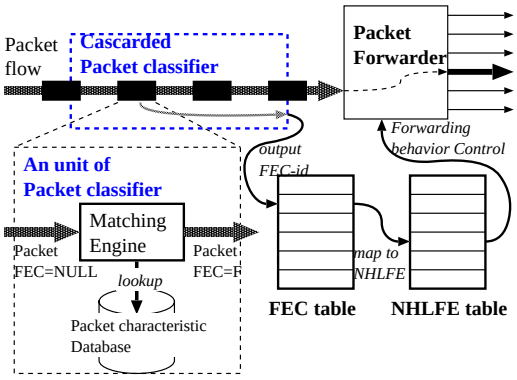

図 4.3. FEC 決定戦略

mbuf 拡張

この FEC 決定戦略を NetBSD のネットワークスタック上で実現するためには、ネットワークスタック間でパケットの受け渡しを行う際に、FEC に関する情報とともに受け渡す必要がある。

FEC に関する情報の受け渡し手段としては、以下の 2 手法が考えられる。

- スタックを実現している関数間で関数呼び出しの際の引数として受け渡す。
- パケットを格納している構造 (mbuf) の持つ情報の 1 つとして受け渡す。

前者は、変更の影響範囲が FEC 決定機構に関連する関数のみに限定される利点が挙げられるが、関連する多くの関数間の API を変更する必要が生じる。後者は、mbuf の構造を拡張するため mbuf を利用しているネットワークスタック全域への影響が懸念されるが、関数間の API を拡張する必要がなくコード上の変更は小さなもので済ますことができる。

AYAME では、今後 IP のみに限らず、さまざまなスタックを MPLS に対応させることを考え、後者 (mbuf の拡張) を選択し実装した。

具体的な mbuf の拡張は、パケットが格納される mbuf 群に付けられるパケット情報 (パケットの全長等) が格納されている *pkthdr* 構造体を拡張し、FEC 情報を格納する事とした。これに対応するため、mbuf の確保/分割等に関わる部分に若干の変更を加えた。

経路表の拡張

ここで述べた FEC 決定戦略においても、上位層でパケットの属する FEC が決定できない場合、最終的にはネットワーク層の情報を用いて属する FEC を決定する。また、NetBSD においては、ほとんどのネットワーク層はパケットの取り扱い方針を経路表に格納している。

そこで AYAME では、経路表の各エントリに対し、FEC 情報を格納できるよう拡張した。経路表に FEC 情報を格納する手法としては、以下の 2 手法が考えられる。

- 次ホップを表す項目に、FEC 情報を格納し MPLS スタックを用いて転送を行うことを指示する。
- 新たに FEC 情報を格納する項目を追加し、ここに FEC 情報を格納する。

ここで、次ホップを表す項目は、既存の各種経路制御機構が直接読み書きする。そのため、前者の手法を採るためには、既存の経路制御機構への変更が必要となる。既存の経路制御機構は、既に各種さまざまなものが提案・実装されており、これらをそのままの形で利用し続けることが困難となる前者の実装は好ましくない。そこで、AYAME では後者の手法を選択し実装した。

具体的には、FEC 情報を格納するために経路表のエントリ内の項目を追加し、この情報を操作するために経路表操作 API の拡張を行った。

4.2 評価

我々の行った AYAME 実装は、パケット転送機構について考察すると、汎用性/拡張性の高いものとなった。現在の実装では、ネットワーク層として IPv4 および IPv6 のみにしか対応していないが、他のネットワーク層をただ単純に MPLS 対応される程度であれば、対応すべき新たなネットワーク層自体への非常に小さな変更とこれに対応した LSE のネットワーク層依存モジュールの実装のみで可能である。これは、設計時から、特定のプロトコル等への依存を極力避け、依存が不可欠な場合はその部分を汎用部と明確に切り分けることを着実に行った成果である。また、現在の実装では、データリンク層として Ethernet のみしか対応していないが、これも、一部の特殊なデータリンクを除いては小さな労力で対応が可能である。ここで、挙げた一部の特殊なデータリンクとは ATM およびフレームリレーを指している。ATM 等のデータリンクでは、パケットに対するラベルの付加手法として、各パケットのヘッダに直接付加せず、データリンクの持つ機能を利用することでラベルの情報を次ホップに伝える。このような場合、データリンク層コードの機能拡張とその拡張機能を用いるデータリンク層依存モジュールが必要となる。しかし、この場合も、LSE の汎用部自体は我々が実装したものを用いることができると考えている。

公開されている MPLS 実装としては、

1) NIST Switch [1]

2) MPLS for Linux (Univ. of Wisconsin)[2]

など数種の実装が存在する。しかし、1) は RSVP を MPLS 上で実現する事を目的とし、2) は LDP の実験を行う事を目的とし、特定用途向け実装との色合いが濃い。我々の AYAME は、設計段階より汎用性と拡張性を強く意識しており、その成果が先にも述べた高い汎用性/拡張性に繋がっている。特に、AYAME の FEC 決定機構はネットワーク層以外の情報を用いた経路制御への道を開いており、既存の IP 経路制御 (宛先アドレスに基づく経路制御) に縛られない新たな経路制御構造の導入も可能としている。

実際にネットワークを運用する場合には、現状の AYAME 実装には問題がある。現状の実装はコードの汎用性等と実装の容易さ優先して行われたため、パケット転送能力に関する最適化をほとんど行っておらず、MPLS の特徴の 1 つである「高速なパケット転送」が十分には生かされていない。しかし、現状の実装においても、既存ネットワーク層を用いたパケット転送と同等以上の性能は維持されており、さらに、各モジュール毎のコードの最適化を行うことによりさらなるパケット転送能力の増強を見込むことができると考えられる。また AYAME 実装の動作検証として、WIDE 9 月研究会 (平成 12 年 9 月 11 日〜14 日/参加者約 250 人) において AYAME 実装のルータ 7 台を持ち込み、ユーザセグメント (IPv4) のバックボーンとして大きな障害もほとんどなく約 4 日稼働した実績が挙げられる。

さらに、我々は、MPLS の特徴として挙げれられる「扱うことが可能な FEC 集合に制限がない」点を用いて、より粒度の細かい FEC を扱う経路制御機構を用いた研究・提案として次のようなものなどを手がけている [188][187]。

- マルチキャスト配送のための高機能配送機構の実現手法
- QoS 対応配送機構の実現手法

これらの設計・実装において、本章で述べた AYAME のパケット転送機能は、その用途での利用に十分耐えられるものである。

4.3 まとめ

我々は、MPLS に対応した LSR 実装 AYAME を提案、設計、実装した。本章では、AYAME の実装の中で、特に、パケット転送機構の実装、その汎用性を実現するために我々が用いた実装手法について述べた。

AYAME のパケット転送機構における、今後の課題としては、その高速化の実現が挙げられる。先に挙げた、モジュール単位でのコードの最適化に加え、各種データベースのキャッシュ機構の整備、パケット処理用ハードウェアとの連携等を検討・実装することによりパケット転送機構を高速化することを検討している。また、最適化後のパケット転送能力等に関する評価も行いたいと考えている。

また、我々は、インターネットにおけるパケット配送機構の高機能化に用いることができる技術の 1 つとして MPLS に注目しており、その評価実装に用いる基盤実装として AYAME を活用していく予定である。

我々は、AYAME は MPLS を用いた研究開発において、その評価実装等を行う際の基礎実装とできるものであると考えている。AYAME は、自由な利用、再配布が可能となるよう実装されており、今後の AYAME 実装の配布等についても検討している。

[1] [NIST Switch] *http://www.antd.nist.gov/itg/nistswitch/*

[2] [MPLS for Linux] *http://nero.doit.wisc.edu/mpls-linux/*