

### エンド間ネットワーク性能向上のための 発見的的手法に基づく アプリケーション層経路制御手法

松田 一仁† 長谷川 剛† 亀井 聡†† 村田 正幸†  
†大阪大学 大学院情報科学研究科  
††NTT サービスインテグレーション基盤研究所

### 研究背景

- アプリケーション層経路制御
  - エンド間ネットワーク性能指標（遅延時間、利用可能帯域等）を用いて経路選択
  - 他エンド端末を経由する経路によりネットワーク性能向上の余地がある
  - IP 層の経路制御はルータ・AS ホップ数、及び ISP 間の契約関係によって経路選択が行われるため

ネットワークシステム研究会 11/09/01 2

### アプリケーション層経路制御の問題点

- 他の AL ノードの経路選択状況を考慮せずに経路を決定すると AL リンクを複数の AL 経路で共有する可能性がある
  - 遅延時間の増加や利用可能帯域の低下等、AL 経路の性能の劣化がおこる
- IP 層の経路制御と比べて通過する IP リンクが増加する可能性がある
- 利用するトランジットリンクが増加しネットワーク全体のトランジットコストが増大する

A-B 間で E-F 間の AL リンクを使用しているとき、C-D 間がこれを考慮せずに経路を選択すると E-F 間のリンクを共有する

A-B 間の IP 層経路制御による経路と比べ A-E-F-B という経路は通過するトランジットリンクの数が増加する

ネットワークシステム研究会 11/09/01 3

### 研究の目的

- 他ユーザの経路選択状況を考慮しないアプリケーション層経路制御はユーザ性能（遅延時間、利用可能帯域など）が劣化する可能性がある
- アプリケーション層で複数ホップの経路を用いることにより、利用するトランジットリンクが増加し、トランジットコストが増加する可能性がある

全体の経路選択状況を考慮してネットワーク性能の向上やトランジットコストの削減を行うアプリケーション層経路制御手法の提案

- アプリケーション層経路選択を最適化問題として定義する
- 定義した最適化問題を発見的的手法で解くアプリケーション層経路制御手法を提案する
  - 広範な状況に対応するため、集中処理・分散処理の両方を考案する

ネットワークシステム研究会 11/09/01 4

### アプリケーション層経路制御の定式化 (1/2)

- IP 層の経路制御のルーティング行列
  - IP ルータ間経路における IP リンクの利用状況を行列で記述
$$A^{IP} = \begin{pmatrix} IP_1^1 & \dots & IP_1^{(N-1)N} \\ \vdots & \ddots & \vdots \\ IP_M^1 & \dots & IP_M^{(N-1)N} \end{pmatrix}$$
- アプリケーション層 (AL) ネットワーク
  - 各 IP ルータに接続されたエンドホストが AL ノードとなり、いくつかの AL ノード間に AL リンクを張ってネットワークを構築する
  - AL ネットワークのトポロジは右式で表せる
  - 各 AL リンクは IP ルータ間の経路を用いる
  - AL ノード間でデータ転送に用いられる経路は、1 本以上の AL リンクを用いて行われる
    - 与えられたトポロジに対し、AL ノードペア  $j$  が利用できる経路は  $R_j^{AL}$  で表せる

$$R_j^{AL} = \{(p_1, p_2, \dots, p_h) | h \geq 1, s_{p_1} = s_j, t_{p_h} = t_j, t_k = s_{k+1} \ (2 \leq h, 1 \leq k \leq h-1), c_{p_k}^{AL} = 1 \ (1 \leq k \leq h)\}$$

存在する AL リンクをたどって到達できる経路

ネットワークシステム研究会 11/09/01 5

### アプリケーション層経路制御の定式化 (2/2)

- アプリケーション層経路制御ルーティング行列
  - アプリケーション層経路制御は AL ノードペアに対して利用可能な AL 経路を選択
  - 選択された AL ノードペア間の経路における AL リンクの利用状況を行列で記述
$$A^{AL} = \begin{pmatrix} AL_1^1 & \dots & AL_1^{(N-1)N} \\ \vdots & \ddots & \vdots \\ AL_{(N-1)N}^1 & \dots & AL_{(N-1)N}^{(N-1)N} \end{pmatrix}$$
- AL 経路のエンド間ネットワーク性能算出
  - 選択された AL 経路のエンド間ネットワーク性能（遅延時間、利用可能帯域、トランジットコストなど）をルーティング行列  $A^{AL}$  と AL ノードペア間のトラヒック要求量を示す行列  $\mathcal{Y}$  を用いて算出する
    - 下式の要素  $d_j^{AL}$ ,  $b_j^{AL}$ ,  $c_j^{AL}$  は AL ノードペア  $j$  間の AL 経路の遅延時間、利用可能帯域、トランジットコストを表す
    - トラヒック要求量に対して AL リンクの遅延時間を計算する関数
    - 各 AL 経路の遅延時間、トランジットコストは AL リンクの遅延時間、トランジットコストを並べた  $f_D(\mathcal{Y})$ ,  $f_C(\mathcal{Y})$  と  $AL_i^j$  の各列の積和
    - 各 AL 経路の利用可能帯域は  $f_B(\mathcal{Y}, A^{AL})$  によって算出される各 AL 経路上の AL リンクの最小の利用可能帯域

$$D^{AL} = (d_1^{AL} \ d_2^{AL} \ \dots \ d_{(N-1)N}^{AL}) = (f_D(\mathcal{Y}), A^{AL})$$

$$B^{AL} = (b_1^{AL} \ b_2^{AL} \ \dots \ b_{(N-1)N}^{AL}) = (f_B(\mathcal{Y}), A^{AL})$$

$$C^{AL} = (c_1^{AL} \ c_2^{AL} \ \dots \ c_{(N-1)N}^{AL}) = (f_C(\mathcal{Y}), A^{AL})$$

ネットワークシステム研究会 11/09/01

### アプリケーション層経路最適化問題

- ユーザ性能やトランジットコストを目的関数とする最適化問題
- AL ノードペア間の遅延時間の平均値を最小化

各 AL ノードペアには利用可能な経路を割り当てる

minimize :  $(\sum_{j \in K} d_j^{AL}) / |K|$

subject to :  $r_j \in R_j^{AL} (j \in K)$

- 各 AL ノードペア間の遅延時間が相互に影響を及ぼさない場合は、それぞれの遅延時間を最小化する問題に等しい
- 実際には各 AL ノードペアが利用する AL パスの重なりにより、遅延時間が相互に影響する
  - 総当たりで解くことが現実的な時間では不可能になるため、発見的手法を用いる

ネットワークシステム研究会

11/09/01

7

### AL リンク・経路のネットワーク性能算出方法

- 遅延時間
  - AL 経路を構成する各 AL リンクの遅延時間の和をその AL 経路の遅延とする
  - AL ノードペア間にトラフィック要求量を設定し、物理帯域とリンク利用率を用いて M/M/1 待ち行列モデルで算出する
- 利用可能帯域
  - AL 経路を構成する各 AL リンクの利用可能帯域のうち最小のものをその AL 経路の利用可能帯域とする
  - トラフィックを流したい AL ノードペアに max-min アルゴリズムに従って割り当てる
- トランジットコスト
  - AL リンクに含まれる AS 間リンクを同一 ISP 内リンク、ピアリングリンク、トランジットリンクに分類し、それぞれに設定したコストの和をトランジットコストとする
  - AL ノードペア間にトラフィック要求量を設定し、上記リンクの種類とトラフィック量によって算出する

ネットワークシステム研究会

11/09/01

8

### 発見的手法に基づくアプリケーション層経路制御手法：集中処理

- 発見的手法を用いて最適化問題の制約条件を満たす解を求め、アプリケーション層経路制御における経路選択に適用する

#### 1. 通常の焼きなまし法 (Simulated Annealing) に基づく手法

- 状態 S を「通信要求を持つ AL ノードペアが用いる経路の集合」とする
  - 初期状態は 1 ホップの AL 経路を用いる
- 現在の状態の近傍状態を選択する
  - 近傍状態とは、現在の状態から一部の AL ノードペアの経路を変更した状態とする
- コスト関数として、最適化問題の目的関数を用いる
- 初期温度や温度の冷却関数がパラメータとなる

Algorithm 焼きなまし法に基づく経路選択

```

1:  $T \leftarrow 0, T \leftarrow T_{init} \rightarrow S \leftarrow S_{init}$ 
2: while  $T > 0$  do
3:    $S_{tmp} \leftarrow Neighbor(S)$ 
4:   if  $Cost(S) > Cost(S_{tmp})$  then
5:      $S \leftarrow S_{tmp}$ 
6:   else
7:      $r \leftarrow Random(0, 1)$ 
8:     if  $r < Probability(T, Cost(S), Cost(S_{tmp}))$  then
9:        $S \leftarrow S_{tmp}$ 
10:  end if
11: end if
12:  $T \leftarrow T + 1$ 
13:  $T \leftarrow Cooling(T, T)$ 
14: end while
                    
```

ネットワークシステム研究会

11/09/01

9

### 発見的手法に基づくアプリケーション層経路制御手法：分散処理

#### 2. 分散焼きなまし法 (Distributed Simulated Annealing) [1] に基づく手法

- 各 AL ノードで独立して実行する
- 各 AL ノードは自身を送信元とする AL リンクのネットワーク性能のみを計測する
- 各 AL ノードは自身を送信元とする AL 経路のみを変更したものを近傍状態とする
  - 集中処理では全 AL 経路を変更の対象としているため、焼きなましの状態遷移に差が生まれると考えられる
- AL 経路情報とネットワーク性能計測結果は AL ノード間で通信を行い、情報交換する
  - 取得した他ノードの情報を用いて焼きなましに用いる状態を更新し、コストを再計算する
  - 情報交換にかかるオーバーヘッドと焼きなましにおけるコスト計算の正確さはトレードオフの関係になる

[1] M. Arshad and M. C. Silaghi, "Distributed simulated annealing and comparison to DSA," in Proceedings of the Fourth Workshop on Distributed Constraint Reasoning, Aug. 2003.

ネットワークシステム研究会

11/09/01

10

### 性能評価

- PlanetLab ノードがアプリケーション層経路制御を用いることを想定し、以下のノード間情報を用いて評価を行う
- 遅延時間, IP 層経路
  - PlanetLab ノード間で traceroute を行うことで取得
- 利用可能帯域, 物理帯域
  - S-cube [2] が公開しているデータを利用
- ASレベルの経路
  - 上記で取得した IP 層経路に Route Views Project [3] で公開しているアドレスプレフィックスと AS 番号の対応データを当てはめて取得
- AS 間のリンク種別
  - 上記 traceroute の結果と CAIDA [4] が公開している AS 間関係情報(トランジットリンク, ピアリングリンク)を用いて取得

[2] Hewlett-Packard Laboratories Scalable Sensing Service, available at <http://networking.hpl.hp.com/s/cube/>.  
 [3] University of Oregon Route Views Project, available at <http://www.routeviews.org/>.  
 [4] University of California CAIDA, available at <http://www.caida.org/home/>.

ネットワークシステム研究会

11/09/01

11

### 評価の設定

- 焼きなまし法に用いるネットワーク性能指標
  - 集中処理では、焼きなましに必要な AL ノード間の各種性能指標はあらかじめすべて取得できているとする
  - 分散処理でも、各 AL ノードは必要な性能指標をすべて取得しており、また全 AL ノード間で計測結果を共有できるとしている
- 初期状態
  - 各 AL ノードペアの 1 ホップの AL 経路, すなわちアプリケーション層経路制御を用いない場合と同等の経路を用いる
- 近傍選択関数
  - 対象とする AL 経路 (集中処理ではすべて、分散処理では自身を送信元とする経路) のうち 1 % の経路を変更したものを近傍状態とする
- 分散処理における情報交換の頻度
  - 焼きなましの反復処理 100 回につき 1 回, AL ノード間で選択した AL 経路の情報を交換する

ネットワークシステム研究会

11/09/01

12

