

AMPSO を用いた VLAN 環境下における 動的な中継点制御のスケラビリティの向上

高橋 謙輔^{†1} 菅原 俊治^{†1}

本稿では、Angle Modulated Particle Swarm Optimization(AMPSO) を用いた VLAN 環境下における動的な中継点制御のスケラビリティの向上を報告する。我々は今までに、VLAN 環境下における中継点を動的に制御することで、冗長パケットを削減する手法を提案してきた。しかし、既存手法では VLAN 数やルータ数の増加に従って探索性能が低下することがわかった。そこで、本稿ではスケラビリティの向上を目的として AMPSO を VLAN 環境に適用させた手法を提案する。最後に、実トラフィックに基づいて作成したシミュレーション環境を用いて既存手法と比較・評価することにより、スケラビリティの向上を示す。

Improving the Scalability of Dynamic IP Routing Points Migration for VLAN Environment using AMPSO

KENSUKE TAKAHASHI^{†1} and TOSHIHARU SUGAWARA^{†1}

We report a improving the scalability of the dynamic migration of the IP routing points in Virtual LAN environments by angle modulated particle swarm optimization (AMPSO). We have developed the method that can adaptively select the routing points dynamically according to the observed traffic patterns and thus reduce the redundant traffic. However, we found that the developed method decrease the search performance according to the increase of the number of VLAN and router. In this paper, we will show that method using AMPSO in order to improve of scalability. Finally, we will evaluate the proposed method using the simulation environment that can generate the traffic based on the packet flows observed in the real-world environment.

1. はじめに

企業や大学などの組織内ネットワークでは仮想 LAN(Virtual LAN, VLAN) の技術を用いてネットワークを構築することが一般的になっている。VLAN はデータリンク層の仮想化技術であり、物理的なネットワーク上に複数の論理ネットワークを重畳させることができる。この技術を用いて、物理的制約に捉われることなく、論理的なネットワークを構築できる。しかし、エンド間の通信の機能を提供するネットワーク層からはこの仮想化が透過であるため、ネットワーク層の通信を中継するルータの配置によってはパケットの転送経路に冗長が発生し、帯域を浪費する可能性がある。

それに対して我々は、ネットワーク内のトラフィックの状況を把握し、ルーティングを行う中継点(ルーティングポイント)を適切に設定することにより、常に冗長性を減らし、動的変動に対応できる手法として、Particle Swarm Optimization (PSO)¹⁾ を用いた手法を提案した²⁾。しかし、この手法では環境中のルータ数や VLAN 数の増加とともに探索性能が低下し、最適解を発見できないことがわかった。そこで、本稿では Angle Modulated PSO (AMPSO)³⁾ を用いた新たな手法を提案し、動的な中継点制御のスケラビリティの向上を目指す。

2. 課題の定義

2.1 分散仮想ルーティング

図 1 に本稿で想定する VLAN ネットワークの構成と、そこで発生する冗長トラフィックの一例を示す。ここでは、中心となる L3 スイッチ(図中 SW-C, 以下コアスイッチ)に、複数の L3 スイッチ(図中 SW-E1 ~ SW-E3, 以下エッジスイッチ)が接続されている VLAN ネットワークを想定する。このように、VLAN ネットワーク中にルーティング機能を持つ機器が複数存在する場合、各論理ネットワークの中継点であるルータの配置によって冗長トラフィックを多発させ、ネットワーク全体の効率を落とす可能性がある。例えば、図 1 のように物理ネットワーク上に 3 つの VLAN (VLAN A, VLAN B, VLAN C。図中 {A, B, C}-En はエッジスイッチ SW-En に接続されている VLAN {A, B, C} を表す) がコアスイッチを経由する形で構成されたとする。ここで、SW-E1 に VLAN A と VLAN B のイ

^{†1} 早稲田大学
Waseda University

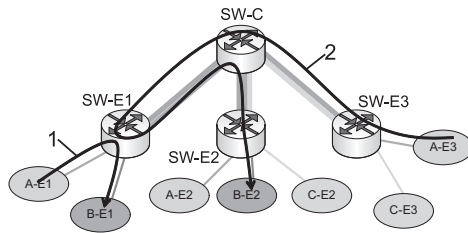


図1 冗長トラフィックの例
Fig.1 An example of redundancy traffic.

インタフェースを設定し（以下、これをルーティングポイントの設定と呼ぶ）ルーティングさせる場合、A-E1とB-E1の間の通信は図1の矢印1のように、エッジスイッチ内で効率的に行われる。しかし、A-E3とB-E2の間の通信について考えると、図1の矢印2のように、SW-CとSW-E1の間の線を往復することになる。そこで、ルーティングポイントをSW-Cに設定すると、A-E3とB-E2の間の通信のような異なるエッジスイッチを通る通信に関しては冗長を回避することができる。しかし、A-E1とB-E1間の通信のような同一のエッジスイッチを通る通信に関してはコアスイッチ経由となり冗長なパケットが発生する。このような冗長パケットを皆無にすることはできないが、ネットワーク帯域を有効に利用し、その効率を高めるためには、実際のトラフィックの流量などの動的な要因を考慮したうえで、冗長パケットをなるべく削減するようなルーティングポイントの制御が必要になる。

そこで我々は、トラフィックの変動に対応し、常に冗長性を抑えることを目的としてPSOを用いた動的中継点制御手法を提案した²⁾。しかし、この手法ではスケーラビリティがなく、環境中のルータ数やVLAN数の増加とともに最適なルーティングポイントの探索性能が低下する。本稿では、AMPSOを用いることによりスケーラビリティを向上させ、探索性能が低下しない手法を提案する。

2.2 PSOの概要

2.2.1 PSO

PSO¹⁾は群知能の一種であり、問題空間における最適解や準最適解を効率的に見つける最適化手法である。PSOでは多数の自律的なノード（以下エージェント）が探索空間に配置される。各エージェントは状態を持ち、評価関数を用いて現在の状態を評価する。そして、自身の探索履歴と、近隣のエージェントの探索履歴に基づいて次状態を決定する。結果的にエージェント群は評価関数の最適解に近づくことになる。

具体的には以下の通りである。エージェント*i*は、*D*次元空間で表現した現在の状態 \vec{x}_i, i の履歴の中での最適状態 \vec{p}_i 、状態変位量 \vec{v}_i の三つのベクトルで表現する。

現在の状態 \vec{x}_i は探索空間における暫定解として考え、アルゴリズムの各繰り返しにおいて評価関数を用いて評価する。もし、現在の状態 \vec{x}_i が探索履歴の中で最適評価値を得ることができれば、現在の状態 \vec{x}_i は履歴の中で最適状態 \vec{p}_i として保存する。その評価値を $pbest_i$ という変数に保存し、後の繰り返しの比較として用いる。次に、各エージェントは隣接関係にあるエージェントとお互いの評価値 \vec{p}_i および $pbest_i$ を通信し合う。そして、隣接関係にあるエージェントの状態と比較し、それらの中で最適評価値となる解を \vec{p}_g として保存する。最後に、現在の状態 \vec{x}_i に状態変位量 \vec{v}_i を加えることにより、新たな解を選択する。ここで、現在の状態 \vec{x}_i の更新は以下の式(1)のようになる。

$$\left. \begin{aligned} \vec{v}_i &\leftarrow \omega \vec{v}_i + \chi_1 r_1 (\vec{p}_i - \vec{x}_i) + \chi_2 r_2 (\vec{p}_g - \vec{x}_i) \\ \vec{x}_i &\leftarrow \vec{x}_i + \vec{v}_i \end{aligned} \right\} \quad (1)$$

ここで、 r_1, r_2 は $[0, 1]$ の範囲の一様分布乱数、 ω, χ_1, χ_2 は定数であり、各要素の相対的影響を決定するものである⁴⁾。式(1)は、PSOにおける各エージェントは、自身の最適値に基づく局所的な知識要素と、隣接エージェントの最適値に基づく社会性要素から次状態を決定することを示している。このように、探索空間を自身の履歴と群近隣の履歴を用いて探索することによって、PSOは最適解を発見する。

2.2.2 AMPSO

AMPSO³⁾は問題空間が離散空間である時に、問題空間と探索空間を分離させ、問題空間の次元を削減することによって効率的な探索を実現するPSOである。AMPSOにおいてエージェント*i*は、問題空間における状態（内部状態） \vec{x}_i と探索空間における状態（外部状態） \vec{x}'_i の二状態を持つ。ここで、内部状態は0もしくは1のビット列で表される。また、外部状態を内部状態に写像するため、内部状態の各ビットに対応させる標本 $\vec{x} = \{b_c\}_{1 \leq c \leq bitnum}$ を持つ（ $bitnum$ は内部状態のビット数）。写像関数は、三角関数を用いた角度変調の式(2)が用いられる。

$$g(x) = \sin\{2\pi(x - a)b \cos[2\pi(x - c)]\} + d \quad (2)$$

従って、内部状態 \vec{x}_i の*j*番目の要素 x_{ij} は対応する \vec{x} の値 b_j を用いて式(3)のように求められる。

$$x_{ij} = \begin{cases} 1 & \text{if } g(b_j) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

AMPPOにおける探索空間は、式(2)で用いられる $[a, b, c, d]$ の4次元である。AMPPOでは、この4次元の探索空間における最適値を探索する。そして、探索空間における外部状態を内部状態に写像することによって評価値を求め探索を続ける。

AMPPOは高次元の離散問題空間を4次元の連続探索空間に削減する。そのため、従来の離散PSO⁵⁾と比較して探索が高速になり、更に高精度になることも報告されている³⁾。

3. 探索性能の低下

3.1 既存手法の概要

既存手法では、エージェント i の状態を $X_i = \{x_{ijk}\}$ で表し、 x_{ijk} が1であればVLAN j のルーティングポイントが k であることを示している。また、式(1)を用いて求めた \vec{v}_i に対してシグモイド関数を用いた値に、VLAN j 毎に合計が1になるように正規化した状態変位量を $V_i = \{v_{ijk}\}$ で表し、VLAN j のルーティングポイントとして、 k を選択する確率を示している。

3.2 探索性能低下の調査

VLANのルーティングポイントの決定には、VLAN毎にルーティングポイントは一つのみという制約がある。従って、既存手法ではVLAN毎に状態変位確率の合計が1になるように正規化を行い、ルーレット選択によりルーティングポイントを決定した。しかし、ルータ数の増加とともに、正規化の影響が強くなるという課題が生じる。ここで、簡単のために履歴の中での最適状態 \vec{p}_i と現在の状態 \vec{x}_i のみで次状態が決定されると考える。ルータ数が3の時に、 $\vec{p}_i = \{0, 1, 0\}$ 、 $\vec{x}_i = \{1, 0, 0\}$ の場合 $\vec{p}_i - \vec{x}_i = \{-1, 1, 0\}$ となり、 $\vec{v}_i = \{0.18, 0.49, 0.33\}$ となる。この時は履歴の中での最適状態が選択される確率が一番高くなる。しかし、ルータ数が4の時に、 $\vec{p}_i = \{0, 1, 0, 0\}$ 、 $\vec{x}_i = \{1, 0, 0, 0\}$ の場合を考えると、 $\vec{p}_i - \vec{x}_i = \{-1, 1, 0, 0\}$ となり、 $\vec{v}_i = \{0.13, 0.37, 0.25, 0.25\}$ となる。この場合、履歴の中での最適状態が選択される確率が0.37なのに対して、履歴の中での最適状態及び現在の状態で選択されていない箇所が選択される確率は合計で0.5と最大になる。このように、ルータ数の増加とともに、履歴の中の最適状態を選択する確率は低下する。ルータ数の増加に伴う選択確率の変化を図2に示す。図2で1は履歴の中での最適状態を選択する確率を、-1は現在の状態を選択する確率を、0は履歴の中での最適状態及び現在の状態で選択されていない状態を選択する確率

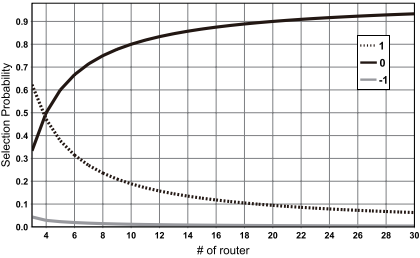


図2 ルータ数の増加に伴う選択確率の変化
Fig. 2 Variation of selection probability according to the increase of the number of routers.

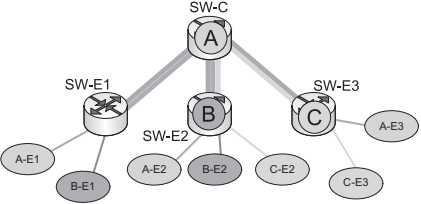


図3 解の表現方法
Fig. 3 An expression of a solution.

を示す。図2より、既存手法ではルータ数の増加とともに最適状態の選択確率が低下していることがわかる。また、本節では簡単のためにVLAN数を1としたが、VLAN毎にこの課題が生じるため、VLAN数の増加とともに全VLANが最適状態を発見することが難しくなる。これらのことより、既存手法ではVLAN数やルータ数の増加とともに探索性能が低下することがわかる。

4. 提案手法

4.1 解の表現方法

本稿での解は、 $S = \{s_1, s_2, \dots, s_{N_{VLAN}}\}$, $s_j \in [0, N_{er} - 1]$ と定義する。ここで、 N_{VLAN} はVLANの数、 N_{er} はルータの数を表しており、 s_j はVLAN j のルーティングポイントを表す。図1においてコアスイッチSW-Cに0を、エッジスイッチSW-Enには n を対応させると、 $S = \{0, 2, 3\}$ は、VLAN A, VLAN B, VLAN CのルーティングポイントがそれぞれSW-C, SW-E2, SW-E3であることを示す(図3参照)。なお、図3の $\{A, B, C\}$ はそれぞれVLAN $\{A, B, C\}$ のルーティングポイントを示す。

4.2 エージェントの状態

4.2.1 内部状態

エージェント i の内部状態を

$$X_i = \begin{pmatrix} x_{i11} & \dots & x_{i1[\log_2 N_{er}]} \\ \vdots & \ddots & \vdots \\ x_{iN_{VLAN}1} & \dots & x_{iN_{VLAN}[\log_2 N_{er}]} \end{pmatrix}$$

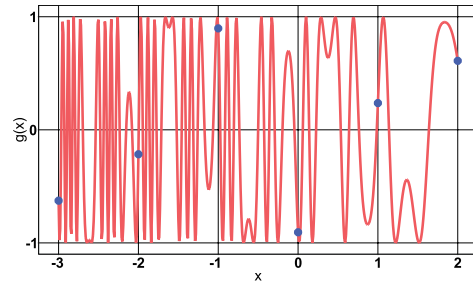


図 4 エージェントの状態例

Fig. 4 An example of an agent state.

$x_{i_{jk}} \in \{0, 1\}$ と定義する．ここで，各行は VLAN j のルーティングポイントのビット列を表す．例えば，

$$X_i = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 1 & 1 \end{pmatrix}$$

である場合は， $S_i = \{(00)_2 \bmod N_{er}, (10)_2 \bmod N_{er}, (11)_2 \bmod N_{er}\} = \{0, 2, 3\}$ を表す．ここで， $\bmod N_{er}$ を行うことにより，実際のルータ数よりも大きな数になることを防ぐ．

4.2.2 外部状態

従来の AMPSO では写像関数として式 (2) を用いた．しかし，本稿では式 (2) から d を取り除いた式 (4) を用いる．

$$g(x) = \sin\{2\pi(x - a)b \cos[2\pi(x - c)]\} \quad (4)$$

従って，エージェント i の外部状態は $\vec{x}'_i = [a, b, c]$ の 3 次元で表せる．また，外部状態を内部状態に写像させるため標本を， $\bar{x} = \{-\lceil \text{bitnum}/2 \rceil, -\lfloor \text{bitnum}/2 \rfloor + 1, \dots, 0, \dots, \lfloor \text{bitnum}/2 \rfloor - 1\}$ と定義する．例えば，図 3 のように $N_{VLAN} = 3, N_{er} = 4$ の環境は， $\text{bitnum} = \lceil \log_2 4 \rceil \times 3 = 6$ となり， $\bar{x} = \{-3, -2, -1, 0, 1, 2\}$ となる．この時， $\vec{x}'_i = [2.29, -0.47, 1.89]$ とすると， $g(x) = \sin\{2\pi(x - 2.29) \times (-0.47) \times \cos[2\pi(x - 1.89)]\}$ ， $g(-3) = -0.63, g(-2) = -0.22, g(-1) = 0.90, g(0) = -0.91, g(1) = 0.24, g(2) = 0.61$ となり（図 4 参照），内部状態 $X_i = \{\{00\}, \{10\}, \{11\}\}$ に写像される．

表 1 フローテーブルの例

Table 1 An example of flow table.

送信元	宛先
A-E1	B-E1
A-E2	C-E3
B-E2	C-E2

4.3 提案アルゴリズムの詳細

4.3.1 初期化

群全体のエージェント数を N_p とする．エージェント i の内部状態 $X_i = \{x_{i_{jk}}\}$ を $\{0, 1\}$ のランダムに，外部状態 \vec{x}'_i を $[-V_{max}, V_{max}]$ の範囲内でランダムに初期化する．

4.3.2 評価値の計算

一定時間間隔 T におけるトラフィックのフローテーブルを作成し，そのフローテーブルに基づいてエージェント i の現在の外部状態 \vec{x}'_i を評価する．ここで“フロー”とは送信元/宛先 IP アドレスや送信元/宛先インターフェースの組み合わせで識別するセッションであり，L3 スイッチなどの中継機器で取得できる情報である．このフロー情報を用いることでネットワーク内のトラフィックを監視できる．その代表的なものとしては NetFlow⁶⁾ などがある．本稿では，フロー情報から式 (5) を用いて求められる冗長発生率 (*redundancy ratio*) を評価値とした．

$$\text{red}(\vec{x}'_i) = \frac{\text{redundant_flow_count}}{\text{all_flow_count}} \quad (5)$$

ここで， all_flow_count はフローテーブル内の全フロー数， $\text{redundant_flow_count}$ はフローテーブル内の冗長経路が発生したフロー数とする．なお，現状では，冗長経路の長さやホップ数，パケットのデータ量は考慮していない．

例えば，図 3 の構成で表 1 のようなフローが一定時間間隔 T で発生したとする．この時にエージェント i の外部状態 \vec{x}'_i を内部状態 X_i に写像したものが $S_i = (1, 1, 2)$ を表しているとすると，A-E1 と B-E1 の間の通信においては，冗長が発生しないが，A-E2 と C-E3 の間の通信及び，B-E2 と C-E2 の間の通信においては冗長が発生する．従って，評価値は $\text{red}(\vec{x}'_i) = 2/3 \approx 0.66$ となる．本稿の目的は冗長トラフィックの発生を最小化することであるので $\text{red}(\vec{x}'_i)$ は小さいほうがよい．

4.3.3 最適値の更新

もし，上記で求めた $\text{red}(\vec{x}'_i)$ が $\text{red}(\vec{p}'_i)$ (エージェント i の探索履歴での最適解を現在の評価関数で再評価したもの) よりも評価が良ければ， $\vec{p}'_i \leftarrow \vec{x}'_i$ として，自身の最適解を更新する．各エージェントは隣接エージェントと $\text{red}(\vec{p}'_i)$ を通信し合い，その中で最適値となる \vec{p}'_i を \vec{p}'_g として更新する．ここで，以前の最適解を現在の評価関数で再評価することによって，環境の動的な変化にも対応し，局所最適解に陥らないようにしている⁷⁾．

4.3.4 状態と変位量の更新

上記で，探索空間における現在の外部状態 \vec{x}'_i ，履歴の中での最適状態 \vec{p}'_i ，隣接エージェ

ントの履歴の中での最適状態 \vec{p}_g を求めることができた。これらを、式 (1) を用いて、状態変位量 \vec{v}_i と現在の外部状態 \vec{x}_i を更新する。この更新によって PSO では各エージェントが 1 世代進化したと考え、第 4.3.2 節に戻り探索を繰り返す。

5. 実験

5.1 シミュレーションモデル

本稿で用いたシミュレーションのネットワーク構成は、図 3 のように、一つのコアスイッチに複数のエッジスイッチが接続し、エッジスイッチに複数の VLAN が接続しているものとする。全てのスイッチでフロー情報を取得し、そのデータを定期的に管理用のマシンに集めているとする。VLAN のトラフィックモデルとしては²⁾と同様に、環境が変化しないランダムモデル、特定モデルと環境が高頻度で変化する動的モデル 1、環境が低頻度で変化する動的モデル 2 の 4 つを使用した。

5.2 実験手法

以下の 2 つの冗長トラフィック削減手法を用いて比較実験を行った。

(1) BPSO²⁾

一定時間間隔毎に離散 PSO の手法を用いてルーティングポイントを決定する。

(2) AMPSO

提案手法。一定時間間隔毎に第 4 節で述べた AMPSO の手法を用いてルーティングポイントを決定する。

本実験では、VLAN 数を 10、ルータ数を {3, 4, 5, 6, 7, 8, 9, 10, 20, 50} と変動させてシミュレーションを行った。ここで、トラフィックモデルは最適解が変動しない特定モデルを用いた。ルータ数の違いにより、各手法が最適解を発見するまでに要したステップ数の変化を比較した。

また、VLAN 数 50、ルータ数 10 の環境で、ランダムモデル、特定モデル、動的モデル 1、動的モデル 2 を用いてシミュレーションを行った。本実験では、1 ステップを 1 秒と想定し、86400 ステップ (24 時間) シミュレーションを行った。ルーティングポイントの移動間隔は 1800 ステップ (30 分) とした。評価指標はルーティングポイントを移動させてから、次に移動するまでの間 (今回の実験では 1800 ステップ) における冗長トラフィック発生率 (式 (5)) とした。

BPSO, AMPSO では 60 ステップ毎に 10 世代進化を行った。PSO における各パラメータは $N_p = 30, \omega = 0.7298, \chi_1 = \chi_2 = 1.49618, V_{max} = 4$ を用い^{(3),(4)}、群を構成している

トポロジーは VON NEUMANN を用いた⁸⁾。

5.3 実験結果

図 5 に AMPSO を用いた手法での最適解発見に要するステップ数の結果を、図 6 に BPSO を用いた手法での最適解発見に要するステップ数の結果を示す。これらの図は、5 回の実験における平均値と標準偏差を表している。また、BPSO を用いた手法では、ルータ数が 9 以上の場合は最適解が発見できなかった。図 6 より、BPSO を用いた手法では、ルータ数の増加とともに最適解に発見に要するステップ数が増加していることがわかる。それに対して、図 5 より、AMPSO を用いた手法では、ルータ数が増加しても、最適解発見に要するステップ数はさして変化していないことがわかる。また、BPSO と比較して、最適解発見までのステップ数が約 1/10 に削減されていることがわかる。

図 7 にランダムモデルの結果を、図 8 に特定モデルの結果を、図 9 に動的モデル 1 の結果を、図 10 に動的モデル 2 の結果を示す。ここで、図 9、図 10 のグラフの上部に示してある特定 n は、その時間のシミュレーションモデルがエッジルータ n に接続されているサブ VLAN 同士が高頻度で通信している特定モデルを、Rand はランダムモデルであることを示している。

図 7、図 8 より、環境が変化しないモデルにおいて、BPSO は最適解を発見できていないが、AMPSO は最初の 1800 ステップで最適解を発見できていることがわかる。図 9 より、環境が高頻度で変化するモデルにおいて、BPSO は最適解を発見できていない。それに対して、AMPSO は環境の変化に追従し最適解を発見できている。しかし、最適解を発見する前に環境の変化が起きる場合もある。図 10 より、環境が低頻度で変化するモデルにおいて、BPSO は最適解を発見できていないが、AMPSO は環境の変化に追従し、最適解を発見できている。

この実験結果より、本提案手法である AMPSO を用いた手法は従来手法 BPSO と比較してスケラビリティが向上したと言える。

6. おわりに

本稿では、AMPSO を用いた VLAN 環境下における動的中継点制御のスケラビリティの向上を示した。提案アルゴリズムでは従来の AMPSO を拡張し、探索空間を 3 次元に削減した。また、エージェントの内部状態を VLAN 環境に適用できるように再定義した。

提案手法の有効性を確認するために、シミュレーション環境において複数のトラフィックモデルを作成し、提案手法と比較手法を使用して実験を行った。その結果、既存手法では最

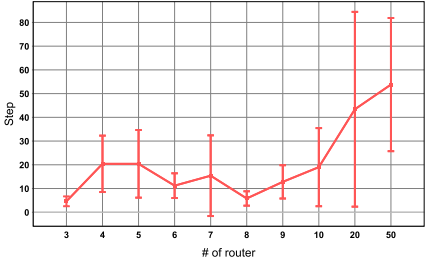


図 5 ルータ数の違いによる最適解発見に要するステップ数 (AMPSO)
Fig. 5 The number of steps required to find the optimal solution by the difference between the number of routers (AMPSO).

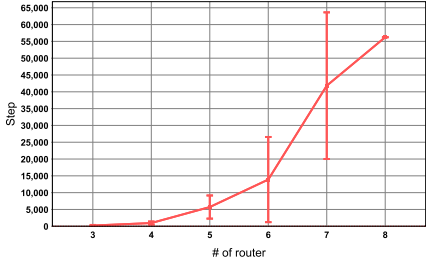


図 6 ルータ数の違いによる最適解発見に要するステップ数 (BPSO)
Fig. 6 The number of steps required to find the optimal solution by the difference between the number of routers (BPSO).

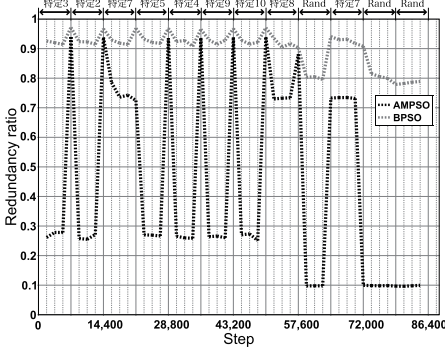


図 9 実験結果 (動的モデル 1)
Fig. 9 Experimental result (dynamic model 1).

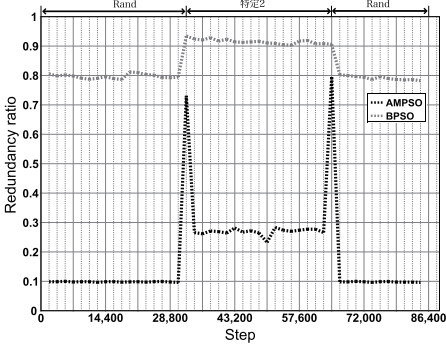


図 10 実験結果 (動的モデル 2)
Fig. 10 Experimental result (dynamic model 2).

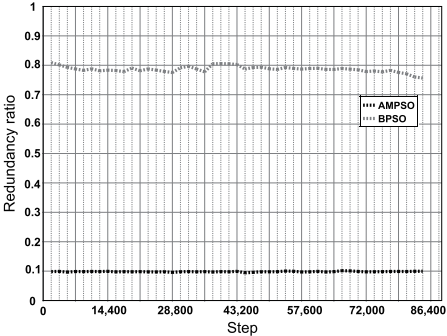


図 7 実験結果 (ランダムモデル)
Fig. 7 Experimental result (random model).

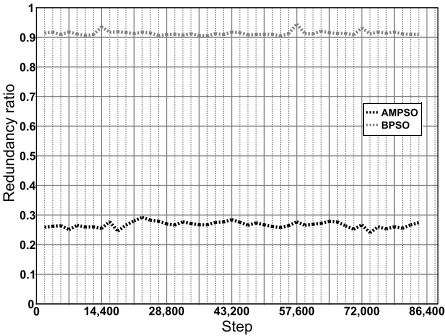


図 8 実験結果 (特定モデル)
Fig. 8 Experimental result (specified model).

最適を発見できなかった高次元の問題に対しても提案手法では最適解を発見可能であり、スケラビリティが向上したことを示した。

今後の課題としては、実データを用いた提案手法の評価が挙げられる。また、今回の実験では一定時間間隔のフローを用いて学習したが、時間や曜日などといった大域的なフローの傾向の学習と組み合わせることが考えられる。今回の実験では、全てのスイッチで取得するフロー情報は一か所に集約されると想定した。しかし、ネットワークの規模に比例して計算時間が増大していくことが予想できるため、フロー情報を分散して収集し、抽象的な情報の

みを一か所に収集する、階層的分散モデルでのアルゴリズムを考案する必要がある。

参考文献

- 1) Kennedy, J. and Eberhart, R.: Particle swarm optimization, *Proc. of the IEEE Int. Conf. on Neural Networks*, Vol.4, pp.1942-1948 (1995).
- 2) 高橋謙輔, 阿部洋丈, 廣津登志夫, 菅原俊治: 群知能を用いた分散仮想ルータのための動的中継点制御, 合同エージェントワークショップ&シンポジウム (2009).
- 3) Pampara, G., Franken, N. and Engelbrecht, A.: Combining particle swarm optimisation with angle modulation to solve binary problems, *The IEEE Congress on Evolutionary Computation*, Vol.1, pp.89-96 (2005).
- 4) Clerc, M. and Kennedy, J.: The particle swarm - explosion, stability, and convergence in a multidimensional complex space, *IEEE Transactions on Evolutionary Computation*, Vol.6, No.1, pp.58-73 (2002).
- 5) Kennedy, J. and Eberhart, R.: A discrete binary version of the particle swarm algorithm, *IEEE Int. Conf. on Systems, Man, and Cybernetics*, Vol.5, pp.4104-4108 (1997).
- 6) NetFlow, http://www.cisco.com/en/US/products/ps6601/products_ios_protocol_group_home.html.
- 7) Carlisle, A. and Dozier, G.: Adapting Particle Swarm Optimization to Dynamic Environments, *Proc. of Int. Conf. on Artificial Intelligence*, pp.429-434 (2000).
- 8) Kennedy, J. and Mendes, R.: Population structure and particle swarm performance, *IEEE Int. Conf. on E-Commerce Technology*, Vol.2, pp.1671-1676 (2002).