

ポリシーに基づく IGP コスト決定手法の提案

夏山 京大[†] 衛藤 将史[†] 門林 雄基[†] 山口 英[†]

[†] 奈良先端科学技術大学院大学 情報科学研究科

〒 630-0192, 奈良県生駒市高山町 8916-5

E-mail: (keita-na,masashi-s,youki-k and suguru)@is.aist-nara.ac.jp

あらまし ネットワークを構成しているルータやレイヤ 3 スイッチには多種多様なベンダの製品が存在し、設定方法はベンダ毎に異なっている。そのため、これらの作業を行なうには管理者がベンダ独自のルータの設定コマンドを把握している必要がある。ネットワークが複雑化する一方で、機器の設定は管理者が独自に作成したスクリプトや手作業によって行なわれているため、管理者の設定ミスによるネットワーク障害が頻発している。そこで、この問題を解決するために、Interior Gateway Protocol(IGP)における、Open Shortest Path First(OSPF) [1] に関してはリンクのコストを最適化するアルゴリズムや、アドレス集約のための方法が提案されている。しかし、現在提案されているアルゴリズムのほとんどはトラフィックエンジニアリングの視点でコスト最適化を計る研究が主で、Internet Service Provider(ISP) や企業ネットワーク設定におけるポリシーを反映しているとはいえない。また、このようなアルゴリズムを用いた制御を用いる場合であっても、ルーティングプロトコル等のネットワークの専門的知識を必要とする。そこで、ネットワーク設定に必要な知識を削減し、設定における人的コストを軽減することで管理、運用コストの減少を目指す。本論文では、管理者のポリシーを反映したネットワーク設定を支援し、設定者の人的コストを軽減するために必要なアーキテクチャを提案する。

キーワード IGP, OSPF, リンクコスト, トラフィックエンジニアリング, ISP

A Proposal of Policy-Based Cost Algorithm for IGP

Keita NATSUYAMA[†], Masashi ETO[†], Youki KADOBAYASHI[†], and Suguru YAMAGUCHI[†]

[†] Graduate School of Information Science, Nara Institute of Science and Technology

8916-5 Takayama-cho, Ikoma-shi, Nara, 630-0192, Japan

E-mail: (keita-na,masashi-s,youki-k and suguru)@is.aist-nara.ac.jp

Abstract Network configuration format may have different way to set-up its configuration. And, it is also necessary for the administrator to decide the configuration parameter of those routers or switches when configuring a network. Generally these parameters are calculated based on the network policy. As the size of a network is becoming larger, the network is becoming more complex. Therefore, this complexity may cause the network administrator prone to make miss configuration of network parameter, which finally may cause a network trouble or faults. In order to avoid this happens, it is desirable that the architecture based on the network policy calculates the network configuration parameter. It can reduce miss configuration as well as eliminate network faults. In this paper, we propose network configuration architecture which support the network administrator based on the network policy. Furthermore, we focus on the cost setting in the OSPF networks, and produce a cost setting algorithm in OSPF.

Key words network configuration, IGP, OSPF, cost, ISP,

1. 背景

近年、インターネットの普及と共にネットワークの規模は日々肥大化している。ネットワークが肥大化するにつれて、ネットワークの設計や管理も複雑化し、管理者が設定しなければならないルータの台数や設定事項が増加する。また、現在のネットワーク管理では様々なルーティングプロトコルが動作しており、管理者は各プロトコルの知識やネットワークのしくみを熟知している必要がある。また、セキュリティ管理、ルータのファームウェアのアップデートなどの管理運用コストは無視できない。そして、管理者の人的ミスやセキュリティアップデートが遅れたために生じるネットワーク障害により ISP や企業のサービスが停止すると金銭的被害や信頼の損失につながってしまう。よって、いかに人的ミスを減らし、オペレータの負荷を減らすかということが今後のネットワーク管理では重要である。現在のネットワーク管理では、Simple Network Management Protocol (SNMP) [2] のような管理用プロトコルを用いた管理支援ツールが利用されている。現存の SNMP を用いたネットワーク管理ツールでは、人が GUI によるアプリケーションやコマンドラインから操作することでベンダ独自性やコマンドの隠蔽を行い、設定のための操作方法の負荷は軽減している。しかしながら、ネットワークにおけるセキュリティや経路選択といったポリシーから各パラメータを決定する作業は管理者が自分で行う必要がある。このような設定時の操作の複雑さは管理者の設定ミスを招き、管理者の意図しない挙動や、ネットワーク障害を引き起こすかも知れない。そのため、管理者がトラフィックを通したい経路や、ネットワークの構成等のポリシーを元に管理者が各設定パラメータを自分で計算し設定することは好ましくない。そこで、管理者の理解しやすいポリシーを入力し、詳細の設定はシステム側が解決することで設定時の管理コストを抑え、人的ミスを減らすべきである。そこで、我々はネットワーク設定の中で IGP において広く用いられているルーティングプロトコルである OSPF の設定の自動化の為に構造を設計し、OSPF におけるコスト設定手法について提案する。

2. ネットワーク設定支援の全体構造

ネットワーク設定では、管理者のポリシーをネットワーク設定パラメータへと変換することが重要な役割であり、現在は管理者の知識や経験を元にポリシー

から各ネットワークの設定へと反映している。そこで、管理者が直接操作する入力は、XML データ構造によって表現された管理者の理解しやすい入力表現を規定する。入力されたパラメータからネットワークの各設定に必要なパラメータとして変換し、設定機能へと入力する。各機能は必要な入出力パラメータを規定したモジュールとして組み込むことで、管理者は実際のネットワークの動作を入力として与えることで実際のパラメータ決定方法等を熟知する必要なく、最終的なネットワークの設定パラメータを得ることが出来る。そこで、現在の IGP で幅広く用いられている OSPF における図 1 のような全体のアーキテクチャ設計を行った。

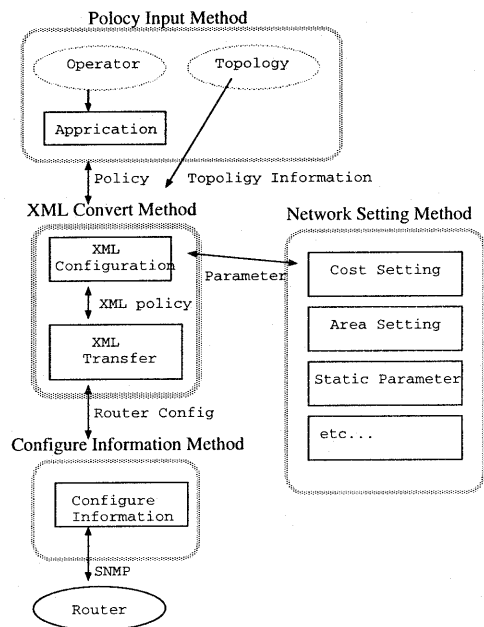


図 1 提案機構の各ブロック構成

2.1 OSPF 設定支援のために必要な機能

OSPF の設定としては、以下の設定項目を考慮する必要がある。各設定では、他の設定に影響を及ぼさないパラメータのみならず、他のルータとの整合性が必要なパラメータや、他の設定パラメータと相互に影響を及ぼすパラメータが存在する。よって、各設定パラメータを設定するだけでなく、ネットワーク全体で整合性が取れているか調べる必要がある。

- 各リンクコスト、経路情報の集約
ルータ間のリンク帯域や、通信遅延、金銭的成本、リンクの信頼制、経路の集約状態を元に設定され、各

ルータからの他のルータへの通信経路が決定される。

- エリアの種類及び分割
管理上の理由や地理的条件等によりエリアの分割の仕方や設定するエリアの種類が決定される。

- IP アドレス、ルータ ID、ルータの優先度
代表ルータの決定や経路集約を考慮に入れた決定がされる。

- Hello Interval や LSA パケットの更新時間
同一ネットワーク内のルータで整合性を必要とするパラメータの設定が行なわれる。

- ABR や ASBR の設定、外部経路広告
エリア間の経路広告や外部経路の設定が行なわれる。前述したようなパラメータは各自律システム (AS) ごとに管理者のポリシーに沿ったパラメータの計算を行ない、設定することが一般的である。ネットワーク設定に必要なポリシーは、複数のパラメータ設定に影響を及ぼす可能性がある為、ネットワーク設定に必要なパラメータを抽出し、ポリシーから変換するパラメータを構造的に設計する必要がある。

2.2 各 Method の機能

全体のアーキテクチャを構成する要素として以下のような機能に分けることが出来る。

- Policy Input Method
GUI等を用いたアプリケーションを通してネットワークのポリシーを入力する。また、Link Management Protocol, Cisco Discovery Protocol(CDP), Extreme Discovery Protocol(EDP) といったプロトコルを用いて接続されているネットワークトポロジ情報を収集する。

- XML Convert Method(XML Configuration)
入力されたポリシーやトポロジ情報を元に XML 記述のポリシーに変換する。又、XML に変換された情報から各ネットワーク設定モジュールに必要なポリシーを抽出し、各モジュールに必要なパラメータに変換した後各モジュールへと伝達する。

- Network Setting Method
XML Convert Method から得られたパラメータを各モジュールに入力し、ネットワークの設定パラメータとして XML Convert Method へ出力する

- XML Convert Method(XML Transfer)
導出されたルータの設定パラメータを XML データ構造に変換し、XSLT を通して各ルータに必要な MIB 情報へと変換する。

- Configure Information Method
各設定項目をルータへ配布する。

2.3 Network Setting Method

図 1 における各 Method の中でも Network Setting Method は入力されたポリシーから OSPF の各設定パラメータへ変換する為の機能を持つ。Network Setting Method では XML Convert Method によって得られたパラメータを基に OSPF に必要な各設定パラメータを決定する。今回、本提案ではコスト設定の支援のために必要なアルゴリズムを検討する。

2.3.1 コスト設定を左右する要素

現在のネットワークは課金システムとして定額制、通信距離による課金、使用したトラフィック量による課金等回線購入時の契約により様々な利用形態がある。そこで、管理者の求めるネットワークに応じた経路制御に適応したコスト設定のために以下のような要素を考慮に入れた、アルゴリズムを検討する。

- 回線の金銭的成本

回線の金銭的成本としては定額制の回線と従量課金制、またネットワークがある利用率を越えると定額制から従量課金制へと変わるサービスがある。定額制サービスや従量課金サービスそれぞれを見てもサービス料金設定方法が様々である。

- 安定性、通信状況の変化

安定性にはリンクが安定しているか、ルータまたはルータのインタフェースが安定しているかが考えられる。

- 回線の帯域、トラフィックの負荷

回線の帯域にはインタフェースの帯域だけでは判断できず途中の経路上で帯域が狭くなるような状況が考えられる。

- 通信の遅延

Round-trip Time(RTT)

3. コスト決定アルゴリズムの関連研究

OSPF のコスト設定のアルゴリズムとして以下のようなものが先行研究として挙げられる。我々は、現在の管理者がネットワーク設定時に導入コストや遅延、帯域を考慮したコスト設定が容易に設定出来るように前章で述べたような必要な制約条件を満たすコストの設定を行なうためのアルゴリズムを必要とする。そこで、どのような経路選択を行ないたいかという制約条件をコストへ変換する為に必要なコスト設定アルゴリズムが必要である。第一にそのようなアルゴリズムが先行研究として存在するアルゴリズムで適用できるか考察する。コスト設定の先行研究を次に示し、我々の必要とする制約条件を満たせる

かどうかを考察する。

3.1 Greedy Algorithm

与える制約条件をプライオリティに変換し、その条件を満たすようなリンクコストの組合せを逐次検索する。アルゴリズムは単純になるが、時間計算量が膨大になると考えられるので、ネットワークが大きくなると現実的な時間で解くことが困難と考えられる。

3.2 発見的アルゴリズム

● 発見的アルゴリズムを用いたコスト最適化アルゴリズム

B.Fortz [3] による OSPF のコスト最適化アルゴリズムでは、トラフィックエンジニアリングの視点でみた発見的アルゴリズムによる最適コスト決定を行っている。必要なパラメータとしてルータの台数、リンク、リンク帯域幅、始点から終点へ向かうトラフィックの利用帯域、現在のリンクコスト、ハッシュ関数等を用いる。このアルゴリズムは、ネットワークの利用効率を向上させるようなリンクコストを求めるといふ点では有効であるが、リンクコストの変化数に制約がなく多くのリンクがコストの変化を必要とすることがあり、特定のトラフィックフローを意図して調節することは出来ない。

● B.Fortz の発見的アルゴリズムの改良アルゴリズム

Huan Pham [5] による OSPF のコスト最適化アルゴリズムは Fortz の local search heuristic アルゴリズムにおいて元となるリンクコストから変化したリンクコスト数に制限を設けることでリンクコストの大幅な変更を防ぐ。これにより、トラフィック要求の増大によるオペレータの人的コストの軽減を計っている。Fortz 同様特定のトラフィックフローを調節する事は出来ない。

3.3 遺伝的アルゴリズム

Mulyana [6] による最適化アルゴリズムは、トラフィックエンジニアリングの視点でみた Genetic Algorithm (GA) による最適コスト決定を行っている。このアルゴリズムでは、各ルータの台数とリンク、及びリンクの帯域幅、始点から終点へのトラフィックの帯域利用率と利用率の平均、現在の各リンクコストを用い変更前のコストと変更後のコストの類似性を制約条件として考慮することによってコストの変化を最小にし、コスト再計算によるネットワークの不安定を防ぐ。アルゴリズムから導かれるコストはリンクの平均利用率と最大利用率を抑えるコストが出力される。既存の GA によるコスト設定に比べ経路情報

伝搬の収束時間が改善されているが、ネットワーク利用率を情報として取得する必要がある。

3.4 既存アルゴリズムの総括

前述した全てのアルゴリズムは、トラフィック・エンジニアリングを目的とした動的なコスト設定を行っており、常に各リンクのトラフィックフローごとのネットワークの利用率を監視する必要がある。また、全てのアルゴリズムにおいて目標はネットワークの利用効率の向上やトラフィックの負荷集中の軽減目的としており、我々が目的とする、金銭的成本や用途に応じて選択したい経路を指定するようなコスト設定のためのアルゴリズムではない。

4. 提案手法

我々が提案するコスト決定アルゴリズムは優先したい経路をエンド to エンドで選択し、また特定のリンクが切れた時の代替経路を選択することで、そのような制約条件を満たすコストを決定する。

例として、図 2 のようなネットワークを考える。A - E - F - H の経路は定額サービスの回線を利用しており、A - D - G - H の回線は従量課金サービスの回線を利用してしている。管理者は、通常は定額制の回線を用いたく、障害時のみ代替経路として従量課金制の回線を用いる場合 図 2 のような経路設定を行いたい。そこで、管理者が要求するポリシーとネットワーク情報を元に 図 2 のような経路決定を行ない、全てのポリシーが実現可能であればそれを満たすようなコストを導出する。

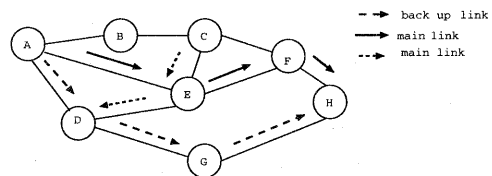


図 2 トポロジ及び要求される経路

4.1 コスト設定アルゴリズム

コスト計算の方法は以下のようにする。

(1) 管理者から入力されたそれぞれの優先経路に対し、図 3 で示すようにそれぞれの始点、終点を判別し、図 4 に示すようにバックトラック法を用いて到達可能な全ての経路を探索する。

(2) 図 5 で示すように代替経路及び、優先経路、始点と終点に到達可能な全ての経路が 優先経路の到達コスト < 代替経路の到達コスト < その他の到達可能な経路のコストを満たすようなコストを求める。

コストの算出には得られた制約条件式から線形計画問題として求める事が出来る。

```
void get_endpoint
{
    for (すべての制約条件について) {
        for (全ての行ノード数) {
            if (p_link->priority_link[i][j][k] == 1) {
                各列の要素が1の数を数える;
            }
        }
        if (同一列上の要素の合計数が1なら) {
            (現在の列が優先経路情報の端点である
             制約条件の始点と終点に登録(始点 < 終点))
        }
    }
}
```

図3 始点、終点を求める関数

```
void get_arrival_path
{
    for (全ての優先経路情報について) {
        if (現在探索中のノードが終点であれば) {
            (通過した経路情報を現在探索している優先経路の始点、終点をもつ
             経路情報として追加)
        }
        else {
            (バックトラック法で全ての経路を探索)
            if (過去に通ったノードであるならば) {
                (一つ前のノードに戻る)
                (戻ったノードに探索していない経路があるか調べる)
            }
            else (探索済みでない経路があるならば) {
                (探索していない経路を通り次のノードへ移る)
            }
        }
    }
}
```

図4 始点から終点に到達可能な全経路を求める関数

```
void get_constraint
{
    for (;) {
        (制約条件違反のフラグを初期化)
        for (全ての優先経路情報について) {
            priority_cost = 優先経路のコストを計算
            for (優先経路の始点、終点へ到達する経路について) {
                (other_cost = 到達可能な経路のコストを計算)
                for (優先したい経路の代替経路について) {
                    (backup_cost[k] = k番目の代替経路のコストを計算)
                    (priority_cost[k] < backup_cost[k] < other_costをみたすか)
                    if (制約条件の関係式を満たさなければ) {
                        (制約条件違反のフラグ設定)
                        break;
                    }
                }
                if (フラグについて制約条件違反していれば) break;
            }
            if (フラグについて制約条件違反していれば) break;
        }
        (全ての関係式を満たしていればコスト計算終了)
        (コスト行列の再計算)
    }
}
```

図5 コスト決定関数

しかし、全ての優先経路情報とネットワーク上の

全てのノードにおいて計算を行なうとノードが増えるにつれ計算量が非常に増大する。そこで上の条件に次のような条件を加え、コストの決定に与えられた制約式からシンプレックス法等の既存の線形計画法アルゴリズムを適用することでコストの全探索を行なう場合に比べ計算量を減らすことが可能である。

4.1.1 計算量軽減のためのアプローチ

まず、ノードが単一リンクのみの制約条件制約条件を持っているノードはそのノードを制約条件式の対象となる行列から除去できる。なぜならば、制約条件のないリンクには、求められた全経路コストの最大値+1のコストを割り当てることによりその他の優先経路より優先される可能性はない。そのため、単一の制約条件しか持たないノードから得られる優先したいリンクは一本しかないため、1を割り当てることでその他の経路より必ず優先される。つまり、コスト設定の為の計算に必要な経路情報は、優先経路を全て網羅したリンク情報から、単一のリンクのみから成る優先経路を除いた閉路のみのリンク情報を考慮するだけでよい。具体的には、以下のようになる。

(1) 図6のように全てのノードとリンクの状態とは別に優先したい経路情報のみからノードとリンクの接続状態のトポロジを作る。

(2) 図7のように、作成されたリンク状態行列から一本の経路で繋がっているようなノードを削除する。単一のリンクでのみ繋がったノードがなくなるまでこの操作を繰り返す。

(3) 単一リンクで繋がっているノードを全て削除した状態で閉路のみで作成されたノードの接続状態のトポロジ及び、そのトポロジ上での優先経路情報から前述のコスト設定を行なう。

(4) 閉路のみの優先リンクのコストが決定されると優先経路情報から削除された単一リンクの優先したいリンクにコスト1を割り当てる

(5) 求めた経路情報で最も大きい経路コスト*優先したい経路の最大ホップ数+1のリンクコストを残りのリンクコストに割り当てる。

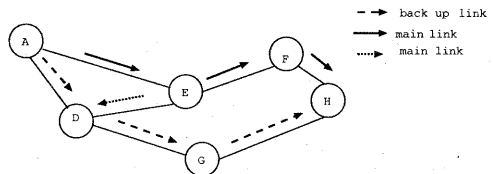


図6 単一リンクの優先経路情報を除いたトポロジ

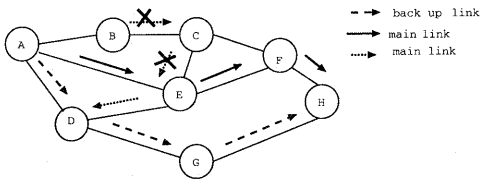


図7 単一リンクの優先経路の削除

rithm Approach for OSPF Weight Setting Problem”

5. ま と め

本提案方式により、優先したい経路及び代替経路を指定することにより OSPF のエリア内におけるリンクコストを決定することができる。しかし、ノード数及び制約条件が増える事で計算量が増大することは否めない。また、既にリンクコストが決定されているネットワークに制約条件を追加した場合、本提案手法ではリンクコストが大幅に変わってしまい、リンクコストを変更した時のネットワーク収束時間に影響を与える恐れがある。

今回提案した方法により、管理者がどのような経路を意図しているかという情報から、そのような条件を満たすリンクコストを導出することが可能になった。しかし、現在の方法ではノードが増えるにつれて計算量が非常に増大する恐れがある。そこで、より計算量を減らす方法を今後考察すると共に、新たに条件変更が加わった時、既存のコストを出来る限り変更せず、コストを決定する方法が必要である。また、ルータ間のリンクに与えられた情報(金銭的成本、回線、ルータの信頼性、通信遅延、帯域)及び管理者が意図するポリシー(安い回線を用いたい、通信遅延を少なくしたい)から、管理者の優先したい経路情報へと変換し、本提案コスト設定方法からリンクコストを決定するための変換機構を作ることで、管理者はポリシーを入力するだけでコスト設定を自動で行なうことが可能である。

文 献

- [1] John T.Moy, “OSPF Version 2”. RFC 2328, April 1998
- [2] J. Case, M. Fedor, M. Schoffstall, J. Davin, “A Simple Network Management Protocol (SNMP)”. RFC 1157, May 1990
- [3] Bernard Fortz and Mikkei Thorup, “Internet Traffic Engineering by Optimizing OSPF weights”. In Proceedings of IEEE INFOCOM 00, Tel Aviv, Israel, Mar. 2000.
- [4] Ashwin Sridharan, Roch Guerin, “Internet Traffic Engineering by Optimizing OSPF weights”
- [5] Huan Pham Bill Lavery, “An Improved Method for Determining Link Weights for Optimising OSPF Routing”
- [6] E.Mulyana Killat. U, “A Hybrid Genetic Algo-