

## 複数ノード故障に対する IP 高速迂回法の検討 Investigation of an IP Fast Reroute Method for Multiple Node Failures

沼田 直人<sup>†</sup> 樽谷 優弥<sup>†</sup> 福島 行信<sup>†</sup> 横平 徳美<sup>†</sup>  
Naoto Numata Yuya Tarutani Yukinobu Fukushima Tokumi Yokohira

### 1. はじめに

現在のインターネットでは、故障が発生したときには迂回経路の再計算を行う。しかし再転送が開始するまでに時間を要するため、高速復旧を実現する IP 高速迂回法の一つとして、複数のリンク故障を迂回する ADST (Arc-Disjoint Spanning Trees) 法[1]が提案されている。本研究では ADST 法を拡張し、複数のノード故障を迂回する NDST (Node-Disjoint Spanning Tree) 法を提案する。

### 2. ADST 法

#### 2.1 ADST の定義

ノードとリンクからなる無向グラフとして与えられるネットワーク形態  $G$  (最小次数を  $k$  とする) において、ノード  $a$ - $b$  間のリンクを  $a$  から  $b$  への有向アークと  $b$  から  $a$  への有向アークに分けて得られる有向グラフ  $G'$  を考える。このとき、 $G'$  の各ノード  $D$  について、以下の条件 (A1) と (A2) を満足する  $k$  本の  $G'$  のスパニングツリーのそれぞれを ADST (Arc-Disjoint Spanning Tree) と呼ぶ (以下、ノード  $D$  のことを、その ADST の根と呼ぶことがある)。

(条件 A1) 各 ADST において、ノード  $D$  の入力次数は 1、出力次数は 0 であり、他のノードの入力次数は 0 以上、出力次数は 1 である。

(条件 A2) 任意の 2 つの ADST において、同じ有向アークは持たない (Arc-Disjoint 性)。

#### 2.2 ADST 法における迂回

ADST 法では、ネットワーク設計段階で  $k$  本の ADST をあらかじめ作成しておき、ネットワークの運用段階では、以下のように、最大で  $k-1$  本のリンク故障を迂回できる (図 1 参照)。

そのパケットの送信元ノードを  $S$ 、送信先ノードを  $D$  とする。 $G$  においてノード  $a$ - $b$  間のリンクが故障した場合、 $G'$  における  $a$  から  $b$  への有向辺と  $b$  から  $a$  への有向辺のいずれも使用できなるとみなす。

(1) 任意の 1 つの ADST (Tree1) を選ぶ。この ADST において  $S$  から  $D$  への経路はリンク故障がない場合、図 1(a) のように  $S$  から 0 本以上の有向アークを経由してノード  $a$  に到達し、次にノード  $b$  に直接到達し、その後  $D$  に到達するものとする。この ADST を使用して、パケットをノード  $a$  まで送信し、ノード  $b$  に送信しようとするが、ノード  $a$ - $b$  間のリンクが故障しているためにノード  $a$ - $b$  間の有向アークが使用できなかったとする。

(2) このとき、 $b$  から  $a$  への有向アーク ( $a$  から  $b$  への有向アークと逆向きのアーク) を持っている別の ADST があればその ADST を選択し、そのような ADST がなければまだ使用していない ADST を任意に選択する。ここで選択した ADST を Tree2 とする。

(3) この Tree2 においてリンク故障がない場合、 $a$  から  $D$  への経路は、 $a$  から 0 以上の有向アークを経由してノード  $c$  に到達し、 $c$  からノード  $d$  に直接到達した後、 $d$  から  $D$  に到達するものとする。この ADST を使用して、実際にノード  $c$  まで到達したとする、次に、ノード  $d$  に行こうとするが、 $c$ - $d$  間のリンク故障のために、 $c$ - $d$  間の有向アークが使用できなかったとする。

(4) このとき、 $d$  から  $c$  への有向アーク ( $c$  から  $d$  への有向アークと逆向きのアーク) を持っている別の ADST があればその ADST を選択し、そのような ADST がなければまだ使用していない ADST を任意に選択する。ここで選択した ADST を Tree3 とする。

(5) 以下、上述した (3) と (4) を繰り返すと、 $k-1$  本のリンク故障の場合、最終的に Tree  $k$  を使用してノード  $D$  に到達することができる。

ADST 法におけるキーアイデアは、上述の (2) や (4) において、故障している有向アークと逆向きの有向アークを持つ ADST があればそれを選んでいくことである。このようにすることで、逆向きのアークを持つ ADST の有無にかかわらず、選んだ ADST ではノード  $b$ - $a$  間 ( $d$ - $c$  間) の有向アークを通らないことを保証できる。すなわち  $a$ - $b$  間 ( $c$ - $d$  間) のリンクの故障を迂回することができる。

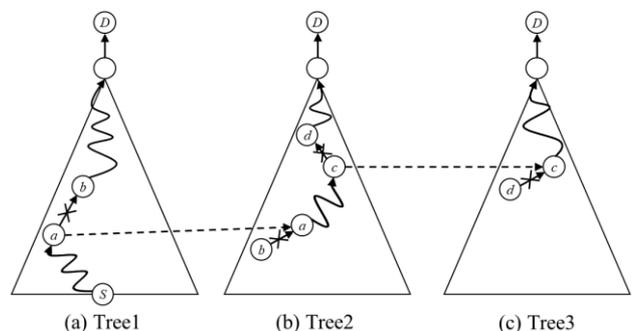


図 1 ADST 法における迂回

### 3. NDST 法

#### 3.1 NDST の定義

2.1 節で定義した  $G'$  の各ノード  $D$  について、以下の条件 (N1) と (N2) を満足する  $k-1$  本の  $G'$  のスパニングツリーのそれぞれを NDST (Node-Disjoint Spanning Tree) と呼ぶ。

(条件 N1) 各 NDST において、ノード  $D$  の入力次数は 1、出力次数は 0 であり、他のノードの入力次数は 0 以上、出力次数は 1 である。

(条件 N2) 任意の 2 つの NDST において、任意のノード  $u$  から  $D$  への有向経路は異なる中間ノードを経由する (Node-Disjoint 性)。すなわち、2 つの有向経路が同じ中間ノードを経由することはない。

### 3.2 NDST 法における迂回

NDST 法では、ネットワーク設計段階で  $k-1$  個の NDST をあらかじめ作成しておき、ネットワークの運用段階では、以下のように、最大で  $k-2$  個のノード故障を迂回できる (図 2 参照)。

(1') ADST 法の手順 (1) における「ADST」を「NDST」に変更し、「ノード  $a-b$  間のリンク故障」を「ノード  $b$  の故障」に変更した手順を実行する。

(2') まだ、使用していない NDST を選択する。ここで、選択した NDST を Tree2 とする。

(3') ADST 法の手順 (3) における「ADST」を「NDST」に変更し、「ノード  $c-d$  間のリンク故障」を「ノード  $d$  の故障」に変更した手順を実行する。

(4') まだ、使用していない NDST を選択する。ここで、選択した NDST を Tree3 とする。

(5') 以下、上述した(3')と(4')を繰り返すと、 $k-2$  個のノード故障の場合、最終的に Tree ( $k-1$ ) を使用してノード  $D$  に到達することができる。

ADST 法では、Tree  $k$  においてノード  $D$  まで到達できるが、NDST 法では 3.1 節の定義のままでは、Tree ( $k-1$ ) において  $D$  まで到達できる保証はない。すなわち、図 2 の場合、Tree3 において、ノード  $c$  から  $D$  に向かう途中で、Tree1 で迂回したノード  $b$  に再び出会う可能性がある。本研究ではこのような状況が起こらないような NDST を作成する方法を用いているが、その方法について紙面の都合上割愛する。

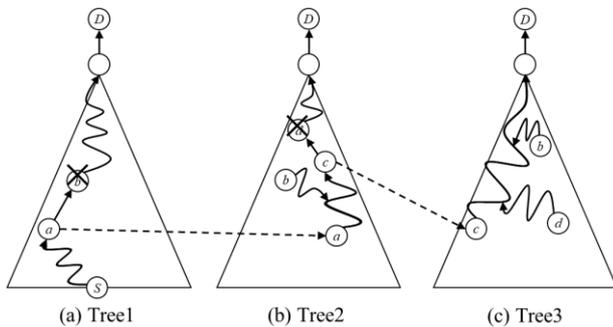


図 2 NDST 法における迂回

### 4. NDST の作成可能性

3.2 節で述べた Tree ( $k-1$ ) で  $D$  に到着できるような制限付き NDST (LNDST と呼ぶ) がどの程度作成されるかについて、シミュレーションにより調べた。作成するにあたって、使用するネットワーク形態を形態作成ツール BRITE[2] を使用し、Waxman (WAX) モデルと Barabasi-Albert (BA) モデルの 2 種類を作成した。両モデルとも、ノードに隣接するノードの最小次数を 4 として、10 ノードから 100 ノードのネットワーク形態をそれぞれ使用した。NDST においてノード  $D$  に隣接するノード (定義より一つである) を仮想送信先ノードと呼び、 $v_d$  と表す。上述したネットワーク形態で LNDST を作成できることを確認した。

次に、上述した LNDST がどのくらいの確率で作成されるのかを調べるために LNDST の作成成功率  $SR^n$  (Success Rate) を下の式(1)で定義する。

$$SR^n = \frac{1}{N} \sum_{n=1}^N \frac{SP_n}{P_n} \quad (1)$$

$P_n$  は  $v_d$  の組み合わせの総数であり、そのうち LNDST を作成できた  $v_d$  の組み合わせの数が  $SP_n$  である。 $D$  の次数が  $m > k-1$  であるとき、NDST に使用される隣接ノードの数は  $k-1$  個である。このとき、選ばれないノードは  $m-(k-1)$  個あり、ノードが選ばれる確率は  $mC_{k-1}$  である。LNDST の成功率を図 3 に示す。図 3 ではノードの最小次数が 4 であるネットワーク形態から各送信先  $D$  を根とした 3 本の NDST を作成した時の成功率を表しており、WAX モデルはノード数が少ないと成功率が低いが、ノード数を増やすと成功率が高くなる。一方、BA モデルはノード数が少ないと成功率が高いが、ノード数を増やすと成功率が下がる。これはネットワーク形態の次数に依存している。各ノード次数が平均ノード次数に近い WAX モデルでは、ノード数が増えると平均ノード次数も高くなるため、成功率が上がる。対して BA モデルは少数のノードに集中しているネットワーク形態であるため、ノード数が増えると次数の低いノードが多くなるため成功率は下がる。

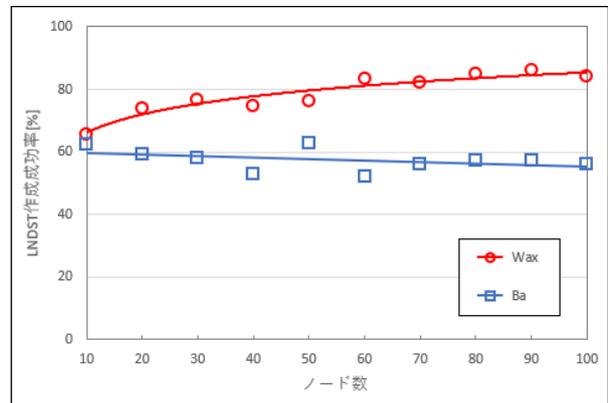


図 3 LNDST 作成成功率とノード数の関係

### 5. おわりに

本研究では、複数ノード故障を迂回できる NDST 法を提案した。

今後の課題は、より大きなネットワーク形態での NDST の作成をシミュレートし、NDST の作成可能なネットワーク形態の条件を明らかにすることである。

#### 参考文献

- [1] Theodore Elhourani, Abishek Gopalan, Srinivasan Ramasubramanian, "IP Fast Rerouting for Multi-Link Failures", IEEE/ACM TRANSACTIONS ON NETWORKING, Vol.24, No.5 (2016).
- [2] A. Medina, A. Lakhina, I. Matta, and J. Byers, "BRITE: An approach to universal topology generation," IEEE MASCOTS, pp. 346-353, pp. 473-486 (2001).