

ネットワークアベイラビリティの高速計算法

1H-5

巽 久行[†], 古屋 貴博[†], 小高 泰陸^{††}, 徳増 眞司[†]
[†] 神奈川工科大学 ^{††} ファーストタイム

1. まえがき

通信ネットワークにおける信頼性の問題は、グラフの問題として解析される。本報告は、その代表的な解析方法である状態空間分解法に基づいた新たなネットワークアベイラビリティの計算アルゴリズムを提案している。通常、状態空間分解法を行う場合、2分木を作成して分解を行うが、2分木の節点の数がネットワークのリンクの順番に依存しているため、一般には木を展開する際の計算量が指数関数的に増大することが知られている。そこで本報告では、木のサイズを削減し、かつアベイラビリティが高いネットワークを構成する部分を優先して2分木展開するアルゴリズムを作成した。このアルゴリズムは、ノードに始点からの前進ラベリングおよび終点からの後退ラベリングと呼ばれるラベリングを施し、どのノードを選択すればアベイラビリティが大きくなるかの順位を前もって登録することで、2分木展開の効率化を図った。提案したアルゴリズムは、ショート・パスが先に動作リンクとして展開されるように構成されているため、アベイラビリティに関して収束性が高く、また展開に関して効率の良い方法となっており、この事実を理論と実験により実証した。

2. アベイラビリティ

アベイラビリティ A は、信頼度と保全度を合成した尺度で、次式で定義される。

$$A = MTBF / (MTBF + MTTR) \quad \dots(1)$$

ここで、信頼度を表す $MTBF$ は平均故障間隔 (Mean Time Between Failures) であり、保全度を表す $MTTR$ は平均修理時間 (Mean Time To Repair) である。また、アベイラビリティと不稼働率の関係は次式で定義される。

$$A = 1 - \text{不稼働率} \quad \dots(2)$$

上式(1)で示したアベイラビリティ計算式は単一系のシステムには適用できるが、本研究で利用したようなメッシュ型ネットワークで適用するには複雑で計算が困難になる。そこで、メッシュ型ネットワークに有効な信頼性多項式 (下式(3)) を用いることで、式(1)で困難であったアベイラビリティの計算が容易になる。

$$P_W(N) = P(C) \cdot P_W(N|C) + (1 - P(C)) \cdot P_W(N-C) \quad \dots(3)$$

ここで、
 N : 対象とする1つのネットワーク
 C : ネットワーク N の構成要素である1つのサブ・ネットワーク

$P(C)$: C が稼働している確率

$P_W(N|C)$: N のうち C が稼働している場合の
アベイラビリティ

$P_W(N-C)$: N のうち C が故障している場合の
アベイラビリティ

である。式(3)は、ネットワークを排他的な2つの部分に分けることによりアベイラビリティを計算するもので、これを状態空間分解法と呼んでいる。

【例1】 図1の、4ノード・ネットワークについて、状態空間分解法でアベイラビリティを計算する。通信は発信ノード S から着信ノード T に対して行われ、 A, B は中継ノードを表す。各リンクにはリンク番号1~5が付けられている。図2の状態空間分解はネットワークのリンクの番号順に2分木を構成していく方法で、このリンクの選択を少し変えるとツリーの節点の数を減少させることが可能であり、計算量が減少する。しかしながら、このツリーの生成によってアベイラビリティが変化することは無い。結局、アベイラビリティの計算式 (信頼性多項式という) は、図2の“+系列”の要素をすべて足したものとなる。

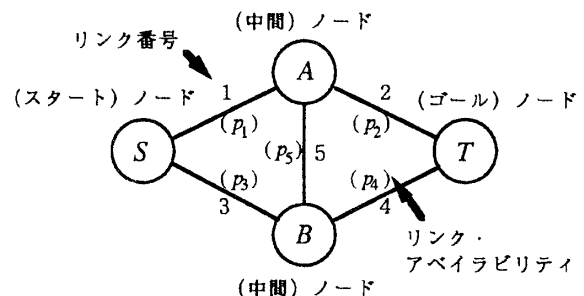


図1. 4ノード・ネットワーク

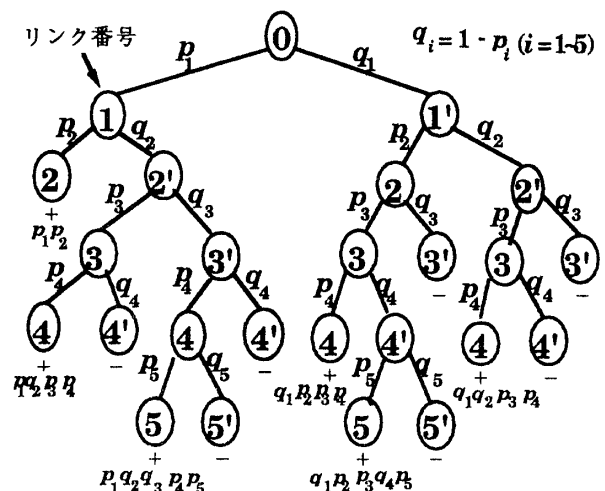


図2. 状態空間分解

Accelerated Computation of Network Availability.

Hisayuki Tatsumi*, Takahiro Furuya*,

Yoshimichi Kotaka**, Shinji Tokumasu*,

* Kanagawa Institute of Technology

** First Time

即ち、次式(4)のようになる。

$$A_V(p_1, p_2, p_3, p_4, p_5) \\ = p_1 p_2 + p_1 q_2 p_3 p_4 + p_1 q_2 q_3 p_4 p_5 \\ + q_1 p_2 p_3 p_4 + q_1 p_2 p_3 q_4 p_5 + q_1 q_2 p_3 p_4 \quad \dots(4)$$

ここで、 $q_i = 1 - p_i$ ($i=1, \dots, 5$)である。

3. 計算アルゴリズム

ノードに関する効果的な展開順序を、発信ノード S と着信ノード T の両側から調べて決定していく。 S, T に関して対称的なアルゴリズムを提案する。このアベイラビリティ計算アルゴリズムは、次の4つのステップから成り立っている。

(1) ステップ1

ネットワークをグラフとして入力する。例として図4のグラフ $G(N, L, P)$ のデータを、次のような方法でデータ化する。ここでノード集合を N 、リンク集合を L 、パスの信頼度集合を P とすると、

$$N = \{S, a, b, c, d, e, f, T\}$$

$$L = \left\{ 1(S, a), 2(a, b), 3(S, b), 4(S, c), 5(b, c), 6(b, d), 7(c, d), \right. \\ \left. 8(c, T), 9(d, T), 10(d, e), 11(e, T), 12(f, T), 13(e, f) \right\}$$

$$P = \{p_i | L \rightarrow [0, 1], 1 \leq i \leq 13\}$$

となる。

(2) ステップ2

ノードにラベリングを行う。ラベリングは前進ラベリングと後退ラベリングの2種類あり、前進ラベリングは、 S をスタートとし、 S からの距離が1となる T 以外のノードを探して図4のようにラベリングする。これを繰り返して、 T 以外のノードがすべてラベリングされたら T に無限大を代入して終了する。また後退ラベリングは前進ラベリングの S と T を逆にした方法で進める。

図4において、各ノードの上半円内に付けられたラベルが前進ラベル、下半円内に付けられたラベルが後退ラベルである。

(3) ステップ3

着信ノード T を除く各ノードに、展開順序を示す1次元のリンクテーブルの生成を行う。生成法は以下の通りである。

【ノード n のリンクテーブル生成ルール】

(a) 登録

$n_i \neq S$ ならば l_i を登録する。

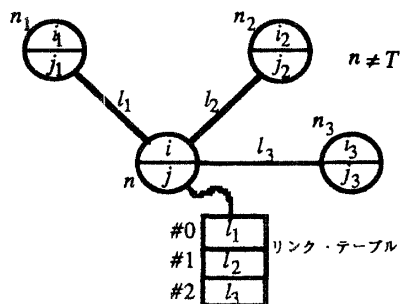


図3. リンクテーブルの生成法

(b) 順位 (上図3を参照)

l_1, l_2 が登録されるものとするとき、

$l_1 > l_2$ なら $l_1 < l_2$

$l_1 = l_2$ で $j_1 \leq j_2$ のとき、 $l_1 < l_2$

で、 l_1 が l_2 より高順位となる。

即ち、これはリンクの展開順序を規定するもので S, T 間のパスが貫通しやすく、またカットが発生しやすいリンクが先に展開されるものとなっている。このように、登録をしていくと図4のようになる。

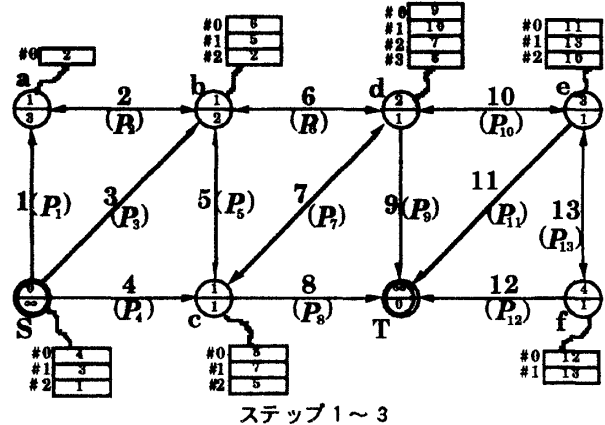


図4. 新アルゴリズム

(4) ステップ4

ステップ3のリンクテーブル順に展開を行う。アベイラビリティ計算木を生成していく際に、リンクテーブルを用いて各分岐点には展開候補テーブルが生成される。

4. 実行結果

本手法で図1のネットワークを評価したところ、表1に示すような理想に近い結果が得られた。また6ノードネットワークでは、番号順による方法では節点の数が125、+節点の数が26となり、本手法では節点の数が69、+節点の数は13となった。この結果より、本手法の方が約半分の値を指すことからプログラムの有効性が確認された。

表1. 4ノードネットワークの結果

	本手法	番号順による方法	理想的展開方法
接点の数	21	25	19
+接点の数	5	6	5
	1 2 (1 2)	1 2 (1 2)	1 2 (1 2)
	1 2 5 4 (1 5 4)	1 2 3 4	1 2 3 4
+系列	1 2 5 3 4	1 2 3 4	1 2 3 4 5 (1 4 5)
	1 3 4 (3 4)	1 2 3 4 (3 4)	1 3 4 (3 4)
0 内は独立パス	1 3 4 5 2 (3 5 2)	1 2 3 4 5 (1 4 5)	1 2 3 4 5 (2 3 5)
		1 2 3 4 5 (2 3 5)	

4. あとがき

ネットワークアベイラビリティの計算に、前進および後退の双方向ラベリングを用いたリンクテーブルの生成により、2分木展開の効率化・高速化が計られた。今後の課題として、多様なノードネットワークのシミュレーションによる評価と実用化展開が挙げられる。

参考文献

[1] Aaron Kershenbaum : Telecommunications Network Design Algorithms, McGraw-Hill, 1993.