

大阪府立工業高等専門学校 正 員 ○若林 拓史
豊橋技術科学大学工学部 学生員 松浦 和人
熊本大学工学部 学生員 山本健太郎

1. はじめに

本稿では、ネットワークの任意の2点間の*n*番目最短経路を求める近似解法を提案する。この*n*番目最短経路の直接の必要性は、道路網の信頼性解析で生じたものである。筆者らが開発した交点法¹⁾という道路網信頼性解析法では、分析対象ノード間のミニマルパス、ミニマルカットを生起確率の高い順に必要とする。このパス・カットの選択問題が、ネットワークのリンク長をリンク信頼度の対数という仮想的な長さで置換すると、当該ノード間の*n*番目最短経路選択問題となることが明らかとなっている¹⁾。また、交通量配分手法でも、転換率曲線を用いる場合には、複数の経路すなわち*n*番目最短経路を必要とする。したがって、*n*番目最短経路を効率よく探索する方法が開発されれば有用である。

提案する方法は、最短経路探索の代表的な解法であるDijkstra法を基本としている。Dijkstra法は計算量が少ないうえに、記憶容量も小さいという利点を有している。本研究ではこの利点を生かし、最短経路の探索後、その経路の構成リンクに微少な増分を与えて再び最短経路探索を行い、この繰り返しにより*n*番目最短経路探索を行なおうとするものである。微小増分の与え方に種々の方法があるので、これらと比較する。この方法は、近似解法であるので厳密性に欠ける欠点があるが、少ない計算時間と記憶容量で*n*番目最短経路を求めることができるという大きな利点を有している。

2. *n* 番目最短経路探索のアルゴリズム

本アルゴリズムは以下のとおりである。

- ① $m = 1$ として最短経路探索 (Dijkstra法) を行ない、第 m 番目最短経路とする。
- ② 第 m 番目最短経路の構成リンクのリンク長に後述する増分を与えて、再び最短経路探索を行なう。
- ③ 新しい経路が選ばれるまで②を繰り返す。最短経路が新しい経路に移ったら、これを第 $(m + 1)$

番目最短経路とする。

- ④ $m = m + 1$ として、②へ戻る。
- ⑤ 最終的に必要とする経路数より多い、適当な数の経路を取り出して終了する。それぞれの経路について、本来のリンク長 (増分を与えないリンク長) の値で経路長を再計算し、上位の必要数を用いる。

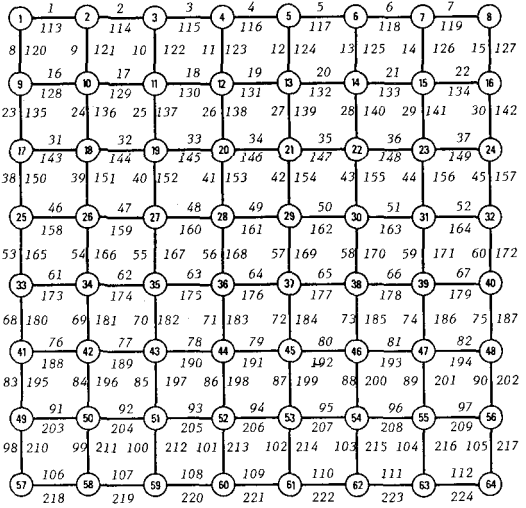


図-1 テストネットワーク

表-1 ノード5～60間における*n*番目最短経路の構成リンク

経路の順位	構 成 リ ン ク											距離
1	12	27	42	57	72	87	206	101				13.7
1	12	131	26	41	56	71	86	101				11.7
3	12	27	42	57	72	191	86	101				11.9
4	12	27	42	57	176	71	86	101				12.0
5	116	11	26	41	56	71	86	101				12.1
6	12	27	146	41	56	71	86	101				12.3
7	12	27	42	57	72	87	102	221				12.4
8	12	27	42	161	56	71	86	101				12.6
9	5	13	28	43	58	177	72	87	206	101		13.0
10	5	13	28	43	58	73	88	207	206	101		13.1
11	5	13	28	43	58	177	72	191	86	101		13.2
11	12	20	28	43	58	177	72	87	206	101		13.2
11	12	27	35	43	58	177	72	87	206	101		13.2
14	5	13	28	43	58	177	176	71	86	101		13.3
14	12	20	28	43	58	73	88	207	206	101		13.3
14	12	27	35	43	58	73	88	207	206	101		13.3
17	12	20	28	43	58	177	72	191	86	101		13.4
17	12	27	35	43	58	177	72	191	86	101		13.4
19	12	20	28	43	58	177	176	71	86	101		13.5
19	12	27	35	43	58	177	176	71	86	101		13.5
19	12	131	26	145	40	48	56	71	86	101		13.5

Hiroshi WAKABAYASHI, Kazuhito MATSUURA, Kentaro YAMAMOTO

3. リンク長増分の与え方

次の3タイプを考える。

- ・方法(1)：上の②で探索された経路の構成リンクすべてに増分を与える。
- ・方法(2)：(1)において起終点ノードに接続するリンクには増分を与えない。これは、これらのリンクは頻繁に利用されることを考慮したためである。
- ・方法(3)：(2)の操作にさらに次の操作を加える。(1)において、それまでに探索された m 個の経路で同一のリンクが何回利用されたかを調べておき、重複利用の度数に応じて増分を減ずる(よく利用されるリンクの増分は小さくする)。これは、リンク長が短いために頻繁に利用されることが予想されるリンクを考慮したためである。

各リンクの増分量は、そのリンクの初期長さの10%、5%、1%、0.5%、0.1%の5ケースを考え、それぞれの増分量を②で順次加算するものとする。

4. 数値計算例

図-1に示すネットワークのノード5～60間で数値計算を行なう。リンク長は[1,2]間の数値を一様乱数で与え、比較基準とする厳密な n 番目最短経路(表-1)は可能性のある経路を総当たりの求めて用いる。各方法、各増分量における経路の発見順(100位まで)を表-2に示す。方法1と2においては探索できない経路も存在するが、ほとんどの経路を探索できる。方法3ではこれがさらに改善され、発見順序も良好

であることがわかる。

参考文献：

- 1) 飯田・若林・吉木：ミニマルパス・カットを用いた道路網信頼度の近似計算法，交通工学，Vol. 23, No. 4, 1988.

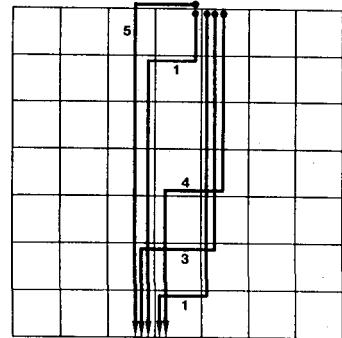


図-2 総当たり法での上位5経路，および近似解法で並べ換え後の上位5経路(両者は一致している)

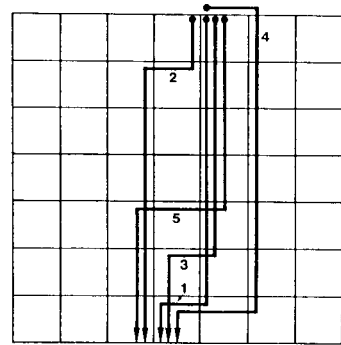


図-3 発見順にみた最初の5経路の一例(方法3)

表-2 近似解法による n 番目最短経路の発見順位(100位以内)

経路の順位	増分0.1%			増分0.5%			増分1.0%			増分5.0%			増分10.0%		
	方法1	方法2	方法3	方法1	方法2	方法3	方法1	方法2	方法3	方法1	方法2	方法3	方法1	方法2	方法3
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
3	5	3	3	5	3	3	5	3	3	7	3	3	9	3	3
4	20	8	5	17	8	5	15	8	5	14	10	5	14	36	5
5	3	5	6	3	5	6	3	5	6	3	6	7	3	7	7
6	36	31	11	35	28	11	32	24	11	22	19	11	20	16	11
7	6	11	13	6	11	13	8	11	14	5	13	13	7	11	12
8	*	*	23	*	*	24	*	*	24	*	*	21	*	*	20
9	14	21	46	16	19	50	13	30	47	43	16	57	53	72	75
10	4	4	4	4	4	4	4	4	4	4	4	4	8	4	4
11	13	18	47	26	20	46	17	83	34	12	*	36	49	*	25
11	19	19	42	22	16	53	45	16	38	96	12	46	44	17	18
11	*	73	45	*	*	39	*	50	48	*	*	44	*	31	36
14	16	15	44	14	15	40	21	17	44	*	42	29	*	23	30
14	18	6	7	15	6	7	16	6	7	13	7	8	37	21	8
14	*	61	17	*	56	17	*	53	17	*	27	18	*	*	29
17	35	22	41	19	17	37	73	20	42	66	58	42	12	*	48
17	*	63	37	*	60	45	*	80	45	*	*	23	89	*	53
19	21	20	40	33	26	42	28	15	36	40	25	37	*	10	45
19	*	68	39	*	*	38	*	*	37	*	*	34	*	*	33
19	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
CPU-TIME	40716	34185	197895	10819	10142	40946	6806	6308	21834	3852	3490	6130	3288	3148	4498

CPU-TIME：単位ms。*：100位以内で発見できなかった経路