社団法人 情報処理学会　研究報告
IPSJ SIG Technical Report

2003－ＡＬ－92　　(3)
2003／11／7

局所情報を用いたランダムウォークの拡張

Jesper Jansson[2]　定兼 邦彦[1]　Wing-Kin Sung[2]　塩崎 真史[3]　山下 雅史[1]

[1] 九州大学大学院システム情報科学研究院
〒812-8581 福岡市東区箱崎 6-10-1　{sada,mak}@csce.kyushu-u.ac.jp
[2] National University of Singapore　{jansson,ksung}@comp.nus.edu.sg
[3] 九州大学工学部電気情報工学科　masashio@tcslab.csce.kyushu-u.ac.jp

グラフのランダムウォークとは１つのトークンが訪れるグラフの頂点の列であり，全点を訪れるために必要な時間 (被覆時間) の解析は様々な応用がある．点数を n とした場合，単純なランダムウォークでは被覆時間が $\Theta(n^3)$ であることがわかっている．これをグラフの頂点の局所的な情報を用いて $O(n^2 \log n)$ にする方法も提案されている．本論文ではこれとは異なる種類の局所情報を用いるランダムウォークを提案する．また，決定的ウォークについても考察する．

An Extended Scheme for Random Walks using Local Information

Jesper Jansson[2]　Kunihiko Sadakane[1]　Wing-Kin Sung[2]
Masashi Shiozaki[3]　Masafumi Yamashita[1] [1]
Department of Computer Science and Communication Engineering,
Kyushu University, Hakozaki 6-10-1, Higashi-ku, Fukuoka 812-8581, Japan
[2] National University of Singapore　{jansson,ksung}@comp.nus.edu.sg
[3] Department of Electrical Engineering and Computer Science,
Kyushu University　masashio@tcslab.csce.kyushu-u.ac.jp

We consider an extension of random walks on a graph. It is well known that the cover time is $\Theta(n^3)$ for graphs with n nodes by the classical random walk. Recently a scheme for random walks using local topological information of a graph has been proposed to reduce the cover time to $O(n^2 \log n)$. In this paper we propose another scheme called *directional random walks* which uses the following local information: (1) the last visited node, (2) the order of the neighbor nodes of each node. We also consider deterministic walks.

1　はじめに

本論文ではグラフのランダムウォークの拡張について論じる．ランダムウォークとはあるトークンが訪れるグラフの頂点の列であり，訪れる頂点はある確率分布に従ってランダムに選ばれる．ランダムウォークの被覆時間 (cover time) とは次のように定義される．

定義 1 $C(G,v)$ を連結グラフ $G=(V,E)$ の頂点 v から始まったランダムウォークがグラフの全ての頂点を訪れるまでの時間の期待値とする．このとき，被覆時間は $\max_{v\in V} C(G,v)$ と定義される．

また，2 頂点 v, w 間の命中時間 (hitting time) $H^G(v,w)$ とは，v から始まったランダムウォークが w に到達するまでの時間の期待値とする．被覆時間の計算には様々な応用が存在し，重要な問題である [4]．Israeli-Jalfon が提案したランダムウォークは，隣接する全ての頂点に等しい確率で移動するものである．つまり，頂点 v から w に移る確率は $\Pr(w|v)=1/\deg(v)$ となる．このとき被覆時間は頂点数 n の任意のグラフについて $O(n^3)$ であることがわかっている [3]．また，lollipop グラフと呼ばれるグラフについては被覆時間が $\Omega(n^3)$ であることもわかっている．

Ikeda ら [1] が提案した手法では，遷移確率を $\Pr(w|v)\propto 1/\max\{\deg(v),\deg(w)\}$ とすることで，命中時間を $O(n^3)$ から $O(n^2)$ に，被覆時間を $O(n^3)$ から $O(n^2\log n)$ に改善している．また，$\Pr(w|v)\propto 1/\sqrt{\deg(w)}$ とする手法でも同じ命中時間，被覆時間を達成している [2]．これらの手法では隣接点の次数というグラフの局所的な情報のみを利用している．

本論文ではこれらとは異なる種類の局所情報を用いたランダムウォークを提案する．また，ランダムではない決定的なウォークについても考察する．

2　局所情報を用いたランダムウォーク

本論文で提案するランダムウォークで用いる局所情報は以下のものである．トークンがグラフの頂点 v にあるとき，次に移る頂点を決める際に

- トークンが v に来る直前に訪れた頂点
- v の全ての隣接点につけられた順序

を利用する．このような情報を利用可能なグラフとして，順序つきグラフを定義する．

定義 2 順序つきグラフとは，グラフの各ノードに対しその隣接ノードに順序がついているものである．次数 d のノード v の隣接ノードを $v_1, v_2, \ldots, v_d$ で表す．便宜的に $v_{d+i}=v_i$ とする．

本論文では以下のようなランダムウォークを考える:

定義 3 (指向性ランダムウォーク) 順序つきグラフ上での指向性ランダムウォークとは，以下のように定義される遷移確率に従うランダムウォークである．

$$\Pr(v_i|v)=\left(\frac{1}{3}\right)^{i-j+1}$$

ここで v は現在の頂点，j は v の直前に訪れた頂点の v から見た順序である．

直感的には，指向性ランダムウォークとは隣の枝へ高い確率で移るようなランダムウォークである．こうすることで小さなサイクルを防ぎ，被覆時間が小さくなることが期待できる．

2.1 応用

指向性ランダムウォークの応用としては，NP 完全問題を解くための局所改良法への適用が考えられる．局所改良法では実行可能解を局所的に変化させ，目的関数の値を改善する．この際に，目的関数の値が最も改善されるように解を変化させると，局所最適解しか求まらない．これを解決するために焼きなまし法，遺伝的アルゴリズム，タブー探索などの発見的手法が用いられている．これらは解を目的関数値が悪くなる方向に一時的に変化させることで局所解から抜け出している．

指向性ランダムウォークを用いて，次のような局所改良法を考えることができる．解空間を順序つきグラフで表す．ノードの順序は，そのノードでの目的関数値が小さい順として定義する．このグラフでの指向性ランダムウォークを行うと，目的関数値が小さくなるノードに高い確率で移っていき，かつランダム性を入れることで局所解から抜け出せる．また，このランダムウォークでは小さなサイクルを防ぐことができるため，限られた時間内に多くのノードを訪れることができ，良い大域最適解を求めることができると思われる．

2.2 実験結果

代表的なグラフに対し，命中時間と被覆時間をシミュレーションで求めた．用いたグラフは完全グラフ K_n，パスグラフ P_n，スターグラフ S_n (中心点から端点までの距離は 2)，lollipop グラフ L_n ($K_{n/2}$ と $P_{n/2}$ を繋いだもの) である．命中時間とと被覆時間は 10000 回の実行の平均である．命中時間は最も大きくなる 2 点間の値である．表 1 からわかるように，被覆時間は $O(n^3)$ より小さいことがわかるが，更なる解析が必要である．

表 1: 命中時間と被覆時間

	完全グラフ		パスグラフ		lollipop グラフ		スターグラフ	
n	命中	被覆	命中	被覆	命中	被覆	命中	被覆
10	10.65	23.09	120.21	169.13	94.99	104.39	64.44	119.35
12	12.85	29.38	182.26	236.20	142.10	154.04	78.06	158.07
14	15.31	36.04	226.81	319.12	203.85	218.72	92.24	200.12
16	17.64	42.98	353.57	419.86	280.32	293.92	106.49	245.24
18	19.96	50.25	418.87	528.73	371.15	389.11	119.96	290.29
20	22.36	57.59	507.70	666.57	471.25	495.83	133.16	338.07
22	24.62	65.31	586.77	846.92	595.42	619.55	146.38	388.14
24	26.56	73.05	748.80	957.77	724.75	761.27	160.19	437.15
26	29.45	80.97	871.57	1176.12	878.53	917.99	174.50	488.83
28	31.32	89.15	1086.05	1378.15	1051.58	1082.91	189.05	541.66
30	34.07	97.20	1173.76	1541.39	1222.84	1268.06	204.48	594.37

3 決定的ウォーク

本節では指向性ランダムウォークの極端な場合，つまり隣の枝に確率 1 で移るような決定的なウォークについて考察する．

定義 4 順序つきグラフ上での決定的ウォークとは，頂点 v を訪れる際に v_i から来た場合は次に必ず v_{i+1} に行くようなウォークである．

順序つきグラフの上で決定的ウォークを行う場合，頂点の順序の付け方によっては全ての点を被覆できない場合がある (図 1 参照)．よって，このような順序つきグラフでのウォークにはランダム性を入れる必要がある．

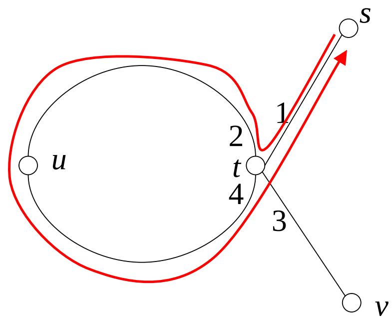

図 1: 決定的ウォークでは被覆できないグラフ

しかし，頂点の順序をうまく定めれば決定的ウォークによって全ての頂点を被覆することができる．以下ではこの定理を証明する．

定義 5 順序つきグラフ上の閉路とは，ある頂点から始まる決定的ウォークである．

全ての決定的ウォークは行き止まりがないため，必ず閉路を構成することに注意されたい．

命題 1 *順序つきグラフの各枝は必ず 1 つか 2 つの閉路に含まれる．*

証明: 1 つの枝は 2 つの方向で使われる．ウォークは決定的であるため，1 つの枝は高々2 つの閉路に含まれる．また，ある枝を使う決定的ウォークは必ず閉路となるため，各枝は少なくとも 1 つの閉路に含まれる．ある枝は 2 つの方向でそれぞれ閉路に使われるが，それらの閉路が同一のものの場合はその枝は 1 つの閉路のみに含まれている．

定理 1 任意のグラフに対し，決定的ウォークが全ての頂点を訪れるように隣接頂点に順序をつけた順序つきグラフが存在する．

証明: 枝数についての帰納法で証明する．グラフが 1 つの枝からなる場合，それは頂点 1 つで自己ループのあるグラフか，頂点 2 つでその間に枝があるグラフとなる．この場合は明らかに成り立つ．命題が枝数 m $(m \geq 1)$ の全てのグラフに対して成り立つと仮定し，新しく枝を 1 つ付け加えた時に命題が成り立つように頂点の順序付けを変更する．

Case 1: 1 つの頂点と 1 つの枝を追加する場合　図 2 参照．

グラフ G に新しい頂点 w と枝 (w, v) を付け加えて $\tilde{G}$ を作る．仮定より G の全点を通る決定的ウォークが存在し，v の前後の頂点の順序は連続したものになっている．v の全ての隣接点の順序に一定の値を加え，ウォークによって v に入ってくる点の順序が d $(\deg(v))$ に，v から出る点の順序が 1 になるようにする．そして新しく加えた枝の v での順序を $\deg(v) + 1$ にする．すると $\tilde{G}$ での新しい決定的ウォークは v の次に w を通るようになり，その他のウォークには変化がないため，全点を被覆するウォークとなる．

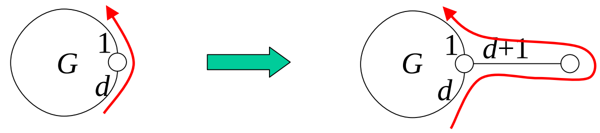

図 2: Case (1)

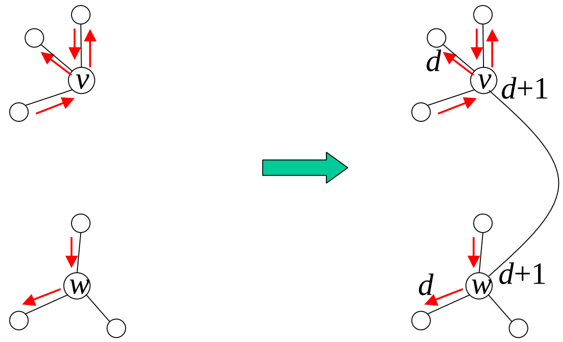

図 3: Case (2a)

Case 2: 2 点間に新しく枝を追加する場合　頂点 v と w の間に新しく枝を加えるとする．枝を加える前の決定的ウォークを C で表す．ある頂点に接続している全ての枝が両方向とも C に含まれているとき，その頂点を満員と呼ぶ．v，w が満員かどうかで場合わけをする．

Case 2(a): v，w が共に満員ではないとき　図 3 参照．両頂点が共に満員ではないので，1 方向のみ使われるか全く使われていない枝が存在する．新しい枝を加える前に，v から出るウォークの順序が $\deg(v)$ になるように v の枝の順序を変える．また，w についても同様に変える．そして新しい枝 (v,w) を加え，その順序を $\deg(v)+1$ と $\deg(w)+1$ にする．すると新しいグラフでの決定的ウォークは元のグラフと同じものになる．

Case 2(b): v，w が共に満員のとき　もし決定的ウォーク上での v から w への最短パス上の全ての頂点が満員の場合 (図 4 参照)，ウォークが枝 (v,w) を使うように枝の順序を変更する．v と w の間の頂点は w から v へ行くによって被覆されるため，全点が被覆される．

決定的ウォーク上での v から w への最短パス上に満員でない頂点があるとする (図 5 参照)．そのような頂点のうち v に最も近いものを u とする．頂点 v から u，u から w，w から v へのパスをそれぞれ p_1，p_2，p_3 と表す．新しい枝 (v,w) を加える前は，u は u_i から訪れたとする．すると u は頂点 u_{i-1} からも訪れていることになる (なぜなら u は満員ではない初めての頂点なので直前の頂点は満員であるから)．u は満員ではないため使われていない有向枝が存在し，その枝を含む閉路 C' が存在する．この閉路は u_{i+1}，u，u_{i+2} の順に頂点を通過する．

新しい枝 (v,w) を付け加えた後，u の順序の i と $i+1$ を交換する．すると新しい決定的ウォークは以下のように変化する．まず v から u に p_1 を使って到達し，次に u_{i+2} に行き，閉路 C' を使って u_i に行き，u に戻る．次に v に戻り，新しい枝を使って w へ行く．w からは p_3 を使って u に行き，次に p_2 を使う．その後 w に戻り，新しい枝を使い v に戻る．新しいウォークは古いウォークの全ての枝を使うため，全頂点を被覆する．

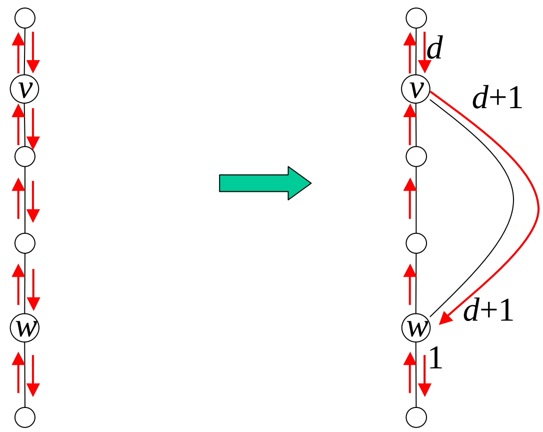

図 4: Case (2b1)

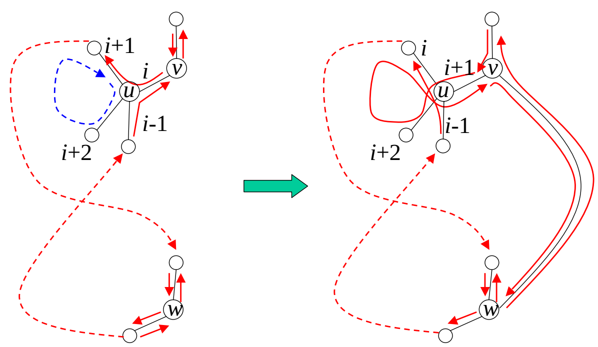

図 5: Case (2b2)

Case 2(c): v, w のうち片方のみが満員の場合　図 6 参照．v が満員でなく，w が満員と仮定してよい．この場合も Case 2(b) 同様に，v を用いる別の閉路 C' が存在するため，それを用いて決定的ウォークを変形すればよい．

4　まとめ

既存手法とは異なる種類の局所情報を用いたランダムウォークを提案した．今後は被覆時間の解析や，NP 完全問題に対する解法への応用を行う．

謝辞

この研究の一部は文部科学省科学研究費の援助を受けた．

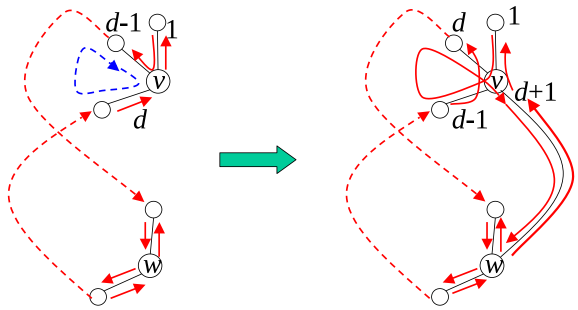

図 6: Case (2b2)

参考文献

[1] S. Ikeda, I. Kubo, N. Okumoto, and M. Yamashita. Impact of Local Topological Information on Random Walks on Finite Graphs, *Proceedings of 30th International Colloquium on Automata, Languages and Programming* (ICALP2003), LNCS 2719, pp. 1054–1067, 2003.

[2] S. Ikeda, I. Kubo, N. Okumoto, and M. Yamashita. Reducint the hitting time and the cover times of random walks on finite graphs by local topological information, *Proceedings of the 2003 International Conference on VLSI*, pp. 203–207, 2003.

[3] D. J. Aldous. On the time taken by random walks on finite groups to visit every state, *Z. Wahrsch. verw. Gebiete*, 62:361–393, 1983.

[4] G. Blom, L. Holst, and D. Sandell. Problems and Snapshots from the World of Probability, Springer-Verlag, New York, NY, 1994.