数理解析研究所講究録
第 1939 巻 2015 年 18-25

離散時間におけるツリーグラフ上での二人ランデヴーサーチゲーム

大阪府立大学大学院 理学系研究科 情報数理科学専攻　関川　圭太 (Keita Sekigawa)
北條　仁志 (Hitoshi Hohjo)
Dept. of Mathematics and Infomation Sciences, Graduate School of Science
Osaka Prefecture University

1　はじめに

現在ランデヴーサーチゲームは、コンピューターウィルスを拡散しているインターネット上での特定のコンピューターの検出やゲーム機器などのすれ違い通信など様々な分野で利用されている。また、相手との待ち合わせのような日常的なことにもランデヴーは発生している。

本稿では離散時間におけるツリーグラフ上での二人ランデヴーサーチゲームを扱う。ランデヴーサーチゲームはグラフの形状やプレーヤーの行動パターンによりいくつかの問題に分類できる。グラフでは線形グラフ、ツリーグラフ、サークルグラフ、ネットワークなどの種類がある。プレーヤーのタイプには協力型と非協力型がある。協力型では互いのプレーヤーが協力してランデヴーを達成しようとする。非協力型ではハイダーとシーカーと呼ばれるプレーヤーが存在し、ハイダーは相手に見つけられたくないと考え、グラフ上のあるノードに隠れるが、シーカーはハイダーを見つけるためグラフ上を探索する。時間には連続型と離散型があり、連続型ではノード上やノードを接続するエッジ上でランデヴーが達成されるが、離散型ではノード上でのみランデヴーが達成される。戦略にはプレーヤー同士が同じ戦略を実行する対称戦略や異なる戦略を実行する非対称戦略がある。

これまでに線形グラフでは Alpern[2]、Alpern et.al.[9]、Ozsoyeller et.al.[20] など数多くの研究がされてきた。Alpern[2] はノード数 n で構成される線形グラフにおいて各ノードに 1 人のプレーヤーが存在する仮定の下でランデヴーが達成される戦略を提案し、その期待ミーティングタイムが $\frac{n}{2}$ に近づくことを証明した。Alpern[9] は連続した 3 つのノード上に置かれた 3 人のプレーヤーによる線上ランデヴーサーチゲームについて考え、出遭ったプレーヤーは情報を共有でき、一緒に行動できるスティッキー戦略を導入した。Ozsoyeller[20] は線形グラフに置かれた 2 体のロボットが、拡張半径という新たな概念を用いてランデヴーを達成させる戦略を提案した。ツリー、サークル、ネットワークのグラフでは Alpern[3]、Dessmark et.al.[12] などにより研究がされてきた。Alpern[3] はネットワーク上の 2 人プレーヤーによるランデヴーサーチゲームにおいて新たにラベルという概念を導入した。Dessmark[12] はツリー、サークル、ネットワークでの 2 人プレーヤーによるランデヴーサーチゲームにおいてグラフの探索アルゴリズム、ラベルの拡張、グラフの縮約方法などを提案した。しかしツリー、サークル、ネットワークでの研究は線形グラフでの研究に比べると論文数が圧倒的に少ない。なぜなら、ツリー、サークル、ネットワークはグラフが大きく複雑化するので、戦略によってはランデヴーが達成されないことがあり、アルゴリズムの考案および数理的定式化が困難であるためである。本研究では、Dessmark[12] のモデルにおいて対象とするグラフをツリーグラフに限定することで、より早くランデヴーが達成されるアルゴリズムを提案し、数値例により期待ミーティングタイムを Dessmark[12] の結果と比較する。

2　本研究のモデル

本稿では離散時間におけるランデヴーについて考える。2人のプレーヤーはノード数がnであるツリーグラフ$T(V,E)$上にランダムに置かれる。各ノード間の距離は1とし、グラフは連結で閉路がなければ各ノードに接続するエッジが何本あってもよいとする。プレーヤーは、自分と相手の初期点の位置や向き、ツリーグラフの形状に関する情報を全く持っていない。プレーヤーは初期点に接続するエッジを等確率で選び、サーチを開始する。このとき両プレーヤーがサーチを開始する時間は同時であるとし、移動速度は単位時間あたり1である。サーチはノード上でステイすることも可能である。プレーヤーは異なるラベルをもち、ツリー探索やラベルの拡張において対称戦略を用いる。ラベルは1で始まる0,1のビット列であり、0はステイ、1は移動を表す。ラベルはツリーグラフTがノード数2の完全グラフに縮約できた時に用いる。縮約の方法は4節で示す。このようなモデルにおいてプレーヤーは期待ミーティングタイムを最小にしたい。

3　表記法

本稿では以下の記号を用いて説明を行う。

n	:	ツリーのノード数
t	:	サーチを開始してからの時間
$k(t)$	:	時間tにおける相対最長パスの長さ
$s(t)$	:	初期点から現状点までの距離
L_i	:	プレーヤーiのラベル
l_i	:	ラベルL_iの長さ
l	:	ラベル長l_iの最小値
$m_1(t)$	:	プレーヤーの現状点から相対最長パスへの距離
$m_2(t)$	:	初期点から相対最長パスへの距離
$k_1(t),k_2(t)$	:	現状点と相対最長パスを結ぶパスと相対最長パスの交点を境に相対最長パスを分けたときのそれぞれのパスの長さ
$k_3(t),k_4(t)$	:	初期点と相対最長パスを結ぶパスと相対最長パスの交点を境に相対最長パスを分けたときのそれぞれのパスの長さ

ここで、最長パスとはツリーグラフTで最もノード間の距離が長いパスを意味しており、相対最長パスとは、時間tまでにサーチした部分グラフの最長パスを意味している。

これらの記号の関係を図示すると図1のようになる。現状点が相対最長パス上にあるとき、$m_1=0$と定義し、現状点と相対最長パスを結ぶパスと相対最長パスの交点が現状点であると考える。初期点が相対最長パス上にあるとき、$m_2=0$と定義し、初期点と相対最長パスを結ぶパスと相対最長パスの交点が初期点であると考える。

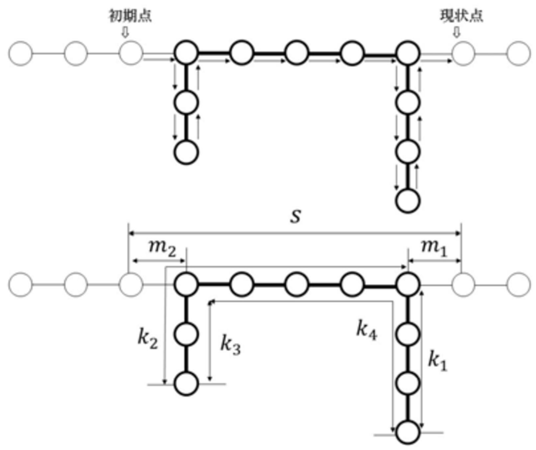

図 1: 右回りの戦略及び記号の関係性

4 アルゴリズム

4.1 ツリー探索のアルゴリズム

Dessmark et.al.[12] では、プレーヤーは右回りか左回りの戦略ですべてのノードをサーチし、初期点に到着した後、ツリーグラフのセントラルノードもしくはセントラルエッジへ向かうようなツリー探索アルゴリズムが用いられた。右回りの戦略とは、隣接するノードに到着した時に最も右のエッジを選びサーチを続ける戦略のことである。同様に左回りの戦略とは、隣接するノードに到着した時に最も左のエッジを選びサーチを続ける戦略のことである。また、セントラルノードは最長パスの長さが偶数であるときに最長パスの真ん中にある一意なノードであり、セントラルエッジは最長パスの長さが奇数であるときに最長パスの中点を含むエッジである。

このアルゴリズムにおいて、すべてのノードをサーチし初期点に到着するまでの残り時間とプレーヤーの初期点および現状点の位置を考慮し、相対最長パスがツリーグラフの最長パスと同一であると確認できたなら、サーチをやめ、相対最長パスのセントラルノードもしくはセントラルエッジへ向かうように改良する。ツリーのノード数が n であるので、すべてのノードをサーチして初期点に到着するまでにかかる時間は $2n-2$ である。よって、時間 t での現状点から残りのすべてのノードをサーチして初期点に到着するまでの残り時間は $2n-2-t$ と表せる。また、時間 t においてまだサーチしていないノード数は $\frac{2n-2-t-s}{2}$ となる。

ある時刻においてプレーヤーの初期点と現状点が以下の条件を満たすなら、セントラルノードもしくはセントラルエッジへ向かうことでミーティングタイムを短縮することができる。

(i) 現状点が相対最長パス上にあり、初期点が相対最長パスの端点であるとき

このとき残り時間が初期点と現状点との距離と同じであれば、すでにサーチしたノードをたどり初期点へ戻るだけであるので、相対最長パスはツリー T の最長パスと一致する。よって

$$2n-2-t=s$$

を満たした段階で、プレーヤーはセントラルノードもしくはセントラルエッジへ向かう。

(ii) 現状点が相対最長パス上にあり、初期点が相対最長パス上にあるがその端点でないとき

このときまだサーチされていないノード数が k_i (i=1,2,3,4) より小さければ、相対最長パスとツリーの最長パスが同じと考えられるので、

$$\frac{2n-2-t-s}{2} \leq k_i, \qquad i = 1,2,3,4$$

を満たすならば、セントラルノードもしくはセントラルエッジへ向かう。

(iii) 現状点が相対最長パス上にあり、初期点が相対最長パス上にないとき

このとき現状点に対しては (ii) と同様の条件を満たせばよい。初期点に対しては相対最長パスと m_2 のパスの交点においてサーチされていないノード数と m_2 の和が k_j (j=3,4) より小さければ、相対最長パスとツリーの最長パスが同じと考えられる。よって、

$$\frac{2n-2-t-s}{2} \leq k_i, \quad i = 1,2 \quad \text{かつ} \quad m_2 + \frac{2n-2-t-s}{2} \leq k_j, \quad j = 3,4$$

を満たすならば、セントラルノードもしくはセントラルエッジへ向かう。

(iv) 現状点が相対最長パス上になく、初期点が相対最長パスの端点であるとき

(i) と同様に

$$2n-2-t = s$$

を満たすならば、セントラルノードもしくはセントラルエッジへ向かう。

(v) 現状点が相対最長パス上になく、初期点が相対最長パス上にあるがその端点でないとき

このとき初期点に対しては (ii) と同様の条件を満たせばよい。現状点に対しては (iii) の初期点の場合と同様に考えると、

$$m_1 + \frac{2n-2-t-s}{2} \leq k_i, \quad i = 1,2 \quad \text{かつ} \quad \frac{2n-2-t-s}{2} \leq k_j, \quad j = 3,4$$

を満たすならば、セントラルノードもしくはセントラルエッジへ向かう。

(vi) 現状点が相対最長パス上になく、初期点も相対最長パス上にないとき

現状点に対しては (v)、初期点に対しては (iii) と同様なので、

$$m_1 + \frac{2n-2-t-s}{2} \leq k_i, \quad i = 1,2 \quad \text{かつ} \quad m_2 + \frac{2n-2-t-s}{2} \leq k_j, \quad j = 3,4$$

を満たすならば、セントラルノードもしくはセントラルエッジへ向かう。

プレーヤーは以上の条件式が成り立つとき、ツリーグラフ T の最長パスがセントラルノードかセントラルエッジのどちらを持つのかを把握することができる。セントラルノードを持つ場合は、セントラルノードに到着するとずっとステイし、ランデヴーを達成させる。セントラルエッジを持つ場合は、セントラルエッジとそれに接続する 2 点のノードで構成される縮約グラフ K_2 に到着すると次節のラベル拡張の手順を実行し、ランデヴーを達成させる。

4.2 ラベル拡張の手順

ラベルは1から始まる0と1で構成されるビット列であり、0はステイ、1は移動を表す。今、両プレーヤーには異なるラベルが与えられるとする。

まず、Dessmark et.al.[12] で提案されたラベル拡張の手順を紹介する。

ラベル拡張アルゴリズム [12]
Step 1. ラベル L_i の各ビットを2つ続けて並べ L_i' を作る。
Step 2. ラベル L_i' の前に10を付け L_i" を作る。
Step 3. ラベル L_i" を繰り返し無限バイナリ文字列 L_i^* を作る。

このアルゴリズムにより生成されるラベル拡張の例は以下のとおりである。

例 1. L_1=1101 → L_1'=11110011 → L_1" = 1011110011
→ L_1^* = 10111100111011110011⋯

このラベル拡張の手順を用いた結果として、二人目のプレーヤーが縮約グラフ K_2 に到着してから $2l+4$ ステップより早くランデヴーが達成されることが分かっている。つまり、プレーヤーは $2l+4$ ステップ後にランデヴーが達成されていないならば、自分は先に縮約グラフ K_2 に到着したプレーヤーであると考えられる。また、プレーヤーに与えられるラベルは1のビットで始まっているので、$2l+4$ ステップ後はステイし続ければ二人目のプレーヤーが縮約グラフ K_2 に到着してから遅くて1ステップでランデヴーが達成される。以上のことを踏まえて Dessmark et.al.[12] のラベル拡張アルゴリズムを改良する。

改良ラベル拡張アルゴリズム
Step 1. ラベル L_i の各ビットを2つ続けて並べ L_i' を作る。
Step 2. ラベル L_i' の前後に10を付け L_i" を作る。
Step 3. ラベル L_i" 以降は0を付けラベル L_i^* を作る。

改良アルゴリズムにより生成されるラベル拡張の例は以下のとおりである。

例 2. L_1=1101 → L_1'=11110011 → L_1" = 101111001110
→ L_1^* = 10111100111000000000⋯

4.3 アルゴリズム

4.1節および4.2節で提案したツリー探索と改良ラベル拡張アルゴリズムを用いたツリーグラフにおける2人ランデヴーサーチアルゴリズムを以下に与える。

ランデヴーサーチアルゴリズム

Step 1 プレーヤーは右回りの戦略か左回りの戦略か決め、ツリーを探索する。

Step 2 以下を実行する。

(i) 現状点が相対最長パス上にあり、初期点が相対最長パスの端点である状況において $2n-2-t=s$ を満たすならば、Step 4へ。

(ii) 現状点が相対最長パス上にあり、初期点が相対最長パス上にあるがその端点でない状況において $\frac{2n-2-t-s}{2} \leq k_i$ (i=1,2,3,4) を満たすならば、Step 4 へ。

(iii) 現状点が相対最長パス上にあり、初期点が相対最長パス上にない状況において $\frac{2n-2-t-s}{2} \leq k_i$ (i=1,2) かつ $m_2 + \frac{2n-2-t-s}{2} \leq k_j$ (j=3,4) を満たすならば、Step 4 へ。

(iv) 現状点が相対最長パス上になく、初期点が相対最長パスの端点である状況において $2n-2-t=s$ を満たすならば、Step 4 へ。

(v) 現状点が相対最長パス上になく、初期点が相対最長パス上にあるがその端点でない状況において $m_1 + \frac{2n-2-t-s}{2} \leq k_i$ (i=1,2) かつ $\frac{2n-2-t-s}{2} \leq k_j$ (j=3,4) を満たすならば、Step 4 へ。

(vi) 現状点が相対最長パス上になく、初期点も相対最長パス上にない状況において $m_1 + \frac{2n-2-t-s}{2} \leq k_i$ (i=1,2) かつ $m_2 + \frac{2n-2-t-s}{2} \leq k_j$ (j=3,4) を満たすならば、Step 4 へ。

Step 3 次のノードに移動し、Step 2 へ。

Step 4 セントラルノードもしくはセントラルエッジへ向かう。

Step 5 **(I)** ツリーグラフ T がセントラルノードをもつとき
セントラルノードに到着するとステイし続ける。

(II) ツリーグラフ T がセントラルエッジをもつとき
セントラルエッジとそれに接続する 2 点のノードで構成される縮約グラフ K_2 に到着すると、ラベルを (a)～(c) のように拡張し、ランデヴーが達成されるまでラベルに従って実行する。

(a) ラベル L_i の各ビットを 2 つ続けて並べ L'_i を作る。

(b) ラベル L'_i の前後に 10 を付け L_i" を作る。

(c) ラベル L_i" 以降は 0 を付けラベル L^*_i を作る。

5 数値例

本稿ではノード数が 5,6 であるツリーグラフのすべての形状において、初期点の位置や出発の向き、戦略、ラベルの長さなどを考慮して可能性のあるすべてのツリーグラフにおけるミーティングタイムを計算し、期待ミーティングタイムを求めた。その期待ミーティングタイムにおいて Dessmark et.al.[12] と比較すると表 1 のようになる。

すべての場合において改良したアルゴリズムの方が、実数値や変数の係数が小さくなっていることが数値的に示せた。ミーティングタイムはツリーグラフの探索を開始してからセントラルノー

	Dessmark.et.al(2008)[12]	改良
$n=5, l_1=l_2$	6.86+0.23l_1	5.60+0.11l_1
$n=5, l_1 \neq l_2$	7.20+0.23l	5.76+0.11l
$n=6, l_1=l_2$	7.96+0.23l_1	6.70+0.12l_1
$n=6, l_1 \neq l_2$	8.36+0.23l	6.94+0.12l

表 1: 期待ミーティングタイムによる比較

ドもしくは縮約グラフ K_2 に到着するまでの時間とそこに到着してから出遭うまでの時間の合計である。シミュレーションの結果、前者は実数値で求められ、後者は実数値と変数に依存した値の和で求められ、後者の実数値は前者の実数値に比べて非常に小さいことも分かっている。ゆえにミーティングタイムの実数値部分の減少はツリー探索アルゴリズムの改良による影響であり、変数に依存した値の減少はラベル拡張の手順の改良による影響であると考えられる。

6 まとめ

本稿では、ツリーグラフ上での2人ランデヴーサーチゲームを扱い、Dessmark et.al.[12] のアルゴリズムをツリーグラフに特化してツリー探索およびラベル拡張を改良することにより新しいサーチアルゴリズムを提案した。数値例では、ノード数が小さいツリーグラフに関して Dessmark et.al.[12] のアルゴリズムより期待ミーティングタイムが小さくなったことを確認した。また、ノード数が大きいツリーグラフに関しては期待ミーティングタイムが大幅に小さくなると推測される。今後の課題としてはサークルやネットワークへの拡張やプログラム実装などが挙げられる。

参考文献

[1] Alpern,S., The Rendezvous Search Problem, SIAM Journal on Control and Optimization **33** (3), 673-683 (1995).

[2] Alpern,S., Minimax Rendezvous on the Line, SIAM Journal on Control and Optimization **34** (5), 1650-1665 (1996).

[3] Alpern,S., Rendezvous Search on Labeled Networks, Naval Research Logistics **49**, 256-274 (2002).

[4] Alpern,S., Serach Games on Trees with Asymmetric Travel Times, SIAM Journal on Control Optimization **48** (8), 5547-5563 (2010).

[5] Alpern,S., A.Beck, Asymmetric Rendezvous on the Line is a Double Linear Search Problem, Mathematics of Operations Research **24** (3), 604-618 (1999).

[6] Alpern,S., A.Beck, Rendezvous Search on the Line with Limited Resources: Maximizing the Probability of Meeting, Operations Research **47** (6), 849-861 (1999).

[7] Alpern,S., A.Beck, Pure Strategy Asymmetric Rendezvous on the Line with an Unknown Initial Distance, Operations Research **48** (3), 498-501 (2000).

[8] Alpern,S., S.Gal, Searching for an Agent Who May or May Not Want to Be Found, Operations Research **50** (2), 311-323 (2002).

[9] Alpern,S., W.S.Lim, Rendezvous of Three Agents on the Line, Naval Research Logistics **49**, 244-255 (2002).

[10] Anderson,E.J., S.P.Fekete, Two Dimensional Rendezvous Search, Operations Research **49** (1), 107-118 (2001).

[11] Chester,E.J., R.H.Tutuncu, Rendezvous Search on the Labeled Line, Operations Research **52** (2), 330-334 (2004).

[12] Dessmark,A., P.Fraigniaud, D.R.Kowalski, A.Pelc, Deterministic Rendezvous in Graphs, Algorithmica **46**, 69-96 (2006).

[13] Gal,S., J.V.Howard, Rendezvous-Evasion Search in Two Boxes, Operations Research **53** (4), 689-697 (2005).

[14] Garnaev,A., Search Games and Other Applications of Game Theory, Berlin, Springer Verlag (1999).

[15] Kikuta,K., A Hide and Seek Game with Traveling Cost, Journal of the Operations Research Society of Japan **33** (2), 168-187 (1990).

[16] Kikuta,K., A Search Game with Traveling Cost on a Tree, Journal of the Operations Research Society of Japan **38** (1), 70-88 (1995).

[17] Kikuta,K., A Rendezvous Search on a Linear Graph, 京都大学数理解析研究所講究録 **1477**, 20-27 (2006).

[18] Kikuta,K., W.H.Ruckle, Rendezvous Search on a Star Graph with Examination Costs, European Journal of Operational Research **181**, 298-304 (2007).

[19] Lim,W.S., S.Alpern, A.Beck, Rendezvous Search on the Line with More Than Two Players, Operations Research **45** (3), 357-364 (1997).

[20] Ozsoyeller,D., A.Beveridge, V.Isler, Symmetric Rendezvous Search on the Line with an Unknown Initial Distance, IEEE Transactions on Robotics **29** (6), 1366-1379 (2013).

[21] Ruckle,W.H., K.Kikuta, Rendezvous Search with Examination Cost on a Finite Graph, 京都大学数理解析研究所講究録 **1457**, 78-85 (2005).