1E2-02

The 32nd Annual Conference of the Japanese Society for Artificial Intelligence, 2018

辺ベクトルを用いた定式化による経路推薦手法の提案

Proposal of Route Recommending Method based on Formulation using Edge Vector

柴田 祐樹 高間 康史
Hiroki Shibata Yasufumi Takama

首都大学東京大学院システムデザイン研究科
Graduate School of System Design, Tokyo Metropolitan University

In the domain of tourism navigation, tourist spots as well as route visiting them are important information to be presented to tourists. However, it is difficult to determine tourist spots to visit and the route all at once. This is a challenging problem. While an existing study has tried to solve such a tourism navigation problem by extending Traveling Salesman Problem (TSP) with introduction of weight for nodes (spots), its formulation became complicated. This paper proposes more simple formulation of the tourism navigation problem, which assigns all factors needed for a solution to only edges. The solution using Simulated Annealing method is also proposed.

1. はじめに

交通手段, 情報収集手段の発達に伴い, 個人が自ら観光地の情報を集めて旅行計画を立てることが容易になってきている. しかし, 個人の趣味嗜好というよりも知名度の高い場所が選択肢へ上がりやすい傾向があり, 当該の観光地が混雑する状況を招き, より個人の趣味に一致した場所や経路の選択機会を喪失している. このような状況が続けば, 観光地としての価値を追求するために, その土地本来の良さよりも宣伝性の良さを狙った街作りに重点が置かれる等の弊害も想定され, 訪問者, 現地民双方にとって損失があると考える. また, 居住地付近の散歩やテーマパーク訪問時においても, 好みのスポットを最短時間でより多く回れば個人の満足度が高くなるわけでは必ずしもなく, 途中通過する街並みや景観等も重要な要素である. 従って, 個人の嗜好や時間的制約などに基づく, 画一的ではない経路推薦手法が必要と考える.

目的地を巡回する問題については巡回セールスマン問題 (TSP: Traveling Salesman Problem) として多くの研究がなされている [Gutin 2006] [Feillet 2005] が, 事前に決定したスポット全てを通る経路を発見するため, 所要時間とのバランスを見ながら訪問スポットを選択するような調整を行いにくい. また, TSP を拡張した最適観光経路問題[松田 2004] が定式化されているが, 途中通過する街並みや, 道の景観等を考慮できず、また辺に基づく定式化である TSP にノードの重みを導入するため定式化を複雑にする要因になっている.

本稿では訪問する観光スポットおよびその経路を決定するために必要な情報をすべて辺の重みとして定式化する手法を提案する. 提案手法では, 経路を辺ベクトルにより表し, Boltzmann 分布に従う確率場として定式化する. これにより従来行われていたノードを用いた定式化に比べ, 簡素な表現が得られる. 確率場による定式化は各種統計, 機械学習手法を適用可能とする. 提案手法で用いる 焼きなまし (SA: Simulated Annealing) 法 [Kirkpartrick 1983], 局所最適化法の導出が Markov Chain Monte Carlo (MCMC) 法を適用する過程で自明に行えることを示し, また, 目的関数の形を工夫することで, 複雑な地図上の問題においても局所最適化法を適用できることを示す. 局所最適化法は極値の少ない目的関数を持つ問題の効率的解法として知られており, これを用いることは大規模な問題に対して有効であると考える. 評価実験では、これら提案手法が観光案内で想定される問題に対して有効に動作することを示す。

連絡先:柴田祐樹, 首都大学東京大学院システムデザイン研究科, 東京都日野市旭が丘 6-6, 04258598606, shibata-hiroki@ed.tmu.ac.jp

2. 関連研究

2.1 観光案内に関連する経路問題

与えられたノードをすべて巡回する最短経路を求める問題は TSP として広く知られている. TSP では各ノード間の負荷 (コスト) を定義し, すべてのノードを巡回する経路のみを解として求める. また, 近似解法の一つとして, 局所最適化法に属する 2-opt 法 (k-opt 法) が存在する. 2-opt 法を用いた最適化では局所最適解へ収束する可能性があるが, 焼きなまし法を併用することで大域的最適解へ近づくことが示されている [Kirkpartrick 1983]. しかし, 観光案内では, すべてのノードを巡回する経路が求められるわけではないので, TSP として定式化することはできない.

松田らは TSP を拡張し, ノードに価値を与え, 既定の時間内で価値の総和を最大化する問題を, 最適観光経路問題として定式化している [松田 2004]. しかし, この定式化は辺に対する重み (負荷) に基づく TSP に対し, ノードの価値 (重み) を導入したことが要因となり複雑となっている. また, ノード間の移動経路は実在の経路から切り離されて抽象化されており, 道に対する個人の好みなどを考慮することができない.

2.2 Boltzmann 分布と焼きなまし法

状態集合 $\mathcal{X}$ に属する多次元確率変数 $\mathbf{x} \in \boldsymbol{\mathcal{X}}$ についての目的関数$\phi(\mathbf{x})$ を最小化する問題において, (1)式で与えられる Boltzmann 分布を考える.

$$p(\mathbf{x}) = \frac{\exp\left\{-\frac{1}{T}\phi(\mathbf{x})\right\}}{\sum_{\mathbf{x}\in\mathcal{X}} \exp\left\{-\frac{1}{T}\phi(\mathbf{x})\right\}} \tag{1}$$

$\phi(\mathbf{x})$ の最適解を与える$\mathbf{x}$ の集合を$\mathcal{X}_t$とする時, (1)式は温度 $T \to 0$ の極限で(5)式となることがわかる.

$$p(\mathbf{x}) = \begin{cases} \frac{1}{|\mathcal{X}_\mathrm{t}|} & \text{where } \mathbf{x} \in \mathcal{X}_\mathrm{t} \\ 0 & \text{where } \mathbf{x} \notin \mathcal{X}_\mathrm{t} \end{cases} \tag{2}$$

(1)式の分布を解析的に求め，直接最適解を得ることは困難であり，一般に焼きなまし法を用いて近似解が求められる．焼きなまし法では(3)式を用いて表される条件付き確率を用いて，温度 T を徐々に低下させながら MCMC 法を適用し，$T \to 0$ での分布を近似する．ただし，$\mathbf{y} \in \mathcal{Y}$ は$\mathbf{x}$ の部分集合であり，$\mathcal{Y}$ は$\mathbf{y}$ が取りうる状態の集合を示す．

$$p(\mathbf{y}|\mathbf{x} \setminus \mathbf{y}) = \frac{\exp\left\{-\frac{1}{T}\phi(\mathbf{x})\right\}}{\sum_{\mathbf{y}\in\mathcal{Y}} \exp\left\{-\frac{1}{T}\phi(\mathbf{x})\right\}} \tag{3}$$

3. 提案手法

本稿で対象とする問題は，総移動時間及び目標総移動負荷の制約がある中で，ユーザの経路に対する満足度を最大化する経路を推薦することである．2.1 節で述べた最適観光経路問題と異なり，スポットの価値も含め解の発見に必要な情報をすべて辺に与えることで定式化する．

3.1 節では，経路を辺のベクトルとして表現し，経路に対する目的関数をエネルギー関数とする Boltzmann 分布から生成されるとモデル化する．これにより経路の拡張縮小，最適化を確率的に扱うことができる．3.2 節では MCMC 法により経路最適化法を導出する，3.3 節で非存在辺の概念を導入し，対応する目的関数を定式化することで複雑な地図上においても局所最適化法を適用可能とする．3.4 節で目標移動負荷，辺に対する好みをモデル化した目的関数を示す．

3.1 経路の表現と Boltzmann 分布による定式化

地図を構成するノード集合を $\mathcal{N}$ とする．各ノードを結ぶ辺から構成される，終始点を通る有効な経路を含むベクトルを，$\boldsymbol{e} = (e_i|e_i = (a_i, b_i), a_i \in \mathcal{N}, b_i \in \mathcal{N})$ と表す．辺の方向は始点 a_iから，終点 b_iへ向かうものとする．経路への辺の追加などに対応するため，辺ベクトルに経路に含まれない自己ループ辺を持たせておく．例えば，Fig. 1 (a) において，青い辺を要素として持つ辺ベクトル $\boldsymbol{e}$ は以下のようになる．

$$\boldsymbol{e} = ((1,2),(2,7),(3,3),(4,4),(5,5),(6,6),(7,8))$$

また，Fig. 1 (b) に示す例では以下となる．

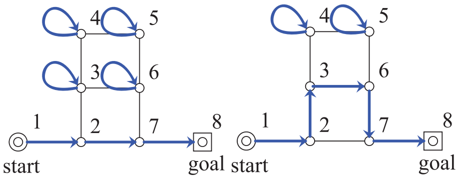
$$\boldsymbol{e} = ((1,2),(2,3),(3,6),(4,4),(5,5),(6,7),(7,8))$$

辺ベクトルに経路および自己ループ辺が含まれるとすることにより，経路長が変化してもベクトル長を固定できる．これにより経路を含む辺ベクトルを確率場として取り扱うことができる．

$\boldsymbol{e}$の確率変数を$\mathbf{e}$とし，その分布$p(\mathbf{e})$を(4)式で与える．

$$p(\mathbf{e}) = \frac{\exp\left\{-\frac{1}{T}\phi(\mathbf{e})\right\}}{\sum_{\mathbf{t}\in\mathcal{E}} \exp\left\{-\frac{1}{T}\phi(\mathbf{t})\right\}} \tag{4}$$

ここで，$\mathcal{E}$ は$\mathbf{e}$ の取りうるすべての状態の集合とする．また，$\phi(\mathbf{e})$ は経路 $\mathbf{e}$ に対する目的関数であり，$\phi(\mathbf{e})$の値が小さいほど$p(\mathbf{e})$の値が高まり，そのような経路が選択される確率が高くなることがわかる．$\phi(\mathbf{e})$ についての最適解集合を $\mathcal{E}_\mathrm{t}$ とする時，(1)式は温度 $T \to 0$ の極限で(5)式となり，$p(\mathbf{e})$ に従う辺ベクトル$\mathbf{e}$は最適経路を与えることがわかる．



(a) (b)

Fig. 1 An example of self loop edges and route extention.

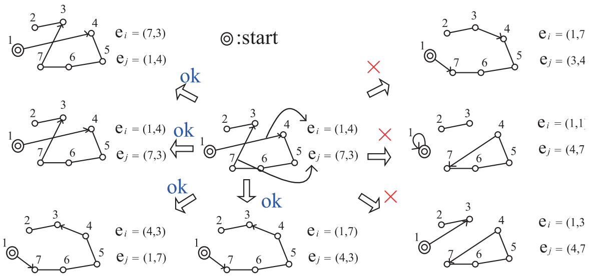


Fig. 2 The edge’s status of $\mathcal{E}_{ij}^{S_2}$ to swap 2 edges.

$$\lim_{T\to 0} p(\mathbf{e}) = \begin{cases} \frac{1}{|\mathcal{E}_\mathrm{t}|} & \text{where } \mathbf{e} \in \mathcal{E}_\mathrm{t} \\ 0 & \text{where } \mathbf{e} \notin \mathcal{E}_\mathrm{t} \end{cases} \tag{5}$$

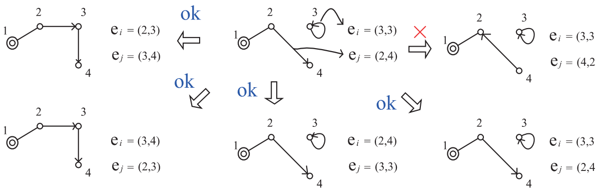
3.2 Markov Chain Monte Carlo 法と局所最適化

(4)式で定義された分布を MCMC 法により求めることは，$\phi(\mathbf{e})$の最適解を与える $\mathbf{e}$ を k-opt 法により求めることに等しい．本稿では 2 辺について最適化を繰り返す 2-opt 法を用いる．最適化対象の 2 辺を確率変数 $\mathbf{e}_i, \mathbf{e}_j \in \mathbf{e}$ で表し，これらを除いた辺ベクトルの部分空間を $\mathbf{e} \setminus \{\mathbf{e}_i, \mathbf{e}_j\}$ とすると，$\mathbf{e}_i, \mathbf{e}_j$ の条件付き分布は(6)式となる．

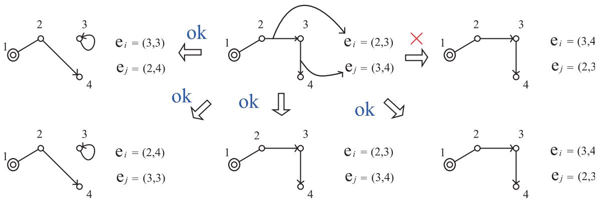
$$p(\mathbf{e}_i, \mathbf{e}_j|\mathbf{e} \setminus \{\mathbf{e}_i, \mathbf{e}_j\}) = \frac{\exp\left\{-\frac{1}{T}\phi(\mathbf{e})\right\}}{\sum_{(\mathbf{e}_i,\mathbf{e}_j)\in\mathcal{E}_{ij}} \exp\left\{-\frac{1}{T}\phi(\mathbf{e})\right\}} \tag{6}$$

ここで，$\mathbf{e}_i, \mathbf{e}_j$ の状態は互いに独立ではなく経路が途切れたり，逆行したりしないようなもののみが許され，そのような状態の組を$(\mathbf{e}_i, \mathbf{e}_j)$，その集合を $\mathcal{E}_{ij}$ とする．2 辺が隣接せず，ともに自己ループ辺でない場合，それらの接続先を入れ替えるための状態組集合 $\mathcal{E}_{ij}^{(S_2)}$ として，Fig. 2 に示す例では，“ok” と書かれた 4 通りのみが含まれる．k 番目の辺 $\mathbf{e}_k$ の要素を $\mathbf{e}_k = (a_k, b_k)$ のように表したとき，$\mathcal{E}_{ij}^{(S_2)}$ は(7)式で与えられる．

$$\mathcal{E}_{ij}^{(S_2)} = \left\{ \begin{matrix} \Big(\big(a_i,b_i\big),\big(a_j,b_j\big)\Big),\Big(\big(a_j,b_j\big),\big(a_i,b_i\big)\Big), \\ \Big(\big(b_i,b_j\big),\big(a_i,a_j\big)\Big),\Big(\big(a_i,a_j\big),\big(b_i,b_j\big)\Big) \end{matrix} \right\} \tag{7}$$



(a) Adding



(b) Removing

Fig. 3 The edges’ status of $\mathcal{E}_{ij}^{(A_2)}$ to add or remove an edge from the route.

また, 経路を拡張縮小するための状態組集合 $\mathcal{E}_{ij}^{(A_2)}$ については, 1 辺が経路に含まれ, 1 辺が自己ループ辺の場合に, 自己ループ辺を追加することによる経路拡張 (Fig. 3 (a)), 隣接する経路上の 2 辺のうち 1 つを自己ループ辺へ変化させ経路から取り除く経路縮小 (Fig. 3 (b)) を考える. Fig. 3 より, 経路拡張, 縮小がともに同じ状態組集合となることがわかり, これらをまとめて (8) 式で与えることができる.

$$\mathcal{E}_{ij}^{(A_2)}=\left\{\begin{matrix}\Big(\big((a_i,b_i),(a_j,b_j)\big),\big((a_j,b_j),(a_i,b_i)\big),\\ \big((a_j,b_i),(a_i,b_j)\big),\big((a_i,b_j),(a_j,b_i)\big)\Big)\end{matrix}\right\} \tag{8}$$

3.3 非存在辺の導入

2 辺の操作だけで一般の経路生成に対応するためには, 経路拡張縮小途中において, 非存在辺が経路に含まれることを一時的に許可する必要がある. Fig. 4 に示す例では (b) の中で赤により示される非存在辺を一時的に許可することで, (a) から (c)の状態まで 2 辺の操作だけで経路を拡張可能となる. もし (b) を認めず, (a) から (c) へ直接遷移させる場合, 3 辺についての状態を考えなければならず、より複雑な操作が必要となる. $\mathbf{e}_i$ が非存在辺の時は罰則$f_p(\mathbf{e}_i)$ が, 存在辺の時は負荷 $f_c(\mathbf{e}_i)$ がそれぞれ与えられるとして, 目的関数$\phi(\mathbf{e})$ を (9) 式で与えることを考える.

$$\phi(\mathbf{e})=\sum_{\mathbf{e}_i\in\mathbf{e}}f_c(\mathbf{e}_i)+\sum_{\mathbf{e}_i\in\mathbf{e}}f_p(\mathbf{e}_i) \tag{9}$$

定数$C_{pb}>1, C_{pc}>0$ を用いて, $\mathbf{e}_i$ が非存在辺の場合に $f_p(\mathbf{e}_i)$ を (10) 式で与えると, $T\to 0$ となるとき, 状態組集合$\mathcal{E}_{ij}^{(S_2)}$, $\mathcal{E}_{ij}^{(A_2)}$ による状態の変更は必ず非存在辺を存在辺へ置き換えるように働く.

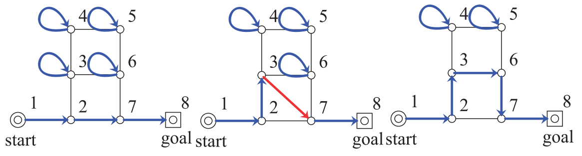
$$f_p(\mathbf{e}_i)=C_{pb}\left(\sum_{s_j\in\mathbf{s}}f_c(s_j)\right)+C_{pc} \tag{10}$$

ここで, $\boldsymbol{s}$ は $\mathbf{e}_i$ の始点と終点を結び, (10)式を最小化する実在最短経路とする. Fig. 5 (a)において, すべての辺の負荷が同じ場合は $\mathbf{e}_i=(2,7)$ に対する $\boldsymbol{s}$ は以下となる.

$$\boldsymbol{s}=\big((2,9),(9,10),(10,7)\big)$$

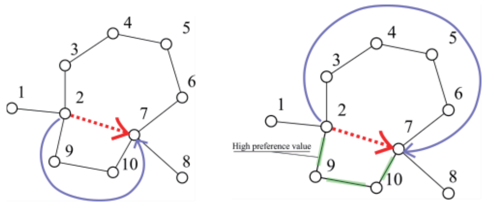
3.4 目標移動負荷と好みの反映

ユーザの辺に対する好みを$f_d(\mathbf{e}_i)$ で与える. 既存研究 [松田 2004] ではノードに対してスポットを表現する値が付加されていたが, 本稿では経路負荷を辺に対する関数$f_c(\mathbf{e}_i)$として与えており, これとの一貫性を取るために辺に対してスポットを割り当てる. 同一ノードを複製した 2 ノード間に距離が 0 の辺を考えることで, ノードに対して割り当てたものと等価な問題を考えることができる. 目標移動負荷値$\mathrm{C_c}$ も考慮した目的関数を(11)式で与える.



(a) (b) (c)

Fig. 4 An example of route extention using non-existing edges.



(a) (b)

Fig. 5 An example of existing shortest route. The Black edges denote existing while red is non existings. The green one have high $f_{\mathrm{d}}(\mathbf{e}_t)$ value.

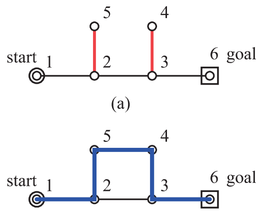


Fig. 6 An example where we have to add $f_d(s_j)$ to $f_p(e)$.

$$\phi(\mathbf{e})=\frac{\beta}{2\sigma^2}\{F_c(\mathbf{e})\}^2+(1-\beta)\left\{F_c(\mathbf{e})-\frac{\sigma^2}{2}\right\}-\sum_{\mathbf{e}_i\in\mathbf{e}}f_d(\mathbf{e}_i)+\sum_{\mathbf{e}_i\in\mathbf{e}}f_p(\mathbf{e}_i) \tag{11}$$

ただし, 条件変数 β, 総移動負荷$F_c(\mathbf{e})$は(12)式で与えられる.

$$F_c(\mathbf{e})=\left|C_c-\sum_{\mathbf{e}_i\in\mathbf{e}}f_c(\mathbf{e}_i)\right|,\quad \beta=\begin{cases}1 & where\ \sigma^2>F_c(\mathbf{e})\\ 0 & where\ \sigma^2\le F_c(\mathbf{e})\end{cases} \tag{12}$$

なお, (11)式は(12)式の条件のもと, $\sigma^2\to 0$ の極限で(13)式となる.

$$\lim_{\sigma^2\to 0}\phi(\mathbf{e})=F_c(\mathbf{e})-\sum_{\mathbf{e}_i\in\mathbf{e}}f_d(\mathbf{e}_i)+\sum_{\mathbf{e}_i\in\mathbf{e}}f_p(\mathbf{e}_i) \tag{13}$$

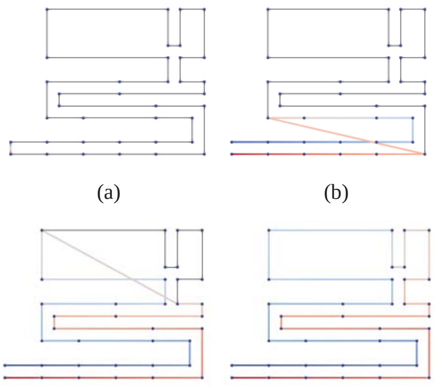
(12)式は総移動負荷$F_c(\mathbf{e})$に対して最小値を持つ目的関数となっており, また, 移動負荷の増加に見合う $f_d(\mathbf{e}_i)$の値が得られるスポットに対し経路の拡張を許可する. σ^2は拡張規模の許容範囲を表現する. また, $f_p(\mathbf{e}_i)$の定義式を(10)式から(14)式に変更する.

$$f_p(\mathbf{e}_i)=C_{pb}\left(\sum_{s_j\in\mathbf{s}}\{f_c(s_j)+f_d(s_j)\}\right)+C_{pc} \tag{14}$$

$\boldsymbol{s}$ は $\mathbf{e}_i$ の始点と終点を結び, (14)式を最小化する実在経路とする. 例えば, Fig. 5 (b)において, 黒と緑の辺はどちらも実在辺であるとし, 黒の辺で$f_c(\mathbf{e}_i)=1, f_d(\mathbf{e}_i)=0$ とし, 緑色の辺では $f_c(\mathrm{e}_i)=1, f_d(\mathrm{e}_i)=1$ とする. この時, $\mathbf{e}_i=(2,7)$ に対し, $\boldsymbol{s}$ は (10)式の場合と異なり, 以下のように与えられる.

$$\boldsymbol{s}=\big((2,3),(3,4),(4,5),(5,6),(6,7)\big)$$

(14)式は 非存在辺を通ってでも, 好みの辺を通った方が目的関数の値を小さくできるような状況を避けるために必要な定義である. そのような状況が起こりえる例を Fig. 6 に示す. (a)に示す赤い辺において$f_d(\mathrm{e}_i)$が十分に高い値を持てば, 非存在辺



(a) (b)

(c) (d)

Fig. 7 The result of edge adding and removing. $T = 0$

(5,4) を通る(b)に示す経路が最適なものとして選ばれる状況が，(11)式を用いている場合にあり得ることがわかる．

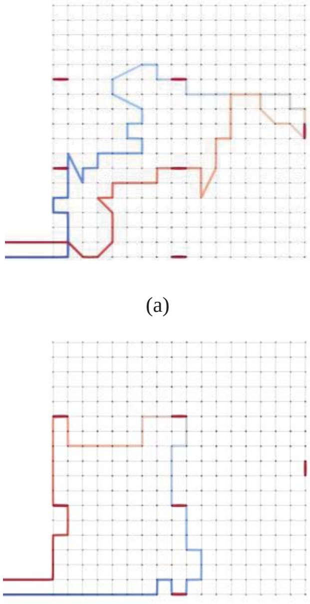
4. 評価実験

Fig. 7 に辺の追加と除外の仕組みが機能している様子を示す．実在辺は黒で表示されており，これら以外のノード対を結ぶ辺はすべて非存在辺である．開始点と終止点が設定されており，開始点と終止点がつながる経路のうち目的関数(9)式が最小となる経路を求める．この問題の最適解は，37 ノードすべてを実在辺のみで通る経路である．(a)に初期状態を示しており，初期経路として開始開始点と終止点を直接結ぶ非存在辺からなる経路を与えている．$f_p(e)$ を定義する(10)式のパラメータとして，$C_{pb} = 1.1, C_{pc} = 0.1$ を用い，$T = 0$ とした．Fig. 7 の(b), (c), (d) はそれぞれ 40, 200, 400 ステップにおける経路である．開始点に近い辺を赤，終止点に近い点を青で，グラデーションを用いて可視化している．局所最適化を繰り返すうちに非存在辺と実在辺の追加と除外を繰り返し，最終的に実在辺のみが残っていることがわかる．

Fig. 8 に(11)式の目的関数で表される目標移動負荷と辺に対するユーザの好みを考慮した問題を解いた結果を示す．ここでは問題を簡潔にするため，18 行 18 列の等間隔におかれたノードからなるマス目上の地図を想定した．ノード数は終始点を含め 326 である．$\sigma = 5, C_c = 60$ とし，赤く表示されている部分は $f_d(e) = 2, f_c(e) = 1$，そのほかの薄い灰色で描画された辺は $f_d(e) = 0, f_c(e) = 1$ としている．これら以外のノード間をつなぐ辺はすべて非存在辺である．(a)は計算開始直後の状態であり，温度T が高いことにより発見的な経路探索を行っていて，非存在辺も含んでいる．斜めの辺はすべて非存在辺である．(b)は計算終了時の状態であり，非存在辺はなくなっており，負荷の目標値として妥当な経路でスポットを巡回する経路を生成していることがわかる．

5. 結論

観光経路推薦に要求される，目標移動負荷および時間に関する制約の中で，訪問すべきスポットとその経路の両方を求める問題に対して，辺のベクトルに対する確率場を Boltzmann 分布を用いてモデル化し，局所最適化法により経路を生成する方法を示した．提案手法は，辺に対するユーザの好みに応じた経路生成が可能なこと，ノードであらわされるスポットに対しても仮想的な辺を配置することで経路推薦が行えることを示した．局所最適化法により効率的に解の探索が行える提案手法は，スケーラビリティに優れるため，広域を対象とした観光案内や，経路上のユーザの好みまで反映させる大規模な問題にも適用可能と考える．今後の展望としては棄却サンプリング [Bishop 2006] を導入し，近傍辺同士の条件付き分布の評価を重点的に行い，遠方への辺検索に要する無駄な探索を抑制することを検討している．また，現在 2 辺の組み合わせによる最適化だけを考慮しているが，より多くの組み合わせを用いた場合についても検証する．さらに，スポットの価値が時間依存性を持つ場合にも対応できるようにする．最適化手法だけではなく，実際の経路推薦に適用された場合のユーザの満足度等，ユーザとのインタラクティブ性を考慮し，経路推薦手法としての有効性についても評価する予定である．



(a)

(b)

Fig. 8 The result of reducted STSP. The red edges have the weight of $f_d(e) = 2, f_c(e) = 1$ while graies have $f_d(e) = 2, f_c(e) = 1$ of it.

参考文献

[Gutin 2006] G. Gutin, A. Punnen.: “The traveling salesan problem and its variations,” Springer Science & Business Media, 2006.

[Feillet 2005] D. Feillet, P. Dejax, M. Gendreau.: “Traveling salesman problems with profits, Transportation science,” Vol. 39, No. 2: pp. 188-205, 2005.

[Helsgaun 2009] K. Helsgaun.: “General k-opt submoves for the Lin–Kernighan TSP heuristic,” Mathematical Programming Computation, Vol. 1, No. 2-3: pp. 119-163, 2009.

[松田 2004] 松田善臣, 名嘉村 盛和, 姜 東植, 宮城 隼夫: 最適観光経路問題とその解法, 電気学会論文誌 C (電子・情報・システム部門誌), Vol. 124, No. 7: pp. 1507-1514, 2004.

[Kirkpartrick 1983] S. Kirkpartrick, C. D. Gelatte, Jr., M. P Vecchi.: “Optimization by simulated annealing, Science,” Vol. 220, No. 4598: pp. 671-680, 1983.

[Bishop 2006] C. M. Bishop, (訳) 元田浩, 栗田多喜夫, 樋口智之, 松本裕治, 村田昇.: “Pattern Recognition and Machine Learning, Springer,” 2006.