

ユーザのリソース消費を考慮した意思決定支援のための 複数観光経路提示手法

平野 陽大^{1,2} 諏訪 博彦^{1,2} 安本 慶一^{1,2}

概要: 一般的に、観光の主目的は満足度を得ることにあり、そのためにユーザは多くの要素（回る観光地やコストなど）や制約（使える時間、予算など）を同時に考慮して計画を立てる必要があるが、これはユーザにとって負担の大きい作業である。この観光経路問題を困難にしている原因の一つに、観光によって得られる満足度と、ユーザが移動や観光地で消費するリソース（金銭・時間・体力）のトレードオフを考えなければならず、単一の最適解を求められないことがあげられる。最終的なユーザの意思決定の手助けを行うためには、これらのトレードオフをすべて考慮した上で、ユーザに複数の解（ツアープラン）を提示することが望ましいが、我々の知る限りではこのようなシステムは存在しない。本論文では、はじめに観光経路探索問題を金銭・時間・体力の初期値を入力とした多目的最適化問題として定式化する。この問題はNP困難であるため、ヒューリスティックアルゴリズムの一つである遺伝的アルゴリズム NSGA-II をベースにした、準パレート最適解を迅速に探索するためのアルゴリズムの提案を行う。最後に、アルゴリズムの有効性を評価するために、京都市東山区にある30ヵ所の観光地を対象に提案アルゴリズムを適用することで、出力される解が有意であり現実的な計算時間で算出できることを示す。

A Multi-Tour-Route Recommendation Method for Decision Support Considering User's Resource Consumption

YODAI HIRNO^{1,2} HIROHIKO SUWA^{1,2} KEIICHI YASUMOTO^{1,2}

1. はじめに

近年、GPS機能が搭載された個人端末の普及により、現在地から目的地に行くための経路を探索するシステムが実現されている。Google Mapに代表されるこのようなナビゲーションシステムは、複数の交通機関を利用することでユーザの望む金銭的・時間的な条件を満たした経路の探索を可能としている。しかしながら、現状のシステムのほとんどはユーザからの詳細な条件の指定が必須であり、希望する経路探索を行うためにユーザは複数の条件を入力することが求められるといった欠点が存在する。この欠点を解消または軽減するため、ユーザのコンテキストや嗜好を考慮し詳細な条件の入力を省略可能なユーザビリティの高い経路計画システムが求められている。

観光の分野においては、NAVITIME TRAVEL^{*1}に代表される様々な観光経路計画システム（ツアープランナー）が存在する。既存のツアープランナーは、ユーザの嗜好を考慮

したプランの推薦を行うが、ユーザはPoI（Point of Interest）の選択などの様々な情報の入力を求められるため、満足のいくプランの立案に多くの時間を要する。この問題を解決する方法として、ユーザによって指定されたエリア内の観光スポットを自動的に選択し、観光スポット間の移動を含むツアープランを推薦する、自動ツアープランニングシステム（以下、Automatic Tour Planning System, ATPS）が提案されている。既存研究として、白石らは金銭コストと観光満足度の間のトレードオフを考慮した多目的最適化問題を解くATPSを提案した([1])。また、Wuらは観光客の残存体力（スタミナ）を制約として考慮し満足度を最大化するATPSの提案を行っている([2])。

一般的に観光においてユーザは満足度や経験を得ることを主目的とするため、ユーザは必ずしも最短経路や最安値での移動を希望するわけではない。しかしながら、ユーザのリソース（金銭、時間、スタミナ）は限られているため、いつも満足度を最大化するような移動を希望するとも限らない。したがってPoI間の移動についてユーザの満足度とリソース（金銭、時間、スタミナ）の間のトレードオフを考慮する必要がある。また、PoIの選択では満足度とリソース消費の間のトレードオフの観点からも複数のPoIを比較

¹ 奈良先端科学技術大学院大学

Nara Institute of Science and Technology

² 理化学研究所 革新知能統合研究センター

RIKEN Center for Advanced Intelligence Project

^{*1} <https://travel.navitime.com/ja/>

することが必用である。このように、ATPS は複数の独立した要因間のトレードオフを抱える最適化問題として定式化することができる。

本稿では、上記の最適化問題をユーザのリソースとしての時間・金銭・体力の3要因と観光で得ることのできる満足度の4つの要因を独立変数として考慮した多目的最適化問題として定式化する。この問題はNP困難であるため、現実的な時間で準最適解を求めるため、ヒューリスティック手法の一つである遺伝的アルゴリズム NSGA-II に基づいたアルゴリズムを設計・実装する。提案アルゴリズムの有効性を評価するために、京都市東山区にある30ヵ所の観光スポットを対象にした観光計画に提案アルゴリズムを適用した。その結果、提案アルゴリズムはトレードオフを考慮した準パレート最適解を、現実的な計算時間である $489.8 \pm 23.4(s)$ で出力できることを確認した。

以降の章構成は以下の通りである。2章では、提案手法に関連した既存研究を概説すると共に、本提案の位置付けを明らかにする。その後、3章では、問題の定式化を行う。4章では、提案アルゴリズムについて述べ、5章で提案アルゴリズムの有効性を示すための評価実験および結果の考察を行う。6章では、5章の結果についての議論を行う。最後に、7章で本論文のまとめを述べる。

2. 関連研究

本章では、本研究と関係する既存研究として観光ナビゲーションシステムと多目的遺伝的アルゴリズムについて概説し、本研究の位置づけを明確化する。

2.1 観光経路探索

既存のツアープランナーシステムのほとんどは、平均レビュースコアが高いもしくはユーザの嗜好に応じたPoIの推薦を行うものである[3], [4], [5]。これらの研究では、観光におけるユーザ満足度の向上に焦点を当てており、フィルタリングシステムを用いたPoIの推薦を行っている。しかし、ルートや観光に関連するその他の要因を含むツアー全体の計画は考慮されていない。また、複数のPoIをつなぐことによってツアープランの推薦を行っている研究がある。倉田ら[6], [7]はCT-Plannerと呼ばれるWebベースの双方向ツアー計画サービスを作成している。このサービスは、ユーザの嗜好を分析しながらルートとツアープランの作成を行うことができる。しかし、観光の満足度とリソースのトレードオフを考慮しておらず、ユーザは複数の情報の入力求められる。

金銭、時間、スタミナなどの観光に関連する他の要因を制約として考慮した研究もいくつか存在する[8], [9], [10], [11]。例えば、Wuら[2]は、残存スタミナ(体力)を拘束条件と見なしたツアー計画サービスを提案している。しかしながら、これらの既存の研究は主な要因としてユーザの満足度

にのみ焦点を合わせており、満足度と他の要因(例えば、金銭、時間、スタミナ)との間のトレードオフは考慮されていない。

満足度とリソースとのトレードオフを考慮した方法もいくつか提案されている。白石ら[1]は満足度と時間のトレードオフを考慮したツアールートを推薦するシステムを提案している。玉城ら[12]は観光のための多目的最適経路問題を定義することでツアーがユーザにもたらす価値と、必用となる金銭のトレードオフを考慮したシステムを提案している。しかしながら、これらの研究は満足度とそれを得るために費やされる要因のうちの一つしか考慮していない。

ユーザは実際のツアー計画の立案を行う際、観光によって得られるもの(経験や満足度)だけではなく、その他の要因(ツアー中に消費する金銭、時間、スタミナ)とのバランス(すなわち、コストパフォーマンス)を考慮している。つまりツアープランナーは満足度とリソース消費の間のトレードオフの観点から、複数のPoIを比較、考慮する必用がある。そこで我々は、ツアープランニング問題を観光客の持つ金銭、時間、スタミナの消費量と観光の満足度を独立変数とする、多目的最適化問題としてモデル化する。一般的に多目的最適化問題を解決する手法として、PSO (Particle Swarm Optimization) や MOGA (Multi-Objective Genetic Algorithm) [13], [14] などがある。NSGA-II は、多様な準パレート最適解を導き出すことができる一般的な MOGA のうちの一つである。本研究の最終的な目標は、複数の多様なツアープランをユーザに提示することである。この目的のためには解に多様性を持たせることが重要である。我々はユーザが保有する3つのリソース、金銭、時間、スタミナの消費量と、ツアーによって得ることのできる満足度とのトレードオフを考慮して、多目的最適化問題として経路最適化問題を解くことによってユーザに複数の多様な解決策を提示する。

3. 観光経路探索問題

本章では、観光経路探索問題を多目的最適化問題として定式化する。

3.1 問題定義

観光及びPoI間の移動によってユーザが消費するリソースを(1)スタミナ、(2)時間、(3)金銭の3要因と設定する。また、考慮する観光におけるユーザの目的を観光満足度を得ることのみに限定することで問題の単純化を行う。本問題は、指定された地域内での観光プランにおけるPoIの選択とPoI間の移動方法を、体力・時間・金銭・満足度の4つの要因を目的関数として、各関数間のトレードオフを考慮する。これは、消費するリソースの最小化(プラン終了後に残るリソースの最大化)と、獲得できる満足度の最大

化を目指す、多目的最適化問題である。

3.2 定式化

PoI を全ての観光地 (PoI) の集合とする。 $\mathbf{X} = \langle x_1, \dots, x_n \rangle$ をツアープランにおいて訪れる観光地 (PoI) の順序付きリストとする。ここで、 $x_i \in PoI$ ($1 \leq i \leq n$) である。 $\mathbf{Y} = \langle y_1, \dots, y_{n-1} \rangle$ を \mathbf{X} の各 PoI 間を移動する時の交通手段 (タクシー, バス, 徒歩など) の順序付きリストとする。 \mathbf{TR} を交通手段の集合とすると、 $y_i \in TR$ ($1 \leq i \leq n-1$) である。また、 $\mathbf{r}_n = (m_n, t_n, s_n)^T$ を観光地 x_n を訪れた後のリソース残量の値のベクトルとする。ここで m_i , t_i , s_i はそれぞれ \mathbf{X} の i 番目の PoI (x_i) を訪れた後の金銭, 時間, スタミナの残量である。 $\mathbf{r}_0 = (m_0, t_0, s_0)^T$ をリソース変数の初期値のベクトルとする。 m_0 , t_0 , s_0 はリソースの初期値であり、それぞれがユーザの予算, ツアーの利用可能時間, 初期スタミナに対応する。 \mathbf{r}_n は式 (1) によってあらわすことができる。ただし、 \mathbf{r}_n の各要素は常に正であることが制約条件となる。

$$\mathbf{r}_n = \mathbf{r}_0 - \sum_{i=0}^{n-1} [\mathbf{CR}(x_i) + \mathit{moveCR}(x_i, x_{i+1}, y_i)] - \mathbf{CR}(x_n) \quad (1)$$

ここで、

$$\mathbf{r}_n = (m_n, t_n, s_n)^T$$

$$\mathbf{CR}(x) = (CM_x, CT_x, CS_x)^T$$

$$\mathit{moveCR}(x, x', t) = (\mathit{moveCM}_{x,x',y}, \mathit{moveCT}_{x,x',y}, \mathit{moveCS}_{x,x',y})^T$$

と表すことができる。 $\mathbf{CR}(x_i)$ は経由地点 x_i での各リソースの消費を、 $\mathit{moveCR}(x_i, x_{i+1}, y_i)$ は経由地点 x_i から経由地点 x_{i+1} までの経路での交通手段 y_i を用いた場合の消費を示す。 $\mathit{moveCM}_{x,x',y}$, $\mathit{moveCT}_{x,x',y}$, $\mathit{moveCS}_{x,x',y}$ はそれぞれ、スポット x からスポット x' に、交通手段 y を用いて移動する時に消費する金銭, 時間, スタミナを表す。これらはあらかじめ与えられる正の値とする。

次に満足度について定義する。 c で表される満足度は、訪れる PoI と、各 PoI 間を移動する際の交通手段により決定する値であると仮定した。このことから、満足度 c は式 (2) ように定式化される。

$$c(\mathbf{X}, \mathbf{Y}) = \sum_{i=0}^{n-1} [\mathit{SAT}(x_i) + \mathit{moveSAT}(x_i, x_{i+1}, y_i)] + \mathit{SAT}(x_n) \quad (2)$$

ただし $\mathit{SAT}(x_i)$ は経由地点 x_i で得られる満足度、 $\mathit{moveSAT}(x_i, x_{i+1}, y_i)$ は経由地点 x_i から経由地点 x_{i+1} の間の経路において、交通手段 y_i を用いて移動する場合に得られる満足度である。ここで定義される満足度は予め与えられている正の値であるとする。

次に制約について述べる。最終観光地 x_n での観光終了後のリソース量は正でなければならない。その制約は次式で表すことができる。

$$0 < m_n \quad (3)$$

$$0 < t_n \quad (4)$$

$$0 < s_n \quad (5)$$

n は、最終目的地以外に 1 箇所以上の観光地を訪問する場合 3 以上となり、全ての観光地数 $|PoI|$ 以下でないといけない。よって、以下の条件が成立する必用がある。

$$3 \leq n \leq |PoI| \quad (6)$$

制約を満たし、残りリソースと満足度を最大化する観光地リスト \mathbf{X} および移動手段リスト \mathbf{Y} を求めることが本問題の目的関数となる。よって目的関数を以下で表すことができる。

$$\mathit{maximize} \ m_n(\mathbf{X}, \mathbf{Y}), t_n(\mathbf{X}, \mathbf{Y}), s_n(\mathbf{X}, \mathbf{Y}), c(\mathbf{X}, \mathbf{Y}) \quad \text{s.t.} \quad (3) - (6) \quad (7)$$

4. 提案手法

3 章で述べた問題は NP 困難であることが知られているナップサック問題を特殊例として含んでいるため NP 困難問題であると考えられる。そのため、実用時間での求解に向けたヒューリスティックアルゴリズムを提案する。本章ではまず、提案アルゴリズムで用いる解のコーディングについて述べ、次にアルゴリズムの詳細を説明する。

4.1 解のコーディング

本アルゴリズムで使用する解候補 (個体) は、図 1 右に示すようにな、PoI 群をなんらかの移動手段を用いて順に訪問する経路であり、各経路は、2 つの連続する PoI 間の移動手段の列として表すことができる。経路は、図 1 左に示す PoI の列と移動手段の列の組に相互変換可能である。遺伝的アルゴリズムの各世代の個体群の保存等は、1 つの列である経路表現 (図 1 右) に変換してから行う。突然変異や交叉といった遺伝子操作は、PoI/移動手段分割表現 (図 1 左) に変換して行う。

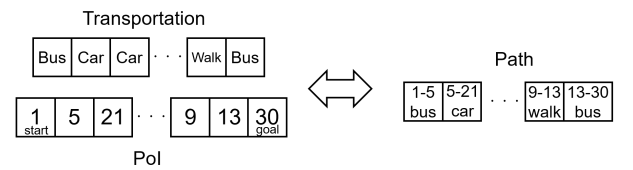


図 1: 解候補の経路表現 (右) と PoI/移動手段分割表現 (左)

4.2 アルゴリズム概要

提案アルゴリズムでは最終的な解の算出に至るまでに、準パレート解の算出と解の精度向上の二段階の計算を行う(図2)。準パレート解の算出では、GAを用いた広範囲の解空間の探索を行う。GAはヒューリスティックアルゴリズムであるため、GAでの探索によって得られた解が、必ずしも全解空間の中でパレート最適解であることは保証されない(保証されるのは算出された解同士に優越の関係がないことのみであり、パレートフロンティアに属す解であることの保証はない)。つまり、より良い(リソース消費と満足度のすべての面において上回る)解が存在する可能性がある。そこで、算出されたそれぞれの解について、2-opt法を用いた局所探索を行うことで解の精度向上を行う。2-optによって更新された解を最終的なアルゴリズムの出力とする。

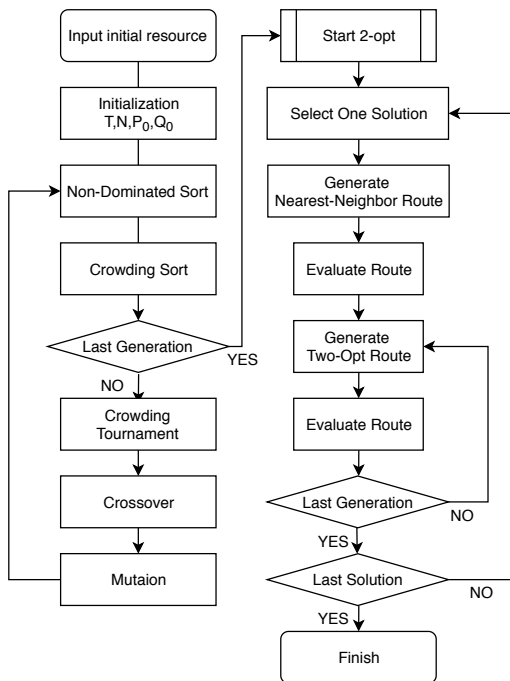


図2: アルゴリズムのフローチャート

表1: GAで用いられる記号とその説明

記号	説明
T	アルゴリズムの最終世代
N	母集団の個体数 ($ P_t = N$, $ Q_t = N$)
P_t	t 世代目の探索母集団 (遺伝子操作などにより更新)
Q_t	t 世代目のアーカイブ母集団 (良解を更新して保存)
R_t	P_t と Q_t の和集合
i	非優越ソートで各個体に与えられる評価値
F_i	ランク i の個体の集合
t	現在のアルゴリズムの世代
α	交叉率
β	変異率

4.3 アルゴリズムの詳細

提案アルゴリズムは、図2左に示すとおり6つのステップから構成されている。アルゴリズムで使用する変数とその説明の一覧を表1に示す。

- (1) 初期化: 初めに世代数(反復回数)を T 、初期母集団の個体数(初期解の数)を N に設定し、初期アーカイブ母集団 $Q_0 = \emptyset$ を初期化する。次に、ランダムに経路データを生成して初期探索母集団 P_0 ($|P_0| = N$) を作成する。
- (2) 非優越ソート: 各世代 t ($1 \leq t \leq T$) において、 $R_t = P_t \cup Q_t$ となる集合 R_t を生成し、この R_t に対して非優越ソートを行い、すべての個体をランク i ごとに分類する。このランクはある解を完全に優越する解の数として算出される。すべての解のランクを決定するために、各解の要素(式(1), (2)によって計算される時間, お金, スタミナ, 満足度)に関して比較を行う。そして R_t に従い、各ランクごとに F_i ($i = 1, 2, \dots$) に解を追加する。
- (3) 混雑度ソート: 非優越ソートにてつけられたランクの上位のものから、次世代の探索母集団 P_{t+1} に、 $|P_{t+1}| + |F_i| \leq N$ を満たす限り、 F_0, F_1, F_2, \dots の順にその要素を加えていく。すなわち、 $P_{t+1} = P_{t+1} \cup F_i$ を実行する。 $|P_{t+1}| + |F_i| > N$ となる場合、その F_i に混雑度ソートを適用し、 F_i のなかでより優れた(他の解と比べて多様性の高い)解を $N - |P_{t+1}|$ 個選択して P_{t+1} に追加する。この時、世代数 $t+1 = T$ ならばアルゴリズムを終了する。
- (4) 混雑度トーナメント: P_{t+1} に含まれる各個体の体力・時間・金銭の残量及び満足度を用いて混雑度トーナメントを行う。トーナメントはランダムに選択した解の組の比較を $n/2$ 回繰り返し、優れた $n/2$ 個の個体を Q_{t+1} に追加する。
- (5) 交叉: Q_{t+1} の中からランダムに $\alpha * N/2$ 組のデータを選択し、これらを親として交叉を実行する(図5)。
- (6) 変異: Q_{t+1} の中からランダムに $\beta * N/2$ 個の個体を選択し(β は変異率)、これらに対し減少変異(図4)を、異なる $\beta * N/2$ 個の個体に対し増加変異(図3)を実行する。その後、 t をインクリメントしたのち次世代の非優越ソートに進む。

4.4 2-optの詳細

- (1) 解の選択: GAで出力された N 個の解 S_1, S_2, \dots, S_N の中から一つの解、 S_i を選択する。ただし、 i はランダムに解につけられた番号であり、未選択の解の番号の最小値とする。

- (2) 最近傍法による準最短経路の生成: S_i の PoI データに対して最近傍法を用いることで、準最短経路を生成する。最近傍法ではスタート地点となる PoI から、最短の徒歩距離で移動できるデータ内の PoI を選択し、次のスタート地点とする。これをすべての PoI を経由して最終目的地に到達するまで行う。移動方法データはランダムに P 個生成して、最近傍法で生成された PoI データと組み合わせ、 P 個の解 ($newS_1, newS_2, \dots, newS_P$) を生成する。
- (3) 準最短経路の評価: 元の解 S_i の評価値 (金銭, 時間, スタミナの残量および満足度) と $newS_j$ の評価値をすべて比較し、すべての要素が少なくとも同等以上の値である時、解 S_i を $newS_j$ に更新する。 j を P まで繰り返し、すべての解 ($newS_1, newS_2, \dots, newS_P$) について比較・更新を行う。
- (4) 2-opt 法による経路の生成: 最初と最後の PoI を除いた解 S_i の PoI データの中から、ランダムに二つの PoI を選択して入れ替える。また、移動方法データをランダムに P 個生成して、の PoI データと移動方法データから Path データ $optS_1, optS_2, \dots, optS_P$ を生成する。
- (5) 2-opt 法に依る経路の評価: S_i の評価値 (金銭, 時間, スタミナの残量および満足度) と $optS_j$ の評価値をすべて比較し、すべての要素が少なくとも同等以上の値である時、解 S_i を $optS_j$ に更新する。 j を P まで繰り返し、すべての解 ($optS_1, optS_2, \dots, optS_P$) について比較・更新を行う。その後、 i をインクリメントして最近傍法による準最短経路の生成に戻る。更新している解が S_N である時、2-opt を終了する。

4.5 変異

本アルゴリズムでは、PoI がランダムに一つ増加する変異 (増加変異) と、PoI がランダムに一つ減少する変異 (減少変異) の二種類を用いる。これは、解の多様性の指標の一つである PoI の数が、多いものまたは少ないものばかりに収束しないように考慮するためである。PoI の数がどちらかに収束してしまうと、GA の探索が局所的なものになっ

てしまい、解の多様性 (大局解の探索) が期待できなくなる。そのため、PoI の数を偏らせることがないように、増加変異と減少変異をそれぞれ変異率の半分の確立で行う。つまり、各個体における変異の関数は、ランダムにどちらかの変異方法が選択される。

増加変異 (Increment-Mutation):

図 3 の例に示すように、初めに、すでに遺伝子に含まれる PoI を除いた他のすべての PoIの中から、追加する PoI をランダムに一つ選択し (例では、PoI7)、また移動方法についてもランダムに選択する (例では、Car)。次に、新たに PoI を挿入する位置をランダムに決定する。この時の挿入位置は、各 PoI の間のどこかが選択される。そして、ランダムに決定された PoI の位置から出ていく方向に対して、ランダムに選択された移動方法を挿入する。

減少変異 (Decrement-Mutation):

図 4 の例において、初めに Path データの長さが 2 の場合はこれ以上に Path を短くできないため、変異無しとなる。十分な経路の長さが担保される場合 (Path データの長さが 2 より大である場合)、削除する PoI の位置をランダムに決定し (例では、PoI21)、削除する PoI の位置と同じ位置の移動方法を削除する (例では、Car)。

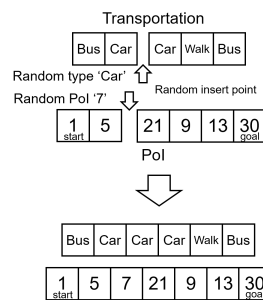


図 3: 増加変異とその例

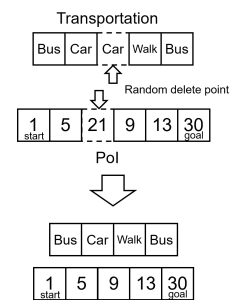


図 4: 減少変異とその例

4.6 交叉

図 5 に示すように、2 つの親個体それぞれに対してランダムにカットポイントが選択され、Parent1 の左 (右) 部分と Parent2 の右 (左) 部分を入れ替える一点交叉を行う。

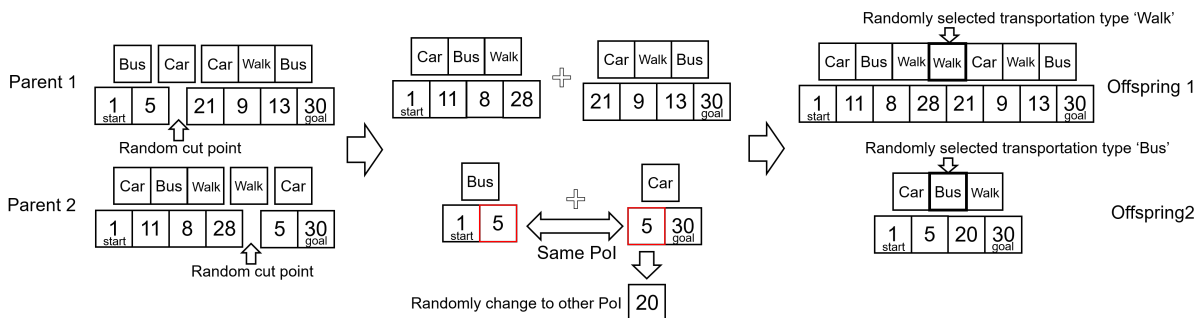


図 5: 一点交叉とその例

表 2: アルゴリズムが出力した解の例

	Remaining Time(s)	Remainig Money(yen)	Reamaig Stamina	Satisfaction	Number of PoI
Solution 1	16350	3993	17085	2100	10
Solution 2	18170	3974	17685	1435	9
Solution 3	15980	4003	16451	2269	9
Solution 4	12520	2869	11608	3522	16

この時、カットポイントによって分割された PoI の間の移動データが一つ削除される。しかしながら、我々のアルゴリズムにおいて、単純に親個体の前後半を入れ替えるだけの一点交叉を行った場合、同じ PoI を複数回含む致死個体（無効な解）を生成する可能性が高い。そのため、PoI が重複していた場合は結合を行う前に、片側の親の左側に含まれる重複している PoI と、もう一方の親の右側に含まれる重複している PoI のうちの一方を、ランダムに重複しない PoI（結合する予定の左右の解の集合のどちらにも含まれていない PoI）に置き換えることによって、致死個体の生成を回避する（図 5 中央）。また、移動手段の列については結合部分に新たにランダムに選択された移動手段を連結時に追加することによって、子の生成を行う（図 5 右図）。

5. 評価実験

本章では、初めに実験環境を、後に実験結果を記述する。

5.1 実験環境

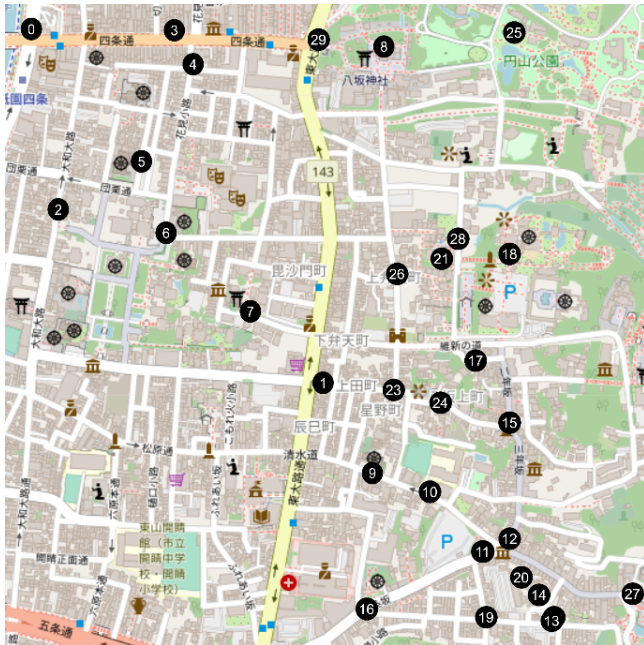


図 6: 東山エリアの PoI のプロット

本実験では、実際にアルゴリズムを京都府京都市東山区にある 30 箇所の PoI を含むエリア（図 6）に対して適用することで、アルゴリズムの出力する解についての評価を行

う。アルゴリズムは Python 言語にて記述し、CPU : Intel Core i7-8550U 1.80GH, メモリ : 16.0GB, OS : Windows 10 Home のスペックを持つマシンで実行する。各経路における金銭・時間の消費データについては、Google Map API から取得したデータを用いる。また、各 PoI で消費する金銭・時間・体力、獲得する満足度及び経路にて消費する体力は経験的に妥当だと思われる値を設定し使用している。実際に必用となる観光満足度のモデル化（[15]で松田らによって提案）と、観光中の体力消費のモデル化は本研究における課題の一つであり、今後の研究で組入れることを計画している。実験結果として、アルゴリズムが出力した解（経路）の残りソースと満足度を比較することで、準パレート最適解として互いに優越関係に無いことを確認する。また、これらの解を地図にプロットして解が準最適とみなせるかどうかの確認・検討を行う。そしてアルゴリズムの実行時間が実用上現実的であるかを確認する。また、6 章では、初期リソースが変化することによって出力される解空間にどのような変化が起こるのか、アルゴリズムによって出力された解がランダム解よりも有意に多様であるのか、2-opt 法が GA で出力された解の質の向上に貢献しているのか、の 3 点について考察を行う。

5.2 実験結果

5.2.1 解の評価

表 2 は初期リソースを時間 : 25000(s), 金銭 : 20000(yen), 体力 : 5000 とした時の 100 個の解のうちの 4 つである。この初期リソースは全 PoI を回りきるのに十分な大きさである。これらの解はそれぞれが優越関係に無い。例えば Solution 1 は Solution 2 と比べて金銭・満足度の 2 点において優れているが、時間・体力の 2 点においては劣っている。表 2 の解はそれぞれ、このように単純に優劣を比較できない関係にある。つまり、アルゴリズムによって求められた解がパレート解ということである。また、図 7, 8, 9, 10 から見て取れるように、最短経路に近いルートを提案していることから、これらの解は少なくとも準最適解であることがわかる。以上のことから、アルゴリズムによって求められた解は準パレート最適解である。

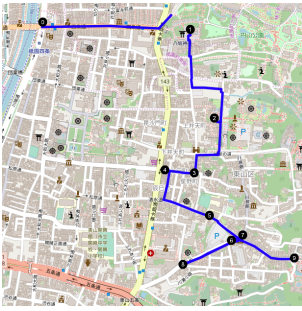


図 7: Solution 1 のルート

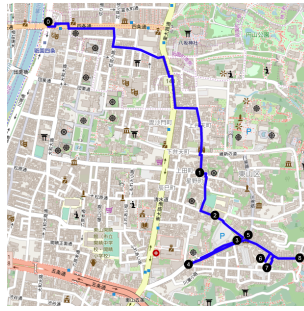


図 8: Solution 2 のルート

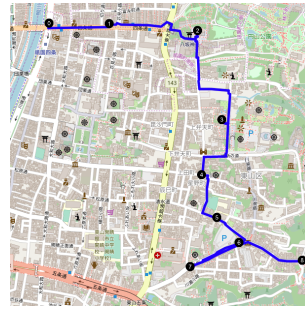


図 9: Solution 3 のルート

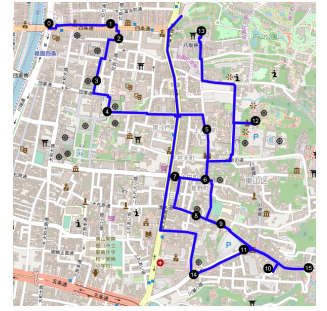


図 10: Solution 4 のルート

5.2.2 計算時間

表 3: アルゴリズム全体の計算時間

	NO.1	NO.2	NO.3	NO.4	NO.5	Average
GA time (s)	241.0	221.4	248.8	252.7	224.9	237.8 ± 14.0
2-opt time (s)	243.8	271.7	259.3	257.8	227.5	256.0 ± 16.9
Sum time (s)	484.8	493.1	508.1	510.5	452.4	489.8 ± 23.4

アルゴリズムの計算時間は、GA の計算時間と 2-opt の計算時間の合計で表される。表 3 は、初期リソースを時間：25000(s)、金銭：20000(yen)、体力：5000 として 500 世代の計算を 5 回行った時の結果である。この初期リソースは全 PoI を回りきるのに十分な大きさとして設定している。この結果から、アルゴリズムは平均 489.8(s) で計算を終了させることができるとわかった。この時間は計算時間として現実的であり、アルゴリズムの性能として満足のいくものである。なお、本手法では、ユーザが観光を計画する際に実際の観光より前にアルゴリズムを実行することを想定しており、オンサイトでの（観光中の）実行は想定していない。観光地でのオンサイトでの利用のためには、計算時間を大幅に（例えば 10 秒以内程度）短縮する必要がある、それは今後の課題である。

表 4 は GA の処理のうち、1 世代を実行するのに必要な時間である。これらの値は 500 世代の GA の試行をした結果の平均値と分散値を示したものである。本アルゴリズムの GA では、表 4 から見て取れるように、初期リソースが大きいほうが 1 世代の計算時間が大きくなっている。これは、初期リソースが小さい場合は大きなリソースが必要となる多くの PoI を回る経路が生成されにくいことが理由である。また、アルゴリズム内の各処理の実行時間は N-D Sort 及び Crowding Sort に必要な時間が最も大きく、逆に Crowding Tournament や Crossover の処理時間は初期リソースに関係

せず一定して小さい値である。つまり、計算時間の改善を行うには、ソート部分のランダムアクセス回数を見直すことが必要である。

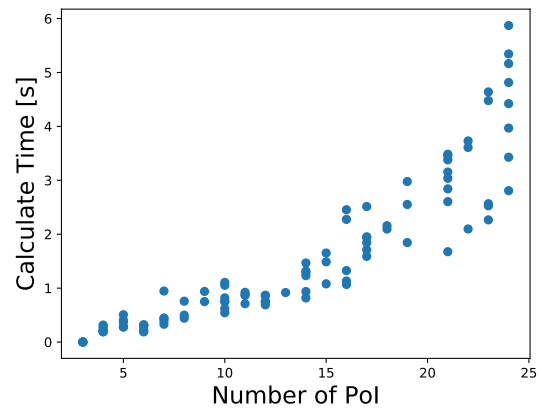


図 11: 2-opt の計算時間

2-opt では N 個の解一つ一つにつき、M 世代の計算を行うため、その計算量は最大 N*M 世代の計算となる。しかしながら PoI が 3 個の経路の場合、出発地点と最終目的地が決まっているため、並び替えが発生しない。このような場合は計算を行わないため、計算時間は 0(s) となる。2-opt による計算時間は図 11 のように PoI の数と正比例する。図 11 から、PoI の数が増えるほど計算時間がばらつくことがわかる。これは PoI の数が増えることで PoI の並び替えの選択肢が増え、解の更新を行うことのできる回数にばらつきが出るのが原因である。計算時間の改善のためには、解の更新にかかる時間を短縮すること、および解の中の 2 点間の入れ替えではなくタブーサーチなどのアルゴリズムを用いることで探索範囲を狭めることが必要である。

表 4: GA1 世代の計算時間

	N-D Sort[s]	Crowding Sort[s]	Crowding Tournament[s]	Crossover[s]	Mutation[s]	Sum[s]
No.1	0.175 ± 0.050	0.176 ± 0.044	0.016 ± 0.004	0.012 ± 0.004	0.072 ± 0.020	0.450 ± 0.111
No.2	0.120 ± 0.017	0.123 ± 0.018	0.014 ± 0.001	0.009 ± 0.001	0.050 ± 0.009	0.316 ± 0.044
No.3	0.119 ± 0.029	0.120 ± 0.030	0.015 ± 0.004	0.009 ± 0.004	0.047 ± 0.013	0.311 ± 0.068

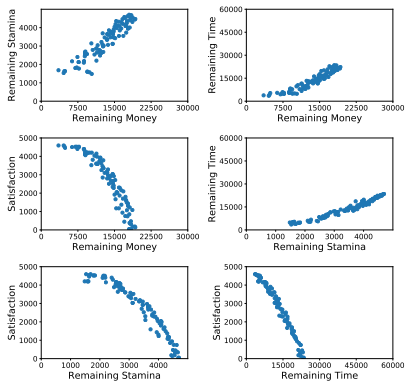


図 12: No.1 の解のプロット

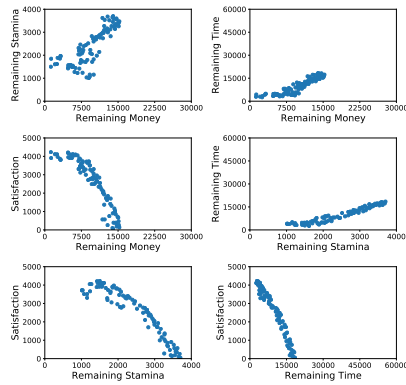


図 13: No.2 の解のプロット

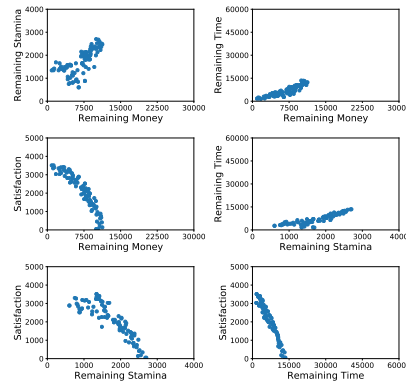


図 14: No.3 の解のプロット

6. 考察

6.1 初期リソースと解の関係

表 5: 初期リソース

	Generation	Time(s)	Money(yen)	Stamina
NO.1	500	25000	20000	5000
NO.2	500	20000	16000	4000
NO.3	500	15000	12000	3000

初期リソースと解の関係を確認するために、表 5 のように初期リソースを設定した。初期リソース No.1 はすべての PoI を十分回れるだけの初期リソースを保有しており、No.2, No.3 はそれぞれ No.1 の初期リソースの 8 割, 6 割の初期リソースを設定した。表 5 の条件において、アルゴリズムが出力した解の分布が図 12 - 図 14 である。これらの図からは金銭、時間、体力の間には正の相関関係が、各リソースと満足度の間には負の相関関係があることがわかった。また、図 12 は図 13, 図 14 と比較して原点から遠ざかる方向に解が広がっていることがわかった。これは初期リソースの値が大きいくちが広い解空間を探索できるためである。低い初期リソースは、解の探索の際に制約となり、探索することができる解空間を狭めるため、初期リソースが大きいくちがより多くの解の探索を行うことができる。図 14 では探索できる解空間が初期リソースによる制限から小さくなっているため、解が密集していることがわかる。さらに、図 12 は図 14 と比べて解のばらつき（解の近似曲線と直交する方向への解空間の広がり）が少ないことがわかった。これは、解空間が狭いときはパレート最適解となる解の選択肢が少なく、パレートフロントから遠いところに解が分布するからである。また、解の多様性と満足させるために、解同士がなるべく等間隔に並ぶように解を出力するが、間隔が十分小さくならず近似曲線と直交する方向へ解が広がってしまうことも、原因の 1 つと考えられる。

6.2 解の多様性

解の多様性を評価するために、NSGA-II において解の多様性を考慮するために用いられている指標である、混雑度距離について考える。混雑度距離 $dist_i$ は、式 (8) のように定義される。

$$dist_i = \sum_{m=1}^n (E_m(i+1) - E_m(i-1)) / (E_m(0) - E_m(n)) \quad (8)$$

$$i \in \{2, \dots, n-1\}$$

式 (8) では、総解数 n として解 $i, i \leq n$ の昇順にソートした評価値を $E_m, m = [money, time, stamina, satisfaction]$ とした。また、式 (8) では定義できない $E_m(0), E_m(n)$ にあたる解（境界個体）においては、 $dist_i = \infty$ とする。この値は金銭・時間・体力の残量と満足度の値をそれぞれ昇順にソートした場合の、両隣の解の間のマンハッタン距離（評価値の差分）を表し、この値が ∞ と定義されている、各評価値の最小値及び最大値の解を除いた残りのすべての解で一定であるとき、個体の分布が均一である。しかし、現実には

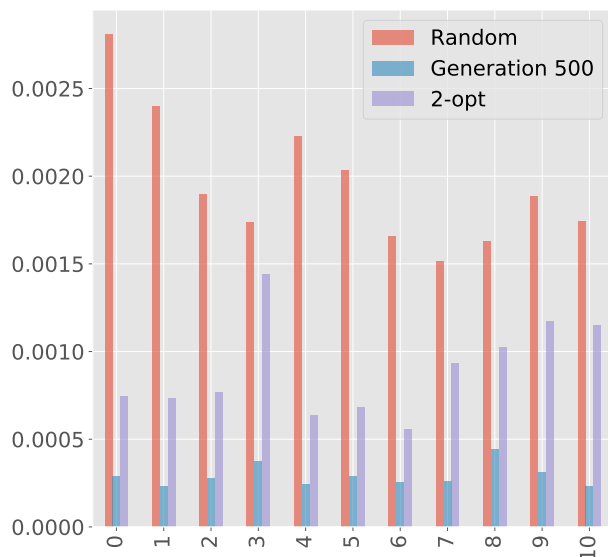


図 15: データの分散値 (ランダム解, GA 最終世代, 2-opt 適用後)

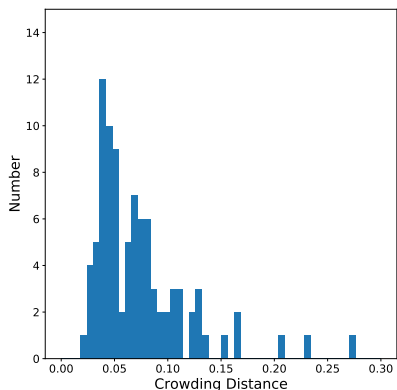


図 16: 混雑度距離 (ランダム解)

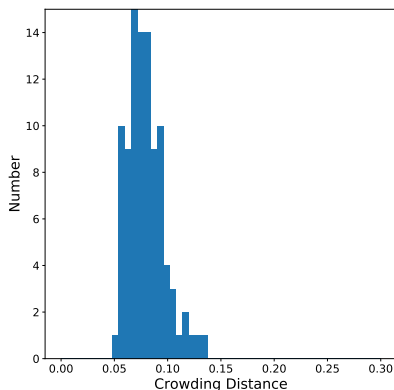


図 17: 混雑度距離 (GA 最終世代)

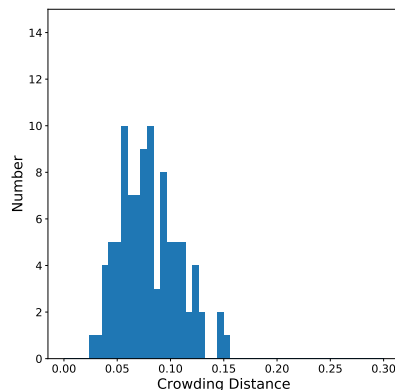


図 18: 混雑度距離 (2-opt 適用後)

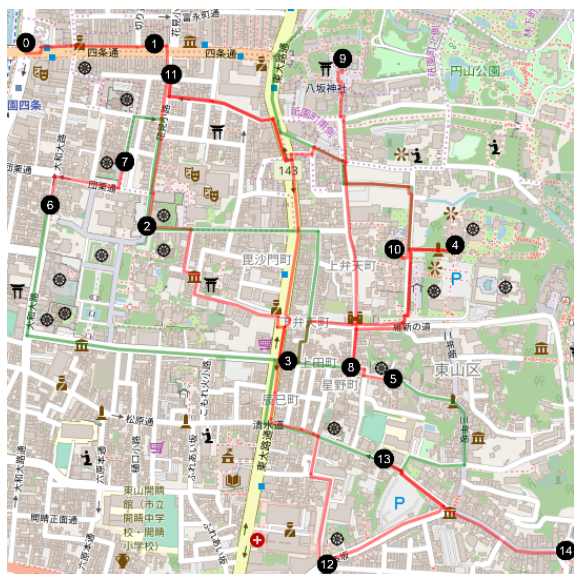


図 19: GA のみを実行して出力した解の一例

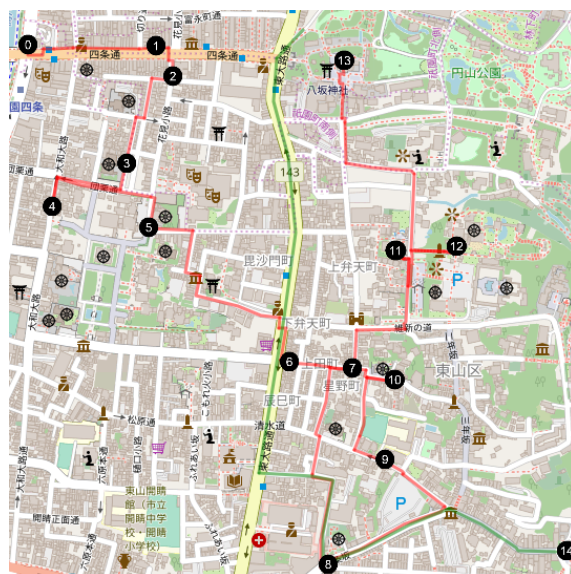


図 20: GA の後に 2-opt を実行して出力した解の一例

混雑度距離が完全に一定になることは考えにくいので、解の混雑度距離の集合の分散を考慮することで、解の多様性の評価を行った。この分散の値が小さいほど、解の間の混雑度距離の差は少なくなり、多様で均一な解集合であると評価できる。図 15 は表 5 の NO.1 の条件で 10 回計算を行い、それぞれのランダム解 (初期解)・GA の 500 世代目の解・2-opt 実行後の解の混雑度距離について、解集合のうち境界個体を除いた集合の分散をプロットしたものである。図 15 において、ランダムに生成した 100 個の解 (図 16) と、GA の 500 世代目の 100 個の解の混雑度距離 (図 17) を比較したところ、図 16 よりも図 17 のほうが混雑度距離の分散が有意に小さいことが分る。つまり、500 世代目の個体は、ランダムに出力された解に比べて多様で均一な解分布が出力されている。これは、GA の中に混雑度距離のばらつきを小さくするような処理があることが原因である。また、2-opt を実行した後の解の集合である図 18 の混雑度距離の分散は、図 17 より大きな値となっている (図 15)。これは、2-opt が解の多様性を考慮せずに、GA によ

て算出された解を基にして局地最適解を求めるアルゴリズムであることから、2-opt 実行後の解は GA によって得られた解の多様性を完全には維持できていないことを示している。しかしながら、図 16 と比べると混雑度距離の分散が小さいことから、GA による最適化を行うことで、後に局所探索を行ってもある程度の解の多様性を担保できることが示された。

6.3 2-opt 法による解の精度向上

2-opt 法による解の精度向上の評価を行うために、得られた解を実際に地図上にマッピングし、その質の評価を行った。図 19 は表 5 の NO.1 の初期リソースを与えた場合に、GA の 500 世代目で算出された解のうちの一つである。この解ではユーザは出発地点と最終到着地点を含めて 15 の PoI をめぐることになっている。そのうち、PoI5・PoI6 の間、PoI7・PoI8 の間、PoI7・PoI8 の間、PoI9・PoI10 の間、の 3 経路は Taxi を利用して、残りの経路は全て徒歩の移

動を推薦されている。しかしながらこの経路は、経路の交差や経路の重複などが多数存在するため、ユーザはより少ないリソース（時間・金銭・体力）で同じ PoI を廻れるであろうことが直観的に理解できる。これは、GA によって求まった解はあくまでも準パレート最適解であり、より優れた解（時間・金銭・体力・満足度の4つの評価値においてすべて上回っている解）が存在する可能性が高いからである。また、図 20 は図 19 と同じ解に対して、2-opt を 200 世代実行して得られた解である。これは準最適解である図 19 の解周辺の局地最適解を求めることに相当する。図 20 で描画された経路は、図 19 と比べて経路内の交差や経路の重複の度合いも少なくなっている。2-opt 法がヒューリスティックアルゴリズムであることから、この解もパレート最適解であるとは言えないが、解の精度向上が行われていることが分かった。これらのことから、2-opt を実行することによって、プロットされた経路は直観的に正しい（我々が計画する時の経路に近い）解を得ることができた。2-opt 法の解の精度を上げる方法として、最近傍法ではなく、ダイクストラ法や A* アルゴリズムを用いた初期解の生成が考えられる。これは、最近傍法では最悪、最も最長となる経路を算出することがあり得るからである。初期解の精度の向上が行われると、2-opt で最適化するのに必要な世代数が少なくなると期待される。

7. おわりに

本稿では、観光経路探索問題に対しリソースとして金銭・時間・体力、獲得物として満足度の4要素を考慮した多目的最適化問題を定義し、本問題を実時間で解く NSGA-II と 2-opt をベースにした多目的遺伝的アルゴリズムを提案した。次に、実際の観光地における 30 箇所の PoI に対して本アルゴリズムを適用し、出力される解の質と求解までの時間の評価を行った。この評価実験では、a) 初期リソースと解の関係、b) 解の多様性、c) 経路の妥当性、d) 計算時間に対して言及を行った。実験結果として、a) リソース間では正の、各リソースと満足度の間では負の相関関係があること、b) ランダムに生成された解に対して優位に多様で均一な解分布が出力されていること、c) アルゴリズムとして妥当な計算時間で解が算出できること、d) GA ではソートが計算時間の支配的要因であり、2-opt では巡る PoI の数が増えるほど計算時間が増えること、などが分かった。また、実験結果として平均 489.8 ± 21.4 (秒) の計算時間でアルゴリズムが準パレート最適解を出力していることが確認できた。今後は GA のソート部分の計算時間の短縮と、2-opt 法のアルゴリズムの改良による計算時間の短縮を目指す。また、今後の課題としては、アルゴリズムによって出力された多様な解の中から、ユーザが求める解（ユーザの中で最も最適解に近いもの）を直観的にかつ負担なく選択できるような UI の開発を行う。

参考文献

- [1] Shiraiishi, T., Nagata, M., Shibata, N., Murata, Y., Yasumoto, K. and Ito, M.: A Personal Navigation System with Functions to Compose Tour Schedules Based on Multiple Conflicting Criteria, *IPJS Digital Courier*, Vol. 1, pp. 528–536 (2005).
- [2] Bing, W., Weihua, S., Yoshihiro, M., Keiichi, Y. and Minoru, I.: Stamina-aware Shigtseeing Tour Scheduling Method, *Information Processing Society of Japan*, Vol. 54, No. 3, pp. 1141–1152 (2013).
- [3] Yu, Z., Feng, Y., Xu, H. and Zhou, X.: Recommending travel packages based on mobile crowdsourced data, *IEEE communications magazine*, Vol. 52, No. 8, pp. 56–62 (2014).
- [4] Interdonato, R., Romeo, S., Tagarelli, A. and Karypis, G.: A versatile graph-based approach to package recommendation, *Tools with Artificial Intelligence (ICTAI)*, 2013 *IEEE 25th International Conference on*, IEEE, pp. 857–864 (2013).
- [5] Chen, X., Zhang, Y., Ma, P., Li, C. and Xing, C.: A Package Generation and Recommendation Framework Based on Travelogues, *Computer Software and Applications Conference (COMPSAC)*, 2015 *IEEE 39th Annual*, Vol. 2, IEEE, pp. 692–701 (2015).
- [6] Kurata, Y. and Hara, T.: CT-Planner4: Toward a More User-Friendly Interactive Day-Tour Planner, *Information and Communication Technologies in Tourism 2014* (Xiang, Z. and Tussyadiah, I., eds.), Springer International Publishing, pp. 73–86 (2013).
- [7] Kurata, Y., Shinagawa, Y. and Hara, T.: CT-Planner5: a Computer-Aided Tour Planning Service Which Profits Both Tourists and Destinations, *ResearchGate* (2015).
- [8] Lucas, J. P., Luz, N., Moreno, M. N., Anacleto, R., Almeida Figueiredo, A. and Martins, C.: A hybrid recommendation approach for a tourism system, *Expert Systems with Applications*, Vol. 40, No. 9, pp. 3532–3550 (online), DOI: 10.1016/j.eswa.2012.12.061 (2013).
- [9] Sebastia, L., Garcia, I., Onaindia, E. and Guzman, C.: e-Tourism: A Tourist Recommendation and Planning Application, 2008 *20th IEEE International Conference on Tools with Artificial Intelligence*, Vol. 2, pp. 89–96 (online), DOI: 10.1109/ICTAL.2008.18 (2008).
- [10] Ricci, F., Nguyen, Q. N., Averjanova, O., Ricci, F., Nguyen, Q. N. and Averjanova, O.: Exploiting a Map-Based Interface in Conversational Recommender Systems for Mobile Travelers (2001).
- [11] Vansteenwegen, P., Souffriau, W., Berghe, G. V. and Oudheusden, D. V.: The City Trip Planner: An expert system for tourists, *Expert Systems with Applications*, Vol. 38, No. 6, pp. 6540–6546 (2011).
- [12] Tamashiro, A., Aguni, S. and Nakamura, M.: Multi-Objective Optimal Routing Problem for Sightseeing and An MOGA for the Problem, *IEICE Technical Report*, Vol. 106 (2007).
- [13] Corne, D. W., Jerram, N. R., Knowles, J. D. and Oates, M. J.: PESA-II: Region-based Selection in Evolutionary Multiobjective Optimization, *Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation*, GECCO'01, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc., pp. 283–290 (2001).
- [14] Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Transactions on Evolutionary Computation*, Vol. 6, No. 2, pp. 182–197 (online), DOI: 10.1109/4235.996017 (2002).
- [15] Matsuda, Y., Fedotov, D., Takahashi, Y., Arakawa, Y., Yasumoto, K. and Minker, W.: EmoTour: Estimating Emotion and Satisfaction of Users Based on Behavioral Cues and Audiovisual Data, *Sensors*, Vol. 18, No. 11 (2018).