

宿泊地を考慮した観光スケジュール作成支援機能の提案

木下 隆正[†] 永田 宗伸[†] 村田 佳洋[†] 柴田 直樹^{††} 安本 慶一[†]
伊藤 実[†]

[†] 奈良先端科学技術大学院大学 情報科学研究科 〒630-0101 奈良県生駒市高山町 8916-5
^{††} 滋賀大学 経済学部 情報管理学科 〒522-8522 滋賀県彦根市馬場1丁目1-1
E-mail: †{takama-k,muneno-n,yosih-m,yasumoto,ito}@is.naist.jp, ††n-sibata@is.naist.jp

あらまし 近年のパーソナルナビゲーションシステムの発達により、携帯端末を利用した様々な経路案内サービスが実用化されている。本稿では、複数日程にわたる観光のためのスケジュール作成機能を提案する。このようなスケジュールは、宿泊地のように「必ずどれかを利用しなければ困る」ような地点を含まねばならない。またスケジュールそのものも長くなる傾向があるため、組合せの数が増えて計算が困難となる。提案手法では、観光地をエリアに分割する。日程におけるそれぞれの日の宿泊地をエリア単位でユーザが指定するようにし、観光の経路におけるエリアの越境を制限することで、計算量を削減する。また、経路を探索するための遺伝的アルゴリズムにおいて、染色体中に対立遺伝子を宿泊地に限定した遺伝子座を持たせる。提案手法を評価するために、奈良、大阪、京都内の39の目的地を入力とするインスタンスを与え、2泊3日の観光スケジュールを算出した。その結果、約38秒で19地点の目的地を巡回するスケジュールが得られた。

キーワード パーソナルナビゲーションシステム, 観光スケジュール作成, 時間制約, 遺伝的アルゴリズム

A method to compose tour schedules with flexible selection of accommodation places

Takamasa KINOSHITA[†], Munenobu NAGATA[†], Yoshihiro MURATA[†], Naoki SHIBATA^{††},
Keiichi YASUMOTO[†], and Minoru ITO[†]

[†] Graduate School of Information Science, Nara Institute of Science Technology, Takayama-chou, 8916-5,
Ikoma-shi, Nara, 630-0101 Japan

^{††} Department of Information Processing and Management, Shiga University, Bamba 1-choume, Hikone-shi,
Shiga, 522-0101 Japan

E-mail: †{takama-k,muneno-n,yosih-m,yasumoto,ito}@is.naist.jp, ††n-sibata@is.naist.jp

Abstract Recent innovations of personal navigation systems have realized various route guidance services using mobile terminals. In this paper, we propose a method to make a sightseeing schedule across multiple days. This type of schedule must include destinations like hotel, such that one has to choose one from many. Also, this type of schedule tends to be longer than single day tour, and thus calculation of schedule is much harder due to increased number of combinations. In the proposed method, the visited region is divided into several areas. The user specifies lodging place of each day by area. The touring route is also restricted so that one won't go back to the area already visited. In order to calculate the route, we use a genetic algorithm such that some alleles are restricted to represent only hotels. In order to evaluate our method, we made a schedule of three-day trip from 39 destinations in Nara, Osaka and Kyoto. We confirmed that our method calculates a tour schedule visiting 19 destinations in 38 seconds.

Key words personal navigation system, tour planning, time restriction, genetic algorithm

1. はじめに

近年の PDA などの携帯端末の高性能化、GPS ユニットの小型化、無線 LAN や PHS、携帯電話などの無線通信手段の発達により、携帯端末上にナビゲーションのシステムが実装されている。このようなシステムは、個人で利用が可能ことからパーソナルナビゲーションシステムと呼ばれ、ユーザのコンテキストに応じて動的に適切なルート案内するシステム [1] や屋内施設でのルート案内 [2]、携帯端末に対し観光案内などの情報を提供するためのシステム [3] などが提案されている。また、ナビゲーションシステムを更に多機能なシステムにしようとする研究として、カーナビで交通事故の情報やお店の情報などさまざまなアプリケーションを統合するシステムがある [4]。これらの既存のシステムは、ある単一の目的地への誘導、あるいは情報の提供を主要な目的としているため、観光時のように複数の地点を巡回するための経路を求めることができない。

複数の地点を巡回する経路を求める問題として、Vehicle routing 問題がある。これは、商品を効率良く配送する際の巡回スケジュールを求める問題であり、遺伝的アルゴリズムを用いて解く方法が提案されている [5]。しかし、Vehicle routing 問題は商品の集積地点や顧客に対する配送に重点を置いているため、観光のための巡回スケジューリング問題としては適切ではない。

我々の研究グループは、観光用パーソナルナビゲーションシステム P-Tour を提案している [6]。これは、ユーザが出発地と出発時刻、帰着地と帰着時刻、複数の観光候補地およびその重要度と時間制約（到着時間帯や滞在時間など）を設定すると、制限時間内で巡回可能かつ最もユーザの希望に添えるような巡回経路と各観光地への到着・出発予定時刻を含むスケジュールを算出し、ユーザに提示する機能を提供するシステムである。

従来の P-Tour は、基本的に日帰りの計画を立案するための機能を備えている。観光船や映画などの時間制約付きの観光地を取り扱う際には、多数の候補の中から時間制約を満たし、かつ満足度の高いものを選び出す手法を取っている。そのため、宿泊地のように「必ずどれかを利用しなければ困る」ような地点を含む経路を求めることが難しい。また、スケジュールが長くなると観光候補地の数及びそれらの組合せの数が増えるため、計算量が大きくなる。これらを解決するために、従来の P-Tour を用いて 1 日ずつ経路を求めて行くことは可能である。しかしこの方法は欲張り法に近いために、前のほうの日程に重要な観光地が集中する傾向があり、また宿泊地は人間が考慮して入しなければならない。

そこで、本稿では宿泊地を考慮した観光スケジュール作成支援機能の提案を行う。この機能の実現のために、以下の工夫を行った巡回経路算出アルゴリズムを遺伝的アルゴリズムを用いて開発する。

- 観光地のエリアによる分割
- 日ごとのスケジューリング分割
- 宿泊地のエリアによる分割

提案機能を評価するために、奈良、大阪、京都内の 39 の目的地

を入力とするインスタンスを与え、2泊3日の観光スケジュールを算出した。その結果、38秒で19地点の目的地を巡回するスケジュールが得られた。

2. 複数日程観光のスケジューリング

2.1 基本方針

P-Tour はスケジュール作成のためにインタラクティブなインターフェースを提供している。ユーザは希望を入力として与えるが、その結果得られたスケジュールが気に入らなければ、入力を修正して再度与えることによって再探索を行うことができる。この過程は何度か繰り返されるために、探索のための時間はなるべく短くする必要がある。そのため、P-Tour では遺伝的アルゴリズム (GA) を用いて準最適解を求める経路探索エンジンを用いている。

2.2 組み合わせの数を減らすための工夫

複数日程になると観光地数が数倍になり、解空間が膨大になる。そこで、多大な観光地数であっても、解空間が膨大にならないように、目的地をエリアごとに分けることで組み合わせの数を減らす。エリアを分ける際に着目すべきことは以下である。

観光地のエリアによる分割: 目的地をエリアと呼ばれる地理的部分領域に分割する。1日の間に観光できる領域は2つのエリアまでに限定することで、可能な組み合わせの数を減少させる。

このエリアは、広すぎると組合せの数が膨大となり、狭すぎると1日に観光できる範囲が狭まるために、適切な大きさに分割されなければならない。分割は探索の直前になされても良いために、ユーザの希望に応じて分割割合を調整することが可能である (県単位、地区単位、市単位、あるいはそれらの組合せ)。図1にエリアを都道府県ごとに分けた例を示す。

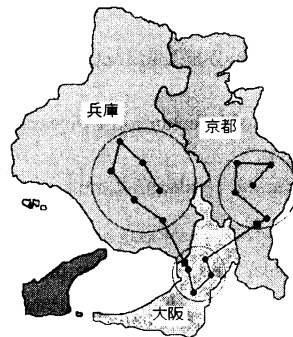


図1 巡回経路探索におけるエリア分け (例: 近畿3県)

この例は、京都エリアに出発地点を持ち、初日に京都エリアで宿泊、2日目に大阪府エリアで宿泊、最終日に兵庫エリアの到着地点を持つ観光経路である。

日ごとのスケジューリング分割: スケジュール立案において難しい点は、終盤のスケジュールが序盤のスケジュールの影響を受けることである。それはスケジュールが長くなるほど深刻である。本稿では、1日ごとにスケジュールを管理することに

よって、巡回にかかる時間がスケジュールに与える影響を抑える。ユーザはそれぞれの日の出発時刻を入力として与える。その前日のいかなる時刻にその日最後の目的地(つまり宿泊地)に到着していたとしても、必ず一定の時刻に出発するようなスケジュールを立案する。

宿泊地のエリアによる分割: 手作業でスケジュールを立案する際には、宿泊地をベースに近場の観光名所を巡回するスケジュールを算出する方法をとることが多い。すると、宿泊地の場所によって、周遊する観光地がある程度決定されてしまう。提案手法では、前述のエリアを利用し、ユーザはエリア単位で宿泊地を指定する。アルゴリズムは宿泊地の重要度と観光地の重要度の双方を同時に考慮し、総合の重要度が高くなるようにスケジュールを立案する。

2.3 問題設定

本節では、宿泊地を考慮した観光スケジュール算出問題を定義する。入力としてデータベース入力とユーザ入力を与えられたときに、評価関数によって与えられるスケジュールの評価値を最大化することが目的となる。

2.3.1 入力

P-Tour は、予め与えられたデータベースの入力と、探索時に与えられるユーザ入力を持つ。ただしユーザは、データベース入力に部分的な変更を加えることもできる(例えば寺が好きだから寺関係の重要度をすべて2倍にするなど)。ここでは、そのような変更はすべて適用済みであるとして扱う。

<データベース入力>

- 地図データ: 道路網を示す有向グラフ $G = (V, E)$ 各エッジには以下の項目を持つ
 - 頂点間距離: 道路網グラフ G のエッジはそれぞれ距離の値を持つ。頂点 v_1, v_2 の間の距離の値は $dist(v_1, v_2)$ で与えられる。
 - 目的地データ: 観光地と宿泊地からなるデータであり、次の項目を持つ。
 - $v_i \in V$: 目的地の位置で地図データの頂点に対応付けられている
 - d_i : 目的地 ($1 \leq i \leq |D|$)
 - pre_i : $D \rightarrow N$: $pre(d_i)$ は目的地 d_i の重要度
 - aco_i : 宿泊地かどうかを示すフラグで、宿泊地なら1、観光地なら0を返す
 - rst_i : 観光地 d_i の到着時刻に対する制約(例: “12:00 以前”)
 - dur_i : 観光地 d_i の滞在時間に対する制約(例: “到着時刻から30分”, “12:00 から30分” など)

<ユーザ入力>

- L : 滞在日数
- 旅程中の出発/到着データ: 以下の6項目を持つ。
 - $pd_s, pd_g \in D$: 初日の出発/最終日の帰着地点を表す。
 - $area_l, (1 \leq l < |L|)$: 最終日を除く宿泊地のエリア。
 - $at_l (1 \leq l < |L|)$: 宿泊地の必須滞在時刻。
 - $ps_l, pg_l (1 \leq l \leq |L|)$: 各日程の出発/到着地点における時間制約を表す。

2.3.2 出力

候補となるスケジュールは、 $s = (D, Stay)$ で表す。 D は全日程の巡回経路であり、 $D = (d'_1, d'_2, \dots, d'_i, \dots, d'_{|D|})$ を表す。 d'_i は i 番目に周る目的地である。また、 $Stay$ は滞在時間の集合であり、 $Stay = (stay_1, stay_2, \dots, stay_{|D|})$ を表す。この時、以下の条件を満たさなければならない。

- 宿泊地の必須滞在時刻 at_l には必ず宿泊地に滞在。
- 同じ観光地を再度通過しない。
- 一日に複数の宿泊地に泊まることはない。
- 各日程について、観光するエリア内の宿泊地に宿泊する。
- 1日に周る観光エリアは2箇所以下である。

2.4 評価関数

各経路 s がどれだけ良いかを表す関数を評価関数と呼び、提案手法では、評価関数 f を次の式で与えることとする。

$$f(S) = \alpha \sum_{j=1}^{|D|} pre_j \cdot timeok(d_j) - \beta \sum_{j=1}^{|D|-1} dist(d_j, d'_{(j+1)}) - \gamma \sum_{l=1}^{|L|} timegoal(pg_l) \quad (1)$$

2 目的地間の経路は A^* 探索により求められる。その距離を移動距離 $dist(d'_j, d'_{(j+1)})$ とする。移動時間は、移動距離から計算する。目的地 $d'_j, d'_{(j+1)}$ 間を移動したとすると、速度を $speed(v'_j, d'_{(j+1)})$ で参照でき、巡回する目的地 d'_j には $stay_j$ 分滞在するとする ($stay_j$ は dur_j を満たす)。 d'_j への到着予想時刻を t_j とすると、 $t_{(j+1)} = t_j + stay_j + \frac{dist(d'_j, d'_{(j+1)})}{speed(d'_j, d'_{(j+1)})}$ で表せる。

関数 $timeok(d_j)$ は、 d_j への到着時間および滞在時間が各時間制約を満たしているとき1、そうでなければ0である。

関数 $timegoal(pg_l)$ は、各日程の最終到着地点の時間と希望到着時間の差を返す。この項により、希望到着時間から遅れたスケジュールは遅れた時間に比例して評価値が悪くなる。

ここで、 α, β, γ は定数(重み係数)であり、 α は巡回経路に含まれた目的地の重要度の和に対する重みである。ただし経路に含まれていても指定された時間制約を満たせていない目的地については、その重要度は加算されない(時間制約を満たした目的地のみの重要度を含んでいる)。 β は総移動距離に対する重みである。総移動距離が長くなると、評価値は悪化する。これは、遠回りをする経路を算出させないためのものである。

3. アルゴリズムの詳細

本章では、提案する複数日程観光スケジュール生成のためのアルゴリズムについて述べる。提案アルゴリズムには組み合わせ最適化の一手法である GA を用いる。

3.1 解のエンコーディング

スケジュールは目的地の順列で表現される(図2)。目的地は巡回する順序で並べられる。また、このリストは1日ごとに分けられている。最終日のものを除き、それぞれのリストの最後の遺伝子座には必ず宿泊地に対応する目的地が割り当てられる。それ以外の遺伝子座には観光地に対応する目的地が割り当てら

れる。リストの長さは可変長である。

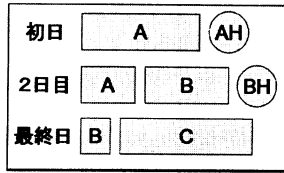


図2 日ごとの染色体 (例: 2泊3日)

エリアの分割: 図2の例では、A,B,CはそれぞれエリアA, エリアB, エリアCを示す。ここでエリアAとして示された帯状の部分は、エリアA内の観光地からなる順列である。また、AH, BHはそれぞれの観光エリアA,Bの宿泊地を表す。2日目は前日観光したエリアAを引き続き観光した後に、次のエリアBを観光している。ここで、1日に観光できるエリアは多くて2とする。また、エリアの境界は1日に1度しか越えることはできないとする。つまり、1日の間に何度もエリア間を行き来することはない。これにより、染色体の構造も簡単になり、ジグザグに移動する経路パターンを防ぐことが出来る。

宿泊地と観光地: 本研究で扱う染色体の遺伝子は、宿泊地と観光地からなり、その二つの要素では取り扱い方が異なる。

観光地は、寺、博物館や公園などの観光スポットを示す。これらは可能ならば1日にいくつかわっても良いが、全日程を通じて同じ観光地を2度以上周ってはならない。また、これら観光地は時間制約を持つかもしれないが、それを満たさないときには巡回しても経路の評価値は増えない(後述)。

宿泊地は、ホテルや民宿などの宿泊施設を示す。これらは宿泊時にのみ利用され、また宿泊地の必須滞在時刻 at_i (例えば午前0:00)にかならずそのいずれかに滞在していなければならない。宿泊地の必須滞在時刻 at_i をまたがない利用は認められず、そのため1日に複数回利用されることはない。しかし、違う日であれば、同じ宿泊地を複数回利用して良い。

3.2 全体の流れ

GAの操作は、初期個体の生成、評価、選択、交叉、突然変異からなる。これらの操作を以下に説明する。個体数を N 、世代数を I とする。

(1) 初期個体の生成: 初期個体群として N の染色体をランダムに生成する。

(2) 評価: 2.4節で示す評価関数を用いる。

(3) 選択: エリート保存戦略とトーナメント選択を用いる。

(4) 交叉: 3.3節で示す交叉を用いる。

(5) 突然変異: 3.4節で示す交叉を用いる。

(6) 終了条件を満たすまで2から5を繰り返す。

3.3 交叉

本提案手法では以下のように二点交叉を行う。

(1) 親1と親2で、同一観光エリア内のそれぞれランダムな場所に交叉点を入れる。

(2) 親1の前半、親2の後半を結合し、子の染色体を作る。

また、以下の場合、該当する観光地を削除する。

(a) 重複しているものは右側のものを削除する。(例: 一泊二日) a,bは異なる観光エリア, AHは観光エリアaの宿泊地, |は交叉点とする。

親1(a1, a2, a5, AH1, a6, |, a7, b1, b4, b5)

親2(a7, a5, AH2, a2, a1, b5, |, b4, b2, b1)

から子を作成すると、

子1(a1, a2, a5, AH1, a6, |, b4, b2, b1)

子2(a7, a5, AH2, a2, a1, b5, |, a7, b1, b4, b5)

となり、子1は重複なしで、子2の()は左と同じものが含まれている。削除後は、

子2'(a7, a5, AH2, a2, a1, b5, |, b1, b4)

となる。

(b) エリアa → エリアb → エリアaなど、一度行ったエリアに再度戻る経路は総移動距離が大きくなる傾向があるため、次のエリアにおける、前のエリアの観光地を削除する。

親1(a1, AH1, a6, |, a7, b1, b2, b8)

親2(a9, a3, AH2, a2, a1, b7, |, b4, b3, b2, b1)

から子を作成すると、

子1(a1, AH1, a6, |, b4, b3, b2, b1)

子2(a9, a3, AH2, a2, a1, b7, |, a7, b1, b2, b8)

となり、子1は問題ないが、子2の削除後(a7 → 消去)は

子2'(a9, a3, AH2, a2, a1, b7, |, b1, b2, b8)

のように修正する。

3.4 突然変異

本アルゴリズムで行う突然変異は、目的地の追加、交換、削除である。以下に観光地および宿泊地の突然変異について述べる。

● 観光地

— 追加: 染色体に含まれていない観光地をランダムな位置に追加する。

— 交換: 染色体に含まれる中で同一エリア内の観光地を、スケジュール上でランダムに交換する。

— 削除: 染色体に含まれる観光地をランダムに削除する。

● 宿泊地

— 追加: 同一エリア内の宿泊地の交換を行う。

4. 複数日程スケジューリングのためのインターフェイス

P-Tourはスケジュール作成のためにインタラクティブなインターフェースを提供している。ユーザは得られたスケジュールが気に入らなければ、入力を修正してから再探索を行わせることができる。本章では、宿泊地を考慮した観光スケジュール作成支援のための再探索機能について述べる。

複数日程に渡るスケジュールは、長く、代替できる観光地の数も大きくなるため、再探索にも時間がかかる傾向がある。スケジュールの一部だけが気に入らないのにすべてのスケジュールを立て直すことは効率的ではない。

提案アルゴリズムで得られるスケジュールは、日程、および

エリアでスケジュールが分割されている(図2)。そこで提案手法では、日程ごとの再探索を可能にする。これは、すでに満足したスケジュールの部分を日程単位(例えば初日と二日目など)で確保しておき、残りの部分だけを再探索するものである。これにより、探索時間が減少するほか、部分的に満足度の高いスケジュールの一部を再探索により失うことを防ぐことができる。ただし、ある日程を確定したとき、その日に利用する(つまり朝と夜の)宿泊地は同様に確保されてしまう。もし宿泊地を変更したいときには、その前後の日程を再探索の対象にしなければならない。

5. 評価実験

5.1 実験環境

実験では、国土地理院発行の数値地図2500(近畿地方2府1県、奈良県、大阪府、京都府)を用いて、一般的なPC(Pentium4 1.7GHz, 512M Memory, WindowsXP pro.)上で提案アルゴリズムを実行した。奈良県、大阪府、京都府をそれぞれ1エリアとし、旅行者の移動速度は自転車と同程度で15km/hとした。

データベース入力として、各エリアごとに10箇所の観光地(表1)、3箇所の宿泊地(表2)を与えた。またユーザ入力として、初日の出発地点及び時刻、初日の宿泊エリア、到着時刻、翌日の出発時刻、2日目の宿泊エリア、到着時刻、翌日の出発時刻、最終日の帰着地点及び時刻を与えた(表3)。

表1 データベース1(観光地)

目的地No.	名称	重要度	希望到着時刻	滞在時間
D1	唐招提寺	5	-	30分
D2	平城京	10	-	30分
...
D10	正倉院	5	-	60分
D14	東本願寺	5	-	60分
D15	平安神宮	3	-	30分
...
D23	金閣寺	8	-	30分
D27	難波	4	-	30分
D28	万博公園	8	-	60分
...
D36	大阪城公園	10	-	60分

表2 データベース2(宿泊地)

目的地No.	名称	重要度
D11	天平ホテル	5
D12	けいはんなホテル	5
D13	奈良パークホテル	10
D24	京都壬花ホテル	10
D25	ゲストハウス トンボ	5
D26	ホテルブラインドシティ山科	5
D37	吹田サニーストンホテル	5
D38	三井アーバンホテル大阪ベイタワー	5
D39	はなてんビジネスホテル	10

表3 ユーザ入力

初日の出発地点	NAIST(奈良)	出発時刻	9:00		
初日の宿泊エリア	奈良県	到着時刻	18:00	出発時刻	10:00
2日目の宿泊エリア	京都府	到着時刻	17:00	出発時刻	9:00
最終日の帰着地点	梅田(大阪)	到着時刻	19:00		

5.2 実験結果及び考察

提案手法を用いて、どのような解が得られるか調べるために、2泊3日の観光スケジュールを30回算出した。その結果、最大で14582、最小値で13236、平均で14076の評価値を持つスケジュールが得られた。その中で最も高い評価値を持つスケジュールを表4に示す。これは19の観光地を巡回する2泊3日のスケジュールで、求めるための処理時間は約38秒であった。

これに対して、宿泊地を自動的に調整する能力を調べるため宿泊地をユーザが予め決定していたと仮定して実験を行った。選ばれる宿泊地については、3×3の9通りの組合せがあるため、これらすべてに対して30回ずつ試行した(つまり270試行)。その結果、評価値の最大値は14612、平均値は13507、最小値は12190であった。いずれの場合においても35秒程度の計算時間が必要であった。しかし、評価値の最大値で提案手法を上回ったのは9通りの宿泊地の組合せのうちの1通りだけであった。つまり、提案手法は準最適な宿泊地の組合せを得ていたと考えられる。

提案手法は柔軟に宿泊地を決定し、2泊3日のスケジュールを作成することができるが、そのために38秒の時間がかかってしまうことがわかった。しかしGAのパラメータの吟味や遺伝演算子(突然変異など)の調整がまだ十分でないため、この値はまだ小さくすることができると考えられる。

表4 出力結果

日程	目的地	到着時刻	滞在時間	出発時刻	
1日目	1.NAIST(出発)	-	-	9:00	
	国立博物館	9:57	30	10:27	
	万葉植物園	10:30	60	11:30	
	新薬師寺	11:32	30	12:02	
	西大寺	12:28	30	12:58	
	平城京	13:13	30	13:43	
	唐招提寺	13:51	30	14:21	
	法華寺	15:38	30	16:8	
	正倉院	16:23	60	17:23	
	奈良パークホテル	18:00	-	10:00	
	2日目	東映太秦映画村	12:7	30	12:37
		平安神宮	13:33	60	14:03
		金閣寺	14:30	30	15:00
元離宮二条城		15:16	60	16:16	
京都駅		16:31	60	17:31	
伏見桃山城		18:4	30	18:34	
3日目	ホテルブラインドシティ山科	19:02	-	11:00	
	西本願寺	11:29	30	11:59	
	東本願寺	12:3	30	12:33	
	万博記念公園	14:43	60	15:43	
	大阪城公園	16:50	60	17:50	
	難波	18:10	30	18:40	
	梅田(帰着)	19:1	-	-	

6. ま と め

本稿では、複数日程の観光において宿泊地を考慮した観光スケジュールの作成支援機能を提案した。また、複数日程の観光問題を定義し、準最適解を求めるためのGAに基づいたアルゴリズムを提案した。大阪府、京都府、奈良県内における39の目的地からなるインスタンスを用いてシミュレートを行い、19の観光地を巡回する2泊3日のスケジュールが生成できることを確かめた。

今後の課題として、最適解との比較、1日に3つ以上のエリアを巡回する経路の生成、公共交通機関の考慮などが挙げられる。

文 献

- [1] Baus, J., Krüger, A., Wahlster, W., "A Resource Adaptive Navigation System", Proceedings of IUI2002: International Conference on Intelligent User Interfaces 2002, ACM Press, New York, 2002.
- [2] Butz, A., Baus, J., Krüger, A., Lohse, M., "A Hybrid Indoor Navigation System", Proceedings of IUI2001: International Conference on Intelligent User Interfaces 2001, ACM Press, New York, pp. 25-33, 2001.
- [3] Cheverst, K., Davies, N., Mitchell, K., and Friday, A., "The Design of an Object Model for a Context-Sensitive Tourist Guide", Computers & Graphics Journal Vol 23. No 6., pp 883-891, 1999.
- [4] 福田, 新, 田中, "複数アプリケーション間における入力フォーカスを管理するJava応用車載端末プラットフォームの開発", マルチメディア, 分散, 協調とモバイル (DICOMO2005) シンポジウム論文集, Vol. 2005, No. 6, pp. 45-48, 2005.
- [5] Z.G. Guo, K.L. Mak., "A Heuristic Algorithm for The Stochastic Vehicle Routing Problems with Soft Time Windows", Congress on Evolutionary Computation Vol 2. No 5., pp 1449-1456, 2004.
- [6] 丸山, 柴田, 村田, 安本, 伊藤 実, "P-Tour: 観光スケジュール作成支援とスケジュールに沿った経路案内を行うパーソナルナビゲーションシステム", 情報処理学会論文誌 次世代移動体通信システム特集, No. 45, Vol. 12, pp. 2678-2687, 2004.