

# 複数日にわたる観光のためのパーソナルナビゲーションシステム

木下 隆正<sup>†</sup> 永田 宗伸<sup>†</sup> 村田 佳洋<sup>†</sup>  
柴田 直樹<sup>††</sup> 安本 慶一<sup>†</sup> 伊藤 実<sup>†</sup>

近年、観光目的地への経路案内や情報提供を行うパーソナルナビゲーションシステムが実用化されている。我々の研究グループでは、複数の観光スポットを効率良く巡回するスケジュールの作成機能を持った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。一般に、観光は複数日にまたがることがあるが、複数日にわたった経路案内を行う場合、スケジュールに含まれる観光地の組合せが多く、また宿泊地を考慮する必要があるため、実行時間でユーザの希望に沿ったスケジュールを算出するには工夫を要する。本論文では、複数日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。提案手法では、観光対象エリアをいくつかの部分エリアに分割し、エリア境界の越境回数を制限することにより組合せ数を減少させる。また最適化手法として遺伝アルゴリズムを用い、宿泊地を符号化する際の遺伝子座を限定することにより、日ごとに宿泊地をただ1つ含む経路を作成する。提案手法を評価するために、東北地方の電子地図（国土地理院数値地図 25000）を対象に、複数日の観光スケジュール作成の実験を行い、31の観光地を巡回する5泊6日の準最適なスケジュールを数十秒で作成できることを確認した。

## A Personal Navigation System to Assist Sightseeing across Multiple Days

TAKAMASA KINOSHITA,<sup>†</sup> MUNENOBU NAGATA,<sup>†</sup>  
YOSHIHIRO MURATA,<sup>†</sup> NAOKI SHIBATA,<sup>††</sup> KEIICHI YASUMOTO<sup>†</sup>  
and MINORU ITO<sup>†</sup>

Recently, various personal navigation system products are available. In this paper, we propose a scheduling function to compose sightseeing tours across multiple days. Scheduling for multiple days is challenging since the number of combinations of destinations becomes large than ever and it must contain an accommodation place for each day. However, the device is required to calculate the sightseeing schedule across multiple days at practical time. In our method, we divide the target area into several sub-areas, and we restrict the number of passing the area boundary to decrease the number of combinations. Also, we propose the genetic algorithm to solve it. In this algorithm, accommodation places are represented in chromosome as exclusive alleles, to make schedules which contain only one accommodation place for each day. In order to evaluate our method, we made various sightseeing schedules across multiple days using the digital map of the Tohoku area (Geographical Survey Institute numeric map 25000), and confirmed that the proposed algorithm could compute the near best schedule in several tens of seconds.

### 1. はじめに

近年、自動車の経路案内を行うカーナビゲーションシステムに加え、携帯電話などでユーザを目的地まで案内するパーソナルナビゲーションシステムが実用化されている。既存のナビゲーションシステムが持つ経路案内機能は、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、観光のように、訪れたい複数の観光スポットを効率良く巡回する目的には不向きである。複数の観光スポット箇所を効率良く訪れるには、旅行会社が提供するパッケージツアー<sup>6)</sup>を利用することが考えられるが、パッケージツアーはあらかじめコースが決められているため、ユーザの希望を完璧に満たすコースが用意されていなかったり、旅行中の予定変更が難しかったりするなど、自由度に欠ける。近年では、旅の目的や予算に合わせて顧客が自由に観光地や宿泊地を決定する観光プランが提供されている（例：マルコポーロプラン）。しかし、これらのフリープランでは、ユーザ自身がから

<sup>†</sup> 奈良先端科学技術大学院大学  
Graduate School of Information Science, Nara Institute  
of Science and Technology

<sup>††</sup> 滋賀大学  
Shiga University

スケジュールリングする必要がある。我々の研究グループでは、複数の観光スポットを時間制約および重要度を考慮しつつ効率良く巡回するスケジュールの作成機能、および、作成したスケジュールに従って利用者に次の目的地へ経路案内する機能を持った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<sup>8)</sup>。しかし、従来の P-Tour では、複数日にまたがる観光スケジュール作成には対応していなかった。スケジュールが複数日にまたがる場合、スケジュールに含まれる観光スポット数が膨大になり、効率の良いスケジュールを作成することが困難になる。また複数日の観光の場合、宿泊費用や夕食の内容、付随施設、周辺環境などに対するユーザの好みを反映して宿泊施設を選定する必要があり、また宿泊地の場所によって、前後に立ち寄ることが可能な観光スポットが限られてくるため、宿泊地の選定が、ユーザの希望に沿ったスケジュールを算出するうえで非常に重要になる。

本論文では、P-Tour で利用可能な、複数日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。提案手法では、組合せ数を減らすために、観光地を複数のエリアに分割し、また宿泊地を必ず含むスケジュールを生成するため、日ごとにスケジュールを分割する。また、それぞれの日の宿泊地を、ユーザがエリアによりおおまかに指定するという形式をとる。また、スケジュールの部分変更機能を提案する。これは、再探索を行う際、必要な部分だけを変更することにより計算時間を短縮する。

以上の機能を実装し、東北 6 県の実際の地図 (国土地理院数値地図 25000) を用いて、複数日の観光スケジュールを求める実験を行った。その結果、108 の目的地を持つインスタンスに対し、50 秒程で 31 の目的地を巡回する 5 泊 6 日のスケジュールを求めることができた。また、15, 16, 17 の目的地を持つインスタンスの最適解と提案手法により得られた解との比較を行った結果、いずれの問題に対しても誤差が 1% 程度である解を求められることを確認した。

## 2. 関連研究

エンタテインメントを目的としたパーソナルナビゲーションシステムとして、ユーザのコンテキスト (位置, 時間, 周辺環境など) に応じて動的に適切なルートを案内するシステム<sup>1)</sup> や屋内施設でのルート案内<sup>3)</sup>, 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<sup>4)</sup> が提案されている。文献 11) では、旅行中すべての期間において案内を行うナビゲーションシステムがないことに着目し、多様な旅行計画、交

通機関乗換えの提供、屋内外の案内のための統合アプリケーションの提案を行っている。文献 9) では、渋滞や交通事故の回避および観光事業の活性化のために、PC や携帯端末または道の駅から、web を通じて、観光地や道路、公共交通機関、天気の情報ユーザに提供する “This Izu Navi” システムを開発している。文献 10) では駅周辺の特定領域のバリアフリープロジェクトの発展を目的とし、健常者だけでなく車椅子、視覚障害の歩行者を対象として、様々な利用法を提供する “HOKO-NAVI” システムを開発している。しかし、これら既存の研究は観光スポットへの順回路および時間をスケジュールリングするための機能を持ち合わせていない。

複数地点を巡回する経路を求める問題として以下のものがある。文献 7) では、2 地点間の移動の間に、コンビニやガソリンスタンドなどの、条件を満たすいずれかの目的地の経路を可能とする経路探索アルゴリズムを提案している。しかし、我々の扱う複数の目的地を重要度を考慮して時間制約付きで巡回する問題に直接適用することはできない。同様の問題としては、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 vehicle routing 問題がある。文献 5) では、この問題をさらに拡張し、顧客の存在や顧客の持つ要求を確率的に表し、ソフトな時間枠を持つ問題として定式化し提案したアルゴリズム HEX (Heuristic Edge-Based Crossover) GA を用いて解く方法が提案されている。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ルに直接適用することはできない。

## 3. 提案手法の概要

本章では、最初に従来研究である P-Tour<sup>8)</sup> の概要について述べた後、複数日にまたがるスケジュールを算出する際に問題となる点について述べ、最後に提案手法における改良点について述べる。

### 3.1 P-Tour の概要

旅行者が観光スケジュールを作成する際には、できるだけ多くの目的地を時間内に効率良く回りたい、各目的地における施設の営業時間やイベントの開催時間、滞在時間などを考慮したい、という要求が存在する。また、候補となる目的地が多数存在し、時間的にすべてを訪れることが不可能な場合には、移動のコストや優先度を考慮してどの目的地を訪れるか決定しなければならない。

P-Tour 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

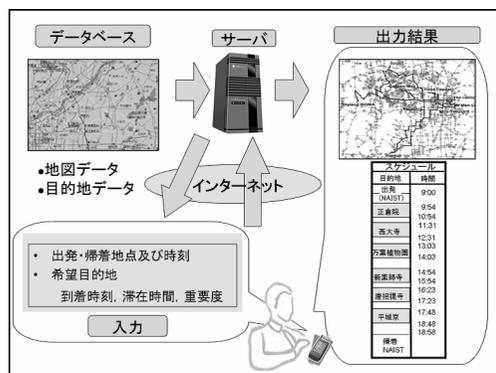


図 1 P-Tour の構成

Fig. 1 The overview of P-Tour.

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有する。また P-Tour は決定したスケジュールに従い、観光中のユーザに経路誘導を行う機能を提供する。

図 1 に示すように、P-Tour は、携帯端末上で実行されるクライアントモジュールとインターネットに接続された PC 上で実行されるサーバモジュールから構成される。なお、サーバから地図、目的地が納められたデータベースが参照できるものとする。P-Tour システムを利用し、ユーザは携帯電話や PDA などを用いて、スケジュール算出やナビゲーションなどのサービスをネットワーク上のサーバから受けることができる。

### 3.2 複数日の観光スケジュールリングを行う際の課題点

従来の P-Tour は、基本的に日帰りの観光計画を立案するための機能を備えている。複数日にまたがる、長いスケジュールの立案は、観光候補地の数およびそれらの組合せの数が増えるため、飛躍的に計算量が大きくなる。これを解決するために、従来の P-Tour を用いて 1 日ずつ経路を求めてゆ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、最初の日程に重要な観光地が集中する傾向があり、また宿泊地はユーザがあらかじめ決め、入力する必要がある。宿泊地の利便性は巡回できる目的地の場所および数に密接に関係する複雑な要素であるが、この評価もユーザが行わなければならない。

また、従来の P-Tour では、日帰りのスケジュールを数秒程度で計算することが可能である。さらに、スケジュールを変更したい際に再計算しても数秒であるため、これらの計算にかかる全時間は数十秒である。しかし複数日のスケジュールになると、問題の規模が

大きくなり、スケジュールの作成に数倍の時間を要し、スケジュールの変更にかかる計算時間が大きくなる。スケジュールの変更は何度か繰り返されるために、探索のための計算時間をなるべく短くする必要がある。

### 3.3 提案手法の概要

提案手法では、(1) 複数日の観光スケジュール立案機能、(2) 実時間性を考慮したスケジュール部分変更機能を実現する。

(1) を実現するにあたり、(i) 目的地の組合せ数を減らすために目的地をエリアごとに分けること、(ii) ユーザは宿泊地はエリア単位で入力し、その中で宿泊地の自動選定を行うこと、(iii) 宿泊地など必ず立ち寄る地点を含ませるために対立遺伝子を宿泊地に限定した遺伝子座を持たせること、の 3 つの方針を採用する。また、(2) を実現するにあたり変更箇所のある日程だけを再探索させることで再計算にかかる時間を短縮させる方針を採用する。

## 4. 提案手法

本章では、経路探索問題の定義、解法のアイディア、およびアルゴリズムについて述べる。

### 4.1 複数日の観光スケジュールリングの問題設定

本節では、複数日の観光スケジュール算出問題を定義する。

#### 4.1.1 入力

P-Tour の入力  $I$  は、あらかじめ与えられたデータベース入力と、探索時に与えられるユーザ入力から構成される。

##### <データベース入力>

データベース入力として、道路地図および目的地のデータが与えられる。複数日にわたる観光は、日帰りの観光に比べ観光地数が数倍になり、探索空間が大きくなる。提案手法では、観光地数が多くなっても探索空間が大きくなりすぎないように目的地をエリアごとに分けることで組合せの数を減少させる。提案手法では、目的地をエリアと呼ばれる地理的な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、可能な組合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。しかし分割の度合いは、ユーザの希望に応じて調整することが可能である。実験においては、1 つの県を 1 つのエリアに割り当てている。

データベース入力は以下の項から構成される。

– 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，データベース上には次の項目を持つ．
  - $v_i \in V$ ：目的地の位置．道路地図データのいずれかの頂点に対応付けられている．
  - $d_i$ ：目的地 ( $1 \leq i \leq |D|$ )
  - $aco_i$ ：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．

<ユーザ入力>

ユーザ入力として，滞在日数や出発帰着時間，目的地の重要度，どのエリアを巡回するかなどのデータを入力する．

重要度は，ユーザがその目的地を訪れたい希望の度合いを示す値である．基本的に，巡回する目的地の重要度の合計が大きい経路が望ましい経路となる．この重要度はデフォルトの値を利用することもできるので，ユーザは特に訪れたい/訪れたくない目的地の値だけを入力してもよい．

手作業でスケジュールを立案する際には，宿泊地を決めてから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，巡回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．提案手法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エリア内で宿泊地をユーザに提示する．また，宿泊地候補を絞ることによって，ユーザが宿泊地を指定することも可能である．

ユーザ入力は以下の項から構成される．

- $L$ ：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．
- 目的地データ：観光地と宿泊地からなるデータであり，ユーザ入力として次の項目を持つ．
  - $pre_i$ ：  $D \rightarrow N$ ：  $pre(d_i)$  は目的地  $d_i$  の重要度．
  - $rst_i$ ：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）．

#### 4.1.2 出力

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．提案手法では，スケジュールにおける時間を 1 日ごとに管理することによって，巡回にかかる時間が後のスケジュールに与える影響を抑える．しかしスケジュール内の観光地は，1 日ごとではなくエリア単位で管理する．このことにより，宿泊地周辺の観光地を，宿泊の前後に巡回するようなスケジュールを得ることが可能になる．また，1 日に回ることでできるエリアの数を制限することにより，ジグザグに移動する経路パターンの生成を防ぐ．

出力であるスケジュールは， $s = (D, Stay)$  で表す． $D$  は全日程の巡回経路であり， $D = \langle d'_1, d'_2, \dots, d'_i, \dots, d'_{|D|} \rangle$  を表す． $d'_i$  は  $i$  番目に回る目的地である．また， $Stay$  は滞在時間の集合であり， $Stay = \langle stay_1, stay_2, \dots, stay_{|D|} \rangle$  を表す．

#### 4.1.3 目的

入力  $I$  に対して評価関数  $f(s, I)$  を最大化するようなスケジュール  $s$  を求めることがこの問題の目的である．実験で用いた評価関数については 4.2.2 項で詳しく述べる．

#### 4.1.4 制約条件

スケジュール  $s$  は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．
- 1 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 日に回る観光エリアは 2 カ所以下である．

### 4.2 経路探索アルゴリズム

本章では，提案する複数日の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合せ最適化の一手法である GA を用いる．

#### 4.2.1 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すこととする（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．

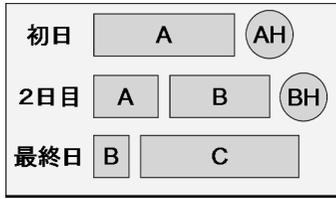


図 2 日ごとの染色体 (例: 2泊3日)

Fig. 2 Chromosome for each day (e.g. 3 days tours).

(1) 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。この構造により、1 日に必ず 1 つの宿泊地を含み、エリア境界を何度もまたいでジグザグに進むような経路を含まないスケジュールを得ることができる。つまり、スケジュールは日程単位で扱われるが、観光地はエリア単位、たとえば  $n$  日目の後半と  $n+1$  日目の前半というような、日程をまたいだ区切りで扱われる。このことにより、1 日ずつ従来手法を用いて求める方法では求めにくい、宿泊後に宿泊エリア周辺を巡回するようなスケジュールをも求めることができる。

(2) 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る。これらを別々に扱うことにより、1 泊ごとに必ず宿泊地を含むような経路を算出することができる。

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 日にいくつ巡回してもよいが、全日程を通じて同じ観光地を 2 度以上巡回してはならない。また、これら観光地は時間制約を持つかもしれない、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。

宿泊地は、ホテルや民宿などの宿泊施設を示す。ユーザに指定されたエリアの中のいずれかの宿泊地が選ばれる。これらは宿泊時のみ利用され、また宿泊地の必須滞在時刻  $at_i$  (たとえば午前 0:00) にかかわ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よい。このことにより、あるエリアを複数日にわたって観光する際に、特定の宿泊地を拠点にするようなスケジュールの生成を許している。

4.2.2 遺伝演算子とアルゴリズムの詳細

GA の操作は、初期個体の生成、評価、選択、交叉、突然変異、局所探索からなる。これらの操作の流れおよび操作法を以下に説明する。個体数を  $N$ 、世代数を  $I$  とする。

- (1) 初期個体の生成
- (2) 評価
- (3) 選択
- (4) 交叉
- (5) 突然変異
- (6) 局所探索
- (7) 終了条件を満たすまで (2) から (6) を繰り返す。

以下にアルゴリズムの詳細を述べる。

初期個体生成

初期個体群として  $N$  の染色体をランダムに生成する。

評価関数

入力  $I$  に対して経路  $s$  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  $f(s, I)$  を次の式で与えることとする。

$$f(s, I) = \alpha \sum_{j=1}^{|D|} pre_j \cdot timeflag(d_j) - \beta \sum_{j=1}^{|D|-1} dist(d_j, d_{(j+1)}) - \gamma \sum_{l=1}^{|L|} goaltime(pg_l) \tag{1}$$

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  $speed(d'_j, d'_{(j+1)})$  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  $timeflag(d_j)$  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  $goaltime(pg_l)$  は、各日程の最終到着地点の希望到着時間からの遅れを返す。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、  $\alpha, \beta, \gamma$  は定数 (重み係数) であり、  $\alpha$  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

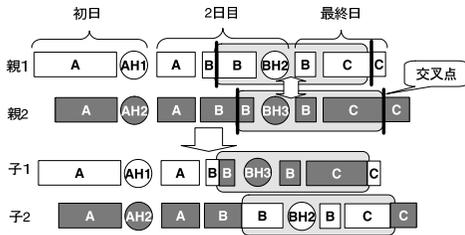


図3 複数日程スケジュールに対する交叉の例

Fig. 3 Crossover for schedule across multiple days.

制約を満たし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）。 $\beta$  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。 $\gamma$  は、各日程の最終到着地点の希望到着時刻からの遅れに対する重みである。これは、なるべく遅れたスケジュールを算出させないためのものである。

また、スケジュールは 4.1.4 項で述べた条件を満たさなければならない。1 つでも満たさない項目がある場合、評価関数の値は 0 となる。

#### 選択

トーナメント選択は、個体群の中からランダムに 2 個体を選び出し、評価値を比べ、優れた評価値を持った個体を次世代に残し、これを、次世代の個数に達するまで行い、次世代の個体群を生成する方法である。このときエリート保存戦略を用いる。これは、個体群の中から最も良い評価値を持った個体（エリート個体）を必ず次世代の個体群に残す方法である。

#### 交叉

染色体の両端の相関関係を破壊する可能性を減少させるため、提案手法では二点交叉を用いる。二点交叉をそのまま用いると 4.2.1 項で述べた染色体構造が破壊されるため、以下の手順を用いて構造を保持する。交叉の例を図 3 に示す。

- (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挿入。
- (2) 親 1 の前半、親 2 の後半を結合し、子の染色体を作成。
- (3) 重複している目的地を削除する。

交叉点を同一エリア内に挿入することにより、エリアの境界をまたぐ回数を 1 日あたり 1 度という制限を守り、交叉演算子に対する探索空間を削減することができる。

#### 突然変異

それぞれの染色体に対して、ランダムな目的地の追

加、交換、削除を行う。目的地が重複した場合ランダムに片方を削除する。また、宿泊地に対しては交換だけを行う。

#### 局所探索

エリート個体に対してランダムに突然変異を用い、評価値が改善されれば染色体を更新する。改善されなければ操作前に戻す。この操作は評価値が改善されるか、指定回数に達すれば終了する。

#### 4.3 スケジュールの部分変更機能

P-Tour では、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システムに再探索を行わせることができる。複数日にわたるスケジュールは、日帰りのスケジュールに比べて期間が長く、可能な組合せの数が飛躍的に大きくなる。そのため、探索のためにかかる時間も大きくなる傾向がある。このとき、スケジュールの一部だけが気に入らないため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、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、日ごとの再探索を可能にする。これは、得られたスケジュール内でユーザが満足している部分を日単位（たとえば初日と 2 日目など）で確保しておき、変更箇所のあるスケジュールのみ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失うことを防ぐことができる。

#### 5. 評価実験

提案手法の有効性を調べるために、(1) 解の品質、(2) 滞在日数に対するスケラビリティ、について実験・評価を行った。

実験では、国土地理院発行の数値地図 25000（東北 6 県、青森県、秋田県、岩手県、山形県、宮城県、福島県）を用いて、一般的な PC（Pentium4 3.0 GHz, 1,536 MB Memory, WindowsXP pro.）上で提案アルゴリズムを実行した。県単位で部分エリアに分け、旅行者は自動車での移動を想定し、40 km/h とした。なお、予備実験から評価関数の係数を  $\alpha = 100$ ,  $\beta = 1$ ,  $\gamma = 10$  とし、GA における個体数、世代数をそれぞれ 300, 1,000 とした。GA におけるこれら 2 つのパラメータを以後標準パラメータと呼ぶ。

##### 5.1 解の品質の評価

提案手法により得られる解の品質を評価するために、最適解との比較について実験を行った。用いたインスタンスは以下のとおりである：データベース入力とし

表 1 最適解との比較

Table 1 Comparison with optimal solutions.

| 総目的<br>地数 | 評価値 | 最適値 | 誤差   | 計算時間<br>(提案手法) | 計算時間<br>(最適解) |
|-----------|-----|-----|------|----------------|---------------|
| 15        | 710 | 715 | 0.7% | 19.6 sec       | 36 min        |
| 16        | 740 | 748 | 1.0% | 19.9 sec       | 3 hour        |
| 17        | 763 | 771 | 1.1% | 20.4 sec       | 17 hour       |

て青森県, 秋田県, 岩手県の道路地図を与え, さらに各県ごとに 15 の観光地と 3 つの宿泊地を設定した. すべての観光地の重要度を 1, 滞在時間は 60 分に設定した. 以上の条件の下, 青森県の竜飛崎を出発し, 秋田県と岩手県内でそれぞれ 1 泊した後, 岩手県内の浄土が浜に到着する 2 泊 3 日のスケジュールを, 提案手法と全探索アルゴリズムによってそれぞれ算出した. 得られた評価値および計算時間を表 1 に示す. これら結果は 10 試行の平均値である.

15, 16, 17 目的地すべての場合において最適値から 1% 程度の誤差の評価値が得られており, 提案手法により得られる解の品質は十分高いと考えられる. また, 17 目的地のインスタンスにおける最適解の探索の計算時間は 17 時間だが, 提案手法は 20 秒ほどで準最適解を導き出すことができた.

## 5.2 滞在日数に対するスケーラビリティ

提案手法がどれくらいの大さのエリア数および目的地数まで適用できるかを示すために, 滞在日数に対するスケーラビリティについての実験を行った. データベース入力として東北 6 県 (青森, 秋田, 岩手, 山形, 宮城, 福島) の道路地図を与え, 各県に 15 の観光地と 3 つの宿泊地を設定した. それぞれの観光地の重要度は 1~3 で, 滞在時間はすべて 60 分である. 総目的地数は  $(15 + 3) \times 6 = 108$  である.

表 2 に, 各滞在日程に対し, 標準パラメータを用いて実験を行ったときの平均巡回目的地数, 平均の総移動距離, 各日程の最終到着地点の希望到着時刻からの遅れおよび計算時間を示す. ここで巡回目的地数とは, 得られたスケジュール内でいくつ観光地を巡回できたかを示す値である. これらの値は 10 試行の平均値である. 提案手法の計算時間は, インスタンスのサイズに対してほぼ線形に増加している. これは, 評価関数においてスケジュールを評価するためにかかる計算時間の増加が原因であると考えられる. また, 遅延時間については, 5 泊 6 日のスケジュールでさえ 5 分ほどの遅れに抑えることができている. これは十分な精度だと考えられる.

これらのような規模の大きなインスタンスに対して得られた解の品質について評価したいが, 最適解を取

表 2 スケジュールの性質および計算時間

Table 2 Schedule properties and computation time.

| 滞在日数    | 巡回目<br>的地数 | 総移動<br>距離 | 各日程の最終到着<br>地点の希望到着時<br>刻からの遅れ | 計算時間     |
|---------|------------|-----------|--------------------------------|----------|
| 2 泊 3 日 | 15         | 441 km    | なし                             | 23.2 sec |
| 3 泊 4 日 | 19         | 598 km    | なし                             | 32.3 sec |
| 4 泊 5 日 | 25         | 747 km    | 2 分                            | 40.8 sec |
| 5 泊 6 日 | 30         | 890 km    | 5 分                            | 51.2 sec |

表 3 極端に大きなパラメータを与えたときの評価値との比較

Table 3 Result of standard parameters compared with big parameters.

| 滞在日数    | 標準パラメータ<br>を用いて得た評<br>価値 | 極端に大きなパ<br>ラメータを用い<br>て得た評価値 | 誤差 (%) |
|---------|--------------------------|------------------------------|--------|
| 2 泊 3 日 | 2,264                    | 2,359                        | 4.1    |
| 3 泊 4 日 | 2,492                    | 2,619                        | 4.8    |
| 4 泊 5 日 | 2,879                    | 3,037                        | 5.2    |
| 5 泊 6 日 | 3,224                    | 3,435                        | 6.2    |

めることは現実的ではない. GA は, 個体数と世代数を増やすことにより, 計算時間とメモリをより多く必要とする代わりに得られる解の品質が良くなることが知られている. そこで本実験では, 標準パラメータを用いた提案手法と, 極端に大きなパラメータ (個体数 1,000, 世代数 5,000) を与えた提案手法を比較した. 表 3 に各滞在日程に対し, 得られた評価値の平均値, およびそれらを比較したときの差異を示す. この結果は 10 試行の平均である. いずれの場合においても, 評価値の差は 5% 程度までに収まった. このことから, 標準的パラメータを与えた場合の提案手法により, 計算時間に対して効率的に良い解が得られると考えられる.

日程ではなくエリアあたりの目的地数が増加した場合にも, 必要となる計算時間が長くなると考えられる. 30 という観光地数は東北地方の旅行ガイドを参考に決定したが, 奈良や京都などの観光地では狭い地域に多数の観光地が密集している. このような場合には, 密集したいくつかの観光地をいくつかセットにして 1 つの観光地として取り扱う, 複数の目的地を巡回する部分経路を経路単位で取り扱う, といった拡張が考えられる.

さらに, 利用人数に対するスケーラビリティの問題がある. しかし提案システムはスケジュール計算のときだけ呼び出されるため, 同時にアクセスされる数はそう多くないと考えられる. また, 探索結果を蓄積し, 初期個体群の生成に利用することにより計算時間を短縮したり, 計算力のある PC を利用できる場合に地図データをダウンロードして手で計算させたりする方

表 4 出力結果: 5泊6日の観光スケジュール

Table 4 Output: 6 days schedule for sightseeing.

| 日程       | 目的地             | 到着時刻  | 滞在時間 | 出発時刻  |
|----------|-----------------|-------|------|-------|
| 1日目      | 龍飛崎 (A6)        | -     | -    | 9:00  |
|          | 青森温泉 (A2)       | 10:44 | 60   | 11:44 |
|          | 黒崎神社 (A15)      | 12:18 | 60   | 13:18 |
|          | 浪岡城址 (A14)      | 13:29 | 60   | 14:29 |
|          | 弘前公園 (A9)       | 14:42 | 60   | 15:42 |
|          | 弘前プリンスホテル (AH2) | 17:55 | -    | 10:00 |
| 2日目      | 黒石温泉郷 (A10)     | 10:38 | 60   | 11:38 |
|          | 康楽館 (B6)        | 12:57 | 60   | 13:57 |
|          | 玉川温泉 (B5)       | 15:32 | 60   | 15:32 |
|          | 大館樹海ドーム (B13)   | 16:40 | 60   | 17:40 |
| 3日目      | 秋田ビューホテル (BH1)  | 17:53 | -    | 9:00  |
|          | 象潟 (B14)        | 11:2  | 60   | 12:2  |
|          | 鳥海山 (Y10)       | 12:31 | 60   | 13:31 |
|          | 最上義光記念館 (Y1)    | 15:39 | 60   | 16:39 |
|          | 上山温泉 (Y3)       | 16:55 | 60   | 17:55 |
| 4日目      | 天童山寺ホテル (YH3)   | 18:00 | -    | 11:00 |
|          | 蔵王 (Y8)         | 11:26 | 60   | 12:26 |
|          | 飯坂温泉 (F10)      | 13:24 | 60   | 14:24 |
|          | 小倉寺観音 (F9)      | 14:38 | 60   | 15:38 |
| 5日目      | 郡山ホテル (FH2)     | 16:45 | -    | 9:00  |
|          | 二本松市歴史資料館 (F15) | 9:35  | 60   | 10:35 |
|          | 福島市民家園 (F11)    | 11:10 | 60   | 12:10 |
|          | 信夫文知摺 (F8)      | 12:20 | 60   | 13:20 |
|          | 白石城 (M7)        | 14:6  | 60   | 15:6  |
| 6日目      | 竹駒神社 (M6)       | 15:49 | 60   | 16:49 |
|          | 松島ホテル (MH2)     | 18:00 | -    | 10:00 |
|          | 松島 (M3)         | 10:6  | 60   | 11:6  |
|          | 伊豆沼 (M10)       | 12:16 | 60   | 13:16 |
|          | 巖美溪 (I10)       | 14:17 | 60   | 15:17 |
|          | 中尊寺 (I9)        | 15:28 | 60   | 16:28 |
| 花巻 (I13) | 17:44           | -     | -    |       |

法などが考えられる。

最後に、標準パラメータを用いて得られた5泊6日のスケジュールの具体例を表4、図4に示す。図中のアルファベットはそれぞれ以下のエリア内の観光地であることを示している：A：青森エリア，B：秋田エリア，Y：山形エリア，F：福島エリア，M：宮城エリア，I：岩手エリア。またHは宿泊地であることを示す。番号はデータベース上の通し番号である。

## 6. おわりに

本論文では、複数日の観光スケジュール作成機能を提供するパーソナルナビゲーションシステムを提案した。複数日の観光スケジュールリング問題を定義し、この問題の準最適解を得るGAに基づいたアルゴリズムを提案した。また、得られたスケジュールをユーザが迅速に修正するための、複数日程における経路の再探索機能を提案した。

実装した経路探索エンジンに対し、国土地理院の数値地図25000を用いて評価実験を行った。東北6県

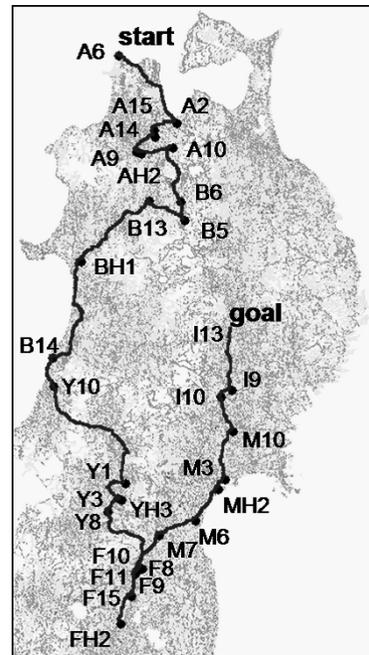


図4 5泊6日の観光経路

Fig. 4 Sightseeing path of 6 days schedule.

における108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、50秒ほどで31の目的地を巡回する5泊6日のスケジュールが生成できることを確かめた。

我々の提案手法では、部分エリアの分割を手作業で行う必要がある。今後は、この分割をユーザの希望を反映して自動的に行う手法を提案していく予定である。

## 参考文献

- 1) Baus, J., Krüger, A. and Wahlster, W.: A Resource Adaptive Navigation System, *Proc. IUI2002: International Conference on Intelligent User Interfaces 2002*, ACM Press, New York (2002).
- 2) Baus, J., Krüger, A. and Wahlster, W.: A resource-adaptive mobile navigation system, *Proc. International Conference on Intelligent User Interfaces*, San Francisco, pp.15-22 (2002).
- 3) Butz, A., Baus, J., Krüger, A. and Lohse, M.: A Hybrid Indoor Navigation System, *Proc. IUI2001: International Conference on Intelligent User Interfaces 2001*, pp.25-33, ACM Press, New York (2001).
- 4) 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).

- 5) Guo, Z.G. and Mak, K.L.: 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) JTB: パッケージツアー .  
http://www.jtb.co.jp/kokunai/PKG/
  - 7) Kanoh, H. and Nakamura, N.: Route Guidance with Unspecified Staging Posts using Genetic Algorithm for Car Navigation Systems, *IEEE Conference on Intelligent Transportation Systems*, pp.119-124 (2000).
  - 8) 丸山敦史, 柴田直樹, 村田佳洋, 安本慶一, 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*, Vol.45, No.12, pp.2678-2687 (2004).
  - 9) Miyauchi, H., Shiga, K., Omoto, M., Hamanaka, T. and Kano, K.: Style of providing information on tourist resort, *World Congress on Intelligent Transport Systems*, pp.3146-3153 (2004).
  - 10) Sakagawa, M., Takeuchi, A., Nishio, I. and Inoue, T.: Pedestrian navigation system in Nagoya, *World Congress on Intelligent Transport Systems*, pp.3373-3380 (2004).
  - 11) Rehrl, K., Leitinger, S., Bruntsch, S. and Mentz, H.J.: Assisting orientation and guidance for multimodal travelers in situations of modal change, *IEEE Intelligent Transportation Systems Society*, Austria (2005).
- (平成 18 年 3 月 31 日受付)  
(平成 18 年 10 月 3 日採録)

#### 木下 隆正

1980 年生。2004 年同志社大学工学部知識工科学卒業。2006 年奈良先端科学技術大学院大学情報科学研究科前期課程修了。現在、松下電器産業(株)に勤務。

#### 永田 宗伸

2006 年奈良先端科学技術大学院大学情報科学研究科前期課程修了。現在、富士ソフト ABC(株)に勤務。遺伝的アルゴリズム、パーソナルナビゲーションシステムの研究に

従事。

#### 村田 佳洋(正会員)

1975 年生。2003 年奈良先端科学技術大学院大学情報科学研究科後期課程修了。2003 年より奈良先端科学技術大学院大学情報科学研究科助手。遺伝的アルゴリズム、エージェント技術等の研究に従事。

#### 柴田 直樹(正会員)

1974 年生。2001 年大阪大学大学院基礎工学研究科情報数理系専攻修士後期課程修了。現在、滋賀大学経済学部情報管理学科助教授。分散システム, ITS, 遺伝的アルゴリズム等の研究に従事。

#### 安本 慶一(正会員)

1991 年大阪大学基礎工学部情報工科学卒業。1995 年同大学大学院博士後期課程退学後、滋賀大学経済学部助手。1997 年モンリオール大学客員研究員。2002 年より奈良先端科学技術大学院大学情報科学研究科助教授。博士(工学)。分散システム, マルチメディア通信システムに関する研究に従事。IEEE/CS 会員。

#### 伊藤 実(正会員)

1977 年, 1979 年, 1983 年にそれぞれ大阪大学基礎工学部卒業, 基礎工学研究科博士前期課程修了, 博士後期課程修了。1979 年より大阪大学基礎工学部助手。1986 年より大阪大学基礎工学部講師。1989 年より大阪大学基礎工学部助教授。1993 年 4 月より現在, 奈良先端科学技術大学院大学情報科学研究科教授。関係データベース理論, オブジェクト指向データベースのアプリケーション, DNA プローブ等の研究に従事。ACM, IEEE 各会員。