

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

丸山 敦史[†] 柴田 直樹^{††} 村田 佳洋[†]
安本 慶一[†] 伊藤 実[†]

本論文では、観光のためのパーソナルナビゲーションシステム “P-Tour” を提案する。P-Tour は、ユーザが出発地と出発時刻、帰着地と帰着時刻、複数の観光候補地と各地への立ち寄り希望度と時間制約（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満足度が高くなるような巡回経路（いくつかの観光地を含む）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。P-Tour は決定したスケジュールに従い、GPS 機能を備えた携帯端末を介し、ユーザにナビゲーション機能を提供する。提案するナビゲーション機能では、現在地を中心とする地図と次の目的地への経路の表示などの空間的な誘導に加え、各目的地での、滞在可能時間の表示や出発時刻の通知などの、時間的な誘導機能を提供する。遺伝的アルゴリズムを用いて準最適なスケジュールを高速に算出するアルゴリズムを設計・開発し、Java サブレットとして実装した。PC や携帯端末からウェブインタフェースを介してスケジュールの作成、ナビゲーション機能が利用できる。市販のカーナビゲーションシステム用のデジタル地図を用いた評価実験により、準最適なスケジュールを実用的時間で案内できることなどを確認した。

P-Tour: A Personal Navigation System with Travel Schedule Planning and Route Guidance Based on Schedule

ATSUSHI MARUYAMA,[†] NAOKI SHIBATA,[†] YOSHIHIRO MURATA,[†]
KEIICHI YASUMOTO[†] and MINORU ITO[†]

In this paper, we propose a personal navigation system for tourism called *P-Tour*. In *P-Tour*, when a tourist specifies the starting location, the departure time, the returning location, the arrival time and the multiple candidate destinations with relative importance and time restrictions on their arrival and staying time, the nearly best schedule is automatically computed. *P-Tour* can efficiently navigate the tourist according to the decided schedule through a portable computing device with GPS. In addition to the standard navigation function to guide users to destinations by displaying a graphical map, *P-Tour* provides temporal guidance for the tourist to follow the schedule. We have developed a route search engine to obtain a semi-optimal solution quickly using techniques of genetic algorithms. The engine has been developed as a Java Servlet and can be used from PCs and portable devices via http protocol. Our experimental results show that our route search engine can compute the nearly best schedule in reasonable time.

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普及により、携帯端末上にパーソナルナビゲーションシステムを実装し、利用する試みが行

われている。KDDI は、携帯電話を使ってカーナビのようにユーザを目的地に誘導する EZ ナビウォーク¹⁰⁾ というサービスをすでに開始している。

現在までパーソナルナビゲーションシステムについて多くの研究がなされている¹⁾⁻⁶⁾。ユーザのコンテキストに応じて動的に適切なルートを案内するシステム¹⁾ や、屋内施設でのルート案内²⁾、自転車による経路作成支援ツール³⁾ などが提案されている。これら既存のナビゲーションシステムでは、ある目的地への誘導、あるいは情報の提供を主要な目的としており、観光のように複数の目的地を限られた時間で効率良く訪

[†] 奈良先端科学技術大学院大学情報科学研究科
Graduate School of Information Science, Nara Institute
of Science and Technology

^{††} 滋賀大学経済学部

Faculty of Economics, Shiga University

れるという用途に使用するには機能が不足していると思われる。

観光のためのナビゲーションシステムに求められる機能は、カーナビゲーションのような現在地から単一目的地への最適経路の案内だけでは不十分で、複数の目的地を様々な制約のもとに効率良く観光するためのスケジュールの作成機能と、ユーザを作成されたスケジュールどおりに各観光地へ誘導するナビゲーション機能が必要となる。

本論文では、観光スケジュール作成機能およびナビゲーション機能の両方を備えるパーソナルナビゲーションシステム P-Tour を提案する。

一般に、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与えられた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、といった要求が存在する。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訪れる目的地の数を減らす必要がある。

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満足度が高くなるような巡回経路（いくつかの観光地を含む）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。

ナビゲーションシステムのための経路探索手法に関する研究では、不特定の複数地点を経由した経路を探索する経路を遺伝的アルゴリズムを用いて探索する手法が提案されている^{8),9)}。この不特定の目的地とは、コンビニエンスストアやガソリンスタンドなど、目的を果たせるならばそれらのうちのどれに寄ってもよいものを指す。しかし、これらの手法は、時間制約付きのスケジュールを求める問題に対して直接適用することができない。そのため、我々は、複数目的地を巡回する時間制約付きの観光スケジュールを算出するために、遺伝的アルゴリズムを用いた経路探索エンジンを開発した。

P-Tour は決定したスケジュールに従い、ユーザに GPS 機能を備えた携帯端末を介しナビゲーション機能を提供する。提案するナビゲーション機能では、現在地を中心とする地図と次の目的地への経路の表示などの空間的な誘導機能に加え、各目的地での滞在可能時間の表示や、出発時刻の通知などの時間的な誘導機能を提供する。また、なんらかの事情により、ユーザ

提案システムの構成概要

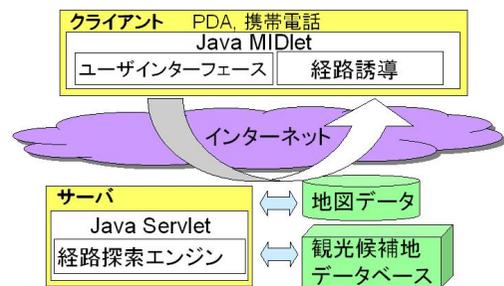


図 1 P-Tour の構成
Fig. 1 Organization of P-Tour.

が指定した経路と別の経路に進んだ場合や、交通渋滞などで予定のスピードで移動ができない場合、ある目的地での出発時刻を遅らせた場合など、ユーザのコンテキストの変化を自動検出し、帰着時刻などの制約を満たすよう残りのスケジュールを動的に修正する機能を提供する。P-Tour の以上の機能は、観光の用途以外に、宅配便の配送順序や、警備員の巡回経路の案内にも応用可能である。

P-Tour をクライアント・サーバシステムとして実装した。観光スケジュールの作成は地図データベースなどの比較的大容量のリソースを必要とし、また経路の算出にはある程度の計算量が必要なため、経路探索エンジンを WWW サーバ上のサーブレットとして実装し、携帯端末からインターネットを介して利用できるように設計した。

市販のカーナビゲーション用地図を用いて実験を行った結果、候補となる目的地を 30 力所指定した場合に、約 5 秒でスケジュールが作成可能なこと、目的地の追加・削除や制約の変更後の再計算は 1 秒未満で可能であること、また得られた解の最適解との満足度の差が 2% 未満であることを確認した。

2. P-Tour の概要

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、サーバから地図および推奨観光候補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 は、クライアントモジュールを通して、ユーザに、観光スケジュール作成機能とスケジュールに沿ったナビゲーション機能を提供する。

2.1 観光スケジュール作成機能

本機能の目的は、複数の観光候補地の中から、与え



図 2 P-Tour の実行画面
Fig. 2 Screen shots of P-Tour.

られた制約を満たし、かつ、最も満足度が高くなる目的地の集合とそれらを巡回する経路を求めることである。P-Tour では旅行者の制約として、総観光時間のほかに、目的地間の距離とユーザの移動速度、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、滞在希望時間などを考慮する。ユーザの移動速度は、交通手段に合わせて（たとえば自動車なら 30 km/h）指定される。

観光スケジュールを作成するため旅行者が入力する情報は、

- (1) 出発地点・時刻/帰着地点・時刻
- (2) 訪れたい観光地の候補の集合
- (3) 各観光候補地に対する重要度（0 以上の整数）
- (4) 各観光候補地に対する希望到着時刻に対する制約（「午後 2 時以前」など）
- (5) 各観光候補地での希望滞在時間に対する時間制約（「1 時間以上」など）

から構成される。ここで (3), (4), (5) は省略可能である。(3) においてユーザは重要度として、特に訪れたい候補地には 2 以上（5 程度）の値を、訪れたくない場合は 0 の値を入力する。省略した場合、1 の値が与えられる。この部分は、「特に訪れたい → 5 点/訪れてもよい → 1 点/訪れたくない → 0 点」という選択式のインタフェースも使用可能である。(4), (5) の項目を利用して施設の営業時間やイベントの開催時間などを指定することが可能である。

以上の入力から P-Tour は、(i) 観光する目的地の集合とそれらの巡回経路、(ii) 移動時間を考慮した各目的地への予想到着時刻と出発時刻、から構成される観光スケジュールを計算し、ユーザに提示する。

出力される経路は、ユーザが指定した各目的地への到着時刻と滞在時間に対する制約と帰着時刻までの制

限時間を満たし、目的地間の移動時間を考慮したうえで、ユーザの満足度が最も高くなる経路である（ユーザの満足度は 3 章で定義する）。ここで、目的地間の移動時間は、目的地間の地図上の最短経路（A*アルゴリズムにより探索）をユーザの移動速度で割って算出した時間を使用する。

ユーザは、観光候補地の追加・削除・変更を行ったり、各候補地について重要度の値や時間制約を変更したり、スケジュールを再計算するという作業を繰り返すことによって、納得のいくスケジュールを求めることができる。

しかし、上記の観光スケジュールを計算する問題は、組合せ最適化問題であり、最適解を求めるには一般に多大な計算時間を要する。インタラクティブな観光スケジュール作成支援のためには、数秒～数十秒程度で計算できることが必須である。そのため、遺伝的アルゴリズム（GA）を用いて準最適解を高速に求める経路探索エンジンを設計・開発した。これについての詳細は 3 章で述べる。

2.2 スケジュールに沿ったナビゲーション

P-Tour は決定したスケジュールに従い、ユーザに GPS 機能を備えた携帯端末を介してナビゲーション機能を提供する。

ナビゲーション機能は「スケジュール表示モード」、
「移動モード」および「滞在モード」の 3 種類のモードを持つ。スケジュール表示モードでは、現在の観光スケジュールにおける、全体経路の表示および進行状況、以降の各目的地への到着予定時刻と滞在時間を図 2 (a), (b) に示すように携帯端末の画面に表示する。スケジュールの進行状況は、図 2 (a) の画面において次の目的地をハイライト表示するとともに、図 2 (b) の画面において次の目的地以降のスケジュールを表示

することで示す．

移動モードでは、カーナビのように、現在地を中心とする地図と次の目的地への経路の表示を行う(図2(c))．滞在モードでは、図2(d)に示すように、現在地での残り滞在可能時間と出発時刻の通知、施設に関する情報などを表示する．なお、移動モードと滞在モード間の切替えは、GPSによる位置情報および現在時刻から自動的に行われる．また、いずれのモードからでもスケジュール表示モードへの切替えは任意のタイミングで実行できる．

2.2.1 コンテキストの監視とスケジュールの再計算

P-Tourでは、GPSによる位置情報、現在時刻から、ユーザがなんらかの事情により、指定した経路と別の経路に進んだ場合や、交通渋滞などで予定のスピードで移動ができない場合、あるいは、ある目的地での出発時刻を遅らせた場合を自動的に検出し、帰着時刻などの制約を満たすよう残りのスケジュールを動的に修正する機能を提供する．スケジュールの修正が行われたときには、そのことを示すアラートを画面に表示し、修正後のスケジュールを確認することができる．

ユーザが修正後のスケジュールが気に入らない場合には、スケジュール作成機能を用いて、観光候補地の変更や、重要度・時間制約の変更を行いスケジュールを再計算することも可能である．

3. 経路探索エンジン

前章で述べたような制約を満たす経路を探索するために、遺伝的アルゴリズム(GA)を利用する．その理由は、GAは探索過程においてつねに解候補を保持しているため、実用的な時間に合わせた探索が可能になるためである．また、遺伝的アルゴリズムは、探索空間の集団内に複数の近似解を持っている．そのため、希望を変更した場合に、その希望を満たす、あるいは、希望に近い近似解が集団内に存在する可能性が高く、集団を再利用することで、高速に再探索を行うことが可能となる．

探索時間を短縮するために、任意の目的地間の距離およびユーザの移動速度が登録されたデータベースを用意しておく．この速度の値は、「自動車の場合は30km/h」「徒歩なら4km/h」など、経験的に得られた値を用いる．もし、データベースに登録されていない目的地がユーザにより入力された場合は、そのつど目的地間の最短経路をA*アルゴリズム⁷⁾を利用して求める．

3.1 問題定義

この問題の目的は、より高いユーザ満足度(評価値)

を持つルートを求めることである．評価値は、希望した目的地が経路に含まれ、また指定された時間制約を満たした場合に増加していくと考えるのが自然である．

地図データは、有向グラフ $G = (V, E)$ で与えられ、目的地集合を $D = \{d_1, d_2, \dots, d_n\}$ と表記する．各目的地データ d_i は、次に示す4つの項、 $v_i \in V$, rt_i , dur_i , そして、 $pre: D \rightarrow N$ からなる．ここで、 rt_i は到着時刻に対する制約(例: “12:00以前”), dur_i は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), $pre(d_i)$ は目的地 d_i の重要度を意味する．

他に、出発/帰着地点 $p_s, p_g \in D$ は、時間制約とともに入力必須項目であり、距離関数 $dist(d_s, d_g)$ は、データベースに登録されている目的地 d_s と d_g の間の距離を返す関数である．データベースに含まれていない場合は、グラフ G より、A*アルゴリズムを用いて計算する．

解候補となる経路を、 $S = \langle d'_1, d'_2, \dots, d'_k \rangle$ で表す．ここで、 $d'_k \in D$, $1 \leq k \leq |D|$, そして、 d'_k は k 番目に回る目的地を表す．

解候補は目的地の順列で表現されるために、同じ部分経路(道路など)を複数回通過する経路を表現することができる．

移動時間は、移動距離から計算する．ユーザが目的地 d'_i, d'_{i+1} 間を移動する速度を $speed(d'_i, d'_{i+1})$ で参照でき、巡回する目的地 d'_i には $stay_i$ 分滞在するとする($stay_i$ は dur_i を満たす)． d'_i への到着予想時刻を t_i とすると、 $t_{i+1} = t_i + stay_i + \frac{dist(d'_i, d'_{i+1})}{speed(d'_i, d'_{i+1})}$ で表せる．

時間制約を満たすことができた目的地の集合は、 $S' = \langle d''_1, d''_2, \dots, d''_j \rangle$ で表される．ここで、 $d''_j \in S$ であり、 d''_j は時間制約が指定され、かつ時間制約を満たした目的地である．また、 j は時間制約を満たすことのできた目的地の数である．ただし、 D の中で、時間制約を指定した目的地の数を m とすると、 $0 \leq j \leq m$ である．

3.2 評価関数

それぞれの解候補経路を評価するために、巡回経路に含まれた目的地と、時間制約を満たした目的地に指定されている重要度の合計を計算する．しかし、重要度を合計するのみの評価だと、遠回りする経路が高い評価値を持つ解として得られるかもしれない．そこで、遠回りを防ぎ、効率の良い経路を求めるための評価値の計算式を以下のように定義した．

$$f(S) = \alpha \sum_{i=1}^k pre(d'_i) + \beta \sum_{i=1}^j pre(d''_i) - \gamma \sum_{i=1}^{k-1} dist(d'_i, d'_{i+1})$$

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（時間制約を満たした目的地を含んでいる）。 β は時間制約を満たした目的地の重要度の和に対する重みである。 γ 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、評価値は悪くなる。

この評価関数では、時間制約を満たしていない目的地が経路に含まれても、重要度を加算している。この理由は、探索途中での有望な解候補の死滅を防ぐためである。ここでは、探索の最後に、時間制約を満たしていない目的地が含まれている経路がないかをチェックし、あった場合には削除することで対処する（3.3節のアルゴリズムのステップ(9)参照）。

3.3 アルゴリズム

探索においては、指定された目的地の集合から制限時間内に巡回可能な数の目的地の部分集合に絞りをこまなければならない。目的地の追加/削除を実現するために、解候補を目的地の可変長リスト構造としてコーディングした。探索の手順は次のようになる。

- (1) 指定した数の解候補（個体群）を生成する。各個体は、目的地集合に含まれる目的地のリストである。生成後、それぞれの個体の評価値を、前述の計算式で計算する。
- (2) 個体群の中からエリート個体を選択する。エリート個体は、個体群の中で最も良い評価値を持った個体である。
- (3) トーナメント選択を行う。トーナメント選択では、個体群の中からランダムに2個体を選びだし、評価値を比べ、優れた評価値を持った個体を次世代に残す。これを、必要個体数に達するまで行い、次世代の個体群を生成する。
- (4) 一点交差を行う。一点交差では、ランダムに2個体を選び、ランダムに選んだ交差点により交差を行い、新たにできた個体の中で重複した目的地の削除を行う。この操作により、目的地の追加と削除が行われる。
- (5) 突然変異を行う。本アルゴリズムで行う突然変異は、2-opt法と目的地の追加である。2-opt法は、染色体上の2点をランダムに選び、交換する操作である。目的地の追加は、染色体に含まれていない目的地を、染色体のランダムな位置

に挿入する操作である。

- (6) 選択、交差、突然変異によって新たに生成された個体群の各個体について、評価値を計算し、古い個体群と置き換える。このとき、エリート個体は保存しておく。
- (7) 選ばれたエリート個体に対して局所探索を行う。ここでは、エリート個体の染色体上のランダムな位置に、染色体に含まれていない目的地の中からランダムに選んだものを挿入する。挿入後に評価値を計算し、評価値が挿入前よりも改善されたら染色体を更新する。改善されなければ挿入前に戻す。
- (8) ステップ(2)から(7)までを1回繰り返すことを1世代と数える。これを、指定した世代数繰り返す。繰り返した後は(9)へ。
- (9) 時間制約を満たしていない目的地を含む個体を削除したうえで、最も良い評価値を持った個体の示す経路を得られた解とする。

4. P-Tour の実装

P-Tour を、クライアント・サーバアプリケーションとして実装した。また、前章で述べた経路探索エンジンは、インターネット上のサーバPCで実行されるJava Servletアプリケーションとして、クライアントモジュールは、携帯端末上にJava MIDletとして実装した。

4.1 ナビゲーション機能

ユーザは、観光スケジュールを立てるために、携帯端末からサーバに、観光候補目的地などを指定したHTTPリクエストを送信する。ユーザのリクエストをもとに、サーバは観光スケジュールを生成し、次のようなHTTPレスポンスをユーザに返す。

- (1) 観光スケジュールの全体を示した地図画像（図2(a)）
- (2) 訪れる目的地の巡回順序と、予想到着/出発/滞在時間（図2(b)）

観光中には、図2(c)のように、ユーザの現在位置を示した地図画像を携帯端末上のディスプレイに表示する。クライアントモジュールは、携帯電話に搭載したGPSからユーザの現在位置を測位し、地図の更新をサーバモジュールに要求する。サーバモジュールは、最新の地図画像（JPEG画像）を生成し、レスポンスとして送信する。地図の更新は、ユーザが目的地間を移動している間、定期的に繰り返し行われる。この更新間隔は、画面サイズやユーザの移動速度、通信コストを考慮して決定される。



図 3 携帯電話用のユーザインタフェース
Fig.3 User interface for mobile phones.

4.2 ユーザインタフェース

携帯電話向けのユーザインタフェースを Java MIDlet として図 3 のように実装した。

このユーザインタフェースを通して、ユーザは次の 2 ステップの入力操作を行う。

- (1) ユーザは、システムが用意した目的地リストの中から、訪れたい目的地を選択する(図 3(a)(b))
- (2) ユーザは、図 3(b)(c) が示すように、それぞれの目的地に対して、重要度と、時間制約を設定する。これらを指定しない場合は、システムのデフォルト値が各目的地に対して設定される。

4.3 コンテキスト変化の検出

ユーザの移動の遅れなどの理由で、ユーザが次の目的地に観光スケジュールの予定時刻どおりに到着できないことを検出した場合、クライアントモジュールは、ユーザに対して警告を行う。この警告では、ユーザにスケジュールで予定していた経路に戻るための経路を提示するか、観光スケジュールを修正してユーザに提示する。スケジュールの修正は、ユーザの現在位置、スケジュール上の未巡回の目的地、をサーバにリクエストとして送信し、スケジュールの再計算を行う。

警告と再計算の具体的な条件は以下のとおりである。

- 何らかの理由により、予定した経路以外の道へ進んだ場合：

ユーザの現在位置から次の目的地までの最短経路の距離をユーザの移動想定速度で割った値を $T_{stopover}$ とする。サーバは、クライアントモジュールの地図の更新時に、 $T_{stopover}$ の値を計算する。たとえば、クライアントモジュールにおいて、現在時刻 t から $T_{stopover}$ 経過した時刻が、次の目的地へのスケジュール上の到着予想時刻を 3 分以上超えると警告し、以下で定義する $GUARD_TIME$ を超えると、再計算を行うように設定する。
- 渋滞などの理由により、次の目的地に、予定の時刻までに到着できないことを検出した場合、もし

くは目的地を出発する時刻が予定より遅かった場合：このような状況を想定し、各目的地にユーザが滞在する時間には、あらかじめ到着の遅れに対する余裕 $GUARD_TIME$ を見込んであると仮定している(すなわち、 $stay_i - GUARD_TIME$ が dur_i を満たす)。 T_{prdarv} を次の目的地への最新の予想到着時刻とする。これは、ユーザの現在位置と次の目的地の間の距離、現在時刻、ユーザの移動速度から計算される。これもサーバで計算され、地図の更新時にクライアントモジュールに通知される。次の目的地への T_{prdarv} が観光スケジュールの予定到着時刻を超える場合、クライアントモジュールは、ユーザに警告を行う。そして、次の目的地への T_{prdarv} が観光スケジュールの予定到着時刻+ $GUARD_TIME$ を超えたときには、クライアントモジュールは、スケジュールの再計算をリクエストする。

5. 評価実験

P-Tour で用いている経路探索エンジンの性能について、(1) 得られたスケジュールの妥当性、(2) スケジュールの計算時間、(3) 得られたスケジュールの最適性、について実験・評価を行った。

実験には、出発地点・帰着地点を奈良先端科学技術大学院大学 (NAIST)、出発/帰着時刻を 9:00 / 21:00 に設定し、奈良県北部の主な名所 30 カ所を対象に観光するという典型的なシナリオを用いた。なお、これらの観光地をすべて観光するのは時間的に無理である。

実験では、カーナビゲーションシステムに利用されている市販のデジタル地図¹¹⁾(奈良県北部、ノード数 3,800)を用いて、一般的な PC (Pentium4 2.4 GHz, 512 MByte Memory, GNU/Linux, Tomcat4.1) 上で経路探索エンジンを実行した。なお、ユーザの移動速度は一律 30 km/h とした。経路探索エンジンにおける評価関数の係数は、予備実験から $\alpha = 20$, $\beta = 200$, $\gamma = 40$ を用いた。また、探索は、評価値がほぼ一定の値に収束する 100 世代まで行った。

5.1 得られるスケジュールの妥当性の評価

まず、ユーザの入力に従って、好みの経路が得られるかどうかについて評価を行った。

表 1 に示すように、30 個の目的地を観光候補地として指定し、13 目的地 $\{D1, \dots, D13\}$ の重要度を 5 に設定し、残りの 17 目的地 $\{D14, \dots, D30\}$ は 1 に設定した。到着希望時刻として、D3 地点には 15:00 以前、D7 地点には 19:30 以前を設定した。滞在時間は、D3 地点には 180 分以上、D7 地点には 60 分以上

に設定した .

上記の設定で経路を探索したところ、図 4 (a) に示すような経路が得られた . この経路は、8 つの目的地 {D1, D2, D3, D7, D8, D9, D10, D12} を含んでいる . この経路に含まれない候補地 D4, D13 について、重要度を 10 に変更したところ、図 4 (b) に示すように、D4, D13 を含む新たな経路が計算できた .

5.1.1 評価関数の妥当性

本経路探索エンジンは、ユーザの満足度を数値化するため、3.2 節で定義した評価関数を用いている . 評価関数が妥当であるかどうか調べるため、評価関数における総移動距離に関する係数 γ を 0, 40, 80 と変化させて、得られる経路の特性を調べた . γ の値が大きいほど、総移動距離が小さい経路を算出する . 探索結果を図 5, 表 2 に示す . 図中の数字は、巡回順序を示している .

表 1 ユーザ入力
Table 1 User input.

目的地 No.	名称	重要度	希望到着時刻	滞在時刻
D1	唐招提寺	5	-	=60 分
D2	薬師寺	5	-	=60 分
D3	法隆寺	5	≤ 15:00	≥ 180 分
D4	藤ノ木古墳	5	-	=90 分
D5	平城京	5	-	=150 分
D6	西大寺	5	-	=30 分
D7	学園前駅	5	≤ 19:30	≥ 60 分
D8	猿沢池	5	-	=60 分
D9	万葉植物園	5	-	=45 分
D10	国立博物館	5	-	=60 分
D11	新薬師寺	5	-	=60 分
D12	正倉院	5	-	=30 分
D13	大安寺	5	-	=60 分
D14	法輪寺	1	-	=60 分
...
D30	法華寺	1	-	=30 分

$\gamma = 0$ の経路は、時間制約 D3, D7 を満たしている . しかし、1 番目から 4 番目までの目的地の巡回経路は、明らかに無駄が多いことが分かる (図 5 (a)). これは、 $\gamma = 0$ であり、移動距離が評価値に影響を与えないためであると考えられる .

次に、 $\gamma = 40$ の経路も時間制約 D3, D7 を満たしているが、 $\gamma = 0$ に比べて自然な巡回経路になっており、良い経路であると考えられる (図 5 (b)). この結果は、 γ の値を大きくしたため、総移動距離が評価値に与える影響が大きくなり、無駄な移動をしなくなったためと考えられる .

最後に $\gamma = 80$ の経路を見ると出発/帰着地点から最

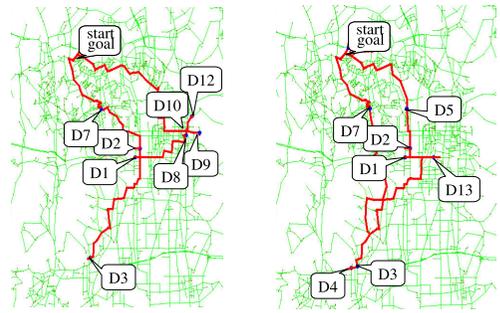


図 4 計算された経路
Fig. 4 Computed routes.

表 2 γ の値による経路の性質
Table 2 Properties of routes depending on γ value.

γ	目的地数	総移動距離	D3 への到着時刻	D7 への到着時刻
0	8	55 km	14:40	19:10
40	8	45 km	14:50	18:50
80	7	23 km	N/A	18:40

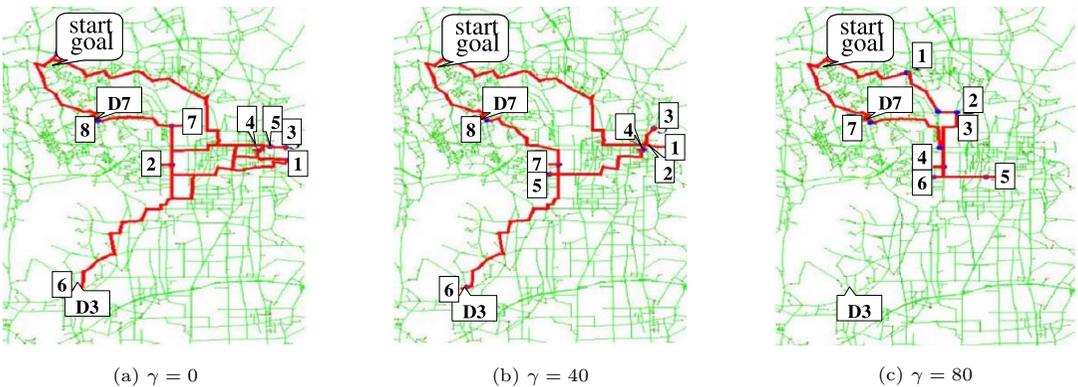


図 5 γ 値による経路の妥当性

Fig. 5 Appropriateness of routes depending on γ value.

も遠い場所である D3 地点を含んでいない(図 5(c)). 総移動距離が評価値に与える重みが大きくなり、時間制約を満たすことよりも、移動距離の制約の方が大きくなった結果、目的地 D3 を回らなくなったと考えられる。また、 $\gamma = 0$ 、 $\gamma = 40$ と比べると、巡回可能な目的地のほとんどが、出発地点から近い位置の目的地となっている。

この実験の結果、 $\gamma = 40$ を使用したときに、適切な経路が計算できていると考える。また、 γ の値をユーザが与えることにより、よりユーザの好みにあった経路を計算することも可能である。 γ と同様に、 α 、 β の適した値も経験的に求めた。 α と β の値は対となっており、 β の値を大きくすると、巡回可能な目的地数を減らして、特定の目的地の時間制約を満たそうとし、逆に α の値を大きくすると、時間制約が指定された目的地を巡回経路から削除する傾向があった。予備実験の結果から、 $\alpha = 20$ 、 $\beta = 200$ の値がそれぞれ適切であるとし採用した。

5.2 スケジュール作成に必要な時間

遺伝的アルゴリズムは、探索の過程でつねに解を保持しているため、探索の途中、すなわち、計算時間を限定して解を出すことが可能である。そこで、表 1 の入力を用いて探索を行い、探索を行う世代数を変え、経路の計算時間と品質の評価を行った。結果を表 3 に示す。なお、観光候補地間の経路と移動距離は、すべてあらかじめ計算されデータベースに登録されているとする。

表 3 に示すように、評価値は探索世代を経るごとに良くなり、50 世代で収束している。また、10、20 世代と 50 世代の経路に大差はなく、10/20 世代でも十分な経路が得られている。探索時間では、50 世代でも 11 秒と十分実用的であると考えられるが、10/20 世代でも十分質の良い経路が得られている。これらの結果から、探索時間が限られたとき(たとえば 5 秒以内)でも、十分に良質な経路を計算することが可能であることが確認できた。

5.2.1 スケジュールの再計算時間

気に入ったスケジュールを見つけた際には、設定を少しずつ変更しスケジュールを再計算するという作業が必要となる。そこで、表 1 の入力に対して探索した経路中の目的地を 1 つ (D3) 削除し、別の目的地 (D5) に対し新たに時間制約を追加するよう入力を変更し、経路の再探索を行った。探索は、変更前の入力に対し行った探索の後の個体群(解候補)を再利用した。

結果、探索に要した時間は 1 秒未満であった。これは、最初の探索の後の個体群に、準最適解が存在し、

表 3 探索世代数による計算時間の変化

Table 3 Computation time depending on number of generations.

世代数	計算時間	評価値
5	2.6 秒	2322
10	3.0 秒	2531
20	5.2 秒	2542
50	11.0 秒	2582
100	15.5 秒	2582

表 4 最適値との比較

Table 4 Differences from optimal values.

目的地数	評価値	最適値	誤差 (%)	計算時間 (P-Tour)	計算時間 (opt)
12	2650	2670	0.8	15.9 秒	49 分
13	2664	2698	1.3	16.3 秒	126 分
14	2708	2744	1.3	15.0 秒	22 時間

同所探索を行うだけで目的の解が得られたためと考えられる。

5.3 最適解との比較

経路を、本経路探索エンジンで求める場合と、総当たり探索によって最適値を求める場合とで、計算時間と評価値を比較した。観光候補地の数を 12、13、14 と変化させ、評価関数の係数を $\gamma = 40$ に固定し、探索世代数として 100 を使用した。結果を表 4 に示す。本経路探索エンジンによる評価値は、10 回試行した結果の平均値である。12、13 目的地ともに、最適値から約 1% の誤差の評価値が得られており、解の質では実用上十分であると考えられる。また、最適解の探索には、14 目的地で 22 時間かかるが、本経路探索エンジンでは 30 目的地を 3 から 6 秒程で準最適解を導き出すことが可能であり、本経路探索エンジンの有効性が示された。本システムで用いる問題は NP-hard のクラスに属するため、最適解を求めるためのコストは問題の規模に対して指数関数的に増加する(表 4 参照)。計算時間のために、15 目的地以上の場合について最適解を求めることはできなかった。

6. おわりに

本論文では、観光のためのパーソナルナビゲーションシステム P-Tour を提案した。P-Tour は、(1) ユーザの好みに合うような、複数の目的地を効率良く巡回するスケジュールを、重要度と時間制約を基に計算する機能、(2) GPS を搭載した携帯端末を通して、ユーザをスケジュールに従って案内する機能、(3) ユーザの状況の変化により、スケジュールどおりに観光が行えないことを検知すると、スケジュールを自動的に修正する機能を提供する。

スケジュール作成機能は、遺伝的アルゴリズムを利用した経路探索エンジンとして設計・実装した。経路探索エンジンを市販のデジタル地図を用いて評価実験を行った結果、一般的な希望に対するスケジュールを5秒ほどで計算できることを確認した。また、ユーザの希望の一部を変更して再探索を行った結果、1秒以内で新たなスケジュールを計算できることを確認した。これらの結果は、十分実用的であると考えている。

P-Tour を、クライアント/サーバシステムとして設計し、サーバモジュールは、WWW サーバ上の Java Servlet として実装した。したがって、web ブラウザを利用すれば、PDA や携帯電話などの様々な携帯端末から利用することが可能である。経路案内機能は、GPS を搭載し、かつインターネット接続可能な携帯端末から利用可能であり、現在、携帯電話、PDA への実装を進めている。

現在の P-Tour の実装は、単一の移動手段（自動車、あるいは徒歩）のみを想定している。今後は、電車やバスなどの交通手段を組み合わせるよう拡張していく予定である。

実験では、観光候補地の数が 30 までの場合を扱ったが、より多くの候補地を扱った場合でも、遺伝的アルゴリズムの世代数をたとえば 50 あるいは 100 世代までに抑えることによって、同様の時間で解を求めることが可能である。しかし、現在の実装では、初期個体は必ず観光候補地と同じだけの遺伝子長を持たせているため、多くの観光候補地を扱うほど、初期個体と準最適解の遺伝子長の差が大きくなり、結果として、解が収束するまでにはより多くの世代数が必要となる。今後、これらの問題を解決した、よりスケラビリティに優れたアルゴリズムを開発する予定である。また、作成したシステムを用いて実験環境での運用実験を行い、ユーザ満足度モデルの改良や、ユーザインタフェースの改良などを今後行っていく予定である。

最後に、地図フォーマットを開示していただいた IT ナビゲーションシステム研究会に感謝する。

参 考 文 献

- 1) Baus, J., Krüger, A. and Wahlster, W.: A Resource Adaptive Mobile Navigation System, *Proc. IUI2002: International Conference on Intelligent User Interfaces 2002*, ACM Press, New York (2002).
- 2) 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).

- 3) Ehlers, M., Jung, S. and Stroemer, K.: Design and Implementation of a Gis Based Bicycle Routing System for the World wide Web (WWW), *International Symposium on Geospatial Theory, Processing and Applications (IS-PRS2002)* (2002).
- 4) Malaka, R. and Zipf, A.: DEEP MAP — Challenging IT research in the framework of a tourist information system, *ENTER 2000*, Barcelona, Spain, pp.15-27 (2000).
- 5) IST: CRUMPET Creation of User-friendly Mobile services Personalized for Tourism, <http://www.ist-crumpet.org/>
- 6) Zipf, A. and Malaka, R.: Developing Location Based Services (LBS) for tourism — The service providers view, *Information and Communication Technologies in Tourism 2001, Proc. ENTER 2001, 8th International Conference*, Montreal, pp.83-92, Springer Computer Science, Wien, NewYork (2001).
- 7) Korf, R.: Real-Time Heuristic Search, *Artificial Intelligence*, Vol.42, No.2-3, pp.189-211 (1990).
- 8) 稲垣, 長谷山, 北嶋: 遺伝的アルゴリズムを用いた複数経由点をともなう経路探索法, 電子情報通信学会論文誌 D-I, Vol.J83-D-I, No.5, pp.504-507 (2000).
- 9) 狩野: 遺伝的アルゴリズムを用いたカーナビのための経路案内方式, 情報処理学会研究報告, Vol.2002, No.21, pp.51-58 (2002).
- 10) au by KDDI: EZ ナビウォーク .http://www.au.kddi.com/ezweb/au_dakara/ez_naviwalk/
- 11) IT ナビゲーションシステム研究会: ナビゲーションシステム研究会 Ver2.2-S 規格フォーマット編, 第 1 版 (1999).

(平成 16 年 4 月 1 日受付)

(平成 16 年 10 月 4 日採録)



丸山 敦史 (正会員)

2003 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, 2003 年ザナヴィ・インフォマティクス入社。カーナビゲーションシステムの開発に従事。



柴田 直樹 (正会員)

2001年大阪大学大学院基礎工学研究科情報数理系専攻博士後期課程修了。2001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、2003年より滋賀大学経済学部助教授。エージェント技術、遺伝的アルゴリズム、形式的設計検証等の研究に従事。



村田 佳洋 (正会員)

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。



安本 慶一 (正会員)

1991年大阪大学基礎工学部情報工学科卒業。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年モンテ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、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。