

満足度と体力を考慮した観光スケジューリング法の提案

武 兵^{†1} 孫 為 華^{†1} 村 田 佳 洋^{†2}
安 本 慶 一^{†1} 伊 藤 実^{†1}

観光においては、ユーザの好む観光地をより多くまわるようなスケジュールを立案することが望ましい。しかし同じ観光地であっても、観光方式や観光時間によって必要な体力が異なり、ユーザの体力が足りなくなってしまう場合がある。本稿では、観光中に休憩をうまく挟むことで体力の範囲内で最も満足度が高くなる観光スケジュールを求める問題を取り扱う。本問題は NP 困難であり、実行時間で最適解を算出することは困難である。実行時間で準最適解を得るため、捕食法に基づいて、複数の観光地を巡回する休憩なしのスケジュールを求め、局所探索として休憩を差し挟むことで解を探索する手法を提案する。提案手法を評価するため、10 か所の観光地を持つインスタンスを用いてシミュレーション実験を行った。その結果最適解の 95.65% の満足度を持つスケジュールを 13 秒で得ることができた。

Stamina-Aware Sightseeing Tour Scheduling Method

BING WU,^{†1} WEIHUA SUN,^{†1} YOSHIHIRO MURATA,^{†2}
KEIICHI YASUMOTO^{†1} and MINORU ITO^{†1}

Tour schedules are wanted to include multiple sightseeing spots taking into account the user's preference, but the stamina of tourists may be insufficient when doing sightseeing. In this paper, we propose the sightseeing scheduling problem to maximize satisfaction. In this problem, break times are allocated in schedule to hold constraint of stamina. This problem is NP-hard, then it is difficult to solve it in practical time. In order to calculate semi-optimal solution in practical time, we propose the method to plan schedule visiting multiple sightseeing spots with no break times and to allocate the break times among schedule using local search technique. To evaluate the proposed method, we compare our method with the Brute force method through the simulation experiment for the instance with 10 sightseeing spots. As a result, the proposed method composed the schedule whose expected satisfaction is 95.65% of the optimum solution in 13 sec.

1. はじめに

近年、ユーザに対し目的地への経路案内や周辺情報の提供を行う様々なパーソナルナビゲーションシステムが提案されている。パーソナルナビゲーションシステムの主な用途として観光への利用が挙げられる。エンタテインメントを目的としたものに、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^{1),2)}や屋内施設でのルート案内³⁾、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⁴⁾がある。また、観光中のユーザ支援を目的に、多様な観光計画や、交通機関乗り換えに関する情報の提供、屋内外の案内のための統合アプリケーションが提案されている⁵⁾。観光では、通常、いくつかの目的地を巡回する。複数の目的地を巡回するような経路を求める問題として巡回セールスマン問題がある。この問題を効率良く解く近似アルゴリズムとして、Lin-Kernighan 法⁶⁾、焼きなまし法 (Simulated Annealing) など、厳密に最適解を求めないことで計算時間を短くする方法がある。より複雑な定義の問題を扱う解法としては、遺伝的アルゴリズム⁷⁾、捕食法⁸⁾⁻¹⁰⁾、大規模局所探索法¹¹⁾といった厳密解を保証しないヒューリスティックアルゴリズムを用いることも多い。しかし、これらは基本的に現在位置から次の目的地を案内するものであり、観光者の体力を考慮して、嗜好に合ったスケジュールを立案する機能を持たない。

我々の研究グループは、観光のためのパーソナルナビゲーションシステム P-Tour¹²⁾を提案している。P-Tour は、ユーザの希望に応じた複数目的地の巡回スケジュールを立案し、また各目的地における案内機能を提供する。また、P-Tour 上に天気変化を考慮した観光案内機能¹³⁾も追加するなど、現実環境に適用する拡張を行っている。しかし、このシステムは、観光者の体力と観光方式に応じて目的地の満足度が変化する問題を取り扱っていない。また観光旅行においては、不慣れな食べ物や水、異なる環境など様々な要因があるため、体力が低下すると病気になりやすい。したがって、観光者の体力にあわせた無理のないスケジュールを作成することが必要である。観光者の体力を考慮した観光方式としては、登山ガイドにおけるコースごとの必要体力の明示や、旅行会社による対象年齢別の旅行プランの推奨などがある。ケーブルカーで登山する場合、体力の消耗度が少なくなるが、得られる満

^{†1} 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

^{†2} 広島市立大学大学院 情報科学研究科

Graduate School of Information Sciences, Hiroshima City University

足度も歩行登山の場合と異なる．体力を考慮した場合，どこでいつ休憩を入れるかのスケジューリングが重要となる．体力と時間の制約条件の下で適宜に休憩を入れ，満足度を最大化するような観光スケジュールは複雑な最適化問題である．

本稿では，観光者の体力と時間を制約とし，適宜に休憩を取りながら，複数の観光地に対し，好きな方式で巡るスケジュールの立案手法を提案する．本手法では，観光地を異なる方式で観光する場合の体力消費と得られる満足度をあらかじめ定義し，スケジュール全体で得られる満足度をできるだけ大きくすることを目的とする．本問題は NP 困難であり，実用時間内に最適解を求めることは難しい．複数の観光地を巡回する最も満足度の高いスケジュールを算出するためには，観光中にうまく休憩をはさむことで，体力残がマイナスにならないようにすることが有効である．提案手法では，このことに着目し，捕食法 (predatory search)^{8)–10)} を用いて，休憩なしのスケジュールをまず求め，休憩を差し挟む回数，タイミングを近傍探索で求めるというアプローチを採用する．提案手法により得られた解の最適性を評価するため，10 か所の観光地を持つインスタンスに対し提案手法と全探索で求めた解と比較した．その結果，最適解の 95.65%以上の満足度を持つ解を平均 12.99 秒で得ることができた．また，欲張り法と比較したところ，20 か所の観光地を持つインスタンスにおいて 164.29%以上高い満足度を持つ解を得ることが出来た．

2. 体力を考慮した最大満足度観光スケジュールリング問題

本章では，体力を考慮した最大満足度の観光スケジュールを算出する問題を定義する．

2.1 観光スケジュールの属性

本研究では，観光スケジュールを立案するために，観光者と観光地の属性を取り扱う．表 1 に対象とする観光者と観光地の属性を示す．

- 観光者は限られた時間（観光時間）と体力（体力上限）で，一つ以上の観光地を巡り，できるだけ高い満足度を得たいとする．観光活動を行えば，体力が少なくなる．使用できる体力を体力残で表す．休憩する場合，回復率と休憩時間に応じて体力残が増える．ただし，体力残が体力上限を超えることはできない．また，観光者は嗜好という属性を持ち，嗜好により，同じ観光地，同じ観光方式でも得られる満足度が異なる．
- 観光地は名前のほか，位置が与えられている．異なる観光地間で移動する場合，所要移動時間は観光地間の距離によって決定される．また，各観光地において選択可能な観光方式が複数あるとし，観光者の嗜好により，得られる満足度が異なる．なお，それぞれの観光方式の所要体力と時間も異なる場合がある．それぞれの方式に対し，所要体力と

時間，得られる満足度が与えられているとする

表 1 対象とする観光者と観光地の属性

対象	属性	叙述
観光者	観光時間	最大観光時間
	体力上限	上限である体力
	体力残	現在の残り体力
	回復率	休憩時分単位の体力回復値
観光地	名前	観光地名
	位置	観光地の道路地図データ
	観光方式	観光地で選択し得る観光方式
	消費体力	観光方式に対する消費体力値
	満足度	観光方式に対する満足度

表 2 観光地に対する観光方式の一例

観光地	観光地 A			観光地 B		観光地 C	
観光方式	A1	A2	A3	B1	B2	C1	C2
消耗体力値	110	200	340	500	700	120	90
滞在時間 (hour)	1	1.5	2	1.5	2	1.5	1
満足度	40	60	75	60	80	90	70

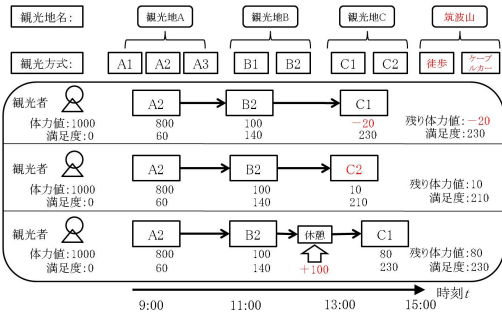


図 1 スケジュールの例

表 2 に観光地および観光方式の例を挙げる．この例のように，同じ観光地でも観光方式が違えば，満足度及び観光者の体力消耗の度合いが変化する．本問題では，観光者の体力に合

わせて満足度が最大になる観光スケジュールを求めることを目的とする．図 1 はスケジュール例である．出発時間を 9 時，帰着時間を 15 時とする．観光者に対して一時的に算出された観光スケジュールが $A \rightarrow B \rightarrow C$ であったとする．観光者が観光地 C の観光方式として $C1$ （体力消耗値が 70）を選択すると，体力値が負の値になるため，このようなスケジュールは実行不可能である．次に，観光地 C の観光方式を $C2$ （体力消耗値が 90）に変更すると，体力内での観光が可能になるが，満足度が低くなる．一方，観光地 B と観光地 C の間に休憩を入れると，体力値が回復するため，体力内でかつ満足度の高い観光が可能になる．この場合に得られる結果として，スケジュールの満足度が 230 となる．本問題の目的は，観光者の体力に関する制約条件を満たしつつ，スケジュールの満足度を最大化することである．また，問題を簡単化するために，「ベンチに座る」など場所にかかわらず行える休憩のみを想定する．休憩が必要な場合，観光後に同じ場所で休憩を行うよう設定する．図 2 は，提案手法で求めた休憩を含むスケジュールの一例である．体力残が 0 以下にならないように，適宜休憩が入っていることが分かる．満足度を最大化するようなスケジュールを求めることが目的である．

出発時間	目的地	観光方式	満足度	体力残	到着時間	帰着時間
9:00	Hotel	By Car	0.0	900.0	9:00	
__9:09	Spot10	method1	106.327	644.0	10:39	
__10:50	Spot1	method2	97.432	303.0	11:50	
__11:50	Spot1	have rest	0.0	343.0	11:54	
__12:02	Spot9	method1	94.134	2.0	13:02	
__13:02	Spot9	have rest	0.0	72.0	13:09	
__13:17	Spot3	method2	90.714	4.0	15:17	
__15:17	Spot3	have rest	0.0	214.0	15:38	
__15:52	Spot8	method1	105.317	9.0	16:52	16:56
Satisfaction of Schedule : 493.924						

図 2 体力と満足度を考慮した観光スケジュールの例

2.2 問題の定式化

データベース入力として，道路地図および観光地，交通手段が与えられる．

- $G = (V, E)$: 道路網を示す有向グラフの道路地図データである．各エッジは以下の重みを持つ．
 - $dist(v_1, v_2)$: 任意の頂点 v_1, v_2 の間の距離を返す関数を表す．

- V : 観光候補地の集合（例：法隆寺，奈良公園など）を表す．
- Q : 観光地の各観光方式の集合を表す．
- P : 移動方式の集合を表す．

観光スケジュールは，目的地と移動方式で構成され，スケジュール $S = \langle A_0, L_1, A_1, L_2, A_2, \dots, L_n, A_n, L_{n+1}, A_0 \rangle$ で表す． A_0 は出発地/帰着地であり， A_i は目的地であり， $A_i = \langle v, actt, rest \rangle$ で表す． L_i は移動方式であり， $L_i \in P$ で表す．ここで， $v \in V, actt \in Q, rest \in Q$ である．各目的地に対して， $actt$ と $rest$ は観光方式であり，特に $rest$ は休憩を表す．各項目のデータ構造は表 (3) を用いて与えられる．記号の意味は表中に簡単に表している．これらの記号を用いて，以下のように問題を定義する．

表 3 データ構造

目的地の属性		移動方式の属性	
$v.arrive$	目的地 A_i の到着時刻		
$actt.conp$	体力消耗		
$actt.recv$	体力回復		
$actt.stayt$	滞在時間		
$rest.restt$	休憩時間	$L_i.movet$	移動時間
$actt.sat$	満足度	$L_i.sat$	満足度

スケジュール S の満足度を与える関数 $Sat(S)$ を以下の式で定義する．

$$Sat(S) = \sum_{i=0}^n A_i.actt.sat + \sum_{i=1}^n L_i.sat \quad (1)$$

$$sat(A_i) = \begin{cases} pre(A_i.v, A_0.v.arrive) & \text{if } A_0.v.arrive \leq S.goalt \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

ここで， $sat(A_i)$ はスケジュール S 内の目的地満足度である． $pre(A_i.v, A_0.v.arrive)$ は目的地 $A_i.v$ を観光後に時刻 $A_0.v.arrive$ に到着する帰着地 A_0 における満足度を表す． $A_0.v.arrive$ は，スケジュール S において帰着地 A_0 に到着する時間を返す関数である． $S.goalt$ は，スケジュール S において帰着時刻に対する制約（例えば，“17:00 まで”）を表す．また，移動方式の満足度は下の式を表す． α は移動時間の閾値である．移動時間が短いほど，得られる満足度が高い．しかし，移動時間が長い場合，閾値 α を超えれば，満足度のペナルティを受ける．よって，移動方式 L_i の満足度を以下の式 (3) で定義する．

$$sat(L_i) = \begin{cases} (\alpha - L_i.movement)/2 & \text{if } sat(A_i) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

ユーザが目的地 A_i での観光終了直後に残っている体力値 $RemSt(A_i)$ は以下の式 (4) と式 (5) で与えられる .

$$RemSt(A_i) = \begin{cases} \min(RemSt(A_{i-1}) + A_i.actt.conp + A_i.actt.recv, MaxSt) & \text{if } i > 0 \\ InitSt & \text{otherwise} \end{cases} \quad (4)$$

$$A_i.actt.recv = A_i.rest.restt \cdot habitus \quad (5)$$

ここで, $MaxSt$ はユーザの最大体力であり, $InitSt$ は出発時の体力値である .

$habitus$ は単位時間内に回復する体力値を表す . $A_i.actt.conp$ は消耗体力値であり, 負の値を持つ . $A_i.actt.recv$ は休憩することにより回復する体力値である .

また, 目的地 v_i の後に, 回復する体力値 $Recovery(i)$ は以下の式 (6) で与えられる .

$$Recovery(i) = \begin{cases} habitus \cdot Rest_i.restt & \text{if } RemSt(A_{i+1}) + A_{i+1}.actt.conp < 0 \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

スケジュールは以下の制約を満たさなければならない .

- 観光の間, ユーザの体力は負の値になってはならない . 全観光地におけるユーザの体力に関する制約条件を式 (7) で示す .

$$\forall A_i.v \in V, A_{i+1}.actt.conp \leq RemSt(A_i) \quad (7)$$

- スケジュール S における観光制約時間条件を式 (8) に示す .

$$\sum_{i=0}^n A_i.actt.stayt + \sum_{i=0}^n A_i.actt.restt + \sum_{i=1}^n L_i.movement \leq S.goalt - S.startt \quad (8)$$

ここでは, $S.goalt$ と $S.start$ は帰着時刻/出発時刻である .

以上を踏まえて本問題の目的関数を以下の式で定義する .

$$\text{Maximize } Sat(S) \text{ s.t. constraints(7) - (8)} \quad (9)$$

2.3 NP 困難性

提案する問題は, 巡回セールスマン問題 (TSP) に出発/帰着点や時間制約, 体力制約等を加えたものである . よって TSP を本問題に帰着することができる . よって, 提案する問題は NP 困難である

3. 提案アルゴリズム

体力を考慮した観光スケジューリング問題を解くため, 本稿では捕食法 (predatory search)^{(8),(10)} を利用した近似アルゴリズムを提案する . 本章ではまず捕食法について述べ, ついでこれを利用した提案アルゴリズムについて述べる .

3.1 捕食法

捕食法は, Linhares が提案した, 動物の捕食行動を模した空間探索手法である . Linhares は巡回セールスマン問題 (TSP)⁽⁸⁾ および大規模集積回路 (VLSI) 設計問題⁽¹⁰⁾ に捕食法を適用した . また, Liu らは解の近傍を再定義することにより, 捕食法を改良した⁽⁹⁾ .

捕食法では, まずランダムに暫定解を生成し, その近傍を探索する . このときより良い解を得られれば探索を続けるが, 規定された繰り返し回数に良い解が得られなければ, この近傍を広げるか, 新たな暫定解を生成する . 近傍の大きさを調整することにより, 広域探索と局所探索のバランスを調整する点に特色がある .

3.2 満足度と体力を考慮した観光スケジューリングアルゴリズム

提案するアルゴリズムを, 暫定解算出フェイズ, 近傍探索フェイズ 2 つのフェイズで構成する . 以下, これらフェイズについて詳しく述べる .

3.2.1 暫定解算出フェイズ

巡回セールスマン問題では, すべての目的地を巡回する経路を求める . しかし本研究で取り扱う問題では, 時間制約に基づいて巡回する目的地をいくつか選抜しなければならない . このとき, 満足度の高い目的地を選抜する必要がある . 提案手法では, まず欲張り法と局所探索法を利用して暫定解を算出する .

(1) すべての目的地 $v_i \in V$ に対して以下を繰り返す .

(a) 目的地 v_i のすべての観光方式 $v_i.actt_j \in A_i$ に対して, 欲張り値 (後述) を計算する . もっとも高い欲張り値を持つ観光方式を, 欲張り観光方式 $v_i.actt_greed$ とする .

(2) すべての欲張り観光方式 $v_i.actt_greed$ の中でもっとも短い滞在時間を最短欲張り滞在時間 $stayt_shortest$ とする .

(3) 観光時間 ($S.goalt - S.startt$) を最短欲張り滞在時間 $stayt_shortest$ で割ったものを, 暫定解の巡回目的地数 $number_greed_destinations$ とする .

(4) $number_greed_destinations$ 個の目的地をランダムに選び, ランダムに並べる . 目

的地 v_i の観光方式は $v_i.actt_greed$ とする．これを暫定解 $current_solution$ とする．

- (5) 以下の処理を用いて暫定解の満足度を計算する．
 - (a) 出発／帰着地点 A_0 から，暫定解 $current_solution$ の目的地を先頭から順に巡回し，出発／帰着地点 A_0 に戻るスケジュール S を生成する．
 - (b) 2.2.2 節に述べた方法でスケジュール S の満足度および帰着時間を計算する．
 - (c) 帰着時間が帰着時間の制約 S_{goalt} を満たさない場合，帰着地点直前の目的地を一つ削除し，ステップ b に戻る．このとき，暫定解 $current_solution$ 自体には改変を加えない．
 - (d) 体力制約を満たさない場合は，もっとも早く体力制約を満たさなくなる箇所の直前の目的地において，スケジュールに休憩を挿入してステップ b に戻る．
 - (e) 得られた満足度を暫定解の満足度 $current_solution_satisfaction$ とする．ただし，暫定解が（帰着時間 S_{goalt} および体力制約以外の）制約条件を満たさない場合は満足度を 0 とする．
- (6) 暫定解を改善するため，以下の処理を $C_{presearch}$ 回繰り返す． $C_{presearch}$ は事前に与えられるパラメータである．
 - (a) 全目的地 V からランダムに目的地 v_a を選ぶ．また，暫定解からランダムに目的地 v_b を選び，これを目的地 v_a と交換する．目的地 v_a の観光方式は $v_a.actt_greed$ とする．この処理により得られた解を変更解 $changed_solution$ とする．
 - (b) ステップ 5 と同様の処理を用いて，変更解の満足度 $changed_solution_satisfaction$ を計算する．
 - (c) 暫定解の満足度 $current_solution_satisfaction$ と変更解の満足度 $changed_solution_satisfaction$ を比較する．改善されていた場合，暫定解 $current_solution$ を変更解 $changed_solution$ で上書きする．
- (7) 暫定解 $current_solution$ が帰着時間 S_{goalt} 以外の制約条件を満たさない場合，ステップ 5.b と同様の処理を用いて，これを満たす解に変更する．ただしこのときは暫定解 $current_solution$ をそのまま改変する（ステップ 5.b では改変しない）．
- (8) 得られた暫定解 $current_solution$ を暫定解算出フェイズの出力とする．
欲張り値は，観光方式ごとに与えられる“ どれだけ効率的に満足度が得られるか ”の指標

となる値である．ここでは， $v_i.actt_j.greed_value$ を表す．“ 単位必要体力あたりの満足度 ”と“ 単位時間あたりの満足度 ”を同時に考慮するため，以下の (10) 式で計算する．

$$v_i.actt_j.greed_value = v_i.actt_j.sat / (v_i.actt_j.stayt + v_i.actt_j.conp / habitus) \quad (10)$$

3.2.2 近傍探索フェイズ

提案手法では，捕食法を利用して暫定解の近傍を探索する．本稿で取り扱う問題では，観光スケジュールは，巡回する観光地，それらを巡回する順番，それらの観光方式により決定される．そこでこれらを $Spotlevel$, $Orderlevel$, $Actlevel$ の三段階で探索する．また，観光スケジュールは $Actlevel$ まで探索されているものとして扱い，記号 A で表す．

- (1) 暫定解算出フェイズで得られた暫定解を A_x とする．
- (2) また， $Spotlevel = 1$ とする． A_x で巡回する目的地数を M とする．目的地集合 V から， M 個の目的地を持つ部分集合 V' をランダムに M 個生成する．ただし重複した場合（同じ部分集合が得られた場合）は片方を破棄して再生成する．
- (3) これら部分集合を，暫定解 A_x と同じ目的地を持つ数の順にソートし， $V'_1 \dots V'_M$ とする．
- (4) $Spotlevel \geq M$ ならアルゴリズムを終了する．
- (5) 暫定解算出フェイズと同様の局所探索アルゴリズムを用いて解 A_y を生成する．ただしこのとき，全目的地 V からでなく，その部分集合 $V'_{Spotlevel}$ のみから目的地を選ぶ（つまり，巡回する順番，観光方式のみを変更する）．
- (6) $Orderlevel = 1$ とする．
- (7) 部分集合 $V'_{Spotlevel}$ をランダムに並び替えた“ 目的地の順番 ”を $3M$ 個生成し，ステップ 2 と同様に，重複した場合は再生成する．これら部分集合を，解 A_y と同じ辺を持つ数の順にソートし，これを $O'_1 \dots O'_{3M}$ とする．ここで辺とは，“ 目的地 v_1 から目的地 v_2 へ移動する ”というような 2 目的地間の順序のことを指す．
- (8) $Orderlevel \geq 3M$ ならステップ 14 へ．
- (9) $O'_{Orderlevel}$ に含まれる観光地の中で，最も多い観光方式の数を N とする． $O'_{Orderlevel}$ の観光方式をランダムに決定したスケジュールを N 個生成し，これらを $A_1, \dots, A_N$ とする．ステップ 2 と同様に，重複した場合は再生成する．これらスケジュールの満足度を，暫定解算出ステップ 5 と同様の手法で計算する（以降，本アルゴリズムでは

すべてスケジュールの満足度を同様に計算する)。最も高い満足度を持つスケジュールを A_{best} とする。

- (10) A_{best} の満足度が A_y の満足度よりも高ければ, $A_y = A_{best}$ としてステップ 5 へ。でなければステップ 12 へ。
- (11) $Orderlevel = Orderlevel + 1$. $Orderlevel = \lceil 3M / \beta \rceil$ ならば $Orderlevel = 3M - \lceil 3M / \beta \rceil$. ただし, ここで $\lceil x \rceil$ は x を小数点以下で四捨五入した値を示す。
- (12) ステップ 8 へ。
- (13) A_x の満足度よりも A_y の満足度が大きければ, $A_x = A_y$ としてステップ 2 へ。でなければステップ 15 へ
- (14) $Spotlevel = Spotlevel + 1$, $Orderlevel = 1$. $Spotlevel = \lceil M / \gamma \rceil$ ならば $Spotlevel = M - \lceil M / \gamma \rceil$.
- (15) ステップ 4 へ。
- ここでは, β と γ は広域探索と局所探索のバランスを調整する重み係数である。

4. 評価実験

本章では, 提案手法の有効性を評価するために以下の 2 つの実験について述べる。

(1) 提案手法により得られた解の最適性を評価するため, 全探索により得られた解, および各比較手法で得られた解と比較した。

(2) 提案手法の計算時間を評価するため, 全探索による計算時間と比較した。

実験環境として, CPU Celeron 2.00GHz, メモリ 2.00GB, OS WindowsXP Professional の PC を用いた。プログラミング言語として Java のバージョン 2.3.2 を用いた。観光者の体力設定として, 体重 65kg の男性を想定した。最大体力値 $MaxSt = 900$, 体力回復率 $habitus = 10$ の値を与えた。実験では, 13500m × 9000m のフィールドの中にランダムな観光地を含んだインスタンスを用いた。観光地の位置は一樣乱数で与えた。観光地の数は 6-24 で (実験による), 各観光地の重要度は 1-100 の間の値の一樣乱数で与えた。実験を簡単化するため, 各観光地の観光方式の数は二つとした。また各観光方式の滞在時間は, 1, 1.5, 2 時間を与えた。以上の設定を表 4 にまとめる。観光地の観光方式はそれぞれ 6 通りとし, 観光方式ごとの消耗体力値は“運動・生活活動の消費カロリー¹⁴⁾”のホームページから計算した。また, 移動方式が自動車として一つしかないを想定し, すべての実験において, 目的地間の移動速度を 30km/h とした。パラメータ α を 30 とした。すなわち, 30 分を超える移動が必要な場合, 満足度ペナルティを受ける。そのほか, アルゴリズムの重み係数は $\beta = \gamma = 3$ とした。

表 4 観光方式・滞在時間の構造の例

観光方式	滞在時間	満足度	体力消費
山に登る a	1h	27	410
山に登る b	1.5h	72	615
山に登る c	2h	85	819
ジョギングと歩行の組合せ a	1h	48	341
ジョギングと歩行の組合せ b	1.5h	79	512
ジョギングと歩行の組合せ c	2h	75	682

4.1 探索効率の評価

提案手法により得られた解の有効性と正確性を評価するため, 全探索および各比較手法により得られた解との比較実験を行った。以下に比較手法について述べる。

4.1.1 比較手法

- 欲張り法: 出発地から出発した後, 次の観光地・観光方式の満足度を最大化することだけを考えて選択する手法。観光者の体力がなくなる時, 休憩を入れて, 帰着時刻まで目的地を巡回する。この手法は次の目的地の満足度だけに注目するため, 移動距離の総和を最小化しようとししない。
- 部分全探索法: 全観光地とそれぞれの観光方式の全組み合わせを作り, 各組み合わせにおいてグリーディにスケジュールを求める。その中の最大満足度を有するスケジュールが解となる。
- ISC 法: 提案手法の暫定解算出フェイズのみを用いた方法。Interim Solution of Calculation (ISC) 法の略記。
- 全探索: 各観光地のすべての観光方式に対して, 全探索で一つの観光地が一つの観光方式を選択し, すべての観光地から目的地と目的地の観光順番を決めて, スケジュールの満足度を計算する。

4.1.2 実験結果

インスタンスとして, 実験 (1) と同じ地図データ 5 個 (地図 1, 2, 3, 4, 5), また, 候補地が 20 個ある地図データ 5 個 (地図 6, 7, 8, 9, 10) を用いた。候補地が 20 個ある地

表 5 比較結果（インスタンス小）

改善率＝（提案手法の満足度－比較対象の満足度）／比較対象の満足度 × 100%					
満足度と改善率	欲張り法	部分全探索法	ISC 法	提案手法	全探索法
地図データ 1	323.57	526.61	529.44	538.88	548.49
比較率	66.54%	2.33%	1.78%		-1.76%
地図データ 2	304.54	451.02	446.36	462.25	475.49
比較率	51.79%	2.49%	3.56%		-2.97%
地図データ 3	430.66	465.95	489.97	501.68	524.49
比較率	16.49%	7.67%	2.39%		-4.35%
地図データ 4	334.27	468.91	483.81	489.23	500.46
比較率	46.36%	4.34%	1.12%		-2.25%
地図データ 5	318.85	466.65	475.76	483.33	492.68
比較率	51.59%	3.57%	1.59%		-1.89%
平均計算時間	0ms	164.57s	0.47s	12.99s	106.2min

図データを用いる時、全探索が実行時間内で計算できなかったため、単純な欲張り法、部分全探索法と ISC 法を比較した。

実験結果を表 5 と表 6 に示す。これらは 5 試行の平均である。表 5 より、全探索が 106.2 分かかかるようなインスタンスに対して、提案手法は、最適解の 95.65%以上の満足度を持つ解を平均 12.99 秒で得ることができた。また、別の三つの比較対象と比べて、実用的な時間内で良い解が得られた。表 6 より、提案手法は多くの場合において、他手法よりも良い満足度を持つ解を得ることができた。一部のインスタンスにおいて部分全探索法よりも悪い解を見つけているが、提案手法ははるかに短い計算時間で解を得ることができた（7.82 時間に対して 62.13 秒）。以上のことから、提案手法は実用的な時間で、高い満足度を持つ解を得ることができると考えられる。

4.2 計算時間の評価

本節では、提案手法の計算時間を評価するため、全探索の計算時間と比較する。実験には、ランダムに生成されたインスタンス 10 個を用いた。地図 10 個で、各観光地数（6,8,10,12,14,16,18,20,22,24）である。また、各候補地の観光方式の満足度もランダムに生成された。全探索は、計算時間が大きくなるため、全探索は観光地数（6,8,10,12）までの場合の計算時間を計測した。全探索の実験結果を表 7 に示す。観光地数 12 の場合、計算時間が 7.08 時間掛ることが分かる。提案手法に対する計算時間の実験結果を図 3 に示す、これらは 5 試行の平均である。提案手法の計算時間に対して、観光地数 18 以下の場合、1 分以内で解を得ることができた。計算時間は観光地数 18 までの場合はほぼ線形に増加したが、20 より上では上昇幅および最大値－最小値の幅が大きくなっている。計算時間は多項式で

表 6 比較結果（インスタンス大）

改善率＝（提案手法の満足度－比較対象の満足度）／比較対象の満足度 × 100%					
満足度と改善率	欲張り法	部分全探索法	ISC 法	提案手法	全探索法
地図データ 6	335.45	540.17	524.08	551.12	-
改善率	64.29%	2.03%	5.16%		-
地図データ 7	432.85	504.73	481.19	507.14	-
改善率	17.16%	0.48%	5.39%		-
地図データ 8	433.09	527.02	494.07	514.94	-
改善率	18.89%	-2.29%	4.23%		-
地図データ 9	439.55	516.06	460.91	479.08	-
改善率	8.99%	-7.17%	3.94%		-
地図データ 10	446.71	466.65	475.76	482.25	-
改善率	7.96%	-5.49%	4.46%		-
平均計算時間	12.4ms	7.82h	2.61s	62.13s	-

おさえられると予想しているが、もう少し大きなインスタンスで実験する必要があると考えられる。

表 7 全探索法の計算時間評価

インスタンス	観光地数 (6)	観光地数 (8)	観光地数 (10)	観光地数 (12)
全探索法	1.37min	18.44min	106.2min	424.93min

5. ま と め

本研究では、観光者の体力と観光時間を制約とし、合計の満足度を最大化するような観光スケジュールの立案手法を提案した。実行時間内に解を求めるために、提案手法は、TSP の解法として注目を集めつつある捕食法を利用している点に特色がある。解の探索効率を評価するため、提案手法と全探索で求めた解と比較した。その結果、最適解の 95.65%以上の満足度を持つ解を平均 12.99 秒で得ることができた。また、欲張り法と比較したところ、20 個の目的地数を持つインスタンスにおいて 1.64 倍の満足度を持つ解を得ることが出来た。

今後の課題として、移動中の体力を考慮することが挙げられる。厳しい天候の元では、移動経路が屋外か屋内か、またバス等を利用できるかが体力消耗に大きな影響を与えることが考えられ、これらの利用スケジュールをうまく立案することは重要であると考えられる。また、目的地の観光自体が体力の回復につながる場合が考えられる。例えばレストランで食事をしたり、温泉で休憩するなど、目的地での休憩も考慮したスケジューリング手法を提案する予定である。

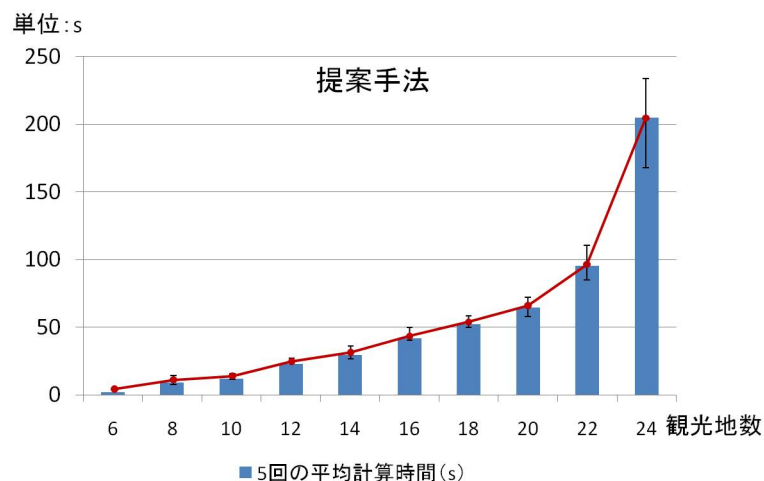


図3 提案手法の計算時間評価

参考文献

- 1) Baus, J., Krüger, A., Wahlster, W., "A Resource Adaptive Mobile Navigation System," Proc. of 2002 Int'l. Conf. on Intelligent User Interfaces 2002 (IUI-02), pp. 15-22 (2002).
- 2) Tang, F., You, I., Guo, M., Guo, S., "Context-aware workflow management for intelligent navigation applications in pervasive environments," Intelligent Automation and Soft Computing, Vol. 16, No. 4, pp. 605-619 (2010).
- 3) Butz, A., Baus, J., Krüger, A., Lohse, M., "A Hybrid Indoor Navigation System," Proc. of 2001 Int'l. Conf. on Intelligent User Interfaces 2001 (IUI2001), pp. 25-33 (2001).
- 4) Cheverst, K., Davies, N., Mitchell, K., Friday, A. and Efstratiou, C., "Developing a context-aware electronic tourist guide: some issues and experiences," Proc. of 2000 ACM Special Interest Group on Computer-Human Interaction (SIGCHI-00) pp.17-24 (2000).
- 5) Rehrl, K., Leitinger, S., Brunsch, S. and Mentz, H. J., "Assisting orientation and guidance for multimodal travelers in situations of modal change," Proc. of 2005 IEEE Int'l. Conf. on Intelligent Transportation Systems (ITSC-05) pp. 407-412 (2005).
- 6) Helsgaun, K., "An Effective Implementation of the Lin-Kernighan Traveling Salesman Heuristic," European Journal of Operational Research, Vol. 126, No. 1, pp. 106-130 (2000).
- 7) 木下 隆正, 永田 宗伸, 村田 佳洋, 柴田 直樹, 安本 慶一, 伊藤 実: 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, 情報処理学会論文誌, Vol.47, No.12, pp.3179-3187 (2006).
- 8) Linhares, A., "State-space search strategies gleaned from animal behavior: a traveling salesman experiment," Biological Cybernetics, Vol.78, No.3, pp.167-174 (1998).
- 9) Liu, C., Wang, D., "Predatory search algorithm with restriction of solution distance," Biological Cybernetics, Vol.92, No.5, pp.293-302 (2005).
- 10) Linhares, A., "Synthesizing a predatory search strategy for VLSI layouts," IEEE Trans. on Evolutionary Computation, Vol.3, Iss.2, pp.147-152 (1999).
- 11) Ahuja, R. K., Ergun, O., Orlin, J. B., and Punnen, A. P., "A survey of very large-scale neighborhood search techniques," Discrete Applied Mathematics 123, pp. 75-102 (2002).
- 12) Maruyama, A., Shibata, N., Murata, Y., Yasumoto, K. and Ito, M., "P-Tour: A Personal Navigation System for Tourism," Proc. of World Congress on ITS, pp. 18-21 (2004).
- 13) 武 兵, 村田佳洋, 柴田直樹, 安本慶一, 伊藤 実: 天気変化を考慮した観光スケジュール群の探索アルゴリズム, 情報処理学会論文誌 数理モデル化と応用, Vol.3, No.1, pp.87-97 (2010).
- 14) Club.Panasonic, "運動・生活活動の消費カロリー," <http://club.panasonic.jp/diet/exercise/mets/index.html>