

多目的遺伝的アルゴリズムによるカーナビゲーションのための経路探索

原 健太[†] 古川 武志[†] 塚原 荘一[‡] 狩野 均[¶]

概要：本論文では、カーナビの経路探索を多目的最適化問題としてとらえ、多目的遺伝的アルゴリズム(MOGA)で経路探索を行う手法を提案する。現行のカーナビでは複数の性質の異なる経路を一度の探索で求めることができない。そこで本手法では経路の経路長、推定旅行時間、運転の快適性の3つの指標を独立した評価関数とし、MOGAを用いて性質の異なる経路を同時に探索して求める。実際のカーナビで用いられているナビ研S規格地図上で実験を行い、GAとDijkstra法とのハイブリッド化、世代交代モデルについて検討を行った。

Routing Guidance for car navigation systems using Multi-Objective Genetic Algorithms

Kenta Hara[†] Takeshi Furukawa[†] Souichi Tsukahara[‡] Hitoshi Kanoh[¶]

Abstract: This paper describes a new route guidance method for car navigation systems using a multi-objective genetic algorithm (MOGA). The problem of selecting a route is a multi-objective optimization problem. But search methods of available car navigation systems cannot give us several routes that differ in kind by one-time search. The proposal method has three independent performance functions (length of route, travel time and amenity of driving) and can provide distinct pareto optimal routes using the MOGA. We performed experiments using the S standard map of the Navigation Systems Researchers' Association, which is the map used in actual car navigation systems. We investigated effectiveness on two terms: a hybrid method of GA and the Dijkstra algorithm, and generation alternation models.

1. はじめに

近年カーナビの普及は著しく、既に1500万台以上が出荷されている。現在カーナビで用いられている経路探索手法はDijkstra法であるが、この手法は厳密解法であるため、計算を終了するまで経路を出力することができない。

著者らは以前①経路の推定旅行時間、②経路長、③運転の快適性の3つの指標を評価値に用いた遺伝的アルゴリズム(GA)による経路探索手法を提案した[1]。このアルゴリズムはDijkstra法と比較して①と②はDijkstra法にはやや劣るが③が優れている経路を求めることができた。しかし実際のカーナビで

は、探索は本質的には多目的最適化問題であると考えられる。そこで本研究では経路探索を多目的最適化問題としてとらえ、多目的遺伝的アルゴリズム(MOGA)[2]を用いて解を求める。またDijkstra法とのハイブリッド化を行うことで経路の質の向上を図る。

以下では、従来手法の問題点と基本戦略、提案する手法のアルゴリズムについて述べる。次に実際のカーナビで利用されているナビ研S規格地図[3]を用いた評価実験結果について述べる。

2. 研究分野の概要

2.1. カーナビゲーションにおける経路探索

実際のカーナビでは旅行時間、移動距離、運転の快適性を考慮した経路を出力する必要がある。本研究では表1に示すように、ドライバーの要求事項を制約条件と見なし、違反点数の合計が最も小さい経路が快適な経路であるとする。一方通行などの交通規則は必ず守らなければいけない制約であるので探索から除外している。

道路網をグラフと見なして、交差点をノード、交差点で区切られた道路の一区画をリンクと呼ぶ。

[†]筑波大学大学院 理工学研究科

Graduate School of Sciences and Engineering, University of Tsukuba

[‡]筑波大学 第三学群情報学類

College of Information Sciences, Third Cluster of Colleges, University of Tsukuba

[¶]筑波大学大学院 システム情報工学研究科コンピュータサイエンス専攻

Department of Computer Science, Graduate School of Systems and Information Engineering, University of Tsukuba

2.2. 従来手法とその問題点

実際のカーナビでは経路探索に Dijkstra 法が用いられている。Dijkstra 法は経路を評価しながら生成する厳密解法であり、計算が終了するまで経路を出力することができず、実時間性が低い。また探索で展開するノードの数も多く、経路を評価する場合に複雑な計算ができない。したがって交通量の予測値や曲がる回数を考慮した経路探索を実時間で行うことが難しい。

GA を用いた探索手法[1]は、反復改善型の探索手法であるので、初期経路集団を生成すれば、探索の途中でも計算を打ち切って解を出力することができる。このアルゴリズムは得られた経路を①経路長、②経路の推定旅行時間、③運転の快適性、の3つの指標で評価しており、このアルゴリズムは Dijkstra 法と比較して、①と②はやや劣るが③が優れている経路を出力することができる。しかし従来手法[1]は①②③の指標の和を評価関数として、これを最適化しているので、3つの指標を独立に評価しながら探索を進めることができない。そのため、一度の探索で性質の異なる経路を同時に出力することができないといった問題点がある。また地図によっては Dijkstra 法より劣る経路を算出することもある。

表1 制約条件と違反点数の例

制約内容	条件	違反点数	遅延時間(秒)
主要道路を通る	高速自動車道	0	-
	国道	2	-
	主要地方道	2	-
	都道府県道	2	-
	基本道	4	-
	その他	6	-
広い道路を通る	L=2以上	0	-
	L=1	1	-
	L=0	3	-
曲がる回数を少なくする	直進	0	0
	左折	3	10
	右折	5	30
信号を通らない	1回	2	20

L:片側車線数

2.3. 提案手法の方針

本手法では 2.2 節で述べた GA の問題点を、評価関数ごとに独立させ、MOGA を用いて経路を求める

ことで解決する。これにより 3 つの指標をそれぞれ最適とする 3 本の経路を同時に求めることができるようになる。

Dijkstra 法とのハイブリッド化を行うことで Dijkstra 法と同等の経路も出力することができるようになると考えている。Dijkstra 法のハイブリッド化方法は、初期個体生成時および初期集団に Dijkstra 法で生成した経路を加えることを行う。また安定して経路が出力できるかどうかをみるために、世代ギャップが小さい世代交代モデルと、交叉で集団全ての個体を利用する世代交代モデルを提案し、その比較を行う。

3. 提案手法

3.1. アルゴリズム

本手法のアルゴリズムを図 1 に示す。コード化、ウイルス集団の生成方法は従来手法[1]と同じなので概要のみを示す。本論文では遺伝子座を交差点(ノード)の通過順、遺伝子をノードの番号とする。したがって個体は可変長となる。国道、主要地方道、都道府県道をウイルスとして抽出し、ウイルス GA を用いて探索を行う。

```

地図データ・出発地・目的地読み込み
ウイルス集団の生成
初期経路集団の生成
指定世代までループ{
  経路の適応度評価
  選択・交叉(Case A または Case B)
}
パレート最適個体抽出
経路の表示
  
```

図1 本手法のアルゴリズム

3.2. 適応度の評価

経路の適応度を経路長、推定旅行時間、運転の快適性の3つの指標を、それぞれ目的関数として用いる。推定旅行時間は道路の種類別にあらかじめ設定しておいた制限速度を用いて算出する。経路に制約違反がある場合は、表1の遅延時間を加算する。運転の快適性については以下の I+II とする。違反点数は表1のものを用いる。

- I. (道路種別、道路幅の違反点数の合計値) ÷ 経路を構成するノード数
- II. 曲がる回数、信号数の違反点数の合計値

3.3. 初期経路集団生成

以下の Step1~3 までを全てのウイルスに対して行い、経路集団を生成する。

Step1:ウイルス集団からウイルス V を 1 つ選択する。

Step2:出発地と出発地に近いウイルスの端までの経路 $R1$ と、目的地と目的地に近いウイルスの端までの経路 $R2$ を、 RTA^* アルゴリズム[4], または経路長をコスト関数とした Dijkstra 法で生成する。なお前者で生成された経路を $RTA^* + Virus$, 後者を $DA(dist)+Virus$ と定義する。

Step3: $R1, V, R2$ を結合して経路 R を生成する。

3.4. 選択・交叉

本手法で利用する選択・交叉方法を以下に 2 種類示す。それぞれ Case A, Case B と定義する。

Case A:

Step1:親 1 を経路集団よりルーレット戦略で選択する。使用する目的関数は経路長とする。

Step2:親 2 を経路集団からランダムに選択する

Step3:親 1 と親 2 に共通のノードがある場合は、その中からランダムにノードを一つ選択して一点交叉し、子 1, 子 2 を生成する。ない場合は Step2 に戻る。親 1 がどれとも交叉できない場合は Step1 にもどる

Step4:親 1, 親 2, 子 1, 子 2 の中でパレート最適なものに次世代に残す。経路集団中に同じ経路が存在する場合は残さない。

Step5:Step1 で使用する目的関数を旅行時間、運転の快適性として Step1~4 を繰り返す。

Case B:

Step1:親 1 を経路集団よりルーレット戦略で選択する。使用する目的関数は経路長とする。

Step2:親 2 を経路集団より選択する

Step3:親 1 と親 2 に共通のノードがある場合は、その中からランダムにノードを一つ選択して一点交叉し、子 1, 子 2 を生成する。ない場合は Step2 に戻る。親 1 がどれとも交叉できない場合は Step1 にもどる

Step4:親 1, 親 2, 子 1, 子 2 の中でパレート最適なものを、親の場合は次世代に残す。子の場合は次世代候補集団に加える。経路集団中に同じ経路が存在する場合は残さない。

Step5:集団中の全ての個体に対して Step2~4 を行う。

Step6:Step1 で使用する目的関数を旅行時間、運転の快適性として Step1~5 を繰り返す。

Step7:次世代候補集団の中からパレート最適な経路を次世代に残す。

4. 評価実験

4.1. 実験方法

本手法の評価を行うために、実際のカーナビで用いられているナビ研 S 規格フォーマット[3]のデジタル地図を利用して実験を行った。実験は乱数のシードを変えて 5 回行った。東京都武蔵野市周辺を対象とした。ノード数は 1029, リンク数は 3037 である。表 2 に比較する手法を示す。

表 2 手法一覧 (括弧内は世代数の上限)

手法	初期経路集団生成方法	世代交代
1	$RTA^*+Virus$	Case A (100)
2	$RTA^*+Virus$	Case B (10)
3	$RTA^*+Virus \& DA(dist)$	Case A (100)
4	$RTA^*+Virus \& DA(dist)$	Case B (10)
5	$DA(dist)+Virus$	Case B (100)

$DA(dist)$: 経路長をコスト関数とした Dijkstra 法

4.2. 本手法の評価

表 3 に実験で得られたパレート最適解の数を示す。表 3 の数値は 5 回の実験の平均値を取った。表 3 の生成個体数は初期個体、交叉で生成された個体の数である。表 3 より初期経路に経路長をコスト関数とした Dijkstra 法を加えることでパレート最適解数が多くなっていることがわかる。

表 4 に各目的関数値に着目した場合の最良経路について示す。 R_d :経路長最小の経路, R_t :旅行時間最小の経路, R_c :違反点数最小の経路と定義する。頻出率は実験 5 回の試行の中でその経路が出てきた割合である。また旅行時間をコスト関数とした Dijkstra 法を比較のため、表 4 に示す。この Dijkstra 法は探索時に信号や右左折による遅延時間を考慮していない。表 4 より次のことがわかる。

- Case B の方が安定して解を出力している
- Case B の方が最良個体をより多く出力している
- Case B の方が最良個体の頻出率が高い
- R_d は手法 1,2,5 よりも手法 3,4 の方がより短い

図 2 にパレート最適解の散布状況を示す。5 回の試行のうち平均的であるものをプロットした。図 2 には手法 2, 4, 5 のみを示している。これは、手法 1 は手法 2 と、手法 3 は手法 4 とほぼ同じ解を出力したためである。図 2 より手法 4 では経路長が短い方にもパレート最適解が分布していることがわかる。

図 3 に本手法と旅行時間をコスト関数とした

Dijkstra 法で生成された経路の例を示す。

表3 実験結果 (括弧内は標準偏差)

手法	パレート最適解数	最終世代個体数	生成個体数
1	4.6 (1.34)	5 (1.41)	626
2	3 (0)	3.2 (0.45)	627
3	15 (1.87)	26 (2.54)	262 (19.1)
4	13.2 (1.92)	14.2 (2.38)	657 (52.6)
5	7 (1.58)	8 (1.58)	323 (44.8)

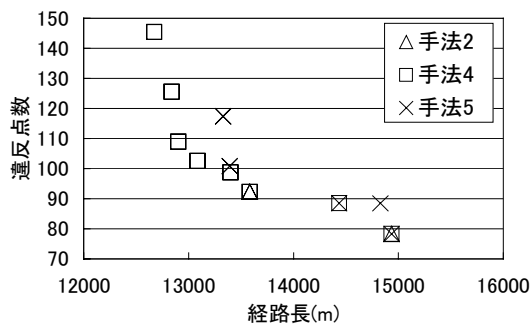


図2 解の散布状況

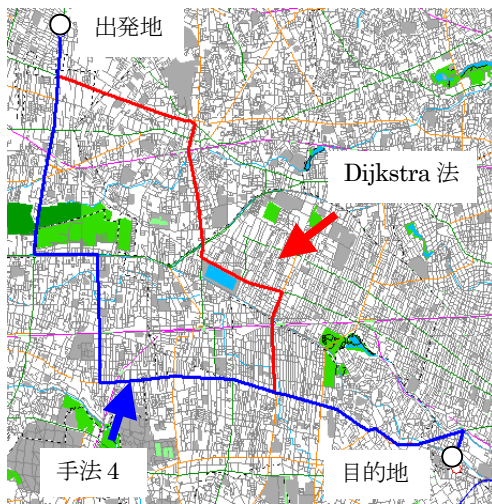


図3 生成された経路の例 (東京都武蔵野市周辺)

5. おわりに

カーナビの経路探索に多目的最適化手法を導入し、MOGA で解を求める手法を提案した。その結果 Dijkstra 法とのハイブリッド化によってパレート最適な経路が増えることを確認した。今後は様々な地域で評価実験を行う予定である。

謝辞

ナビ研 S 規格地図の CD-ROM フォーマットに関する資料をご提供いただいた IT ナビゲーションシステム研究会殿に感謝の意を表します。

表4 各目的関数に着目した最適経路の質

手法		経路長 (m)	旅行時間(秒)	違反点数	頻出率 (%)
1	R_d	13583	1693	92	100
	R_t	14230	1652	87	40
	R_v	14787	1746	76	80
2	R_d	13583	1693	92	100
	R_t	13583	1693	92	100
	R_v	14787	1746	76	40
3	R_d	12673	2349	145	100
	R_t	13583	1693	92	100
	R_v	14787	1746	76	60
4	R_d	12673	2349	145	100
	R_t	13583	1693	92	100
	R_v	14787	1746	76	80
5	R_d	13324	2055	117	100
	R_t	14830	1721	78	100
	R_v	14830	1721	78	100
Dijkstra 法		13587	1706	94	-

参考文献

- [1] 狩野, 中村, 中村: 知識の集団を用いた GA による不特定な立ち寄り地を含む経路探索, 人工知能学会論文誌, Vol.17, No.2, pp145-152 (2002).
- [2] IT ナビゲーションシステム研究会: Format Guide Book S 規格 (Version2.2), (1997).
- [3] E.Zitzler and L.Thiele: Multiobjective evolutionary algorithms: A comparative case study and the strength parato approach, IEEE Transaction on Evolutionary Computation, Vol.3, No.4, pp.257-271 (1999).
- [4] R. E. Korf: Real-time Heuristic Search, Artificial Intelligence, Vol.42, No.2-3, pp.189-211 (1990).