

# 乗換案内における 利用者知識を反映するインタラクティブな経路編集方式

清水 祐弥<sup>†</sup> 小林 亜樹<sup>††</sup>

<sup>†</sup> 工学院大学大学院 工学研究科電気・電子工学専攻 〒163-8677 東京都新宿区西新宿 1-24-2

<sup>††</sup> 工学院大学 情報学部情報通信工学科 〒163-8677 東京都新宿区西新宿 1-24-2

E-mail: <sup>†</sup>cm15011@ns.kogakuin.ac.jp, <sup>††</sup>aki@cc.kogakuin.ac.jp

あらまし 現在の乗換案内インタフェースは、経済合理的な指標を選択することにより探索を行っているが、システム側で想定していない評価指標による検索ができず、利用者自身の経験を表示される経路情報に反映させることが難しい。本研究では、既知の区間に対する利用者自身の経験を表示経路に反映させることができる経路編集方式を提案する。表示されている経路から変更する区間を指定することで、段階的に利用者が表示したい経路に近づき、最終的に希望の経路を表示することができる。

キーワード 乗換案内, 経路情報, ユーザインタフェース

## 1. はじめに

公共交通の円滑な利用を目的として乗換案内<sup>(注1)</sup> サービスが利用されている。このような公共交通の利用経路を検索する操作では、利用者は経済合理的な消費行動に基づいた指標だけでなく、自身の知識や経験、日常的な外的要因を組み合わせ、経路を閲覧することにより行動の意思決定をしている [1]。これには、利用者の路線網に対するマニアックな知識だけでなく、既に経験したことのある事象に対する主観的評価や、利用する日の天候や環境、利用者や同行者の体調などの要因が含まれる。

現在普及している乗換案内のインタフェースでは、経路を表示する順序としてシステム側で想定した経済合理的な指標による選択肢が設けられている。利用者はこれを選択し、表示された検索結果を閲覧、比較することで経路を決定する。しかし、乗換経路探索 Web サービスの利用者調査によると、乗換案内の利用者のうち 8 割が出力された検索結果のうち 1 ページ目<sup>(注2)</sup> しか見ずに経路を決定し、結果として約 8 割の利用者が経路 1 を選択していることから、本来結果画面において複数経路を比較して経路を決定するはずが、実際には利用者が検索エンジンの推薦案にただ従っているという傾向にある [2]。

筆者らは、表示された経路を 2 次元空間図を用いて比較することで満足いくような経路に決定することができる、経路比較式乗換案内インタフェース [3] [13] を提案している。このインタフェースでは、乗換案内システムが導出した経路と比較したい経路を 2 次元空間図を用いて 1 画面上に並べて表示し、利用者がこれを比較することで利用者が満足いくような経路を選択することを目的としたものである。このインタフェースを用いて

経路情報同士を比較するためには、比較対象となる経路を画面上に表示する必要があるが、既存のインタフェースでは検索画面に入力できる項目に限りがあることから、利用者知識による不満を解消した経路をスムーズに表示することは難しい。

そこで本研究では、表示された経路に対し変更を行う区間を指定することで比較対象とする経路をスムーズに表示する経路編集式インタフェースを提案する。本稿ではまず、現在普及している乗換案内のシステムについてモデル化を行い、現在の乗換案内インタフェースを用いた操作について整理する。次に、区間経路編集により利用者知識による不満を解消した経路を表示する経路編集手法を提案し、試作システムの実装を行う。さいごに、利用者知識による不満のうち「既知の区間経路」による不満を解消する経路をスムーズに表示できるかについて、既存のインタフェースと比較することで検証を行う。

## 2. 提案手法

### 2.1 乗換案内システムのモデル化

乗換案内システムは、利用者がシステムに条件を入力するのに用いる検索インタフェース (Search Interface; SI), 入力された条件に従い経路の探索を行う検索アルゴリズム (Search Algorithm; SA), 探索された経路に対して順位付けを行う順位付けアルゴリズム (Route Ordering Algorithm; ROA), 利用者が検索結果を閲覧し比較するのに用いる結果表示インタフェース (Result Indication Interface; RII) により構成される (図 1)。

検索インタフェース (Search Interface; SI) は、利用者が乗換案内で検索を行うためのクエリを入力するインタフェースである。テキストボックス形式で自由に入力ができる一般的な Web 情報検索インタフェースと異なり、乗換案内では予めシステム側で想定している検索クエリと合致するような入力を行う必要がある。現在普及しているインタフェースでは、利用者は利用者は発駅、着駅、経由駅で表される検索対象区間の情報と、時

(注1): 本来“乗換案内”という単語はジョルダン株式会社の提供する公共交通利用経路案内システムを指すが、本稿では一般的な呼称に習い経路案内サービス全般を指す言葉として用いる。

(注2): ここでは Yahoo!路線情報の 1 ページの表示上限である上位 3 経路を指す。

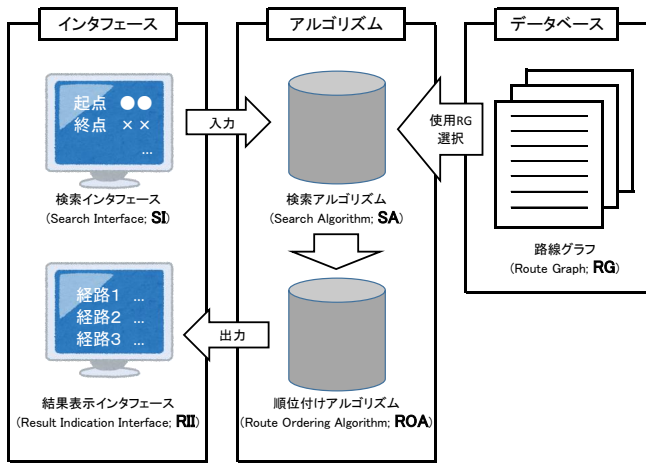


図1 乗換案内システムの構成

図2 検索画面

刻、優等列車や飛行機などの利用条件、表示する経路の順位付け条件を入力する。

代表例として、株式会社駅探の提供する乗換案内<sup>(注3)</sup>のSIの画面を図2に示す。発駅や着駅、経由駅など利用者によって明らかに決定しているものについてはテキストボックス形式で、時刻や順位付け条件など必ず1項目が選択されるものはセレクトボックス形式で、利用条件など項目毎に可否を選択するものについてはチェックボックス形式で入力を行う。テキストボックス形式以外は予め初期値が設定され、特に時刻に関しては大半の利用者が初期値(Webサイトにアクセスした時刻)のまま経路探索を行っている[4]。

検索アルゴリズム(Search Algorithm; SA)は、SIで入力された条件に基づき、条件に合致する経路を探索するアルゴリズムである。SAではまず、入力された時刻と利用条件により使用する路線グラフ(Route Graph; RG)を決定する。例えば、SAで出発時刻を指定した上で、利用交通機関とに制限をしなかった場合、対象利用交通機関を全て含み、列車がそれぞれ時刻情報を含んだ有向グラフとして表現されたRGが用いられる。また、SAで時刻を指定せず、普通列車以外は使用しないと指定した場合、普通列車のみが走る路線情報のグラフで表現されたRGが用いられる。

SAは利用するRGを決定すると、利用可能な経路を探索す

| 時間順                                      | 料金順                                       | 乗換回数順                                   |
|------------------------------------------|-------------------------------------------|-----------------------------------------|
| 1 08:37 → 13:10 早 案 4時間33分 14,920円 乗換 2回 | 2 08:10 → 13:10 早 安 案 5時間0分 14,710円 乗換 2回 | 3 08:04 → 14:09 安 案 6時間5分 14,710円 乗換 2回 |
| 4 10:03 → 14:31 案 4時間28分 15,620円 乗換 2回   | 5 08:00 → 15:05 7時間5分 15,810円 乗換 3回       |                                         |

|                                                  |                  |
|--------------------------------------------------|------------------|
| 1 08:37 発 → 13:10 着 早 案 14,920円 乗換 2回 案 山 交通費を保存 | きっぷ 8,750円 定期代表示 |
| 1本前で出発 1本後で出発                                    | 印刷 メール デキスト      |
| 08:37 新宿 出発 時刻表 構内図 地図 山 駅を保存 ホテル予約              |                  |
| 20分 10.6km 14番線 7駅 J R山手線(内回り) 渋谷方面行き 前後の列車      |                  |
| 08:57着 品川 時刻表 構内図 地図 山 駅を保存 ホテル予約                |                  |
| 1時間33分 359.2km のぞみ213号 新大阪行き 前後の列車               |                  |
| セットがお得! 新幹線+宿泊 ホテルとセットで新幹線が特別割引!                 | 指定席 4,830円       |
| 10:40着 名古屋 時刻表 構内図 出口案内 地図 山 駅を保存 ホテル予約          |                  |
| 2時間22分 166.7km (ワイドビュー) ひだ7号 富山行き 前後の列車          | 指定席 1,340円       |

図3 結果画面

る。経路探索のアルゴリズムにはDijkstra法のような経路グラフにおける一般的な最短経路探索を応用した、公共交通機関の利用における経路探索に特化したものが用いられる[5][6]。SIにおいて経由駅が指定された場合は、起点から経由駅までの経路と、経由駅から終点までの経路をそれぞれ探索し、連結することにより経路を得る。

順位付けアルゴリズム(Route Ordering Algorithm; ROA)は、SAで探索された経路をSIで指定された順位付け条件に基づき並び替えるアルゴリズムである。予めSIで設定された順位付け条件に従って検索結果を並べ替えることで、利用者が検索したい経路が結果表示画面上ですぐに発見できるようにする。順位付け条件には利用者が評価指標としてよく利用すると考えられる経済合理的な消費行動に基づいた項目が用いられ、駅探の場合は「時間順」「料金順」「乗換回数順」の3項目から1つを選択する。

結果表示インタフェース(Result Indication Interface; RII)は、ROAによって並べ替えられた検索結果のうち上位数件を表示するインタフェースである。経路情報同士を比較したり、思うような結果が得られなかった際にSIで入力した条件を変更して検索しなおす再検索操作を行ったり、実際に行動する際の情報を取得するために用いられる。

駅探の場合、最上部に各経路の簡単な比較(総所要時間や総

(注3): <http://ekitan.com/>

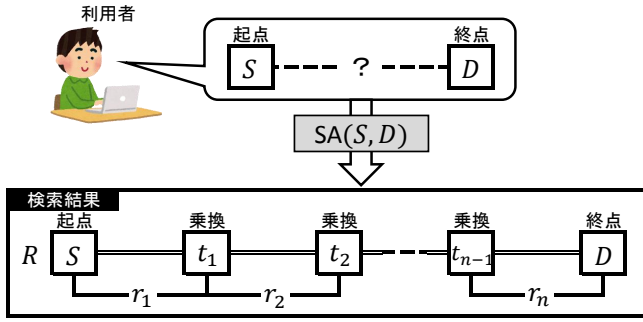


図 4 SA による導出経路

料金など) をする概要表が示され、その下に続けて各経路の詳細が表示される (図 3)。経路の詳細表示では総時間、総料金、乗換回数といった ROA で用いられる指標に加え、途中駅の駅名、各駅での乗換所要時間、乗車する列車名など、実際に移動する際に必要な情報が記載される。

## 2.2 既存インタフェースにおける利用モデル

ここでは、S 駅から D 駅へ向かう利用者を対象に、乗換案内システムを用いて終点に早く着く経路を探索する場合を考える。

利用者はまず、SI に「S 駅から D 駅へ向かう」という探索対象駅を指定する条件と、ROI での順序付け条件を入力して検索を行う。入力した条件は検索アルゴリズム  $SA(S, D)$  に渡され、S 駅から D 駅へ向かう経路  $R$  を得る。この時、S から D への経路  $R(S, D)$  は乗り換え駅  $t_1, t_2, \dots, t_{n-1}$  と  $S, D$  により構成される区間経路  $r_1, r_2, \dots, r_n$  の接続 ((1) 式) として表される (図 4)。

$$\begin{aligned} R(S, D) &= r_1 | r_2 | \dots | r_n \\ &= [S, t_1] | [t_1, t_2] | \dots | [t_{n-1}, D] \end{aligned} \quad (1)$$

SA が導出した全ての  $R(S, D)$  は ROI に渡される。ROI は入力された順序付け条件に基づき  $R$  を並べ直し、その結果を RII に渡す。RII では一般に表示可能な件数が設定されており、ROI により順位付けされた経路の上位数件を画面上に表示する。

ここで利用者は、表示された経路  $R$  に不満がある場合、SI や RII を操作することにより表示されていない経路を表示しようとする。現在普及している乗換案内インタフェースでは、表示する経路を変更する手段として、ROA の変更、RII 上での下位経路閲覧、SA への経由入力、手動での経路連結などの手段がある。

### • ROA の変更

検索結果を確認した時点で ROA に変更の余地がある場合、出力に不満足な場合は順位付け条件を変更することで再度 ROI による並べ替えを行い、ほかの経路を表示することが可能である。インタフェースによっては表示画面上に順位付け条件を変更するタブが表示されているものもあり、利用者はこれを実行することで ROI を異なる条件で再実行することができる。

### • RII 上での下位経路閲覧

一般に RII 上に表示されている経路数は 2 以上であり、同一画面上に複数経路を表示していることから、第 2 候補以下を選択することで不満が解消される場合もある。ただし、既存

の RII では経路の詳細情報は図 3 のように画面下方向に長く伸びているため、この操作を行う場合、画面下部へのスクロール操作と複数経路を比較検討するだけの手間がコストとして発生する。

### • SA への経由駅入力

上の 2 つでも解消できない場合、利用者知識を活用した経由駅指定により SA への入力を変更することが考えられる。一般にこの操作は“再検索”と呼ばれ、SI の入力画面に戻ったり、あるいは SI と同様の再検索入力窓が RII の最下部に配置されているものもある。SI に「S 駅から V 駅を経由して D 駅へ向かう」という探索対象区間の指定をすると、検索アルゴリズムでは S 駅から V 駅へ向かう探索  $SA(S, V)$  と V 駅から D 駅へ向かう探索  $SA(V, D)$  を行い、得られた経路を連結することで経路  $R'$  を得る ((2) 式)。

$$R' = SA(S, V) | SA(V, D) \quad (2)$$

これにも、SI への入力変更操作分のコストが発生する。

### • 手動での経路連結

それでも不満の解消が困難であると利用者が判断した場合、想定される検索経路を複数に分割した検索操作を行い、利用者の手元において経路を連結することを試みる。これには、より多くの利用者知識が必要であるためより多くの手間コストが発生する上に、分割された経路の情報は接続された利用を想定していない料金および時間を導出したものであるため、運賃については別に計算が必要な場合があったり、分割した駅での乗り換え時間を考慮した検索をしなければならないなど、正確な情報を取得するには高度な技術と知識を必要とする。

## 2.3 提案する経路編集手法

本研究では、RII において再検索の対象とする区間を指定し対象区間のみを変更することで、利用者の表示経路に対する不満を解消した経路を表示することを支援する、経路編集式乗換案内を提案する。

本手法では、RII に表示された経路  $R$  に対し、不満点があり検索操作を行う区間と、現在表示されているものに不満点がない区間を指定する。利用者はまず RII 上のインタフェースを操作し、変更を希望する区間と変更を行わない区間の境目となる駅  $t_i$  と、起点方向と終点方向どちらを変更したいかを指定する。ここでは終点方向を指定する場合において説明を行うが、 $S, D$  は経路  $R$  において  $t_i$  に対して対称的なため、同様の手法で対応できる。

RII は指定された情報をもとに SA を用いて「 $t_i$  駅から D 駅まで」の探索を行い、経路  $R_2 = SA(t_i, D)$  を得る。すでに前の探索で得られている  $R$  から  $R_1 = [S, t_1] | \dots | [t_{n-1}, t_i]$  を得ることができるため、 $R_1$  と  $R_2$  を連結することで  $R$  と異なる経路  $R''$  ((3) 式) を得ることができる (図 5)。

$$\begin{aligned} R'' &= R_1 | R_2 \\ &= [S, t_1] | \dots | [t_{n-1}, t_i] | SA(t_i, D) \end{aligned} \quad (3)$$

この検索の特徴は、SA での経由入力とは異なり、あらかじめ表示されている RII に対して不満の対象となっている区間を

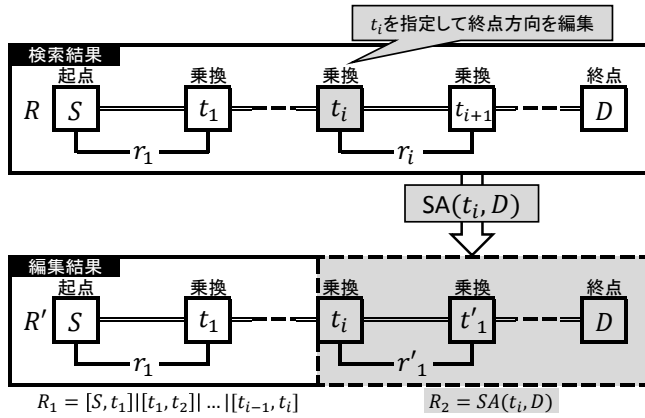


図5 提案手法による経路編集

指定し検索を行っていることであり、以下のような利点が得られる。

- ・ 不満対象区間指定による絞り込み

既存のSAでの経由駅入力では、再検索前に表示していた経路情報のうちの部分に不満があるのかという利用者の情報を伝えることができず、再検索操作を行うと起点から途中駅および途中駅から終点両方に対して指定したSAとROIを使用して探索を行う。そのため、変更を希望していない箇所についても経路が変化することがあり、再検索を行った結果表示される経路すべてを再度見直す必要がある。

一方で、提案手法は不満の含まれる経路区間をRII上で指定して検索するため、不満の含まれない区間については操作を行う前と同一であり、利用者は指定した区間のみを閲覧し、比較するだけで済む。さらに、検索対象の区間も短くすることができるため、不満のない経路の順位が相対的に上がり、すぐに発見することができる可能性が高くなる。

- ・ 表示経路の除外

既存インタフェースによる検索条件の変更操作は、現在表示されている経路に対して明示的に不満であるということを伝えられないため、条件を変更し再検索を行った後であっても、再検索前に表示されていた経路が変更された条件にも合致していた場合、同じ経路が再び表示されてしまう。

一方で、提案手法は操作対象の区間における表示されていた行き方に不満があるということが明示できるため、これを操作後の表示経路から除外することができる。これにより、不満のない経路の順位が相対的に上がり、すぐに発見することができる可能性が高くなる。

## 2.4 ユースケース

乗換案内の利用者が結果画面の情報を閲覧することにより、利用者自身の経験や知識が組み合わされ、新たに様々な要求が発生することが予想できる。例えば、

- ・ 「A駅は大きすぎて乗り換えが大変なので通りたいくない」(駅への不満)
- ・ 「β線は揺れが激しいので酔ってしまうから乗りたくない」(路線への不満)
- ・ 「Γ線を運営するJ社はICカードが使えないので使いた



利用者の知識  
「S駅からD駅まで行きたい」  
「α駅からD駅までは、a線で行くことができる」  
「β駅からD駅までは、b線で行くことができる」

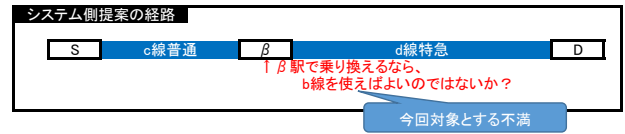


図6 対象とするユースケース

くない」(利便性への不満)

- ・ 「β線とΓ線の乗換はいつもギリギリになるので使いたくない」(乗換への不満)
- ・ 「B駅からC駅へ向かうのならよく利用するδ線を使いたい」(駅と路線への不満)
- ・ 「C駅のΓ線乗り場は寒いので、ビルの中にあるε線乗り場から乗りたい」(乗り場への不満)

などが考えられる。これらの要求は、乗換案内のSIの入力内容に直接指定する内容は含まれていないため、SIやRII上の情報を閲覧し、利用者が操作の内容を判断したうえで不満を解消するような経路を表示する操作を行う必要がある。

本論文では、解消したい不満として利用者既知の途中駅と利用経路による「既知の区間経路の利用」を対象とする。利用者は、検索条件である「S駅からD駅へ向かう」という知識の他に、「 $t_x$ 駅から $t_y$ 駅へは、 $R_{xy} = [t_x, t'_1] || [t'_1, t'_2] || \dots || [t'_m, t_y]$ という方法で行ける」という知識を持ち合わせている。

利用者は、検索した結果 $t_x$ 駅と $t_y$ 駅を乗換駅として利用しているにもかかわらずこの区間に $R_{xy}$ を利用していない場合、「 $t_x$ 駅と $t_y$ 駅を利用しているのなら $R_{xy}$ を利用するのに、なぜ利用しないのだろう」という疑問を持つ。これを検証するには、S駅から $t_x$ 駅、 $t_y$ 駅からD駅までの区間はそのままに、 $t_x$ 駅から $t_y$ 駅については $R_{xy}$ を利用する経路を表示し、変更前の経路と比較する必要がある。

このような検索開始時には明確な目的でないような情報取得においては、目的の明確な情報取得と異なり、興味を持った情報に対して単純な操作によりスムーズに情報取得へ導くことが重要とされる[7]。そこで今回は、このうち $t_y$ 駅をD駅とした状況、つまりある駅から終点駅までの行き方を複数知っている利用者が、既知の行き方を利用する経路を発見できるかどうか(図6)について検証を行う。

## 2.5 既存研究における表示経路変更

乗換案内に関する研究は主にSAの最適化に重点が置かれており、近年においてもバスネットワークにおける経路探索システム[8][9]や代替経路探索手法の提案[10]などが行われているが、表示される経路情報に対し利用者の希望を反映させることを対象とした研究は多くない。

これまでに、利用者の知識や経験を提案経路に反映する手法として、利用者の行動履歴や購買履歴から趣味や興味を推測し表示経路に反映する手法[11]が提案されている。しかし、これは通勤・通学などのルーチンの利用する経路に対し変更を反



映させることを目的とした研究であり、表示された経路に不満があったり、探索毎に評価が変化したりする状況は対象としていない。

また、制約ベースの可視化手法を用いたインタラクティブシステムにより表示ルールを切り替え、利用者の多様な要求に応える手法[12]が提案されているが、検索システムと可視化モデルの提案のみとなっており、実装および評価には至っていない。

現在提供されているサービスの中には、列車の遅延や運休が発生した際に、現地の列車の遅れや混雑状況を該当路線の利用者に投稿してもらい<sup>(注4)</sup>、それを検索結果に並べて表示することで経路の判断に役立てるようなサービスが提供されているものもあるが、実際に提案される経路の順位に反映されるような段階には至っていない。また、近年列車の運行情報や車両ごとの混雑状況を鉄道会社が提供するようなサービス<sup>(注5)</sup>が増えつつあり、このような旅客動向に関する情報を提案経路に反映することなども考えられるが、現在のところこれらを利用して表示経路を変更するような乗換案内サービスは確認していない。

これらの研究やシステムは、主に SA に対し変更を加えたり、RII に対し新たなアルゴリズムや別のサービスの情報を加えることで利用者毎や利用場面に特化した検索結果を得るものである。しかし本提案手法は、SA や ROA といったアルゴリズムには手を加えず、また SA や RII に対し新たなアルゴリズムを加えることもせず、RII における入力インタフェースに変更を加えるのみであるという特徴がある。これにより、アルゴリズムに手を加えたり、新しいデータベースを追加することなく、高度な検索を実現できる。

### 3. 試作システム

今回は提案手法のうち、指定した駅と終点駅との間を指定して編集する機能について、Web ブラウザ上で実行できるシステムとして実装を行った。本システムは、「検索画面」「駅決定画面」「結果表示画面」「経路編集画面」の4画面にて構成されており、このうち「検索画面」および「駅決定画面」については、検索画面において指定できる項目が「乗車駅」「下車駅」「日付」のみである(図7)以外には、既存のインタフェースと変更点はない。

本実装では、SA および ROA には株式会社駅探の乗換 ASP サービス(標準時分版)<sup>(注6)</sup>を用いたため、具体的な発着時刻を指定することはできないが、これは使用する探索グラフが時刻付きデータによるか否かに対応するのみであるため、本質的な違いはない。表示される駅名は、使用するサービスの駅情報によるものを使用する。

結果表示画面は、縦軸が総料金、横軸が総時間の2次元空間図[13]で表される(図8)。各経路は橙色の駅ノード、青色または赤色の乗車エッジ、灰色の徒歩エッジで表される。この画面上では、利用者は2次元空間上の長さや高さ、色により各経路

| メディア情報システム研究室 経路編集式乗換案内システム       |                                 |
|-----------------------------------|---------------------------------|
| 乗車駅                               | <input type="text" value="新宿"/> |
| 下車駅                               | <input type="text" value="高山"/> |
| 2017                              | 年 3 月 6 日                       |
| <input type="button" value="検索"/> |                                 |

図7 検索画面

の変数同士を比較することができる。

各経路情報を示す線分集合の終端駅の右側に「削」ボタンと「編」ボタンが配置される。「削」ボタンは該当の経路を削除し、「編」ボタンは該当の経路を基準として経路比較画面に遷移する。

本実装では、駅決定画面から結果表示画面に遷移する場合、終点駅に早く着く上位3つの経路を表示する。この順位は、使用したシステムの SA および ROA の実装に依存する。

利用者は表示された経路中に不満な区間がある場合、任意の経路の右側にある「編」ボタンを押すことにより、経路編集画面(図9)に移行する。経路編集画面では、選択した経路の途中駅の上に右矢印のボタンを配置している。利用者はどの駅から先を変更するかを決めたら、該当する駅ノードの上にあるボタンを押す。これにより、指定した駅から起点側の区間が表示されている経路と同様で、且つ指定した駅から終点側の経路が表示されて行き方経路と異なる経路を探索し、経路比較画面に編集前の経路と並べて表示する。

今回は実装の都合上、起点から終点までの探索で早く着く経路上位30件のうち、条件に該当した経路を最大で3件まで追加するようにした。表示する経路と順位は、使用したシステムによる結果に依存している。

## 4. 実験

### 4.1 評価項目

本実験では、対象とした既知の途中駅と利用経路による不満を解消する経路探索について、操作難易度と所要時間の観点から有効であるかどうかを検証する。評価対象を RII で表示された経路に対する変更操作に限定するため、表示された経路同士の要素比較や、発見した経路が実験参加者にとって満足いくものであるかどうかについては検証しない。経路編集手法が対象とする探索の達成に有効かどうかのみを検証対象とする。

比較インタフェースとして、一般的な乗換案内インタフェースを有する Yahoo! 経路情報<sup>(注7)</sup>の乗換案内サービスを使用する。これは、著者らが把握している2016年12月現在無料で提供されている乗換案内サービスのうち、実装した試作システムと同様な標準時分検索に対応している唯一のサービスであるためである。なお、インタフェースについては前述した駅探の乗換案内インタフェースとほぼ同様のものである。

### 4.2 設定課題

本実験では、課題として「起点駅」「終点駅」「条件」を提示し、起点駅および終点駅を入力した上で検索操作を行い、条件

(注4) : <http://www.jorudan.co.jp/unk/live.html>

(注5) : JR 東日本アプリ (<http://www.jreast-app.jp/>) など。

(注6) : <https://go.ekitan.com/service/asp/transit/index.shtml>

(注7) : <http://transit.yahoo.co.jp/>

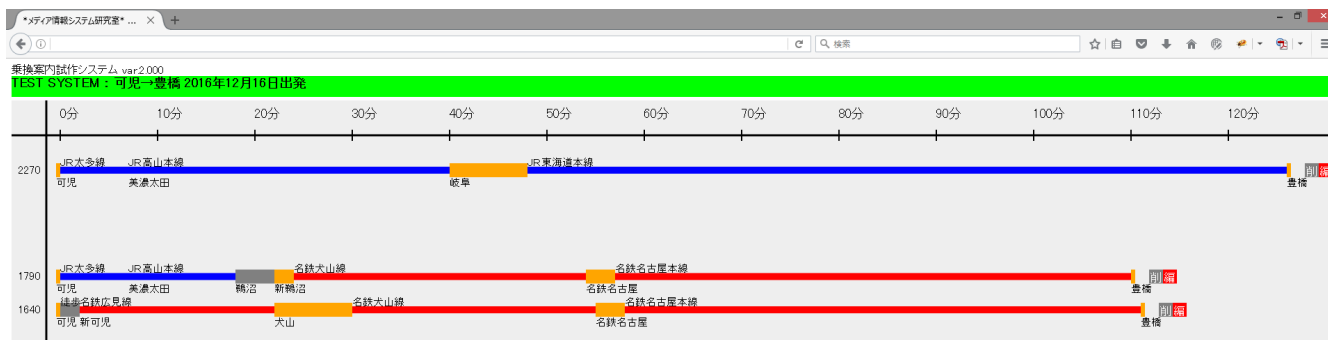


図 8 結果表示画面

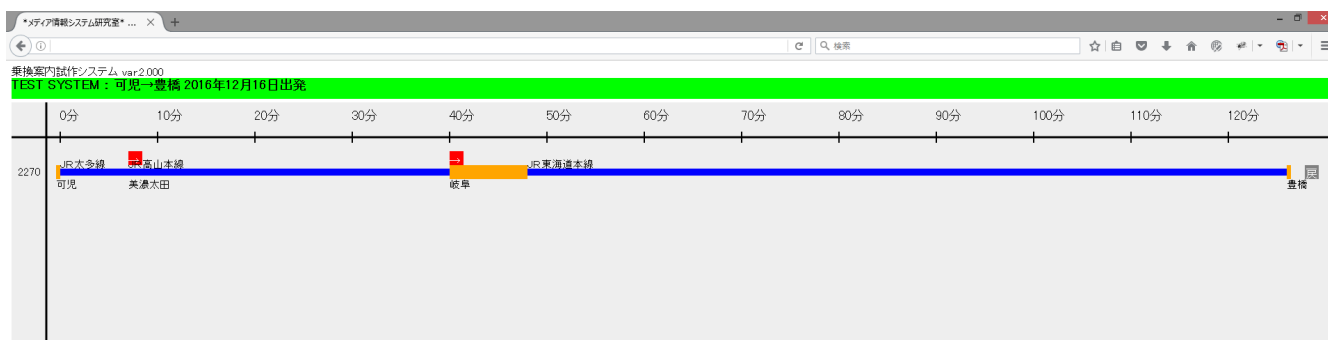


図 9 編集画面

に合致する経路を探索してもらう。

「条件」には利用者知識，ここでは乗り換える駅と利用する路線の情報を記載する。提案手法は，経路について知識を持った利用者が RII 上で検索結果を閲覧した上で，表示経路を変更する操作を対象としている。本実験ではこの操作を再現するため，1 つの問題につき経路が表示可能な条件と，所要時間が長すぎる，折り返し乗車になってしまう等の理由により SA が経路として導出しない条件の 2 つを提示する。実験参加者にはいずれか 1 つは表示不可能な条件であることを伝え，起点駅と終点駅のみで検索を行い RII 上に表示される結果を基に，利用する条件を選択してもらう。

表 1 問題例

|        |                                     |
|--------|-------------------------------------|
| 起点駅    | 高田馬場 (たかだのばば)                       |
| 終点駅    | 高尾 (たかお)                            |
| 条件 (1) | 新宿駅から京王線で高尾駅へ                       |
| 条件 (2) | 高麗川駅から八高線に乗り，<br>八王子駅で中央線に乗り換えて高尾駅へ |

問題例を表 1 に示す。この場合，条件 (2) は実装で用いた SA が導出しない経路であり，条件 (1) を利用することになるが，第 1 候補として新宿駅から中央線を利用する経路が表示されるので，これを京王線を利用する経路を表示するように操作してもらう。

#### 4.3 実験手順

本実験の実験参加者は，20 代の大学生および大学院生の男性 8 名である。

まず，参加者に今回実験に使用するインタフェースの操作方

法を説明し，例題を 1 問解いてもらう。次に課題を提示し，その課題に合致する経路を探索してもらう。起点駅と終点駅をインタフェースに入力し検索を行い，RII に表示された経路を閲覧した上で，いずれかの条件に合致する乗換駅を発見した場合，その駅から条件に従うような経路を探索してもらう。条件に合致する経路を発見できた場合，どちらの条件に合致したかを記入してもらい，次の問題へ進む。

問題は A 群と B 群のそれぞれ 2 問づつ用意し，それぞれのインタフェースを用いて解いてもらう。その際，操作画面をキャプチャし，画面をどのように操作しているのか，また正答経路を表示できているかを確認する。

実験終了後にアンケートを実施し，普段の乗換案内の利用状況について調査した。

### 5. 実験結果と考察

実験後のアンケートの結果，実験参加者 8 名は全員乗換案内インタフェースの利用経験があり，利用傾向として利用者調査 [4] から大きく逸脱するような参加者は確認されなかった。実験の所要時間は一人あたり約 15 分であった。

#### 5.1 参加者毎の所要時間と正答数

各実験参加者の探索所要時間と正答数を表 2 に，探索所要時間の散布図を図 10 に示す。なお本実験では，探索所要時間として各問題における「結果画面が最初に表示されてから答えを記入し次の実験へ移る (最後の問題では解答を終了する) まで」の時間を計測し，2 問の合計を示す。“既存”は既存システムである Yahoo! 経路情報，“提案”は試作システムを指す。所要時間比は，提案システムが既存システムに対してどの程度の時間

表 2 探索所要時間と正答数

| 実験参加者       | A   | B   | C   | D   | E   | F   | G   | H   |
|-------------|-----|-----|-----|-----|-----|-----|-----|-----|
| 既存:所要時間 [s] | 168 | 123 | 105 | 232 | 257 | 129 | 234 | 226 |
| 既存:正答数 [問]  | 2   | 1   | 2   | 2   | 2   | 2   | 1   | 2   |
| 提案:所要時間 [s] | 111 | 64  | 125 | 91  | 98  | 89  | 154 | 84  |
| 提案:正答数 [問]  | 2   | 2   | 2   | 2   | 2   | 2   | 2   | 2   |
| 所要時間比 [%]   | 66  | 52  | 119 | 39  | 38  | 69  | 65  | 37  |

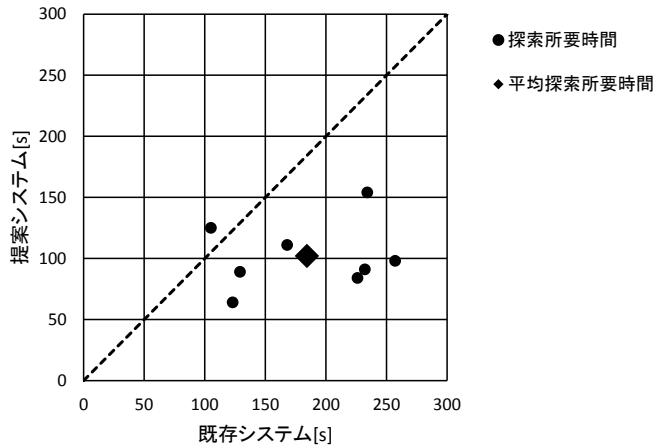


図 10 参加者毎と平均の探索所要時間

がかかったかを示すもので、提案システムの所要時間を既存システムの所要時間で割った値に 100 を掛けたものである。

探索所要時間を見ると、実験参加者 8 名のうち、7 名の探索所要時間が短縮され、短縮された 7 名は 31%~63%の時間短縮となった。一方で、実験参加者 C の探索所要時間は 19%の増加となった。

正答率を見ると、提案システムでは全参加者が 2 問とも正解しているが、既存システムでは参加者 B と参加者 G が 1 問づつ誤答した。実験画面と実験後アンケートを確認したところ、参加者 B は課題の条件に合致しない経路を閲覧した際に、誤って合致していると判断してしまい探索を終了しており、また参加者 G は結果画面上に正答となる経路は表示されていたものの、その経路を発見できずに探索操作を諦め終了している様子が確認できた。

比較システムの平均所要時間は 184[s]、提案システムの平均所要時間は 102[s] となり、提案システムは既存システムと比較して約 45%の所要時間短縮が見られた。各実験者におけるシステム毎の所要時間について t 検定を行ったところ、 $p = 0.007$  となり有意差があることを確認した。

## 5.2 探索内容毎の所要時間の検証

探索所要時間を比較操作時間 (R11 画面表示時間) と再検索操作時間 (既存)・編集操作時間 (提案) に分類した平均所要時間を図 11 に示す。再検索操作時間は画面上に表示されている再検索画面に入力を開始してから検索ボタンを押すまで、編集操作時間は編集ボタンを押してから経路編集操作を完了するまでの時間を指す。また、再検索操作および編集操作をまとめて変更操作と呼ぶ。

比較操作においては所要時間平均が 148[s] から 94[s] に約 37%の減少、変更操作においては 36[s] から 8[s] に約 77%の減

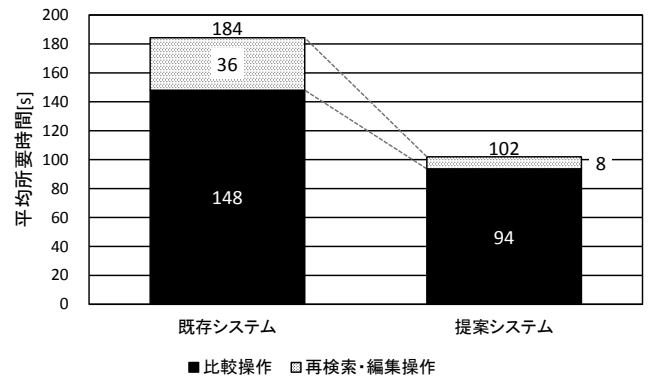


図 11 分類した平均探索所要時間

少が見られた。操作画面を確認したところ、変更操作においては ROA の指定を変更した参加者はまれであり、ほとんどの参加者が経由駅を指定することにより表示経路を変更していた。

2.2 節で述べた手間コスト要因のうち、第 2 候補以下を選択する操作は比較操作時間に、ROA の指定変更および利用者知識による経由駅指定は再検索操作時間に相当する。なお本実験においては、既存システムにおいて経路を手で連結する操作をした参加者は見られなかった。

### 5.2.1 比較操作

比較操作時間の減少について、参加者の操作画面を確認したところ、以下の 2 つの要因が考えられた。

#### ● 下位の経路の発見

比較インタフェースでは、再検索後において希望の結果に絞り込むことができず正答が下位となり、解答となる経路を発見するために下位の経路詳細表示までスクロールした参加者や、Yahoo!経路情報による表示条件である上位 8 件に入らず、再度再検索を行う参加者が多く見られた。また、再検索操作を行わず、最初に表示された経路の中から下位にある正答の経路を発見する参加者もいた。

一方で提案インタフェースでは、全ての課題において 1 回の編集操作後に表示される最大 3 件の経路の中に解答となる経路が表示されていた。これより、編集する区間を指定することにより希望の経路が上位になるような指定が行われたことで、スクロールや再検索による追加の手間コストが発生せず、比較操作の所要時間が短縮されたと考えられる。

#### ● 経路情報の把握

2 次元空間を用いた乗換案内表示は、1 画面上に複数の経路情報を表示することにより経路の詳細情報把握に対し所要時間を短縮する効果がある [14] [15]。

今回、比較インタフェースにおいて正答となる経路が表示されている状況で、参加者が正答を発見できずに次の経路へスクロールしてしまい、RA での検索条件を変更する操作が見られた。一方で提案インタフェースでは、全ての課題において表示情報から条件となる経路を見逃す操作は確認できなかったため、これにより比較操作の所要時間が短縮されたと考えられる。

### 5.2.2 変更操作

変更操作時間の減少について、同様に操作画面を確認したと

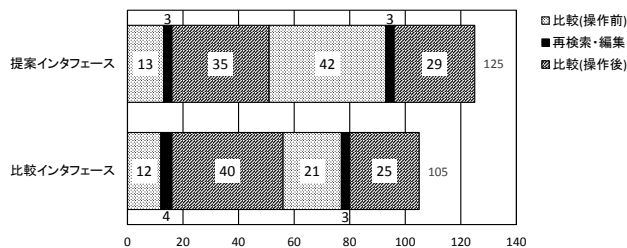


図 12 被験者 C の探索所要時間の詳細

ころ、以下の 2 つの要因が考えられた。

- 入力インタフェースの違い

比較インタフェースにおける再検索操作のうち、特に経由駅の追加操作においては、入力窓の指定とキーボードによる操作が必要となる。一方で提案インタフェースではマウスによるボタン操作のみとなっているため、入力デバイス同士を行き来する時間と、キーボードの操作時間の分だけ所要時間が短縮できていると考えられる。

- 入力時の再考時間

比較インタフェースにおいて特に 2 度目の再検索を行う場合においては、再検索画面において手が止まったり、また様々な検索指標をクリックし検討したりするような様子が見られた。

これは、1 度目の再検索には特にこのような様子が見られなかったことから、1 度目の再検索で正答が得られると想定していた際に、他に条件に合致する経路を上位に表示する SA および ROA の条件を入力する方法が分からず、これにより再検索画面上での操作時間が延びたのではないかと考えられる。

### 5.3 所要時間が延びた参加者の考察

既存インタフェースよりも提案インタフェースの方が所要時間が延びてしまった参加者の操作について、なぜそのような結果が得られたのかを考察する。

実験参加者 C の操作所要時間の詳細を図 12 に示す。ここでは、比較操作における所要時間を再検索および編集操作を行う前と行う後に分類している。

再検索・比較時間や、操作後の比較時間についてはほとんど差がないが、提案インタフェースにおける 2 問目の編集操作前の比較操作において他の比較操作よりも 2 倍以上の時間がかかっていることが分かる。これについて実験参加者 C に確認したところ、該当する問題において徒歩を示す表示が太い矩形で表示されていることから、移動時は細い線で表示されるという参加者の直観的理解と繋がらず、経路の情報を理解に時間がかかったとの回答を得た。

よって、この結果は経路編集方式が要因ではなく 2 次元空間図における経路表示方法に要因があり、2 次元平面インタフェースにおいては視点移動の煩雑さが所要時間増加の要因となる可能性が示唆されていることから [13]、今回もこれが影響したものと考えられる。

## 6. おわりに

本研究では、経路比較式乗換案内インタフェースにおける経

路編集操作について、表示された経路情報を基に区間経路を指定し編集する方式を提案し、既知の途中駅と区間経路を対象とした不満を解消した経路の表示について既存インタフェースと比較して有用であるかを検証した。

今後は、他の不満に対する解決について検証を行うとともに、経路比較観点からの 2 次元空間表示の有用性についても実験および検証を行うことで、インタフェースの検討を進めていく。

## 謝 辞

本研究の一部は科研費 (26242013) の助成を受けたものである。

## 文 献

- [1] K.Masaki, R.Otsuka, H.Fujikura and H.Shiizuka, “Change in User’s attitude by giving the unknown Information”, International Symposium for Emotion and Sensibility 2008, Proceeding, pp.130–134, 2008.
- [2] 寺部 慎太郎, 齊藤 あづさ, 郷原 翔一, “鉄道経路探索 Web サービスの利用者調査”, J-Rail 講演論文集 2009(16), pp.457–458, 2009-12-02.
- [3] 清水 祐弥, 小林 亜樹, “乗換案内での経路選択のためのファセットナビゲーション”, 第 7 回データ工学と情報マネジメントに関するフォーラム (DEIM2015), D6-1, 2015-03-03.
- [4] 寺部 慎太郎, 重里 光佑, 内山 久義, “鉄道経路探索 web サイトに関する利用実態の特徴分析”, 土木計画学研究・講演集, 37, No.341, 2008-6.
- [5] 菊地 新, 高木 啓三郎, 加藤 誠己, “動的ネットワーク表現に基づく列車・航空便への最適乗継系列探索の手法とその実際への応用”, 情報処理学会論文誌 38(4), pp.915–926, 1997-04-15.
- [6] 半田 恵一, 田中 俊明, “乗換え案内サービスにおける経路探索手法”, 電子情報通信学会論文誌. D-I, J88-D-I(10), pp.1525–1533, 2005-10-01.
- [7] 川島 晴美, 佐藤 吉秀, 関口 裕一郎, 佐々木 努, 大久保 雅且, 奥雅博, ““気付き”に着目した情報提供システム「HotWindow」の開発”, 電子情報通信学会論文誌. D, J89-D(11), pp.2445–2457, 2006-11-01.
- [8] 川村 尚生, 楠神 元輝, 菅原 一孔, “徒歩移動を考慮するバス経路探索システム”, 情報処理学会論文誌 46(5), pp.1207–1210, 2005-05-15.
- [9] 川村 尚生, 菅原 一孔, “バスネットワークのための実用的な経路探索システム”, 情報処理学会論文誌 48(2), pp.780–790, 2007-02-15.
- [10] 大橋 俊道, 唐崎 幸弘, “乗り換え案内サービスにおける代替経路検索技術について”, 電気学会研究会資料 TER, 交通・電気鉄道研究会 2006(75), pp.35–38, 2006-10-27.
- [11] 中村 啓之, 高原 高 赫, 清弘 晃史, 峯 恒憲, “マルチモーダル乗換案内推薦システム Patrash の提案と試作に向けて”, 電子情報通信学会技術研究報告. AI, 113(332), pp.47–52, 2013-11-21.
- [12] 酒井 恵光, 中山 健, “ユーザの意図を反映した乗換経路情報抽出のためのインタラクティブシステム”, 電子情報通信学会技術研究報告. LOIS, 114(321), pp.55–58, 2014-11-14.
- [13] 清水 祐弥, 小林 亜樹, “経路編集を用いた乗換案内のための経路比較インタフェースの提案”, 第 8 回 Web インテリジェンスとインタラクション研究会予稿集 (インタラクション支援), 2016-06-05.
- [14] 近藤 聡士, 小林 亜樹, “乗り換え案内結果の集約化インターフェースの提案”, 情報処理学会全国大会講演論文集 2012(1), 3P-4, pp.709–710, 2012-03-06.
- [15] 清水 祐弥, 小林 亜樹, “経路編集を用いた乗換案内における経路探索方式”, 第 8 回データ工学と情報マネジメントに関するフォーラム (DEIM2016), H1-1, 2016-02-29.