卒業論文

観光経路決定支援システムのための大規模事例ベースの作成

指導教員　山崎　勝弘　教授

立命館大学理工学部情報工学科４回生

2170910095-2　福井　勝史

2170910030-8　柿田　貴史

95.1.27

内容梗概

本論文では、メモリベース推論による観光経路決定支援システムを実現するための大規模事例ベースの作成、及びシステム全体としての実験を行ない、並列化の時間的評価、結果としての解の評価、ルールベース推論による同システムとの評価について述べる。

大規模事例ベースは、観光地を5つのブロックに分割し、それぞれのブロック毎に大量の事例を生成することによって実現する。さらに、各ブロックを2つのサブブロックに分割した場合の大規模事例ベースも作成し実験を行なう。

事例ベースが大規模(最大160000個)になるため並列計算機AP1000を使用し検索時間の短縮を図る。ここで事例の生成に時間を奪われないように1台のワークステーション上で必要分の事例を生成し、それをAP1000の各プロセッサに取り込むようにする。

評価の方法として、実験を行い検索時間と検索率、解の質を求め、さらにルールベース推論による同システムとの比較を行う。

本論文では2章で既存の観光経路決定支援システムの概要を示し、モジュール毎に説明する。3章では本研究に使用する並列計算機AP1000の構成を示すと共に、新たなシステムの構成を示す。

4章では本論文の中心となる大事例ベースについて説明し、本研究で使用するサブブロックの説明、事例の生成方法の説明を示す。

5章では実際のシステムでの実験結果を記述し、その結果に基づいて考察を行う。

目 次

図目次

表目次

1 はじめに

現在、エキスパートシステムの代表的な問題解決法としてルールベース推論（Rule-Based Reasoning:RBR）と事例ベース推論（Case-Based Reasoning:CBR）が挙げられる。ルールベース推論が規則によって知識を表現する推論方式であるのに対し、事例ベース推論は規則と事例によって知識を表現する推論方式である。過去に発生した問題解決事例を蓄積し、新たな問題にであった時、事例群のなかから類似した問題を見つけ出し問題解決に利用する[1]。

事例ベース推論とは基本的に次のような推論方式である[1]。

1. 過去に経験した事例（問題・解答・解答を得る筋道）を事例ベースに蓄えておく。
2. 新たな問題の解答を得たい時は、まずその問題に類似した概知の問題を事例ベースから検索する。
3. 概知の問題の事例を利用（修正）することにより当該問題の解答を得る。
4. 当該問題をその解答や反省材料とともに事例ベースに収納する。

これまで、事例ベース推論を用いた観光経路決定支援システムの研究が行われてきたが[4]、課題として事例ベース推論における実行時間の削減が考えられている。実行時間の割合を考えると、最も大きい割合をしめる事例修正時間を削減することが挙げられる。ここで、近年注目されつつあるメモリベース推論(Memory-Based Reasoning:MBR)の考え方が有効となってくる。

メモリベース推論では基本的に以下のように推論する[1]。

1. 実例をデータベースに登録する。
2. 入力と実例の間に類似度（距離）を定義する。
3. 入力と最も似た実例を模倣して出力を得る。

メモリベース推論は「知的活動は、ルールではなく過去の経験的記憶を中心にして行われる。」という仮説に基づく考え方であり、事例ベース推論は「問題解決を、過去の事例の適用・修正によって行う」という考え方である。基本的な考え方は同じであるが、大きな違いは事例ベース推論では類似事例を見つけ出し、修正を加えることによって解を得るが、メモリベース推論では検索した結果をほとんど直接解とするところにある[2]。これより、事例修正時間の削減が可能となってくる。

本研究ではメモリベース推論を実現するため大規模な事例ベースを作成し、事例数を大幅に増やすことによって類似度の高い事例を検索可能にし、事例修正部の負担を減らした。また、現在注目されている並列マシンAP1000を用いて事例検索部を並列化した。システム構成としては、事例生成部・事例検索部・事例修正部からなる。事例生成部で事例を生成し、並列マシン上の各プロセッサに均等に分散する。事例検索部で生成された事例から当該事例を検索し、事例修正部により解の修正・結合を行う。小規模事例ベースから徐々に実験を行い、既存のシステムとの比較からメモリベース推論の有効性を見い出すことを目的とする。

本論文では2章に既存の観光経路決定支援システムの概要を述べる。3章では並列マシンAP1000の構成、メモリベース推論を使用した本研究でのシステム構成を述べ、4章において大規模事例ベースの生成について述べる。また、5章ではメモリベース推論の有効性を述べるため実験を行ない、時間的評価・修正による解の評価及びルールベース推論との比較を行なう。最後にこの実験結果の考察もこの章で行なう。

2　既存の観光経路決定支援システムの概要

2.1　システム構成

既存のシステムの構成を図 1に示す。また、各部の役割について簡単に説明する。

2.1.1　ユーザインターフェース部

ユーザインターフェース部は、ユーザが行きたい観光地を入力する入力部と、推論結果を表示する出力部で構成される。

2.1.2　制御部

制御部は、入力されたデータをもとに事例ベース推論部を起動して類似事例の検索を行い解を求める。ここで類似事例が存在しなければ、ルールベース推論部を起動して解を求める。

2.1.3　事例ベース推論部

事例ベース推論部は、事例検索部と事例修正部、交通手段作成部に分かれる。事例検索部ではユーザの入力に対する類似事例を検索し、事例修正部では検索した事例の修正を行う。そして、交通手段作成部で各観光地間の交通手段を求めて解とする。

2.1.4　ルールベース推論部

ルールベース推論部は経路作成部と交通手段作成部に分かれる。経路作成部では、観光地の座標を基に最適解にできるだけ近い経路を作成し、交通手段作成部では事例ベース推論部と同様に各観光地間の交通手段を求めて解とする。

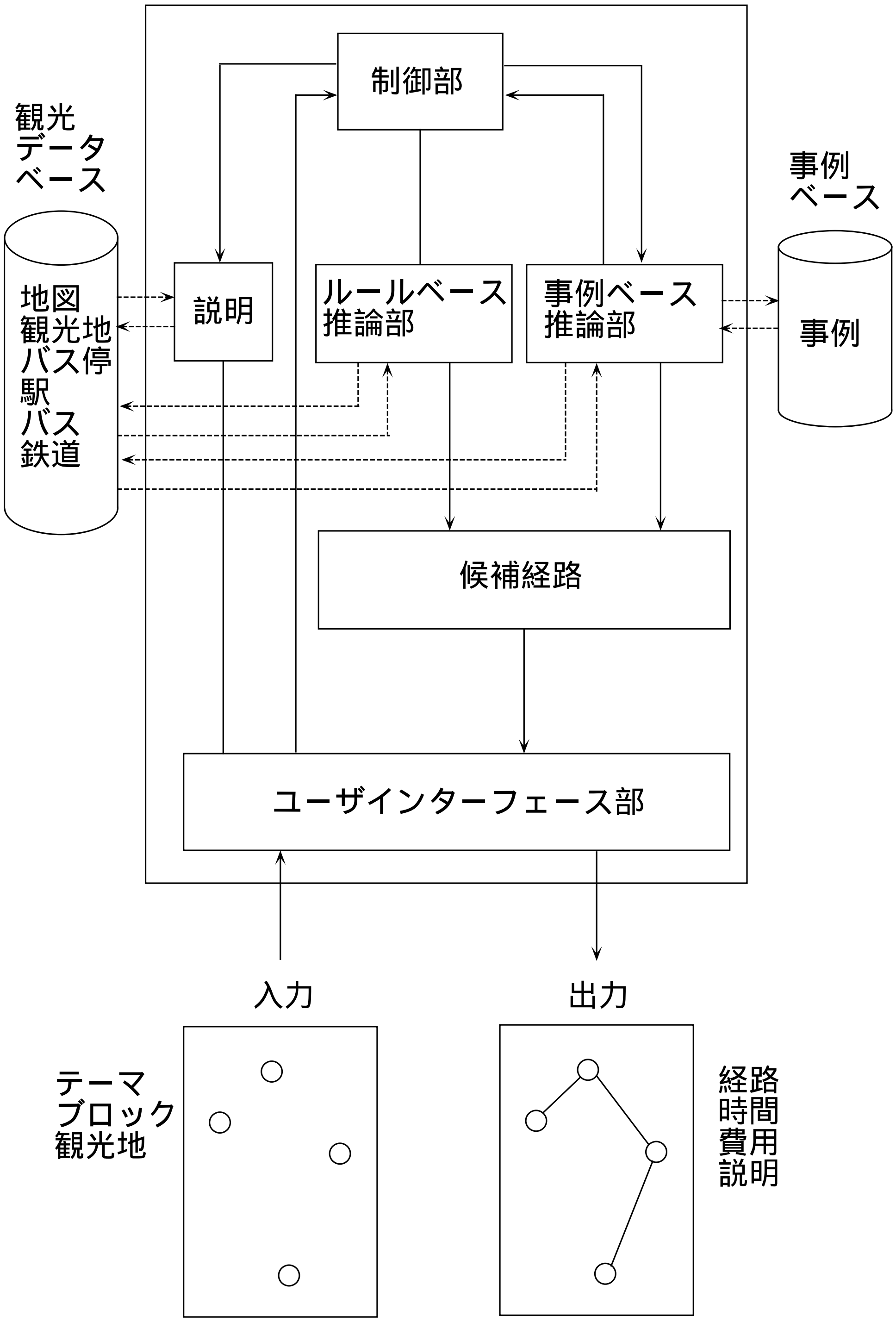


図 1: 既存のシステム構成

3　並列マシン上でのメモリベース推論による経路生成

2 章では，これまでに研究を進めてきた事例ベース推論による観光経路決定支援システムの概要について述べた．この章では，これまでのシステムの課題となった処理時間の短縮を解決するために，メモリベース推論の概念を基に作成した本システムについて述べる．

本システムは，高並列マシン AP1000 上で，大規模事例ベースから類似事例検索を並列処理で行う．この効果としては，事例数の増加により極めて類似した事例検索を可能にし，事例修正の処理負担を和らげ，事例検索を並列化することによって高速に事例検索を行うことが考えられる．

また事例は，全てブロック毎 (洛中, 洛東, 洛北, 洛西, 洛南) の事例を扱い，複数事例の概念による事例検索，修正を行う．この理由としては，全ブロックの事例を考えた場合，考えられる事例の組合せは 10^{24} 通りにも到り，各ブロックに分割することによって，事例の組合せ数を減らすことができるからである．また、さらに事例の組合せ数が減るようにブロックを 2 つのサブブロックに分割した場合についても考える。

ここから，本システムで使用した並列マシン AP1000 のアーキテクチャ，システム構成，経路生成における各処理について述べていく．

3.1　並列マシン AP1000

並列マシン AP1000 の構成を図 2に示す。[3] AP1000 は富士通研究所で開発され、メッセージ通信を基本とした分散メモリ型の並列計算機である。2に AP1000 の構成を示す。プロセッシングエレメント（セル）は、１６台から１０２４台まで接続可能である。AP1000 は、三種類のネットワーク T-Net、B-Net、S-Net を持ち、それぞれ以下の通信を高速に行うネットワークである。

- T-Net(Torus-Net)　セル間の１対１通信
- B-Net(Broadcast-Net)　ホストとセル間、あるいはセル間の放送、分配、収集
- S-Net(Synchronized-Net)　バリア同期．論理積を得るステータス.

さらに、メッセージの送信あるいは受信におけるメッセージハンドリングのオーバーヘッドを削減するために、各セルにはメッセージコントローラ（MSC）が搭載されている。

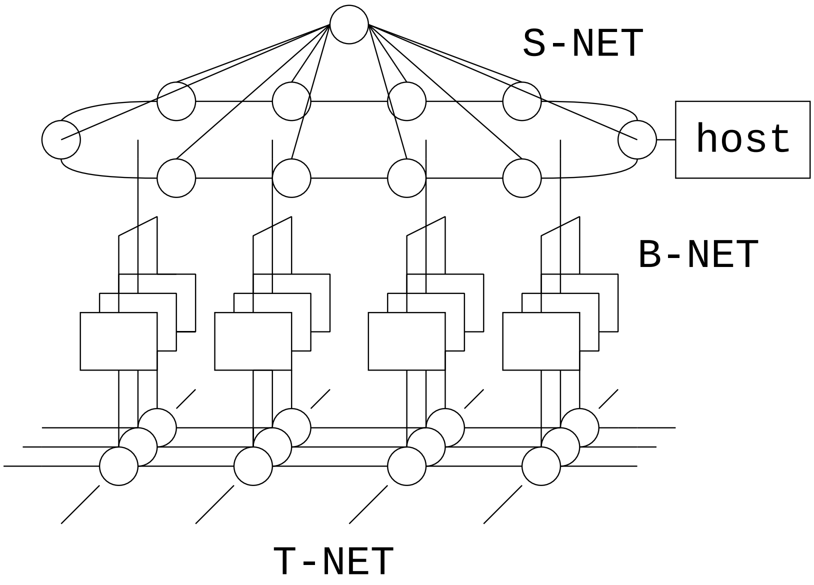


図 2: AP1000 の構成

3.2　システム構成

本システムは，従来の観光経路決定支援システムと同様に，事例ベース推論部，ルールベース推論部から成る．システムの構成を図3に、システム全体の流れを図4に示す．各部について以下に簡単に説明する。

ホスト部 大規模事例ベースの事例を B-Net を使って各セルに分散していく。セルから送られてきた類似事例に修正が必要ならば修正を加え、ブロック間の交通手段を作成しこれを解とする。

類似事例が見つからなかった時は RBR 部を起動させ解をもとめる。

セル部 ホストから送られてきた事例で事例ベースを構成する。各セルで類似事例を検索し、次に B-Net を使って各セルでの全候補事例を比較し、最も相互含有率の高いものを類似事例としてホストに送る。

RBR 部 RBR 部は経路作成部と交通手段作成部に分かれる。経路作成部では、観光地の座標を基に直線距離が最短になるように経路を作成し、交通手段作成部では各観光地間の交通手段を求めて解とする。

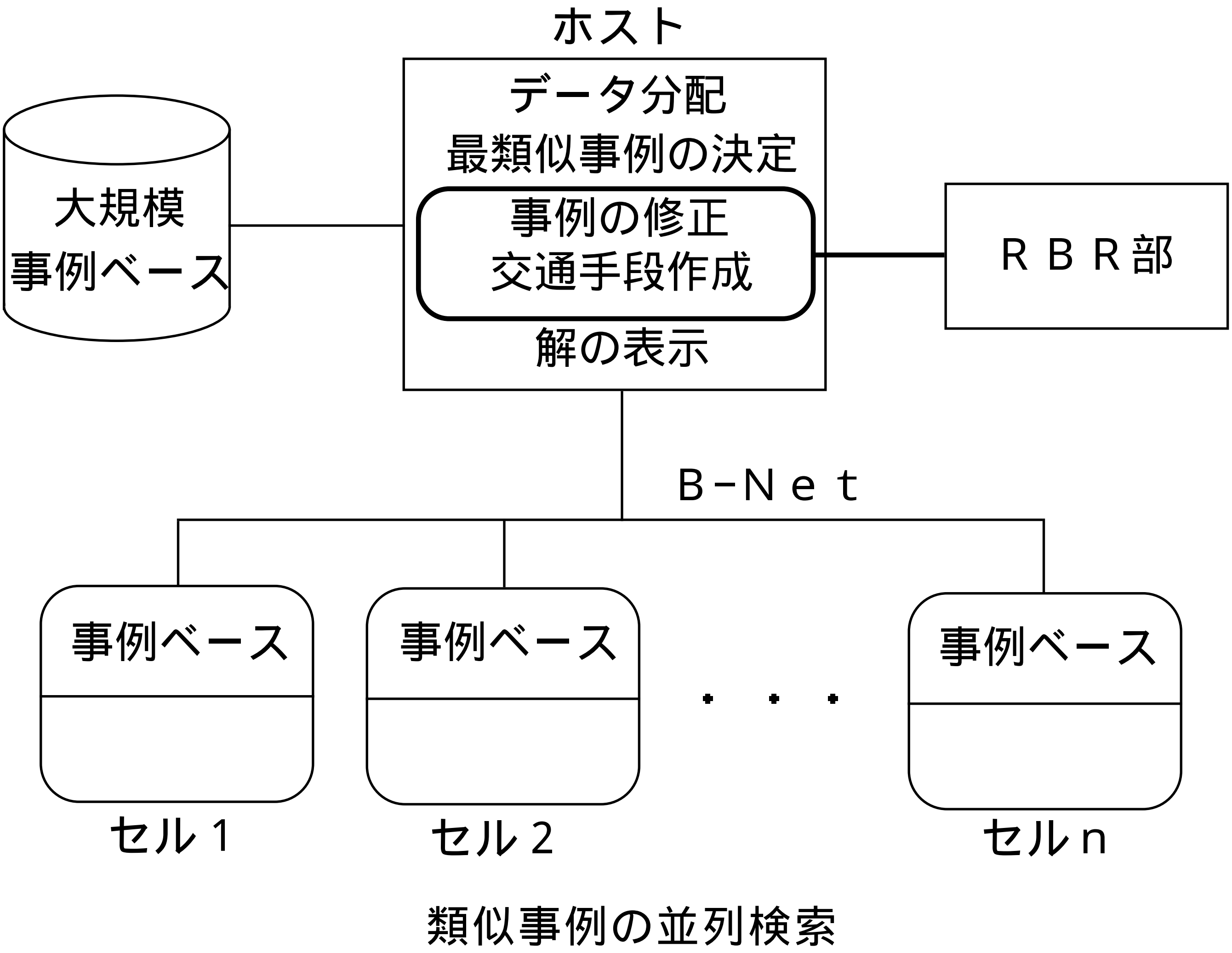


図 3: 本研究におけるシステム構成

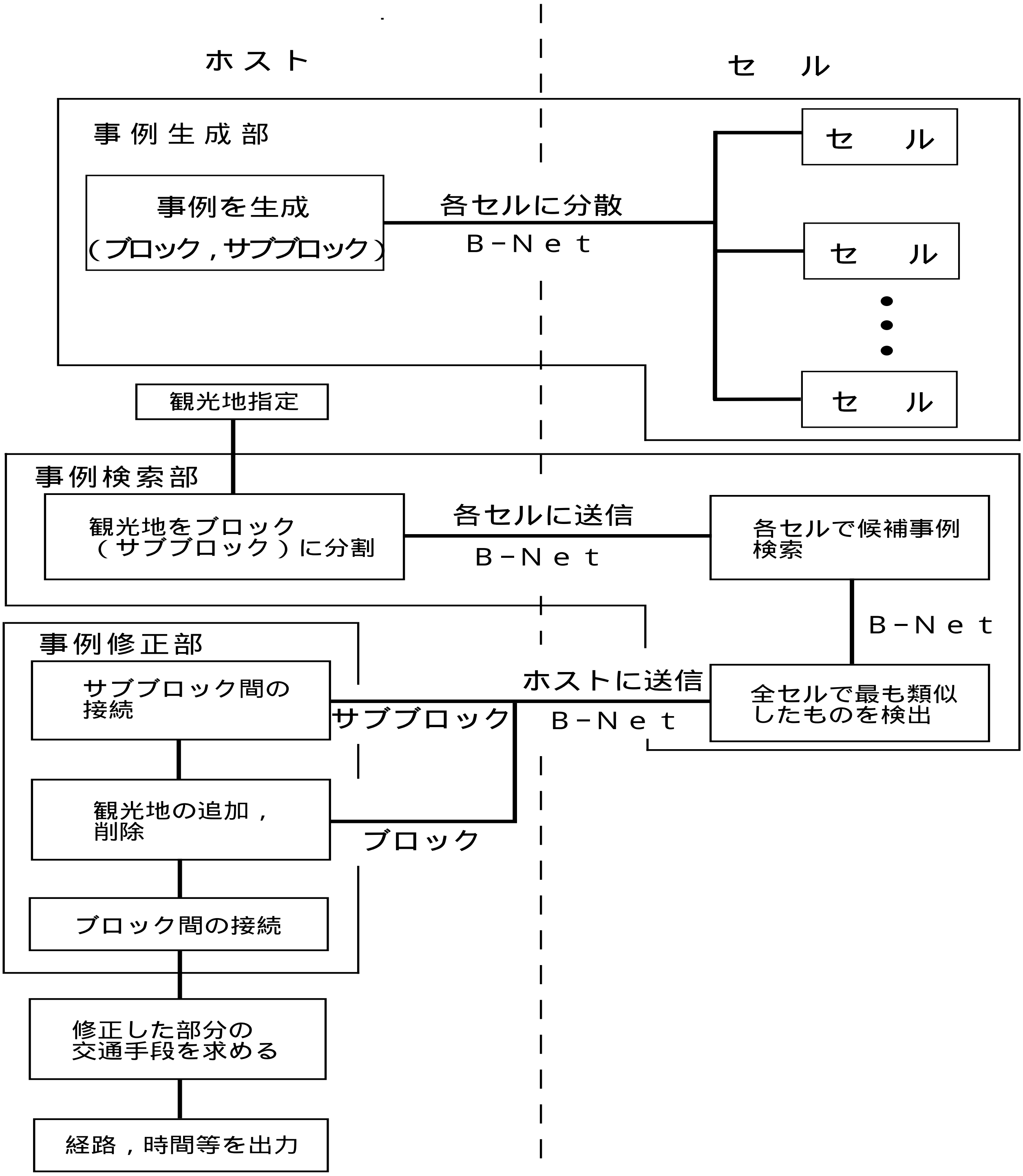


図 4: システム全体の流れ

3.3 事例生成部

事例生成部は，AP1000 上に大規模事例ベースを実現するために，事例を生成し，生成した事例を AP1000 の各セルに分散するという役割を果たす．以下に事例生成部の処理手順を示す．

1. まず乱数を使って観光地コードを発生させる．ここで発生させる観光地は全てブロック毎に，また事例数の各ブロックの長さ毎の振り分けは，観光地の組合せ (Combination) を基に，それに比例した数で行う．各ブロックの組合せは，正規分布の形になっており，最大数の半分の長さの事例が最も多く生成される．表 1 に各ブロックの観光地の組合せ数を示す．洛東の組合せ数が最も多く，最も少ない洛北の約 4000 倍にもなっている．

表 1: 各ブロックの組合せ数

ブロック	事例の長さの範囲	事例の組合せ数
洛中	2 ~ 17	131054
洛東	2 ~ 23	8388584
洛北	2 ~ 11	2036
洛西	2 ~ 22	4194281
洛南	2 ~ 13	8178
	計	12724133

2. 次に発生した各観光地の交通手段，移動時間を全て求め，移動時間が短い順にソートする．そして移動時間が最短になるように観光地を結んでいき経路を求める．ここで求めた経路が 1 つの事例となる．

   各観光地間の交通手段は，観光データベースを基に最短となるものを選び，移動時間はその交通手段と距離から求める．移動時間の計算方法は，観光地から最寄りのバス停 (駅) までの徒歩時間，バス (鉄道) の移動時間，待ち時間の合計を取る．ここで，徒歩は 4km/h，バスは 16km/h，鉄道は 21km/h(京福)，30km/h(地下鉄，京阪)，待ち時間は 10 分で計算する．

   この経路生成法が，ルールベース推論と異なる点は，距離ではなく交通手段を求めた上での移動時間が最短になるようにしているところである．

3. ここで生成した事例をファイルに書き込む．事例数が多くなる場合は分割して，複数のファイルに事例を書き込んでいく．

4. AP1000 上において，ファイルに一旦書き込まれた事例をホストで読み込み，B-Net を使って，ホストから各セルに順番に分散していく．事例の分散は各セルにおいて，各ブロックの長さ毎に事例が満遍なくいき渡るように行う．

3.4　事例検索部

事例検索部は，指定観光地を基にAP1000の各セル上で，並列処理で類似事例検索を行う．

事例検索部の処理手順としては，まず指定された観光地を，5つあるブロック毎に分割する．そして各ブロック毎の指定観光地をB-Netを使って，ホストから各セルへ送信する．この時，ブロックで指定された観光地が1以下ならば，経路として成り立たないため，そのブロックの類似事例検索は行わない．この場合は，事例検索後，事例修正部で他のブロックとの接続を行うのみとなる．

各セルは，送信された指定観光地を基に類似事例検索を行う．検索方法としては，指定観光地と事例観光地の一致を調べ，双方に対する含有率が共に75%以上の類似基準を満たし，その中で最も高い相互含有率を持つものを各セルの候補事例とする．ここで含有率は，事例に含まれる指定観光地数のそれぞれの観光地数に対する割合，そして相互含有率とは双方の含有率の平均を意味する．(1)式に相互含有率を求める式を示す．

$$S = \frac{s_n/f_n + s_n/c_n}{2} \tag{1}$$

S:相互含有率(%)

s_n:事例に含まれる指定観光地数

f_n:指定観光地数

c_n:事例観光地数

従来のシステムでは，事例観光地に対する含有率だけを類似基準としていたが，この相互含有率を用いることにより，指定観光地との共通部分の密度が増し，事例を修正する可能性をいっそう減らす効果に繋がる．

次に，B-Netを使って各セルでの全候補事例を比較し，最も相互含有率が高いものを類似事例として，その事例データをセルからホストへ送信する．

3.5　事例修正部

事例修正部は，事例検索部で検索した各ブロックの類似事例の修正を行い，ブロック間の接続を行う．

まず，各ブロックの類似事例において，事例にない指定観光地を追加し，指定されていない観光地を事例から削除する．ここで，指定観光地の追加は，事例の経路中に挿入した場合の距離が最も最短になる箇所に行う．各ブロックの事例の相互含有率が100%である場合はこの処理を行う必要はなく，その形のまま使用する．

また，4章で述べるサブブロックの事例を用いる場合は，まずサブブロックの事例を接続し，ブロック単位にしてからこの事例修正を行う．

次に，各ブロックの事例を距離が最短になるように接続していく．方法としては，各ブロックの事例の両端同士の距離を全て求め，距離が短い順に求めた辺をソートする．後は，

距離が短い順に全ブロックが1本の経路となるように辺を選んでいく．ここで，距離の短い辺を選ぶにあたり，途中で閉路にならないようにする．

この処理によって，指定された全観光地は経路の形となり，交通手段作成部で修正を行った部分に関して，観光データベースを基に交通手段を求めて最終的な解とする．

ここで注意する点は，たとえ各ブロックの類似事例の相互含有率が全て100%であったとしても，少なくともブロック間の接続を行うので，完全に事例修正を行わないのは，1ブロックのみで観光地が指定された場合に限られる．

4 大規模事例ベースの作成

メモリベース推論を実現するために大規模な事例ベースを用意する。
ここでは事例ベースの構成及び生成方法、並列計算機の各プロセッサへの分散方法について述べる。

4.1 事例ベースの構成

大規模事例ベースは、ブロック、サブブロックごとに大量の事例を作成することによって構成される。ここでサブブロックとは、事例の組合せ数をブロック毎の場合よりもさらに少なくするために各ブロック内の観光地を交通的に無駄のないように、さらに２つに分割したものである。これにより、サブブロックにおける事例の最大組合せ数はブロックに比べ約１／８００となる。表2にブロックとサブブロックの振り分けの例を、表3、4にブロックとサブブロックにおける事例の例を示す。

事例を生成する際、乱数を使って観光地を発生させている。この時に観光地の重複が起こらないように新しく発生した観光地がすでに登録されていないかをチェックし、登録されていなければ新しい観光地として記憶する。これを指定された長さの事例になるまでくりかえす。各ブロック、サブブロックにおいて以上のことをくりかえし行ない大量の事例を生成する。ブロック、サブブロックの事例は、それぞれの範囲で乱数を発生させて作り出している。

事例は各ブロック、サブブロックで長さごと（2～そのブロック、サブブロックの観光地数）に作っていく。このときすべての事例を作成することはできないので、各長さ毎の最大事例数の、各ブロックの最大事例数に対する比率分だけ数を割り振って事例を作っている。以下に割り振る事例数の計算方法を示す。

$$bl_{\mathrm{num}} = \frac{b_{\mathrm{num}}}{b_{\mathrm{max}}} \times bl_{\mathrm{max}} + 1$$

bl_{num}: 各ブロック (サブブロック) の長さ毎に振り分けられる事例数

bl_{max}: 各ブロック (サブブロック) の長さ毎の最大事例数

b_{num} : 各ブロック (サブブロック) に振り分けられる事例数

b_{max} : 各ブロック (サブブロック) の最大事例数

表 2: 観光地の振り分け

洛中ブロック	
下鴨神社	北野天満宮
大報恩寺	西陣織会館
茶道総合資料館	楽美術館
承天閣美術館	御所
二条城	神泉苑
二条陣屋	六角堂
西本願寺	古代友禅苑
渉成苑	東寺
観智院	

洛中ブロックのサブブロック	
サブブロック a	サブブロック b
下鴨神社	二条城
北野天満宮	神泉苑
大報恩寺	二条陣屋
西陣織会館	六角堂
茶道総合資料館	西本願寺
楽美術館	古代友禅苑
御所	渉成苑
承天閣美術館	東寺
	観智院

表 3: 洛中ブロックの事例の表現

	大報恩寺	——	西陣織会館	——	二条城	——	東寺
（テーマ）	絵		伝統工芸		庭建絵		建仏
（ブロック）	中		中		中		中
（交通手段）		５９		９		地下鉄⇒２０２	
（移動時間）		２３分		２３分		４５分	

表 4: 洛中のサブブロックの事例の表現

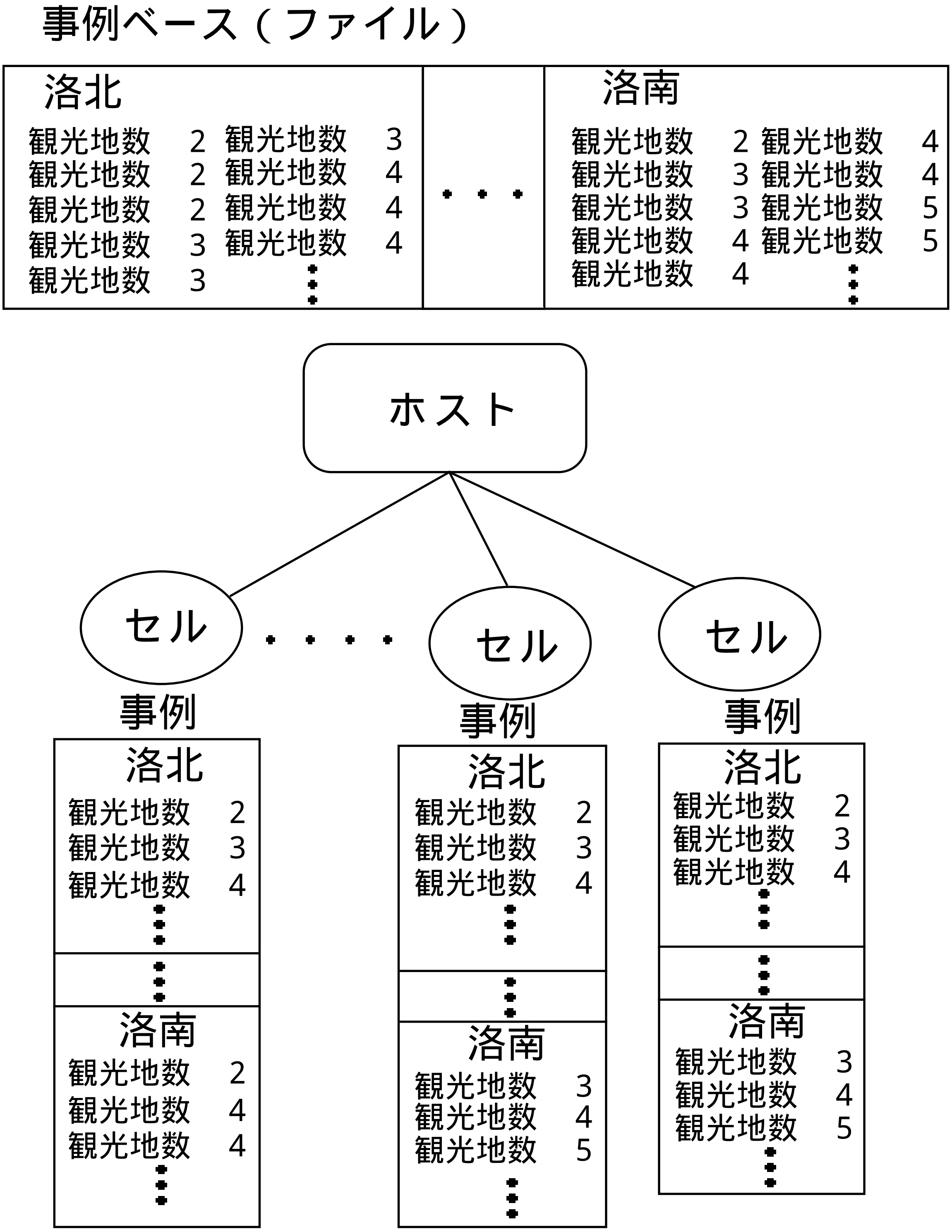
	大報恩寺	——	西陣織会館		東寺	——	二条城
（テーマ）	絵		伝統工芸		建仏		庭建絵
（ブロック）	中		中		中		中
（交通手段）		５９				２０２⇒地下鉄	
（移動時間）		２３分				４５分	

4.2 事例の生成と各プロセッサへの分散

図 5に事例の各プロセッサへの分散方法を示す。

メモリベース推論を実現するために大規模な事例ベース（最大事例数１６００００個、約７０ MB）を作成する。３章に述べた事例生成部で作成するが、生成時間がかなり大きくなるため（１６００００個で約９時間）１台のワークステーション上で事例を生成し、AP1000 上での時間的負荷を減らす。できた事例は複数のファイルに格納する。ブロック、観光地数、及び生成する事例の数は実験段階ごとに変えていく。この時、AP1000 上の各プロセッサに分散し易いように事例を長さ（観光地数）毎に並べておく。

生成された事例ファイルを AP1000 に取り込み、B-Net を使い各プロセッサに事例を分散する。各プロセッサ毎の当該事例の存在確率、検索にかかる時間的負荷をほぼ均等にするため、事例の長さを考慮しなければならない。即ち、同じ観光地数の事例がそれぞれのプロセッサに同数存在するように配置する。生成時に長さ毎に並べているので、１事例ずつ順番に取り込んでいけば良い。各プロセッサへ事例を分散し終わると、検索へ移る。



事例ベース（ファイル）
洛北
観光地数 2
観光地数 2
観光地数 2
観光地数 3
観光地数 3
観光地数 3
観光地数 4
観光地数 4
観光地数 4
洛南
観光地数 2
観光地数 3
観光地数 3
観光地数 4
観光地数 4
観光地数 4
観光地数 4
観光地数 5
観光地数 5
ホスト
セル
セル
セル
事例
洛北
観光地数 2
観光地数 3
観光地数 4
洛南
観光地数 2
観光地数 4
観光地数 4
事例
洛北
観光地数 2
観光地数 3
観光地数 4
洛南
観光地数 3
観光地数 4
観光地数 5
事例
洛北
観光地数 2
観光地数 3
観光地数 4
洛南
観光地数 3
観光地数 4
観光地数 5

図 5: 事例の分散

5　実験と考察

実験を行うにあたり、まずは少数事例で検索・結果の正確性をチェックする為に、１ブロックだけで実験を行う。その後、少しずつ事例を増やして行き、サブブロックを使用した実験を行う。

5.1　実験方法

実験の手順を以下に示す。

1. まず、洛北ブロックにおいて、事例数１６０、８００、１６００個で事例検索を行う。また、プロセッサ数を４、８、１６台、指定観光地数を４、５、６、７個と変化させて実験を行う。テスト回数は２０回とする。評価としては、セル台数ごとの実行時間、事例数の増加に対する事例検索率について行う。この実験は後に行う実験の予備段階と考えてよい。

2. 洛北、洛南の２ブロックにおいて、事例数１６００、４８００、８０００個、指定観光地数を８、１２、１６、２０個とし、１と同様に実験を行う。

3. 洛中、洛北、洛南の３ブロックを併合し、事例数１６０００、４８０００、８００００個でそれぞれ実験を行う。セル台数は８、１６、３２台、指定観光地数は１２、１６、２０、２４、２８個で行う。テスト回数は２０回とし、事例検索時間、事例検索率、平均相互含有率に加え、事例数毎の事例修正時間、交通手段作成時間、検索・修正された解の質（移動時間、直線距離）の計測を行う。また、RBR による処理時間（経路生成時間、交通手段作成時間）、生成された解の質の計測を行い、両者の比較を行う。

4. 洛西、洛東の２ブロックを併合し、事例数４８０００、８００００、１６００００個で実験を行う。プロセッサ数は１６、３２、６４台、指定観光地数は１６、２４、３２個で行い、テスト回数は５０回とする。ここでは、事例検索時間、事例検索率、平均相互含有率の計測を行う。

5. 全ブロックにおいて事例数４８０００、８００００、１６００００個で実験を行う。プロセッサ数は１６、３２、６４台、指定観光地数は２０、４０、６０、８０個で行う。この実験により、プロセッサ数に対する実行時間、事例数の増加に対する事例検索率、事例修正によって結合された解（全ブロックにわたる）の評価を行っていく。解の評価の際に、従来行ってきたルールベース推論との解の比較も行なう。

6. 最後に、全ブロックにおいてサブブロックを使用した実験を行なう。事例数１６００、３２００、４８００で、プロセッサ数は４、８、１６台、指定観光地数は２０、４０、６０、８０個とし、５と同様の実験を行なう。

5.2 実験結果

5.2.1 事例検索時間

実験 5.1.1、5.1.2、5.1.3、5.1.4、5.1.5 による平均事例検索時間をそれぞれ表 5 ~ 表 9に示す。サブブロックによる実験の平均事例検索時間を表 10、11、12に示す。また実験 5.1.1、5.1.2 の相対速度を図 6に、実験 5.1.3、5.1.5 の相対速度をそれぞれ図 7、図 8に示す。各図とも横軸にセル台数を、縦軸に相対速度をとっている。そしてブロックとサブブロックの事例検索時間の比較を図 9に示す。図 9は横軸に指定観光地数、縦軸に平均事例検索時間をとっている。各表、各図の相対速度は，同一の事例ベースにおいて，最小のセル台数による事例検索時間を 1 とした場合のものである．

各表より，セル台数の増加に対して事例検索時間が短縮されていることが分かる．表 5 の事例数が少ない場合は，並列効率は余り良くなく，事例数 1600 において，セル台数 4 台から 16 台に対する相対速度は 1.51 倍程度である．しかし，事例数の増加により事例検索時間も増大していき，最大で，表 8の事例数 160000 においてセル台数 16 台から 64 台に対し，相対速度は 3.68 倍になっている．

表 5: 実験 5.1.1 での平均事例検索時間

事例数	セル台数	事例検索時間 (s)	相対速度
160	*4*	0.0338	1
	8	0.0324	1.04
	16	0.0317	1.07
800	*4*	0.0441	1
	8	0.0381	1.16
	16	0.0349	1.26
1600	*4*	0.0578	1
	8	0.0440	1.31
	16	0.0382	1.51

表 6: 実験 5.1.2 での平均事例検索時間

事例数	セル台数	事例検索時間 (s)	相対速度
1600	*4*	0.0838	1
	8	0.0685	1.22
	16	0.0608	1.38
4800	*4*	0.1449	1
	8	0.0983	1.47
	16	0.0765	1.89
8000	*4*	0.2089	1
	8	0.1305	1.60
	16	0.0919	2.27

表 7: 実験 5.1.3 での平均事例検索時間

事例数	セル台数	事例検索時間 (s)	相対速度
16000	*8*	0.3707	1
	16	0.2278	1.63
	32	0.1529	2.42
48000	*8*	0.8649	1
	16	0.5006	1.73
	32	0.2991	2.89
80000	*8*	1.5437	1
	16	0.8723	1.77
	32	0.4970	3.11

表 8: 実験 5.1.4 での平均事例検索時間

事例数	セル台数	事例検索時間 (s)	速度向上比
48000	*16*	0.9150	1
	32	0.5135	1.78
	64	0.2700	3.39
80000	*16*	1.3904	1
	32	0.7742	1.80
	64	0.4659	3.53
160000	*16*	2.398	1
	32	1.224	1.959
	64	0.6520	3.68

表 9: 実験 5.1.5 での平均事例検索時間

事例数	セル台数	事例検索時間 (s)	速度向上比
48000	*16*	0.5892	1
	32	0.3552	1.66
	64	0.2363	2.49
80000	*16*	0.9432	1
	32	0.5145	1.83
	64	0.3050	3.09
160000	*16*	1.6908	1
	32	0.8865	1.90
	64	0.4951	3.42

サブブロックでの実験による平均事例検索時間を以下に示す。

表10: 事例数1600での平均事例検索時間

観光地数	セル台数	事例検索時間(s)	速度向上比
20	*4*	0.1420	1
	8	0.1215	1.17
	16	0.1182	1.20
40	*4*	0.3080	1
	8	0.2251	1.37
	16	0.2149	1.43
60	*4*	0.5048	1
	8	0.2722	1.85
	16	0.2539	1.99
80	*4*	0.2997	1
	8	0.2620	1.14
	16	0.2480	1.21

表11: 事例数3200での平均事例検索時間

観光地数	セル台数	事例検索時間(s)	速度向上比
20	*4*	0.2674	1
	8	0.1833	1.46
	16	0.1290	2.07
40	*4*	0.3738	1
	8	0.3326	1.12
	16	0.2938	1.27
60	*4*	0.3901	1
	8	0.3389	1.15
	16	0.2734	1.43
80	*4*	0.4458	1
	8	0.3944	1.13
	16	0.3736	1.19

表 12: 事例数 4800 での平均事例検索時間

観光地数	セル台数	事例検索時間 (s)	速度向上比
20	*4*	0.3201	1
	8	0.2497	1.28
	16	0.1413	2.27
40	*4*	0.4401	1
	8	0.2794	1.58
	16	0.2452	1.80
60	*4*	0.5294	1
	8	0.3367	1.57
	16	0.2903	1.82
80	*4*	0.4226	1
	8	0.3240	1.31
	16	0.2708	1.56

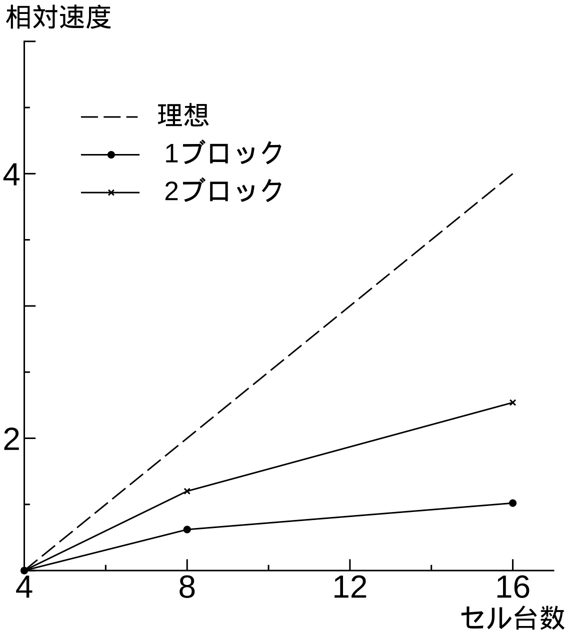


図 6: 実験 5.1.1、5.1.2 における検索時間の相対速度

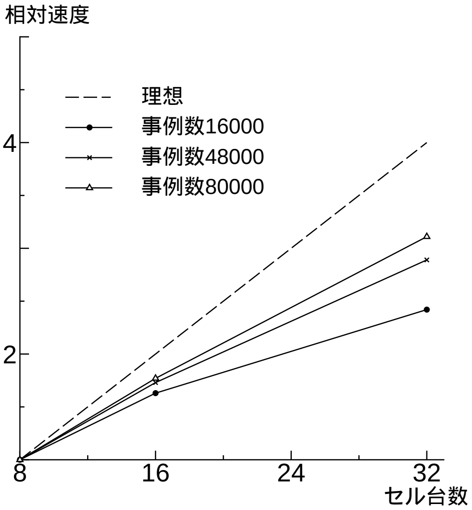


図 7: 実験 5.1.3 における検索時間の相対速度

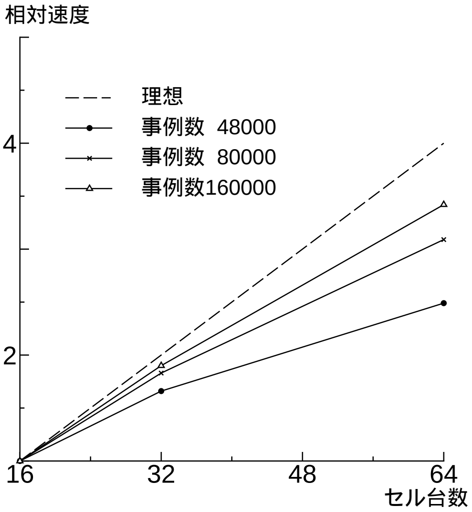


図 8: 実験 5.1.5 における検索時間の相対速度

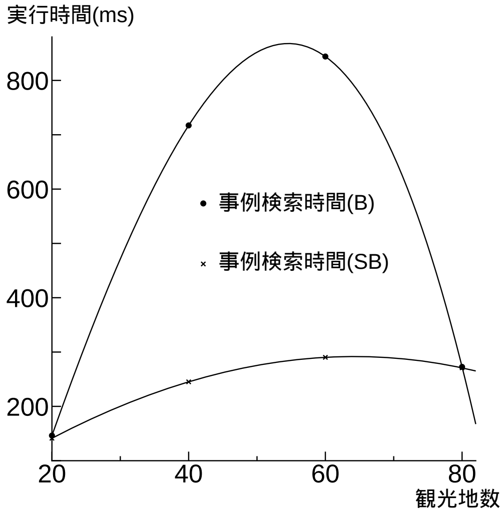


図 9: ブロックとサブブロックの事例検索時間の比較

5.2.2　事例検索率と平均含有率

各実験による事例検索率と平均相互含有率をそれぞれ表 13 ~ 表 18にしめす。各表の事例検索率は，乱数による指定観光地入力のテスト回数に対し，各ブロックの類似事例検索が成功した割合を示し，平均相互含有率は，検索された類似事例の相互含有率の平均を示す．

各表より，各々の実験において，事例数が増加するほど，事例検索率，平均相互含有率共に向上していることが分かる．

表 13: 実験 5.1.1 での事例検索率と平均含有率

事例数	事例検索率 (%)	平均含有率 (%)
160	93.8	86.66
800	100.0	94.26
1600	100.0	98.11

表 14: 実験 5.1.2 での事例検索率と平均含有率

事例数	事例検索率 (%)	平均含有率 (%)
1600	86.3	91.43
4800	93.8	96.25
8000	97.5	98.26

表 15: 実験 5.1.3 での事例検索率と平均含有率

事例数	事例検索率 (%)	平均含有率 (%)
16000	86.0	91.05
48000	93.0	94.24
80000	95.0	96.91

表 16: 実験 5.1.4 での事例検索率と平均含有率

事例数	事例検索率 (%)	平均含有率 (%)
48000	63.2	85.27
80000	70.7	86.01
160000	83.3	88.60

表 17: 実験 5.1.5 での事例検索率と平均含有率

事例数	事例検索率 (%)	平均含有率 (%)
48000	36.9	89.72
80000	39.6	89.20
160000	42.8	89.64

表 18: 実験 5.1.6 での事例検索率と平均含有率

事例数	事例検索率 (%)	平均含有率 (%)
1600	28.9	92.2
3200	45.9	93.2
4800	52.4	95.4

5.2.3 解の質 (移動時間，直線距離)、実行時間

実験 5.1.3、5.1.5、5.1.6 における解の質をそれぞれ表 19、20、21に示し、解の質の比較を図 12、13に示す。各表は，テストをして類似事例検索が成功した場合において，MBR と RBR の両推論で求めた解の質を比較して，MBR の方が RBR よりも優れている割合を示している．

各表より MBR と RBR の解の質を比較すると，移動時間では MBR の解の方が RBR の解より優れている場合が多く，直線距離では RBR の方が MBR より優れている場合が多くなっている．さらに各表を比較して，事例数の増大に対し，更に移動時間の観点で MBR が RBR より優れている割合が増えてきている．

また実験 5.1.3、5.1.5、5.1.6 において、１つの解を生成するための平均実行時間をそれぞれ表 22、23、24に示し、事例修正時間と交通手段作成時間を図 10、11に示す。両図とも横軸に指定観光地数を、縦軸に実行時間をとっている。

ここで移動時間の比較は

$$\frac{MBR\text{が}\ RBR\text{より優れていた回数}}{\text{検索数}}$$

を表している。

表 19: 実験 5.1.3(事例 80000、32 台) における MBR と RBR の解の質の比較

観光地数	平均移動時間 (分)			平均直線距離 (Km)			移動時間の比較
	MBR	RBR	$\frac{MBR}{RBR}$	MBR	RBR	$\frac{MBR}{RBR}$	
12	404.53	438.95	0.92	35.06	30.97	1.13	15/18
16	504.85	554.69	0.91	43.54	37.66	1.16	18/19
20	624.96	674.94	0.93	53.45	44.67	1.20	16/18
24	718.51	770.36	0.93	60.23	48.78	1.23	19/20
28	798.89	852.38	0.94	67.48	52.82	1.28	16/20

表20: 実験5.1.5(事例数160000、64台)におけるMBRとRBRの解の質の比較

観光地数	平均移動時間(分)			平均直線距離(Km)			移動時間の比較
	MBR	RBR	$\frac{MBR}{RBR}$	MBR	RBR	$\frac{MBR}{RBR}$	
20	621.51	618.92	1.01	52.06	49.14	1.06	5/12
40	1067.80	1060.19	1.01	84.43	72.14	1.17	36/83
60	1371.20	1378.25	0.99	102.88	84.26	1.22	19/30
80	1515.36	1630.72	0.93	112.35	91.77	1.22	10/10

サブブロックによる実験(実験5.1.6)による、解の比較を以下に示す。

表21: MBRとRBRの解の比較

事例数	観光地数	検索数	平均移動時間(分)			平均直線距離(Km)			移動時間の比較
			MBR	RBR	$\frac{MBR}{RBR}$	MBR	RBR	$\frac{MBR}{RBR}$	
1600	*20*	5/500	645.92	613.33	1.05	53.36	47.45	1.12	1/5
	40	8/311	1054.28	1022.98	1.03	76.55	65.29	1.17	3/8
	60	8/23	1445.93	1374.26	1.05	103.26	81.45	1.27	2/8
	80	8/8	1577.48	1596.02	0.98	111.64	90.56	1.23	7/8
3200	*20*	8/227	657.35	625.85	1.05	54.85	49.63	1.11	2/8
	40	8/43	1107.23	1060.86	1.02	80.09	68.79	1.16	1/8
	60	8/13	1415.19	1387.65	1.02	99.81	83.35	1.20	3/8
	80	8/8	1586.39	1596.02	0.99	108.15	90.56	1.19	6/8
4800	*20*	8/83	619.20	606.90	1.02	48.78	44.95	1.09	3/8
	40	8/40	1135.06	1085.73	1.05	82.99	69.34	1.20	1/8
	60	8/10	1403.15	1381.81	1.02	98.17	82.49	1.19	4/8
	80	8/8	1583.81	1596.02	0.99	109.90	90.56	1.15	7/8

表22: 実験5.1.3(事例数80000、32台)の平均実行時間

観光地数	MBR				RBR		
	事例検索時間（s）	事例修正時間（s）	交通手段作成時間	計	経路作成時間（s）	交通手段作成時間	計
12	0.3142	0.0028	0.0041	0.3211	0.0169	0.0057	0.0226
16	0.3085	0.0045	0.0051	0.3181	0.0497	0.0073	0.0570
20	0.5984	0.0035	0.0024	0.6043	0.1175	0.0089	0.1264
24	0.6393	0.0043	0.0060	0.6496	0.2923	0.0210	0.3133
28	0.6248	0.0042	0.0028	0.6318	0.6557	0.0360	0.6917

表23: 実験5.1.5(事例160000、64台)の平均実行時間

観光地数	MBR				RBR		
	事例検索時間（s）	事例修正時間（s）	交通手段作成時間	計	経路作成時間（s）	交通手段作成時間	計
20	0.1463	0.0109	0.0053	0.1625	0.1123	0.0076	0.1199
40	0.7172	0,0121	0.0067	0.7360	1.6693	0.0122	1.6815
60	0.8440	0.0156	0.0068	0.8664	8.1129	0.0155	5.1284
80	0.2727	0.0137	0.0083	0.2947	25.3830	0.0192	25.4022

表24: 実験5.1.6(事例数4800、16台)の平均実行時間

観光地数	MBR				RBR		
	事例検索時間（s）	事例修正時間（s）	交通手段作成時間	計	経路作成時間（s）	交通手段作成時間	計
20	0.1413	0.0096	0.0056	0.1565	0.1131	0.0070	0.1201
40	0.2452	0.0120	0.0071	0.2643	1.6990	0.1105	1.6801
60	0.2903	0.0143	0.0076	0.3122	8.5365	0.0155	8.5520
80	0.2709	0.0159	0.0064	0.2932	25.7776	0.0178	25.9556

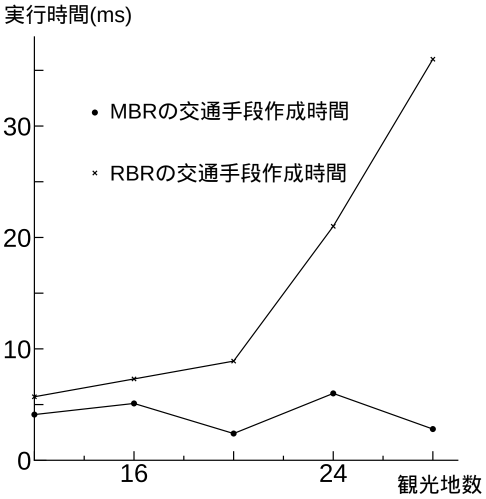


図 10: MBR と RBR の交通手段作成時間の比較 (3 ブロック)

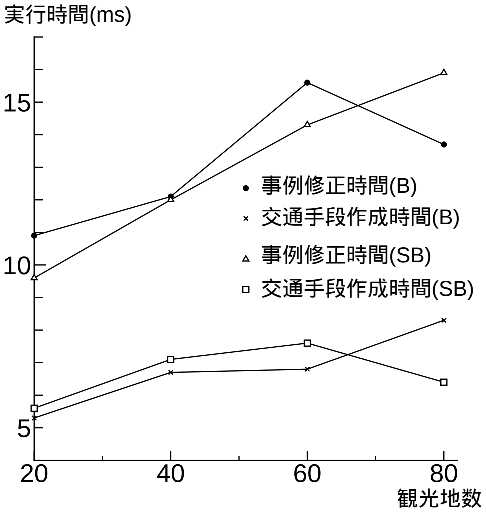


図 11: ブロックとサブブロックの事例修正時間、交通手段作成時間の比較

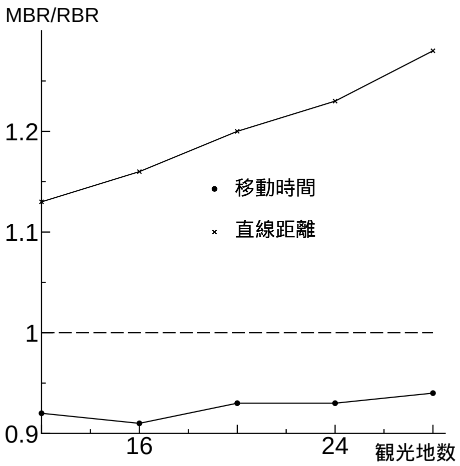


図 12: 実験 5.1.3 における解の質

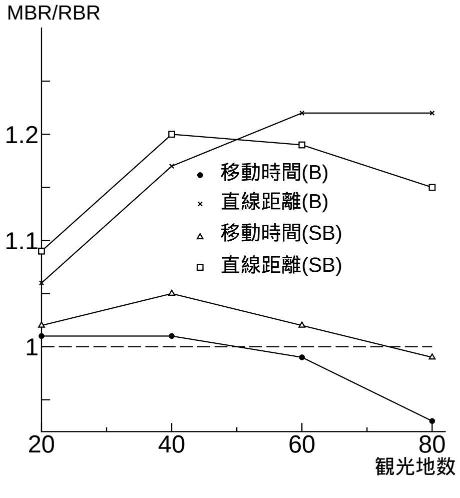


図 13: ブロックとサブブロックの解の質の比較

5.3 考察

5.3.1 事例検索時間

5.2.1の表5から表12より、各表ともセル台数の増加に従い事例検索時間が短縮されている。各実験の相対速度をみると、セル台数が4倍になるに従い最小で1.07倍、3.68倍の速度向上が認められた。

図6で1ブロックでの速度向上比と2ブロックでの速度向上比を比較すると、2ブロックでの向上比の方が良い結果となっている。また図7、図8と比較しても速度向上比はブロック数が多いほど良い結果が得られることがわかる。

図9はブロックとサブブロックの平均事例検索時間を比較したものである。この図を見てまずわかることは、ブロック、サブブロックの相方において指定観光地数が４０から６０の範囲で事例検索時間が最も大きくなっていることである。これは観光地数が全部で８６個あり、その組合せの数が５０前後で最大となるためである。またサブブロックの方が曲線が低くなだらかになっているのは、ブロックよりも事例の組合せの数が非常に少ない(ブロックに比べ約１／８００)ためである。このことからサブブロックに分割することによって事例の組合せ数を減らし、事例検索時間を削減するといった試みは成功したものと思われる。

5.3.2 事例検索率と平均相互含有率

5.2.2の表13から表18より、事例数が増加するに従い事例検索率、平均相互含有率はともに向上していることがわかる。これは事例数が増加すればそれに応じて当該事例の存在確率増加するからである。そのため事例の組合せ数の少ないブロックでの実験の方が相互含有率は高くなっている。しかし、表13から表15を比べてみると事例が増加しているにもかかわらず事例検索率、平均含有率は少しずつ低下している。さらに表16、表17では大幅に低下している。これは観光地数毎の事例が正規分布の比率に基づいて作られているため、指定観光地数が少数の場合に事例の存在確率が低くなるためである。

事例の増加に従い事例検索率、平均相互含有率が向上するのは、大規模事例ベースの有効性を表しており、またサブブロックのように組合せ数を減らすことの有効性も示している。

5.3.3 解の質(移動時間、直線距離)、実行時間

表19、20、21でMBRとRBRの解の比較を行なっている。図12に実験5.1.3におけるMBRとRBRの解の比較を、図13にブロックとサブブロックの解の比較を示している。両図とも縦軸の値が１よりも上の領域ではRBRの方が優れており、１より下の領域では

MBR の方が優れていることを表している。図 12をみると移動時間の比較において MBR が RBR よりも優れている。このような結果になったのは事例は移動時間が最短になるようにつくられており、RBR では直線距離が最短になるように経路生成を行なっているためである。また図 13によって、ブロックによる解の方がサブブロックによる解よりも優れていることがわかる。

図 11にブロックとサブブロックの事例修正時間、交通手段作成時間の比較を示している。サブブロックを用いた場合、事例修正時間や交通手段作成時間がブロックに比べ、多くなるものと思われたが、この図からはブロックとサブブロックの間に大きな違いはみられない。これはサブブロックを使用することが想像以上に有効であることを示している。MBR では指定観光地数が増えても実行時間に大きな差はあらわれないが、RBR では指定観光地数に比例して実行時間が増大している。これは RBR においてはそれぞれの観光地で最短経路を作成しているためであり、MBR の有効性を強く表したものである。

5.3.4 MBR の得失

本システムの実験結果より、総合的に以下のことが挙げられる。

- 事例検索時間の評価では大規模な事例ベースに対しても並列化により有効な対処が可能であるという結論が得られた。
- サブブロックを使用して事例の組合せ数を減らすことによる事例検索時間の削減が非常に有効であることが示された。
- 事例検索率、平均相互含有率の評価では、大規模事例ベース、サブブロックによる事例の組合せ数の削減の有効性が示された。
- 解の質の評価では、MBR の概念で求めた解が解として成り立つものであるという結果が得られた。
- 実行時間の評価では、指定観光地数の増加に対し MBR の方が RBR よりも非常に有効であるという結果が得られた。

以上より、MBR の有効性を得ることができた。

本研究における問題点をあげると、まず事例ベースに容量がかかり過ぎるという点があげられる。しかしこれは、今回の研究でも行なったサブブロックによる事例の組合せ数の削減によって解決できる問題であると思われる。

また、もう１つの大きな問題として事例生成に非常に多くの時間がかかるという点があげられる。本研究においても事例生成に多大な時間を費やしてしまった。やはりこの問題も今後解決しなければならないものと思われる。

6　おわりに

本論文では、事例ベース推論による京都観光経路決定支援システムにおいて、メモリベース推論を適用するための大規模事例ベースについて述べ、更にシステム全体としてのメモリベース推論の有効性を示すため、実行時間・事例検索率・事例修正による解の評価及びルールベース推論との解の比較を行なった。評価の方法として、少数事例による実験から少しずつ規模を拡大し、最大１６００００個の事例を使い実験を行なった。

事例検索時間ではプロセッサ台数を４倍にすると３．６８の速度向上比が得られ、事例による負荷の増大にともない速度向上比も向上するという結果が得られた。
事例検索率、平均相互含有率では事例数の増加によりともに向上するという結果を得た。
またサブブロックによる事例の組合せ数の削減が非常に有効であることが示された。
解の質ではMBRは移動時間の面で、RBRは直線距離の面ですぐれているという結果が得られた。

総合的な観点から、ＭＢＲの有効性が伺える。

本研究で扱った事例は与えられたディスク容量の関係で上限があったので扱った時間が小さいものとなってしまった。しかし、得た結果から、メモリベース推論の有効性が得られたので、今後、機会があればさらに大規模な事例ベースを構築すると良い。

今後の課題として、現在の研究での実行時間はあまりにも短く比較しづらいので、大規模な巡回セールスマン問題や、ジョブショップスケジューリング問題などに応用することが考えられる。
また各観光地数毎の事例生成数を改善も求められる。

謝辞

本研究において御指導して下さいました山崎勝弘教授に感謝致します。更に本研究の共同研究者であり、多くの助言をいただきました西村高志 氏に感謝致します。また、貴重な御意見をいただきました山崎研究室の皆様に感謝の意を表します。

参考文献

[1] 松原　仁： 推論技術の観点からみた事例に基づく推論、人工知能学会誌、vol.7、No.4、pp567-575(1992).

[2] 北野　宏明： 超並列人工知能、人工知能学会誌、vol.7、No.2、pp244-262(1992).

[3] 堀江　健志、石畑　宏明、清水　俊之、池畑　守夫： 高並列計算機 AP1000 のアーキテクチャと性能評価、電子情報通信学会論文誌 D-I、Vol.J 75-D-I、pp.637-645(1992).

[4] 森田　真弘： 事例ベース推論による観光経路決定支援システム、立命館大学大学院理工学研究科修士論文 (1993).

[5] 小林　康弘： 事例ベース推論、人工知能学会誌、vol.8、No.1、pp26-36(1993).

[6] 河野　毅、末田　直道、田中　利一： CBR システムの構築環境、人工知能学会誌、vol.9、No.1、pp91-99(1994).

[7] Brian P.Kettler, James A.Hendler, William A.Andersen, and Matthew P.Evett: Massively Parallel Support for Case-Based Planning, IEEE EXPERT, pp9-14(1994).

[8] Evangelos Simoudis: Using Case-Based Retrieval for Customer Technical Support, IEEE EXPERT, pp7-11(1992).

[9] 奥田　健三、山崎　勝弘： 事例ベース推論とその応用例、情報処理、vol.31、No.2、pp.244-254(1990).

[10] 山崎　勝弘、奥田　健三： 事例ベース推論による観光経路決定支援システム、情報処理学会研究報告、91-AI-75-12、pp107-115(1991).

[11] 山崎　勝弘、奥田　健三： 事例ベースによる観光経路決定支援、情報処理学会第４４回全国大会、6Q-1(1992).

[12] 高橋　徹： 観光経路決定支援システムの事例ベース推論部の作成、立命館大学理工学部情報工学科卒業論文 (1992).

[13] 森田　真弘： 観光経路決定支援システムのルールベース推論部の作成、立命館大学理工学部情報工学科卒業論文 (1992).

[14] 西村　高志： 事例ベース推論による観光経路決定支援システムの改善と評価、立命館大学理工学部情報工学科卒業論文 (1993).

[15] 富士通研究所： AP1000 使用手引書、第 1.3 版 (1993).

[16] 富士通研究所： AP1000 ソフトウエア概説書、第 1.3 版、(1993).