

仮想空間を用いた 状況適応型歩行者ナビゲーションシステムの提案

日裏 博之[†] 上島 紳一[†]

[†] 関西大学大学院総合情報学研究科

〒 569-1095 大阪府高槻市霊仙寺町 2-1-1

hiura@kcn.ne.jp, ueshima@res.kutc.kansai-u.ac.jp

あらまし 近年, 移動体情報端末の高性能化に伴い, 利用者を目的地まで誘導する歩行者ナビゲーションが実現されている. これらの多くのシステムでは, 携帯端末向けに適した案内地図の作成や, 場所案内文の生成といった利用者をガイドするための地図の表示形式に関する研究が行われている. これらの研究は歩行者にとっての視覚的な分かりやすさに焦点が置かれており, 利用者の状況に応じた移動経路の選択や目的に応じた経路探索については考えられていない. 本稿では, これを実現するために空間の持つ特徴をメタデータとする論理仮想空間を構築する. さらに, 空間をメッシュに分割して各地点に得点付けを行うことで, 実空間上で状況に応じて歩行者が選択する移動ルートを検索するシステムについて提案する. 更に, シミュレーションにより, 本システムの有用性について考察する.

キーワード GIS, 空間情報処理, 歩行者ナビゲーション, 空間 DB

Proposal of Situation-Adaptive Pedestrian Navigation System with Virtual Space

Hiroyuki HIURA[†] and Shinichi UESHIMA[†]

[†] Graduate School of Infomatics, Kansai University

Abstract Due to recent remarkable progress of mobile environments, pedestrian navigation systems(PNS), which guide users to their destinations have been developed. In most systems, researches have been focused on visibility of maps, and display styles of maps in order to guide pedestrians to destination smoothly, such as (1) generation of guide map for mobile phones, (2) spot guidance for pedestrian, etc. On the other hand, situation-adaptive path finding strategy is out of scope in their systems. In this paper, we propose an experimental path finding strategy of PNS for this aim by constructing logical virtual space as meta data. In our approach, we partition this virtual space into mesh, and provide scoring method dependent on situations in order to find paths to destination for pedestrian. We also show numerical simulation result to verify effectiveness of our approach.

Key words GIS, Spatial Infomation, Pedestrian Navigation, Spatial DB

1. はじめに

モバイル通信環境の発展や基盤整備, また計算機の小型化・高性能化にともなって携帯電話や PDA といった移動体情報端末の普及は急速に進展している.[1] 携帯電話端末からのイン

ターネット利用は増加の一途をたどっており, 各種情報源への窓口となっている. このような情勢の中, ITS(高度道路交通システム)の一環として, 携帯端末を用いて歩行者をある場所から目的地まで誘導するシステムである歩行者ナビゲーションシステムが検討されている. 例えば, 「EZ ナビウォーク」[2] や

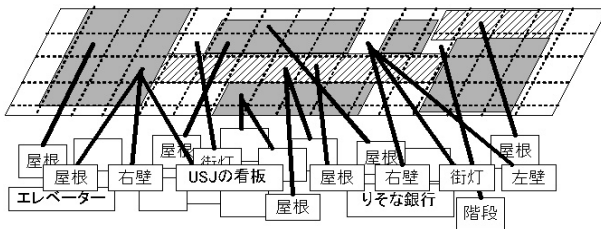


図 1 論理仮想空間における各地点の属性の例

表 1 空間属性の種類

種類	主なオブジェクト
移動制限物 (A1)	建物 (壁), 交差点, 階段, エスカレーター, etc
状況依存物 (A2)	屋根, 街灯, 路面, etc
表示物	店名, 看板, 広告物, etc

「M-stage」[3] の位置情報コンテンツである「とほなび」等の製品としてすでに実用化されつつある。また、PDA を用いた同サービスも存在し、歩行者ナビゲーションの潜在的な需要の高さを表している。

本論文では、状況に適応した歩行者ナビゲーションシステムを提案する。現状の歩行者ナビゲーションシステムは状況に適応した経路探索を行っている製品は存在しない。ここで述べる状況に適応した経路探索とは、実環境の変化によって歩行者は目的地までの移動経路を変更することがあり、その経路を探索することを意味する。例えば、雨天の場合、歩行者は目的地までの移動経路が最短でなくても雨に濡れにくい経路を選択することがあり、夜間の場合は街灯に照らされている明るい道路を選択することがある。つまり、歩行者の移動経路は単に最短経路に従うわけではなく、ある状況下において必要な空間の特徴を持つ経路を選択していることを意味している。これは特に地下街やアーケードが多数存在する都市中心部で顕著に見られる。既存の歩行者ナビゲーションシステムは地図を対象としているが、ここで提案する本システムは地図を対象とはしておらず、実空間をメッシュ状に分割し、各々のメッシュに対応する実空間の持つ特徴をメタデータとして格納した論理仮想空間を対象とする。その仮想空間内に得点付けや重みを付けることで最適な経路探索を行う。これは、既存の地図では地下街や建物内部の詳細な情報が不足しており、また空間の持つ特徴を記述することが困難であることに起因する。また、対象範囲は都市中心部付近といったような限られた短小矩形とする。

以下、本論文では、2章で論理仮想空間の構成方法について述べ、3章でその構築方法について述べる。4章で論理仮想空間に対する得点付け手法と状況に応じた経路の探索法について述べ、5章でこれらの手法を用いてシミュレーションによる実験を行い、その結果について述べる。6章で関連研究と本論文との位置付けを述べ、最後に今度の課題について触れる。

2. 空間の特徴をメタデータとする論理仮想空間

本章では、本システムにおいて提案する論理仮想空間の特徴について述べる。

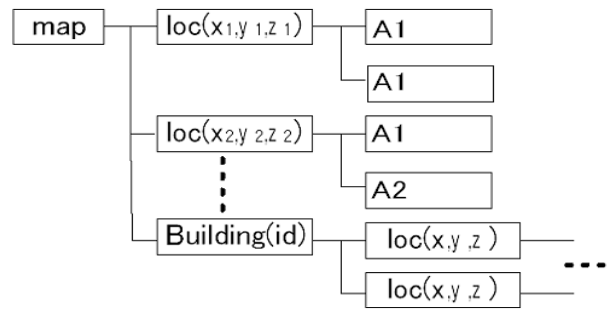


図 2 論理仮想空間のデータ構造例

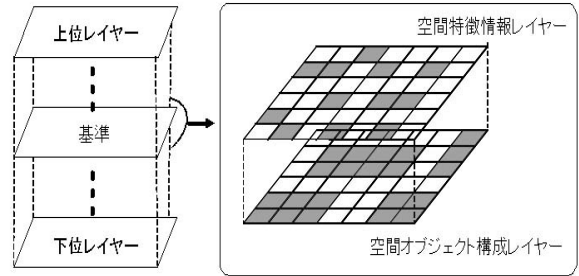


図 3 論理仮想空間のレイヤー構造

2.1 論理仮想空間の表現方法

仮想空間とは、現実世界ではなく計算機によって仮想的に作られた空間を指す。計算機の高性能化により、現在では利用者が3次元仮想空間内を自由に散策できるシステムも存在し、多くの研究[4]が行われている。しかし、本論文で述べる論理仮想空間とは可視的、及び空間内での散策を可能にするものではなく、単に計算機の主記憶上に展開された論理的な仮想空間と定義する。

通常、ナビゲーションシステムには地図データが用いられる。地図データは自治体発行の都市計画図等をベクトル化したデータであり、そこから抽出された道路や建物に対してナビゲーションに関する処理が行なわれる。このベクトル化された地図データでは道路の有無や繋がりは完全に把握することができるが、その道路の、ある位置における空間の特徴や道路以外での歩くことができる場所については把握することができない。そのため、目的地まで雨に濡れずに進む移動経路の探索や街灯があたる比較的明るい道路の探索を行うことは不可能である。また、地下街やデパート内は地図自体が存在しないため、その空間をナビゲーション空間にすることができない。このため、都市中心部付近といった限られた短小矩形の歩行者をターゲットとする場合、地図は不適切であると言える。

そこで、本システムの論理仮想空間は現実空間をメッシュ状に区切り、メッシュ中の一つ一つに対応する位置座標を格納系とし、その位置に存在する空間の特徴をメタデータとして格納する。図1のように、ある位置座標における空間の特徴は多数存在する。図中の網掛け部分は建物を表し、斜線部はアーケードを表している。例えば、アーケードや建物はその空間の持つ特徴である空間属性の一つとして屋根属性を持つ。このように定義することにより、雨天の場合は、屋根属性を持つ空間に着目することで雨に濡れない経路を計算することが可能になる。主な空

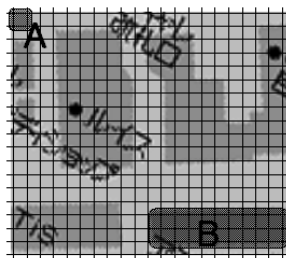


図 4 メッシュに区切られた
地図空間

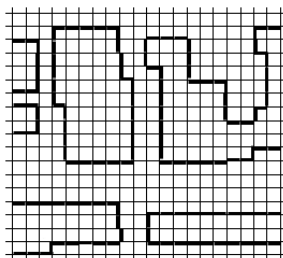


図 5 障害物情報の
論理仮想空間化

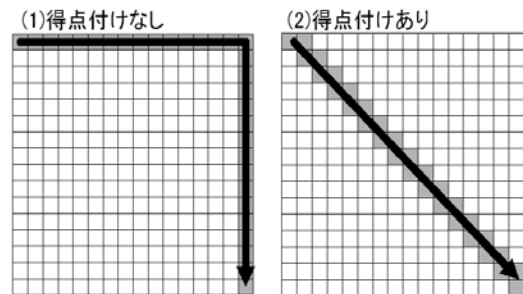


図 6 得点付けによる経路探索の違い

間属性を表 1 に挙げる。論理仮想空間のデータ構造は図 2 に示すように、位置情報を元に各メッシュを同列の枝として構成し、その空間属性を葉とする木構造で表現する。

論理仮想空間はこれらの空間属性を柔軟に格納するため、XML のようなテキストをベースとする。XML をベースとした論理空間で現実空間を表現すると、次のような利点がある。

- 拡張性が高い
- 記述 (構築) しやすい
- 階層構造を持たせやすい
- ファイルサイズが小さい

また、現実空間をメッシュに区切る構造のため、経路探索等におけるアルゴリズムが簡素化できる利点を持つ。

2.2 論理仮想空間のレイヤー構造

本システムの論理仮想空間は多層のレイヤー構造を持っている。このレイヤー構造は、デパート等の建物や歩道橋、地下街といった各オブジェクトの相対的な高さの概念から構成される。地面の高さを基準とし、デパートの 2 階や歩道橋はその上位レイヤーと考え、地下街は下位レイヤーとする。また、それぞれのレイヤーはさらに 2 つのレイヤー層から成り立っており、下位レイヤーに表 1 の移動制限物と状況依存物といった各空間の物理的な情報を格納する空間オブジェクト構成レイヤーを持ち、上位レイヤーに表 1 の表示物といった各空間の見目の特徴を保持する空間特徴情報レイヤーを持つ。図 3 にその概念図を示す。網掛け部はその空間の属性を保持していることを示しており、図 1 はこの 2 つのレイヤーを重ね合わせたものである。それぞれのレイヤーは個別に情報を持っており、移動経路の探索は空間オブジェクト構成レイヤーに対して行い、目的地までの経路案内は空間特徴情報レイヤーの情報を元に行う。

3. 論理空間の構築方法

本章では、前章で述べた論理仮想空間の表現方法から実際にテキストベースの論理空間を構築する際のスキーマ設計及び構築方法について述べる。

3.1 現実空間のメッシュ化

現実空間をメッシュで区切り論理仮想空間を構築するに至り、現実空間にある歩行を妨げる建物や壁といった移動制限物を論理仮想空間にマッピングする場合、その形状がメッシュとは完全に一致しないことがある。これは、メッシュが離散的であるのに対し、現実空間のオブジェクトは連続量であるためである。

```
<!DOCTYPE map [
<ELEMENT map (location | building)* >
<ELEMENT location (top_wall | bottom_wall
| right_wall | left_wall | ceiling | light )* >
<ELEMENT building (location | name)* >
<ELEMENT name (#PCDATA) >
<ELEMENT top_wall EMPTY >
<ELEMENT bottom_wall EMPTY >
<ELEMENT right_wall EMPTY >
<ELEMENT left_wall EMPTY >
<ELEMENT ceiling EMPTY >
<ELEMENT light EMPTY >
<ATTLIST location id ID #REQUIRED >
<ATTLIST building id ID #REQUIRED >
<ATTLIST building path CDATA #REQUIRED>
]>
```

図 7 論理仮想空間のデータスキーマ

そのため、それらのオブジェクトを仮想空間にマッピングする際に、メッシュの形状に合わせてそのオブジェクトの位置情報を修正しなければならない。ここでは、比較的簡単な方法として、マッピングした際にオブジェクトが重なるメッシュの一番外側に壁が存在することとし、オブジェクトを包み込む形で仮想空間に取り込む。

図 4 は地図をメッシュで区切ったものである。図中の濃色が建物等の移動を制限するオブジェクトである。この地図の空間を論理仮想空間として構築し、移動制限物に関してだけ表示したのが図 5 である。オブジェクトが重なったメッシュの外側に沿って壁を作成している。

この際、メッシュを小さくすることで対象とする空間を精緻に表現できる。その一方、データ量は膨大になる。しかし、メッシュを大きくすれば、データ量は抑えられるが空間を精緻に表現することはできない。つまり、トレードオフの関係にある。そこで、本システムでは、歩行できる道幅を考慮し、およそ 1m 間隔でメッシュを区切ることにする。

3.2 論理仮想空間のスキーマ

図 7 は論理仮想空間におけるメタデータのスキーマを定義している。各メッシュは“location”タグの“id”属性によって管理し

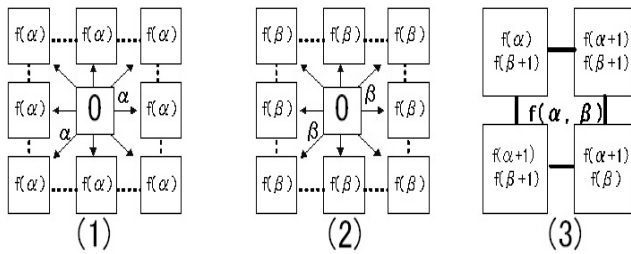


図 8 得点付け手法

ている。"id"属性はそのメッシュの中心点の経度、緯度情報とする。例えば、図 4 における A 地点のメッシュの場合、`<location id="192.5, 34.6"> </location >`

となり、この空間が持つ属性は"location"タグの子要素として記述する。また、その属性として移動制限物の情報を"top_wall", "bottom_wall", "right_wall", "left_wall"の4種類のタグで定義する。この定義によりオブジェクトをメッシュで包み込むことができる。そして、状況依存物の情報として"ceiling", "light"タグを定義する。これらのタグは空間の状態を表示だけの情報のため、空要素とする。

"building"タグは複数のメッシュから一つのオブジェクトが構成される場合、そのメッシュに対応する"location"タグの親要素となる。図 4 中、B 地点は一つのオブジェクトである。"building"タグは"id"属性と"path"属性を持つ。"id"属性はそのオブジェクトを一意に認識できる任意の値をとり、"path"属性はそのオブジェクトが通り抜けることが可能なら"yes"の値を、それ以外は"no"の値をとる。

4. 得点付け手法による経路探索

本章では、論理仮想空間を利用した状況に依存した経路を探索する方法について述べる。

4.1 メッシュ化された論理仮想空間での最短経路

道路ネットワークにおける一般的な最短経路の探索は、Dijkstra のアルゴリズムで解くことができる。道路が交差する点をノードとし、道路をリンクとするグラフ構造をとるのが一般的であり、この際、道路のリンク長を道路の長さや時間で換算したコストを用いて Dijkstra のアルゴリズムを適用する。各道路の長さは個別の値であるため、出発地から目的地までの最短経路はコストが最低になる経路である。ところが、本システムでは空間属性を格納するために空間をメッシュ状に区切って管理しており、道路の長さといった情報は格納していない。そのため、経路探索にはメッシュ間の繋がりを考慮して、あるメッシュはその上下左右のメッシュと繋がっているとすると4方向近傍側とし、メッシュをノード、メッシュ間の繋がりをリンクするグラフ構造とした。

このままでは、各メッシュ間の長さは一定のため長さの違いによる最短経路を算出することができず、Dijkstra のアルゴリズムを適用すると図 6(1) のようになり、始点の位置を (S_i, S_j) とし終点の位置を (G_i, G_j) とするとユークリッド空間での長

さは

$$L = |G_j - S_j| + |G_i - S_i| \quad (1)$$

となる。一般的にメッシュ1マスを移動単位と考えると、論理空間での最短経路は一意に決定しにくい自由度のある経路集合が解となる。ところが、歩行者を誘導するナビゲーションに必要なのは目的地までの目印である。そのため、探索解の周囲の情報を目印とする場合、自由度のある経路集合の内、ユークリッド空間(現実空間)での最短経路にもっとも近い経路を最短経路であると定義する。つまり、出発地から目的地までの最短経路 L は

$$L = \sqrt{(G_j - S_j)^2 + (G_i - S_i)^2} \quad (2)$$

に最も近い経路であり、ここでは図 6(2) の経路とする。

4.2 メッシュ空間への得点付け

上述したように、メッシュ間の長さは一定のためリンク長のコストを利用した Dijkstra のアルゴリズムを適応させることは困難である。そこで、出発地の位置と目的地の位置の相対的な距離関係を利用し、最短経路に近いメッシュのリンク長のコストを動的に低くし、最短経路から遠ざかるメッシュのリンク長のコストは遠ざかる距離に応じて動的に高くなるように各メッシュに得点付けを行った。その方法を以下に示す。

step1 出発地点の位置を対応するメッシュ空間にあてはめ、その地点のメッシュの得点は0点とする。なおこの地点のメッシュを始点メッシュと呼ぶ。

step2 始点メッシュを基準とし正方形の波紋が広がるようにメッシュを得点化していく。この際、始点メッシュから α 番目に離れた波紋の位置に存在する全てのメッシュの得点

$$f(\alpha) = (dis_\alpha \times 2) \quad (3)$$

とする(図 8(1)). dis_α は α 番目までの波紋の数である。

step3 目的地の位置を対応するメッシュ空間にあてはめ、その地点のメッシュの得点を0点とする。なおこの地点のメッシュを終点メッシュと呼び、出発地点からの得点とは別に管理する。

step4 終点メッシュからも(2)と同様の方法で得点化する。終点メッシュから β 番目に離れた波紋の位置に存在する全てのメッシュの得点

$$f(\beta) = (dis_\beta \times 2) \quad (4)$$

とする(図 8(2)).

step5 これら2つの得点を合計したものをそのメッシュの総得点とする。この得点は出発地点から目的地までの最短経路からどれくらい離れているかを示す指標になる。

step6 最後にメッシュの繋がりのリンク長のコストとして、メッシュ間の得点の平均

$$f(\alpha, \beta) = \frac{(dis_\alpha \times 2 + dis_\beta \times 2)}{2} \quad (5)$$

を与える(図 8(3)).

上記の方法で各メッシュ間のリンク長に対してコストを与えることで、Dijkstra のアルゴリズムを適用した。この得点付けを行うことにより、得点付けを行わない結果であった図 6(1) から最短経路である図 6(2) を導くことができた。

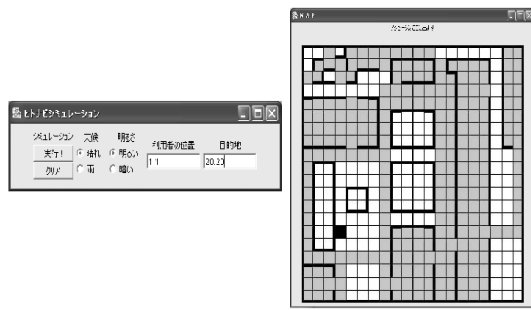


図 9 プロトタイプシステム

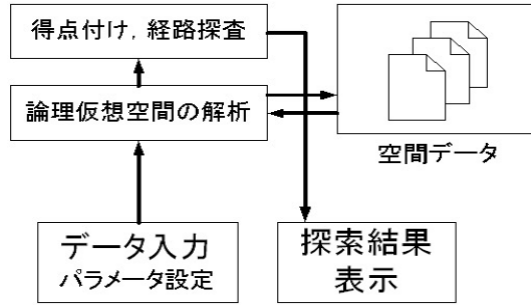


図 10 プロトタイプシステムの構成

4.3 空間属性に対する重み付け

出発地点と目的地点の双方から距離に応じて得点付けをおこなうことにより、最短経路を導くことができた。だが、我々の目的は空間の特徴に着目した環境等の状況に応じた移動経路の変更である。これを表現する方法として、ある状況時に必要な属性を持つ空間に対して、その地点に対する重みを付ける事で表すことにする。ある状況時に必要な属性とは、例えば雨天時なら屋根属性(屋根の有無)や路面属性(アスファルト、砂利道 etc)、夜間時なら光源属性(街灯等)といったものが挙げられる。

具体的な方法としては、環境の状況に応じて、上述したメッシュ空間の距離に応じた得点に対して必要な空間属性が存在する地点に一律に負の重み d_1 をかける。これは Dijkstra のアルゴリズムが総コストが一番低く(または高く)なるような経路が解になるためである。またそれ以外の地点には一律に正の重み d_2 をかける。この重みをかける値の設定によっては、

- 「遠回りになってもしっかり雨に濡れない経路を探したい」
- 「雨には濡れたくないが、遠回りするくらいなら多少濡れても近い道を探したい」

といった経路の探索が可能になる。

5. シミュレーションによる実験

4. で述べた手法をもとにシミュレーションを行ったので、そのシステムと結果について述べる。

5.1 プロトタイプシステム

本システムのプロトタイプは JAVA を用いて構築した。プロトタイプシステムのインターフェースにはシミュレーションを行う条件を設定するウィンドウと移動経路を表示するウィンドウから成り立っている。シミュレーションを行う条件として、本

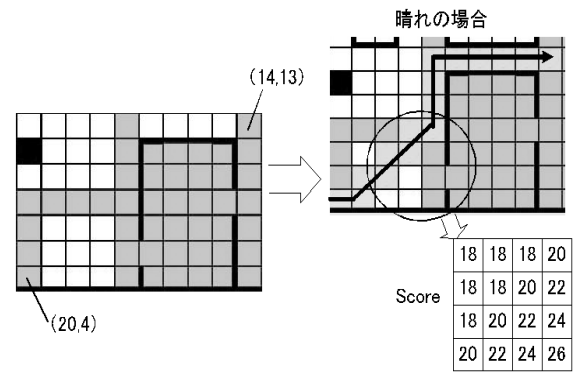


図 11 状況による移動経路の探索結果

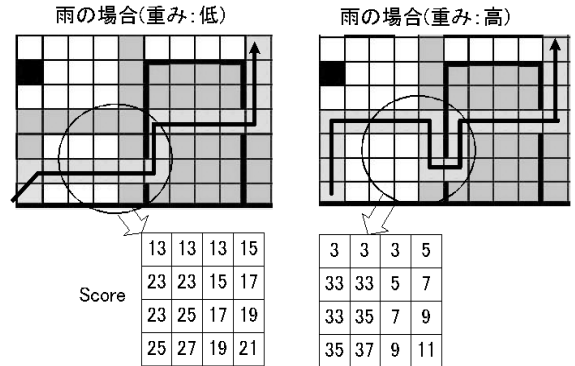


図 12 重みのかけ方による移動経路の違い

システムでは晴雨の設定、昼夜の設定が可能であり、出発地点と目的地点をそれぞれ入力する形になっている。

プロトタイプシステムの一連の流れを図??に示す。シミュレーションの設定を行うウィンドウからシステムにデータとして各種パラメータ値が受け渡される。論理仮想空間の解析部分では XML で記述された空間のデータを XML パーサを利用し主記憶上に隣接行列として展開する。この際、パラメータをして入力された出発地点、目的地点がある建物内に存在するならば、その建物の”building”要素の子要素及び、ルートの子要素である全ての”location”タグの要素を格納する。また、”building”タグの path 属性が”yes”の値の子要素のみを展開し、”no”の値の要素は展開しない。これはメッシュの全ての情報を主記憶上に展開するとデータ量で圧迫されるので、目的地までの移動に通り抜けることができない建物部を排除して展開していることになる。

次に各メッシュに得点付けを行い、さらに重み値を対応する隣接行列の各行列にスコアとして与える。隣接行列ではノード間が移動可能ならその両端のノードを A,B とした場合 A 行 B 列は 1 となるが、ここでは A 行 B 列はメッシュの得点に重み値を加算したものになる。最後に Dijkstra アルゴリズムによる経路探索を行い、確定した経路をウィンドウに表示する

5.2 シミュレーションによる結果と評価

図 9 の左がシミュレーション設定ウィンドウ、右が表示ウィンドウである。ここでは実験のために現実空間をモデルとしていない 20 マス × 20 マスの仮想的な空間を構築した。表示ウィ

ンドウには、その論理仮想空間が示されている。表示ウィンドウ内に表示されている太線はその方向に壁があることを意味しており、網掛け部分はその地点に屋根属性があることを意味している。また、壁で囲まれているところは建物である。

シミュレーションの条件として、出発地点を(20,4)、目的地点を(14,13)の座標値にした。また、昼夜は「昼」に設定し、「晴れ」と「雨」の2通りの場合についてシミュレーションを行った。結果は「晴れ」の場合が図11中の右図になり、薄い網掛け部分が移動経路を示している。ここでは見易さを考慮して矢印を表示している。「晴れ」の場合は空間属性が考慮されないため、2地点間の最短経路が表示されている。また、「雨」の場合は図12の左図のようになり、比較的屋根のある空間を通っていることが分かる。さらに「雨」の場合に対する重みのかけ方を強くすることで図12の右図の結果のように、遠回りになっても全く雨に濡れない経路を探索することも可能になった。

以上のことから、本論文で提案するシステムは環境などの状況に適応した経路を探索できることが分かった。

6. 関連研究

歩行者ナビゲーションシステムに関する研究は、表示方式に関する研究と経路探索に関する研究が行われている。

表示方式に関しては、携帯端末の表示性能の制約から詳細な地図情報を表示するのではなく、地図情報のセマンティック情報とジオメトリック情報を利用して携帯端末向けに適した案内地図を作成するシステム[5]や人の空間認知の観点から、歩行者のナビゲーション支援に有効である目標物に対する移動方向を記述する表現、目標物に対して相対的な位置関係を示す空間語表現を行う場所案内文を作成するシステム[6]、歩行者の視点に立った指示案内文と風景画像を表示するシステム[7]など様々な研究が行われている。

また、経路探索に関する研究としては遺伝的アルゴリズムを利用して動的な情報検索を短時間で可能とした、最短経路探索を対象とした研究[9][10]や道路網を階層化して表現し、人の直感に合った最適経路を効率よく求める研究[11]等がある。

本システムと他の研究の一番の違いは、本システムでは地図情報ではなく論理仮想空間を利用しているところである。表示方式に関する研究の多くは、目的地までの経路探索にはDijkstraアルゴリズムを用いており、移動経路はほぼ最短の経路になると考えられる。また、経路探索に関する研究に関しても提案手法は様々であるが、本質的に目的地までの最短経路をいかに高速で計算できるかに主眼が置かれている。本システムは、目的地までの経路を最短にするのを目的とはしていない。本システムにおける移動経路の探索の特徴は、目的地までの経路を状況の違いによって最適な経路を計算することである。ここで述べる最適な経路とは目的地までの最短経路と同意ではなく、状況によっては遠回りをして歩行距離が伸びたととしても、その状況下における一番よいとされる経路を提供することである。つまり、本システムはヒトの歩行という特徴に重点をおいている。これは、移動経路の探索に関して、本システムと他の研究の大きな違いであり、本システムの特徴の一つである。

7. おわりに

本論文では、空間の特徴に着目した、環境などの状況に適応した歩行者ナビゲーションシステムを提案した。都市中心部付近といったような限られた範囲では、目的地までの経路は複数存在し、そのため地下街や空間の属性情報を持たない地図では対応できないため、空間属性を記述できるメッシュ化した論理仮想空間を構築した。また、得点付け手法や空間属性に対して重みをかけることにより状況に応じた経路を探索することが可能になり、シミュレーションにより示した。

今後の課題として、膨大になるメッシュ間の繋がりを考慮したデータ構造の仕組み、現実の都市中心部に対するのシミュレーション実験、携帯電話を利用した歩行者に対するナビゲーションの提供、などの検討が挙げられる。

文 献

- [1] 総務省通達
(http://www.soumu.go.jp/joho_tsusin/pressrelease/japanese/sogo-tsusin/031128-1.html)
- [2] KDDI
(<http://www.au.kddi.com/>)
- [3] NTTDoCoMo
(<http://www.nttdocomo.co.jp/p.s/mstage/>)
- [4] 日裏, 西出, 北村, 大西, 森田, 小林, 上島, ' デジタルキャンパスのアメニティ管理とIPv6 インターネットを用いた講義の宅配環境', DBWEB2002
- [5] 藤井, 杉山, ' 携帯端末向け案内地図作成システムの開発', 情報処理学会論文誌, Vol.41 No.9 pp.2394-3003(2000).
- [6] 藤井, 杉山, ' 歩行者ナビゲーション支援のための場所案内文作成手法', 電子情報学会論文誌, D-II Vol.J82-D-II No.11 pp.2026-2034(1999).
- [7] 福井, 白川, 歌川, 重野, 岡田, 松下, ' 携帯電話における歩行者ナビゲーション情報の表示方法に関する提案と評価', 情報処理学会論文誌, Vol44 No.12 pp2968-2978(2003).
- [8] Lynch, K.: The Image of the City, MIT Press, Cambridge, Massachusetts(1960).
- [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 02-ITS-8, pp.51-58(2002).
- [10] 狩野, 小塚 ' CA 法による広域道路交通シミュレータを用いた経路案内方式の評価', 情報処理学会研究報告 02-ITS-8, pp.37-43(2002).
- [11] 丹波, 吉田, 福村, ' 道路網の階層的表現にもとづく経路探索アルゴリズムと地図情報システムへの応用', 情報処理学会論文誌, Vol.31 No.5 pp.659-665(1990).