

# 包含概念を用いた拡張 R-tree による被災地管理

三栗 寛之<sup>†</sup> 向 直人<sup>†</sup> 渡邊 豊英<sup>†</sup>

<sup>†</sup> 名古屋大学大学院情報科学研究科 〒464-8603 名古屋市千種区不老町

E-mail: <sup>†</sup>{hiro,naoto,watanabe}@watanabe.ss.is.nagoya-u.ac.jp

あらまし 災害の発生を予知することは困難であり、それによる被害を完全に抑え込むことも不可能である。故に、効果的な災害対策を採って発生に備えることが重要である。本研究では情報技術を用いた災害対策の一つとして、被災者を避難所へ誘導する問題を扱う。災害によって引き起こされた被害により、被災地には被災者が安全に行動できない危険領域が存在し、これは時々刻々と変化する。被災者を安全に避難所へ導くためには、この領域を避けた経路を設定する必要があるが、この領域が固定のものでないために効果的な誘導が困難である。本手法では、危険領域とその領域の周囲に存在する、予測による危険領域を用いて R-tree を拡張した木構造を構築し、領域を階層的に管理する。これにより、各被災者に対して効果的な避難経路を設定することが可能となる。

キーワード 被災者ナビゲーション, 拡張 R-tree, 危機管理

## Disaster-Area Management to Navigate Refugee by Using Extended R-tree with Inclusive Concept

Hiroyuki MIKURI<sup>†</sup>, Naoto MUKAI<sup>†</sup>, and Toyohide WATANABE<sup>†</sup>

<sup>†</sup> Department of Systems and Social Informatics, Graduate School of Information Science,  
Nagoya University, Furo-cho, Chikusa-ku, Nagoya, 464-8603, Japan

E-mail: <sup>†</sup>{hiro,naoto,watanabe}@watanabe.ss.is.nagoya-u.ac.jp

**Abstract** It is difficult to foretell when disasters occur and it is also impossible to protect persons and cities completely against damages occurred by disasters. Thus, it is important to set up counter plans against disasters in advance. In our research, we deal with a matter that we supply refugees in disaster areas with information about a safe route to a shelter. In disaster areas, there are some regions where refugees cannot go through safely in consequence of damages by a disaster. And such regions change their forms and expand their sizes dynamically. In this paper, we construct inclusively extended R-tree by using the regions and their estimated future conditions. Then, we propose managing data about the regions by using the tree structure. This makes it possible to select an effective route to a shelter for each refugee.

**Key words** Refugee Navigation, Extended R-tree, Crisis Management

### 1. はじめに

世界各地で地震や集中豪雨による浸水や土砂災害、火山災害等、様々な大規模災害が発生し、多くの人々が被害に遭っている。このような災害は、今後もこれまでと同様に発生すると考えられ、様々な対策が立てられてきた。これまでの災害対策は、防潮堤、砂防ダム、水門等のインフラの整備であり、これらによって災害の発生自体を抑止し、災害が発生したとしても被害を発生させないことを目標としてきた。

しかし、人間の努力や工夫によって自然災害を完全に防ぐことは困難である。近年、これまでの災害対策と共に、災害による被害を軽減することを目指す、「減災」という考え方が重

要視されるようになってきた [1], [2]。この考えにより、インフラに代表されるハードウェア面での対策に加えて、様々な情報技術を用いたソフトウェア面での対策も立てられるようになった。具体例としては、洪水による河川の氾濫や土砂災害の発生を予測して警報を発するシステム、地震発生時にガスや電気を一時的に止めることで二次災害の発生を予防するシステム、災害発生後に画像処理によって大まかな被害状態を把握するシステム等が挙げられる。また、計算機上でのシミュレーションから災害対策において有効な知見を得ようとする試みもされている [3], [4]。本研究では、減災を目指す情報システムの一つとして、被災地における被災者ナビゲーションシステムを取り上げ、特に大地震発生時に各被災者に対して避難所への避難経路を設

定する問題を扱う。

大地震が発生した際には、多数の人が「危険な領域」あるいは「将来危険になる領域」にいと想定され、安全な領域（避難所）に誘導することが必要である。また、誘導システムの処理能力には限界がある。故に、本研究では各被災者の経路設定を工夫することで、低い計算負荷でより多くの人を安全な領域（避難所）に導くことを目標とする。

大地震が発生すると、都市部には危険領域が生じる。この危険領域は、地震による建物倒壊や二次災害で火災などが発生した領域に相当し、被災者はこの領域内を安全に通過できない。したがって、避難者はこの領域を通過しないという制約条件が存在する。ここで問題となるのは、危険領域が空間的に固定ではないことである。すなわち、経路設定時に通過不能として除外しなければならない領域が時々刻々と変化する。このため、ある時点においてどこに危険領域が存在するかという情報だけを利用して、安全な避難経路を設定することは困難である。そこで、既知の危険領域から将来の被災地の状況を予測し、被災地内の領域を危険度に基づいていくつかのクラスに分類する。そして、より安全な領域から順に通行可能として経路を求めることで、その状況において可能な限り安全な避難経路を得る。

また、経路設定に必要な計算負荷を抑えなければならない。被災地には避難経路設定の対象者が多数存在するが、経路設定に使用可能な処理能力は限られており、必ずしも全ての被災者に対して即時的に経路を設定できるわけではない。このような状況では、各被災者に必要となる計算負荷を抑えることによって、より多くの被災者を誘導できる。先に挙げた予測情報を取り入れて経路を設定すると、可能な限り安全な経路を得ることができるが、予測情報を取り入れない場合と比較すると計算負荷が増大し、避難経路を設定できる人数は減少してしまう。このため、避難経路の安全性と避難経路設定が可能な人数との間にはトレードオフ関係が存在すると言える。故に、計算負荷を抑えつつ安全な経路を設定するための手法として、R-treeを拡張した木構造による危険領域の索引手法を提案する。この木構造では、それぞれのノードが少なくとも一つ以上の矩形情報を保持し、木構造の各階層は、各ノードで管理する危険領域に相当する。また、木構造中で上位階層のノードが持つ矩形情報は下位階層のノードが持つ矩形情報を包含し、より危険度の低い領域を併せて管理する。木構造を用いて各領域の状態を判定することにより、安全な避難経路が存在する見込みが薄い領域内の経路探索を打ち切ることができ、効率的な避難経路設定が可能となる。これにより、被災者一人に対しての経路設定に必要な計算負荷は抑えられ、より多くの被災者を誘導できる。

本稿の構成は以下である。2章で各被災者を避難所へ誘導する問題を形式化し、本問題での目標を定義する。提案するアプローチとして、危険領域の変化予測を3章で、危険領域を索引付けする木構造について4章で述べる。5章では、この木構造を用いた被災者に対する経路設定を述べる。6章では、シミュレーション実験の結果を示す。最後に、7章でまとめと今後の課題を述べる。

## 2. 形式化

被災地  $M$  は2次元平面上に存在し、建物集合  $B$  と道路ネットワーク  $G$  とで構成される(式(1))。建物集合  $B$  は被災地内の建物の集合であり(式(2))、各建物は全て矩形で表現される(式(3))。また、道路ネットワーク  $G$  は、ノード集合  $N$  とエッジ集合  $E$  から構成され、各ノードは交差点を、各エッジはノード間を繋ぐ道路セグメントをそれぞれ表す(式(4))。

$$M = (B, G) \quad (1)$$

$$B = \{b_1, b_2, \dots\} \quad (2)$$

$$b_i = ((x_{left}, y_{top}), (x_{right}, y_{bottom})) \quad (3)$$

$$G = (N, E) \quad (4)$$

$$N = \{n_1, n_2, \dots\}$$

$$E = \{|n - n'| : n, n' \in N\}$$

被災地には、危険領域の集合  $D$  が存在する。 $n$  個の危険領域が存在する場合、式(5)のように表される。 $d_i$  はそれぞれ地震による建物倒壊や地震に誘発された火事等に相当する。これを簡略化するために、ある時刻  $t$  における各危険領域を矩形を用いて式(6)のように表現する。また、 $d_i$  は時間の経過と共に拡大する(式(7))。これは延焼や建物の雪崩的な倒壊に相当する。式(7)において  $\delta x_{i1}, \delta x_{i2}, \delta y_{i1}, \delta y_{i2}$  は非負の値であり、単位時間あたりに  $d_i$  がどれだけ拡大するかを表す。

$$D = \{d_1, d_2, \dots, d_n\} \quad (5)$$

$$d_i(t) = ((x_{left}(t), y_{top}(t)), (x_{right}(t), y_{bottom}(t))) \quad (6)$$

$$d_i(t+1) = ((x_{left}(t) - \delta x_{i1}, y_{top}(t) - \delta y_{i1}), (x_{right}(t) + \delta x_{i2}, y_{bottom}(t) + \delta y_{i2})) \quad (7)$$

被災地  $M$  上に存在する被災者の集合を被災者集合  $R$  とする。 $n$  人の被災者が存在するとき、 $R$  は式(8)で表される。各被災者は、自身の  $x-y$  平面上での位置情報  $(p_x, p_y)$  と避難所集合  $S$  に属する一つの避難所への経路情報  $Route$  を保持し、式(9)のように表現される。避難所集合  $S$  は建物集合  $B$  の部分集合であり、各避難所  $s_i$  は公園や小中学校、公民館等に相当する(式(10))。被災者はそれぞれ道路ネットワーク上のいずれかのノードから出発し、避難所集合  $S$  の要素  $s_i$  に近接するいずれかのノードに到達することを目標として移動する。この際、被災者は避難所への誘導に従う。また、被災者は危険領域とされている領域を通過することはできない。周囲を危険領域で囲まれて、避難所への経路が存在しない場合、または、拡大する危険領域にのみこまれたりした場合には、行動不能とする。

$$R = \{r_1, r_2, \dots, r_n\} \quad (8)$$

$$r_i = ((p_x, p_y), Route) \quad (9)$$

$$S = \{s_1, s_2, \dots\} \subseteq B \quad (10)$$

避難所への経路情報は、被災者の現在地ノードから始まる、ノードによるリスト構造で表現される(式(11))。この経路情報は、被災者の避難開始時に設定され、その後は一定時間毎に更新される。また、被災者が経路情報に従って道路ネットワーク  $G$  上を避難所  $s_i$  へ進む際に、危険領域によって次に到達予定ノードまでの経路が遮られた場合にも更新される。簡略化のために、経路情報の更新には時間を必要としない。また、安全に避難所に到達可能な経路が複数存在する場合には、危険領域の拡大を考慮してより危険になりにくい領域を通過する経路が優れていると考える。

$$Route = (n_i, n_j, \dots) \quad (11)$$

以上に定義した環境下で被災者を避難所へ誘導する。そして、その結果において式(12)を最小化することを目標とする。式(12)において  $Num\_of\_Reached$  は避難所に到達できた被災者の数を表す。また、 $Cost(i)$  は1人の被災者が避難開始から避難所への移動が完了するまで、もしくは行動不能になるまでの、避難経路設定に必要な探索コストを表す。我々はこれを被災者誘導コストと呼び、5章で詳しく述べる。式(12)は被災者一人を避難所へ導くためにどれだけの計算能力が必要であるかを表し、経路設定のコスト対効果を評価する指標である。

$$\frac{\sum_{i=1}^N Cost(i)}{Num\_of\_Reached} \quad (12)$$

### 3. 危険領域の変化予測

被災者が安全に通過できない危険領域を避けて、避難経路は設定されなければならない。ここで問題となるのは、危険領域が空間的に固定の領域ではないことである。すなわち、危険領域は時々刻々と変化する。また、災害発生時には、被災地内の全ての領域の状態変化を逐次取得することは不可能である。故に、経路を設定する時点で利用可能な情報は、それ以前のものであり、実際の危険領域とは異なる可能性がある。さらに、ある地点が通行可能かどうかを判定する際に必要となるのは、経路を設定する時点の情報ではなく、各被災者がその地点を通る時点の情報である。以上より、経路設定の時点でどこに危険領域が存在するかという情報だけを利用して安全な避難経路を設定することは困難である。また、この情報のみを用いて設定した経路では、被災者は行く手を危険領域によってしばしば遮られるため、頻繁な経路更新が必要となり非効率である。

この問題点を解決するために、将来の危険領域の状態を予測して経路設定時に利用する。具体的には、ある時点においてその領域がどのくらいの確率で危険領域であるかを表す危険度を導入し、危険度が小さい領域を優先して避難経路を設定する。本章では、変化予測で用いる仮定と変化予測例について述べる。

#### 3.1 変化予測における仮定

変化予測は以下の仮定のもとで行う。

(1) 既知の危険領域は時間の経過と共に周囲に拡大する。また、危険領域が突如として発生することはない。

(2) 一度危険領域となった地点はそれ以後は常に危険領域であり、危険領域が縮小することはない。

(3) 被災地の GIS データが使用可能であり、建造物の耐震性や耐火性、密度等を考慮可能である。

(4) 被災地のどこが危険領域であるかという情報のある時間間隔で取得できる。

現実の状況下では、周囲に危険領域が存在しない地点に突如として危険領域が発生することが起こり得る。また、一度危険領域となった地点が時間の経過や救助活動等によって危険領域ではなくなることも考えられる。しかし、これらを考慮して予測することは非常に困難である。故に、(1)、(2)のように考え、既知の危険領域が周囲へ拡大することだけを仮定する。そして、(4)から得たある時点の危険領域の情報を元に、将来危険領域がどのくらい拡大するか(3)を用いて予測する。

#### 3.2 予測例

以上の仮定のもとで危険領域の周囲への拡大の起こり易さを予測し、被災地内の全領域を危険度を用いて分類する。危険度は、将来のある時点においてその領域がどのくらいの確率で危険領域となるかを表す。すなわち、危険度が高い領域は将来危険領域となる確率が高く、危険度が低い領域は将来危険領域となる確率が低い。仮定より、既知の危険領域の周りには危険度が高い領域が付随して存在し、危険領域から遠ざかるにつれて危険度が下がる。よって、同心円状の予測結果が得られる。1つの危険領域に対する拡大予測結果を、危険度を閾値としたクラスに分類すると図1のような結果を得る。

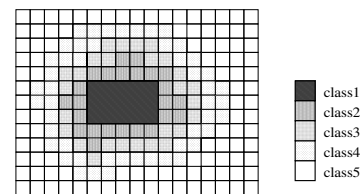


図1 危険領域の周囲への拡大予測結果

Fig.1 Estimation of spreading dangerous region

図1では、中央に既知の危険領域が黒い矩形で表現されており、その周囲に将来の予測される危険領域が表現されている。図中には、中央の矩形に近いところから5つのクラスの領域が存在する。まず、最も危険度が高いクラスの領域は中央の黒い矩形である。この領域は既知の危険領域であり、将来においても常に危険領域である。次に、危険度が2番目に高いクラスは中央の矩形のすぐ周囲の領域である。この領域は、予測した時点では安全であったが、将来に危険領域となる可能性が最も高い。同様に、中央の矩形から外側に行くに従って危険度が3番目、4番目に高いクラスの領域が表現されている。そして、最も外側に危険度が最も低いクラスの領域が存在する。このクラスには、1番目から4番目のクラスの領域に含まれなかった領域全てが属する。

先に2章で定めたように、全ての危険領域は矩形で表現される。故に、図1のように予測結果が矩形とならない場合には、それぞれのクラスに対応する領域について、それぞれの Minimum Bounding Rectangle (MBR) を考えて近似する。図1に対してこの近似処理を行うと、結果として図2を得る。

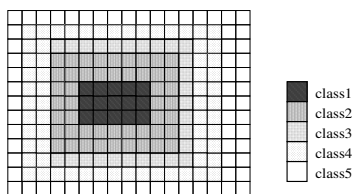


図 2 図 1 に対する矩形近似処理結果

Fig. 2 Approximating danger region shown in Fig. 1

本稿では、図 1, 図 2 のように 5 つのクラスに分類した場合についてのみ検討する。また、5 つのクラスに分類した領域を危険度が高い方からクラス 1 ~ クラス 5 の領域と呼ぶ。

予測結果を用いて、被災者ナビゲーション問題における避難経路を以下のように求める。最初に、将来における危険度が最も低いクラス 5 の領域のみを通行可能として避難経路を探す。もし、経路が存在しない場合にはクラス 4 の領域も通行可能として避難経路を探索する。それでも避難経路が存在しない場合には、以下同様にクラス 3, クラス 2 と 1 クラスずつ通行可能な領域を増やして避難経路を探索する。このような手順で経路を設定することで、各被災者の置かれている状況において、可能な限り安全な経路を求めることが可能となる。

#### 4. RNR-tree を用いた危険領域管理

よく知られた空間オブジェクトの管理手法の一つに、R-tree が存在する [5]。R-tree は平衡木の構造を持っており、空間オブジェクトを効率的に管理する。また、対象とする問題に応じて、様々な拡張 R-tree が提案されている [6], [7]。

被災者ナビゲーション問題においては、既知の危険領域情報の管理だけでなく、将来の予測による危険領域の管理も必要となる。また、多くの被災者に即時的に避難経路を設定するために、各被災者に対する避難経路設定に必要な計算負荷を抑えなければならない。故に、将来の危険予測でのクラス分けによる危険領域を、クラス毎に木構造の階層に対応付けて索引付けする拡張 R-tree, Refugee Navigating R-tree (RNR-tree) を提案する。RNR-tree は、避難経路が存在する見込みが薄い領域の探索を効率的に打ち切るため、避難経路を導く際の計算負荷を抑えることができる。

以下で、RNR-tree の構造、ノードの定義、特徴、構成法、RNR-tree 独自の危険度に基づく問合せについて述べる。

##### 4.1 RNR-tree の構造

RNR-tree は、被災者ナビゲーション問題向けに R-tree を拡張した木構造である。RNR-tree を用いて予測による危険領域を索引付けした RNR-tree の例を図 3 に示す。図 3 では、最上位階層にルートノード、最下位階層に葉ノードがそれぞれ表現されている。また、葉ノードはクラス 1 の矩形、葉ノードより一つ上の階層の中間ノードはクラス 2 の矩形、さらに一つ上の階層の中間ノードはクラス 3 の矩形、さらに上の階層の中間ノードはクラス 4 の矩形でそれぞれ表現されている。これは、各階層でどのクラスの領域を危険領域とみなすかに対応する。すなわち、より上位階層のノードほど危険領域の広がりに対し

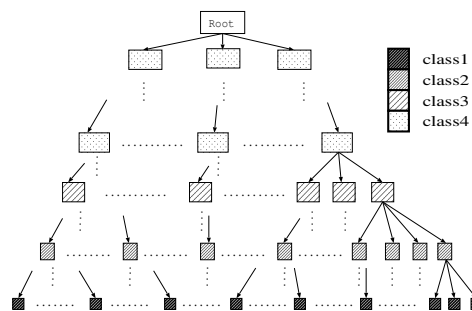


図 3 RNR-tree の構成図例

Fig. 3 An example of constructed RNR-tree

て余裕を持った領域を危険領域として保持する。

##### 4.2 RNR-tree のノード

RNR-tree は、図 2 のような構造を葉ノードとする。この葉ノードを式 (13) のように表現する。式 (13) において、 $Rect(class \alpha)$  はクラス  $\alpha$  の矩形であり、それぞれの矩形は式 (14) のように矩形の左上の座標、右下の座標の組で表現される。

$$leaf(i) = (Rect(class1), Rect(class2), \\ Rect(class3), Rect(class4)) \quad (13)$$

$$Rect(class \alpha) = ((x_{left}, y_{top}), (x_{right}, y_{bottom})) \quad (14)$$

中間ノードはクラス 2, 3, 4 の矩形の中から、存在する階層に応じて、いずれかの矩形をその階層での危険領域とする。そして、この矩形をそのノードでの最小の矩形とし、この矩形より危険度が低いクラスに対応する全ての矩形を持つ。すなわち、RNR-tree における中間ノードは式 (15) のように表現される。

$$intermediate(i) = \{Rect(class2), Rect(class3), \\ Rect(class4), level, density\} \quad (15)$$

式 (15) において、 $level$  はこの中間ノードでの最小の矩形を表すクラスレベルであり、2 ~ 4 がこの値として設定される。 $density$  は中間ノードの最小の矩形に対する危険領域の密度である。例えば、ある中間ノードにおいて、クラス 3 の矩形が最小である場合、 $density$  はこの中間ノードのクラス 3 の矩形内に一つ下の階層のノードのクラス 3 の領域がどのくらい存在するかを表す。また、 $density$  は後述の RNR-tree に対する危険度に基づく問合せで用いられる。

##### 4.3 RNR-tree の特徴

RNR-tree の構造上の特徴を以下に挙げる。

(1) それぞれのノードが一つ以上の矩形を持つ

葉ノードは式 (13)、中間ノードは式 (15) でそれぞれ表現される。すなわち、葉ノードは 4 つの矩形を持ち、中間ノードはその階層に応じて 1 ~ 3 つの矩形を持つ。

(2) 中間ノードが持つ矩形の数は、その子ノードが持つ矩形の数より 1 だけ少ない

子ノードが葉ノードである場合、中間ノードはクラス 2、

3, 4の3つの矩形を持つ。子ノードが中間ノードである場合、階層に応じて以下の3通りに分けられる。子ノードがクラス2, 3, 4の3つの矩形を持つ場合には、中間ノードはクラス3, 4の2つの矩形を持つ。子ノードがクラス3, 4の2つの矩形を持つ場合、中間ノードはクラス4の矩形のみを持つ。子ノードがクラス4の矩形のみを持つ場合には、中間ノードもクラス4の矩形のみを持つ。

そして、ノードが持つ最小の矩形がその階層における危険領域に対応し、*level* が最小の矩形のクラスレベルを表す。また、式 (15) において中間ノードが矩形を持たないクラスの  $Rect(class \alpha)$  は有効な矩形情報を持たない。

(3) 中間ノードの各クラスの矩形は、子ノードの該当クラスの全ての矩形に対する MBR である

中間ノードがクラス3, 4の矩形を持つ場合、クラス3の矩形は子ノードのクラス3の矩形全てに対する MBR であり、クラス4の矩形は子ノードのクラス4の矩形全てに対する MBR である。

#### 4.4 RNR-tree の構成法

RNR-tree の、葉ノードの挿入・削除などの基本的な問合せは R-tree に準じる。しかし、それぞれのノードが持つ矩形が1つとは限らないため、中間ノードを作成する際の処理が若干異なる。以下で中間ノードがその子ノードに対してどのように作成されるかを2つの場合に分け、図を用いて示す。

(1) 矩形を複数持つノードからの中間ノード作成

まず、矩形を複数持つノードの親ノードとなる中間ノードの構成を示す。図4には3つの葉ノードが存在し、それぞれが4つの矩形を持つ..

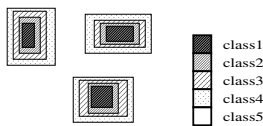


図4 3つの葉ノード  
Fig. 4 3 leaf nodes

この3つの葉ノードの親ノードに対応する中間ノードを作成する。最初に、作成する中間ノードが持つ最小の矩形を考える。この場合、子ノード(3つの葉ノード)の最小の矩形はクラス1であるので、作成する中間ノードの最小の矩形はクラス2となる。すなわち、作成する中間ノードにおいてはクラス2の矩形を危険領域として捉える。全葉ノードのクラス2の矩形に対する MBR を図4に加えると、図5のようになる。

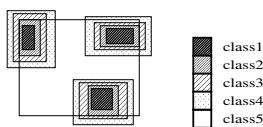


図5 図4の葉ノードに対する中間ノード作成途中経過  
Fig. 5 A process of making an intermediate node for 3 leaf node in Fig.4

同様に、中間ノードのクラス3, 4に対応する矩形を作成

する。すなわち、全子ノードのクラス3の矩形、クラス4の矩形に対する MBR をそれぞれ作成する。そして、作成した矩形をそれぞれ塗りつぶすと、図6のようになる。

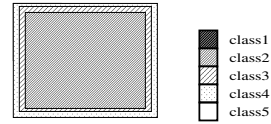


図6 図4の葉ノードに対する中間ノード作成結果  
Fig. 6 An intermediate node for 3 leaf node in Fig.4

図6では、中央にクラス2の矩形が表現されており、クラス3, 4の矩形がこれを階層的に包含する構造で表現されている。これらをそれぞれ作成する中間ノードのクラス2, 3, 4の矩形とする。以上のようにして作成した図6を3つの葉ノードの親ノードに対応する中間ノードとする。また、同様に矩形を複数持つ中間ノードに対しても、親ノードとなる中間ノードが作成される。

(2) 矩形を1つ持つノードからの中間ノード作成

次に、矩形を1つしか持たない中間ノードの親ノードとなる中間ノードの構成を示す。図7には3つの中間ノードが存在し、それぞれがクラス4の矩形のみを持つ。

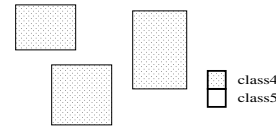


図7 3つの中間ノード  
Fig. 7 3 intermediate nodes

この3つの中間ノードの親ノードとなる中間ノードを作成する。最初に、作成する中間ノードが持つ最小の矩形を考える。子ノード(3つの中間ノード)の最小の矩形はクラス4の矩形である。矩形を複数持つノードからの中間ノード作成では、子ノードの最小の矩形より1段階だけ危険度が低いクラスの矩形を、作成する中間ノードが持つ最小の矩形とした。しかし、今回は子ノードとなる中間ノードがクラス4の矩形のみを持つので、作成する中間ノードの最小の矩形もクラス4のものとする。すなわち、作成する中間ノードにおいてもクラス4の矩形を危険領域として捉える。3つの子ノードのクラス4の矩形を全て包含する MBR を作成して塗りつぶすと図8のようになる。

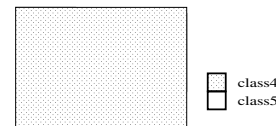


図8 図7の中間ノードに対する中間ノード作成結果  
Fig. 8 An intermediate node for 3 intermediate nodes in Fig.7

図8にはクラス4の矩形のみが表現されている。この矩形を作成する中間ノードのクラス4の矩形とする。以上のようにして作成した図8を3つの中間ノードの親ノードに対応する中間ノードとする。これは R-tree における中間ノード作成と全く同じ処理手順である。

#### 4.5 危険度に基づく問合せ

先に述べたように、危険度が低い領域から順次通行可能として避難所への経路を設定するためには、危険度があるレベル以上の領域を危険領域とする場合に、各地点が危険領域であるかどうかを判定する必要がある。この判定を行う問合せを「危険度に基づく問合せ」と呼び、式 (16) のように表す。RNR-tree がこの問合せをどのように処理するかを以下で述べる。

$$Query = ((x, y), level) \quad (16)$$

まず、危険度に基づく問合せにおいて、検索地点が危険領域かどうかを決定する手順を示す。この問合せでは、検索地点  $(x, y)$  と検索クラス  $level$  から、検索地点の危険度が検索レベルの危険度以上であれば検索地点は危険領域であるとし、検索レベルの危険度未満であれば危険領域ではないとする。具体的には、RNR-tree のルートノードを初期の検索ノードとして以下の手順で判定する。

- (1) 検索ノードの最小の矩形が検索地点を包含するか？  
Yes 2へ  
No 検索ノードでは危険領域と判定されない
- (2) 検索ノードの最小の矩形のクラスが検索レベルと等しいか？  
Yes 3へ  
No 検索ノードの全子ノードを順に検索ノードとして1へ
- (3) 検索ノードの最小の矩形と、その子ノードの最小の矩形のクラスが等しいか？  
Yes 検索ノードの全子ノードを順に検索ノードとして1へ  
No 4へ
- (4) 検索ノードの密度が一定値以上か？  
Yes 危険領域である  
No 5へ
- (5) 検索地点を包含する検索クラスの矩形を持つ、検索ノードの子ノードが存在するか？  
Yes 危険領域である  
No 検索ノードでは危険領域と判定されない

この手順で探索し、「危険領域である」と判定されれば検索地点は危険領域である。また、「検索ノードでは危険領域と判定されない」と判定された場合には、他に検索対象のノードが存在すればそれを次の検索ノードとして手順(1)から順に行う。検索対象のノードが存在しなければ、検索地点は危険領域ではない。

手順(1)では、中間ノードにおいて検索地点は危険領域ではないと判定する。これは、R-treeのある階層において、どの中間ノードにも包含されない地点は、その中間ノードより下位階層のどのノードにも包含されないと判定することとほぼ同意である。また、手順(4)では、中間ノードにおいて危険領域であると判定する。RNR-treeの中間ノードは図6のように表現されるため、ある危険度を閾値とした場合の危険領域は、その危険度を危険領域と捉える階層の最小の矩形に必ず包含される。したがって、最小の矩形に包含される場合のみ、検索地点は危険領域である可能性がある。この可能性の大小は、各中間ノードが保持するノード内密度  $density$  から判断できる。すなわち、 $density$ が高い場合、中間ノードに含まれる地点はその下位階層のノードにも含まれる可能性が高く、この逆も成り立

つ。この性質を用いて、可能性がある程度高いと判断した場合、検索地点が危険領域であると判定する。この2つの判定により、検索地点が危険領域である場合、危険領域ではない場合の双方において、必ずしも葉ノードに到達するまで検索する必要がなくなり、検索地点が危険領域であるかどうかを効率よく判定できる。

また、このように判定することで被災者ナビゲーション問題における経路設定も効率化される。これは、被災者ナビゲーション問題において、ノード内密度  $density$ が高い中間ノードの下位階層を探索していくことは非効率的であることによる。ノード内密度が高い中間ノードには、その最小の矩形内にその下位階層のノードが多く存在する。このため、このような中間ノードに含まれる地点を葉ノードまで検索するとその地点は危険領域である可能性が高い。よって、その地点を通過する経路は安全な避難経路とならない。また、検索地点が危険領域ではない場合でも、ノード内密度が高いと、検索地点の近くには多くの危険領域が存在すると考えられる。このとき、検索地点を通過する避難経路が通過することとなる、検索地点近くの点は危険領域である可能性が高い。よって、検索地点を通過する安全な避難経路が存在しないために、検索結果が有効に使われないことが多い。以上より、本問題において、ノード内密度が高い中間ノードの下位階層を検索することは、安全な避難経路が存在する見込みが薄い領域で経路探索をすることに相応し、非効率的となることが多い。先に述べた手順(4)では、検索ノードのノード内密度  $density$ を調べ、この値が一定値以上である場合には危険領域とみなし、その下位階層での検索を打ち切る。これにより、避難経路が存在する見込みが薄い領域での経路探索を打ち切ることができ、結果として効率的に経路設定が可能となる。

## 5. 避難経路設定

### 5.1 避難経路設定法

我々は各被災者に対する避難経路設定を、被災者の現在位置から避難所への最短経路を求める問題の変形とみなす。最短経路を求めるアルゴリズムとしては、Dijkstra法がよく知られている。Dijkstra法では、道路ネットワーク上のノード間の移動コストが与えられ、探索開始ノードから移動可能なノードを順次展開することにより、全ノードへの最短経路が求められる。我々は、移動可能なノードの選択に、危険度による制約条件を追加することで、避難所への安全な最短経路を求める。

我々は、危険領域の管理方法が異なる3種類の経路設定法を定義する。手法Aでは、危険領域が動的に拡大することを考慮せず、既知の危険領域のみをR-treeで索引付けする。したがって、危険度に基づく問合せでは既知の危険領域に含まれる領域のみを危険領域とし、この領域以外を通行可能として避難経路を設定する。手法B、Cでは、3章に述べたように危険領域が動的に拡大することを考慮して予測を行うが、危険領域の索引付け方法が異なる。手法Bでは、予測された危険領域をクラス毎にR-treeで索引付けする。すなわち、予測された危険領域のクラス1~クラス4の矩形を、4つのR-treeで個別に索



引付けする．これに対し、手法 C では予測による危険領域を RNR-tree で索引付けする．そして、手法 B, C では、予測される危険領域情報を用いて以下のように経路を設定する．最初に、危険度が最も低いクラス 5 のみを通行可能とし、その他の領域を危険領域であり通過不能として避難経路を探索する．この際に避難経路が存在しない場合には、クラス 5 の領域に加えてクラス 4 の領域も通行可能として避難経路を探索する．それでも避難経路が存在しない場合には、同様にクラス 3 の領域、クラス 2 の領域と 1 つずつ通行可能な領域を加えて避難経路を探索し、可能な限り安全な経路を得る．

また、全ての手法において、Dijkstra 法における展開中のノードから他のノードへの移動コスト更新を、展開中のノードとの間を結ぶエッジが通行不能な領域を通過しないノードのみに行うことで、通行不能な領域を避けた経路とする．この際、エッジを一定間隔で分割したいいくつかの点を用いてエッジと領域との交差判定を行う．すなわち、全ての点が通行可能な領域に含まれれば、そのエッジは通行可能とする．また、手法 B, C では、危険領域を通過せずに各ノードまで到達可能かどうかを判定する際に、対象とする被災者の位置情報から各ノードまでの距離によって概算される移動所要時間が考慮される．例えば、既知の被災地の状況から 10 分後、30 分後、50 分後の状況を予測したとする．この場合、予測をした時点から 10 分以内に被災者が通過すると考えられる地点においては、10 分後の予測領域に基づき通過可能かどうかを判定する．これに対して、予測した時点から被災者が通過するまでに 10 ~ 30 分経過すると考えられる地点においては、30 分後の予測領域に基づき通過可能かどうかを判定する．同様に、30 分以上が経過すると考えられる地点においては、50 分後の予測領域に基づき通行可能かどうかを判定する．このように、移動所要時間から、各地点を分割して、それぞれに対応する予測領域に基づいて判定することにより、状況に即した経路設定が可能である．

## 5.2 被災者誘導コスト

我々は、避難経路の設定に必要な計算負荷を被災者誘導コストと呼び、以下のように定義する．

まず、危険度に基づく問合せにおける比較処理にコスト 1 が必要とみなす．先に述べたように、危険度に基づく問合せでは、少なくとも 1 つのノードを検索して検索地点が危険領域であるかどうかを判定する．したがって、危険度に基づく問合せを 1 回行うと、検索されたノードの数だけのコストが必要となる．避難所への経路設定には、多くの地点に対する危険度に基づく問合せがなされる．故に、1 回の問合せに対して最小で 1 生じるコストを、全ての問合せに対して足し合わせたものを 1 人の被災者に対して避難経路を 1 回設定するために必要となるコストとする．また、被災者が避難所に到達するまで、もしくは、被災者が危険領域に囲まれて行動不能となるまでに、避難経路情報が複数回設定される場合が存在する．この際には、各避難経路設定に必要なコストを算出し、その総和を 1 人の被災者に対しての被災者誘導コストとする．

被災者誘導コストは、危険領域情報を管理するデータベースからの情報取得に必要な計算負荷に対応する．すなわち、被災

者誘導コストを抑えることは、避難経路設定に必要な計算負荷を抑えることに対応する．したがって、経路を被災者誘導コストを抑えて設定することにより、限られた処理能力の中でより多くの被災者を誘導できる．

## 6. 実験

提案手法の有効性を示すために計算機上で評価実験を行った．まず、実験の環境設定を説明する．その後、実験結果を示しそれに対して評価及び考察する．

### 6.1 環境設定

被災地を、 $6\text{ km} \times 8\text{ km}$  に相当する矩形のマップとする．この中に交差点に相当する約 400 のノードと約 200 の建物を配置し、建物の中からランダムに 3 個の避難所を設定する．このマップにいくつかの危険領域をランダムな位置に与え、時間の経過と共に不確かさを持たせながら徐々に拡大させる．道路ネットワーク上のノードをランダムに 30 個選択して避難開始点とし、それぞれの避難開始点に、先に述べた 3 種類の経路設定法によって避難所への経路が与えられる 3 人の被災者を配置する．そして、それぞれに経路を設定することによって避難所へ誘導する．被災者はシミュレーションの 1 ステップあたり  $50\text{ m}$  移動可能とし、実験開始時及び 20 ステップが経過する毎に経路情報が更新される．また、経路情報に従って移動する際に行く手を危険領域に遮られた場合にも、経路情報が更新される．

以上の設定のもとで、実験の最初に与える危険領域数を変化させながら、各危険領域数に対して被災者の誘導実験を 100 回行った．すなわち、各危険領域数に対して 3000 人の被災者を誘導したことに相当する．

### 6.2 実験結果

図 9 に、3000 の被災者を誘導し、避難所に到達した被災者数を示す．図 9 において *methodA* は手法 A, *methodB* は手

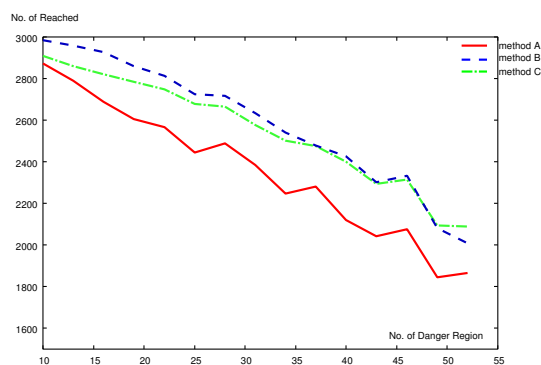


図 9 避難所に到達した被災者数

Fig. 9 Number of refugees reached a shelter

法 B, *methodC* は手法 C による結果を示す．全ての手法による結果が大局的に見て右下がりとなっている．これは、危険領域が増えると被災者が通過可能な領域が減少し、被災者は避難所に到達しにくくなることを示す．また、ほぼ全ての場合において最も多くの被災者を避難所へ導くのは手法 B であり、次に手法 C が僅差で続く．また、いずれも手法 A によって避難所へ誘導される被災者数を上回る．これは、先に述べた経路設定

の方法の違いを考慮すると妥当である。危険領域が少ない場合には手法 B と手法 C の差が大きくなり、手法 C の結果は良くない。これは、この場合には、実際には通過可能な領域での探索を打ち切ることが起こり易いためだと考えられる。危険領域が少ない場合、各中間ノードが持つ面積は大きくなり易い。大きな面積を持つ中間ノードと小さな面積を持つ中間ノードを比較すると、*density* の値が等しい場合、大きな面積を持つ中間ノードの方がその内部に安全な経路が存在する可能性が高い。したがって、危険領域が少ない場合には大きな面積を持つ中間ノードの下位階層で探索を打ち切ることによって安全な避難経路を潰す結果になり易く、これが手法 C の結果を悪化させていると考えられる。

図 10 は 3000 の被災者に対する被災者誘導コストを表す。図

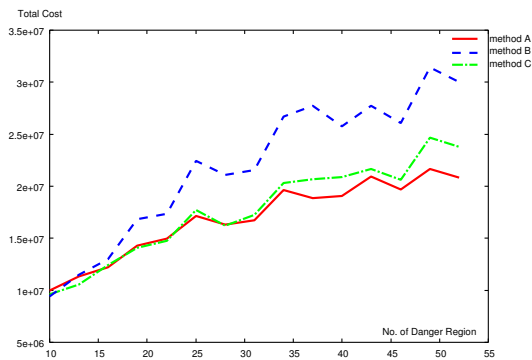


図 10 被災者誘導コスト

Fig. 10 Total costs

10において *methodA* は手法 A, *methodB* は手法 B, *methodC* は手法 C による結果を示す。全ての手法による結果が大局的に見て右上がりとなっている。手法 A の結果におけるコストの増加は、被災者の行く手が遮られることによって、経路の再探索回数が増えたことに因る。手法 B, C の結果におけるコストの増加には、これに加えて低い危険度を閾値とした経路探索によって避難経路を得られないことが増えたことによる増分が含まれる。危険領域が多く存在する場合には、危険度が低い領域のみを通行可能とした避難経路は存在しないことが多く、この場合先に述べた手順で避難経路を探索すると結果的に無用なコストを費す。手法 C においては、この無用なコストが *density* の値による打ち切りによって幾分抑えられる。故に、手法 B と手法 C で比較すると手法 C に必要なコストの方が小さくなり、危険領域が増えた際の増分も小さく抑えられていると考えられる。

図 11 は 3000 の被災者に対する誘導を行った際の、目的関数 (式 (12)) の値を表す。図 11 において *methodA* は手法 A, *methodB* は手法 B, *methodC* は手法 C による結果を示す。全ての手法による結果が大局的に見て右上がりとなっており、危険領域が増えると先に挙げた理由により被災者を一人導くために必要となるコストが大きくなることを示す。また、手法 C による結果はほぼ全ての場合において手法 A, B による結果を下回る。これは、被災者を一人避難所へ導くために必要となるコストが手法 C において最小となることを意味し、提案手法の有効性を裏付ける。

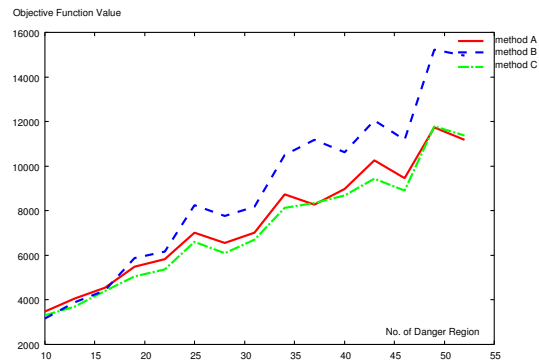


図 11 目的関数値

Fig. 11 Objective Function Value

## 7. まとめ

本稿では、各被災者に対して避難経路を設定することで避難所へ誘導する、被災者ナビゲーション問題を扱った。最初に、計算機上でのシミュレーション実験を行うために被災者ナビゲーション問題を形式化し、評価基準を関数として定義した。次に、避難経路の安全性と避難経路を設定可能な人数との間に存在するトレードオフ関係に対処するために、危険領域の変化予測と予測結果を索引付けする R-tree を拡張した木構造、RNR-tree について述べた。最後に、シミュレーション実験を行い、RNR-tree による危険領域管理の有効性を示した。

今後の課題は、RNR-tree に対する危険度に基づく問合せの改良である。本稿で述べた方法では、危険領域が少ない場合に安全な避難経路が存在する領域を危険領域と判定し易いため、実際には存在する安全な避難経路を得ることができない事態が生じる。具体的な解決策としては、*density* による検索打ち切り判定の際にそのノードの面積や危険度を用いた新たな判定条件を設定することが挙げられる。

## 文 献

- [1] 仲谷 善雄. 大規模災害に対する減災情報システム (前編), 情報処理学会誌, Vol. 45, No. 11, pp. 1164-1174, 2004.
- [2] 仲谷 善雄. 大規模災害に対する減災情報システム (後編), 情報処理学会誌, Vol. 45, No. 12, pp. 1255-1265, 2004.
- [3] 石田 亨, 中西英之, 高田司郎. デジタルシティにおける危機管理シミュレーション. システム制御情報学会誌, Vol. 46, No. 9, pp. 524-531, 2002.
- [4] 田所諭, 北野宏明, 高橋友一, 松野文俊, 竹内郁雄. RoboCup-Rescue: 情報科学の緊急災害対応問題への挑戦, 情報処理学会誌, Vol. 41, No. 4, pp. 412-418, 2000.
- [5] Antonm Guttman. R-TREES: A Dynamic Index Structure for Spatial Searching, Proc. of ACM SIGMOD 1984, pp 47-57, 1984.
- [6] Jinfeng Ni, Chinya V. Ravishankar, and Bir Bhanu. Probabilistic Spatial Database Operation, Proc. of 8th International Symposium LNCS2750, pp 140-158, 2003.
- [7] Simonas Saltenis, Christian S. Jensen, Scott T. Leutenegger, and Mario A. Lopez. Indexing the positions of continuously moving objects, Proc. of ACM SIGMOD 2000, pp 331-342, 2000.