動的環境における障害物領域予測法

正員 内田 雅文 (青山学院大)

正員 井出 英人 (青山学院大)

正員 横山 修一 (工学院大)

Forecast Method of Obstacle Region in Dynamic Environment

Masafumi Uchida, Member, Hideto Ide, Member (Aoyama Gakuin University), Syuichi Yokoyama, Member (Kogakuin University)

In this paper, we study ways to avoid obstacles in case of action of an autonomous robot in a dynamic environment. We considered that a problem of obstacle avoidance result in one of the robustness against environment changes in a motion plan. Supposing that it is able to forecast of a working environment of the future and to plan ahead motion for the future, it is possible that a robot takes action in response to environment changes. As a result, we may hope for an improvement in the robustness of a motion plan.

This study especially take notice of a path of a mobile robot in a motion plan of an autonomous robot. We considered that a transfer from a pattern of environment information to one of plan is a path plan and studied ways to process an occupied region pattern of a working environment in parallel. This paper proposes a technique for to forecast of environment changes and to plan ahead motion for the future.

キーワード：自律ロボット，経路計画，障害物回避，並列処理，予測

1. はじめに

作業環境内に移動障害物が存在する場合や複数のロボットにより協調行動する場合，移動するロボットには他の移動体を回避する行動が必要になる。いわゆる動的環境下の障害物回避である。移動体(移動障害物や他のロボット)の行動を考慮しながら行動計画することは，静的環境下の障害物回避に比べ，非常に困難な点が多い。しかし，すでにさまざまなアルゴリズムが提案されている^{(1)~(3)}。ロボット間に通信手段を設ける方法⁽⁴⁾や他のロボットの軌跡をもとに時空空間内で経路計画する方法⁽⁵⁾，あらかじめロボット間に交通ルールを設ける方法⁽⁶⁾などである。ただし，これらの方法は各移動体の位置，姿勢および運動が特定できることを仮定するものである。位置，姿勢，運動の特定は，環境内にあるすべての移動体について，各々を観察，追跡することによって行われるが，移動体数の増加とともに実時間性は損なわれてしまう。各ロボットが看板を掲げ，自分の所在，状況を表示する方法⁽⁷⁾も提案されているが，移動障害物には適用できない。

ロボットビジョンの分野では，視覚情報の実時間処理として動き検出，動作理解，移動体認識等が行われている。複数の移動体に対してそれぞれ注視点をおき，並列的に各物体の状況を解析しようとする試みもある⁽⁸⁾⁽⁹⁾。しかし，

本研究では動的環境下の障害物回避の場合，各移動体それぞれを観察し，各々について位置，姿勢，運動を特定する必要はないものと考えている。すなわち，移動体はすべて障害物の一部と見なし，ロボットが侵入できない領域(以後，占有領域と呼ぶ)のパターンとして一括して扱う。占有領域パターンは同時にロボットが侵入可能な領域のパターンを示すものであるから，侵入可能な領域さえ特定できれば，占有領域パターンは容易に得ることができる。したがって，従来のように個々の移動体の状況の特定しなくても，障害物回避は可能と考える。ただし，移動体を回避する場合，移動体の将来の行動を予測することは重要であり，より安全かつ効果的な回避行動の計画が可能になる。移動体の「仮の位置」なる概念を導入した方法⁽¹⁰⁾や将来の移動体の位置を確率密度関数として予測する方法⁽¹¹⁾なども提案されているが，これらも前述の仮定のもとで論じられたものである。

本研究ではこれまで障害物回避の経路計画を一種のパターン変換として扱ってきた^{(12)~(17)}。そこで本稿では作業環境を占有領域のパターンとしてとらえて，過去のパターンから将来のパターンを予測する方法を提案する。将来のパターン，すなわち環境変化を予測することによって計画の環境変化に対する頑健性(以後，これをロバストネスと呼ぶ)の向上が期待できる。本稿で提案する方法では，各移

動体の位置、姿勢および運動を特定する必要がなくなり、未知の移動体にも対応できる。また、作業環境の複雑さを考慮する必要もない。筆者は文献(17)において正方格子状に配置した情報処理素子(以後、素子群と呼ぶ)による経路計画の並列アルゴリズムを提案している。そこで本稿で提案する占有領域パターンの予測方法も、この正方格子状配置の素子群なる概念に従う並列アルゴリズムとした。その結果として実時間性に優れた占有領域パターンの予測が可能になるとともに、将来の占有領域パターンの予測と経路計画から現在の経路計画までを素子群の並列処理によって一貫して行うことが可能になった。

2. 素子群による経路計画法の概要

正方格子状配置の素子群による並列アルゴリズムの概念を文献(17)の経路計画法を例に以下に示す。

障害物が置かれた2次元作業環境を $n \times n$ の正方格子状に領域分割し、離散表現する。分割された各領域の格子上の位置 $(i, j)(i, j=1, 2, \dots, n)$ における時刻 t の環境を $e_{(i,j)}(t)$ と表わす。 $e_{(i,j)}(t)$ は障害物の有無、ロボット、ロボットの移動目標のいずれかを表わし、(1)式とする。ただし、ロボットおよび移動目標の正方格子上の位置は、それぞれ $(i_R, j_R), (i_G, j_G)$ とする。以後、 $e_{(i,j)}(t)$ を環境要素、 (i, j) を $e_{(i,j)}(t)$ における位置要素と呼ぶことにする。

$$e_{(i,j)}(t) = \begin{cases} 3, & (i, j) = (i_G, j_G) \\ 2, & (i, j) = (i_R, j_R) \\ 1, & (i, j) \in \text{obstacle} \\ 0, & \text{otherwise (non-obstacle)} \end{cases} \quad \dots\dots\dots (1)$$

各環境要素 $e_{(i,j)}(t)$ に対して $e_{(i,j)}(t)$ を処理する図1のような情報処理素子 $\epsilon_{(i,j)}$ を考える。素子 $\epsilon_{(i,j)}$ の時刻 t にお

図1 環境要素 $e_{(i,j)}$ と素子 $\epsilon_{(i,j)}$ の関係
Fig. 1. The Relation Between Environment Factor $e_{(i,j)}$ and Element $\epsilon_{(i,j)}$.

ける出力を $z_{(i,j)}(t)$ と表わす。このとき素子 $\epsilon_{(i,j)}$ の入出力関係を図2に示す。素子 $\epsilon_{(i,j)}$ は時刻 t において、隣接する4つの素子 $\epsilon_{(i-1,j)}, \epsilon_{(i+1,j)}, \epsilon_{(i,j-1)}, \epsilon_{(i,j+1)}$ の出力 $z_{(i-1,j)}(t-1), z_{(i+1,j)}(t-1), z_{(i,j-1)}(t-1), z_{(i,j+1)}(t-1)$ を $\epsilon_{(i,j)}$ とそれぞれの素子間を連結するリンク $L_{(i,j)}(t-1; i-1, j), L_{(i,j)}(t-1; i+1, j), L_{(i,j)}(t-1; i, j-1), L_{(i,j)}(t-1; i, j+1)$ を介して受け取り、 $z_{(i,j)}(t)$ を出力するものと仮定する。ただし、リンクは任意の $\epsilon_{(k,l)}, (k, l=1, 2, \dots, n)$ に対して(2)式を満足するものとする。

$$L_{(i,j)}(t; k, l) = L_{(k,l)}(t; i, j) = \begin{cases} 1 \\ 0 \end{cases} \quad \dots\dots\dots (2)$$

ここで便宜上、位置要素 (i, j) に対して隣接する位置要素 $(i-1, j), (i+1, j), (i, j-1), (i, j+1)$ を(3)式のように (I, J) で表わすことにする。

$$(I, J) = \begin{cases} (i-1, j) \\ (i+1, j) \\ (i, j-1) \\ (i, j+1) \end{cases} \quad \dots\dots\dots (3)$$

$\epsilon_{(I,J)}$ の出力 $z_{(I,J)}(t-1)$ に対する $\epsilon_{(i,j)}$ が受け取る入力を $L_{(i,j)}(t-1; I, J) \cdot z_{(I,J)}(t-1)$ とする。このとき経路は、 $L_{(i,j)}(t; I, J)$ および $z_{(i,j)}(t)$ をそれぞれ更新し、

$$L_{(i,j)}(t; I, J) = 1 \quad \dots\dots\dots (4)$$

となる位置要素 (i, j) の列(以後、これを経路列と呼ぶ)として得る。ただし、経路列が (i_R, j_R) から (i_G, j_G) までの経路を表わすためには、次の3点が条件になる。

- ① (i_R, j_R) が経路の始点、 (i_G, j_G) が経路の終点であること。
- ② $(i_R, j_R), (i_G, j_G)$ 以外に端点がないこと。
- ③ 分岐路がないこと。

条件①を満足するためには、 $\epsilon_{(i_R, j_R)}$ から出るリンク $L_{(i_R, j_R)}(t; I_R, J_R)$ の総和、および $\epsilon_{(i_G, j_G)}$ から出るリンク $L_{(i_G, j_G)}(t; I_G, J_G)$ の総和がともに1である必要がある。条件②を満足するためには、経路列における隣り合う位置要素間の距離は1である必要がある。したがって経路列中の位置要素を (i, j) とするとき、 (i, j) に隣り合う位置要素は (I, J) で表わされる4つの要素の中のいずれかである。条件③を満足するためには、 $(i, j) \neq (i_R, j_R)$ かつ $(i, j) \neq (i_G, j_G)$ を満たす $\epsilon_{(i,j)}$ から出るリンク $L_{(i,j)}(t; I, J)$ の総和が、経路列に (i, j) が含まれる場合2、それ以外は0で

図2 素子 $\epsilon_{(i,j)}$ の入出力
Fig. 2. Input/Output of Element $\epsilon_{(i,j)}$.

ある必要がある。

条件①～③を満足するように $L_{(i,j)}(t; I, J)$ および $z_{(i,j)}(t)$ を更新して有効な経路列を得る。各素子のリンク $L_{(i,j)}(t; I, J)$ および出力 $z_{(i,j)}(t)$ の更新が、素子群全体としては環境情報を並列的に処理し、経路を計画している。

以上、素子群による経路計画法において用いた記号を整理すると次のようになる。

t : 時刻

n : 正方格子の一辺の格子数

(i, j) : 格子上的位置, 位置要素

(i_R, j_R) : ロボットの位置要素

(i_G, j_G) : ロボットの移動目標の位置要素

$e_{(i,j)}(t)$: 時刻 t における (i, j) の環境要素

$\varepsilon_{(i,j)}$: $e_{(i,j)}(t)$ を処理する情報処理素子

$L_{(i,j)}(t; k, l)$: 時刻 t において $\varepsilon_{(i,j)}$ と $\varepsilon_{(k,l)}$ を連結するリンク

$z_{(i,j)}(t)$: 時刻 t における $\varepsilon_{(i,j)}$ の出力

(I, J) : (i, j) の四方に隣接する位置要素

3. 素子群による占有領域パターンの予測

〈3・1〉 占有領域パターンの予測 位置要素 (i, j) で示される領域の中心位置座標を $x_{(i,j)}$ とするとき, $x_{(i,j)}$ を (i, j) の位置と呼ぶことにする。ここで環境要素 $e_{(i,j)}(t)$ を(5)式とする。

$$e_{(i,j)}(t) = \begin{cases} 1, & x_{(i,j)} \in \text{occupied region (obstacle)} \\ 0, & \text{otherwise (non-obstacle)} \end{cases} \quad (5)$$

2次元作業環境は、環境要素の行列 $E(t)$ によって(6)式のように離散表現される。ただし、行列を記す場合、各要素の添字は (i, j) を ij のように省略して記し、以後これに準拠することにする。

$$E(t) = \begin{bmatrix} e_{11}(t) & e_{12}(t) & \cdots & e_{1j}(t) & \cdots & e_{1n}(t) \\ e_{21}(t) & e_{22}(t) & \cdots & e_{2j}(t) & \cdots & e_{2n}(t) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ e_{i1}(t) & e_{i2}(t) & \ddots & e_{ij}(t) & \ddots & e_{in}(t) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ e_{n1}(t) & e_{n2}(t) & \cdots & e_{nj}(t) & \cdots & e_{nn}(t) \end{bmatrix} \quad (6)$$

このとき占有領域パターンの予測は、時刻 t において $E(t+h)$, ($h=1, 2, \dots$) を予測することに相当する。

各位置要素に対して $E(t)$ を入力データとして受け取る情報処理素子を配置する。すなわちここに、正方格子状配置の素子群が仮定される。各素子の出力関数 $f_{(i,j)}$ は入力 $E(t)$ に対して(7)式とする。

$$f_{(i,j)}(E(t)) = \begin{bmatrix} e_{(i,j)}(t) \\ \bar{e}_{(i,j)}(t+h) \end{bmatrix} \quad (7)$$

各素子は時刻 t において $e_{(i,j)}(t)$ を出力すると同時に、時刻 $t+h$ における出力の予測値 $\bar{e}_{(i,j)}(t+h)$ を出力するものと仮定する。ただし、 $\bar{}$ は予測値を表わしている。素

子群としては、図3のように入力 $E(t)$ に対して $E(t)$ と(8)式で表わされる予測値 $\bar{e}_{(i,j)}(t+h)$ の行列 $\bar{E}(t+h)$ を出力することになる。

$$\bar{E}(t+h) = \begin{bmatrix} \bar{e}_{11}(t+h) & \bar{e}_{12}(t+h) & \cdots & \bar{e}_{1j}(t+h) & \cdots & \bar{e}_{1n}(t+h) \\ \bar{e}_{21}(t+h) & \bar{e}_{22}(t+h) & \cdots & \bar{e}_{2j}(t+h) & \cdots & \bar{e}_{2n}(t+h) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \bar{e}_{i1}(t+h) & \bar{e}_{i2}(t+h) & \ddots & \bar{e}_{ij}(t+h) & \ddots & \bar{e}_{in}(t+h) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \bar{e}_{n1}(t+h) & \bar{e}_{n2}(t+h) & \cdots & \bar{e}_{nj}(t+h) & \cdots & \bar{e}_{nn}(t+h) \end{bmatrix} \quad (8)$$

したがって $E(t+h)$ の予測 ($\bar{E}(t+h)$ の算出) は、各素子における $e_{(i,j)}(t+h)$ の予測 ($\bar{e}_{(i,j)}(t+h)$ の算出) に帰着する。

各位置要素の位置 $x_{(i,j)}$ は定数である。したがって各素子は(9)式のような定数行列 $\Lambda_{(i,j)}$ をあらかじめ備えることができる。ただし、 $|x_{(k,l)}x_{(i,j)}|$ は $x_{(k,l)}$, $x_{(i,j)}$ 間の距離を表わす。

$$\Lambda_{(i,j)} = \begin{bmatrix} |x_{11}x_{ij}| & |x_{12}x_{ij}| & \cdots & |x_{1i}x_{ij}| & \cdots & |x_{1n}x_{ij}| \\ |x_{21}x_{ij}| & |x_{22}x_{ij}| & \cdots & |x_{2i}x_{ij}| & \cdots & |x_{2n}x_{ij}| \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ |x_{k1}x_{ij}| & |x_{k2}x_{ij}| & \ddots & |x_{ki}x_{ij}| & \ddots & |x_{kn}x_{ij}| \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ |x_{n1}x_{ij}| & |x_{n2}x_{ij}| & \cdots & |x_{ni}x_{ij}| & \cdots & |x_{nn}x_{ij}| \end{bmatrix} \quad (9)$$

$e_{(i,j)}(t+h)$ を予測するにあたり、位置 $x_{(i,j)}$ と占有領域に含まれた位置 $x_{(k,l)}|_{e_{(k,l)}(t)=1}$ 間の距離に注目する。 $E(t)$ と $\Lambda_{(i,j)}$ を用いると、 $x_{(i,j)}$ と $x_{(k,l)}|_{e_{(k,l)}(t)=1}$ 間の距離の最小値 $\lambda_{(i,j)}(t)$ が(10)式として算出でき、

$$\lambda_{(i,j)}(t) = \min_{k,l} (\Lambda_{(i,j)}^{k,l} \times E(t)^{k,l}) \quad (10)$$

時刻 t までに時系列データ

$$\{\dots, \lambda_{(i,j)}(t-2), \lambda_{(i,j)}(t-1), \lambda_{(i,j)}(t)\}$$

が得られる。ただし、 $\Lambda_{(i,j)}^{k,l}$, $E(t)^{k,l}$ は、行列 $\Lambda_{(i,j)}$ および $E(t)$ の k 行 l 列目の要素、 $\min$ は最小値を表わす。

時系列データ $\{\dots, \lambda_{(i,j)}(t-2), \lambda_{(i,j)}(t-1), \lambda_{(i,j)}(t)\}$ より $\lambda_{(i,j)}(t+h)$ の予測値 $\bar{\lambda}_{(i,j)}(t+h)$ を算出したときの(11)式

図3 素子群の入出力

Fig. 3. Input/Output of Elements.

を $e_{(i,j)}(t+h)$ の予測値 $\bar{e}_{(i,j)}(t+h)$ とする。

$$\bar{e}_{(i,j)}(t+h) = \begin{cases} 1, & \bar{\lambda}_{(i,j)}(t+h) < 1 \\ 0, & \text{otherwise} \end{cases} \quad \dots\dots\dots (11)$$

すなわち絶えず変化する占有領域の先端の位置が時刻 $t+h$ において $x_{(i,j)}$, ((3)式参照)よりも近くにあると予測されれば, $e_{(i,j)}(t+h)$ は 1, それ以外は 0 と予測する。したがって $e_{(i,j)}(t+h)$ の予測は, $\lambda_{(i,j)}(t+h)$ の予測 ($\bar{\lambda}_{(i,j)}(t+h)$ の算出)に帰着する。

本研究では突発的な事象による環境変化は考慮していない。人が移動体を回避するときも突発的な変化への対応は困難であり, この条件設定は妥当なものと考えられる。本稿では, 各時刻における時系列パターンは過去の時系列パターンの延長上に現われるものと仮定し, $\lambda_{(i,j)}(t+h)$ の予測において自己回帰モデルを適用した。

(3・2) 時系列パターンの予測 素子 (i, j) における時刻 t までの時系列データを $\Phi_{(i,j)}$ で表わし, (12)式とする。

$$\Phi_{(i,j)} = \{\lambda_{(i,j)}(t-\eta+1), \lambda_{(i,j)}(t-\eta+2), \dots, \lambda_{(i,j)}(t-2), \lambda_{(i,j)}(t-1), \lambda_{(i,j)}(t)\} \quad \dots\dots (12)$$

$\Phi_{(i,j)}$ は η 次元ベクトルである。各素子は図 4 のように時刻 t において時系列データ $\Phi_{(i,j)}$ を蓄えているものと仮定する。 $\lambda_{(i,j)}(t+h)$ の予測にあたり, 時系列パターンは時間間隔 h で取り出されたデータの時系列において回帰するものと仮定する。この仮定に基づき, $\Phi_{(i,j)}$ から時間間隔 h で取り出した (13)~(17)式のような m 次元ベクトル $\phi_{(i,j)}^{[1]}, \phi_{(i,j)}^{[2]}, \dots, \phi_{(i,j)}^{[3]}, \dots, \phi_{(i,j)}^{[\eta-1-(m-1)h]}, \phi_{(i,j)}^{[\eta-(m-1)h]}$ より (18)式における回帰係数 $b_1, b_2, \dots, b_{m-1}$ を算出する。

$$\phi_{(i,j)}^{[1]} = \{\lambda_{(i,j)}(t-(m-1) \cdot h), \lambda_{(i,j)}(t-(m-2) \cdot h), \dots, \lambda_{(i,j)}(t-2h), \lambda_{(i,j)}(t-h), \lambda_{(i,j)}(t)\} \quad \dots\dots (13)$$

$$\phi_{(i,j)}^{[2]} = \{\lambda_{(i,j)}(t-(m-1) \cdot h-1), \lambda_{(i,j)}(t-(m-2) \cdot h-1), \dots, \lambda_{(i,j)}(t-2h-1), \lambda_{(i,j)}(t-h-1), \lambda_{(i,j)}(t-1)\} \quad \dots\dots (14)$$

$$\phi_{(i,j)}^{[3]} = \{\lambda_{(i,j)}(t-(m-1) \cdot h-2), \lambda_{(i,j)}(t-(m-2) \cdot h-2), \dots, \lambda_{(i,j)}(t-2h-2), \lambda_{(i,j)}(t-h-2), \lambda_{(i,j)}(t-2)\} \quad \dots\dots (15)$$

$$\vdots$$

$$\phi_{(i,j)}^{[\eta-1-(m-1)h]} = \{\lambda_{(i,j)}(t-\eta+2), \lambda_{(i,j)}(t-\eta+2+h), \dots$$

$$\dots, \lambda_{(i,j)}(t-\eta+2+(m-3)h), \lambda_{(i,j)}(t-\eta+2+(m-2)h), \lambda_{(i,j)}(t-\eta+2+(m-1)h)\} \quad \dots\dots (16)$$

$$\phi_{(i,j)}^{[\eta-(m-1)h]} = \{\lambda_{(i,j)}(t-\eta+1), \lambda_{(i,j)}(t-\eta+1+h), \dots, \lambda_{(i,j)}(t-\eta+1+(m-3)h), \lambda_{(i,j)}(t-\eta+1+(m-2)h), \lambda_{(i,j)}(t-\eta+1+(m-1)h)\} \quad \dots\dots (17)$$

$$\lambda_{(i,j)}(t) = \sum_{p=1}^{m-1} b_p \cdot \lambda_{(i,j)}(t-p \cdot h) \quad \dots\dots\dots (18)$$

ただし, (13)~(17)式および(18)式より Φ の次元, すなわち時系列データ数 η は (19)式を満たす必要がある。

$$\eta \geq (m-1) \cdot (h+1) \quad \dots\dots\dots (19)$$

ここで $\lambda_{(i,j)}(t+h)$ の予測値 $\bar{\lambda}_{(i,j)}(t+h)$ を (20)式とする。

$$\bar{\lambda}_{(i,j)}(t+h) = \sum_{p=1}^{m-1} b_p \cdot \lambda_{(i,j)}(t-(p-1) \cdot h) \quad \dots\dots\dots (20)$$

$\bar{\lambda}_{(i,j)}(t+h)$ から (11)式に従い $\bar{e}_{(i,j)}(t+h)$ を算出すると, 素子群は (8)式のように $E(t+h)$ を予測することになる。

以上, 素子群による占有領域パターンの予測において用いた記号を整理すると次のようになる。

- $x_{(i,j)}$: (i, j) で示される領域の中心位置座標。 (i, j) の位置
- E : 環境要素の行列。 2 次元作業環境
- ij : 行列を記す場合の添字 (i, j) の省略形
- h : 時刻間隔
- $f_{(i,j)}$: (i, j) にある素子の出力関数
- $\bar{}$: 予測値
- $\Lambda_{(i,j)}$: $x_{(i,j)}$ から任意の (k, l) の位置 $x_{(k,l)}$ までの距離の行列
- $\lambda_{(i,j)}$: $x_{(i,j)}$ から占有領域にある $x_{(k,l)}$ までの距離の最小値
- $\Lambda_{(i,j)}^{kl}$: 行列 $\Lambda_{(i,j)}$ の k 行 l 列目の要素
- $E(t)^{kl}$: 行列 E の k 行 l 列目の要素
- min: 最小値
- $\Phi_{(i,j)}$: $\lambda_{(i,j)}$ の時系列データ
- η : $\Phi_{(i,j)}$ のデータ数
- m : $\Phi_{(i,j)}$ から時間間隔 h で取り出したデータ数
- $\phi_{(i,j)}^{[p]}$: $\Phi_{(i,j)}$ から取り出した m 次元時系列データ
- b_p : p 番目の回帰係数

4. 計算機シミュレーションの結果および考察

図 5 のように占有領域が周期 30 単位時間で円運動する作業環境を考える。この作業環境を 8×8 , ($n=8$) で離散表現し, 6 単位時間先 ($h=6$) の作業環境の予測を計算機シミュレーションした。図 6 は位置要素 (1, 1) にある素子が時刻 $t-6$ において予測した $\bar{\lambda}_{(1,1)}(t)$ を示している。図 5 のような作業環境の場合, $\lambda_{(1,1)}(t)$ は周期 30 の正弦波となり, 図 6 の太線のようになる。 $\lambda_{(1,1)}(t)$ の予測にあたり, Φ のデータ数 η の設定が問題になる。 η を増やせば予測値の確度が向上することは明らかであるが, η は各素子が蓄えるデータ数であり, 可能な限り少数に抑えたい。この

図 4 素子の入出力と機能
Fig. 4. Input/Output and Function of An Element.

図 5 シミュレーションのための作業環境 1
Fig. 5. Working Environment No. 1 for Simulation.

図 6 $\lambda_{(1,1)}(t)$ の予測結果
Fig. 6. Forecast of $\lambda_{(1,1)}(t)$.
Line: $\hat{\lambda}_{(1,1)}$, $m=2, 4, 6, 8$, $h=6$, $\eta=(m-1) \cdot (h+1)$
Bold: $\lambda_{(1,1)}$

シミュレーションでは本予測法の特性を明らかにするため η は (19) 式の最小値 ($\eta=(m-1) \cdot (h+1)$) とした。図 6 の実線は $m=2, 4, 6, 8$ のときの $\hat{\lambda}_{(1,1)}(t)$ を示している。 $m=2$ の場合、(20) 式より $\lambda_{(1,1)}(t-6)$ がそのまま出力される。したがって $\hat{\lambda}_{(1,1)}(t)$ の波形は $\lambda_{(1,1)}(t)$ に対して位相が遅れている。 $m=4$ では、位相遅れは多少解消されているものの波形に乱れが生じている。周期 30 の環境変化をデータ数 $\eta=21$ の ϕ で捕らえようとしたことが原因である。 $m=6$ では位相遅れはほとんど解消されている。 $\eta=35$ となり、周期 30 の環境変化は ϕ 内に表現されているはずであるが、まだ多少波形に乱れを残している。作業環境を $n=8$ で離散表現する際の丸め誤差が原因であり、 n を大きくとることにより解消できるものである。 $m=8$ ($\eta=49$) では $\lambda_{(1,1)}(t)$ をほぼ完全に予測している。図 7 は蓄えるデータ数を十分に多くとったとき ($\eta=180$) の時刻 $t-6$ における $\hat{\lambda}_{(1,1)}(t)$ を示している。ただし、環境変化の周期は 180 単位時間である。 $\eta=180$ より周期 180 の環境変化は $m=2, 4, 6, 8$ のどの場合においても ϕ 内に捕らえている。 $m=2$ では位相が完全にずれており、予測が全く行われぬ。 m が大きくなるとともに $\lambda_{(1,1)}(t)$ の波形に近づき、予

図 7 $\lambda_{(1,1)}(t)$ の予測結果
Fig. 7. Forecast of $\lambda_{(1,1)}(t)$.
Line: $\hat{\lambda}_{(1,1)}$, $m=2, 4, 6, 8$, $h=6$, $\eta=180$
Bold: $\lambda_{(1,1)}$

測精度が上がっている。ただし、 η は有限であるから、無闇に予測精度を上げてみたところで環境変化が十分に表現されるとはいえない。したがって、 m は η および経路計画の環境変化に対するロバストネスを考慮して決定していく必要がある。以上、 η と m はハードウェア(ロボット)が環境変化をどこまで許容できるかによって決まるパラメータである。

図 8a は時刻 $t, t+3, t+7, t+11$ における作業環境である。32×32 で表現されたこの環境は、2 つの移動体 A, B が存在する動的環境である。図 9 のように移動体 A は下から上へ単位時間当たり格子 1 マスの速さ、B は右から左へ A の 1/2 の速さで移動している。この環境において各時刻ごとに素子群による経路計画を行った場合、図中実線のようになった。ただし、ロボットは単位時間当たり最大 2 マスの速さで移動できるものとした。一方、図 8b は時刻 t において $t+3, t+7, t+11$ の作業環境を予測し ($h=3, 7, 11$)、経路計画を行った結果である。この予測において前述のパラメータ η と m は

$$\eta=300, \quad m=21$$

とした。図は環境変化をほぼ完全に予測していることを示しており、経路計画において十分利用できるものになっている。予測結果をもとに経路計画すると図 8b の実線のようになった。得られた経路は各時刻ごとに得られる経路とほぼ同等である。

以上のような結果より、素子群を利用した占有領域の予測法および経路計画法によると、時刻 t の時点で将来の経路計画が予測できることを示すものである。

5. む す び

本研究は、自律型ロボットが動的な作業環境において行動する際、障害物をいかにして回避するかを検討したもの

図 8 占有領域パターンの予測結果および経路計画

Fig. 8. Forecast of Occupied Region Pattern and Path Plan.

図 9 シミュレーションのための作業環境 2. 移動障害物 A と B の関係

Fig. 9. Working Environment No. 2 for Simulation. Relation Between Mobile Obstacle A and B.

である。ロボットの行動は自ら構築した計画に従うものと仮定すると、障害物回避は行動計画における環境変化に対するロバストネスの問題に帰着する。本稿ではこの問題に対する方策として環境変化を予測する方法を採用した。将来の作業環境を予測し、行うべき行動の準備があらかじめできれば、環境変化に即座に対応でき、ロバストネスの向上が可能になる。そこで本稿では環境変化を予測し、将来行うべき行動の計画を前もって構築する方法を提案した。

環境変化の予測によるロボットの行動のロバストネスに関する検討は従来より報告されている。ただし、そのほと

んどは環境内に想定した各移動体の位置、姿勢および運動が特定できることを仮定している。実環境において実時間で環境内に存在するすべての移動体の位置、姿勢、運動を特定することは困難である。

本研究ではロボットの行動計画の中でも特に移動ロボットの経路計画を取り上げた。経路計画を環境情報から計画へのパターン変換として扱い、作業環境の占有領域パターンを並列的に処理する方法を検討した。提案した予測方法は、素子群を利用して過去の占有領域パターンから将来のパターンを推定するものである。並列アルゴリズムに基づくことから、経路計画の実時間性を損なうことはない。また、作業環境をパターンとして扱うことから、環境内の個々の物体、すなわち個々の移動体に着目する必要がない。以上、占有領域パターンの予測法は、経路計画法と組み合わせることにより、素子群による経路計画の予測方法として構成することができた。

今後は、実際の移動ロボットシステムへ適用するとともに、予測される将来の経路計画と各時刻における計画との比較、評価方法を検討していく予定である。

(平成 8 年 3 月 25 日受付, 同年 7 月 18 日再受付)

文 献

- (1) 劉・有本:「局所センサ情報を用いた未知環境における複数の移動ロボットの運動計画法」, 日本ロボット学会誌, Vol. 10, No. 2, p. 216 (1992)
- (2) 太田・新井:「群秩序生成可能な追従戦略を用いた複数移動ロボット系の動作計画」, 日本ロボット学会誌, Vol. 12, No. 4, p. 603 (1994)
- (3) 山内・石川・加藤:「2 台の移動ロボットを用いた同一対象物の位置検出について」, 日本ロボット学会誌, Vol. 12, No. 1, p. 112 (1994)
- (4) S. Premvuti & S. Yuta: "Consideration on the cooperation of multiple autonomous mobile robots", International Workshop on Intelligent Robots and Systems (IROS'90), p. 59 (1990)
- (5) K. Kant & S. Zucker: "Planning collision-free trajectories in time-varying environment: a two level approach", Proc. of Int. Conf. on Robotics and Automation 1988, p. 1644 (1988)
- (6) 加藤・西山・武野:「交通ルールの適用による複数の移動ロボットの協調行動」, 日本ロボット学会誌, Vol. 12, No. 2, p. 291 (1994)
- (7) 新井・木村他:「移動ロボット相互の位置・姿勢実時間計測システムの開発」, 日本ロボット学会誌, Vol. 12, No. 3, p. 472 (1994)
- (8) 井上・稲葉他:「局所相関演算に基づく実時間ビジョンシステムの開発」, 日本ロボット学会誌, Vol. 13, No. 1, p. 134 (1995)
- (9) 森・稲葉・井上:「複数注視点の協調に基づく動物体の視覚追跡」, 日本ロボット学会誌, Vol. 13, No. 7, p. 1053 (1995)
- (10) 市川・原:「群ロボットシステムにおける群知能の発現特性に関する研究—空間探索能力とその応用—」, 日本ロボット学会誌, Vol. 13, No. 8, p. 1138 (1995)
- (11) Q. Zhu: "Hidden Markov model for dynamic obstacle avoidance of mobile robot navigation", IEEE Trans. on Robotics and Automation, Vol. 7, No. 3, p. 390 (1991)
- (12) S. Yokoyama, K. Fukushima, & M. Uchida: "Study of avoiding obstacles by neural networks", 1992 IEEE Int. Conf. Intelligent Control and Instrumentation, p. 412 (1992)
- (13) M. Uchida, S. Yokoyama, & H. Ide: "Development of autonomous mobile robot for obstacle avoidance", 1994 IEEE Instrumentation and Measurement Technology Conference, Vol. 1, 179-180 (1994)
- (14) 横山・内田・福島・柿沼:「障害物回避問題におけるポテンシャル場の自律生成」, 日本ロボット学会誌, Vol. 10, No. 4, p. 510 (1992)

- (15) 内田・井出：「自律型ロボットのためのSPM法による経路計画」，電気学会論文誌C, Vol. 114-C, No. 9, p. 964 (1994)
- (16) 内田・井出：「SPM法による経路計画と移動ロボットシステムへの応用」，電気学会論文誌C, Vol. 115-C, No. 6, p. 101 (1995)
- (17) 内田：「移動ロボットのための最短経路計画」，電気学会論文誌C, Vol. 115-C, No. 9, p. 1079 (1995)

内田 雅文 (正員) 1964年9月6日生まれ。1990年工学院大学大学院工学研究科修士課程修了(電気工学専攻)。工博。移動ロボット，ニューラルネットワークなどの研究に従事。現在，青山学院大学理工学部電気電子工学科助手。1993年電気学会優秀論文発表賞受賞。IEEE，日本ロボット学会，神経回路学会会員。

井出 英人 (正員) 1970年工学院大学大学院工学研究科修士課程修了(電気工学専攻)。1985年UCLA応用科学科客員研究員。1986年UCLA脳研究所客員教授。青山学院大学理工学部講師，助教授を経て，同教授。現在，理工学部長。工博。感覚の計測，感覚代行の研究に従事。日本ロボット学会，電子情報通信学会，計測自動制御学会会員。

横山 修一 (正員) 1970年工学院大学大学院工学研究科修士課程修了(電気工学専攻)。工学院大学講師，助教授を経て現在，同教授。工博。保護継電器の双対理論を応用，クレーンの振れ止め制御，ロボットの障害物回避， H_∞ 制御による温度制御などの研究に従事。IEEE，日本ロボット学会，計測自動制御学会などの会員。