自律移動ロボットにおける実時間行動探索

藤澤 加絵^{*1} 早川 聡一郎^{*2} 青木 猛^{*3}
鈴木 達也^{*1} 大熊 繁^{*1}

A Real-Time Action Search for Autonomous Mobile Robot

Kae Fujisawa^{*1}, Soichiro Hayakawa^{*2}, Takeshi Aoki^{*3},
Tatsuya Suzuki^{*1} and Shigeru Okuma^{*1}

A real-time action search for an autonomous mobile robot using Evolution Strategy (ES) is proposed. By searching optimal actions for facing situations in real time, the robots are relieved from the problems of categorization of the situations and a-priori-definition of rules. The basic idea of the real time action search is to balance computational amount with exactness of solutions. In order to realize this trade-off, a two-stage evaluation technique is introduced under a framework of Anytime Algorithm. The prediction and action watch dog systems are also used. Then, the feasibility is shown through the experimental results.

Key Words: Evolution Strategy, Fuzzy Logic, Real-Time Search, Anytime Algorithm, Two-Stage Evaluation, Autonomous Mobile Robot

1. はじめに

複数のロボットに何らかの行動探索能力を持たせることを目的とした自律移動ロボットの研究が盛んに行われている。ロボットに行動探索能力を持たせるためには、ロボットを取り巻く状況を何らかの形で計算機上に表現・分類し、各状況ごとに行動探索を行わせる必要がある。FC (Fuzzy Classifier) や LA (Learning Automaton), Q-learning などの強化学習, ポテンシャル法やファジィ推論のようなルールベースの手法はこの概念を実現したものである。前者の強化学習を用いた手法としては、田野ら [1] の確率的学習オートマトンを用いた多足歩行機械の歩容行動の学習や、浅田ら [2] の Q-learning によるサッカーロボットの行動獲得手法、山村ら [3] の Q-learning の発展形である k-確実探索法、さらに藤井ら [4] の学習を複数の段階に分け、多層構造化する手法などがある。筆者らも障害物回避能力を獲得するため、下位レベルで階層ファジィルールにより各障害物に対する回避のための速度操作量およびステアリング操作量を個別に決定し、上位レベルにおいて学習オートマトンにより両制御入力間の干渉を生じないような適切な合成ルールを学習する手法を報告した [5]。これらの手法はどれもロボットを取

り巻く状況を離散的に表現し、各状況での適切な行動を経験を通して学習し、獲得していくものである。そのため、状況の表現法が異なると各状況における学習結果、すなわち各状況における行動が異なってくる。多くの場合、この状況の表現方法は設計者が試行錯誤的に決定している。このため、ロボットが適切な行動を獲得するような状況分類を行うことは容易なことではない。また、後者のルールベース手法には、石川ら [6] のプランナにより生成された大域的な経路をファジィ制御により追従しながら局所的な障害物の回避を行う手法や、永田ら [7] の、ファジィ推論とニューラルネットワークを組み合わせた移動障害物回避手法、橋本ら [8] の静電ポテンシャルと Sliding Mode を用いたマニピュレータの障害物回避、新井ら [9] の仮想インピーダンスを用いた複数移動ロボットの協調動作獲得手法などがある。これらの手法はロボットを取り巻く状況に対する行動をそれぞれの手法でルール化するものであるが、ロボットの状況に応じた行動のルール化をする際の状況の表現の仕方や、ルール化の方法、ルール数の設定、数々のパラメータの調整などは強化学習と同様に設計者が試行錯誤的に決定しなければならない。

これらに対し本論文では、実時間で (1) 状況認識 (2) 行動探索 (3) 行動実行を行うことでロボットが直面する状況に適切に対応できる実時間行動探索手法を提案する。計算機パワーの進歩を考えると、その場その場で解を見つけながら行動するシステムの有用性はますます高まるものと思われる。

以下、本論文では「実時間探索」という言葉を使うが、人工知能の分野でも「実時間探索」という言葉が用いられている [10]。

原稿受付 1998 年 2 月 17 日

^{*1}名古屋大学大学院

^{*2}豊田工業大学

^{*3}名古屋市工業研究所

^{*1}Graduate School of Nagoya University

^{*2}Toyota Technological Institute

^{*3}Nagoya Municipal Ind. Res. Inst.

これは全空間に対して解を生成するオフライン探索に対する一定時間内の解探索を指しており、本論文で用いる実時間探索とは異なる意味を持つ。本論文における実時間とは、状況の変化速度に対する時間を陽に意識したものであり、また実時間探索はある状況においてその状況が変化するよりも十分早い時間で行動探索機構の行動探索が収束することを意味する。

上記のような探索に関する研究として、坪内ら [11] の移動障害物の動作を等速直線運動で近似し、その場その場の状況に応じてプランニングを行うものや井上ら [12] の移動障害物の動作を予測し、その予測行動を考慮に入れて状況に適したポテンシャル法のパラメータを探索する手法などがあるが、本研究では人工知能の分野で実時間探索を実現させるために提案された Anytime Algorithm [13] [14] の枠組を用いるものとする。この Anytime Algorithm を用いた探索手法として、Ferrari ら [15] の研究がある。これは複数の評価基準を基に複数の行動探索を行うことで行動計画のロバスト性を高め、最終的に決定された一つの行動計画を基に集中管理型システムがすべてのロボットに対し行動を割り当てる手法であるが、これに対し、本研究では自律型ロボットの行動探索を目的とし、Anytime Algorithm の枠組を用いることで、自律移動ロボットの実時間行動探索手法を実現する。さらに、本論文では文献 [13] や [14] で提案された Anytime Algorithm の枠組を拡張し、複数の仕様を持つ実時間探索手法を実現する。

提案する手法では以下の考え方を導入する。まず、実時間探索を実現するためにロボットの行動計画システムに予測を導入する。また Anytime Algorithm の枠組に沿った探索手法として本論文では ES (Evolution Strategy) [16] [17] を使用する。これは、近年注目を集めている生物の進化過程を工学的に模倣した探索手法である。ES は複雑な問題に対する解探索能力があり、同様の探索手法である GA (遺伝的アルゴリズム) に比べ収束時間が早いという長所がある。この探索手法を用いることで、Anytime Algorithm が持つべき性質を保証することができる。さらに、本論文で述べる Anytime Algorithm が持つべき性質 [13] を保証するために行動監視機構を導入し、行動計画システムと並列に動作させる。ロボットを取り巻く状況が変化し、予測誤差が閾値を越えた場合、行動監視機構が行動計画システムの行動探索を打ち切り、探索途中の行動を出力することで予測誤差に速やかに対処する。最後に、Anytime Algorithm の考え方を複数の仕様を持つ行動探索に拡張するため、行動計画システムに 2 段階評価関数を導入する。

提案する上述の手法を用いて実時間探索を行うことで、あらかじめすべての状況を計算機上に表現することなくロボットは状況に適した行動を実時間で獲得することが可能である。

以下、第 2 章では問題設定、第 3 章では実時間探索の定義、第 4 章では実時間探索手法を示す。第 5 章はシステムの構成を示し、第 6 章でシミュレーション、第 7 章で実験結果を述べる。

2. 問題設定

本論文では具体例として自律移動ロボットの移動障害物回避問題を取り上げる。はじめに以下の前提条件を与える。

前提条件

Fig. 1 Parameter vector of robot's environment

- ロボットは速度およびステアリング操作を同時に用いることにより障害物を回避する
- ロボットが行動する空間には移動障害物のみが存在する
- 移動障害物は最大 3 台とする
- 移動障害物は基本的には等速直線運動を行うが、突然経路を変えることもある
- ロボットの経路に制約を与えない
- ロボットは滑らかに動作する
- ロボットにはあらかじめ目標点が与えられている

上記のような前提条件下で障害物回避行動計画システムを構築する。また、システムにおいてロボットおよび障害物の状況は定位置に固定されたカメラから得られ、この情報に基づき Fig. 1 に示すようにロボットと障害物との相対距離 (L) および相対角度 (RA)、ロボットに対する障害物の相対速度 (W) およびロボットの進行方向に対する相対速度ベクトルの角度 (RU) を計算する。また、カメラから得られたロボットの位置からゴールまでの距離 (RTL) と、ロボットの進行方向に対するゴール方向の角度 (RTA) を計算する。このように計算されるパラメータのうち、 L , RA , W , RU をロボットの状況を示すパラメータベクトル $P = (L, RA, W, RU)$ と定義し、以後ロボットの状況をこのパラメータベクトルにより表現する。

3. 実時間探索の意味

本論文で提案する手法は、実時間で (1) 状況認識 (2) 行動探索 (3) 行動実行を行うことにより直面する状況に適した行動を探索する。ここで言う実時間とは“状況認識時と行動実行時の状況にずれがない程度の短い時間”を意味する。実際にはロボットを取り巻く状況は常に変化するため、ロボットが状況認識時の状況下で行動探索を行っている間に実際の状況は状況認識時から変化し、状況認識時と行動実行時の状況に何らかの

ずれを生じる。したがって、本論文では、実時間探索を以下のように定義する。

ロボットを取り巻く状況にあるパラメータ群を用いて表現し、それをベクトル表現したパラメータベクトルの時間変化率に対して、十分に速くロボット自身が行動探索を行うこと

つまり実時間探索とは、単に探索に要する時間が短いことを意味するのではなく、あくまで状況を表すパラメータベクトルの時間変化に比べて探索時間が十分に短いことを意味するものとする。例えば、時間変化の小さい状況では探索に要する時間がある程度かかっていても状況変化が小さいので状況認識時と行動実行時の状況にずれが少なく、探索後の行動は実際に行動を行うときの状況に適したものとなる。反対に、状況の時間変化が大きい場合、状況認識時と行動実行時の状況に大きなずれが生じ、探索された行動は行動実行時の状況に適さない可能性が生じる。このように、探索に費やしている時間は状況を表すパラメータベクトルの時間変化に依存する。

また、状況を表すパラメータベクトルのおおよその時間変化は、上記の (1), (2), (3) を実行する制御周期を決める上でも重要な要因となる。この関係については次章で詳しく述べる。

4. 実時間探索手法

4.1 予測の導入

4.1.1 実時間探索における予測の必然性

実時間で (1) 状況認識 (2) 行動探索 (3) 行動実行を考える場合、ロボットを取り巻く状況は常に変化をしているため、ロボットが行動探索を行っている間に状況が変化し、状況認識時と行動実行時の状況にずれが生じる場合も多い。ロボットは状況認識時の状況で行動探索を行うため、この状況のずれにより、探索結果として得られた行動が状況に適さない場合が生じる。特に、行動探索にかかる時間が長い複数移動障害物回避問題を考える場合、状況認識時と行動実行時の状況にずれが生じやすく、結果として行動探索結果が行動実行時の状況に適さない場合が生じやすい。このような問題を解決するためには行動探索を行うときにあらかじめ状況認識時と行動実行時の状況のずれを何らかの形で補償する必要がある。本論文ではこのずれを補償する方法として予測を導入する。状況認識時の状況から行動実行時の状況を予測し、その予測状況に基づいて行動探索を行うことにより、行動認識時と行動実行時の状況の間のずれをできる限り少なくする手法を用いる。

本論文における予測法を以下に述べる。

Fig. 2 にロボットおよび障害物のおかれた状況を示す。図中の $R[n]$ および $O[n]$ はステップ n におけるロボットおよび障害物の位置を表す。この位置を状況認識時とし、行動実行時は次のステップ $n+1$ とする。先ほど述べたようにロボットは行動実行時の状況を予測して行動探索を行うが、以下の手順に従い行動実行時の状況を予測する。

- (1) ロボットの現在の位置 $R[n]$ と行動から次のステップでの位置 $R[n+1]$ を計算する
- (2) 障害物は等速直線運動を行っているとは仮定し、一つ前のス

Fig. 2 Prediction

テップにおける位置 ($O[n-1]$) と現在の位置 ($O[n]$) から行動実行時の障害物の位置 $O[n+1]$ を予測する。

- (3) 求めた $R[n+1], O[n+1]$ よりロボットの状況を示すパラメータベクトル $\hat{P}$ を計算する

このようにして予測した行動実行時の状況に基づいて適切な行動を探索することにより状況に適応する。

4.1.2 予測と制御周期の関係

状況を表すパラメータベクトルの時間変化率は予測と制御周期を決定する上で重要な要因となる。また、予測を用いる場合、予測誤差を考慮しなければならない。本節では移動障害物回避問題における予測と制御周期の関係、および予測と予測誤差の問題について述べる。

Fig. 3 は制御周期の異なるロボットおよび障害物の時間的位置関係を示している。

本論文で用いる複数移動障害物回避行動計画システムは反射的に行動を起こすものではなく、一定制御周期ごとに行動探索を行い、その行動を実行する。前節で述べたように、このシステムでは次のステップ時を行動実行時としているため、システムは 1 制御周期後の状況を予測する。ここで Fig. 3 に示すような制御周期の異なるロボットおよび障害物の位置関係を考えてみる。簡単のためロボットを中心としたある半径以内に障害物が入ってきた場合にロボットは障害物回避をはじめとする (回避開始半径)。前節で述べたように現在のロボットと障害物の位置 ($R[n], O[n]$) およびそのときの行動から次の制御周期の位置 ($R[n+1], O[n+1]$) を予測してその位置における適切な行動を探索する。ここで Fig. 3 (a) と (b) を比較すると、次の制御周期の位置 $R[n+1]$ においてどちらも障害物が障害物回避半径に入っていないのでロボットは回避行動を取る必要がないと判断し、ゴール方向へ直進する。しかし、Fig. 3 (a) の制御周期の場合、 $[n+1]$ ステップ時の状況から予測されたその次の制御周期の位置 $R[n+2]$ においてロボットの障害物回避開始半径

Fig. 3 Relation between sampling time and robot's action

に障害物が入るためロボットは障害物回避行動を取ることが可能である。これに対して Fig. 3 (b) では、ロボットおよび障害物の状況変化速度に対し必要以上に長い制御周期であるため、図に示すように行動探索を行っている間に障害物と衝突する可能性が生じる。したがって Fig. 3 (b) のような場合を避けるためにはある長さ以下の制御周期にしなければならない。この制御周期の上限はロボットと障害物が衝突する時間、すなわちロボットと障害物の相対速度に依存する。また、ロボットと障害物の相対速度はロボットの状況を表すパラメータベクトルの時間変化率に直接影響する。結果として、制御周期の上限はパラメータベクトルの時間変化率に依存する。

一般に自律移動ロボットの行動探索は探索空間が広いため行動探索に多くの時間を必要とする。しかし、実時間探索を行う場合、上述のように探索時間には上限が存在するためこの時間以内に探索を終了しなければならない。また、制御周期の下限についてであるが、ロボットが最低限の状況認識、行動探索、行動出力を行うために要する時間以上の制御周期を取らなければロボットは次に取るべき行動を決定できず、制御不可能になる。以上よりロボットの制御周期はロボットの状況を表すパラメータベクトルの時間変化率から決定する制御周期の上限と行動決定プロセスに必要な下限の間としなければならないことが分かる。

4.2 Anytime Algorithm の概念の導入と拡張

前節に示したように提案する手法では状況認識時と行動実行

時の状況のずれを少なくするために予測状況において探索を行う。しかし、予測状況が実際の状況からずれる場合も考えられる。そこで本手法では Anytime Algorithm [13] の枠組を導入・拡張し、予測と実際の状況が異なる場合に対処する。

4.2.1 Anytime Algorithm

Anytime Algorithm は人工知能の分野において実時間探索を実現するための方法論として提案された。Anytime Algorithm とは探索時間が増加するにつれて、解の質を改善するアルゴリズムの総称である。様々なサブシステムに対し求める解の質に応じて適切に時間を割り当てることが可能であるため、システム全体として効率的な探索を行うことが可能であり、複雑に変化する環境に対しても状況に応じて探索時間の再配分をすることが可能である。以下の性質が満たされる探索アルゴリズムを Anytime Algorithm と呼ぶ。

- (1) 解の質の測定が可能 (Quality Measure)
- (2) 予測可能性 (Predictability)
- (3) 中断・継続可能性 (Interruptibility and Continuation)
- (4) 解の質の単調増加性 (Monotonicity)

すなわち、Anytime Algorithm では

- (1) 探索アルゴリズムの解 (出力) は、質 (測度) を持つ
- (2) 探索アルゴリズムは、探索時間と得られる解の質に対する静的な情報を持つ。したがって、割り当てられる探索時間に対し、得られる解の質を推定することができる。
- (3) 探索アルゴリズムは、環境の変化に応じて探索を打ち切り、途中の解を出力することや、サブシステムに対する時間割り当てを変更 (再配分) することができる。
- (4) 探索アルゴリズムの解の質は、探索時間の増大とともに向上する

という性質を持つ。

自律移動ロボットの行動探索アルゴリズムに上述の Anytime Algorithm の枠組をそのまま導入する際、以下に示すような問題が生じる。まず第 1 に、与えられた時間に対する解の質に関する事前情報 (性質 2) を持つことは困難である。これは、ロボットを取り巻く様々な状況が考えられ、同等の質を持つ解を得るためにかかる探索時間が状況に応じて異なるからである。例えば、ロボットの周りに障害物が存在しない簡単な状況では、高い質を持つ解の探索にそれほど時間を要しない。しかし、ロボットの周りに障害物が複数存在する複雑な状況の場合、高い質を持つ解を探索するのに多くの時間を必要とする。このため、性質 2 のような与えられた探索時間に対する解の質の情報を持つことは不可能である。第 2 に、本研究では複数の仕様を持った行動の探索を目的としている。本研究の場合、(1) 障害物を回避する (2) ゴールへ向かう、という二つの仕様を持つ。このため二つの仕様を同時に満たす解の質が両方の仕様に対して性質 4 のように単調増加することを保証することは困難である。そこで Anytime Algorithm の枠組を拡張し、動的環境下で多目的な Anytime Algorithm の枠組を持つ探索アルゴリズムに対し以下の性質を定義する。

- (1) 解の質の測定 (Quality Measure)
- (2) 仕様の優先度に基づく探索
- (3) 中断・継続可能性 (Interruptibility and Continuation)

Fig. 4 Prediction and obstacle's action

上述の性質を満たすために、本研究では探索手法として Evolution Strategy (ES) を用いる。ES は探索空間内において複数の探索点からなる集団を扱い、各探索点を個体とみなして各個体に対し評価値を与え、評価値の高い個体を繁殖させながら探索空間における最適な点を探索する。これは、遺伝的アルゴリズムと同様、途中解と解に対する評価値を保持しているため上記の性質 1 および性質 3 を満たすことが可能な探索手法である。さらに、性質 3 を満たすために行動監視機構を導入する。最後に性質 2 を満たすため、2 段階評価関数を導入する。以下に行動監視機構および 2 段階評価関数について示す。

4.2.2 予測誤差と行動監視機構の導入

行動計画システムが予測を行う際、障害物は等速直線運動するという仮定のもとで障害物の位置を推定している (Fig. 4 (a))。しかし、障害物が等速直線運動と異なる運動を行い、結果として予測された状況と実際の状況が大きく異なる場合も生じる。(Fig. 4 (b)) この場合、行動探索を続けても得られる結果は役に立たない。このように予測状況と実際の状況とが異なる場合には、前節の性質 3 により探索アルゴリズムは探索を中断し、新たな時間割り当ておよび探索を行う。そこでこの性質を保証するために行動監視機構を導入し、行動計画システムと並行して動作させる。この行動監視機構は行動計画システムより短い周期で障害物の行動監視を行い、行動実行時における位置を推定する。そしてその予測位置と行動計画システムの予測位置の誤差を逐次計算する。本論文では行動監視周期を制御周期の 10 分の 1 とした。

Fig. 5 に本稿で提案する障害物回避行動計画システムと行動監視機構のタイムチャートを示す。

Fig. 5 (a) は障害物が行動計画システムの予測通りの行動を取っている場合、Fig. 5 (b) は障害物が予測と異なる行動を取った場合のシステムのタイムチャートである。通常、行動計画システムは、ロボットおよび障害物の状況認識を行い、現在の状況から行動実行時の状況を予測する。そして、予測状況における適切な行動の探索を開始する。探索が終了すると探索結果に基づきロボットに対する入力操作量を計算する。行動監視機構はこのシステムの動作と並列に障害物の動作を監視し続け、監視状況から次の行動実行時の状況 P を予測する。行動監視機構は監視状況から求めた P と状況認識時に行動計画システムが求めた予測状況 $\hat{P}$ との予測誤差 ($e = |P - \hat{P}|$) を計算する。

障害物が予測と同じ行動を取っている場合、上記の予測誤差は小さい。しかし、障害物が予測と異なる行動を取っている場合、予測誤差は大きくなる。したがって、予測誤差がある閾値以上の値を示した場合を障害物の行動が予測と外れたとみなし、

Fig. 5 Time chart of the system

Fig. 5 (b) に示すようにシステムに対し行動監視機構が探索打ち切り命令を出す。この命令が出ると、システムでは行動探索を速やかに終了し、探索途中の結果に基づきロボットの入力操作量を計算する。

4.2.3 評価切替と探索打ち切りの妥当性

予測状況と実際の状況が異なる場合に対処するため行動監視機構を導入した。予測が外れた場合、行動監視機構が探索打ち切り命令を出し、ロボットは探索途中の結果に基づきロボットへの入力操作量を計算する。しかし、探索途中の解を出力する場合、探索が不十分ため解の精度が不十分であるという問題が考えられる。探索アルゴリズムが Anytime Algorithm である場合、前節で示した性質 2 によって探索打ち切り時間から探索途中で出力される解の質を推定することが可能とならなければならない。自律移動ロボットの移動障害物回避問題のような動的環境に適用することを考える場合、性質 2 を保証するような解の質を推定するための静的な情報を生成することは困難であり、また今回探索手法として使用した ES (Evolution Strategy) において、探索時間に対する解の質の推定するための情報を生成することは困難である。そこで、この問題に対処するため、本論文では評価切替を行う探索手法を導入する。以下、移動障害物回避問題における評価切替について述べる。

障害物回避問題において、与えられる動作仕様としては (1) 障害物を避ける、(2) 目標点へ向かう、の 2 点が挙げられる。これらのうち (1) は (2) よりも優先度が高い。したがって、探索を途中で打ち切り、解を出力したとしてもその解が最低限仕様 (1) を満たす解である必要がある。これを実現するために本論文では以下のように評価関数を途中で切替えて (以下、2 段階評価と呼ぶ) 探索を行う。

第 1 段階 移動障害物に対する危険度を下げる行動に対し評価を与える

Fig. 6 Interruption of action search

第2段階 移動障害物に対する危険度を下げ、かつゴールに向かう行動に対し評価を与える

まず第1段階では、障害物に対する危険度(5.1節)を下げる行動に対し高い評価を与える評価関数を用いる。この評価関数による探索がある程度収束した段階で、そこまでの探索結果を初期値とし、第2段階の評価関数に切替えて探索を続行し、最終的な解を求める。

このように評価基準を2段階にし、第1段階で(1)の仕様を満たす行動探索を開始することで、Fig. 6に示すように予測が外れ、探索を打ち切り探索途中の結果を出力しても最低限障害物回避することが可能となる。もっともこの場合でも、第1段階の探索が終了する以前に探索打ち切り指令が与えられた場合は対応できない可能性が生じる。このような場合、最悪では障害物の進行方向に向かう行動を取ることも考えられる。しかし、第1段階目の評価のような最低限の仕様を満たす解は比較的短時間に探索可能であると考えられる。もし第1段階目の探索が終了する以前に探索打ち切り要求がなされた場合に障害物の進行方向に向かうような最悪な行動を取った場合でも、ロボットが探索開始を始めた位置からそれほど変化していない状況であるため、探索を打ち切ることで予測と異なる実際の状況を認識することができ、新たな状況での再プランニングを行うことが可能となり、前の不適な行動をリカバーすることが可能となる。また、行動探索を打ち切る必要がない場合は第2段階の評価基準に従って制御周期いっぱいまで行動探索を行うことでより最適な回避行動を獲得することができる。

5. システムの構成

5.1 システムの構成

本論文で提案する方法論を用いた自律移動ロボットの障害物回避行動計画システムの詳細について以下に述べる。Fig. 7に移動障害物回避行動計画システムの構成を示し、システムの動作の流れを以下に示す。

1 ステップ目において、ロボットは設定されたロボットの標準速度であらかじめ与えられた目標点へ進み出す。2 ステップ目より以下の手順でロボットが次に取るべき行動を探索する。

Fig. 7 System of motion planning for mobile obstacle avoidance

- (1) カメラから得られたロボットおよび障害物の位置情報と前ステップ時に決定した行動より、次のステップ時にロボットおよび障害物が存在する位置を予測する (Sensors)。
- (2) (1) で求めた予測状況での障害物に対する静的危険度 α および動的危険度 β [18] を各障害物に対し個別に計算する。また、ゴール方向へのステアリング操作量を計算する (Hierarchical Fuzzy Rules)。
- (3) (2) で求めた静的危険度 α および動的危険度 β [18] を入力として、追加学習型ニューラルネットワーク [19] より各障害物に対する速度およびステアリング操作量を個別に計算する (ここで使用するニューラルネットワークは学習結果を保持したまま、新たな学習を行うことができる追加学習能力を有したニューラルネットワークである。このニューラルネットワークにシステム構築前に ES によって探索された単体移動障害物時の回避のための速度操作量およびステアリング操作量を十分学習させたものである) (Neural Network with Additional Learning Ability)。
- (4) 仮想空間内において、予測状況に最適な上記二つの操作量の合成重みを以下の手順により探索する (Virtual Space)。
 - (a) ES (Evolution Strategy) により求めた遺伝子の染色体の重みを用いて次節に示す合成ルールより回避のための各操作量およびゴール方向へのステアリング操作量を合成する (Real-Time Evolution Strategy)。
 - (b) 合成された操作量を用いて予測された状況において仮想的に行動を行う (Action in the Model)。

- (c) その行動の結果を評価し、行動に対する評価値を計算する (Performance Evaluator).
- (d) 評価値を ES に返し、ES により新たな回避のための操作量合成重みの探索を行う。
- (e) (a) から (d) を一定回数行う
- (5) (4) において、評価値の一番高かった回避のための各操作量の合成量を次のステップ時の行動のための制御入力とする (Control Input for the Next Action).
- (6) 次のステップ時に先に求められた制御入力に従い行動する (Action in the World).

以上の手順に従い、ロボットは状況に応じた行動のための各操作量の合成重みを実時間で探索し、一ステップごとに適切な行動を行う。なお、4.2.3 項で述べたように評価は 2 段階評価を使用している。第 1 段階目の評価は、Fig. 7 に示す静的危険度 α と動的危険度 β の平均値を下げる行動に高い評価を与えるものとする。以下に評価式を示す。

$$\Delta D = d(n+1) - d(n+2)$$

$d(n)$: n ステップ目の静的および動的危険度の平均値

上式の ΔD の値がそのまま行動に対する評価値となる。したがって、静的危険度 α および動的危険度 β の平均値の変化分が大きい行動に対し高い評価が与えられることとなる。第 2 段階目の評価は、静的危険度と動的危険度の平均値を下げ、かつゴールまでの距離を短くする行動に対し高い評価を与える。この第 2 段階目の評価はファジィ行動評価器となっており、これに関しては文献 [18] を参照されたい。

5.2 ES による回避操作量の合成方法

本システムではロボットが障害物回避のための速度操作量とステアリング操作量を次式で定義される合成ルールに従い合成する。

合成ルール

$$V_{input} = V_{now} + \sum_{r=1}^k w_{vr} \cdot vd[r]$$

$$S_{input} = rta \cdot (1 - w_s) + \sum_{r=1}^k w_{sr} \cdot ster[r]/k$$

ただし、

V_{input} : 速度制御入力

S_{input} : ステアリング制御入力

V_{now} : 現在の速度

$vd[]$: 回避のための速度操作量

$ster[]$: 回避のためのステアリング操作量

rta : ゴール方向

w_{vr} : 回避のための速度の重み

w_{sr} : 回避のためのステアリングの重み

w_s : 回避のためのステアリングの重みの和

k : 障害物の数

である。本論文で探索手法として用いる ES は上記の合成ルールにおける回避のための各操作量の重み (w_{vr} , w_{sr}) を遺伝

子とし、状況に適切な値を探索する。探索手法の ES であるが、本システムでは個体数を 6 とし、エリート戦略により三つの親から三つの子を選択し、突然変異のみを用いて親から子を選択する方式を用いる。

6. 実験

6.1 実験装置

本論文で提案する手法を検証するために実機による実験を行った。使用した実験装置を以下に示す。

実験では移動ロボットとして Khepera を使用する。また、ロボットの動作範囲は 135×220 [cm] の滑らかな平面である。三次元座標計測装置である Quick-MAG によりロボットおよび障害物の座標をホストコンピュータが獲得する。コンピュータは FUJITSU FMV (Pentium200MHz) を使用する。また実験での制御周期は 700 [msec] とする。ホストコンピュータは絶対座標系での計算は行わず、ロボット、障害物、ゴールの座標から相対的な情報に変換し、その情報を基に提案する手法に基づき操作量の演算を行う。計算された操作量はホストコンピュータから赤外線リモコンにより移動ロボットに送信され、ロボットはその操作量情報をもとに動作を行う。また、実験で使用する移動ロボットの走行誤差、および測定器の計測誤差は予測誤差として考慮しないこととする。

6.2 実験結果

実験は、障害物が等速直線運動を行う場合と、障害物があるステップにおいて突然動作を変える、すなわち予測と異なる運動を行う場合を想定する。移動障害物は各実験において 3 台の移動障害物を使用した。その結果を Fig. 9~Fig. 11 に示す。

Fig. 9 は障害物が等速直線運動を行っている場合である。また、Fig. 10 と Fig. 11 は障害物が 12 ステップ目 (図中の動作変更点) で突然動作を変える場合の実験結果である。図の○は

Fig. 8 Experimental setup

各ステップごとのロボットおよび障害物の位置を示し、●は10ステップごとのそれぞれの位置を示す。Fig. 9では障害物がロボットの前方からロボットへ向かっており、ロボットは障害物を検知するとまず加速を始め、その後ステアリングを組み合わせることで障害物1を回避し、ゴールへたどり着いている。Fig. 10では、3方向から障害物がロボットに向かっており、障害物3は12ステップ目で突然進行方向右30度へと変化している。ロボットは始め標準速度で障害物1を回避し、その後、障害物2と3を回避しようとする。しかし12ステップ目で障害物がロボットの予測と異なる運動を取るため、途中で探索を中止し、探索途中の結果を出力することでゴールへ向かうことを諦め、回避行動を取っている。その後、速やかに現在の状況を認識直して新たな環境における行動探索を行っている。障害物2と3を速度とステアリングを用いて回避し、ゴールに向

Fig. 9 Experimental result (No.1)

Fig. 10 Experimental result (No.2)

Fig. 11 Experimental result (No.3)

かっている。Fig. 11の場合、Fig. 10と同様に障害物3が突然右へ曲がっている。ロボットは、障害物の動作が予測と異なっているを検知し、Fig. 10の場合と同様、探索を打ち切って障害物を回避する動作を取っている。その後、状況の再認識を行い、状況が危険であると判断したため、ステアリングと加速を最大限に用いることにより速やかに障害物3を回避し、ゴールへ到達している。

上記の実験結果より、本論文で提案する手法の有効性を実機により確認した。

7. 考 察

本論文で提案する手法の有効性を考察する。実機ではデータ採取が困難なためシミュレーションによる考察を行った。比較のために、提案する手法(2段階評価と行動監視機構による探索打ち切り)、2段階評価のみ、および1段階評価と行動監視機構による探索打ち切りの三つの場合のシミュレーションを行った。シミュレーションを行う際、すべての場合においてロボットのスタート位置および障害物の軌跡はすべて同じものとした。また、予測を外れた場合の本論文で提案する手法の有効性を確認するため、10ステップにおいて障害物が予測と異なる行動を取るものとした。

このようにして行ったシミュレーションのデータをFig. 12～Fig. 13に示す。このデータは8ステップから12ステップでの評価に使用する危険度の変化分とゴールまでの距離の変化分の時間変化を示している。各図で示す実線と破線はそれぞれ障害物に対する危険度の変化分とゴールまでの距離の変化分である。危険度の変化分およびゴールまでの距離の変化分はそれぞれ大きい方が危険度を下げるおよびゴールまで近付くという評価が高いものである。Fig. 12およびFig. 13の実線の仕切り線はそれぞれステップの切れ目を示しており、1ステップ600[msec]である。またFig. 12の破線の仕切り線は第1段階と第2段階の評価関数が変わる時刻を示している。

Fig. 12の10ステップ目において、行動監視機構が障害物の予測と異なる動作を検知し、探索を途中で打ち切っている。そ

Fig. 12 Time chart of change degree of distance to goal and danger degree (2stage evaluation with action watch dog)

Fig. 13 Time chart of change degree of distance to goal and danger degree (only action watch dog)

の後の 11 ステップおよび 12 ステップ目を見てみると、前半でゴールへの障害物に対する危険度の変化分を上げる行動を選択し、また、第 2 段階目の評価に移ると危険度の変化分をある程度犠牲にしながらゴールへと向かう行動を選択していることが分かる。一方で、1 段階評価を採用している Fig. 13 の場合、このような傾向は見られず、状況によって危険度の変化分を重視したりまたゴール方向の距離の変化分を重視する行動を探索していることが分かる。

上記のような結果から以下のようなことが考えられる。2 段階評価を採用している場合、第 1 段階の評価関数よりロボットはゴールまでの距離の変化分を犠牲にし、障害物に対し危険度を下げる、すなわちロボットから遠ざかる行動を探索する。続けて第 2 段階の評価に切り替わるとそれまでの解を初期値とし、今度は危険度のある程度犠牲にしてゴールまでの距離の変化分を上げる行動を探索することとなる。

このような特徴を持つ 2 段階探索と併用して探索打ち切りを用いることで、予測が外れた場合でも探索途中の結果の行動によりいったん障害物から離れる行動を取ることで障害物との衝突を回避することができる。また、予測状況が外れた場合探索を打ち切ることで状況の再認識が可能となり、速やかに新しい状況に適応する行動を探索することが可能となる。また、障害物の動作が予測と同じ場合、第 2 段階の評価を続けることで障害物を回避しつつゴールへ向かう適切な行動を探索することができる。

以上のように行動監視機構による探索打ち切りと 2 段階評価を用いた手法の有効性を確認した。

8. まとめと今後の課題

本論文では、行動監視機構による探索打ち切りと 2 段階評価関数を用いた実時間探索による自律移動ロボットの行動探索手法を提案し、シミュレーションにより本論文で提案する手法の考察を行い、実験により有効性を確認した。

近年 FPGA 等を用いて探索処理をハードウェア化することにより探索処理速度の向上を目的とした研究が行われている [21]。これらの成果を取り入れることにより提案手法の有用性はさら

に高まるものと思われる。また、今後の課題としては探索により得られた結果を知識として蓄える枠組の開発が挙げられる。

参考文献

- [1] 田野, 三上, 嘉数: “強化学習を用いた多足歩行機械の適応的歩容計画”, SICE'93, pp.149-154, 1993.
- [2] 浅田, 野田, 俵積田, 細田: “視覚に基づく強化学習によるロボットの行動獲得”, 日本ロボット学会誌, vol.13, no.1, pp.68-74, 1995.
- [3] 山村, 宮崎, 小林: “強化学習の特徴と発展の方向”, システム/制御/情報, vol.39, pp.191-196, 1995.
- [4] T. Fujii, Y. Arai, H. Asama, I. Endo: “Multilayered Reinforcement Learning for Complicated Collision Avoidance Problems,” Proc. of the 1998 IEEE Int. Conf. on Robotics and Automation, vol.3, pp.2186-2191, 1998.
- [5] 青木, 岡, 鈴木, 大熊, 早川: “多段階学習による障害物回避のための行動獲得”, 日本ロボット学会誌, vol.15, no.4, pp.648-656, 1997.
- [6] 石川: “ファジィ制御を用いた自律型移動ロボットの誘導方式の検討”, 日本ロボット学会誌, vol.9, no.2, pp.149-161, 1991.
- [7] 永田, 土谷: “移動ロボットの障害物回避に関する知的制御”, 日本ロボット学会誌, vol.11, no.8, pp.1203-1211, 1993.
- [8] 橋本, 國井, 原島: “静電ポテンシャル法および Sliding Mode を用いた多自由度マニピュレータの障害物回避制御”, 日本ロボット学会誌, vol.11, no.8, pp.1220-1228, 1993.
- [9] 新井, 太田: “仮想的なインピーダンス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.11, no.7, pp.1039-1046, 1993.
- [10] 石田, 新保: “実時間探索による経路学習”, 人工知能学会誌, vol.11, no.3, pp.411-419, 1996.
- [11] 坪内, 浪花, 有本: “平面を移動する複数の移動障害物とその速度を考慮した移動ロボットのプランニングとナビゲーション”, 日本ロボット学会誌, vol.12, no.7, pp.1029-1037, 1994.
- [12] 井上, 井上, 大川: “複数移動障害物の行動予測に基づく自律移動ロボットのオンライン回避行動計画”, 日本ロボット学会誌, vol.15, no.2, pp.249-260, 1997.
- [13] J. Grass: “Reasoning about Computational Resource Allocation—An introduction to anytime algorithms,” the ACM magazine, 1996.
- [14] S. Zilberstein: “Using Anytime Algorithms in Intelligent Systems,” AI MAGAZINE, vol.17, no.3, pp.73-83, 1996.
- [15] C. Ferrari, E. Pagello, M. Voltolina, J. Ota and T. Arai: “Varying Paths and Motion Profiles in Multiple Robot Motion Planning,” Proc. 1997 Int. Symp. Computational Intelligence in Robotics and Automation, pp.186-193, 1997.
- [16] R. Salomon: “Increasing Adaptivity through Evolution Strategies,” The Fourth International Conference on Simulation of Adaptive Behavior (SAB96).
- [17] 長尾, 上杉, 畠中, 魚崎: “協調行動を組み込んだ進化戦略”, 第 6 回 FAN シンポジウム講演論文集, pp.374-377, 1996.
- [18] 青木他: “階層ファジィルールを用いた自律移動ロボットの障害物回避動作計画”, 電気学会 C 論文誌, vol.13, no.7, pp.535-542, 1993.
- [19] 早川, 吉竹, 鈴木, 大熊: “学習結果に基づいたロボットマニピュレータの逆ダイナミクス獲得法—追加学習能力を有するニューラルネットワークの適用—”, 第 14 回ロボット学会学術講演会予稿集, no.1, pp.33-34, 1997.
- [20] Clark, Arkin, Ram: “Learning Momentum: On-line Performance Enhancement for Reactive Systems,” Proc. of the IEEE International Conference on Robotics and Automation, 1992.
- [21] 矢野, 橋山, 大熊: “HDL による進化型計算回路の設計”, 電気学会全国大会講演論文集, no.3, p.80, 1997.
- [22] 内田, 井出, 横山: “動的環境における障害物領域予測法”, 電気学会 C 論文誌, vol.117-C, no.2, 1997.
- [23] 大久保: “リアルタイムシステムの現状と課題”, システム/制御/情報, vol.41, no.5, pp.147-153, 1997.
- [24] 島川: “リアルタイム監視・制御システムの現状と課題”, vol.41, no.5, pp.172-178, 1997.
- [25] L.P. Kaelbling: Recent Advance in Reinforcement Learning. Kluwer Academic Publishers, 1996.

藤澤加絵 (Kae Fujisawa)

1974年9月11日生。1997年3月，名古屋大学工学部電気工学科卒業，同年4月名古屋大学大学院工学研究科博士課程前期課程電気工学専攻入学，現在に至る。ロボットの行動獲得に関する研究に従事，電気学会の会員。（日本ロボット学会学生会員）

青木 猛 (Takeshi Aoki)

1962年3月21日生。1984年3月，名古屋大学工学部航空学科卒業。1986年3月，同大学大学院修士課程修了。同年4月，名古屋市経済局工業研究所入庁。知能ロボット，メディカルエレクトロニクスなどの研究に従事。計測自動制御学会，日本ファジィ学会，日本人工臓器学会の会員。（日本ロボット学会正会員）

大熊 繁 (Shigeru Okuma)

1948年2月16日生。1970年3月，名古屋大学工学部電気学科卒業。1972年3月，同大学大学院修士課程修了。1974年ケース・ウェスタン・リザーブ大学大学院修士課程システム工学専攻修了，1977年名古屋大学大学院博士課程中退。同年，名古屋大学工学部助手。講師を経て，1984年，同助教授。1990年同教授，現在に至る。1978年，工学博士。この間，ロボティクスとパワーエレクトロニクスの研究に従事。電気学会，計測自動制御学会，日本機械学会，IEEEの会員。（日本ロボット学会正会員）

早川聡一郎 (Soichiro Hayakawa)

1968年5月1日生。1996年3月，名古屋大学工学研究科電気工学専攻博士課程後期課程修了，博士（工学）。同年4月，豊田工業大学助手，現在に至る。ロボットマニピュレータと移動ロボットの知能制御に関する研究に従事。電気学会，電気通信情報学会，計測自動制御学会，システム制御情報学会の会員。（日本ロボット学会正会員）

鈴木達也 (Tatsuya Suzuki)

1964年1月3日生。1991年3月，名古屋大学大学院博士課程後期課程電子機械工学専攻修了。同年4月，名古屋大学工学部電気学科助手。1995年同講師，1998年9月よりU.C.Berkely 客員研究員，現在に至る。ロボasts制御，学習制御を用いたモーションコントロール，離散事象システム理論を用いた知的システムに関する研究に従事（工学博士）。1994年システム制御情報学会奨励賞，1995年電気学会論文賞受賞，1996年日本ロボット学会研究奨励賞をそれぞれ受賞。電気学会，計測自動制御学会，システム制御情報学会，IEEEの各会員。（日本ロボット学会正会員）