

## 有限視野を持つ群ロボットのための一点収束アルゴリズムと その誤差に対する強度の評価\*

大麻 宜伸<sup>†</sup> 鈴木 一郎<sup>‡</sup> 山下 雅史<sup>†</sup>

(<sup>†</sup>広島大学 工学部 <sup>‡</sup>ウイスコンシン大学-ミルウォーキー校)

群ロボットの協調において、各ロボットが共有する座標系の存在が重要な役割を果たす。共有座標系の獲得問題をロボットをある幾何図形(点, 直線, 円)に収束させる問題に帰着することで解決する方法は Suzuki と Yamashita によって検討されたが, ここでは, ロボットが完全な無限視野を持つことが仮定されていた [2]。そこで Ando 他は, 現実のロボットへの実現を考慮し, 視野を有限に制約した上で, 点への収束問題を検討し, そのアルゴリズムを提案した [1]。しかし, アルゴリズムの正当性の証明において, 彼らは現実のロボットにそぐわない次の強い条件を仮定している。(1) ロボットは大きさを持たない点であり, (2) 視覚センサーや制御系には誤差を持たない, 本稿では, 実際のロボットへの実装に当たり, これらの条件がアルゴリズムに与える影響をシミュレーションによって検討し, このアルゴリズムが, 大きさを持ち, 視覚センサーや制御系の誤差が避けられないような実ロボットに対して十分うまく動作することを実証する。

## Point Formation Algorithm for Multiple Mobile Robots with Limited Visibility and Evaluating the Robustness against Errors\*

Yoshinobu OASA<sup>†</sup> Ichiro SUZUKI<sup>‡</sup> Masafumi YAMASHITA<sup>†</sup>

(<sup>†</sup>Hiroshima University <sup>‡</sup>Univ. of Wisconsin — Milwaukee)

Ando et al. proposed a point formation algorithm for mobile robots with limited visibility and show its correctness under the assumptions that (1) a robot is a point, and (2) sensors and control mechanisms are reliable. In this manuscript, we demonstrate that the algorithm is extremely robust against sensor and control errors, and therefore it is implementable on real robots.

\*本研究の遂行は一部以下の各機関からの補助金および援助を得て行なわれた。文部省科学研究費補助金(重点領域研究-07243219,08233219;一般研究(C)-06680324,07680360;基盤研究(C)-08680370),米国NSF(IRI-93707506),米国ONR(N00014-94-1-0284)。大阪大学基礎工学部日立寄付講座。

# 1 はじめに

Suzuki と Yamashita[2] は, 群ロボットの共有座標系獲得問題がロボットをある幾何図形に整列させる問題に帰着可能であることを動機付けとして, (1) ロボットを点, (2) ロボットの視野は無限, そして (3) ロボットは正確に動作する, という強い仮定を導入して整列問題を解析した. Ando 他 [1] は, 上記の問題をより現実的なモデル上で検討することを目指し, ロボットの無限視野に関する仮定を外して一点収束問題を検討した. このアルゴリズムは, 現在のロボットの位置だけを入力情報としてそのロボットの次の時点の位置を決めるので, ロボットが (有限だが) 十分に広い視野半径を持つ場合には, 自己安定, すなわち有限回のセンサーおよび制御の一時故障に耐えるという特長がある. しかし, 現実のロボットは, (1) 点ではなく, (2) 視野はそれほど広くなく, しかも (3) センサーおよび制御誤差がある. そこで, このような実ロボットに対しても安藤のアルゴリズムが有効であることを実証することを目的として, アルゴリズムの誤差に対する強度をシミュレーション実験によって評価した.

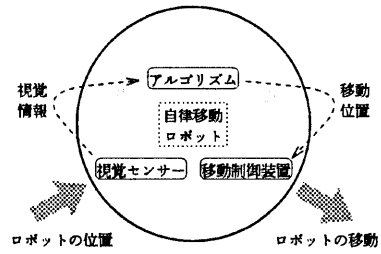


図 1: 自律移動ロボットのモデル.

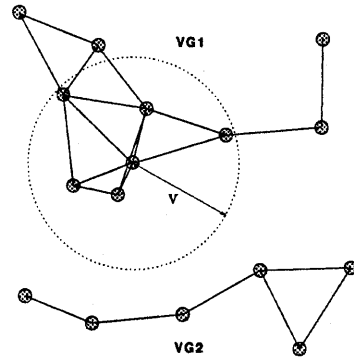


図 2: 可視グラフ.

## 2 群ロボット, 一点収束問題, アルゴリズム

群ロボットの形式的な定義は [1] に譲り, ここでは必要な範囲で対象とするロボットを非形式的に説明する. 視角センサーを持ち自律的に行動する, 移動ロボット  $r_i$  の一群  $R = \{r_1, r_2, \dots, r_n\}$  を考える (図 1). ロボットはそれ以外のセンサーや通信機能を持たない. すべてのロボットはハードウェアおよびソフトウェア的に同一であり, それらを制御する集中制御装置は存在しない (したがってロボットは非同期的に動作する). すなわち, 群ロボットのシステムでは, 各ロボットが自律的に, 1) ある時刻に目覚めて視野範囲内のロボットの位置を認識し, 2) 搭載するアルゴリズムに従って自分が向かう位置を計算し, 3) 直ちにその位置に移動する, という手続きを繰り返す.

各ロボットは同じアルゴリズムの下で動作し, こ

のアルゴリズムだけがロボットの持つ初期情報であり, それ以外の情報は初期化された時点では何も持たない. 特に, ロボットは共通の座標系を持たないと仮定する. その結果, すべてのロボットを一点に集合させる問題も, たとえば, “原点に移動する” といった簡単な命令によって解決できなくなっていることに注意せよ.

ここで, 一点収束問題を定義する. 無限視野のロボットを対象とする場合には, 一点収束問題は, 単純にロボットをある点に収束させる問題と定義できたが, 有限視野の場合には, 与えられた初期位置において互いに見えない位置におかれたロボット群の取り扱いを定める必要がある.

ロボットの初期位置  $P_0$  を決めると,  $R$  を節点集合とし, ロボット  $r_i$  と  $r_j$  が互いに “見える” 位置にあるとき, かつそのときに限り (無向) 枝  $(r_i, r_j)$  を持つような, 可視グラフ  $VG = (R, E)$  が決まる (図 2).

有限視野を持つ群ロボットに対する一点収束問題

を、どのような初期状態  $P(0)$  に対しても、 $VG$  の各連結成分に属するすべてのロボットを (連結成分ごとに異なるかもしれない) ある点に “集合” させる問題であると定義する。

### 3 Ando-Suzuki-Yamashita のアルゴリズム

Ando 他 [1] は以下に述べるアルゴリズム ASY を開発し、(1) ロボットは点、(2) ロボットの可視領域は半径  $V > 0$  の円盤、(3) 移動時間は無視でき (しかし一時点に移動できる距離はある小さな定数  $\epsilon > 0$  で抑えられる)、最後に、(4) ロボットのセンサーや移動制御に誤差は存在しない、と仮定するとき、ASY が正しく一点収束問題を解くことを厳密に証明した。ASY の厳密な記述は [1] に譲り、ここでは、その概略を非形式的に説明する。

ある時刻  $t$  の任意のロボット  $r$  を考え、 $r$  の可視範囲にあるロボットの位置を  $P_r(t)$  とする。ASY は、 $P_r(t)$  を入力とし、 $r$  が移動すべき点  $x$  を出力するアルゴリズムである。 $x$  は  $r$  が現在存在する点の  $\epsilon$ -近傍内の点である。

$P_r(t)$  に属する点に対する最小包含円の中心を  $c_r(t)$  とする。基本的には、ASY は  $r$  を  $c_r(t)$  にできる限り近付ける (図 3)。しかし、 $r$  の移動距離を制約する要因は  $\epsilon$  の他にもう 1 つ存在する。ASY は  $VG$  の連結成分が決してそれ以上細分されないことを保証するために、どのロボット  $r' \in P_r(t)$  についても、 $x$  が、 $r$  と  $r'$  の中点を中心とし半径が  $V/2$  の円内に存在するように  $r$  の移動距離を制約する (図 4)。なお、このような  $x$  の存在範囲は容易に  $P_r(t)$  から計算できる。

ASY は非常に故障に対して強いアルゴリズムである。定義から、ASY はどの初期状態からも正しく一点収束問題を解決する。したがって、可視グラフに属していた枝がロボットのセンサー誤差や制御誤差の結果消滅することはないという仮定の下で、このような誤差が無限回生することがなければ、ASY は正しく一点収束問題を解決するという意味で自己安定である。

しかしながら、この理論的結果は本章の冒頭に挙げたいくつかの強力な仮定に基づいているので、(例

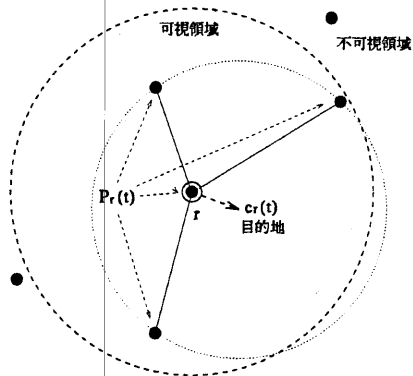


図 3: 一点収束アルゴリズム ASY.

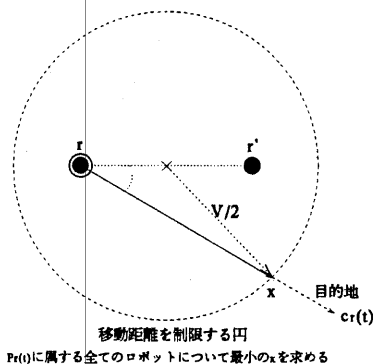


図 4: ASY における移動制限の説明図.

えば、点ではなく容積を持つ) 実ロボットに実装した場合においても、ASY が上記のように強力であるかどうかは検討の余地がある。次章では、シミュレーションによって ASY が実ロボットにおいても十分に強力な自己安定性を持つことを示す。

## 4 シミュレーション

### 4.1 シミュレーションモデルと評価尺度

[ロボットモデル] 各ロボットは視野半径 50 (単位距離) を持つ直径 1 の円盤であり、一度の移動における最大移動距離 (速度と呼ぶ) は  $\epsilon = 1$  である。ロボットは衝突回避時を除き、目的地  $x$  に直進する。衝突回避は速度を半分を落し、互いに左に回避すること

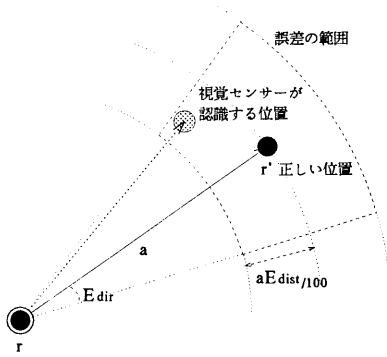


図 5: 誤差モデル

で行なう。(誤差の影響で衝突した場合には、衝突した地点で移動を中断する。) 同じ連結成分に属するすべてのロボットが半径  $d = 3\sqrt{n}/2$  の円内に集合したときをもって、ロボットが収束したと定義する。ここに、 $n$  はロボットの総数である。半径  $d$  は、 $n$  台のロボットがちょうど収容できるような大きさに設定してある。初期状態ではロボットは一辺が 200 の正方形内にランダムに置かれる。

[誤差モデル] 誤差としてセンサー誤差  $E_{sense}$  と制御誤差  $E_{contr}$  を考察する。センサー誤差、制御誤差ともに、方向誤差  $E_{dir}$  と距離誤差  $E_{dist}$  がある。誤差の尺度は方向誤差については最大ずれ角 ( $^\circ$ )、距離誤差については最大相対誤差 (%) である。例えば  $E_{sense} = e$  は、方向誤差、距離誤差ともに誤差が  $e$  であることを意味する。すなわち、方角が  $x^\circ$ 、移動距離が最大  $ax$  ずれることを意味する。ここに、 $-e \leq x \leq e$  は一様分布に従うランダム変数であり、 $a$  は正しい移動距離である (図 5)。

[評価方法] 評価尺度として、(1) 移動回数、(2) 直線移動距離 (初期位置と最終位置との直線距離)、そして (3) 可視連結成分数の変化、を採用した。

ロボット総数  $n$  を基本的なパラメータとして、各実験項目に対してそれぞれ試行を 100 回行ない、評価尺度のそれぞれについて最大値、平均値、分散などの基本的統計量を求める。

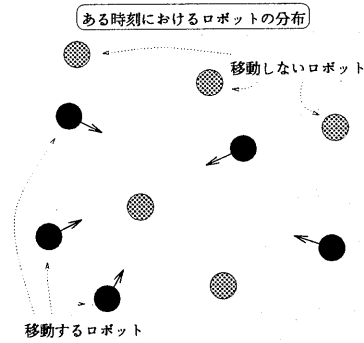


図 6: 非同期制御と同期制御

## 4.2 シミュレーション実験とその結果

この実験の困難さの一面は、検討されるべきパラメータの多様性にある。そこで、影響の薄いと考えられるパラメータをいくつか同定し、パラメータ数を減少させることから実験を開始する。

**実験 1 – 同期システムと非同期システム:** 視野半径を 50 に取り、誤差を排除して、ロボット移動の同期の程度がシステムに及ぼす影響を調べた。同時に移動するロボットの割合 (図 refFig) を 10-100% の間で変化させ、ロボット数  $n = 10, 25, 50, 100$  のそれぞれに対して、対象とする 3 つの評価尺度を測定した。(移動するロボット群はランダムに選択した。) 結果のグラフは省略するが、いずれの場合も、どの尺度も同期の程度には依存しないことが分かった。

**実験 2 – ロボットの陰の影響:** 容積を持つロボットシステムでは、あるロボットの裏に隠れたロボットはたとえ距離的に近くとも視野に入らず、アルゴリズムの実行において重要な影響を持つと考えられる (図 7)。そこで、ロボット数  $n$  をパラメータとして、ロボットの陰に入ったロボットを視野に入れない場合と、陰の影響を無視して、視野半径内にあるすべてのロボットが見える場合を比較検討した。ただし、誤差は排除してある。図 8 に結果の一部として、総移動回数 (一群のロボットが移動した総回数 - 図では STEP と記す) を示す。図から分かるように、少なく

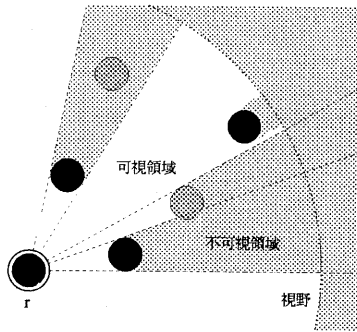


図 7: ロボットの陰

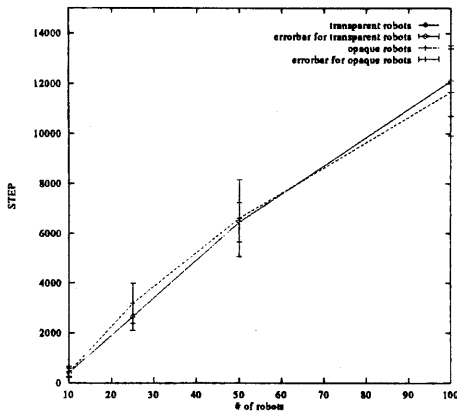


図 8: 実験 2 の結果. ロボットの陰を考慮した場合とそうでない場合についての, 総移動回数比較. (同期システム, 視野半径 50, 誤差無し.)

とも総移動回数の平均については陰の影響は薄い. しかしながら, ばらつきに対しては影響を見受けることができる (エラーバーは標準偏差で記述してある). しかしながら, 総直線移動距離 (一群のロボットが移動した総直線距離 - 図では DISTANCE と記す) については (図は省略するが) 両者の相違は無視できる程度であった.

以上の考察から, 同期の程度と陰の影響の有無をパラメータから排除し, 以下のすべての実験では対象は同期システムであり, 陰の影響は考慮しないことにした.

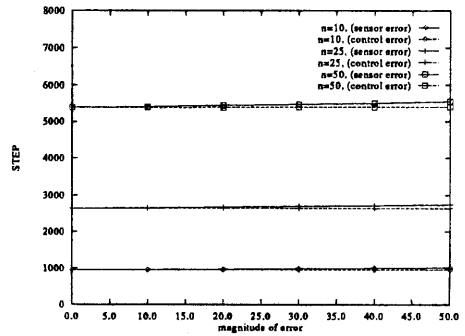


図 9: 実験 3 - 無限視野における誤差が平均総移動回数に及ぼす影響. (同期システム, 陰の影響無し.)

**実験 3 - 無限視野:** 視野半径がアルゴリズムに与える影響を概観するための基礎的資料として, 視野半径が無限の場合に誤差がシステムに与える影響を検討した. センサー誤差と制御誤差をそれぞれ 0 - 50 の間で変化させ, ロボット数  $n = 10, 25, 50$  のそれぞれの場合について, 対象となる評価尺度を測定した. (センサー誤差と制御誤差のどちらかをパラメータとしているとき, 他方の誤差は 0 としている.) 結果の一部として, 平均総移動回数と平均総直線移動距離をそれぞれ図 9 と 10 に示す.

結果として, (1) いずれの場合もシステムはほとんど誤差の影響を受けていないが, (2) 総移動回数においては, センサー誤差が制御誤差よりもわずかに強い影響を持つのに対して, (3) 総直線移動距離においては, 制御誤差がセンサー誤差よりもわずかに強い影響を持つことが見てとれる. (しかし, この差は重視しない方が正しいかもしれない.)

この結果は, 誤差が 50, すなわち角度が最大  $50^\circ$ , 距離が最大 50%, 程度ずれるような場合であっても, それがまったく収束速度に影響を与えないことを示しており, 非常に興味深い.

以上の結果を踏まえて, 視野半径が 50 の場合について, 誤差のシステムに与える影響を少し詳しく考察した.

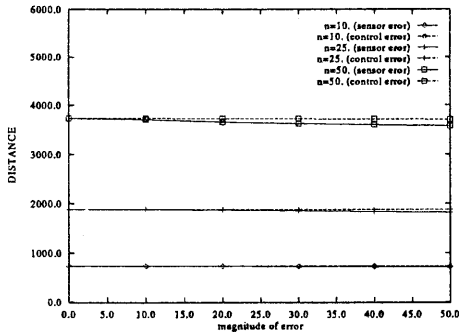


図 10: 実験 3 - 無限視野における誤差が平均総直線移動距離に及ぼす影響。(同期システム, 陰の影響無し.)

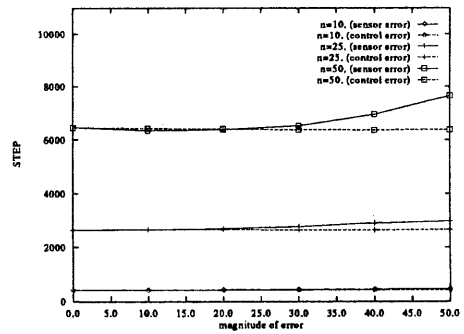


図 11: 実験 4 - 有限視野における誤差が平均総移動回数に及ぼす影響。(同期システム, 陰の影響無し, 視野半径 50.)

実験 4 - 有限視野: 視野半径を 50 に固定したことを除き, 実験 3 と同じ条件で実験を行なった. 結果の一部として, 平均総移動回数と平均総直線移動距離をそれぞれ図 11 と 12 に示す.

結果として, (1) 総直線移動距離においては, 制御誤差がセンサー誤差よりもわずかに強い影響を持つことが見てとれるが, その差はわずかで無視できる程度であるのに対し, (2) 総移動回数においては, ロボット数  $n$  が増加するにつれて, 誤差 30 を境として, センサー誤差が大きな影響を持ち始めることが明らかに見てとれる. しかしながら, 制御誤差が与える影響はほとんど観察されない. すなわち,  $n = 50, E_{sense} \geq 30$  の場合について, 総移動回数は急激に増加するにもかかわらず, 総直線移動距離は一定に保たれている. このことは, 大きな誤差にかかわらず, 最終的には誤差のない場合と同じような位置に収束しているからであると考えられる. (このことは, 別の実験により確かめられた.)

実験 5 - 方向誤差と距離誤差: 有限視野では, 実験 4 からセンサー誤差がシステムに重要な影響を与えることが明らかになった. そこで, 本実験では, センサー誤差の影響をより詳細に検討することにし, 実験 4 と同じ枠組ではあるが, 今回は, センサー誤差中の, 方向誤差と距離誤差のシステムに与える影響を検討した. 結果の一部

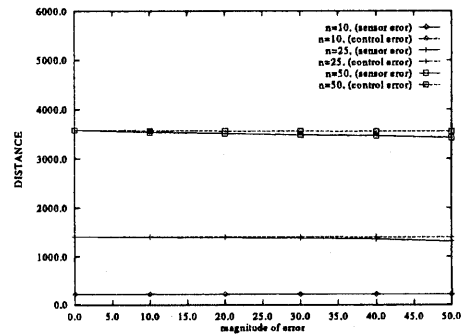


図 12: 実験 4 - 有限視野における誤差が平均総直線移動距離に及ぼす影響。(同期システム, 陰の影響無し, 視野半径 50.)

として, 平均総移動回数と平均総直線移動距離をそれぞれ図 13 と 14 に示す.

結果として, (1) 総直線移動距離は (実験 4 から予想されたとおり) 方向誤差と距離誤差のどちらの影響もまったく受けていないが, (2) 総移動回数においては, ロボット数  $n = 50$  に対しては, 誤差 30 を境として, 方向誤差と距離誤差の影響を徐々に受け始めることが分かる. 距離誤差の影響をより大きく受けていることが観察できるが, 図 11 の場合のように影響は大きくない.

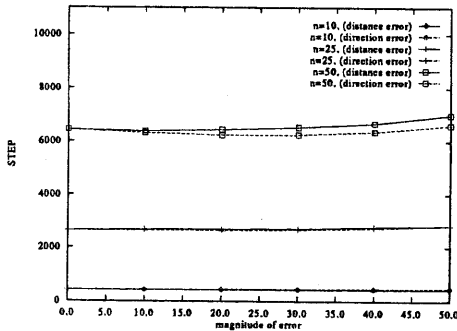


図 13: 実験 5 - 有限視野におけるセンサー誤差が平均総移動回数に及ぼす影響。(同期システム, 陰の影響無し, 視野半径 50, 制御誤差 0.)

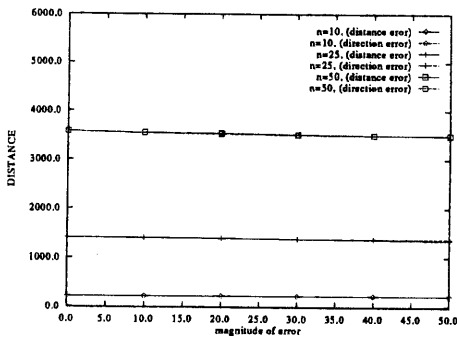


図 14: 実験 5 - 有限視野におけるセンサー誤差が平均総直線移動距離に及ぼす影響。(同期システム, 陰の影響無し, 視野半径 50, 制御誤差 0.)

## 5 おわりに

本報告では, ASY の実ロボットへの実装の可能性を吟味する目的で, その誤差に対する強度をシミュレーション実験によって検討した. 前章の実験結果から, 多くの実用的な応用例において誤差が 30 を越えて 50 に迫るような状況は考えにくいと思われるので, ASY が十分に実用に耐えるアルゴリズムであることが実証された. そこで今後, 東京理科大学原研究室で稼働するロボット群に実装し, ロボット群の振舞いを検討して行く予定である.

最後に, 実験結果が示すいくつかの矛盾を思わせ

る現象を検討するとともに, 残された問題を指摘する. 実験 3 と実験 4 を比較すると,  $n = 10, 25, 50$  のそれぞれの場合について, 総移動回数が, 無限視野では, 1000, 2700, 5300 であるのに対し, 視野半径が 50 の場合には 400, 2700, 6400 を示している.  $n = 10$  の場合について, 無限視野が有限視野よりこれらの尺度で劣る理由は, 定義から, 無限視野では常にすべてのロボットが 1 つの連結成分に属するのに対して, 有限視野では, 特に  $n$  が小さい場合には, 可視グラフは多くの連結成分に分かれ, その結果 ASY は高速に終了する.  $n$  が大きくなると, 多くの場合, 有限視野の場合においてもすべてのロボットが 1 つの連結成分に属し, したがって, 無限視野の効果が現れる ( $n=50$  の場合を見よ). データは省略するが,  $n = 50$  の場合, 90% 以上の試行では可視グラフが連結になり, また連結成分の変化もほとんど見られなかった (平均初期連結成分数は 1.20, 平均最終連結成分数は 1.29 であった).

総直線移動距離についても同じ傾向が観察されるが, さらに不思議なことに,  $n = 50$  の場合においても有限視野が健闘し, 大きな誤差の場合でも, わずかであるが無限視野を上回る評価を与えている. これは, 誤差のために連結成分が細分され, その結果  $n$  が小さい場合と同じ理由によって総直線距離が減少したと予想されるが, (上記の連結成分の変化回数から示すデータはこれを指示する), さらに検討を加える必要がある. この事実は, 前章で述べた矛盾点, すなわち  $n = 50, E_{sense} \geq 30$  においては, 総移動回数が増加するにもかかわらず, 総直線移動距離が減少する傾向にあることとも関連がある. さらに, 実験 3 と 4 から, 視野半径が誤差の強度に重要な関わりを持つことが明らかになりつつあるが, より小さな視野半径を持つ場合を検討することによって, さらに様々な事実が明らかになると期待している.

## 参考文献

- [1] H. Ando, I. Suzuki and M. Yamashita, "Formation and agreement problems for synchronous mobile robots with limited visibility," *Proc. of the 1995 IEEE Int'l Symp. on Intelligent Control*, Aug. 1995, pp. 453-460.

- [2] I. Suzuki and M. Yamashita, "A Theory of Distributed Anonymous Mobile Robots — Formation and Agreement Problems," TR-94-07-01, Dept. of Electrical Eng. and Computer Science, Univ. of Wisconsin — Milwaukee, July 1994.