

特別研究報告

題目

サイボーグインセクトによる迅速な被災者発見のための自己組織型
移動制御手法の提案

指導教員

若宮 直紀 教授

報告者

北浦 直

2019 年 2 月 12 日

大阪大学 基礎工学部 情報科学科

平成 30 年度 特別研究報告

サイボーグインセクトによる迅速な被災者発見のための自己組織型移動制御手法の提案

北浦 直

内容梗概

主な用語

目 次

1	はじめに	3
2	サイボーグインセクト	5
3	フロッキング	6
4	サイボーグインセクトのモデル	7
4.1	進行方向ベクトルの初期値決定	7
4.2	進行方向ベクトルを変更するかの判定	7
4.3	回転角の決定	8
4.4	1step 分前進	8
4.5	面と面が構成する境界線に接したかの判定	8
4.6	次の進行方向ベクトルの決定	8
4.6.1	境界線に平行なベクトル	9
4.6.2	境界線に平行なベクトルの逆ベクトル	9
4.6.3	現在の方向を保ちつつ, 新しく接した面に平行なベクトル	9
4.6.4	ベクトルの決定	10
5	制御モデル	12
5.1	斥力の合計を計算	13
5.1.1	サイボーグインセクトからの斥力	13
5.1.2	障害物からの斥力	13
5.1.3	斥力の合計	14
5.2	進行方向の決定	14
5.3	パラメータ設定	14
6	実験結果	16
7	おわりに	17
	参考文献	18

1 はじめに

災害が起きた場合、倒壊した建物内に被災者が取り残されることが起こりうる。倒壊した建物内に捕らわれた被災者の探索において、現在は災害救助犬やスコープカメラなどの人間の能力を補助するような手法が広く使われている [1]。しかし、倒壊した建物内は狭い隙間であったり、アスベストやほこり、危険物の存在により人間や災害救助犬が入れないような環境である可能性がある [2]。また、救助活動の中で2次災害が起きてしまう事例も少なくない。そこで、現在の探索手法では探索できないような狭い空間を探索可能で、探索中の2次災害の危険性を低くすることができるサイボーグインセクトを用いた被災者探索の研究がなされている [3]。

サイボーグインセクトを被災者探索に活用するための研究として CINEMa(Cyborg Insect Networks for Exploration and Mapping) があげられる [4]。この研究の中では、サイボーグインセクトへ制御を与えて任意の方向へ向かわせたり、サイボーグインセクトが位置推定をするためのアルゴリズムの実験がされていたりする。しかし、CINEMa はがれきなどが散乱するような悪条件下における被災者探索のための構成要素の確立は行われているが、それらを用いて被災者探索を効率的に行うためのアルゴリズムなどは提案されていない。つまり、群で探索しているにも拘らず複数個体がほぼ同一の場所を探索することで空間全体を探索するのにかかる時間が増加してしまう恐れがある。

そこで、サイボーグインセクトの群れが効率的に空間全体を探索できるような制御を与えたい。しかし、救助者は倒壊した建物内の詳細な環境を知ることはできないため、サイボーグインセクト自身が自律的に制御を行い空間探索を行う必要がある。

また、複数のレスキューロボットが行う自律的なアルゴリズムの1つであるフロッキングという手法がある。フロッキングとは、周囲のロボットに対して、Separation, Alignment, Chesion の3つの動作をとることで複数個体が群れとなって動くようなアルゴリズムである。このフロッキングを被災者探索に適用される研究が行われているが、フロッキングをサイボーグインセクトに適用して被災者探索を実現している研究はなされていない。また、これらの研究でなされているフロッキングは基本的に連続で操作を行っており、断続的に制御を行っている研究は少ない。しかし、サイボーグインセクトに与えるフロッキングの制御は生物に電気信号を流すためできるだけ頻度が低いことが望ましい。

そこで、本研究では CINEMa で提案されているようなサイボーグインセクトの群れに対するフロッキング制御手法の提案を行う。サイボーグインセクトのモデルと断続的な制御をシミュレーション上に実装し、制御されていない場合の探索と制御された場合の探索を比較する。ここでは、サイボーグインセクトに対する断続的なフロッキング制御でも探索にかかる時間の短縮ができることを確認する。

以降，第2章では，本研究に用いるサイボーグインセクトに関する説明を行い，第3章でフロッキング動作の説明を行う．第4章でサイボーグインセクトが制御をされていない場合の動作について述べ，第5でサイボーグインセクトへ与える制御について述べる．第6章では実験結果について述べる．第7章では本研究のまとめと今後の課題について述べる．

2 サイボーグインセクト

この章では、本研究で考えるサイボーグインセクトの概要と、サイボーグインセクトに取り付けられる機材、サイボーグインセクトに関する条件の説明を行う。

サイボーグインセクトとは、文献 [3] で示されているような、昆虫に機械を埋め込み人間が操作できるようにした昆虫に計器類を乗せて被災者探索を可能にしたものを指す。

サイボーグインセクトには、神経刺激による移動制御、センシング、サイボーグインセクトの位置推定、および救助者や周りのサイボーグインセクトとの通信のための機材が乗せられている。

サイボーグインセクトに対する制御は、壁で仕切られた迷路の指定された経路を通るなどの実験を通して、サイボーグインセクト自身の動きに加え一定の操作ができることを示している [4]。

センシングのために、様々なセンサをサイボーグインセクトに積載すると文献 [4] で示されている。文献 [4] では、3 方向の加速度計、ジャイロスコープ、磁力計、指向性マイクと全指向性マイク、ブザーとスピーカー、赤外線検知器、ガス検知器がサイボーグインセクトへ取り付ける機材として挙げられていた。

被災者のセンシングに関して、複数のマイクでの音の強度を比較することで音源方向の特定をしてサイボーグインセクトから見た被災者の方向を特定することができていた。また、サイボーグインセクトに取り付けられたマイクとスピーカーを利用することによって、位置推定を行っていた。

今研究では、神経刺激による移動制御は十分に可能として考える。被災者や障害物のセンシングは、半径 2m 以内ならできると考え、距離精度は 10cm 単位、角度の精度は 10 度単位として考える。また、サイボーグインセクト同士の通信可能範囲は 5m とする。

3 フロッキング

この章では、フロッキングについて説明する。

フロッキングとは、自然界の鳥や魚の群れで見られるような、複数の個体が同じ目標に向かい連動して動くような行動である。この自然界で見られる行動を限定された範囲のセンシング能力とコミュニケーションにより再現したモデルが知られている。文献 [5] では衝突回避、中心移動、方向調整の3つの動作によりフロッキングのモデル化を行っている。それぞれの動きは以下のようにとなっている。

衝突回避の動作により、フロッキングを行っている個体群はそれぞれの距離を一定以上に保つことができる。衝突回避のために、まず、検知できる範囲の中にいる他の個体を探す。その後、近くにいた個体から自分へのベクトルを計算し、正規化を行い、さらに自分と見つけた個体との距離でそのベクトルを割る。これを一定距離内の周りの個体すべてに繰り返すことで衝突回避の動作のためのベクトルが得られる。

中心移動の動作によって、フロッキングを行っている個体群は離れることなく群れとしての形を保つことができる。中心移動のために、まず、衝突回避と同様に検知できる範囲の中にいる個体を探す。そして自分を除いた検知できる範囲の中にいる個体群の重心を計算して、その重心へのベクトルを計算する。

方向調整の動作によって、フロッキングを行っている個体群は同じような速度、向きで動くことができる。方向調整のために、衝突回避と同様に検知できる範囲の中にいる個体を探す。そして、それらの個体群の速度の平均を求め、自分の速度との差を求める。この速度の差が方向調整のためのベクトルとなる。

最後に、上記3つのベクトルからフロッキングを行うためのベクトルを求める。この時に、それぞれを足すだけでもフロッキングとして動く場合も存在するが、基本的には、それぞれを正規化したのちに、重み付け係数を用いてスケーリングすることが有用であるといわれている。

さらに、上記のフロッキングの基本的な動作に加えて、障害物ある環境でのフロッキングを考えた研究もなされている [6]。また、フロッキングを用いて被災者探索を行う研究も存在する。例えば、文献 [7] では群ロボットによる空間探索において、フロッキングを用いて各ロボットの探索領域の重複の軽減を行っている。

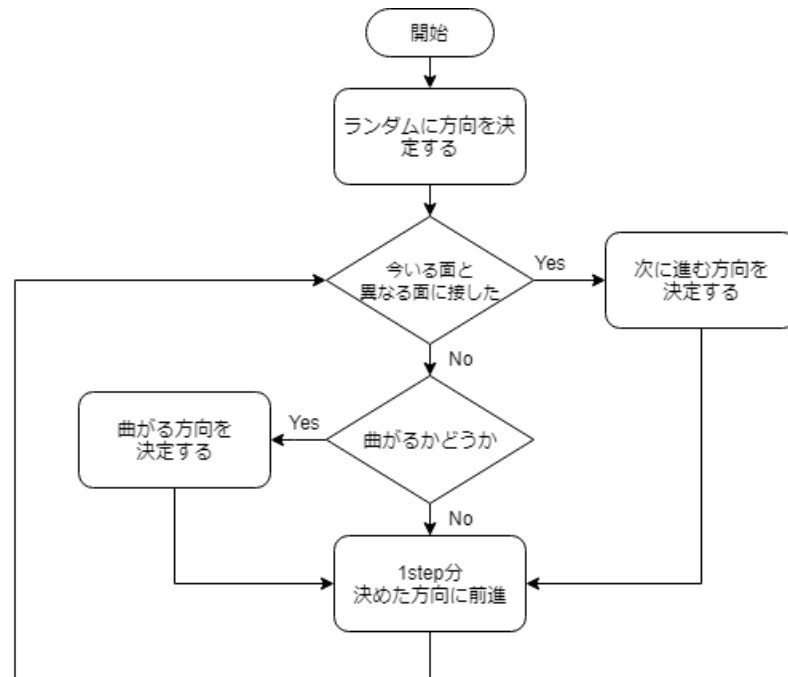


図 1: 制御されていないサイボーグインセクトのアルゴリズムのフローチャート

4 サイボーグインセクトのモデル

この章では、サイボーグインセクトが制御を受けていない場合の動きのモデルについて説明を行う。

制御をされていないサイボーグインセクトは、図 1 のフローチャートに示されているアルゴリズムに従って行動する。このアルゴリズムでの 1step は 0.1 秒とする。また、サイボーグインセクトの速度は 60cm/s とする [?].

4.1 進行方向ベクトルの初期値決定

探索が開始されたときに実行される。どの方向に進むのかをランダムに決定する。以下の章ではサイボーグインセクトが進む方向の単位ベクトルを進行方向ベクトルと呼ぶ。

4.2 進行方向ベクトルを変更するかの判定

サイボーグインセクトは、1step 毎に現在の進行方向ベクトルを別の進行方向ベクトルに変更するかの判定を行う。この時に、進行方向ベクトルを変更する確率は 1%とする。

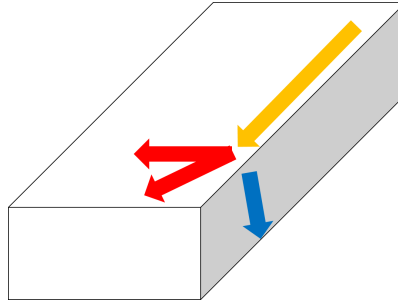


図 2: 回転できるベクトル (赤) と回転できないベクトル (青) の例

4.3 回転角の決定

現在の進行方向ベクトルからどれだけ回転するかを決定を行う。この時、現在の進行方向ベクトルから角度が大きく異なる進行方向ベクトルほど選択する確率が低くなるように設定した。また、現在接している面から別の面へ移動するようなベクトルには回転しないものとする。図 2 のように赤い矢印であらわされるベクトルには回転する可能性が存在し、青のベクトルのように接する面が変わるベクトルには回転しないものとする。

ここで、回転する角度は、平均 0、分散 $\frac{\pi}{6}$ の正規分布に従って決定される。

4.4 1step 分前進

サイボーグインセクトは、進行方向のベクトルに 1step 分前進する。また、途中で障害物があったり接している面が途切れたりなどで 1step 分の距離を進めない場合は、進めるところまで進み、その位置で停止する。

4.5 面と面が構成する境界線に接したかの判定

サイボーグインセクトは、現在いる場所が面と面が構成する境界線に接しているかどうかを 1step 毎に判定を行う。

4.6 次の進行方向ベクトルの決定

サイボーグインセクトは次の進行方向ベクトルを計算する。次に進行方向ベクトルの候補として、図 3a のような 3 種類のベクトルを計算する。この時、サイボーグインセクトがすでに境界線に沿って進んでいる場合は、図 3b のように、新しく接した境界線に沿うような 2 つのベクトルを計算する。

1. 境界線に平行なベクトル

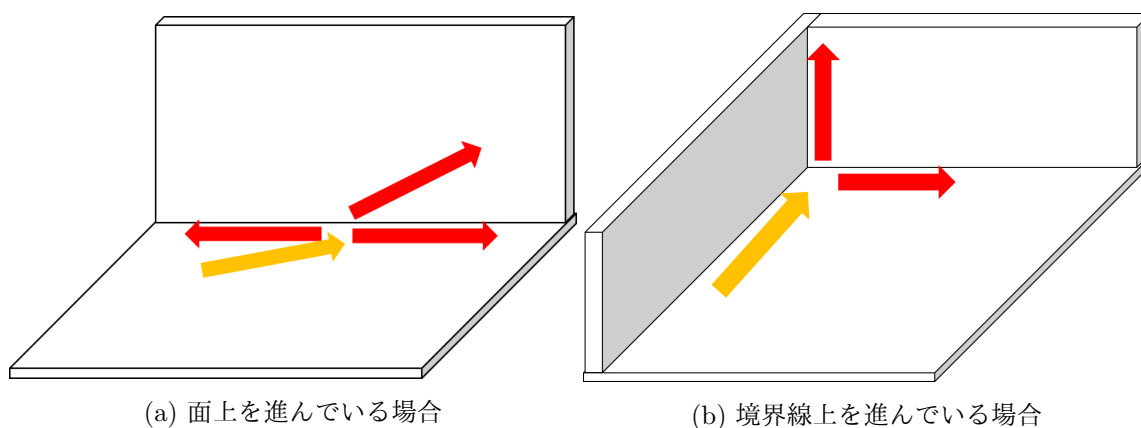


図 3: 次に進む方向のベクトル

2. 境界線に平行な 1. の逆ベクトル
3. 現在の方向を保ちつつ、新しく接した面に平行なベクトル

それぞれの具体的な計算方法は以下の章で説明する。

4.6.1 境界線に平行なベクトル

新たに接した境界線からベクトルを求めて、そのベクトルに平行な単位ベクトルを次の進行方向ベクトルの候補とする。3b のように境界線上を進んでいる場合、新たに接した境界線が 2 つあるため、その 2 つを次の進行方向ベクトルの候補とする。

4.6.2 境界線に平行なベクトルの逆ベクトル

4.6.1 章で求めたベクトルの逆ベクトルを次の進行方向ベクトルの候補とする。

4.6.3 現在の方向を保ちつつ、新しく接した面に平行なベクトル

現在の進行方向ベクトルを新たに接した境界線を軸にして回転することで次の進行方向ベクトルの候補を計算する。現在の進行方向ベクトルの境界線に垂直な成分から、新しく検知した面に平行で境界線に垂直なベクトルに反時計回りでなす角を ϕ としたときに、現在の進行方向ベクトル \vec{v} を ϕ だけ回転することで、次の進行方向ベクトルの候補 \vec{l} が得られる。

回転角 ϕ は図 4b のように、現在の進行方向ベクトルと新たに検知した面に平行で境界線に垂直なベクトルを求めて、進行方向ベクトルの垂直成分から反時計回りになす角を求める。

境界線に平行なベクトルを \vec{b} とした際に、次の進行方向ベクトルの候補 \vec{l} は式 1 によって

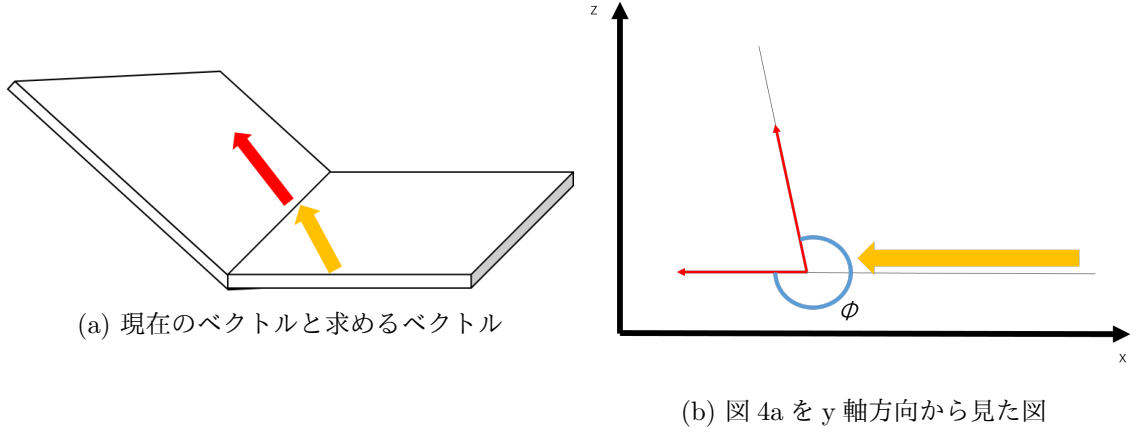


図 4: なす角 ϕ の位置

あらわされる.

$$\vec{l} = R_{\vec{b}}(\phi)\vec{v} \quad (1)$$

ここで $R_{\vec{b}}(\phi)$ は境界線に平行なベクトル \vec{b} を軸に角度 ϕ だけ回転させる回転行列とする. 回転行列 $R_{\vec{b}}(\phi)$ は式 2 で計算される.

$$R_{\vec{b}}(\phi) = \begin{pmatrix} b_x^2(1 - \cos \phi) + \cos \phi & b_x b_y(1 - \cos \phi) - b_z \sin \phi & b_z b_x(1 - \cos \phi) + b_y \sin \phi \\ b_x b_y(1 - \cos \phi) + b_z \sin \phi & b_y^2(1 - \cos \phi) + \cos \phi & b_y b_z(1 - \cos \phi) - b_x \sin \phi \\ b_z b_x(1 - \cos \phi) - b_y \sin \phi & b_y b_z(1 - \cos \phi) + b_x \sin \phi & b_z^2(1 - \cos \phi) + \cos \phi \end{pmatrix} \quad (2)$$

4.6.4 ベクトルの決定

4.6.1 章から 4.6.3 章で計算したベクトル \vec{l}_i から, 次の進行方向ベクトルを決定する. 次の進行方向ベクトルは, 基準ベクトルとの差異による確率に従って決定される. 基準ベクトルは, 現在の進行方向ベクトルと重力ベクトルとの和によって計算される. まず, 現在の方向ベクトル \vec{v} と重力ベクトル \vec{g} から基準となるベクトル \vec{a} を計算する式は以下の通りである.

$$\vec{m} = \vec{v} + \frac{49}{60}\vec{g} \quad (3)$$

$$\vec{a} = \frac{\vec{m}}{|\vec{m}|} \quad (4)$$

ここで, \vec{g} の係数は, 0.1 秒分の自由落下の計算から求めている. また, 単位ベクトルである \vec{l}_i と合わせるために \vec{a} も単位ベクトルとなるように求めている.

次に、各ベクトルを選択する確率は式 5 によって計算される。

$$P(\vec{l}_i) = \frac{1}{n-1} \left(1 - \frac{|\vec{a} - \vec{l}_i|}{\sum_{k=1}^n |\vec{a} - \vec{l}_k|} \right) \quad (5)$$

ここで、 \vec{l}_i は 4.6.1 章から 4.6.3 章で計算した次の進行方向ベクトルの候補で、 $i = (1, \dots, n)$ である。

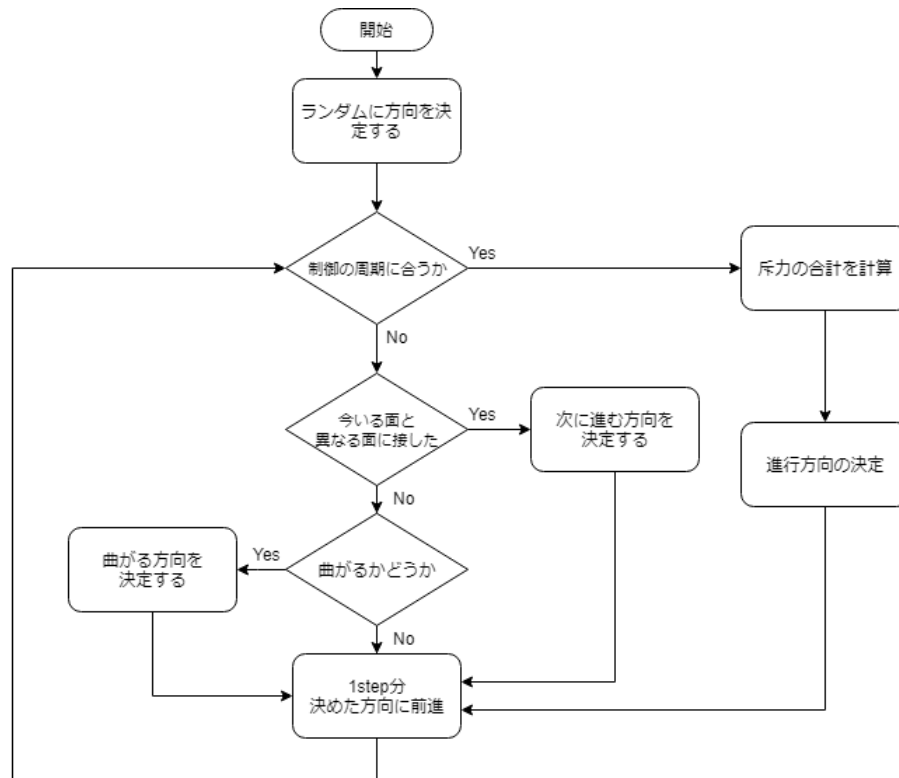


図 5: サイボーグインセクトを制御する場合のアルゴリズムのフローチャート

5 制御モデル

この章では、4 章で説明したサイボーグインセクトのモデルへの制御を説明する。

この制御の目的は、空間全体の探索にかかる時間を短縮することである。また、サイボーグインセクトへの負担や消費電力の軽減のため、サイボーグインセクトに対して常に制御を行うのではなくある周期ごとに断続的な制御を与える。

フロッキングによる空間探索の手法 [] と同様に、探索の冗長性を小さくすることで探索にかかる時間の軽減されると一般的に言うことができる。そこで、複数のサイボーグインセクトが同一の領域を探索することがないように、フロッキングの衝突回避のアルゴリズムを応用する。また、本研究でサイボーグインセクトに用いると仮定しているマダガスカルゴキブリは、一般的に障害物に沿って移動することが知られている。同様の場所を繰り返し移動することで、探索が冗長になってしまう可能性があると考えられる。よって、壁に沿って移動している場合は壁から離れさせるような制御を行うことで、探索にかかる時間の軽減を目指す。

制御を加える場合の全体のアルゴリズムは図 5 のフローチャートであらわされる。

ここでの制御の周期 T はパラメータとする。

5.1 斥力の合計を計算

あるサイボーグインセクトに対して、距離の近いサイボーグインセクトと障害物から斥力が働くとして考え、その斥力の合計を計算する [?].

5.1.1 サイボーグインセクトからの斥力

まず、周囲のサイボーグインセクトと通信を行い、通信距離内にサイボーグインセクトがいるかどうかを確認する。自身の通信距離内にサイボーグインセクトがいた場合、次のようにして斥力を求める。座標 (x, y, z) にいるサイボーグインセクトが、座標 (x_a, y_a, z_a) にいるサイボーグインセクト a から受ける斥力 \vec{f}_a は式 6 で求められる。

$$\vec{f}_a = \left(\frac{x - x_a}{d_a^2}, \frac{y - y_a}{d_a^2}, \frac{z - z_a}{d_a^2} \right) \quad (6)$$

個体 a との距離は式 7 により求められる。また、この時の距離 d_a は通信可能距離である 5m を 1 とするような正規化を行う。

$$d_a = \sqrt{(x - x_a)^2 + (y - y_a)^2 + (z - z_a)^2} \quad (7)$$

また、通信距離内にいるサイボーグインセクトが複数存在する場合は、以下の式で斥力の合計を計算する。ここで、 A は通信距離内にいるサイボーグインセクトの集合とする。

$$\vec{F}_A = \sum_{k \in A} \vec{f}_k \quad (8)$$

5.1.2 障害物からの斥力

この制御は、距離 r 以内にある障害物から斥力を計算することで実現する。ここでは、今自分がいる面からは斥力は受けないものとして考える。

この場合の斥力を求める式は式 9 となる。また、この時の距離 d_o はパラメータ r で正規化を行う。

$$\vec{o} = \left(\frac{x - x_o}{d_o^3}, \frac{y - y_o}{d_o^3}, \frac{z - z_o}{d_o^3} \right) \quad (9)$$

$$d_o = \sqrt{(x - x_o)^2 + (y - y_o)^2 + (z - z_o)^2} \quad (10)$$

また、周囲の障害物からサイボーグインセクトにかかる斥力の合計は以下の式であらわされる。

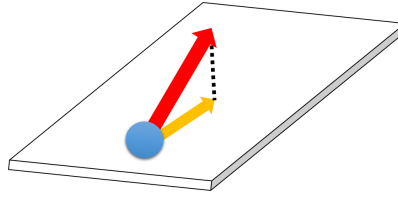


図 6: 斥力の合計と決定した進行方向ベクトル

$$\vec{F}_o = \sum_{k \in O} \vec{o}_k \quad (11)$$

O はサイボーグインセクトから距離 r 以内にある障害物の集合である．

5.1.3 斥力の合計

5.1.1 章と 5.1.2 章で求められる斥力を足すことで，ある個体にかかる斥力の和が計算される．よって，斥力 \vec{F} は α をパラメータとして式 12 であらわされる．

$$\vec{F} = \alpha \vec{F}_A + (1 - \alpha) \vec{F}_o \quad (12)$$

5.2 進行方向の決定

計算した斥力から次の進行方向ベクトルを決定する．

5.1 章で計算した斥力が今いる面に対して垂直な成分を持っている場合は，今いる面に射影したベクトルを次の進行方向ベクトルとする．図 6 の赤い矢印が他のサイボーグインセクトや障害物から受ける斥力の合計とした場合，今いる面に平行な単位ベクトルである黄色の矢印が実際に決定される進行方向ベクトルとなる．

5.3 パラメータ設定

この章では，サイボーグインセクトを制御する場合に考えられるパラメータの設定を決定した．

パラメータとして，以下の 4 つを設定する．

N サイボーグインセクトの群れの数 (匹)

T 制御を行う周期 (step)

r 障害物から斥力を受ける最大距離 (m)

α 斥力の合計するときの重み

N は、今回簡単のためとして 100 を考えている。 r は、壁沿いに動くことが多いことを理由として障害物からの斥力を設定しているため、大きい値である必要はないと考えた。そこで、壁から 0.2m を基準とすることにした。 α は、 r に依存すると考えられる。サイボーグインセクト同士の探索領域が重ならないようにするには、サイボーグインセクト同士の距離が 4m 以上である必要がある。そこで、壁から離れる制御とサイボーグインセクト同士が離れる制御に関して、壁沿いに動いている状態で、壁の逆側 4m の距離にサイボーグインセクト 1 匹がいる場合は斥力が釣り合っほしいと考えた。そこで、 α を求めるために式 13 が成り立つ。

$$\alpha = \quad (13)$$

また、以上のパラメータを設定した場合に T を変化させながら実験を行い探索率の推移を比較する。

6 実験結果

7 おわりに

参考文献

- [1] Wen-Ta Chiu, Jeffrey Arnold, Yaw-Tang Shih, Kuang-Hua Hsiung, Hsueh-Yun Chi, Chia-Huei Chiu, Wan-Chen Tsai, and William C. Huang. A Survey of International Urban Search-and-rescue Teams following the Ji Ji Earthquake. *Disasters*, vol.26, no.1, 2002, pp.85-94
- [2] Casper J and Murphy R R. Human-robot interactions during the robot-assisted urban search and rescue response at the world trade center. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 33(3), 2003, 367-385
- [3] Alper Bozkurt, Robert F. Glimour, Ayasa Sinha, David Stern, and Amit Lal. Insect-Machine Interface Based Neurocybernetics. *IEEE Trans Biomedical Eng.*, vol.56, no.6, 2009, pp.1727-1733
- [4] Alper Bozkurt, Edgar Lobaton, and Mihail Sichitiu. A Biobotic Distributed Sensor Network for Under-Rubble Search and Rescue. *Computer*, vol.49, no.5, pp.38-46, 2016
- [5] Craig Reynolds. Steering behaviors for autonomous characters. In *Proceedings of Game Developers Conference*, pages 763-782, 1999.
- [6] Ali E. Turgut, Hande Celikkanat, Fatih Gokce, and Erol Sahin. Self-organized flocking in mobile robot swarms. *Swarm Intelligence*, vol.2, pp97-120, 2008
- [7] Noury Bouraqadi, and Arnaud Doniec. Flocking-Based Multi-Robot Exploration. *Conttol Architectures of Robots*. 2009