

2012年度 卒業論文

リアルタイム 3DCG におけるニホンミツバチの
行動を表現する手法の提案

指導教員：渡辺 大地 講師
三上 浩司 准教授

メディア学部 ゲームサイエンスプロジェクト
学籍番号 M0109501
寺岡 優

2012年度 卒業論文概要

論文題目

リアルタイム 3DCG におけるニホンミツバチの
行動を表現する手法の提案

メディア学部

学籍番号：M0109501

氏名

寺岡 優

**指導
教員**

渡辺 大地 講師
三上 浩司 准教授

キーワード

3DCG、群知能、ミツバチ、蜂球、AI

近年、コンピュータの高性能化やコンピュータグラフィックスの表現技術の向上に伴い、コンピュータ内でリアルタイムに動かすことができる3次元形状モデルの数が増加した。それにより、数百を超える、生物の形状を模した3次元形状モデルが群れとして動く様子を、1度に表現することも可能になった。しかし、制作者が数百を超える3次元形状モデルの一体一体にそれぞれ状況、集団に合った動作命令をするのは、面倒で時間がかかる。その負担を緩和する手法としてboidアルゴリズムと呼ばれるものがある。結合、整列、分離と呼ばれる3つのシンプルなルールを連携することで、多くのモデルが群れとして動く様子を表現することができる。しかし、特殊な集団行動を行う生物をboidアルゴリズムだけで表現するのは難しい。本研究で取り上げるニホンミツバチは、蜂球形成行動と呼ばれる特殊な集団行動を行う。蜂球形成行動は、外敵であるスズメバチに向かって取り囲むように襲いかかって球体に近い形を作る行動である。この球体を蜂球と呼び、蜂球を形成しているニホンミツバチが発する熱により、スズメバチを蒸し殺すことができる。また、個体の位置によって上昇移動や下降移動、方向転換をする群飛飛翔を行う。本研究では、ニホンミツバチの集団行動を表現するために、boidアルゴリズムの機能を追加、変更する手法を提案する。また、この手法を使って実験をし、ミツバチの集団行動を表現することができた。

目次

第1章	はじめに	1
1.1	研究の背景と目的	1
1.2	本論文の構成	3
第2章	ミツバチの特徴	4
2.1	種類と防衛行動	4
2.2	飛翔	6
第3章	ニホンミツバチの群れの動きを 表現する手法	8
3.1	モデルの基本的な制御方法	8
3.2	boid アルゴリズム	10
3.2.1	結合ルール	12
3.2.2	整列ルール	15
3.2.3	分離ルール	17
3.2.4	boid アルゴリズムによる位置の更新	20
3.3	追加・変更するルール	21
3.3.1	群飛ルール	21
3.3.2	オオスズメバチに襲いかかるためのルール	25
3.4	boid アルゴリズムと追加したルールによる位置の更新	31
第4章	評価	33
4.1	群飛飛翔を行うニホンミツバチの群れ	34
4.2	蜂球形成行動を行うニホンミツバチの群れ	38
4.3	考察	41
第5章	まとめ	42
	謝辞	43
	参考文献	44

目次

2.1	蜂球 (Takahashi による画像)	5
2.2	ミツバチの群飛飛翔 (「花を増やそう!みつばち百花」より引用)	6
2.3	蝶と蜂の目的地への飛翔	7
3.1	モデルのローカル座標系と回転運動	10
3.2	boid アルゴリズム適用前	11
3.3	boid アルゴリズム適用後	11
3.4	仲間の影響を受ける範囲内のモデル	11
3.5	結合	13
3.6	整列	15
3.7	分離	18
3.8	上昇移動の影響を受ける範囲	22
3.9	群れの中心を通過して上昇する様子	23
3.10	群れの頂上から下降して周辺へと向かう様子	23
3.11	引力	30
3.12	斥力	31
4.1	使用したモデル	33
4.2	群飛飛翔を行うニホンミツバチの群れ (0 秒)	34
4.3	群飛飛翔を行うニホンミツバチの群れ (2 秒)	34
4.4	群飛飛翔を行うニホンミツバチの群れ (4 秒)	35
4.5	群飛飛翔を行うニホンミツバチの群れ (6 秒)	35
4.6	群飛飛翔を行うニホンミツバチの群れ (8 秒)	36
4.7	基本ルールのみでの群れ	37
4.8	群飛ルールを追加した群れ	37
4.9	基本ルールのみでの群れ	37
4.10	群飛ルールを追加した群れ	37
4.11	蜂球形成行動を行うニホンミツバチの群れ (0 秒)	38
4.12	蜂球形成行動を行うニホンミツバチの群れ (1 秒)	39
4.13	蜂球形成行動を行うニホンミツバチの群れ (2 秒)	39
4.14	蜂球形成行動を行うニホンミツバチの群れ (3 秒)	39

4.15 蜂球形成行動を行うニホンミツバチの群れ(4秒)	40
4.16 蜂球形成行動を行うニホンミツバチの群れ(50秒)	40
4.17 蜂球形成行動を行うニホンミツバチの群れ(50秒)を拡大	40

第 1 章

はじめに

1.1 研究の背景と目的

近年、コンピュータの高性能化やコンピュータグラフィックス (以下、CG) の表現技術の向上に伴い、コンピュータ内でリアルタイムに動かすことができる 3次元形状モデル (以下、モデル) の数が増加した。それにより、数百を超える、生物の形状を模したモデル (以下、生物モデル) が群れとして動く様子を、一度に表現することも可能になった。人やアリ、鳥 [1] などの様々な生物が、集団として動く様子も表現できる。しかし、制作者が数百を超える生物モデルの一体一体にそれぞれ状況、集団に合った動作命令をするのは、面倒で時間がかかる。そのため、多くのモデルをシンプルなルール (AI:Artificial Intelligence) で連携して動くようにできれば、制作者の負担を減らすことができる。それを実現するための技術として群知能 [2] と呼ばれるものがある。

群知能とは、単純な機能しか持たないルールを連携することで、本物に近い、生物の群れの動きを表現することができるというものである。群知能を行うための有名な手法として、1987年に Reynolds が考案した boid アルゴリズムがある [3][4]。結合、整列、分離という 3つのルールを連携して、複数の生物モデルに適用し、表現することができる。boid アルゴリズムは理解が簡単で利用しやすいため、多くの研究者が生物の群れのシミュレーションに利用している。また、ゲーム制作や映画制作に利用した制作者もいる [5][6]。しかし boid アルゴリズムだけでは、より本

物に近い群れを表現することは難しい。そのため多くの研究者は、実現したい生物の動きの特徴を、生物モデルを構成するパーツの動作に取り入れたり、boid アルゴリズムのルールに新たなルールを追加したりすることで、より本物に近い群れの動きを表現している。朱ら [7] は、トンボの飛翔を表現するために、トンボモデルの翅が、実際のトンボの翅のはばたき動作の通りに動くようにし、その翅のはばたき動作から力を求めた。その力を利用することで飛翔する飛翔モデルを作り、boid アルゴリズムのルールの一部を利用することで、複数のトンボの飛翔を表現した。石塚 [8] は、マイワシの群れの遊泳を表現するために、マイワシの特徴である、一箇所に留まり遊泳する動きを実現するためのルールを boid アルゴリズムに追加した。これらの研究のように、boid アルゴリズムを応用することで様々な生物の群れの動きが表現できる。本研究では、boid アルゴリズムを応用し、ニホンミツバチの群れの動きを表現する。

蜂の群れの動きの表現に関する研究では、社会性昆虫としての特徴を対象としたものが多い。社会性昆虫とは、ミツバチやアリなどのように、群れの中で多様な仕事の分業を行い、お互いに協力し合うことによって群れ全体の生活を維持する昆虫のことである。古川ら [9] は、巣の中の複数のミツバチを採蜜係と貯蔵係に分け、蜜の貯蔵行動のシミュレーションを行い、蜜需要量によって、採蜜係と貯蔵係のミツバチの行動が変化することを表現した。Karaboga[10] は、ミツバチが採蜜するための場所を複数用意し、ミツバチダンスと呼ばれる、餌の場所を仲間に知らせる動きによって、採蜜に向かう場所が変化することを表現した。

これらのように、ミツバチの行動を表現したものはいくつかあるが、ミツバチの飛翔時の特徴や蜂球形成行動を表現する研究は、私が知る限りない。蜂球形成行動とは、ニホンミツバチという蜂が、外敵であるスズメバチに対して行う防衛行動である。複数のニホンミツバチが、襲ってきたスズメバチに対して、突進して取り囲んでいき、球体に似た形を作る。これが蜂球である。蜂球を形成するニホンミツバチが発する熱により、スズメバチを蒸し殺すことができる。また蜂には、群飛飛翔と呼ばれる、群れを成して飛翔する行動をする際に、上昇移動や下

降移動、方向転換をする動きも観察される。本研究では、蜂球形成行動と、群飛飛翔を表現することを目的とする。boid アルゴリズムとそれに追加するルールを用いてニホンミツバチの行動が表現ができているかの実験を行い、提案手法の有用性を確認した。

1.2 本論文の構成

本論文は全5章で構成する。2章でミツバチの特徴について述べ、3章では提案手法について述べる。4章で、提案手法により表現したニホンミツバチの行動の評価と考察を行う。最後に5章で、まとめと今後の展望について述べる。

第 2 章

ミツバチの特徴

2.1 種類と防衛行動

ミツバチにも、様々な種類が存在する。日本にいるミツバチの中で特に多く生息している種は、セイヨウミツバチと呼ばれる種である。日本にはニホンミツバチという種が昔から生息していたが、セイヨウミツバチが日本で広まったことで、数が減っていった [11]。その理由として、セイヨウミツバチの方が養蜂に優れていることと、セイヨウミツバチがニホンミツバチに対して盗蜂を行うことが挙げられる [12]。養蜂とは蜂を飼育することであり、盗蜂とは蜜を盗むことである。盗蜂によりニホンミツバチの群れが弱体化して再起不能に陥る例は少なくない。このように、セイヨウミツバチはニホンミツバチに対して強い立場にある。しかし、セイヨウミツバチには戦うとほぼ敗北することになる、恐ろしい天敵がいる。それはスズメバチである。セイヨウミツバチだけでなく、ミツバチの種全体にとっても天敵であり、毎年多くのミツバチがスズメバチの犠牲となっている。スズメバチはミツバチを餌として好む。ミツバチを捕まえ、頭、翅、肢、腹部を切り落とし、余った部分を肉団子状にして巣に持ち帰り、幼虫の餌にする。これにより、一日に多くのミツバチが犠牲となる。20000 匹から 30000 匹を擁するミツバチの群れが、20 匹から 30 匹のオオスズメバチの集団攻撃により、3 時間ほどでほぼ全滅することもある [13]。セイヨウミツバチは、オオスズメバチなどの自分より一回り大きい外敵にも単独で戦いを挑むため敗北してしまうのである。

ニホンミツバチはスズメバチに対しては強力な防衛行動を行う。セイヨウミツバチとは違い、ニホンミツバチは外敵に対して単独では戦いを挑まない。複数のニホンミツバチが協力して戦いを挑む。ニホンミツバチの行動には、蜂球形成行動というものがある [14][15][16]。オオスズメバチがニホンミツバチを襲う際に、数十匹から数百匹のニホンミツバチがオオスズメバチに向かって取り囲むように突進して行き、球体に近い形を作って閉じ込める。この球体が蜂球である。蜂球を形成するニホンミツバチから発せられる熱により、オオスズメバチを蒸し殺すことができる。蜂球形成行動は、オオスズメバチのみでなく、キイロスズメバチといった蜂に対しても行う。オオスズメバチに対しては巣の中で蜂球を形成するが、キイロスズメバチに対しては巣の外で蜂球を形成することが多い。図 2.1 は実際の蜂球であり、蜂球内にはキイロスズメバチが存在している。ニホンミツバチはオオスズメバチを発見すると、巣の中で待ち構え、侵入してきたら蜂球形成行動を採る。キイロスズメバチは、餌を集め終わって巣に戻ってくるニホンミツバチを、巣の前で待ち構えて捕まえるが、逆に巣の門番のニホンミツバチに捕まると蜂球形成行動を受ける。本研究では、巣の中にいるオオスズメバチに対する蜂球形成行動を表現する。



図 2.1: 蜂球 (Takahashi による画像)

2.2 飛翔

昆虫の飛翔には様々な種類があるが、本研究では群飛飛翔について取り上げる。多くの昆虫が群れを成して飛ぶことを群飛飛翔または群飛と呼ぶが、群飛飛翔の様子は昆虫によって様々である。蜂の群飛飛翔は斑点状に分布する。また、群れの頂上から降下する途中に、側方へと進むことで周辺に向かったり、群れの中心を通過して再び上昇する動きをすることがある [17]。図 2.2 はミツバチの群飛飛翔を示したものである。



図 2.2: ミツバチの群飛飛翔 (「花を増やそう!みつばち百花」より引用)

<http://bee-happy.seesaa.net/>

蜂にはもう一つ、飛翔の際の特徴がある。蜂は飛翔の際、目標に向かって直線的に進むことができる。これが可能なのは、蜂は羽ばたき振動数が高く、一度の羽ばたきに対して移動量が少なく、体の上下動が小さいためである [18]。これが蝶などの昆虫の場合、羽ばたき振動数が低く、移動量が大きいため、体の上下動が大きくなる [19]。図 2.3 は蝶と蜂の目的地への飛翔を図示したものである。

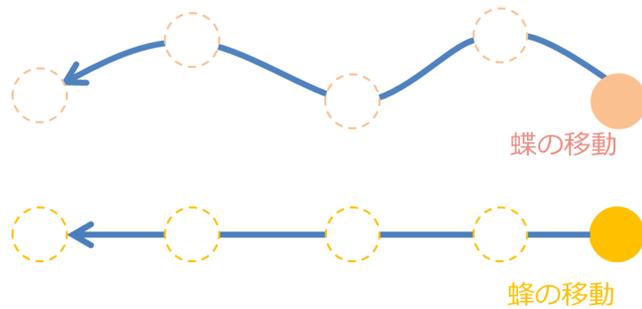


図 2.3: 蝶と蜂の目的地への飛翔

ただし、蜂が直線的に進む飛翔距離は長くなく、頻繁に方向を変える様子も見られる。

目標に向かって直線的に進むことができる、頻繁に方向を変えるとといった特徴はトンボの飛翔時の特徴に似ている。その一方で、周波数や翅の動かし方は異なっている。トンボの場合、周波数は低い。ただし、翅を互い違いに動かしているのので、体の上下動が小さく、直線的に進むことができる [7]。そのため 3.1 節では、トンボが飛翔する際の制御法を参考にし、蜂用に改良した。

第 3 章

ニホンミツバチの群れの動きを 表現する手法

この章では、ニホンミツバチの群れの動きを表現するために用いた、boid アルゴリズムと boid アルゴリズムに追加したルール、変更したルールについて述べる。複数のモデルに、boid アルゴリズムと追加・変更したルールを適用し、ニホンミツバチの群れの動きを表現する。

3.1 モデルの基本的な制御方法

この節では、利用するモデルが3次元空間内でどのような移動をするかについて述べる。3次元空間全体を表す座標系をグローバル座標系と呼び、モデルはグローバル座標系の中に配置している。モデルの位置は、3次元空間の中心を原点とした、右手座標系による x, y, z 軸上の値で表す。 y 軸は高さ、 xz 平面は水平である。高さを表す y 軸において、上方向は $+y$ 方向を指す。奥行きを表す z 軸において、手前方向は $+z$ 方向を指す。幅を表す x 軸において、右方向は $+x$ 方向を指す。グローバル座標系に対して、モデルの重心を原点とした座標系をローカル座標系と呼び、モデルにとって前方向は $-z$ 方向を指す。モデルにとって上方向は $+y$ 方向、進行方向に対して右方向は $+x$ 方向を指す。モデルのローカル座標系の z 軸に平行であり、 $-z$ 方向を指すベクトルを、モデルの方向ベクトルと呼ぶ。この方向ベク

トルは単位ベクトルであり、グローバル座標系で表す。モデルは基本的に、方向ベクトルの向きに向かって真っ直ぐに進む。基本となる移動量と変化量の値を足した値に、モデルの方向ベクトルを掛けて求めたベクトルがモデルの速度ベクトルとなる。変化量の値は3.2.3項で述べる分離ルールによって、値が変化する。ただし後に述べる、結合、分離、追加するルールにより、モデルの方向ベクトルの向きとは異なる方向に進むこともある。モデルには高さの位置制限も設け、表現する群れの動きに応じて、モデルが移動できる高さの範囲を変更した。範囲外にモデルが出そうになったら、範囲内に留まるようにした。

モデルは上下左右に向きを変える事が可能である。モデルのローカル座標系の y 軸を回転軸として、左右に回転運動をすることをヨーイングという。また、モデルのローカル座標系の x 軸を回転軸として、上下に回転運動をすることをピッチングという。飛翔行動時はピッチ角に制限を設けた。ピッチ角とは、モデルのローカル座標系の z 軸とグローバル座標系の xz 平面とのなす角である。これは、過度なピッチングによる、不自然な飛翔を防ぐためである。本研究では、 $\pm 10^\circ$ とした。また、ヨーイングによる横滑りを防ぐために、バンク角にも制限を設けた。バンク角とは、モデルのローカル座標系の x 軸とグローバル座標系の xz 平面とのなす角である。バンク角に制限を設けることで、モデルが必要以上に傾くことを防ぐことができる。本研究では、 $\pm 25^\circ$ とした。ピッチ角とバンク角の値には、トンボモデルの飛翔を表現する制御法で使われる値を参考にした。これは2.2節で述べたように、トンボの飛翔時の特徴が蜂の飛翔時の特徴に似ていたためである。なお、ヘディング角には制限は設けなかった。ヘディング角とは、モデルのローカル座標系の z 軸とグローバル座標系の yz 平面とのなす角である。図3.1は、モデルのローカル座標系と各軸を回転軸とした回転運動を図示したものである。

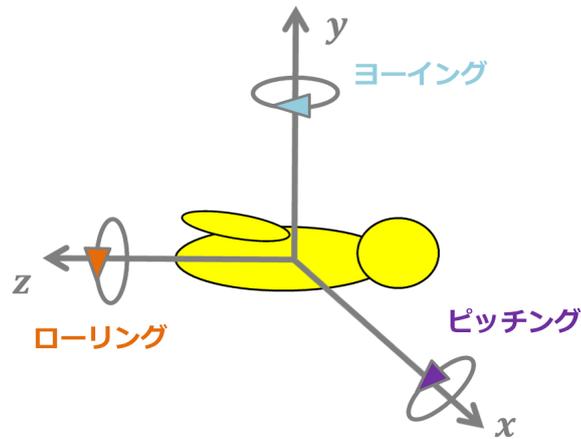


図 3.1: モデルのローカル座標系と回転運動

3.2 boid アルゴリズム

この節では、複数のモデルが群れとして動く様子を表現するための手法について述べる。本研究では、生物の群れの動きを表現する手法として、boid アルゴリズムを用いる。boid アルゴリズムは、1987年に Reynolds が考案したアルゴリズムである。結合、整列、分離という3つのルールを組み合わせることで、群れとしての動きを表現できる。

図 3.2 は boid アルゴリズムを適用する前のもので、矢印の形をしたモデルが不規則に移動している。矢印の向きは、モデルの方向ベクトルを表している。図 3.3 は boid アルゴリズムを適用したもので、モデルがまとまって、仲間と向きを合わせつつ移動している。

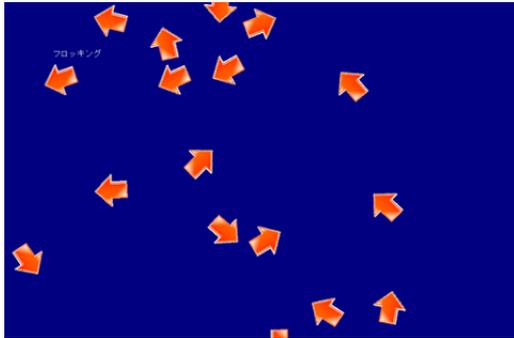


図 3.2: boid アルゴリズム適用前

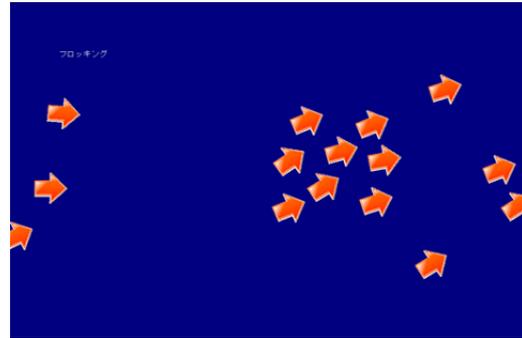


図 3.3: boid アルゴリズム適用後

本研究では、3.2.1 項～3.2.3 項による計算方法により、直線的に移動するニホンミツバチの群れの動きを表現する。

各ルールについて述べる前に、計算対象となるモデルと、ルール内で計算に使う複数の要素について述べる。図 3.4 で示す、仲間の影響を受ける範囲内のモデルが計算対象となる。

r : 仲間の影響を受ける距離

\mathbf{X}_i : 対象のモデルの位置

\mathbf{X}_j : 仲間の位置

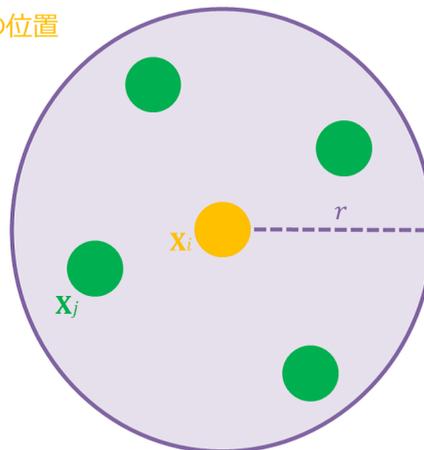


図 3.4: 仲間の影響を受ける範囲内のモデル

すなわち、各モデルが $|\mathbf{X}_j - \mathbf{X}_i| \leq r$ を満たしている必要がある。ここで、 i と j は各モデルの個体番号を表わし、 $i \neq j$ である。紫色の点線が仲間の影響を受け

る距離 r 、紫色の実線で囲んだ円が仲間の影響を受ける範囲を示している。黄色の点が計算対象のモデルの位置ベクトル \mathbf{X}_i 、緑色の点が仲間の位置ベクトル \mathbf{X}_j を示している。

計算対象のモデルの進行方向に対して、右方向を向いている単位ベクトル \mathbf{L}_i は以下の式 (3.1) で計算する。

$$\mathbf{L}_i = \mathbf{M}_i \times \mathbf{H}_i \quad (3.1)$$

\mathbf{L}_i は、計算対象のモデルのローカル座標系の x 軸に平行である。計算対象のモデルの方向ベクトルを \mathbf{M}_i とする。計算対象のモデルにおけるローカル座標系の y 軸に平行であり、進行方向に対して上方向を向いた単位ベクトルを \mathbf{H}_i とする。 \mathbf{H}_i は、グローバル座標系で表す。

3.2.1 結合ルール

結合ルールとは、計算対象のモデルを、範囲内にいる仲間の中心位置に向かって誘導するルールである。範囲内にいる仲間の位置ベクトルを足し、その和を範囲内にいる仲間の数で割ることで、中心位置ベクトルを求めることができる。図 3.5 は、計算対象のモデルを中心位置に向かって誘導する様子を図示したものである。結合ルールを行うことにより、複数のモデルをまとめることができる。

r : 仲間の影響を受ける距離
 \mathbf{U}_i : 仲間の中心位置
 \mathbf{X}_i : 対象のモデルの位置
 \mathbf{X}_j : 仲間の位置

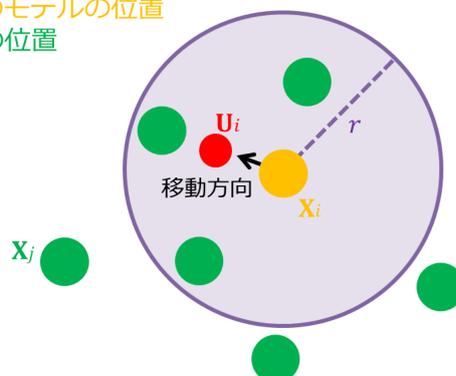


図 3.5: 結合

紫色の実線で囲んだ円が仲間の影響を受ける範囲、紫色の点線が仲間の影響を受ける距離 r 、黒色の矢印が移動方向を示している。黄色の点が計算対象のモデルの位置ベクトル \mathbf{X}_i 、緑色の点が仲間の位置ベクトル \mathbf{X}_j 、赤色の点が仲間の中心位置ベクトル \mathbf{U}_i を示している。

結合ルールを適用した場合は、計算対象のモデルのローカル座標系の x 軸に平行な移動と、ローカル座標系の y 軸に平行な移動を同時に行う。これらの平行移動は、仲間の中心位置ベクトル \mathbf{U}_i に近づくようにする。仲間の中心位置ベクトル \mathbf{U}_i を、次の式 (3.2) で示す。

$$\mathbf{U}_i = \frac{\sum_{j \in (A_i)} \mathbf{X}_j}{n(A_i)} \quad A_i = \{ j \mid |\mathbf{X}_i - \mathbf{X}_j| \leq r \} \quad (3.2)$$

$n(A_i)$ は A_i の集合の要素数とする。

計算対象のモデルの位置ベクトルから仲間の中心位置ベクトルへの正規化した方向ベクトル \mathbf{B}'_i を、次の式 (3.3) で示す。

$$\mathbf{B}'_i = \frac{\mathbf{U}_i - \mathbf{X}_i}{|\mathbf{U}_i - \mathbf{X}_i|} \quad (3.3)$$

グローバル座標系の xz 平面において、 \mathbf{U}_i が、計算対象のモデルの方向ベクトル \mathbf{M}_i の右側または左側のどちらに存在しているかを調べ、計算対象のモデルのロー

カル座標系の x 軸に平行な移動が、右方向または左方向のどちらに向かうかを定めるために必要な値 K_i を以下の式 (3.4) で計算する。

$$K_i = \begin{cases} 1 & (\mathbf{L}_i \cdot \mathbf{B}'_i > 0) \\ -1 & (\mathbf{L}_i \cdot \mathbf{B}'_i < 0) \\ 0 & (\mathbf{L}_i \cdot \mathbf{B}'_i = 0) \end{cases} \quad (3.4)$$

計算対象のモデルのローカル座標系の x 軸に平行であり、進行方向に対して右向きのベクトル \mathbf{L}_i に K_i を掛けることで、ローカル座標系の x 軸に平行な移動のための単位ベクトルが求まる。 \mathbf{U}_i の位置に近づくために、ローカル座標系の x 軸に平行な移動のための単位ベクトルの向きが、 $K_i = 1$ の場合は右方向、 $K_i = -1$ の場合は左方向を向くようになる。

グローバル座標系の yz 平面において、 \mathbf{U}_i が、 \mathbf{M}_i の上側または下側のどちらに存在しているかを調べ、計算対象のモデルのローカル座標系の y 軸に平行な移動が、上方向または下方向のどちらに向かうかを定めるのに必要な値 J_i を以下の式 (3.5) で計算する。

$$J_i = \begin{cases} -1 & (\mathbf{H}_i \cdot \mathbf{B}'_i > 0) \\ 1 & (\mathbf{H}_i \cdot \mathbf{B}'_i < 0) \\ 0 & (\mathbf{H}_i \cdot \mathbf{B}'_i = 0) \end{cases} \quad (3.5)$$

計算対象のモデルのローカル座標系の y 軸に平行な、進行方向に対して上向きのベクトル \mathbf{H}_i に J_i を掛けることで、ローカル座標系の y 軸に平行な移動のための単位ベクトルが求まる。 \mathbf{U}_i の位置に近づくために、ローカル座標系の y 軸に平行な移動のための単位ベクトルの向きが、 $J_i = 1$ の場合は上方向、 $J_i = -1$ の場合は下方向に向くようになる。

ローカル座標系の x 軸に平行な移動のための単位ベクトルと、ローカル座標系の y 軸に平行な移動のための単位ベクトルを求めた後、それらの単位ベクトルに平行移動量の大きさを掛ける。結合ルールによる平行移動量の大きさは c_i とし、式 (3.6) で示す。その式の計算に必要となる n は、平行移動量の大きさとして定めた

値とする。

$$c_i = \frac{|\mathbf{U}_i|}{n} \quad (3.6)$$

以上の要素をまとめた、結合ルールによる移動のための速度ベクトル \mathbf{C}_i を、式 (3.7) で示す。

$$\mathbf{C}_i = K_i c_i \mathbf{L}_i + J_i c_i \mathbf{H}_i \quad (3.7)$$

3.2.2 整列ルール

整列ルールとは、計算対象のモデルの向きを、範囲内にいる仲間の平均的な向きに合わせるように誘導するルールである。範囲内にいる仲間の方向ベクトルを足し、その和を範囲内にいる仲間の数で割る。この計算により、仲間の平均的な向きのベクトルを求める。図 3.6 は、計算対象のモデルを平均的な向きを向くように誘導する様子を図示したものである。

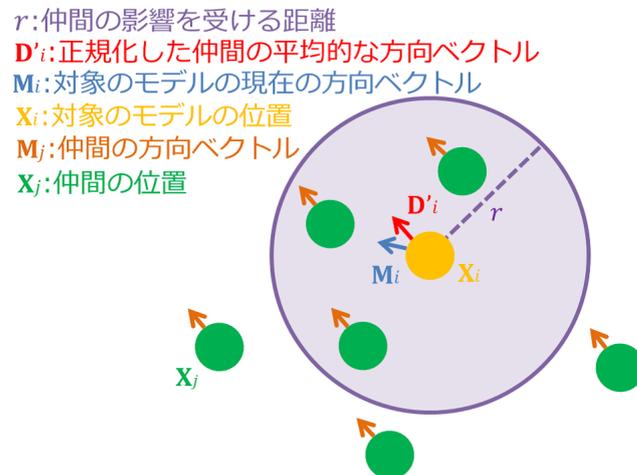


図 3.6: 整列

紫色の点線が仲間の影響を受ける距離 r 、紫色の実線で囲んだ円が仲間の影響を受ける範囲を示している。黄色の点が計算対象のモデルの位置ベクトル \mathbf{X}_i 、緑色

の点が仲間の位置ベクトル \mathbf{X}_j を示している。青色の矢印が計算対象のモデルの方向ベクトル \mathbf{M}_i 、赤色の矢印が正規化した仲間の平均的な方向ベクトル \mathbf{D}'_i 、茶色の矢印が各仲間の方向ベクトル \mathbf{M}_j を示している。 \mathbf{M}_j は単位ベクトルであり、グローバル座標系で表す。

仲間の平均的な方向ベクトル \mathbf{D}_i と、正規化した仲間の平均的な方向ベクトル \mathbf{D}'_i を、式 (3.8) と式 (3.9) で示す。

$$\mathbf{D}_i = \frac{\sum_{j \in (A_i)} \mathbf{M}_j}{n(A_i)} \quad A_i = \{ j \mid |\mathbf{X}_i - \mathbf{X}_j| \leq r \} \quad (3.8)$$

$$\mathbf{D}'_i = \frac{\mathbf{D}_i}{|\mathbf{D}_i|} \quad (3.9)$$

整列ルールを適用した場合、計算対象のモデルは平均的な向きを向くように方向ベクトルを更新する。計算対象のモデルのローカル座標系の x 軸と y 軸を回転軸とした回転運動を行うことで、方向ベクトルを更新する。回転する方向は、式 (3.10) と式 (3.11) で求めることができる。グローバル座標系の xz 平面において、 \mathbf{D}_i が、 \mathbf{M}_i の右側または左側のどちらに存在しているかを調べ、計算対象のモデルのローカル座標系の y 軸を回転軸とした回転運動が、左右のどちらに回転するかを決めるために必要な値 Z_i を以下の式 (3.10) で計算する。

$$Z_i = \begin{cases} -1 & (\mathbf{L}_i \cdot \mathbf{D}'_i > 0) \\ 1 & (\mathbf{L}_i \cdot \mathbf{D}'_i < 0) \\ 0 & (\mathbf{L}_i \cdot \mathbf{D}'_i = 0) \end{cases} \quad (3.10)$$

$Z_i = -1$ の場合は右方向、 $Z_i = 1$ の場合は左方向に回転運動をするようになる。 $Z_i = 0$ の場合は回転運動は行わない。

グローバル座標系の yz 平面において、 \mathbf{D}_i が、 \mathbf{M}_i の上側または下側のどちらに存在しているかを調べ、計算対象のモデルのローカル座標系の x 軸を回転軸とした回転運動が、上下のどちらに回転するかを決めるために必要な値 Y_i を以下の式 (3.11) で計算する。

$$Y_i = \begin{cases} 1 & (\mathbf{H}_i \cdot \mathbf{D}'_i > 0) \\ -1 & (\mathbf{H}_i \cdot \mathbf{D}'_i < 0) \\ 0 & (\mathbf{H}_i \cdot \mathbf{D}'_i = 0) \end{cases} \quad (3.11)$$

$Y_i = 1$ の場合は上方向、 $Y_i = -1$ の場合は下方向に回転運動をするようになる。
 $Y_i = 0$ の場合は回転運動は行わない。整列ルールによる回転運動は、以下の手順で計算する。計算対象のモデルがローカル座標系の y 軸を回転軸とした左右への回転を行うヨーイングと、ローカル座標系の x 軸を回転軸とした上下への回転を行うピッチングのための式 (3.12) を示す。

$$\mathbf{M}'_i = \begin{pmatrix} \cos Z_i a_i & \sin Y_i a_i \sin Z_i a_i & \cos Y_i a_i \sin Z_i a_i \\ 0 & \cos Y_i a_i & -\sin Y_i a_i \\ -\sin Z_i a_i & \sin Y_i a_i \cos Z_i a_i & \cos Y_i a_i \cos Z_i a_i \end{pmatrix} \mathbf{M}_i \quad (3.12)$$

計算対象モデルの新たな方向ベクトルを \mathbf{M}'_i とする。 a_i は整列ルールによる回転角度で、式 (3.13) で計算する。

$$a_i = T(\arccos \mathbf{M}_i \cdot \mathbf{D}'_i) \quad (3.13)$$

T は定数である。

3.2.3 分離ルール

分離ルールとは、計算対象モデルを、範囲内にいる仲間と衝突しないように誘導するルールである。仲間との距離が近い場合、計算対象のモデルは仲間から離れる。図 3.7 は、計算対象のモデルを仲間と衝突しないように誘導する様子を図示したものである。

r : 仲間の影響を受ける距離
 l : 仲間を近づけない距離
 \mathbf{X}_i : 対象のモデルの位置
 \mathbf{X}_j : 仲間の位置

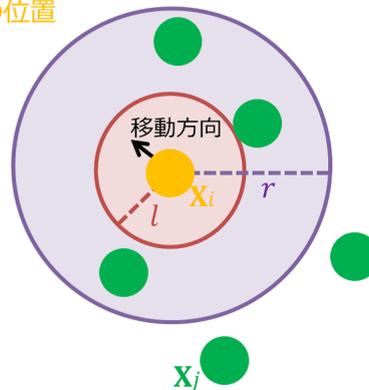


図 3.7: 分離

紫色の点線が仲間の影響を受ける距離 r 、紫色の実線で囲んだ円が仲間の影響を受ける範囲を示している。薄い赤色の点線が仲間を近づけない距離 l 、薄い赤色の実線で囲んだ円が仲間を近づけない範囲を示している。黄色の点が計算対象のモデルの位置ベクトル \mathbf{X}_i 、緑色の点が仲間の位置ベクトル \mathbf{X}_j 、黒色の矢印が計算対象のモデルの移動方向を示している。

分離ルールを適用した場合は、計算対象のモデルのローカル座標系の x 軸に平行な移動と、ローカル座標系の y 軸に平行な移動を同時に行う。これらの平行移動は、各仲間の位置ベクトル \mathbf{X}_j から離れるようにする。計算対象のモデルの位置ベクトルから各仲間の位置ベクトルへの方向ベクトル \mathbf{I}_i を、式 (3.14) で示す。

$$\mathbf{I}_i = \mathbf{X}_j - \mathbf{X}_i \quad (3.14)$$

計算対象のモデルの位置ベクトルから各仲間の位置ベクトルへの距離 (長さ) は $|\mathbf{I}_i|$ である。グローバル座標系の xz 平面において、 \mathbf{I}_i が、 \mathbf{M}_i の右側または左側のどちらに存在しているかを調べ、計算対象のモデルのローカル座標系の x 軸に平行な移動が、右方向または左方向のどちらに向かうかを定めるために必要な値 P_i を以下の式 (3.15) で計算する。

$$P_i = \begin{cases} -1 & (\mathbf{L}_i \cdot \mathbf{I}_i > 0) \\ 1 & (\mathbf{L}_i \cdot \mathbf{I}_i < 0) \\ 0 & (\mathbf{L}_i \cdot \mathbf{I}_i = 0) \end{cases} \quad (3.15)$$

計算対象のモデルのローカル座標系の x 軸に平行な、進行方向に対して右向きのベクトル \mathbf{L}_i に P_i を掛けることで、ローカル座標系の x 軸に平行な移動のための単位ベクトルが求まる。 \mathbf{I}_i の位置から離れるために、ローカル座標系の x 軸に平行な移動のための単位ベクトルの向きが、 $P_i = -1$ の場合は左方向、 $P_i = 1$ の場合は右方向を向くようになる。

グローバル座標系の yz 平面において、 \mathbf{I}_i が、 \mathbf{M}_i に平行な直線の上側または下側のどちらに存在しているかを調べ、計算対象のモデルのローカル座標系の y 軸に平行な移動が、上方向または下方向のどちらに向かうかを定めるのに必要な値 Q_i を以下の式 (3.16) で計算する。

$$Q_i = \begin{cases} 1 & (\mathbf{H}_i \cdot \mathbf{I}_i > 0) \\ -1 & (\mathbf{H}_i \cdot \mathbf{I}_i < 0) \\ 0 & (\mathbf{H}_i \cdot \mathbf{I}_i = 0) \end{cases} \quad (3.16)$$

計算対象のモデルのローカル座標系の y 軸に平行な、進行方向に対して上向きのベクトル \mathbf{H}_i に Q_i を掛けることで、ローカル座標系の y 軸に平行な移動のための単位ベクトルが求まる。 \mathbf{I}_i の位置から離れるために、ローカル座標系の y 軸に平行な移動のための単位ベクトルの向きが、 $Q_i = 1$ の場合は上方向、 $Q_i = -1$ の場合は下方向を向くようになる。

ローカル座標系の x 軸に平行な移動のための単位ベクトルと、ローカル座標系の y 軸に平行な移動のための単位ベクトルを求めた後、それらの単位ベクトルに平行移動量の大きさを掛ける。分離ルールによる平行移動量の大きさを s_i とし、式 (3.17) で示す。その式の計算に必要となる u は、平行移動量の大きさとして定めた値とする。

$$s_i = \frac{u}{|\mathbf{I}_i|} \quad (3.17)$$

以上の要素をまとめた、分離ルールによる平行移動のための速度ベクトルを、式 (3.18) で示す。

$$\mathbf{S}_i = P_i s_i \mathbf{L}_i + Q_i s_i \mathbf{H}_i \quad (|\mathbf{X}_j - \mathbf{X}_i| \leq l) \quad (3.18)$$

さらに分離ルールでは、速度調整も行う。3.1 節で述べたように、計算対象のモデルは boid アルゴリズムを適用しない場合、自身の方向ベクトルの向きに向かって真っ直ぐに進む。この動きを実現するための、計算対象のモデルが持つ速度ベクトル \mathbf{W}_i を式 (3.19) で示す。

$$\mathbf{W}_i = (b_i + m_i) \mathbf{M}_i \quad (3.19)$$

b_i は基本となる移動量の値である。 m_i は変化量の値であり、式 (3.20) で示す。

$$m_i = \begin{cases} \frac{|l_i|}{e} & (|\mathbf{X}_j - \mathbf{X}_i| \leq l) \\ m_0 & (|\mathbf{X}_j - \mathbf{X}_i| > l) \end{cases} \quad (3.20)$$

式 (3.20) の計算に必要となる、 e は平行移動量の大きさとして定めた値である。 e は定数である。これにより、計算対象のモデルが仲間に近づくほど、移動速度が遅くなる。この m の値は、計算対象のモデルが、仲間を近づけない距離よりも離れた距離にいる場合、初期値 m_0 に戻す。

以上が、boid アルゴリズムの提案手法における基本ルールの説明である。

3.2.4 boid アルゴリズムによる位置の更新

結合、分離ルールを適用し、計算対象のモデルの位置を更新するための式 (3.21) を示す。

$$\mathbf{X}'_i = \mathbf{X}_i + (\mathbf{W}_i + \mathbf{C}_i + \mathbf{S}_i) \Delta t \quad (3.21)$$

\mathbf{X}'_i は、計算対象のモデルの新しい位置ベクトルである。 Δt は単位時間を表す。計算対象のモデルの位置ベクトル \mathbf{X}_i に、計算対象のモデルが持つ速度ベクトル \mathbf{W}_i と、結合ルールによる移動のための速度ベクトル \mathbf{C}_i 、分離ルールによる移動のた

めの速度ベクトル S_i を足し合わせたものが、計算対象のモデルの新たな位置ベクトルとなる。その後、計算対象のモデルの方向ベクトルを、整列ルールによって求めた、計算対象のモデルの新たな方向ベクトルへと更新する。

boid アルゴリズムの基本ルールを複数のモデルに適用することで、直線的に移動する、ニホンミツバチの群れの動きが表現できる。しかし、boid アルゴリズムの基本ルールを用いるだけでは、蜂の群飛飛翔や蜂球を形成するための群れの動きは表現できない。そこで、より本物に近いニホンミツバチの動きを表現するために、第2章で述べたニホンミツバチの特徴を実現するためのルールを追加または変更を行う。

3.3 追加・変更するルール

この節では、群飛飛翔を行うニホンミツバチの群れと外敵に対して襲い掛かるニホンミツバチの群れを表現する手法を述べる。

3.3.1 群飛ルール

群飛飛翔を行うニホンミツバチの群れを表現するために追加するルールとして、群飛ルールを追加する。これは2.2節で述べた、蜂の群飛飛翔のためのルールであり、群れの頂上から降下する途中に、側方へと進むことで周辺に向かったり、群れの中心を通過して再び上昇する動きを表現する。

最初に、結合ルールで求めた、仲間の中心位置から、上昇移動の影響を受ける範囲を決める。図3.8は上昇移動の影響を受ける範囲を図示したものである。

r : 仲間の影響を受ける距離
 d : 上昇の影響を受ける距離
 X_i : 対象のモデルの位置
 X_j : 仲間の位置
 U_i : 仲間の中心位置

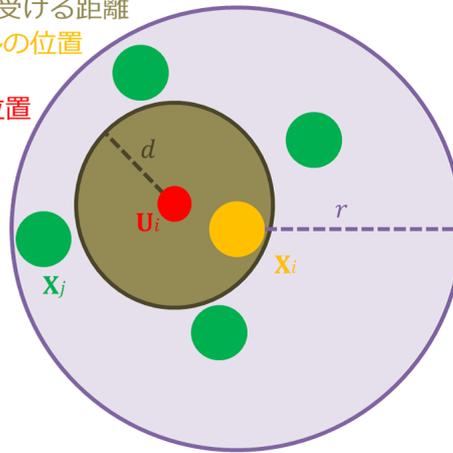


図 3.8: 上昇移動の影響を受ける範囲

ベージュ色の点線が上昇移動の影響を受ける距離 d 、ベージュ色の実線で囲んだ円が上昇移動の影響を受ける範囲を示している。紫色の点線が仲間の影響を受ける距離 r 、紫色の実線で囲んだ円が仲間の影響を受ける範囲を示している。黄色の点が計算対象のモデルの位置ベクトル X_i 、緑色の点が仲間の位置ベクトル X_j 、赤色の点が仲間の中心位置ベクトル U_i を示している。計算対象のモデルは、 $|U_i - X_i| \leq d$ を満たすときに上昇移動を行う。上昇移動を行っているモデルを、上昇状態のモデルとする。

計算対象のモデルのグローバル座標系での y 座標の位置が、仲間の影響を受ける範囲内にいる全ての仲間の位置の値より高い場合、計算対象のモデルは下降移動を行う。下降移動は、計算対象のモデルのグローバル座標系での y 座標の位置が仲間の中心位置より低くなるまで行う。また、降下する途中、右方向か左方向に向かうように誘導する。下降移動を行っているモデルを、下降状態のモデルとする。図 3.9 は、横方向から見た、群れの中心を通過して上昇する計算対象のモデルの様子を図示したものである。図 3.10 は、上方向から見た、群れの頂上から降下して周辺へと向かう計算対象のモデルの様子を図示したものである。

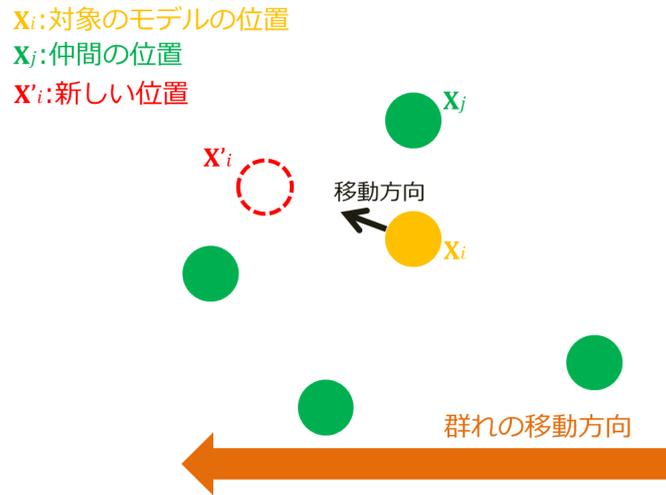


図 3.9: 群れの中心を通過して上昇する様子

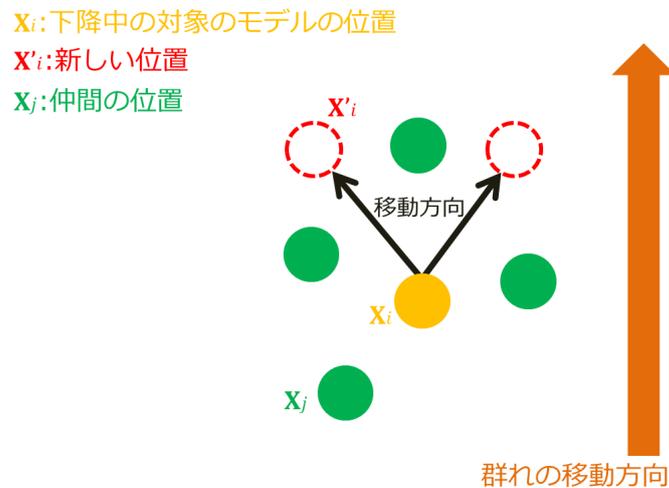


図 3.10: 群れの頂上から下降して周辺へと向かう様子

黄色の点が計算対象のモデルの位置ベクトル X_i 、緑色の点が仲間の位置ベクトル X_j 、赤色の点が計算対象のモデルの新しい位置ベクトル X'_i を示している。黒色の矢印が計算対象のモデルの移動方向、茶色の矢印が群れの移動方向を示している。

群飛ルールよる、計算対象のモデルの移動は、計算対象のモデルの方向ベクトルの向きを条件に合わせて、ピッチングとヨーイングにより調整をすることで実現できる。ピッチングのための式 (3.22) を示す。

$$\mathbf{M}'_i = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos O_i f_i & -\sin O_i f_i \\ 0 & \sin O_i f_i & \cos O_i f_i \end{pmatrix} \mathbf{M}_i \quad (3.22)$$

f_i は基準となる回転角度であり、定数である。 O_i は係数である。 O_i の値は条件によって異なり、以下の式 (3.23) ように分かれる。

$$O_i = \begin{cases} 1 & (|\mathbf{U}_i - \mathbf{X}_i| \leq d \text{ かつ } \mathbf{X}_{i,y} \leq \mathbf{X}''_{j,y}) \\ -1 & (\mathbf{X}_{i,y} > \mathbf{X}''_{j,y}) \\ 0 & (|\mathbf{U}_i - \mathbf{X}_i| > d \text{ かつ } \mathbf{X}_{i,y} < \mathbf{X}''_{j,y}) \end{cases} \quad (3.23)$$

計算対象のモデルの位置ベクトルの x 成分を $X''_{i,x}$ 、 y 成分を $X''_{i,y}$ 、 z 成分を $X''_{i,z}$ とする。一番高い位置にいる仲間の位置ベクトルの x 成分を $X''_{j,x}$ 、 y 成分を $X''_{j,y}$ 、 z 成分を $X''_{j,z}$ とする。 $O_i = 0$ の場合、boid アルゴリズムによる移動のみを行う。この移動を行っているモデルを、通常状態のモデルとする。 $O = -1$ で下方向へ移動する際は、右方向か左方向かのどちらかにヨーイングを行い、側方に向かうようにする。ヨーイングのための式 (3.24) を示す。

$$\mathbf{M}'_i = \begin{pmatrix} \cos n_i & 0 & \sin n_i \\ 0 & 1 & 0 \\ -\sin n_i & 0 & \cos n_i \end{pmatrix} \mathbf{M}_i \quad (3.24)$$

計算対象のモデルの新たな方向ベクトルを \mathbf{M}'_i 、計算対象のモデルの現在の方向ベクトルを \mathbf{M}_i とする。 n_i は回転角度である。計算対象のモデルが下降状態となったら、 20° または -20° の値を n_i として用いて、その角度分左右のどちらかに回転する。この回転運動は下降中に一度のみ行う。下降中の計算対象のモデルは、結合ルールと整列ルールの2つは適用しない。

boid アルゴリズムに群飛ルールを追加して適用することで、群飛飛翔を行うニホンミツバチの群れが表現できる。ただし、計算対象のモデルが上昇状態の時は、整列ルールによる回転運動の後に、上昇移動のための回転運動を行う。

3.3.2 オオスズメバチに襲いかかるためのルール

オオスズメバチに対して襲い掛かるニホンミツバチの群れの表現には、2つの段階がある。1つ目は、ニホンミツバチの群れがオオスズメバチに向かって移動する段階である。2つ目は、ニホンミツバチの群れがオオスズメバチに接触して取り囲み、蜂球を形成する段階である。これらの段階に入ったら、整列ルールは適用しない。オオスズメバチに向かって移動する段階では、ニホンミツバチの向いている方向はオオスズメバチの位置に集中し、蜂球を形成する段階では、ニホンミツバチの向きはバラバラになるためである。

ニホンミツバチの群れがオオスズメバチに向かって移動する段階と蜂球を形成する段階では、結合ルールを変更する。結合ルールは通常、範囲内にいる仲間の位置から中心位置を求めるが、この段階ではオオスズメバチのモデル(以下、外敵)の位置 E を中心位置とする。そのため、変更後の結合ルールでは、仲間の影響を受けなくなる。また、結合ルールではこれまで自身のローカル座標系の x 軸と y 軸に平行な移動をしてきたが、ニホンミツバチの群れがオオスズメバチに向かって移動する段階と蜂球を形成する段階の、結合ルールによる行動は、整列ルールと同じように、モデルの方向ベクトルを更新することによる、ピッチングとヨーイングを用いた行動に変わる。結合ルールによる回転運動は、計算対象のモデルが外敵の位置に向かうように方向ベクトルを更新する。これにより、外敵の周りを飛び回って取り囲むような回転運動をする移動ができるようになる。また、分離ルールも変更する。分離ルールでは通常、仲間との衝突を防ぐために、計算対象のモデルは自身のローカル座標系の x 軸と y 軸に平行な移動を、仲間のいる方向とは逆方向に行っていた。しかし、外敵に向かって移動する段階では、計算対象のモデルの方向ベクトルを仲間から遠ざけるように更新する。外敵に向かって移動する段階と蜂球を形成する段階で、結合と分離ルールによる移動を回転運動を用いたものに変更したのは、外敵に襲いかかる際の曲線的な動きや、蜂球を形成する段階のわらわらとしたニホンミツバチの動きを表現するには、自身の方向

ベクトルを更新していく回転運動を用いた移動が適していると判断したためである。計算対象のモデルが外敵に接触し、蜂球を形成する段階になったら、そのモデルの移動速度を遅くする。モデルが持つ速度ベクトル \mathbf{W}_i を構成する、基本となる移動量の値 b_i を小さくし、変化量の値 m_i は 0 にする。また、蜂球を形成する段階では、仲間を近づけない距離 l の値を小さくする。

まずはじめに、外敵に向かって移動する段階での、計算対象のモデルの方向ベクトルを更新する方法について述べる。結合ルールを適用した場合、計算対象のモデルは外敵の位置を向くように方向ベクトルを更新する。グローバル座標系の xz 平面において、外敵の位置ベクトルが、計算対象のモデルの方向ベクトル \mathbf{M}_i の右側または左側のどちらに存在しているかを調べ、計算対象のモデルのローカル座標系の y 軸を回転軸とした回転運動が、左右のどちらに回転するかを決めるために必要な値 K_i を以下の式 (3.25) で計算する。 K_i は、変更前の結合ルール内で、移動する方向を決める際に使用した値である。

$$\mathbf{K}_i = \begin{cases} -1 & (\mathbf{L}_i \cdot \mathbf{N}'_i > 0) \\ 1 & (\mathbf{L}_i \cdot \mathbf{N}'_i < 0) \\ 0 & (\mathbf{L}_i \cdot \mathbf{N}'_i = 0) \end{cases} \quad (3.25)$$

$K_i = -1$ の場合は右方向、 $K_i = 1$ の場合は左方向に回転運動をするようになる。 $K_i = 0$ の場合は回転運動は行わない。 \mathbf{N}_i は、計算対象のモデルの位置ベクトルから外敵の位置ベクトルへの方向ベクトルである。 \mathbf{N}'_i は、計算対象のモデルの位置ベクトルから外敵の位置ベクトルへの正規化した方向ベクトルである。 \mathbf{N}_i と \mathbf{N}'_i は以下の式 (3.26) と (3.27) で示す。

$$\mathbf{N}_i = \mathbf{E} - \mathbf{X}_i \quad (3.26)$$

$$\mathbf{N}'_i = \frac{\mathbf{E} - \mathbf{X}_i}{|\mathbf{E} - \mathbf{X}_i|} \quad (3.27)$$

グローバル座標系の yz 平面において、 \mathbf{E} が、 \mathbf{M}_i の上側または下側のどちらに存在しているかを調べ、計算対象のモデルのローカル座標系の x 軸を回転軸とした回転運動が、上下のどちらに回転するかを決めるために必要な値 J_i を以下の式

(3.28) で計算する。 J_i は、変更前の結合ルール内で、移動する方向を決める際に使用した値である。

$$J_i = \begin{cases} 1 & (\mathbf{H}_i \cdot \mathbf{N}'_i > 0) \\ -1 & (\mathbf{H}_i \cdot \mathbf{N}'_i < 0) \\ 0 & (\mathbf{H}_i \cdot \mathbf{N}'_i = 0) \end{cases} \quad (3.28)$$

\mathbf{H}_i は、計算対象のモデルのローカル座標系の y 軸に平行な、進行方向に対して上向きのベクトルである。 $J_i = 1$ の場合は上方向、 $J_i = -1$ の場合は下方向に回転運動をするようになる。 $J_i = 0$ の場合は回転運動は行わない。

外敵に向かって移動する段階での、結合ルールによる回転運動は、以下の手順で計算する。計算対象のモデルがローカル座標系の y 軸を回転軸とした左右への回転を行うヨーイングと、ローカル座標系の x 軸を回転軸とした上下への回転を行うピッチングのための式 (3.29) を示す。

$$\mathbf{M}'_i = \begin{pmatrix} \cos K_i o'_i & \sin J_i o_i \sin K_i o'_i & \cos J_i o_i \sin K_i o'_i \\ 0 & \cos J_i o_i & -\sin J_i o_i \\ -\sin K_i o'_i & \sin J_i o_i \cos K_i o'_i & \cos J_i o_i \cos K_i o'_i \end{pmatrix} \mathbf{M}_i \quad (3.29)$$

計算対象モデルの新たな方向ベクトルを \mathbf{M}'_i とする。外敵に向かって行動する段階での結合ルールによる回転角度は o_i と o'_i とする。 o_i と o'_i は定数である。外敵に向かって行動する段階では、ピッチングとヨーイングのための回転角度が異なる。この段階では、外敵に対して円を描くように取り囲んで飛翔するためのヨーイングの方が、必要となる回転角度が大きいためである。

蜂球を形成する段階での結合ルールによる回転角度は k_i とし、式 (3.30) で示す。

$$k_i = B(\arccos \mathbf{M}_i \cdot \mathbf{N}'_i) \quad (3.30)$$

B は定数である。蜂球を形成する段階での結合ルールによる回転運動は、式 (3.29) で用いた o_i と o'_i の代わりに k_i を用いる。それにより、計算対象のモデルの方向ベクトルを更新することができる。

分離ルールを適用した場合は、計算対象のモデルは自身の方向ベクトルを仲間から遠ざけるように更新する。これは、変更前の分離ルールと同様に、 $|\mathbf{X}_j - \mathbf{X}_i| \leq l$ を満たしている必要がある。グローバル座標系の xz 平面において、計算対象のモ

デルの位置ベクトルから各仲間の位置ベクトルへの方向ベクトル \mathbf{I}_i が、 \mathbf{M}_i の右側または左側のどちらに存在しているかを調べ、計算対象のモデルのローカル座標系の y 軸を回転軸とした回転運動が、左右のどちらに回転するかを決めるために必要な値 P_i を以下の式 (3.31) で計算する。 P_i は、変更前の分離ルール内で、移動する方向を決めるために使用した値である。

$$P_i = \begin{cases} 1 & (\mathbf{L}_i \cdot \mathbf{I}_i > 0) \\ -1 & (\mathbf{L}_i \cdot \mathbf{I}_i < 0) \\ 0 & (\mathbf{L}_i \cdot \mathbf{I}_i = 0) \end{cases} \quad (3.31)$$

$P_i = -1$ の場合は右方向、 $P_i = 1$ の場合は左方向に回転運動をするようになる。 $P_i = 0$ の場合は回転運動は行わない。

グローバル座標系の yz 平面において、 \mathbf{I}_i が、 \mathbf{M}_i の上側または下側のどちらに存在しているかを調べ、計算対象のモデルのローカル座標系の x 軸を回転軸とした回転運動が、上下のどちらに回転するかを決めるために必要な値 Q_i を以下の式 (3.32) で計算する。 Q_i は、自由に飛翔するミツバチの群れを表現する際に分離ルール内で、移動する方向を決めるために使用した値である。

$$Q_i = \begin{cases} -1 & (\mathbf{H}_i \cdot \mathbf{I}_i > 0) \\ 1 & (\mathbf{H}_i \cdot \mathbf{I}_i < 0) \\ 0 & (\mathbf{H}_i \cdot \mathbf{I}_i = 0) \end{cases} \quad (3.32)$$

$Q_i = -1$ の場合は下方向、 $Q_i = 1$ の場合は上方向に回転運動をするようになる。 $Q_i = 0$ の場合は回転運動は行わない。

外敵に向かって移動する段階での、分離ルールによる回転運動は、以下の手順で計算する。計算対象のモデルがローカル座標系の y 軸を回転軸とした左右への回転を行うヨーイングと、ローカル座標系の x 軸を回転軸とした上下への回転を行うピッチングのための式 (3.33) を示す。

$$\mathbf{M}'_i = \begin{pmatrix} \cos P_i \alpha'_i & \sin Q_i \alpha_i \sin P_i \alpha'_i & \cos Q_i \alpha_i \sin P_i \alpha'_i \\ 0 & \cos Q_i \alpha_i & -\sin Q_i \alpha_i \\ -\sin P_i \alpha'_i & \sin Q_i \alpha_i \cos P_i \alpha'_i & \cos Q_i \alpha_i \cos P_i \alpha'_i \end{pmatrix} \mathbf{M}_i \quad (3.33)$$

外敵に向かって移動する段階では、分離ルールによる回転角度は α_i と α'_i し、式 (3.34) と式 (3.35) で示す。

$$\alpha_i = \frac{w}{|\mathbf{I}_i|} \quad (|\mathbf{X}_j - \mathbf{X}_i| \leq l) \quad (3.34)$$

$$\alpha'_i = \frac{w'}{|\mathbf{I}_i|} \quad (|\mathbf{X}_j - \mathbf{X}_i| \leq l) \quad (3.35)$$

ヨーイングの際に用いる回転角度を α_i とする。ピッチングの際に用いる回転角度を α'_i とする。 α_i を求めるための基準となる回転角度を w とする。 α'_i を求めるための基準となる回転角度を w' とする。 w と w' は定数である。外敵に向かって移動する段階では、ピッチングとヨーイングそれぞれの基準となる回転角度が異なる。ピッチングとヨーイングの基準となる回転角度が異なるのは、この段階での結合ルールと同じように、ヨーイングの方が必要となる回転角度が大きいためである。

蜂球を形成する段階での分離ルールによる回転角度を β_i と β'_i とし、式 (3.36) と式 (3.37) で示す。

$$\beta_i = \frac{\gamma}{|\mathbf{I}_i|} \quad (|\mathbf{X}_j - \mathbf{X}_i| \leq l) \quad (3.36)$$

$$\beta'_i = \frac{\gamma'}{|\mathbf{I}_i|} \quad (|\mathbf{X}_j - \mathbf{X}_i| \leq \frac{l}{2}) \quad (3.37)$$

ヨーイングの際に用いる回転角度を β_i とする。ピッチングの際に用いる回転角度を β'_i とする。 β_i を求めるための基準となる回転角度を γ とする。 β'_i を求めるための基準となる回転角度を γ' とする。 γ と γ' は定数である。蜂球を形成する段階での分離ルールによる回転運動は、式 (3.33) で用いた α_i の代わりに β_i 、 α'_i の代わりに β'_i を用いることで、計算対象のモデルの方向ベクトルを更新することができる。ただしピッチングは、式 (3.37) で示した通り、計算対象のモデルと仲間との距離が非常に近い場合のみ行う。

ニホンミツバチの群れが外敵に接触して取り囲み、蜂球の形を綺麗に表現するために、引力ルールを追加する。引力ルールとは、計算対象のモデルが蜂球の範囲外に出ようとしたら、範囲内に留まるように誘導するルールである。図 3.11 は、

引カルールにより、蜂球の範囲内に留まるように移動する、計算対象のモデルの様子を図示したものである。

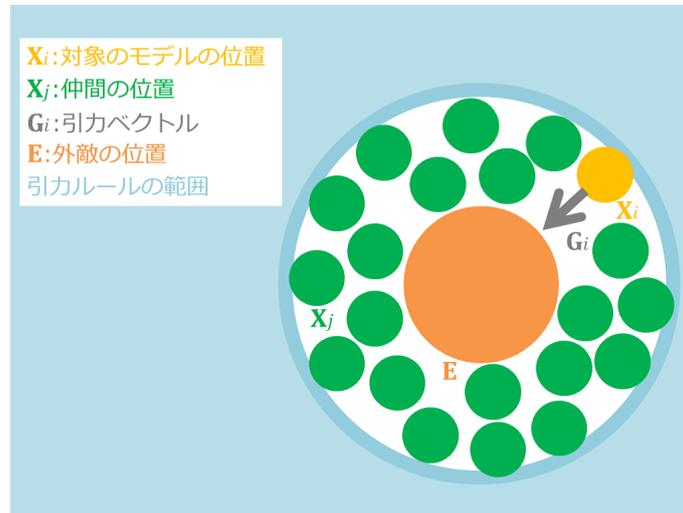


図 3.11: 引カ

黄色の円の中心が計算対象のモデルの位置ベクトル X_i 、緑色の円の中心が仲間の位置ベクトル X_j 、オレンジ色の円の中心が外敵の位置ベクトル E 、灰色の矢印が引カベクトル G_i を示している。水色の実線と、その実線で囲んだ円の外側が引カルールの影響を受ける範囲を示している。また、水色の実線で囲んだ円が蜂球の範囲を示している。

式 (3.38) は、引カルールの数式化したものである。

$$G_i = \begin{cases} 0 & (|N_i| < V) \\ N'_i Z & (|N_i| \geq V) \end{cases} \quad (3.38)$$

N_i は、計算対象のモデルから外敵への方向ベクトルであり、 N_i を正規化したベクトルを N'_i とする。 V は、引カルールの影響を受ける距離であり、定数である。また、 Z は定数である。

また、外敵と計算対象のモデルが重なりあわないようにするために、斥カルールを追加する。斥カルールとは、計算対象のモデルが斥カルールの影響を受ける範囲内に入ったら、範囲外に出るように誘導するルールである。図 3.12 は、斥カ

ルールにより、範囲内に入らないように移動する、計算対象のモデルの様子を図示したものである。

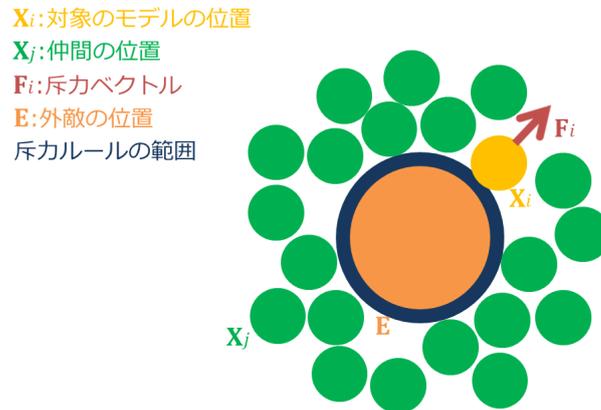


図 3.12: 斥力

黄色の円の中心が計算対象のモデルの位置ベクトル X_i 、緑色の円の中心が仲間の位置ベクトル X_j 、オレンジ色の円の中心が外敵の位置ベクトル E 、薄い赤色の矢印が斥力ベクトル F_i を示している。紺色の実線で囲んだ円が斥力ルールの影響を受ける範囲を示している。

式 (3.39) は、斥力ルールを数式化したものである。

$$F_i = \begin{cases} 0 & (|N_i| > R) \\ -N'_i Z & (|N_i| \leq R) \end{cases} \quad (3.39)$$

R は、斥力ルールの影響を受ける距離であり、定数である。

3.4 boid アルゴリズムと追加したルールによる位置の更新

この節では、外敵に向かって移動する段階と蜂球を形成する段階での位置の更新について述べる。外敵に向かって移動する段階での、計算対象のモデルの新しい

い位置 \mathbf{X}'_i は、式 (3.40) で計算できる。

$$\mathbf{X}'_i = \mathbf{X}_i + (\mathbf{W}_i)\Delta t \quad (3.40)$$

\mathbf{X}_i は、計算対象のモデルの位置ベクトルである。 \mathbf{W}_i は、計算対象のモデルが持つ速度ベクトルである。位置を更新した後に、結合ルールによる回転運動と分離ルールによる回転運動を行う。

蜂球を形成する段階での、計算対象のモデルの新しい位置は、式 (3.41) で計算できる。

$$\mathbf{X}'_i = \mathbf{X}_i + (\mathbf{W}_i + \mathbf{G}_i + \mathbf{F}_i)\Delta t \quad (3.41)$$

引力ルールで求めたベクトル \mathbf{G}_i と斥力ルールで求めたベクトル \mathbf{F}_i を、計算対象のモデルの位置ベクトル \mathbf{X}_i に足し合わせることで、計算対象のモデルの動きを表現できる。位置を更新した後に、結合ルールによる回転運動と分離ルールによる回転運動を行う。

第 4 章

評価

この章では、3章で述べたルールをモデルに適用し、それにより表現することができた群れの動きについて評価する。図 4.1 が使用したモデルを示したものである。



図 4.1: 使用したモデル

このモデルをニホンミツバチとして扱い、群飛飛翔を行うニホンミツバチの群れと、オオスズメバチに襲いかかるニホンミツバチの群れの動きを表現した。モデルに群飛ルールを適用すると、現在の行動の状態によって、自身の色が変わる。

また、シミュレーションを行うスペースは、周期境界条件になっている。実装したプログラムは、グラフィックス API の OpenGL[20] をベースとした 3次元グラフィックスツールキットである Fine Kernel Tool Kit System[21][22] を利用した。

4.1 群飛飛翔を行うニホンミツバチの群れ

群飛飛翔を行うニホンミツバチの群れの様子を表現するために、モデルは100体を使用した。図4.2～図4.6は、群飛飛翔を行うニホンミツバチの群れの様子を示したものである。各図において左側が上から見た図であり、右側が手前から見た図である。

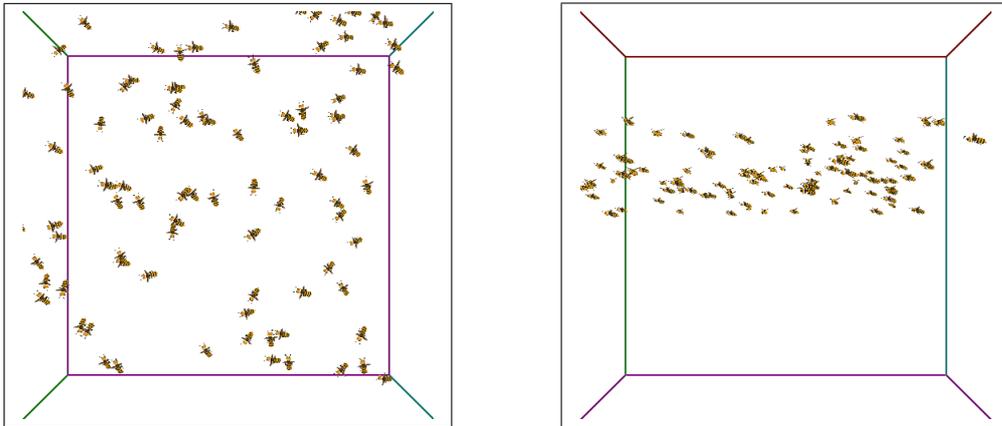


図4.2: 群飛飛翔を行うニホンミツバチの群れ (0秒)

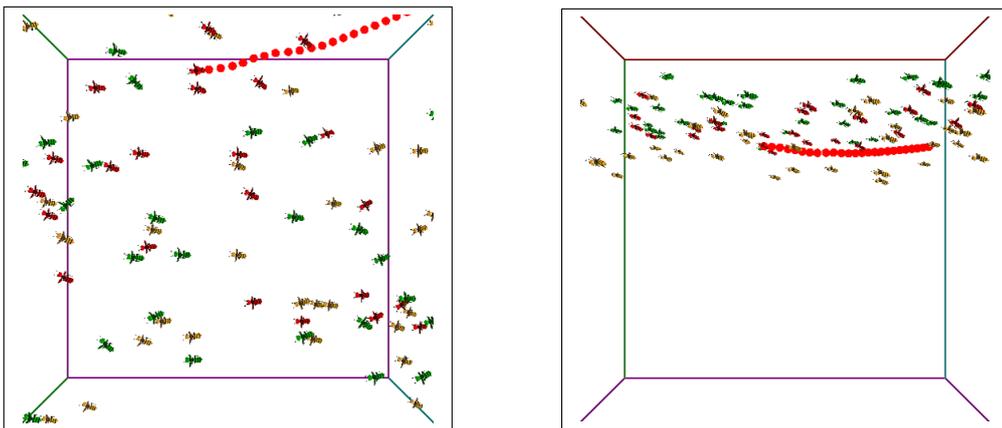


図4.3: 群飛飛翔を行うニホンミツバチの群れ (2秒)

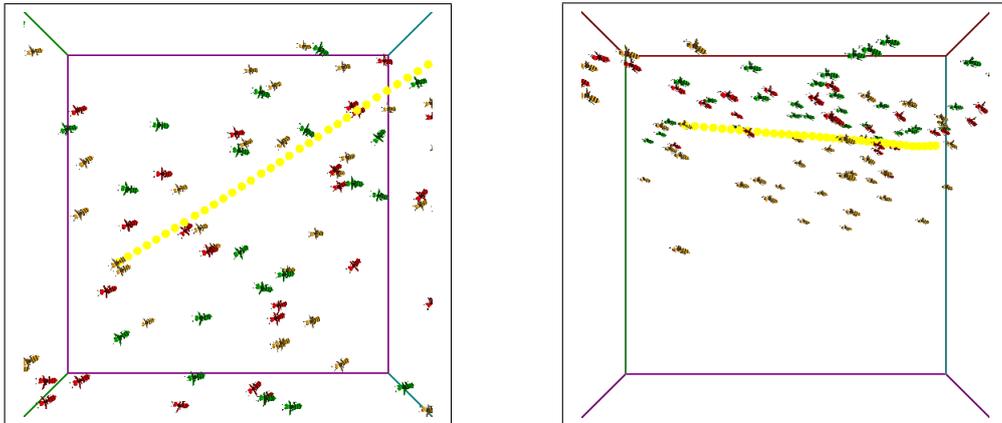


図 4.4: 群飛飛翔を行うニホンミツバチの群れ (4 秒)

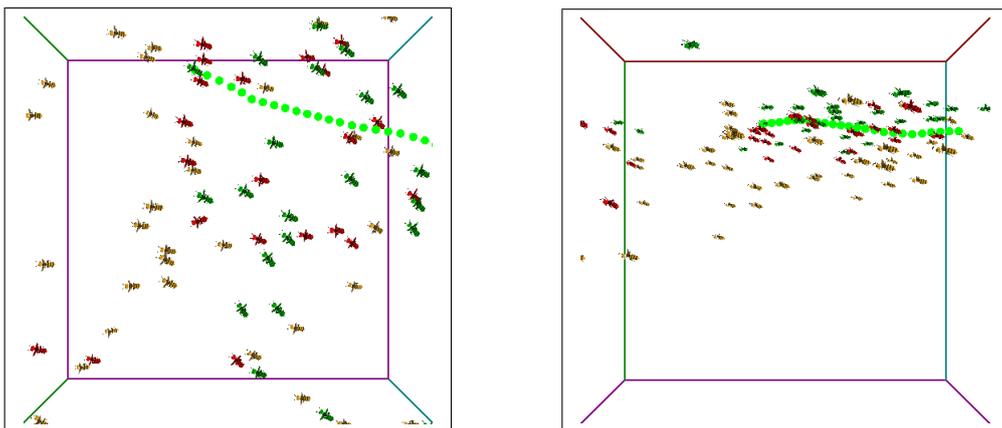


図 4.5: 群飛飛翔を行うニホンミツバチの群れ (6 秒)

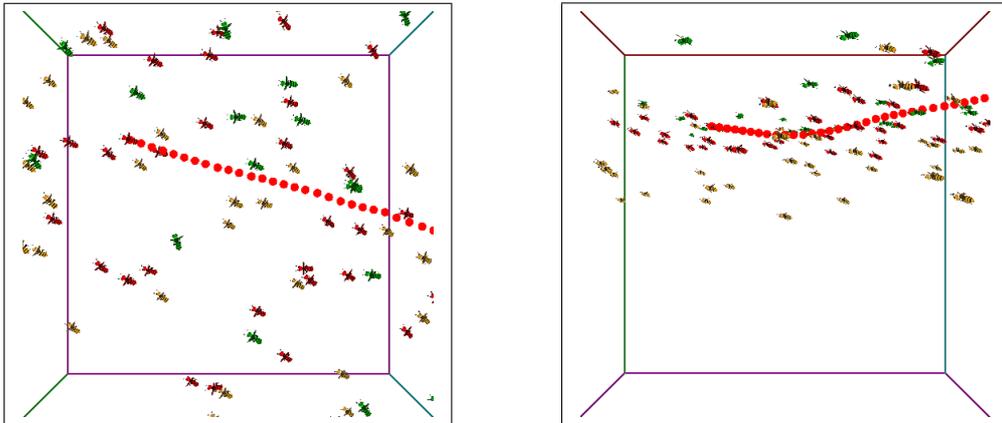


図 4.6: 群飛飛翔を行うニホンミツバチの群れ (8 秒)

群飛飛翔を行うニホンミツバチの群れを表現する手法として、boid アルゴリズムに群飛ルールを追加し、モデルに適用した。それにより、群れの中心を通過して上昇する動きと群れの頂上付近から下降する際に側方に向かい、群れの周辺に向かう動きを表現できた。赤色のモデルは上昇状態のモデルである。青色のモデルは下降状態のモデルである。黄色のモデルは通常状態のモデルである。また、1つのモデルから移動の軌跡が表示されている。その軌跡の色が赤色の場合は上昇状態の移動、青色の場合は下降状態の移動、黄色の場合は通常状態の移動である。図 4.7～図 4.10 は、動作開始から 20 秒後の、boid アルゴリズムの基本ルールのみを適用したモデルと、boid アルゴリズムの基本ルールに群飛ルールを追加して適用したモデルの、自由に飛翔している様子を比べたものである。図 4.7 と図 4.8 は上から見た図である。

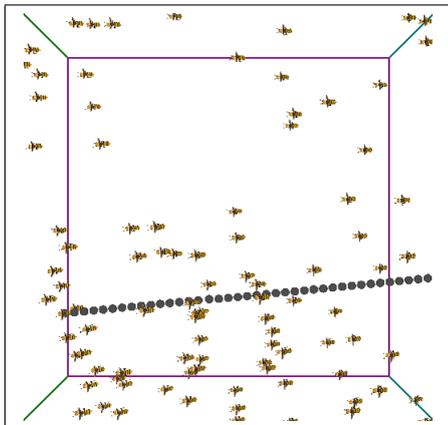


図 4.7: 基本ルールのみでの群れ

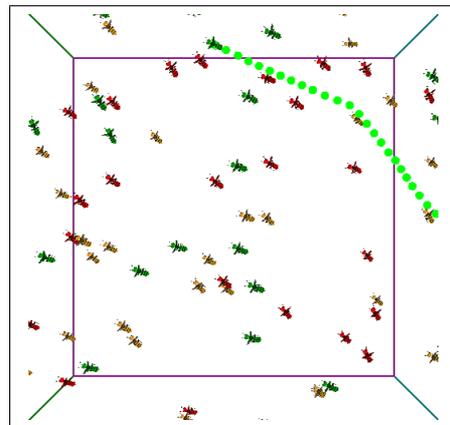


図 4.8: 群飛ルールを追加した群れ

図 4.9 と図 4.10 は手前から見た図である。

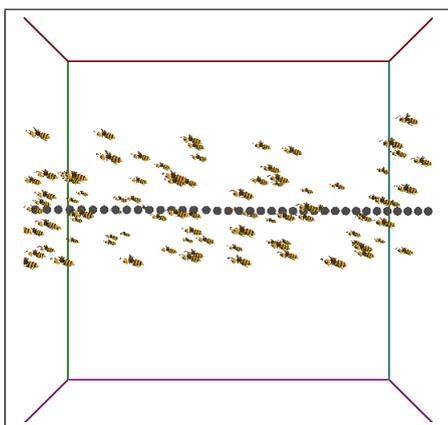


図 4.9: 基本ルールのみでの群れ

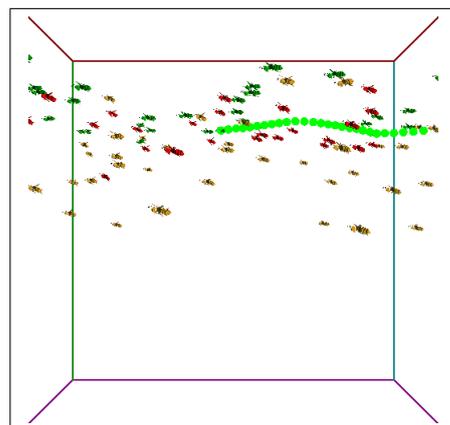


図 4.10: 群飛ルールを追加した群れ

boid アルゴリズムの基本ルールのみを適用した方は、上から見るとまとまりながら、ほぼ全てのモデルが同じ向きを向いて、真っ直ぐに移動している。そのため蜂の特徴である、直線的な移動は表現できているが、移動方向を頻繁に変える様子は表現できていない。そのため、蜂の群飛飛翔とは呼べない。群飛ルールを追加した方は、多くのモデルが同じような方向を向きつつも、広く分布している。また、各モデルが直線的な移動を行いながらも、上昇移動や下降移動、方向転換

を行い、頻繁に移動方向を変える様子も表現できている。そのため、蜂の群飛飛翔が表現できていると言える。

4.2 蜂球形成行動を行うニホンミツバチの群れ

蜂球形成行動を行うニホンミツバチの群れの様子を表現するために、モデルは200体使用した。以下の図は蜂球形成行動を行うミツバチの群れの様子を示したものである。左側が上から見た図であり、右側が手前から見た図である。各図の中の、中央に見える緑色の球体がオオスズメバチモデルである。オオスズメバチモデルは移動は行わない。青色の球体が蜂球の範囲である。また、灰色のブロックを地面として扱う。

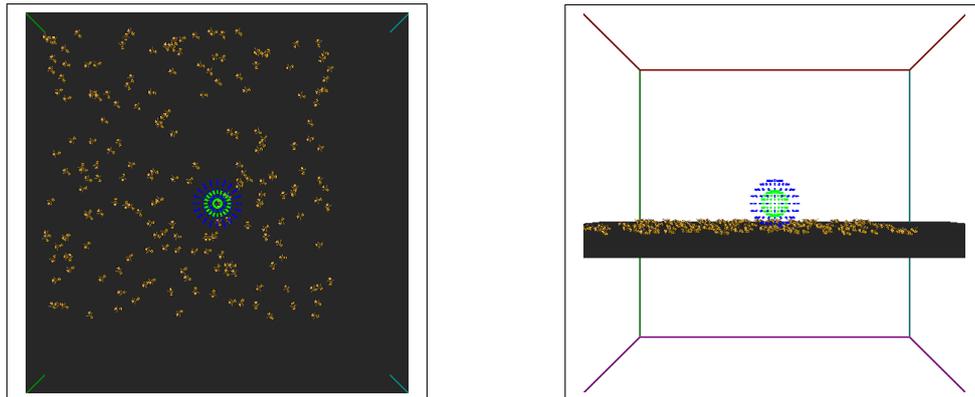


図 4.11: 蜂球形成行動を行うニホンミツバチの群れ (0 秒)

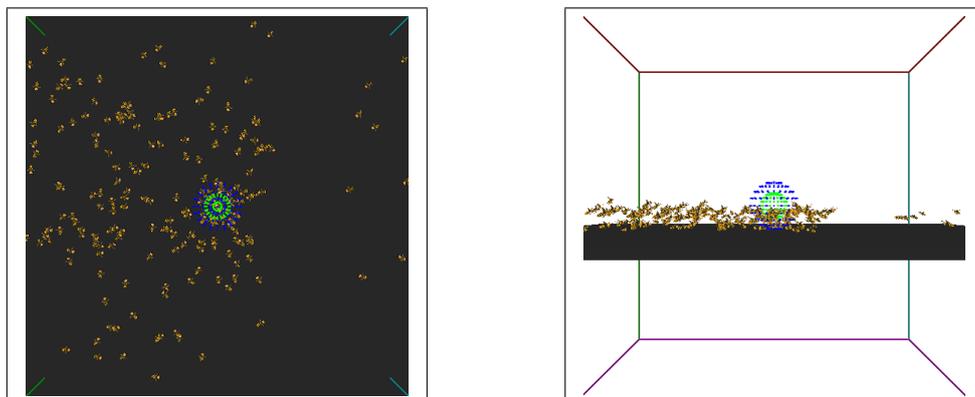


図 4.12: 蜂球形成行動を行うニホンミツバチの群れ (1 秒)

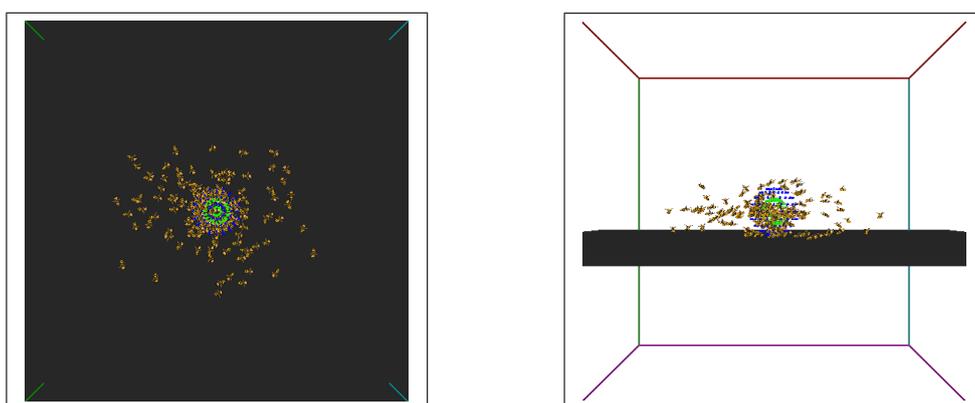


図 4.13: 蜂球形成行動を行うニホンミツバチの群れ (2 秒)

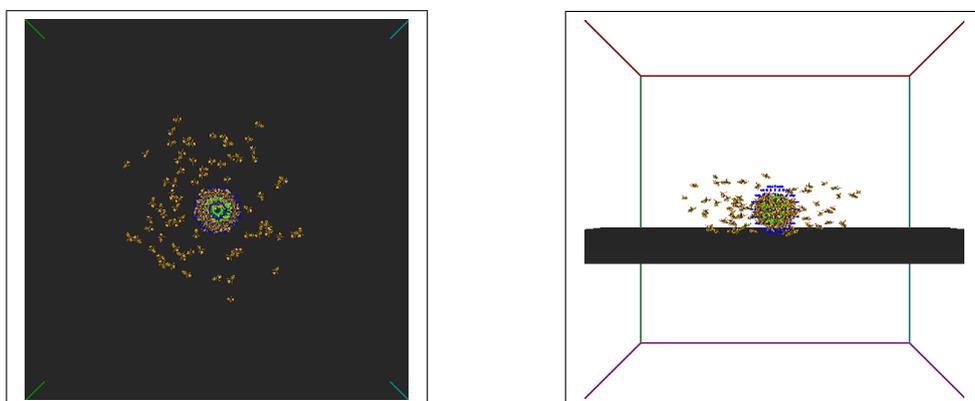


図 4.14: 蜂球形成行動を行うニホンミツバチの群れ (3 秒)

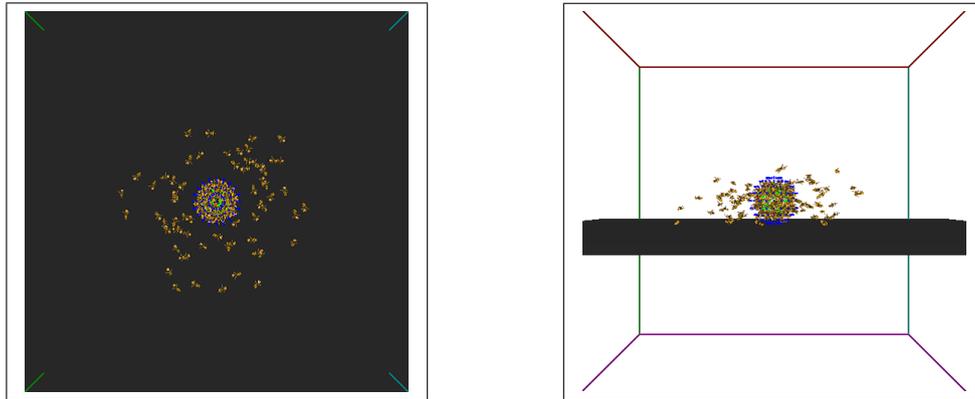


図 4.15: 蜂球形成行動を行うニホンミツバチの群れ (4 秒)

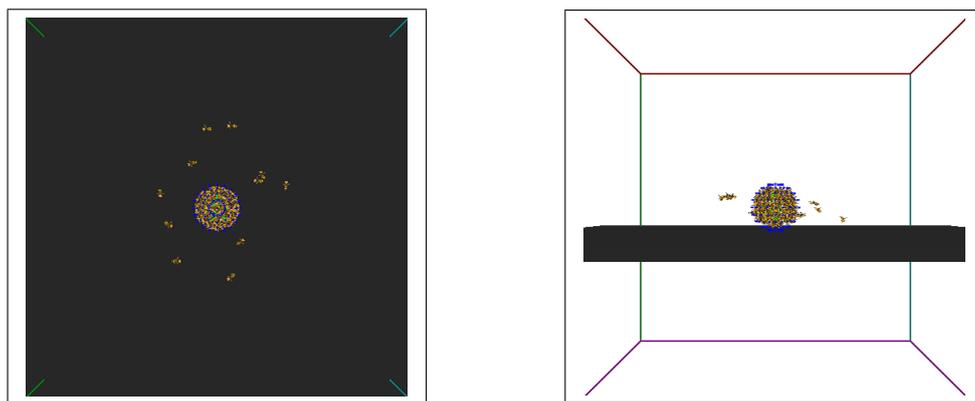


図 4.16: 蜂球形成行動を行うニホンミツバチの群れ (50 秒)

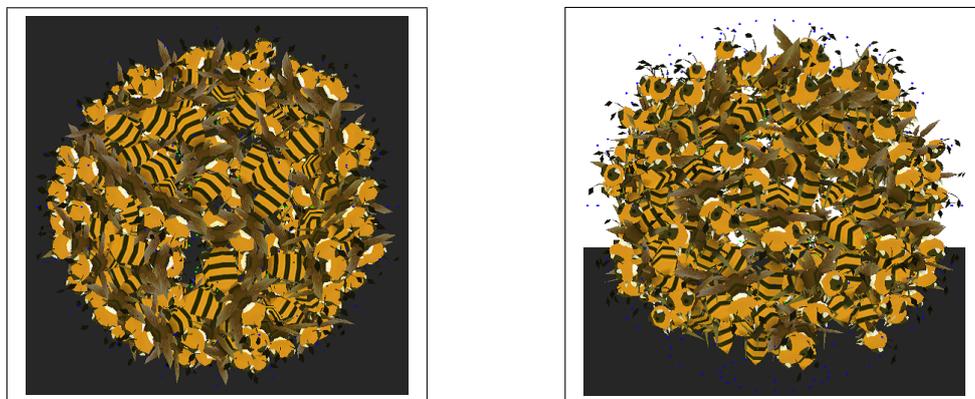


図 4.17: 蜂球形成行動を行うニホンミツバチの群れ (50 秒) を拡大

図 4.11～図 4.17 は、地面にいるニホンミツバチ達がオオスズメバチに襲い掛かる様子を表現したものである。結合ルールによる回転運動により、蜂球を形成するための曲線的な移動を行っている。また、引力ルールと斥力ルールを使い、きちんとした球状の蜂球を作っている。また、蜂球を形成しているモデル同士が衝突を避けるために、分離ルールによる回避行動を行っている様子を表している。分離ルールにより、ニホンミツバチ達が蜂球を形成するための、わらわらとした動きの様子が表現できている。

4.3 考察

boid アルゴリズムの結合、整列、分離という基本ルールに、蜂の群飛飛翔を表現するための群飛ルールを追加することで、群飛飛翔を行うニホンミツバチの群れの動きを表現することができた。また、基本ルールを変更し、引力と斥力ルールを追加することで、蜂球形成行動を行うニホンミツバチの群れの動きを表現することもできた。しかし、いくつかの問題点もある。群飛飛翔を表現する際に、モデルによっては移動中に細かくギザギザした動きをすることがあった。これは、衝突を回避するために、分離ルールが強く働いたことが原因だと考えられる。また、オオスズメバチに向かって移動する段階では、蜂球の形成に加われずに、形成中の蜂球の周辺を回転運動し続けてしまうモデルがでてきてしまった。蜂球を形成する段階では、モデル同士が近すぎると、重なってしまうことがあった。また、今回の研究では、オオスズメバチは移動しない条件であったが、実際のニホンミツバチはオオスズメバチに出会うと牽制行動を取り、隙を狙って蜂球形成行動を仕掛けることがある。そのため、その行動を取り入れることができれば、より本物に近いニホンミツバチの行動を表現できるようになるであろう。

第 5 章

まとめ

本研究では、ニホンミツバチの群飛飛翔と蜂球形成行動を表現する手法を提案することを目的とした。群飛飛翔と蜂球形成行動を表現するために、boid アルゴリズムと追加したルールを用いた。boid アルゴリズムのルールを用いて、直線的に飛翔するニホンミツバチの群れを表現した。その群れに、群飛ルールを追加することで、位置によって上昇行動や下降行動、方向転換を行っていく、蜂の群飛飛翔を表現することができた。また、boid アルゴリズムのルールを変更し、外敵であるオオスズメバチに襲い掛かり、取り囲んで行く動きを表現することができた。また、引力ルールと斥力ルールを追加し、整った形の蜂球を形成することもできた。しかし、課題点もある。今回の研究では、オオスズメバチは移動しない条件であったが、実際のニホンミツバチは、オオスズメバチに出会うと牽制行動を取り、隙を狙って蜂球形成行動を仕掛けることがある。そのため、その行動を取り入れることができれば、より本物に近いニホンミツバチの行動を表現できるようになるであろう。

謝辞

本研究を締めくくるにあたり、渡辺先生、三上先生、岸本先生に深く感謝致します。渡辺先生には、特に熱い指導をして頂きました。石川先生には、数式をはじめとした、研究についての相談に何度も乗って頂きました。竹内先生には、非常に素敵な蜂モデルの使用許可を頂きました。蜂モデルを用いることにより、多くの方々により気持ち悪くなった(褒め言葉)という評価を頂くことができました。阿部先生には、ネタの仕込み方について熱く指導して頂き、最終発表者の心構えについて深く学ぶことができました。ゲームサイエンスプロジェクトの学部生や院生の皆様には、大変多くの手助けをして頂きました。特に、2013年1月18日に行われたエクストリームスポーツにおいて、私の手助けをして下さった全ての方々に感謝致します。最後に、約3年もの間、私の生活を支えて下さった、林家の方々に最大限の感謝を捧げます。ありがとうございました。これからもよろしくお願い致します。

参考文献

- [1] 巻口誉宗. 群知能シミュレーションにおける異方性の創発とその解析評価. 北海道大学, 2010.
- [2] 三宅陽一郎. Chrome hounds におけるチーム AI. IGDA 日本, 2007.
- [3] Craig W. Reynolds. Flock, herds, and schools : A distributed behavioral model. *In Computer Graphics 21(4)*, 1987.
- [4] Glenn Seemann David M. Bourg. ゲーム開発者のための AI 入門. オライリー・ジャパン/オーム社 (発売), 2005.
- [5] 森川幸人. マッチ箱の脳 使える人工知能のお話. 2000.
- [6] デジタルゲームの教科書制作委員会著. デジタルゲームの教科書 : 知っておくべきゲーム業界最新トレンド. ソフトバンククリエイティブ, 2010.
- [7] 朱朝江. 村岡一信. 水野尚. 空気力学に基づくトンボのビジュアルシミュレーション. 芸術科学会論文誌 Vol.2 No.4, pp. 146–155, 2003.
- [8] 石塚拓磨. リアルタイム 3DCG における密集した魚群の遊泳表現に関する研究. 東京工科大学, 2005.
- [9] 古川まき. 鈴木泰博. 森敏彦. ミツバチの採蜜・貯蔵行動を表現するモデルの作成. 名古屋大学大学院情報, 2007.

- [10] D. Karaboga. An idea based on honey bee swarm for numerical optimization. *Erciyes University*, 2005.
- [11] 山田養蜂場ミツバチ研究支援サイト. ミツバチについての基礎知識. <http://hobeey.bee-lab.jp/index.html>, 1月現在 2013.
- [12] 吉田忠晴. ニホンミツバチ-生態とその飼育法 Ⅷ-. ミツバチ科学 18(4), pp. 165–174, 1997.
- [13] 松浦誠. スズメバチの生態-なぜミツバチを襲うか-. ミツバチ科学 6(2), pp. 53–64, 1985.
- [14] 小野正人. オオスズメバチの採餌戦略に対するニホンミツバチの防衛戦略. ミツバチ科学 17(1), pp. 27–30, 1996.
- [15] 酒井哲夫. ミツバチのはなし. 技報堂出版, 1992.
- [16] 佐々木正己. ニホンミツバチ 北限の *Apis cerana*. 海游舎, 1999.
- [17] アンドレイ K. ブロドスキイ. 小山重郎. 小山晴子訳. 昆虫飛翔のメカニズムと進化. 築地書館, 1997.
- [18] 豊島喜郎. 間接飛翔筋型昆虫飛翔構造部の3次元コンピュータモデルの構築. 長岡技術科学大学, 1997.
- [19] 安齋祐一. 村岡一信. 千葉則重. 齋藤伸自. 翅のはばたきによる力を考慮した蝶の飛翔モデル. 情報処理学会論文誌, 2000.
- [20] OpenGL.org. Opengl. <http://www.opengl.org/>.
- [21] Fine Kernel Project. Fine kernel toolkit system. <http://fktoolkit.sourceforge.jp/>.
- [22] 渡辺大地. リアルタイムグラフィックスのためのツールキットに関する研究. 1996.