

群知能における多段創発メカニズム構築手法の提案

須賀 聖^{1,a)} 坪井 一晃² 栗原 聡¹

概要: 今日、身の回りに存在する人工物は、ほとんどのものがトップダウン型の設計で構築されている。しかし、高度に複雑なシステムを構築する場合、トップダウン型の設計方法を適用することが困難である。一方、高度に複雑なシステムの例として生物が挙げられる。生物の身体は、細胞→臓器→身体のように多段階の創発によるボトムアップ型の設計で構築される。従って、高度に複雑なシステムを構築する際に、多段創発は非常に有用な手法であると考えられる。しかし、現状では多段階の創発が起こる数理モデルはまだ存在していない。そのため、本研究では、二次元仮想物理環境上に、単純な規則で行動するエージェントを多数生成し、エージェントの振る舞いをシミュレートすることで、群知能による多段創発メカニズムの実現を試みた。その結果、生成されたエージェント同士の相互作用で新たな能力を獲得した上位レベルの個体の創発と、創発された上位レベルの個体の相互作用によって、さらに新たな能力を獲得した、より上位レベルの個体の創発を確認することができた。しかし、時間経過により、エージェントが大きな塊となり動作不能となるケースも観察された。複雑な能力を創発させるためには、エージェント間の有機的な結合方法を構築し、進化的手法による結合の創発といった方法を検討する必要がある。

Proposal for multi-layered emergence mechanism based on swarm intelligence

SUGA SATOSHI^{1,a)} TSUBOI KAZUAKI² KURIHARA SATOSHI¹

1. はじめに

今日、身の回りに存在する人工物は、ほとんどのものがトップダウン型の設計で構築されている。トップダウン型の設計とは、構築する対象物を設計する際に、対象物を部品に分解して設計することである。また、分解された部品を設計する際には、更に小さな部品に分解して設計が行われる。このように、構築するシステムのマクロレベルの要素を分解して、ミクロレベルの要素を設計していくのでトップダウン型と呼ばれている。しかし、高度に複雑なシステムを構築する場合、システムの分解や部品の組み立てが非常に困難であり、トップダウン型の設計方法を適用す

ることが困難である。

一方、高度に複雑なシステムの例として生物が挙げられる。生物の身体は、細胞が集まることによって構築されている。この細胞の群が互いに相互作用を起こし、自己組織化することによって、生物の身体が構築される。つまり、細胞が自発的に生物の身体を生成していると言える。このように、ミクロレベルの構成要素の相互作用及び自己組織化によって、マクロレベルの要素が創発することでシステムの構築を行う設計をボトムアップ型の設計という。高度に複雑なシステムを構築する際に、ボトムアップ型の設計は非常に有用な手法であると考えられる。

また、高度に複雑なシステムは多層構造を持つものが多い。Havel は複雑系の指標として“スケールの厚み” [1] を提唱している。これは、複雑なものは、各スケールが重要な機能を持つということを意味している。具体例として、人間の身体を例に説明すると、まずメートルオーダーとして人間の身体全体の形がある。次にセンチメートルオーダーでは、胃や心臓、脳といった臓器レベルの機能を持つ。

¹ 電気通信大学大学院情報理工学研究所
The University of Electro-Communications, Chofu, Tokyo
182-8585, Japan

² 電気通信大学大学院情報システム学研究所
The University of Electro-Communications, Chofu, Tokyo
182-8585, Japan

a) ssuga@ni.is.uec.ac.jp

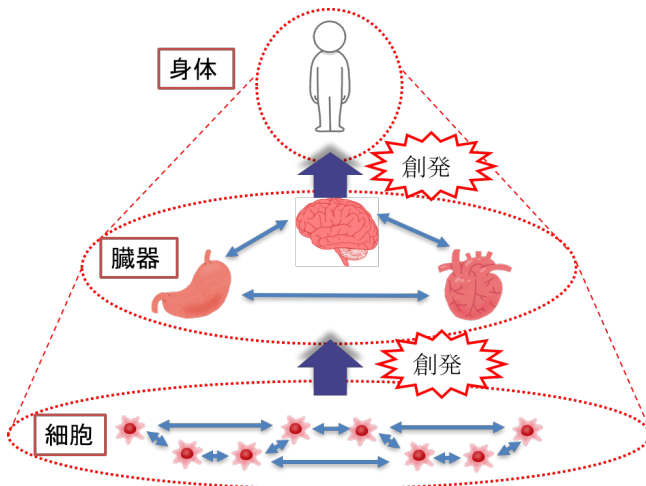


図1 多段創発の例 (人間の身体)

さらに小さなスケールのオーダーでは細胞レベルの機能を持ち、さらに小さいスケールでは、分子レベルや原子レベルの機能を持つ。このように、複雑なシステムは多層構造を持ち、各層で機能を持っている。

前述より、複雑なシステムは多層構造を持ち、ボトムアップ型による設計が有用であると考えられる。従って、複雑なシステムの構築には、階層的なボトムアップ設計が鍵となると考えられる。階層的なボトムアップ設計とは、多段創発である。人間の身体も多段創発によって構築されている。人間の身体は、細胞同士の相互作用によって、臓器が創発され、創発された臓器同士の相互作用によって身体が創発される(図1)。このように、ミクロレベルの相互作用によって創発された、マクロレベルのものが、更に相互作用を起こすことによって、よりマクロレベルの要素を創発するメカニズムを多段創発という。複雑なシステムの構築には、多段創発メカニズムの確立が重要であるといえる。

そこで本稿では、2次元仮想物理環境上に、単純な規則で行動するエージェントを多数生成し、エージェントの振る舞いをシミュレートすることで、群知能による多段創発メカニズムを実現する。生成されたエージェント同士の相互作用によって、上位レベルの個体を創発し、創発された上位レベルの相互作用によって、さらに上位レベルの個体を創発させるような、シミュレーションのルールと環境の設定を行うことで、階層的な群知能における多段創発メカニズムの構築を試みる。

2. ボトムアップ型による設計の研究

以前よりボトムアップ型のシステムの応用に関して様々な研究が行われている。例えば、Reynolds[2]によって提案された、鳥の群れ行動をシミュレーションしたBoidが挙げられる。Boidは、鳥に見立てたエージェントに単純なルールを与える。具体的には、(1)結合：近傍のエージェントの中心の方向へ近づく；(2)整列：近傍のエージェン

トの平均速度に速度や方向を合わせる；(3)分離：近隣のエージェントから最小限の距離を取るよう離れる。以上3つの単純なルールのみをエージェントに適用する。これらのエージェントからなる集団は、個々のエージェントが単純なルールに従って行動しているのにもかかわらず、鳥の群れが示すような統一的な集団飛行の挙動を創発する。Boidは、個々のエージェントの相互作用により、ボトムアップ的に統一的な集団飛行のシステムを設計していると言える。このシステムは、現在では映画やアニメーションの特殊効果に応用されている。

Boidのように、生物の集団行動に注目したボトムアップ型のシステム設計の研究として、Kennedyら[3]によって提案された粒子群最適化(PSO)という手法が挙げられる。PSOは、主に最適化の手法として使われている。基本的なPSOでは、複数の個体が互いに情報を共有しながら探索空間を動き回り、最適解を求めるアルゴリズムである。具体的には、個々のエージェントは、位置ベクトル、速度ベクトルを持ち、近傍のエージェントと自身の個体が発見した最適位置(最良評価値を獲得した場所)の情報を互いにやりとりしながら、群全体としての最適位置を発見(創発)するシステムである。PSOは、個々のエージェント間の情報のやり取りといった相互作用により、ボトムアップ的に最適解を創発するシステムであると言える。近年では、PSOを用いて多目的最適化問題を解く方法が提案されている[4]。

生物の行動に注目したボトムアップ型のシステム設計の研究としては、Dorigoら[5]によって提案された蟻コロニー最適化(ACO)という手法も工学的手法として研究・応用が盛んである。ACOは、蟻の群知能から着想を得た最適化手法である。蟻は餌を集めるとき、最初はランダムに探索を行うが、餌を見つけると、フェロモンを出しながら餌を巣へと持ち帰る習性がある。また、他の蟻は散布されたフェロモンに引き寄せられて行動し、餌を見つけると、自身もフェロモンを出して、餌を巣に持ち帰る。このようにして、餌と巣の間の経路がフェロモンで形成される。また、このフェロモンは揮発性であり、時間が経てば蒸発する。従って、餌と巣の間の経路が長いほど、その経路に散布されたフェロモンは蒸発しやすい。そのため、結果として、最終的にフェロモン濃度の高い部分は餌と巣の最短距離の経路として創発される。この仕組みを数理モデル化したものがACOである。ACOは巡回セールスマン問題のような組み合わせ最適化問題に対して有効性が確認されている[6]。

このように、現状では、ボトムアップ型のシステムは、最適化の手法として大きな成果を挙げている。また、ACOやPSOに関する近年の研究は、最適化計算の効率に焦点を当てたものが多い[7]。ここで、前述した研究成果の創発パターンをまとめると、Boidは1羽の行動ルールから集団の統一行動の創発、PSOは、1つの個体の行動と個体

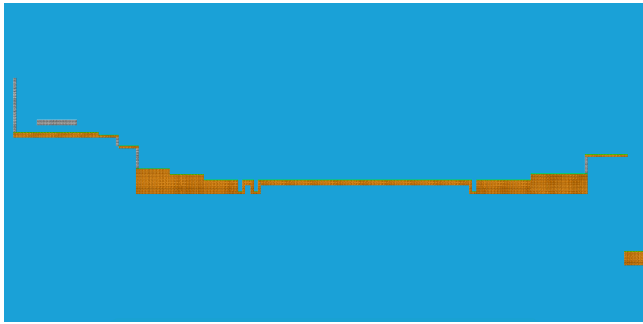


図 2 実験環境

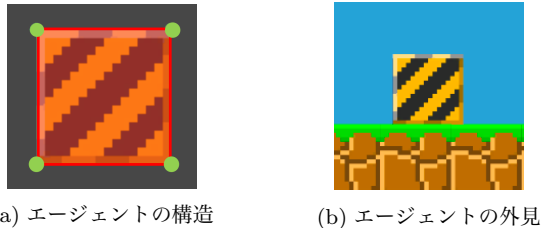


図 3 エージェントの構造

間の通信から群全体の最適解の創発, ACO では蟻のフェロモンコミュニケーションから最適解の創発である. ここで, 創発回数に注目すると, いずれの研究成果においても創発回数は 1 回のみである. 第 1 章で述べたように, 複雑システムの構築手法として, ボトムアップ型のシステム設計の応用を考えた場合, 多段階の創発が必要であると考え. しかし, 多段階にわたって創発が起こるようなシステムは, まだ存在していない. そこで本稿では, 多段階の創発をコンピュータ上に再現するために, マルチエージェント環境における新たな創発メカニズムを提案する.

3. 多段創発メカニズムの提案

3.1 実験環境

本研究では, 実験環境として 2 次元仮想物理環境エンジンである Unity を用いる (図 2).

エージェントが行動する環境は, 地形および障害物の配置によって形成される. 物理演算を適用しているので, エージェントは重力の影響や壁に衝突した際は, 反作用の力などを受ける.

3.2 エージェントの構造

エージェントは図のような単純な構造を持つ (図 3). 図 3(a) における, 赤枠内部は衝突判定のコライダーを表している. コライダーとは, 物理衝突のためのオブジェクト形状のことを意味している. 赤枠内部はエージェントのボディ部と言える. 緑丸は, エージェントの回転移動のためのモーターを表している. このモーターは, 地面や障害物, 他のエージェントとの衝突判定は行わないように設定している. 従って, エージェントが外部から物理的な影響を受けるのは赤枠内の範囲に何らかの作用があったときのみ

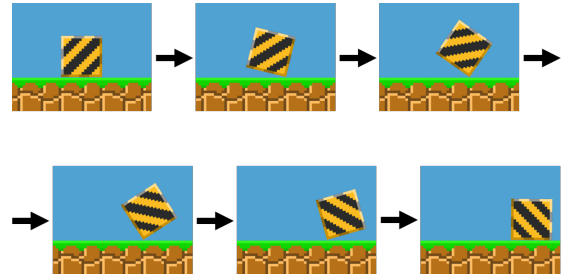


図 4 回転移動の様子

である. 緑丸のモーターは透明に設定してあるため, エージェントの見た目は図 3(b) のようになる. モーターは全部で 4 つあり, ボディ部の各頂点に完全に接着している.

3.3 エージェントのルール

各エージェントは以下のような単純な性質しか持たない.

(1) 回転移動

エージェントの構造から, モーターは, エージェントの頂点に完全に接着している. 従って, モーターが回転するとき, エージェントのボディ部も一緒に回転する仕掛けとなっている. また, モーターは, モーターが取り付けられているボディの頂点が地面に接地しているときにしか回転できない. また, 回転する際はモーターの位置を地面に固定して回転する. 従って, エージェントは, 地面とエージェントの頂点との接点を中心として弧を描くような軌跡を描きながら回転移動を行う (図 4).

(2) 結合

エージェントが他のエージェントと衝突したとき, エージェントは衝突したエージェントと結合する. エージェントの結合パターンは 3 種類ある. 辺で結合するパターンが 1 種類と, 頂点で結合するパターンが 2 種類である. 結合の種類は等確率で選択される. 結合パターンの違いによって, 複合エージェントの回転移動時の挙動に違いが生じる (図 5). 従って, 結合によって生じるエージェントに多様性を持たせることができる.

(3) 生命力

エージェントは体力を持つ. 複合エージェントの場合, 自身を構成する単一エージェントが持つ体力の総和を複合エージェントの体力とする. エージェントの体力は, エージェントが移動できなかった場合に 1 つ減少する. 体力が 0 になったエージェントは消滅する.

3.4 シミュレーションの流れ

シミュレーションは図 6 に示した流れで実行される. シミュレーションの初期状態では, 単一エージェントを N 個生成する. その後, 生成されたエージェントは, 前述したエージェントのルールに従って行動する. まず, 各エー

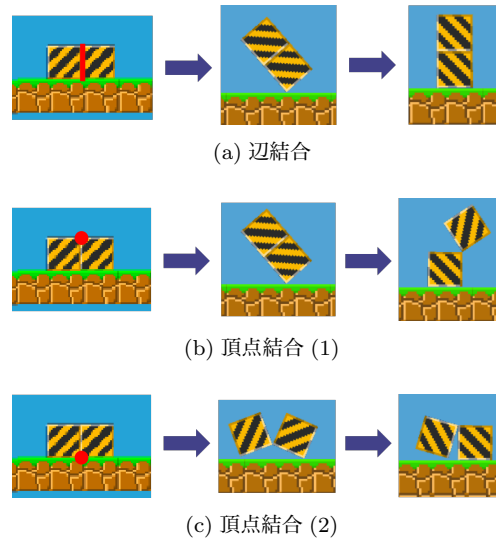


図 5 結合パターンとその挙動

エージェントは移動方向の選択を行う。方向は左右の2種類であり、等確率で選択される。次に、選択された方向に向けて回転移動を行う。続いて、移動判定を行う。移動の可否は、回転対象となるモーターを含んでいる単一エージェントの中心が、一定距離移動できたかどうかによって判断する。移動を行うことができた場合は、再度同じ方向へ回転移動を行う。一方、移動を行うことができなかった場合は、エージェントの持つ生命力を1つ減少させる。このとき、生命力が1以上であれば、移動方向の選択を再度行い、生命力が0になってしまった場合は、エージェントは消滅する。また、エージェントは回転移動中、停止中に問わず、他エージェントと衝突したとき、衝突したエージェントと結合を行う。結合してできた個体は、新たな複合エージェントとして生成する。また、本システムでは、シミュレーションの過程で誕生した複合エージェントに関して、複合エージェントを構成する単一エージェントの配置と、各単一エージェントの結合パターンをリストに記録している。リストには、初期状態の単一エージェントも含む。また、シミュレーションの過程で、エージェント同士の結合や消滅が行われる結果、エージェント数が0になってしまうことが起こりうる。そこで、一定の間隔で、新たなエージェントを、リストに登録されているエージェントの中から、一つを等確率で選択して生成する。

3.5 創発の基準

多段創発メカニズムを構築するのに際して、上位レベルの個体が創発されたと判定できる基準が必要である。図 [8] は、生物学における階層を判定する基準として以下を挙げている。

- 包含関係：ある複雑さのレベル (階層レベル) に属する階層単位がいくつか集まって、より複雑な階層単位を作る

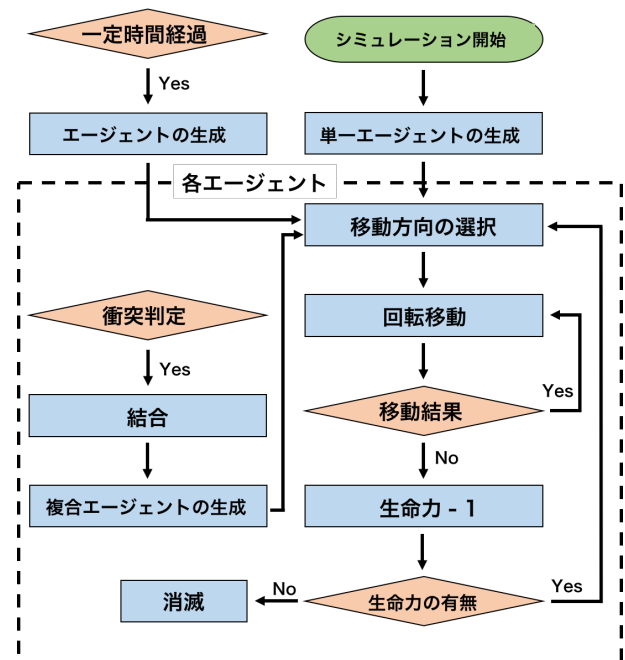


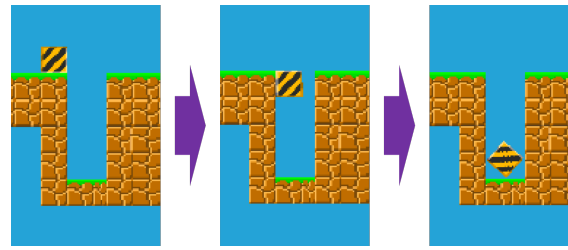
図 6 シミュレーションの流れ

- 新機能の付加：上位の階層単位は下位の階層単位には見られない、新しい機能をもつ

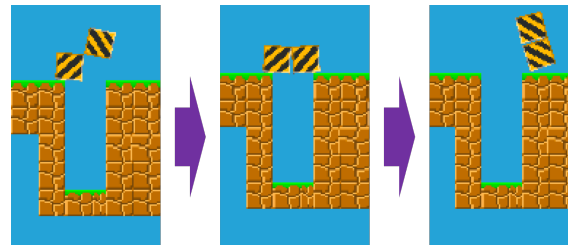
従って、本シミュレーションでは、下位レベルのエージェントのみでは獲得できない能力を、下位レベルのエージェント同士の相互作用によって獲得できた場合に、上位レベルの個体が創発されたと定義する。

4. 実験結果

図2の環境を用いて、エージェントのシミュレーションを行った。その結果、まず図7のようなエージェントの様子が観察された。図7(a)は単一エージェントの様子を表しており、図7(b)は単一エージェントの相互作用によって発生した複合エージェントの様子を表している。図7では、

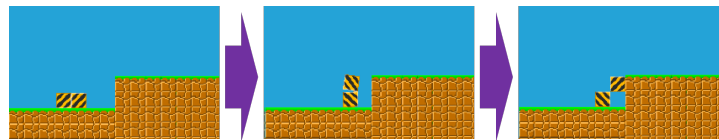


(a) 単一エージェント

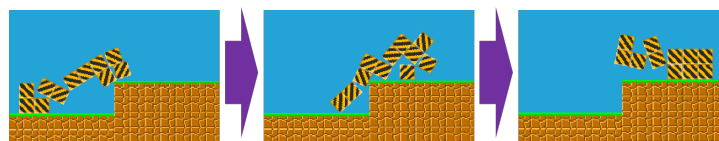


(b) 相互作用の結果生じた複合エージェント

図 7 第 2 層エージェントの創発



(a) 第 2 層エージェント



(b) 相互作用の結果生じた複合エージェント

図 8 第 3 層エージェントの創発

エージェントの進行方向に、地形環境により生じた穴が存在する場合のエージェントの振る舞いを表している。このとき、単一エージェントの場合は、回転移動を行うと穴に落ちてしまう。一方、単一エージェントの相互作用によって発生した複合エージェントでは、穴に落ちることなく先のフィールドへ移動することができるエージェントの様子が観察された。従って、図 7(b) のような複合エージェントは、単一エージェントのみでは獲得できない、穴を乗り越えるという新たな能力を獲得できたと言える。よって、単一エージェントを下位の階層単位とした上位レベルの階層をもつエージェントが創発されたと言える。観察された創発は 1 回目の創発であるため、図 7 では、第 2 層エージェントが創発されたと言える。

創発された第 2 層エージェントは、穴を越えた先のフィールドを回転移動する。しかし、一定時間経過すると、エージェントは、図 8(a) のように高さのある段差に差し掛かる。第 2 層エージェントでは、図 8(a) の高さの段差を乗り越えることはできない。その後、第 2 層エージェントは、段差より手前のフィールドで回転移動や、他のエージェン

トとの結合を繰り返した。一定時間が経過すると、図 8(b) のように高さのある段差を乗り越えることができるエージェントが現れた。従って、図 8(b) のような、第 2 層エージェントと第 2 層以下のエージェントとの相互作用によって生成された複合エージェントは、第 2 層エージェントのみでは獲得できない、高さのある段差を乗り越えるという能力を獲得できたと言える。よって、第 2 層エージェントを下位の階層単位とした上位レベルの階層をもつエージェントが創発されたと言える。図 8(b) では、観察された創発は 2 回目の創発であるため、第 3 層エージェントが創発されたと言える。また、創発されたエージェントが、更に相互作用を起こして、新たな創発現象を発現させたことから、多段創発の様子が観察されたと言える。

第 3 層エージェントは、段差を越えた先のフィールドを移動した後、更に大きな段差に衝突する。この段差を乗り越える能力の獲得を目指し、回転移動と他エージェントとの結合を繰り返した。しかし、第 3 層エージェントの創発には至らず、図 9 のように、大きな塊となり、動作不能な状態となるケースが観察された。

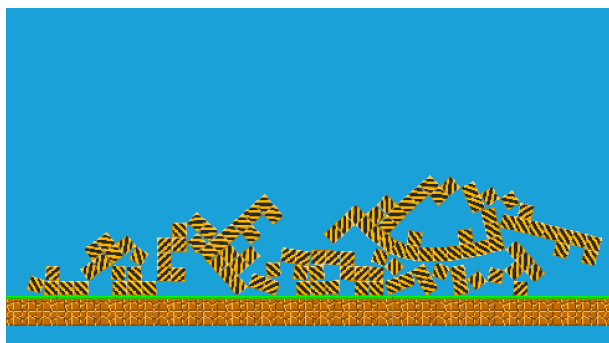


図 9 大きな塊となり動作不能となるエージェント

これは、本研究で提案した創発方法において、エージェントの相互作用が衝突時の結合のみであるため、複合エージェントの動作に複雑が生じず、高度な機能の創発には至らなかったためであると考えられる。

5. まとめと今後の課題

本研究では、二次元仮想物理環境エンジンを利用し、単純な規則で行動するエージェントを用いたマルチエージェント環境において、群知能による多段創発メカニズムを構築する手法を提案した。シミュレーションの結果、2段階の創発を確認することができ、群知能による多段創発によって新たな環境適応のための能力を獲得できることの可能性を確認することができた。しかし、時間経過により、エージェントが大きな塊となり動作不能な状態となるケースも観察され、複雑な能力の創発には至らなかった。

今後の課題は、複雑な能力を創発させるために、エージェント間の有機的な結合方法を検討し、進化的手法によって、より複雑な能力を獲得できる複合エージェントの創発が課題である。また、多段創発を工学的に利用するためには、多段創発を制御する必要がある。現在、ボトムアップ型で成り立っているものは進化の結果論として出来上がったものである。従って、多段創発を制御できているとは言えない。そこで、多段創発を工学的に利用するために、意図したものを創発する方法論の確立を目指していく。

参考文献

- [1] Ivan M Havel. Artificial thought and emergent mind. In *IJCAI*, pp. 758–766, 1993.
- [2] Craig W Reynolds. Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH computer graphics*, Vol. 21, No. 4, pp. 25–34, 1987.
- [3] J Kennedy and R Eberhart. Particle swarm optimization. In *Neural Networks, 1995. Proceedings., IEEE International Conference on*, Vol. 4, pp. 1942–1948. IEEE, 1995.
- [4] Jonathan E Fieldsend and Sameer Singh. A multi-objective algorithm based upon particle swarm optimisation, an efficient data structure and turbulence. 2002.
- [5] Marco Dorigo, Gianni Di Caro, and Luca M Gambardella. Ant algorithms for discrete optimization. *Artificial life*, Vol. 5, No. 2, pp. 137–172, 1999.
- [6] Eric Bonabeau, Marco Dorigo, and Guy Theraulaz. In-

spiration for optimization from social insect behaviour. *Nature*, Vol. 406, No. 6791, pp. 39–42, 2000.

- [7] Sabrina Oliveira, Mohamed Saifullah Hussin, Andrea Roli, Marco Dorigo, and Thomas Stützle. Analysis of the population-based ant colony optimization algorithm for the tsp and the qap. In *Evolutionary Computation (CEC), 2017 IEEE Congress on*, pp. 1734–1741. IEEE, 2017.
- [8] 国まりな. 生物の複雑さを読む 階層性の生物学. 平凡社, 1996.