

遺伝的プログラミング技法を用いた多出力二分決定グラフの進化

食物連鎖におけるマルチエージェントの進化シミュレーション

Evolution of n -BDD Using Genetic Programming Method a Food Chain Simulation of Evolutional Agents

森脇 康介*† 横井 大祐* 犬塚 信博* 伊藤 英則*
Kousuke Moriwaki Daisuke Yokoi Nobuhiro Inuzuka Hidenori Itoh

* 名古屋工業大学 知能情報システム学科
Nagoya Institute of Technology, Department of Intelligent and Computer Science, Nagoya 466-8555, Japan.

1998年2月19日 受理

Keywords: evolution, genetic programming, BDD, quasi-ecosystem, co-evolution, food chain.

Summary

Genetic algorithm (GA) is an effective method to solve combinatorial optimization problems. The efficiency depends on how genes expressed. Although gene expression using finite state automata has already been proposed, it is not always suitable for genetic operation. We propose an gene expression using n -BDD, which lets genetic operations work efficiently. We show a simulation of Quasi-Ecosystem including herbivores, carnivores and plants as an experimental model to confirm the efficiency of our method compared with systems using finite state automata and classifier systems. Our experiments tell superiority of the n -BDD expression at the adjustment speed of animals' action strategy. In the Quasi-Ecosystem, strategies of animals' behavior are evolved in the genetic expressions. We also observed food chain in the environment, in which quick evolution of animals' behavior is necessary, using n -BDD expression.

1. はじめに

自然界には異なる種類の生命体が被食者、捕食者の関係を持ち、それぞれが種族を絶滅させず世代交替してゆく食物連鎖関係が存在している。図1に植物、草食動物、肉食動物の3種類に単純化した食物連鎖のモデルを示す。ここでは、草食動物が植物を捕食（植物は草食動物に被食される）、肉食動物が草食動物を捕食する（草食動物は肉食動物に被食される）。また、草食動物と肉食動物が捕食体を発見できずに餓死すれば共にその場所に植物が発生するモデルを表現している。[Takashina 94]は生命体個々を独立のプログラムで記

述し、個々の生命体が食物連鎖系の中で自己適応してゆく疑似生態系モデルのシミュレーションを行った。個々の生命体（以降エージェントと呼ぶ）は遺伝的アルゴリズムでより高度な行動戦略を獲得し、次世代にそれを遺伝させてゆく。このシステムはエージェントがと

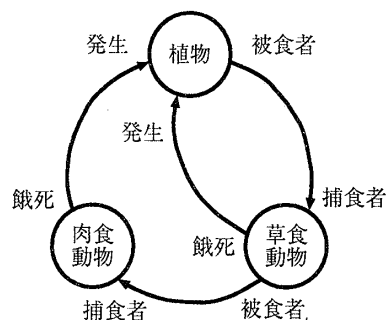


図1 食物連鎖関係

† 現在、NTTコミュニケーションウェア株式会社に勤務とともに、名古屋工業大学に在籍。

る行動戦略の組合せ最適化問題の一種として捉えることができる。これまでにこのような問題の解決手法の一つに [Takashina 94] で用いられた有限状態オートマトン [Fogel 67] やクラシファイア・システム [Holland 86] などが発表されている。しかしながら、問題によっては遺伝子の表現方法が不向きであったり、形質遺伝に難のある場合があった。

簡単で遺伝的操作が効果的に作用できる表現手法として BDD [Akers 78] (Binary Decision Diagram: 二分決定グラフ) を n 通りの出力に拡張した n -BDD (2章で詳述) を用いることを提案してきた [森脇 96a, Moriwaki 96b, 森脇 97a, Moriwaki 97b]。本稿ではこの手法が有効な表現方法であることを疑似生態系モデルを用いて実験により示す。さらに、有限状態オートマトンやクラシファイア・システムとの比較実験により有効な手法であることを示す。

2. n -BDD の遺伝的操作

2.1 n -BDD

BDD (二分決定グラフ) は [Akers 78] によって考案された論理関数の表現方法の一つである。その記憶効率や処理速度の面での優秀性から LSI の CAD の分野を中心にさまざまな分野で応用されてきている。また組合せ最適化問題の解法にも BDD を用いることが試みられ [柳谷 93]、これまで現実的な時間で解くことのできなかつた問題に対して大きな力を発揮している。

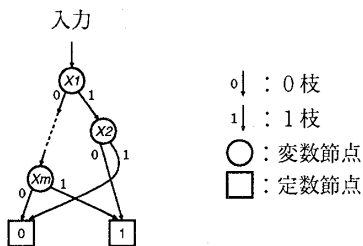


図2 二分決定グラフ (BDD)

図2で丸で表される節点を変数節点と呼び、すべての変数節点は枝と呼ばれる矢印を持つ。0枝と1枝の2種類が存在し、変数節点の左下から出る枝を0枝、変数節点の右下から出る枝を1枝と呼ぶ。変数節点に書かれた変数の値によって次の節点に分岐する。すなわち、変数節点に書かれた変数の値が0のときは0枝をたどり、1のときは1枝をたどる。こうして最終的にたどり着く四角で表される節点に書かれた値を出力値とする。この四角で表される節点は定数節点と呼ばれ

る。ここで、任意の変数節点から出る0枝および1枝は共に必ず下位の変数節点か、または定数節点を指す。これによって BDD 内に閉路やループを生じることはない。変数節点 B が変数節点 A よりも下位である (変数節点 A が変数節点 B よりも上位である) とは、与えられた入力変数順序 $X_1, \dots, X_i, \dots, X_m$ に対して、変数節点 A が変数 X_i の値により分岐するとき、変数節点 B は変数 $X_{(i+1)}, \dots, X_m$ のいずれかにより分岐することである。BDD の出力値は定数節点の値である。定数節点の持つ値は0か1の2通りである。以下に、この BDD を拡張する。本稿では各節点の分岐方

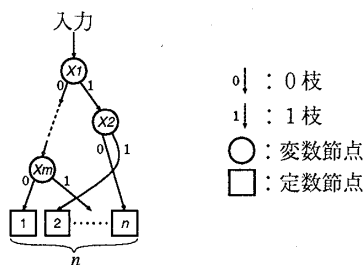


図3 多出力二分決定グラフ (n -BDD)

法は BDD と同一とし、 n 通りの出力ができるよう拡張する (図3参照)。これを n 出力値を持つ BDD、すなわち多出力二分決定グラフ (以下 n -BDD と略記する) と呼ぶことにする。 n -BDD では定数節点の数を n 個とする。 n -BDD は BDD と同様に一定長の2値変数列を入力し、定められた n 通りの定数の中から入力に応じて一つの値を出力する。

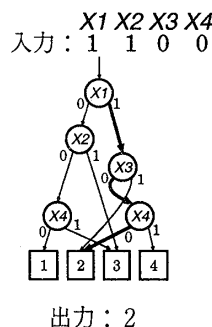


図4 入力値から出力値を決定する例

例えば図4のような n -BDD の入力ビット列が $(X_1, X_2, X_3, X_4) = (1, 1, 0, 0)$ のとき、まず変数 X_1 が1なので1枝をたどる。次に X_3 が0であるので0枝をたどる。次に X_4 も0であるので0枝をたどって定数節点2にたどり着く。この値2を出力値とする。この例では、出力結果は X_2 の値には依存しない。ここで、BDD における既約化手法 [Bryant 86] が n -BDD

でもそのまま使えるためコンパクトで一意的な表現が可能であり、またグラフ操作としての演算手法についても BDD で研究された手法が利用できる。

2.2 遺伝的操作

GA に用いられる遺伝子表現にはビット列などの 1 次元系列表現が一般的である。その他の表現としては状態遷移を遺伝子表現に用いた有限状態オートマトンや IF-THEN ルールまたはその集合を遺伝子表現に用いたクラシファイア・システムが報告されている。これらは環境から得られる情報を入力として、何らかの行動を決定する場合に有用である。

本稿では遺伝子表現として n -BDD を用い、これに節点から出る枝の変更（突然変異）、節点追加、節点削除の三つの基本的遺伝的操作を定義する*1（図 5 参照）。

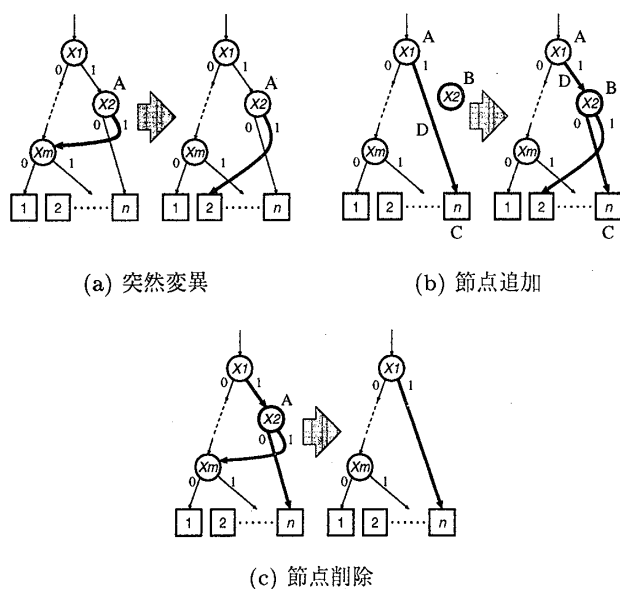


図 5 n -BDD の遺伝的操作

i) 突然変異

図 5 の (a) に示すように、ランダムに選ばれた変数節点 A の持つ 0 枝または 1 枝の指す向きを変更する。但し、新たに指す接点は任意の定数節点または A よりも下位の任意の変数接点とする。

ii) 節点追加

図 5 の (b) に示すように、新たな変数節点をランダムに選んだ位置に追加する。この変数節点を B とする。B よりも上位の変数節点 A から出る枝あるいは最上位の枝の中からランダムに選ばれた枝を D とする。ま

*1 遺伝的プログラミングにおいては遺伝的操作として交叉が導入される場合が多いが、枝が必ず定数節点または下位の変数節点を指さなければならない制約が n -BDD に存在するので、本稿では交叉の導入を省略している。

た、D によって指されている節点を C とする。このとき D の指す向きは C から B に変更される。また、節点 B の 0 枝または 1 枝のいずれか一方は節点 C を指すように設定される。節点 B の残りの一方の枝はランダムに選ばれた下位の任意の変数節点または任意の定数節点を指すように設定される。

iii) 節点削除

図 5 の (c) に示すように、ランダムに選んだ変数節点を削除する。この変数節点を A とする。A の持つ 0 枝および 1 枝も削除され、A を指していたすべての枝は A の持つ 0 枝または 1 枝が指す節点を指すように設定される。

3. n -BDD の入力、出力

3 種類のエージェント（肉食エージェント、草食エージェント、植物エージェント）がそれぞれ複数存在する環境の中で食物連鎖関係が成り立っていく様子をシミュレートする。各々のエージェントは n -BDD を遺伝子として持ち、この遺伝子がエージェントの行動戦略を決定する。これらの遺伝子を使った GA によってエージェントの行動戦略を進化させ、生態系全体を安定した食物連鎖が見られるように最適化していく。ここで遺伝子の入力となるのは環境から得られる表 1 に示す知覚情報を表したビット列である。また出力値は図 6 に示すそのステップ*2のエージェントの行動である。

表 1 入力ビット列の割り当てと意味（例）

X1	自分が空腹である
X2	自分が満腹である
X3	肉食エージェントが自分の視界の遠くに存在する
X4	肉食エージェントが自分の視界の近くに存在する
X5	草食エージェントが自分の視界の遠くに存在する
X6	草食エージェントが自分の視界の近くに存在する
X7	植物エージェントが自分の視界の遠くに存在する
X8	植物エージェントが自分の視界の近くに存在する

4. 疑似生態系の最適化

4.1 単純モデルによる実験

実験環境

20 マス×20 マスのフィールド上で、肉食エージェントと草食エージェント各 1 個体を使って行動戦略の最適化実験を行う。肉食エージェントの行動戦略を固定し、草食エージェントの行動戦略は GA により進化さ

*2 1 ステップはエージェントが知覚情報の入力と行動の出力を各 1 回行う時間である。

せる。固定された戦略により、肉食エージェントは、視界に草食エージェントがない限り、ランダムに8近傍に移動し続け、草食エージェントが視界に入った場合はこれの捕食を試みる。さらに、ここでの肉食エージェントは無限のエネルギーを持つこととする。

草食エージェントは、知覚情報をもとに図6に示す walk, runaway, eat, do-nothing, approach の5種の行動から一つを出力として選択する。草食エージェントは肉食エージェントから逃がれながら、かつ餓死しないように植物エージェントを捕食する行動戦略を獲得していく。

結果と評価

第1世代の集団を草食エージェント30個体から構成し、各個体を独立に評価する。フィールド上で草食エージェントが生き延びたステップ数を適応度とし、5個体エリート戦略を用いて次世代個体を選択してい

た。実験結果が確率により左右されやすいため実験を100回行い、各世代毎の適応度の平均値をとった。

遺伝子に与える知覚情報は表2に示す6通りの入力変数順序で実験した。草食エージェントは1個体なので X5, X6 は用いない。図7に結果を示す。

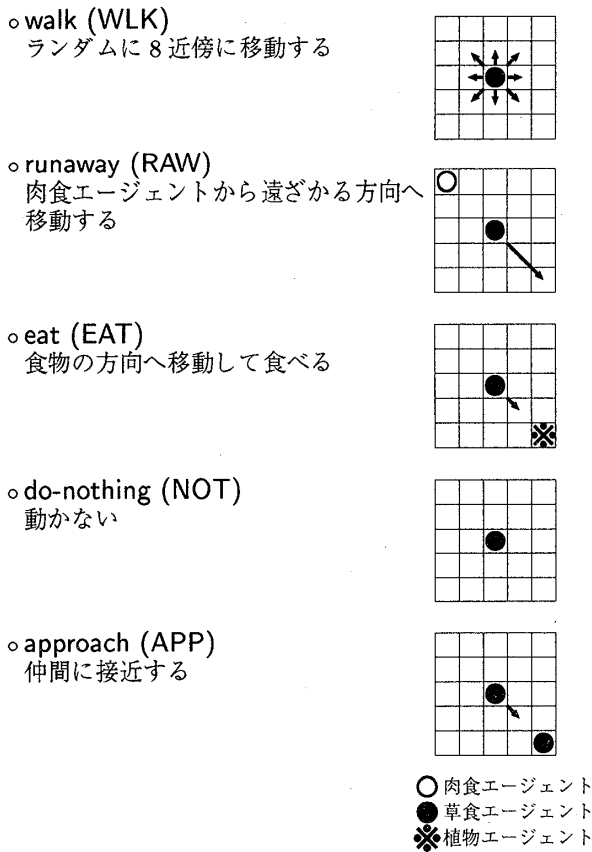
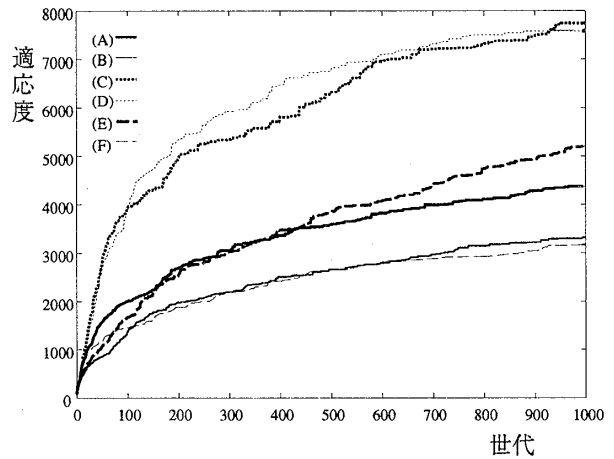


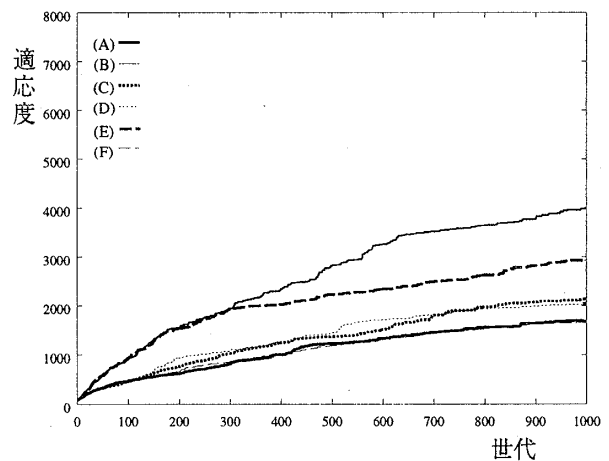
図6 エージェントが選択できる行動

表2 入力変数の順序

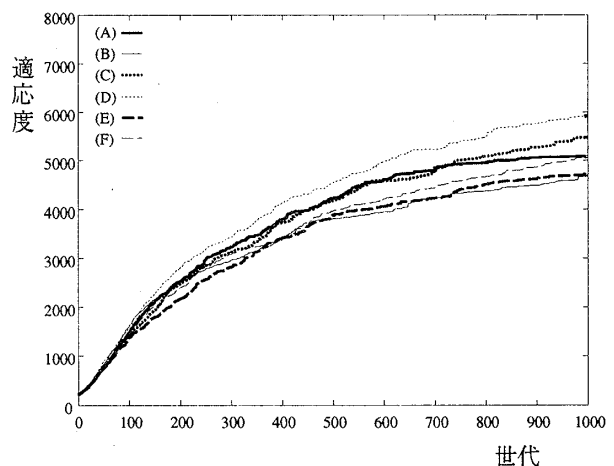
(A)	X1 → X2 → X3 → X4 → X7 → X8
(B)	X1 → X2 → X7 → X8 → X3 → X4
(C)	X3 → X4 → X1 → X2 → X7 → X8
(D)	X3 → X4 → X7 → X8 → X1 → X2
(E)	X7 → X8 → X1 → X2 → X3 → X4
(F)	X7 → X8 → X3 → X4 → X1 → X2



(a) n-BDD



(b) 有限状態オートマトン



(c) クラシファイア・システム

図7 入力順序別適応度推移

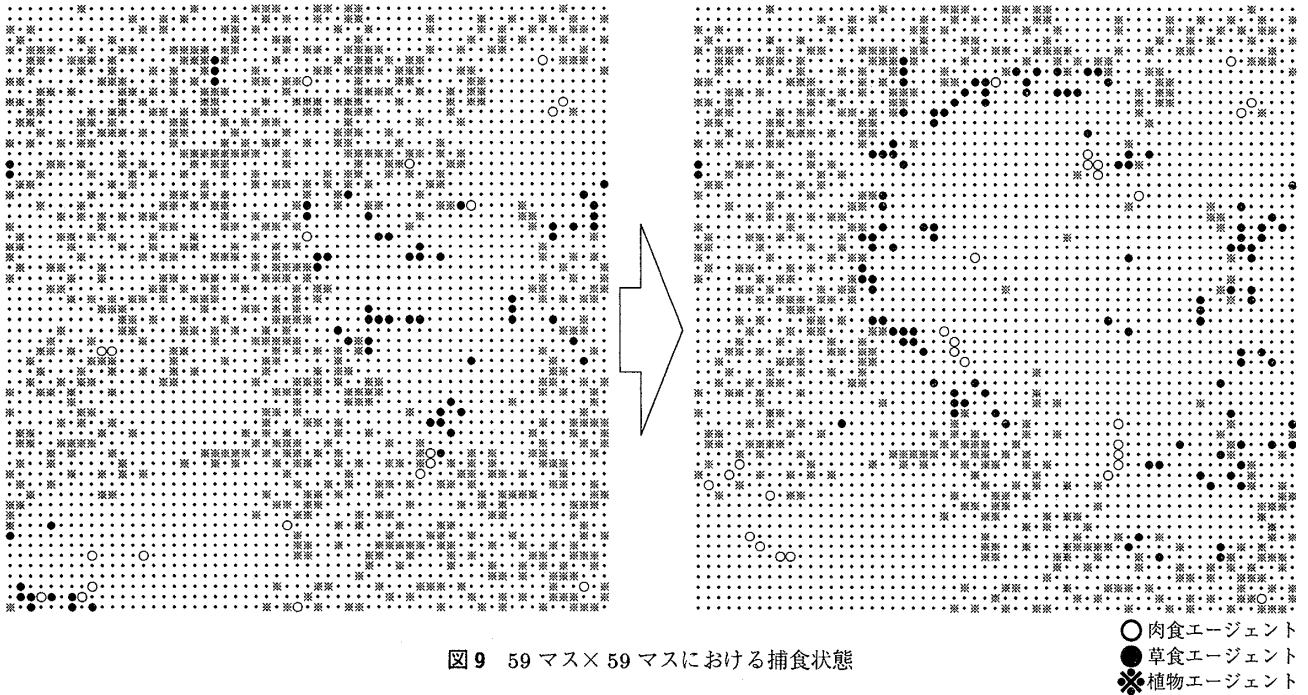


図9 59マス×59マスにおける捕食状態

○肉食エージェント
●草食エージェント
※植物エージェント

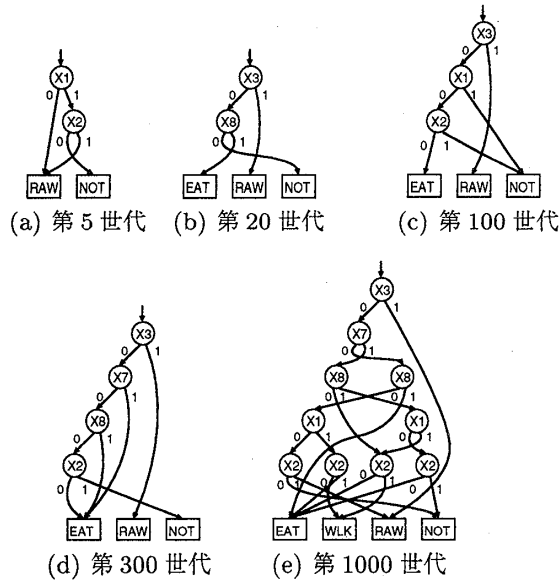


図8 単純モデルの草食エージェント BDD の推移 (例)

n -BDD では肉食エージェントを知覚する変数を先頭にした場合に最も適応度の上昇が速い。逆に有限状態オートマトン^{*3}では肉食エージェントを知覚する変数を末尾にした場合に適応度が速く上昇した。クラシファイア・システム^{*4}では入力変数とクラシファイアの条件部をパターンマッチングするので順序による影響が殆んど見られなかった。

図8に草食エージェントの n -BDD が世代を追って進化の様子を、典型的な実行例をもとに示す。先の

*3 有限状態オートマトンによる遺伝子表現は [Takashina 94] と同様に [Fogel 67] の方法にもとづいて行った。

*4 クラシファイアの条件部を入力メッセージとマッチさせ発火したクラシファイアの行動部を出力値とした。

実験で適応度の上昇が最も速かった変数順位 (D) の場合を示す。世代を追って BDD が複雑化し、生き延びるための行動を一つ一つ獲得してゆくことが分かる。第20世代では、視界内の肉食エージェントから逃げる行動を獲得している。また、食べる行動に関して、第100世代までは極力動かないことでエネルギー温存を図っているのに対し、第300世代以降は積極的に食べる行動を獲得している。また、第1000世代では、周りにある植物エージェントに応じてより洗練された行動を得ている。

4・2 疑似生態系での実験

実験環境

150マス×150マスのフィールド上で、肉食エージェントと草食エージェント複数個体を相互干渉させ行動戦略を自己適応させる実験を行う。肉食エージェントと草食エージェントは表1の知覚情報を入力とし図6の5種類の行動から一つを出力として選択する。肉食エージェントと草食エージェントはエネルギー値を持っており、肉食エージェントは草食エージェントを、草食エージェントは植物エージェントを食べることによりエネルギーを増加させる。エネルギーがある一定値を超えると親に遺伝的操作を施した子供を作り親の50%のエネルギーを分け与える。エネルギーが0になるとエージェントは餓死し、その周辺に一定個数の植物エージェントが発生する。肉食エージェントと草食エージェントは行動戦略を GA によって進化させ、動的に変化する環境に最適な行動戦略を獲得していく。

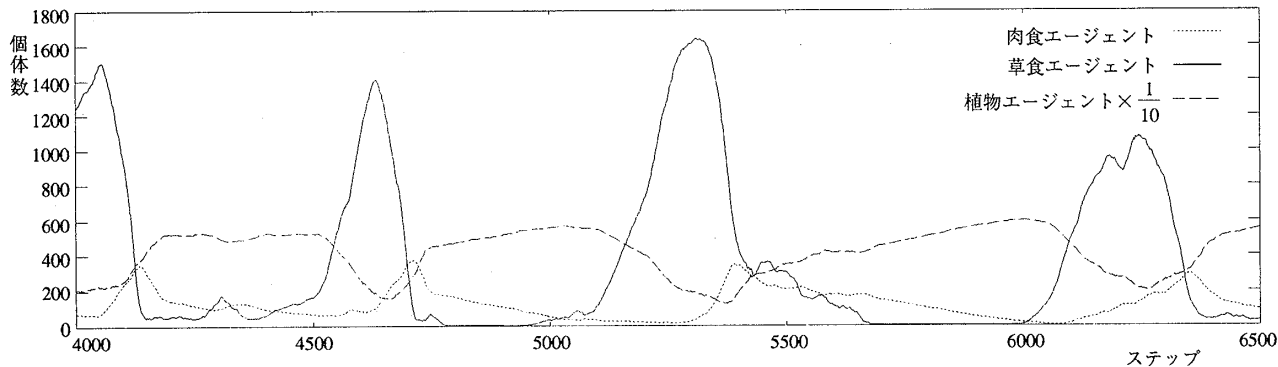


図10 食物連鎖時の種別個体数推移

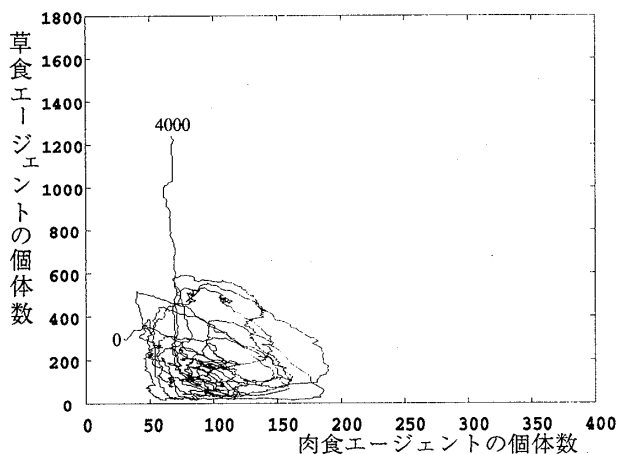


図11 0~4000 ステップの個体数推移

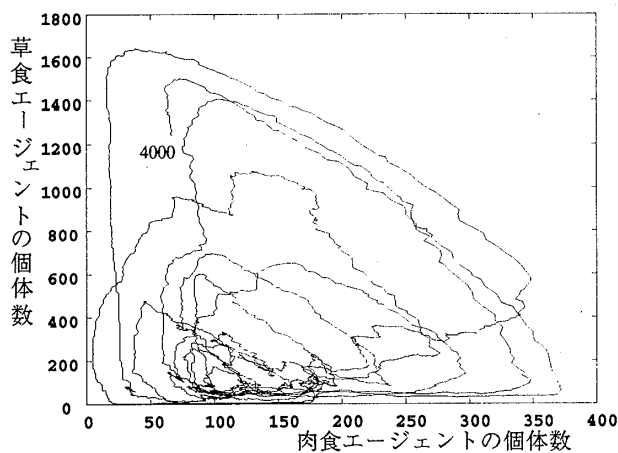


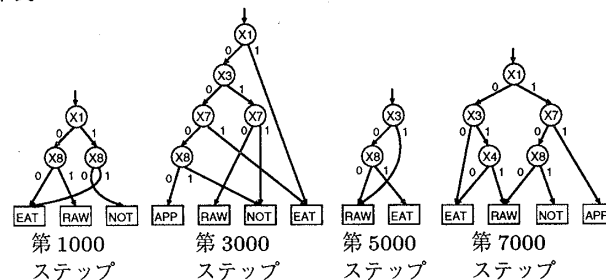
図12 4000~8000 ステップの個体数推移

結果と評価

捕食者-被食者系の個体群モデルを表す Lotka-Volterra 方程式 [Haberman 81] より食物連鎖関係が成立した生態系で横軸に捕食者数, 縦軸に被食者数をとったとき渦巻状のグラフが現れることがいえる.

図11と図12は進化の前と後の個体数推移を表しており, 進化した後ではより明確な渦が確認できる.

草食エージェント



肉食エージェント

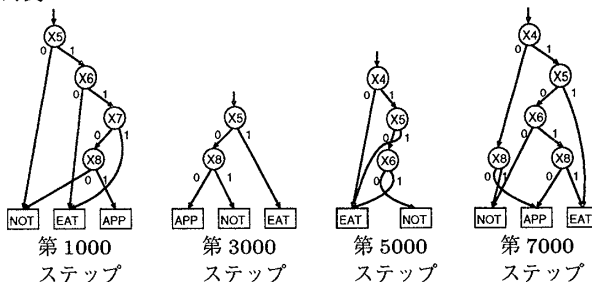


図13 複数モデルのBDDの推移(例)

図13に, 疑似生態系で進化した草食, 肉食の各エージェントの内, 第1000, 3000, 5000, 7000ステップの代表的BDDを示す. 各ステップの疑似生態系では, 多数のエージェントが存在する. 各エージェントは, 各々異なったBDDに従って行動をするが, その多くは, 疑似生態系で生き残った優秀な少数のエージェントの子孫にあたるため, 行動に類似性を持つ. ここで示す代表的BDDは, BDDへの入力(8ビット, 256パターン)に対応する行動の一致度に基づいて分類した場合の, 最大集団の中から選んだ, 平均的BDDである.

草食エージェント, 肉食エージェント共に, 第1000, 3000ステップまでは, 食べる行動を中心に行う個体が代表となっている. その後, 他のエージェントを意識する行動を獲得した個体が現れる. 草食の第5000ステップを代表する個体では, それまでと異なり, 逃げる行動を中心とし, 逃げる必要のない場合にのみ, 食べる

行動をとる。肉食エージェントでは、第5000, 7000ステップで、他の肉食エージェントに応じて行動していることが注目される。

5. 考 察

4・1節において n -BDD は有限状態オートマトンよりも、また変数順序を適当に与えることによりクラシファイア・システムよりも良好な結果を得た。草食エージェントが生き延びるためには、植物の探索や自分が空腹・満腹かよりも肉食エージェントに対する警戒を優先させることが必要になる。 n -BDD では、このように変数順位を重視すべき順にした場合に特に良好な結果を示している。これは n -BDD が、下位の変数の値に依らず上位の変数節点から定数節点に直接ジャンプできる階層的なグラフ構造を持っているためである。また、 n -BDD では節点毎に受理する入力変数が対応づけられている。節点やそこから伸びる枝に対して遺伝的操作を行っているので、これは入力変数の持つ意味(表1の環境条件)を考慮した関数の調整といえる。これに対し有限状態オートマトンではループや閉路が存在するため状態が受理する入力変数は必ずしも一意には決まらない。複数の入力変数を受理し得る状態やそこから伸びる矢印に対し遺伝的操作を行うことになるので、一度の遺伝的操作が与える関数の変化が大きくなってしまい、これが局所探索能力を下げている。大域的探索能力については n -BDD の根、有限状態オートマトンの初期状態に近い部分に遺伝的操作を行うことにより得られる。従って局所・大域の両方の探索能力を併せ持つ n -BDDの方が有限状態オートマトンよりも探索能力が優れている結果が現れたと結論づけられる。一方、クラシファイア・システムについては入力値に対し出力値が一意に決まらないため、例えば草食エージェントが肉食エージェントから逃げるときに runaway の出力を出し続けられないといった現象が生じる。この性質が適応度を下げていると考えられる。またクラシファイア・システムではいずれの入力変数順序の場合でも平均したグラフを示している。これはクラシファイア・システムがクラシファイアの条件部と入力変数のパターンマッチングを行うため、即ち各入力変数を平等に扱っていることが原因だと考えられる。4・1節の実験の様に入力変数に優先順位がある場合には各変数の重要度が異なる。その優先順位が既知であれば、その変数順位を適当に設定することにより、 n -BDD はクラシファイア・システムよりも効率のいい探索が可能になる。

4・2節においては速かな適応能力が要求される動的環境での実験に n -BDD を用い、こちらも良好な結果が得られた。図10は食物連鎖が現れたときの種別個体数推移のグラフである。フィールド上では以下の四つの局面が繰り返された。

- (1) 草食エージェントが植物エージェントを食べて次世代個体を生み、群れを形成して、さらに植物エージェントが残存する場所へ移動していく。
- (2) 肉食エージェントがこの草食エージェントの群れを発見・捕食し、次世代個体を生むことで群れを形成する。
- (3) 草食エージェントの個体数が減少し肉食エージェントの群れが餓死する。その場に植物エージェントが発生するため、植物エージェントの分布に偏りが現れる。
- (4) 植物エージェントの分布が偏っているので、生き残った少数の草食エージェントがこれに接触するまで、個体数の変動が小さい状態が続く。

図10で、肉食や草食エージェントに比べ植物エージェントのピークには幅が見られるのは局面(4)に起因する。また、局面の移行のタイミングがエージェントの行動戦略の進化の度合いのほか、植物エージェントの分布やエージェントの群れの規模にも依存するので個体数増減の周期は一定にはなっていない。

6. お わ り に

本稿では遺伝的アルゴリズムにおける遺伝的操作が効果的に作用するための表現方法として n -BDD を提案し、実験によりその有効性を確かめた。実験では他の遺伝子表現方法に比べ n -BDD が良好な結果を得たが、クラシファイア・システムが本稿の実験に不向きであったように、 n -BDD もあらゆる問題に対して最も有効な遺伝子表現方法ではなく、問題に適切な遺伝子表現方法を選択する必要がある。関数表現としての BDD の優秀性を論理式のためだけに限定せず多くの分野で有効に利用できるという意味でその応用範囲がさらに広がっていくことが期待される。

BDD を遺伝プログラミングに用いる研究としては、論理設計を目的とした H. Sakanishi 等の研究がある [Sakanishi 97]。[Sakanishi 97]の方法は、BDD 一般でなく、木構造のものに限定することで、交叉を導入している点が異なる。本研究では、BDD 一般、即ち、二つの枝が一つの節点に合流することを許すことにより、交差を用いずに探索の効率を得ている。

今後は n -BDD に既約化手法を適用し探索空間を限

定することについて検討したい。また、BDD 内の変数順序は探索の制御に役立つ一方、バイアスの固定にもなっている。適切な変数順序の発見も含んだ探索手法の開発は、今後に残されている。BDD は、本研究で示した通り、入力情報に対して行動を一つに決定する問題に適している。多くのマルチロボットシステムは、単純なエージェントの行動決定問題としてモデル化されており、これらへの応用は今後の課題である。

謝 辞

本研究の一部は文部省科学研究費補助金 (#10780227) および栢森情報科学振興財団研究助成金の支援を受けている。

◇ 参 考 文 献 ◇

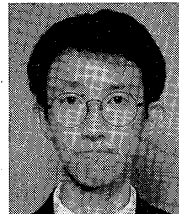
- [Akers 78] S. B. Akers: Binary Decision Diagrams, IEEE Trans. Comput., pp. 509-516, 1978.
- [Bryant 86] R. E. Bryant: Graph-Based Algorithms for Boolean Function Manipulation, IEEE Trans. Comput., pp.677-691, 1986.
- [Fogel 67] L. J. Fogel, A. J. Owens, and M. J. Walsh.: Artificial Intelligence Through Simulated Evolution, John Wiley & Sons, 1967.
- [Haberman 81] R. Haberman: 群個体成長の数学モデル, 現代数学社, 1981.
- [Holland 85] J. H. Holland: Properties of the Bucket Brigade, Proceedings of an International Conference on Genetic Algorithms and their Applications, 1-7, John J.Grefenstette (Ed.), Carnegie-Mellon University, Pittsburgh, 1985.
- [Holland 86] J. H. Holland: Escaping brittleness: The possibilities of general-purpose learning algorithms applied to parallel rule-based systems. In R. S. Michalski, J. G. Carbonell & T. M. Mitchell (Eds.), Machine learning: An artificial intelligence approach (Vol. 2), Los Altos, CA: Morgan Kaufmann, 1986.
- [伊庭 94] 伊庭斉志: 遺伝的アルゴリズムの基礎 -GA の謎を解く-, オーム社, 1994.
- [石浦 93] 石浦菜岐佐: BDD とは, 情報処理, Vol. 34, No. 5, 1993.
- [北野 93] 北野宏明: 遺伝的アルゴリズム, 産業図書, 1993.
- [湊 93] 湊 真一: 計算機上での BDD の処理技法, 情報処理, Vol. 34, No. 5, 1993.
- [森脇 96a] 森脇康介, 犬塚信博, 山田雅之, 世木博久, 伊藤英則: n 出力二分決定グラフを遺伝子とした GA による疑似生態系の最適化, 第 10 回人工知能学会全国大会, pp.601-604, 1996.
- [Moriwaki 96b] K. Moriwaki, N. Inuzuka, M. Yamada, K. Itoh, H. Seki and H. Itoh: Self adaptation of Agent's Behavior using GA with n -BDD, 5th IEEE International Workshop on Robot and Human Communication, Tsukuba Japan, Nov 11-14, pp.96-101, 1996.
- [森脇 97a] 森脇康介, 犬塚信博, 山田雅之, 世木博久, 伊藤英則: 進化的マルチエージェントによる生態系シミュレーション, 第 28 回人工知能基礎論研究会, pp.6-11, 1997.
- [Moriwaki 97b] K. Moriwaki, N. Inuzuka, M. Yamada, H. Seki, and H. Itoh: A Genetic Method for Evolutionary Agents in a Competitive Environment, Soft Computing in Engineering Design and Manufacturing,

P.K. Chawdhry, R. Roy and R.K. Pant (eds.), Springer-Verlag, 1997.

- [Sakanishi 97] H. Sakanishi, T. Higuchi, H. Iba and Y. Kakazu: Evolution of Binary Decision Diagrams for Digital Circuit Design Using Genetic Programming, Evolvable Systems: From Biology to Hardware, T. Higuchi M. Iwata and W. Liu (eds.), LNCS serie No.1259, Springer-Verlag, 1997.
- [Takashina 94] T. Takashina and S. Watanabe: Study of self adaptive behavior in quasi-ecosystem, in Proceedings of the Third Parallel Computing Workshop, Kawasaki Japan, November 1994.
- [Takashina 95] T. Takashina, K. Yoriki and S. Watanabe: Strategical Diversity and Self Adaptive Behavior in Quasi-Ecosystem, 並列人工知能研究会 (第 6 回), SIG-PPAI-9403, 1995.
- [渡部 93] 渡部悦穂, 久木元裕治: BDD の応用, 情報処理, Vol. 34, No. 5, 1993.
- [柳谷 93] 柳谷雅之: 組合せ最適化問題の BDD による解法, 情報処理, Vol. 34, No. 5, 1993.

[担当委員: 小林重信]

著 者 紹 介



森脇 康介(学生会員)

1996 年名古屋工業大学工学部知能情報システム学科卒業。1998 年同大学院工学研究科博士前期課程電気情報工学専攻修了。同年 NTT コミュニケーションウェア(株)入社。また、1998 年同大学院工学研究科博士後期課程電気情報工学専攻に進学, 在籍中。現在に至る。遺伝的アルゴリズム, 人工生命の研究に従事。情報処理学会会員。

<moriw@juno.ics.nitech.ac.jp>



横井 大祐

1995 年名古屋工業大学工学部知能情報システム学科入学。現在, 同学科在学中。遺伝的アルゴリズム, 人工生命に興味を持つ。

<dysk@juno.ics.nitech.ac.jp>



犬塚 信博(正会員)

1987 年名古屋工業大学工学部情報工学科卒業。1992 年同大学院工学研究科博士課程電気情報工学専攻修了。博士(工学)。同年, 同大学電気情報工学科助手。1993 年同大学知能情報システム学科助手。1998 年同学科講師。現在に至る。1994 年より 1996 年まで, 日本学術振興会海外特別研究員として英国インペリアルカレッジへ出張。人工知能, 特に機械学習, 知識表現に関する研究に従事。情報処理学会, 電子情報通信学会各会員。 <inuzuka@ics.nitech.ac.jp>



伊藤 英則(正会員)

1974 年名古屋大学大学院工学研究科博士課程電気・電子専攻満了。工学博士号取得。同年日本電信電話公社入社, 横須賀研究所勤務。1985 年(財)新世代コンピュータ技術開発機構に出向。1989 年より名古屋工業大学教授。知能情報システム学科。これまでに数理言語理論とオートマトン, 計算機ネットワーク通信 OS, 知識ベースシステム, 人工知能などの研究と開発に従事。電子情報通信学会, 日本ファジイ学会, 情報処理学会各会員。 <itoh@ics.nitech.ac.jp>