

論文 / 著書情報
Article / Book Information

題目(和文)	進化的手法によるマルチエージェントの協調行動の創発に関する研究
Title(English)	
著者(和文)	原章
Author(English)	
出典(和文)	学位:博士(工学), 学位授与機関:東京工業大学, 報告番号:甲第5163号, 授与年月日:2002年3月26日, 学位の種別:課程博士, 審査員:
Citation(English)	Degree:Doctor (Engineering), Conferring organization: Tokyo Institute of Technology, Report number:甲第5163号, Conferred date:2002/3/26, Degree Type:Course doctor, Examiner:
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

進化的手法によるマルチエージェントの
協調行動の創発に関する研究
(博士論文)

東京工業大学
大学院総合理工学研究科
物理情報システム創造専攻
99D24107 原 章

指導教官：長橋宏 教授
外部指導教官：長尾智晴 教授

目次

第1章 序論	4
1.1 背景と目的	4
1.2 本論文の構成	5
第2章 進化的手法を用いたマルチエージェントの行動制御に関する従来研究	6
2.1 はじめに	6
2.2 遺伝的アルゴリズム (GA)	6
2.2.1 遺伝操作	7
2.2.2 GA の応用例	9
2.3 遺伝的プログラミング (GP)	11
2.4 GP によるマルチエージェントの行動制御	12
2.4.1 GP の個体とエージェントの対応関係	12
2.4.2 Homogeneous モデル	13
2.4.3 Heterogeneous モデル	13
2.4.4 従来手法の問題点とグループ構造を利用したモデル化	15
2.5 おわりに	17
第3章 自動グループ構成手法 (ADG; Automatically Defined Groups)	18
3.1 はじめに	18
3.2 提案手法 (ADG) の定義	18
3.2.1 ADG における GP の個体構造	18
3.2.2 交叉方法	19
3.2.3 グループ突然変異	22
3.2.4 選択	23
3.3 おわりに	23
第4章 マルチエージェントによる荷物搬送問題への ADG の適用	25
4.1 はじめに	25
4.2 荷物搬送問題の環境設定	25
4.3 実験結果と考察	26
4.3.1 ADG による個体集団の進化と獲得されたグループ構造	26
4.3.2 従来法 (Homogeneous/Heterogeneous モデル) との比較	29

4.4	エージェント数や必要な役割分担数が増加した場合の検証	32
4.4.1	実験環境の設定	32
4.4.2	実験結果と考察	33
4.5	おわりに	33
第5章	タイルワールド環境におけるマルチエージェント制御への ADG の適用	37
5.1	はじめに	37
5.2	タイルワールドの環境設定	37
5.3	実験結果と考察	41
5.3.1	ADG により獲得されたグループ構造と行動の様子	41
5.3.2	従来法 (Homogeneous/Heterogeneous モデル) との比較	41
5.4	おわりに	43
第6章	動的環境への適応のためのグループ構造の変更方法の検討	45
6.1	はじめに	45
6.2	グループ移動記号によるメンバの移動による動的環境への適応	45
6.2.1	グループ移動記号の設定	45
6.2.2	動的環境の設定	47
6.2.3	実験結果と考察	48
6.2.4	グループ移動記号に関するまとめ	50
6.3	内部状態ごとの行動最適化による動的環境への適応	50
6.3.1	内部状態を考慮したエージェントモデル	50
6.3.2	動的環境の設定	50
6.3.3	実験結果と考察	52
6.3.4	内部状態ごとの行動最適化に関するまとめ	53
6.4	おわりに	53
第7章	ADG を用いた人工株式市場の構築と解析	55
7.1	はじめに	55
7.2	人工市場研究の背景	55
7.3	本研究における市場研究へのアプローチ	56
7.4	本研究における人工株式市場の概要	58
7.4.1	GP を用いた戦略形成	58
7.4.2	市場の構成	58
7.4.3	投資家群のモデル化	59
7.5	人工株式市場の構築	60
7.5.1	人工市場の現実性の評価方法	60
7.5.2	GP の終端・非終端記号の設定	61
7.6	実験結果と考察	62

7.6.1	獲得した人工市場とその株価変動	62
7.6.2	人工市場におけるグループ構造の解析	63
7.6.3	各グループの役割の検証	67
7.6.4	株価の大変動の発生メカニズムの解析	69
7.7	おわりに	71
第8章	結論	74
8.1	まとめ	74
8.1.1	ADG の提案について	74
8.1.2	マルチエージェント問題への適用による ADG の有効性の検証について	75
8.1.3	グループ構造の変更方式の検討について	75
8.1.4	人工株式市場の構築について	76
8.2	今後の課題	77
	謝辞	78
	参考文献	79
	付録	84
付録 A	タイルワールドでの最良個体の木構造	84
付録 B	タイルワールドでの最良個体の行動の様子	86
付録 C	人工市場における株価暴落時の取引の様子	87
付録 D	本研究に関する発表文献	91

第1章 序論

1.1 背景と目的

近年、多数のエージェントやロボットによる問題解決が注目を集めている [1, 2, 3, 4, 5]. これは複数のエージェントの協調により問題を効率的に解決することができるためである. 具体的には, 単一のエージェントには能力の限界があってもそれらの協調により処理能力を向上できること, エージェント群の並行処理による処理効率の向上, エージェント群の一部が故障しても問題を解決できるという耐故障性の高さなどが理由として挙げられる.

複数のエージェントの協調動作や動的な環境下での作業のためには, エージェント同士や環境との相互作用を行うプログラムを記述する必要がある. しかし, 様々な状況が想定され, これを人手で書くことは容易なことではない. そのため, エージェントが自律的に行動し環境に適応するマルチエージェントシステムの実現が重要であり, 進化的な手法が有効であるとされている [6].

自律ロボットのようなエージェントの行動制御プログラムは, エージェントの基本行動要素を組み合わせた木構造で表現できる. そこで本論文では, 進化的計算法の一つで, 木構造を最適化する手法である遺伝的プログラミング (Genetic Programming; GP) [7] を用いて, マルチエージェントの行動制御を行う.

GP を用いたマルチエージェントの行動制御は, これまでも盛んに行われているが, 従来研究では, 同一の行動規則を持つ均質なエージェント群の相互作用によって, 全体としてより高いレベルの秩序ある群行動を創発し, 問題を解決する試みが数多くなされている [7, 8]. しかし複雑なタスクの処理には, 各自が役割に応じた異なった行動をとり, チーム内で分業を行なうといった協調行動が不可欠である [9]. そのため本論文では特に, 不均質なエージェント群の生成と解析を扱う. 集団内の行動規則の多様化は, システムの高機能化につながるとともに, 異種ロボット間やロボットと人間との協調システムの実現, 多様な行動主体が存在する現実社会への応用などに重要である.

そこで本論文では, 不均質な大規模エージェント群を効率的に生成するため, 集団のグループ構造に着目した新たな手法である自動グループ構成手法 (ADG; Automatically Defined Groups) を提案する. この手法は, 進化の過程で, 問題解決に必要なグループ数, 各グループへのエージェントの割り当て, 各グループの行動制御プログラムを自動的に最適化するものである.

この手法を, ロボット群による荷物搬送問題へ適用し, 従来手法と比較するこ

とにより有効性を検証する。また現実社会への応用として、株式市場を取り上げ、市場モデルを構築することにより、株価変動メカニズムの解析を行う。

1.2 本論文の構成

本論文の構成は以下のとおりである。2章では、進化的計算法の概要、中でも特に遺伝的アルゴリズムと遺伝的プログラミングについて述べ、GPによるマルチエージェントの協調行動の創発に関する従来研究について紹介する。3章では、提案手法ADGについて説明する。そして4章では、エージェントの能力に応じて異なる行動が必要とされるマルチエージェントによる荷物搬送問題を設定し、ADGを適用した結果とその考察について述べる。また5章では、問題解決に群行動が要求される、拡張したタイルワールド環境を設定し、ADGを適用した結果とその考察について述べる。次に6章では、動的環境への適応を目指して、グループ構造を変更しながら問題解決にあたるシステムの実現方法の検討を行う。さらに7章では、実社会における大規模マルチエージェントシステムである株式市場を取り上げ、提案手法を利用した株式市場のモデル化と株価変動メカニズムの解析を行う。最後に7章で、結論と今後の課題について述べる。

第2章 進化的手法を用いたマルチ エージェントの行動制御に関 する従来研究

2.1 はじめに

本章では、まず、進化的計算法の一種である遺伝的アルゴリズムについて述べる。そして、木構造プログラムを扱えるように遺伝的アルゴリズムを改良した手法である遺伝的プログラミングについて説明する。次に、遺伝的プログラミングを用いた多数のエージェントの行動制御に関する従来研究について述べる。

2.2 遺伝的アルゴリズム (GA)

遺伝的アルゴリズム (Genetic Algorithm; GA) は生物の進化の過程から着想された探索手法であり、1975年に Holland [10] により提唱された。この手法は、問題の解を表す個体集団に対して遺伝操作を繰り返すことによって与えられた問題に適した解を作り出そうとするものである [10, 11, 12, 13, 14, 15]。

解を表す個体は、基本的に固定長の文字列によって1次元的に表現される。これは生物の染色体に相当する。各世代は個体の集合により構成される。遺伝的アルゴリズムの処理の流れを図 2.1 に示す。

初期集団の生成では、一般的には決められた個体数の染色体をランダムに生成する。

初期集団が生成されると各々の個体に対して適応度の評価を行なう。適応度はその個体が表す候補解が与えられた問題に対してどの程度適しているかを表すもので、個体の持つ染色体と問題の環境から計算される。この適応度の値が大きいほど優れているとする場合が多い。

この適応度の値を基に遺伝操作が行われる。遺伝的アルゴリズムの基本的な遺伝操作には次の3つがある。

- 選択 (selection)
- 交叉 (crossover)

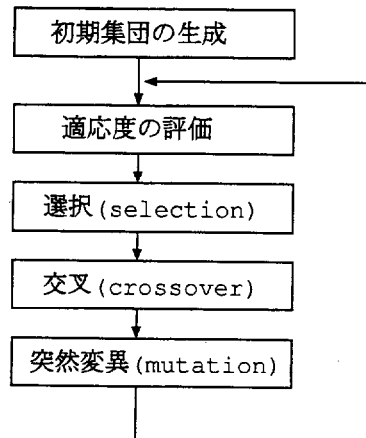


図 2.1: 遺伝的アルゴリズムの処理の流れ

- 突然変異 (mutation)

まず選択により，次世代に残す個体を選ばれる．基本的に，適応度の高い個体が生き残りより多くの子孫を残す機構となる．逆に適応度の低い個体は，環境にうまく適応していない個体とみなされ消滅させられる．これは自然淘汰の原理を反映したものである．

選ばれた個体群から，交叉率と呼ばれるある確率で2つの染色体の組みを選び出し，交叉と呼ばれる操作を行う．交叉では2つの染色体間で部分的に遺伝子を組み替えて新しい個体を発生させる．

次に各々の個体に対して突然変異を行なう．これは突然変異率と呼ばれるある確率で染色体の1部の値を変化させる操作である．

遺伝操作には様々な方式があるが，これは次節で説明する．これらの操作が終了すると，新しい世代の個体集団が作られたことになる．そしてこの新たな集団に対して，また適応度の評価，選択，交叉，突然変異を行い，さらに新たな世代を作っていく．この進化の過程がうまくいけば世代が進むにつれて適応度の向上がみられ，得られた個体はその問題に対して高度に進化した解となる．この処理を適応度がある値よりも大きくなった時や決められた世代数が経過したなどの終了条件を満たすまで繰り返す．

2.2.1 遺伝操作

選択

遺伝的アルゴリズムでは新しい世代の個体群を作り出す際，適応度を基にして個体を選択する．このときに集団から個体を選択する方法の1つにトーナメント方式がある [12, 15]．

トーナメント方式は元の集団の中から決められた数の個体(トーナメントサイズという)をランダムに選択し、その中で最も適応度の高い個体を選択する。この過程を決められた集団数が得られるまで繰り返すという方法である。

その他の選択の手法としては、適応度に比例した確率で個体の選択を行なうルーレット方式、各個体の適応度の順位で個体の選択回数を決めるランク方式などがあるが、トーナメント方式は

- トーナメントサイズを変更することによって、各個体の選択のされ易さを変更できる。
- 選択のされ易さが適応度の上下関係のみに依存し適応度の値とは無関係なので、各個体の適応度が接近している場合や適応度が負の値を取る時にも適用できる。

といった特徴があるので他の手法と比較して優れていると言われる [12]。

交叉

交叉は2つの個体の染色体をランダムな位置で部分的に交換し新しい個体を2つ作る。交叉はある一定の確率で起こり、これを交叉率と呼ぶ。交換する部分の選択の仕方によって一点交叉、複数点交叉、一様交叉などの種類がある。一点交叉、一様交叉の例を図 2.2 に示す。

一点交叉はランダムに決めた交叉点を境として2つの個体の染色体を入れ換える。一様交叉ではマスクと呼ばれる0と1からなるランダムなビット列を用意し、ビットが1のところでは染色体を入れ換える。

交叉が行なわれることにより、新しい個体が発生し進化が生じる。

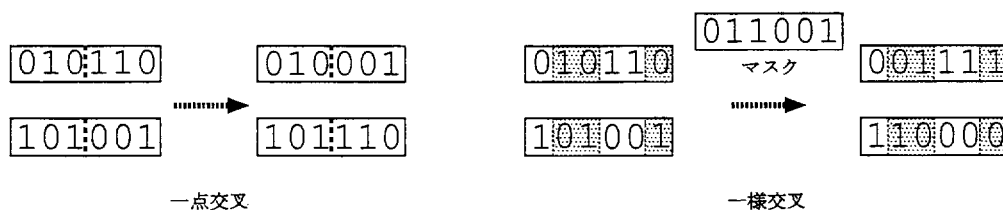


図 2.2: 交叉

突然変異

突然変異では染色体の遺伝子に相当する各ビットを一定の確率で変化させる操作である。この突然変異を起こす確率を突然変異率と呼ぶ。0と1からなる染色体

の各ビットを突然変異率の生起確率で、0を1、あるいは1を0に変更する。突然変異の例を図2.3に示す。

この突然変異の操作によって交叉だけでは生じない遺伝子を持つ個体が生成できる。また探索が局所解に陥ることを防ぐことができる。

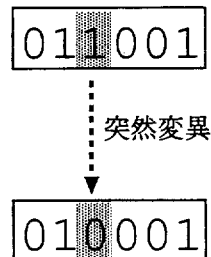


図 2.3: 突然変異

2.2.2 GA の応用例

GA は組合せ最適化をはじめ、さまざまな領域の問題に適用が可能である。本節では GA の応用例の一つとして、画像中からの顔領域の抽出に適用した例について述べる。

人物の顔が画像中のどこにあるか分からないような画像を原画像とし、その画像中の人物の顔を抽出する処理は、個人識別の前処理、人物の計数、写真焼付けの際の露光の自動設定などのために重要である。文献[16]では、位置、大きさ、画像に対して鉛直な軸に関する回転角度が不明な、任意方向を向いた人物の顔の候補領域をモノクロ静止階調画像から検出する手法を提案している。

この手法は、あらかじめ人物から作成した図2.4に示したさまざまな角度を向いた顔のテンプレートを図2.5のように、原画像上にさまざまな位置、大きさ、回転角度で重ねて比較するパターンマッチングに基づいている。この方法ではテンプレートを原画像に重ねる際の位置、大きさ、回転角度のパラメータの組合せの場

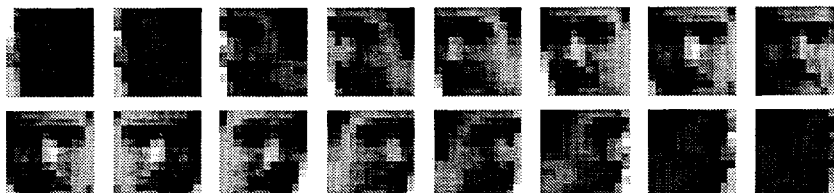


図 2.4: 顔抽出に用いるテンプレート画像

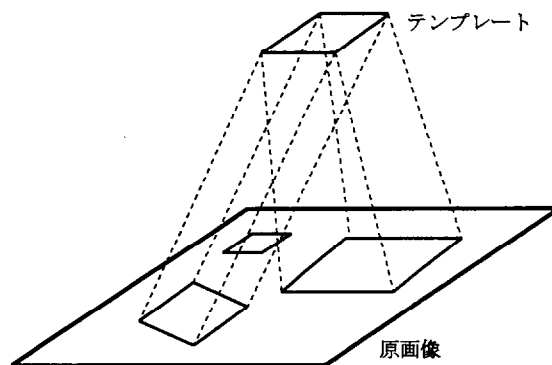


図 2.5: テンプレートマッチングの概念図

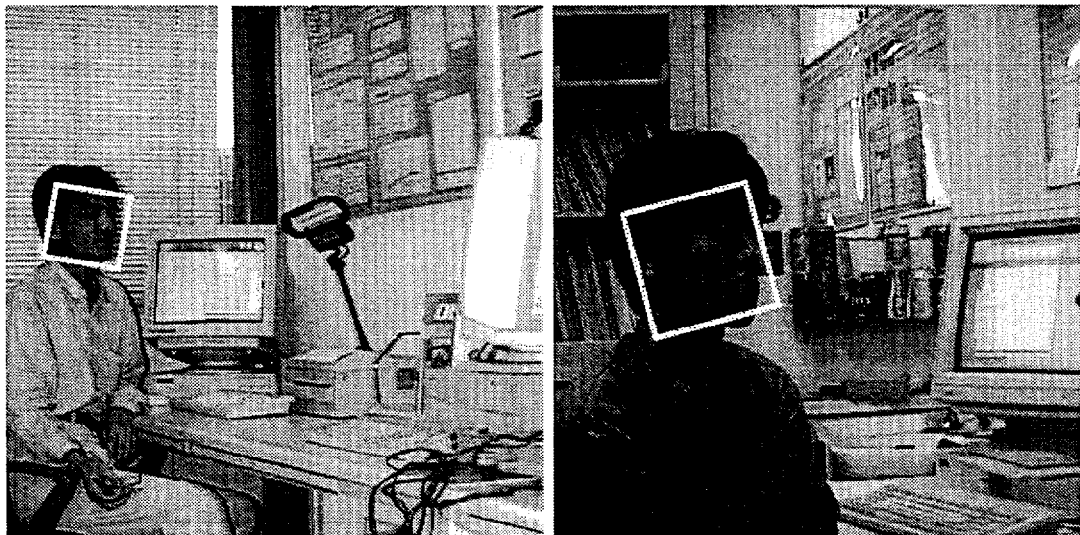


図 2.6: GA による顔領域の抽出結果

合の数が膨大になるため、全探索することは実用上不可能である。そこで GA を用いてパラメータの最適化を行う。すなわちテンプレートを原画像上に重ねる際のパラメータの列を染色体とし、そのときのテンプレートと原画像とのマッチング率を適応度として個体集団を進化させる。これによってマッチングの回数が膨大になることを避けることができる。

この手法は、図 2.6 のように、原画像中の顔の向き、位置、大きさ、回転角度および背景が変化した場合でも、人物の顔の領域が良好に抽出できる。また、GA の方がランダムサーチに比べて、探索速度および探索精度ともに効率良く顔の候補領域を求めることができることを確認している。

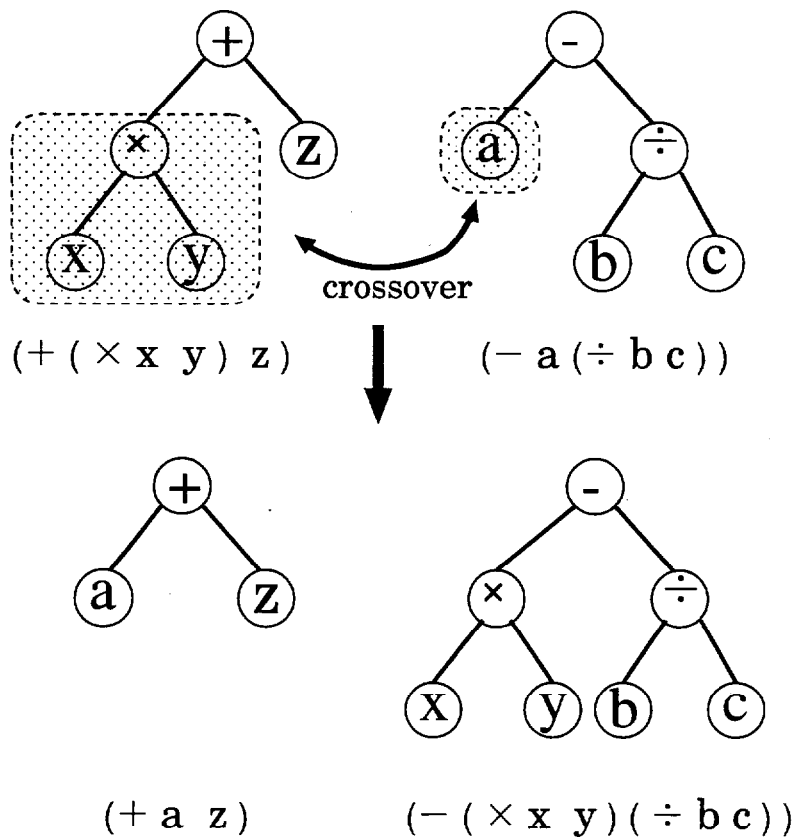


図 2.7: GP における交叉の例

2.3 遺伝的プログラミング (GP)

遺伝的プログラミング (Genetic Programming; GP) は, 1992 年に John Koza [7] により提案された. GA が主に一次元のビット列として表された問題の解を最適化する手法であるのに対して, GP は木構造のような構造的な表現を扱うことができるように拡張した手法である [7, 17, 18, 14, 19]. GP では木構造を表す個体に対して, GA と同様に交叉や突然変異を行ない世代交代を繰り返すことで, より問題に適した木構造を持つ個体を獲得する. GP による交叉の例を図 2.7 に, 突然変異の例を図 2.8 に示す.

複雑な数式や Lisp の S 式なども木構造で表現できるため, 木構造を直接扱うことのできる GP は, 最適値探索や組合せ最適化だけでなく, 関数同定問題や画像処理 [20] などさまざまな分野への適用が可能である. 木構造はエージェントの行動規則のようなプログラムを記述することにも適していることから, GP はマルチエージェントの分野にも極めて有用であると考えられる [21].

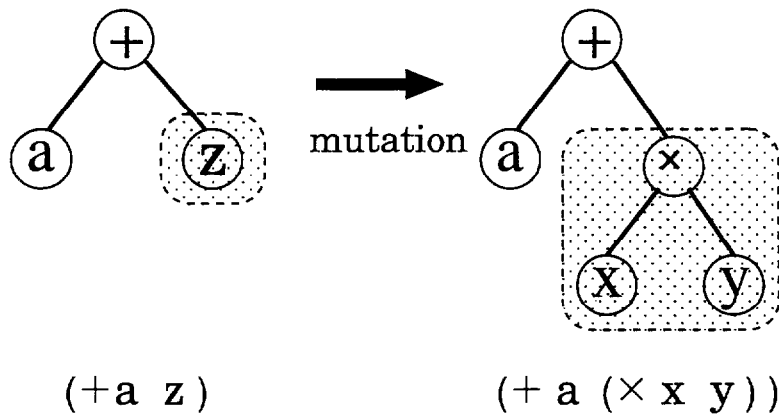


図 2.8: GP における突然変異の例

2.4 GP によるマルチエージェントの行動制御

2.4.1 GP の個体とエージェントの対応関係

GP によりマルチエージェントの行動規則を獲得する手法として、大きく分けて次の 2 通りの方法が考えられる。

手法 1 各エージェントを GP の一個体とし、各々のエージェントの行動を別々に評価し共進化などにより集団全体のパフォーマンスを向上させる手法

手法 2 エージェント集団を GP の一個体として考え、集団全体のパフォーマンスをもとに、進化により効率的なエージェント集団を獲得する手法

エージェント集団が役割分担をしながら協調して問題解決を行なう場合、手法 1 では他のエージェントを助ける行動を取るような直接的に集団のパフォーマンスに反映しないエージェントの評価が難しい。例えば複数のエージェントによる荷物搬送問題において、あるエージェントが他のエージェントの進行方向にある障害物を排除するような動作を行なった場合でも、その動作が効果的であったかどうかを直接評価することは困難である。しかし手法 2 では集団のパフォーマンスにより集団に属するエージェントを等しく評価するため評価が容易である。先の荷物搬送問題の例では、一定時間内に運んだ荷物の数などにより集団内の全てのエージェントを等しく評価すれば良い。集団のパフォーマンスに直接貢献していないエージェントの評価も可能となるため、役割分担などの複雑な協調動作の実現も可能になると考えられる。特に手法 2 では集団を一つの個体として考える際に、集団中の全てのエージェントが同一の行動規則を持つ Homogeneous モデルや、集団中の各エージェントが別々の行動規則を持つ Heterogeneous モデルなどが提案され性能の評価実験などが行なわれている [22, 23, 24]。

2.4.2 Homogeneous モデル

環境内の全てのエージェントが同一の行動規則にしたがって行動するとき、この集団を Homogeneous 集団と呼ぶ。図 2.9 のように、GP においては各エージェントは同じ木を参照する。各エージェントは同一の行動規則に従うが、各々の置かれた位置や状態に応じて木の出力が変化し、さまざまな行動を取ることが可能であるため、協調して問題を解決することが可能である [8]。

Homogeneous モデルでは、環境内のエージェントの数が増加しても、GP の各個体は一つの木を保持するだけですみ、GP による木構造の探索領域が増大することを抑えることができる。

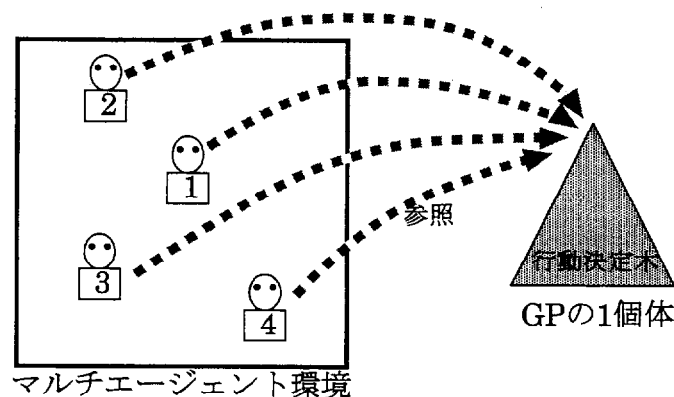


図 2.9: Homogeneous モデル

2.4.3 Heterogeneous モデル

環境内の各エージェントが別々の行動規則にしたがって行動するとき、この集団を Heterogeneous 集団と呼ぶ。図 2.10 のように、GP においては一個体中に各エージェントが参照する複数の木を保持することになる。このため Heterogeneous モデルでは交叉の方法として、図 2.11 に示すように、対応する木同士でのみ交叉を行うように制限する Restricted Breeding や、任意の木同士での交叉を許す Free Breeding など様々な手法が提案されている [22, 23, 25, 26]。しかし GP の染色体である木構造を別々のプールに分解することで多様性を維持しエージェントの役割分担の発生を高めるという観点から、交叉を対応する木同士に制限する Restricted Breeding が良いとされている [22]。

Luke らは Serengeti World と呼ばれる追跡問題において Homogeneous モデルと Heterogeneous モデルとの性能の比較実験を行ない、エージェントの能力が高く、複雑な協調動作が必要となるにつれて Heterogeneous モデルの方が性能が良くなることを示した [22]。

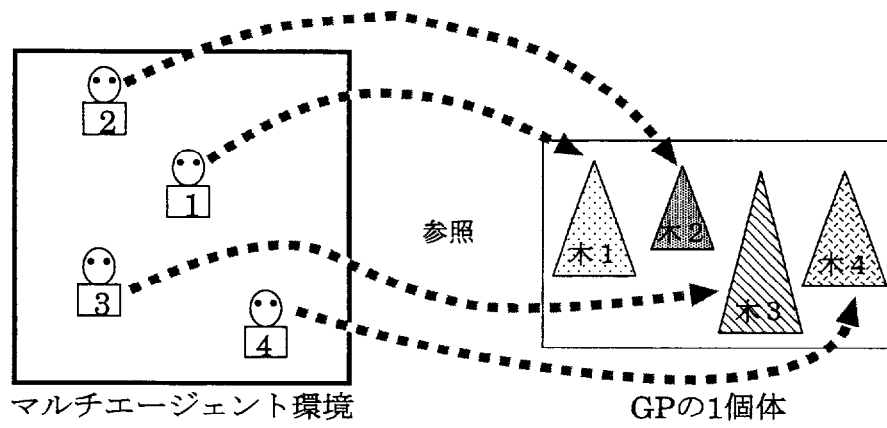


図 2.10: Heterogeneous モデル

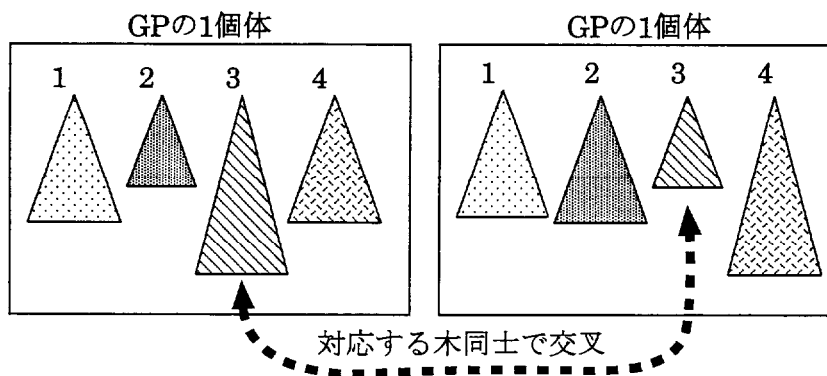


図 2.11: Restricted Breeding

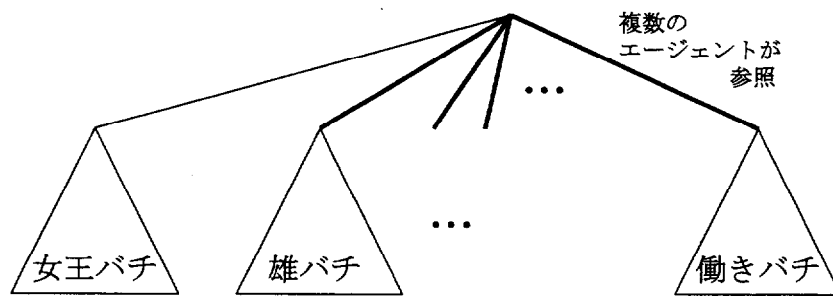


図 2.12: 蜂の階級

2.4.4 従来手法の問題点とグループ構造を利用したモデル化

Heterogeneous モデルでは、環境内のエージェントが各々の役割に応じて特殊化され、Homogeneous モデルに比べて、より複雑な協調動作を実現できると考えられる。しかし GP の一個体がエージェントの数だけ木構造を保持する必要があるため、環境内のエージェント数が増加すると、それに応じて木構造の探索領域が膨大になり GP による探索効率が悪化してしまう。実際にこれまでに行なわれている GP によるマルチエージェントの協調行動の創発に関する研究においても、探索効率の悪化などの要因から、Heterogeneous モデルの適用はエージェント数が 2~5 といった比較的小さな数の環境に留まっているのが現状である [22, 27, 23, 24, 28, 29]。

一方、自然界の蜂や蟻などは、数百、数千という個体から構成される大規模なマルチエージェントシステムとみなすことができる。蜂や蟻のような社会性昆虫は生存に必要な役割分担数に応じていくつかの階級にグループ分けされ、各自が所属する階級に応じた仕事を行ない、全体として高度なシステムを実現している。例えば一つの巣の中の何百という働きバチは同じ行動規則に従って行動していると考えられる。この集団を GP の個体として表現すると図 2.12 のように、全体としては Heterogeneous モデルのように複数の木構造を持つが、各々の木は Homogeneous モデルのように複数のエージェントが参照するといった、2つのモデルが混在した形となる。この構造を GP に応用すれば探索領域が膨大になることを抑え、複雑な協調行動を実現できると考えられる。

このようなグループ化の考えを応用した研究としては、文献 [30, 31, 32, 33, 34] が挙げられる。文献 [30] では、RoboCup におけるサッカーチームの行動最適化を扱っている。11 のエージェントをあらかじめ 6 つのグループに分けた疑似 Heterogeneous モデルを用いて、効率的に行動最適化を行った。また、文献 [31, 32] では、実時間生産スケジューリング問題に GP を適用した。各工程を担当する機械をエージェントととらえたマルチエージェントシステムにおいて、作業の性質上、ボトルネックにあたる機械とそうでない機械とにあらかじめグループ分けし、機械 1 台 1 台を扱うよりも効率的に最適化を行った。この実験では、Homogeneous モデルよりもグループ化モデルの方が高い性能をあげたことを確認している。

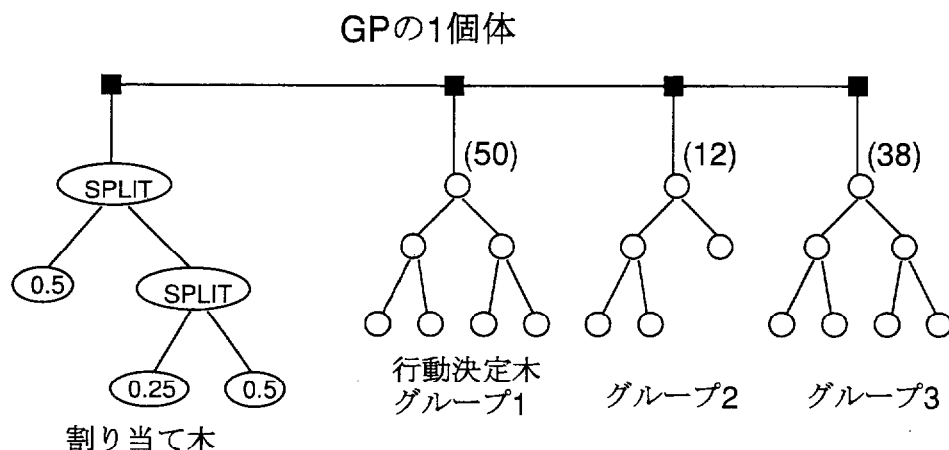


図 2.13: Legion system における個体の構造

このような従来研究からも分かるように、多数のエージェント群をグループ化の考えを用いて制御することは重要である。しかし、一般的に、これから解決すべき未知の問題において、あらかじめ問題解決に必要な役割分担数を知ることは困難である。汎用的な手法の実現のためには、問題に応じた適切なグループ構造が自律的に獲得できることが必要とされる。

この問題に取り組んだ研究として、Legion system [35, 36] がある。この手法では、各グループの行動制御プログラムとともに、各々のグループにどの程度の数のエージェントを割り当てるかを計算するプログラムを用意し、同時に最適化を行う。Legion system における GP の 1 個体の構造を図 2.13 に示す。v エージェント割り当て木では、木のルートから順に深さ優先探索で評価され、SPLIT ノードでは、その第 1 引数の値の割合で、エージェント群が分割される。エージェントの総数が 100 の場合、図 2.13 では以下のように分割される。

第 1 グループ 総数 $100 * 0.5 = 50$ (エージェント)

第 2 グループ 残り $50 * 0.25 = 12$ (エージェント)

第 3 グループ 残り 38(エージェント)

この手法により、各グループへ適切な数のエージェントを分配できると考えられる。

ただし、エージェントに能力差がある場合や各自の位置関係などが重要な問題の場合には、どれだけの数を割り当てるかという各グループの比率だけではなく、能力など個々の特性を考慮したエージェントの分配も必要である。すなわち、エージェント群のグループ化には、

- いくつかのグループが必要か。
- 各グループにどのエージェントを割り当てるか。

- 各グループの行動制御プログラム。

の3点が重要であり、このような点を考慮したグループ構造の自動獲得は従来手法では困難である。

2.5 おわりに

本章では、まず、進化的計算法の一種である遺伝的アルゴリズム、および遺伝的プログラミングの概要について述べた。

特に、遺伝的プログラミングは、アルゴリズムが未知の場合が多いエージェントの行動制御用プログラムの自動生成に有効である。ただし、マルチエージェントの行動制御にGPを適用する際には、従来使用されてきたHomogeneousモデル、Heterogeneousモデルには各々問題がある。GPの探索効率と得られるエージェント群の協調行動の複雑さにはトレードオフの関係があるため、大規模で、かつ役割分担といった複雑な協調動作を行うエージェント群の生成は困難であった。

この問題の解決策として、グループ構造を利用した研究も行われ有効性が検証されているが、自律的なグループ構造の獲得は実現できていない。

そこで次章では、エージェント群の協調行動に重要な

- 問題解決に必要なグループ数(役割分担数)
- 各グループへのエージェントの振り分け
- 各グループの行動制御プログラム

の3点を、進化の過程で自動的に獲得する手法である自動グループ構成手法ADGを提案する。

第3章 自動グループ構成手法 (ADG; Automatically Defined Groups)

3.1 はじめに

本章では、多数のエージェント群の行動制御のために我々が提案した、自動グループ構成手法 (ADG; Automatically Defined Groups) の具体的なアルゴリズムについて述べる。前章で述べた従来研究からも分かるように、多数のエージェント群をグループ化の考えを用いて制御することは重要である。しかし、一般的に、これから解決すべき未知の問題において、あらかじめ問題解決に必要な役割分担数を知ることは困難である。ADGは、エージェント群をいくつかのグループに分けるべきか、各グループにはどのエージェントを割り当てるべきか、各グループの行動制御プログラムはどのようなものにすべきか、という3点を、進化の過程で自動的に獲得する手法である。ADGの概念図を図3.1に示す。以下で本手法におけるGPの個体構造や遺伝操作について詳しく述べる。

3.2 提案手法 (ADG) の定義

3.2.1 ADGにおけるGPの個体構造

ADGにおけるGPの1個体の構造を図3.2に示す。この図において、三角形は木構造プログラムを表し、その上部に付けられた数字は、その木構造プログラムを参照して行動決定を行うエージェントの番号を表している。実線で結びつけられた木構造プログラムの一まとまりが、GPにおける1個体に相当する。なお、本章では、簡単のため、GPの1個体が4つのエージェントからなる場合の例を用いて、交叉等の具体的な処理を説明する。

図3.2に示すように、ADGでは行動決定のために同一の木を参照するエージェントの集合をグループと定義する。GPの各個体はいくつかの木構造を持ち、それぞれの木は異なるグループの特別なプログラムとして機能する。

また、GPの初期個体集団には、エージェント群をランダムにグループ分けした個体を用いることとする。

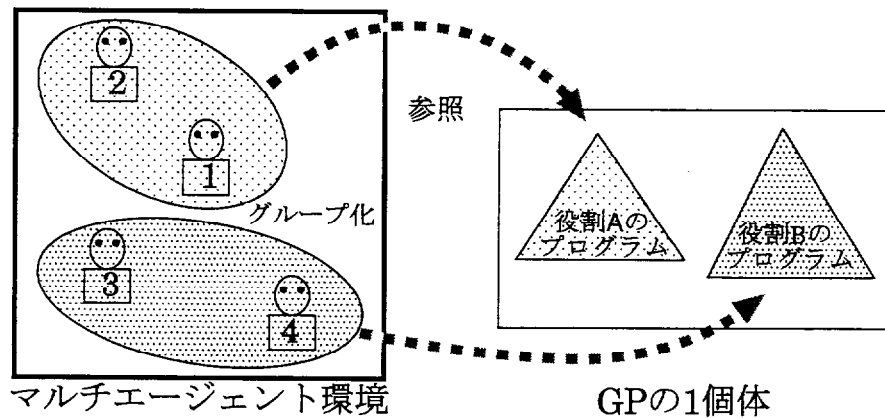


図 3.1: ADG の概念図

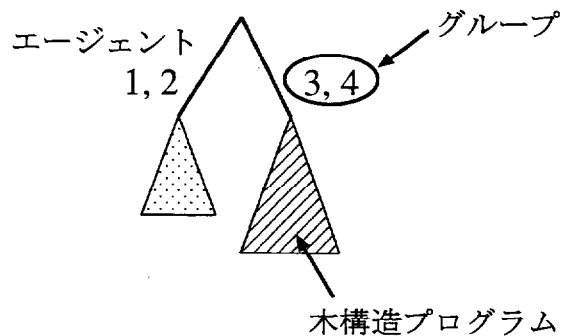


図 3.2: ADG における 1 個体の構造

以下で、ADG における遺伝操作、選択の方法について述べる。

3.2.2 交叉方法

ある個体のエージェント 1 が参照する木は他の個体のエージェント 1 が参照している木とのみ交叉を行なうといったように、基本的には Restricted Breeding と同様に対応する木の間で交叉を行なう。具体的な処理手順を以下に示す。

1. GP の個体集団から任意に選んだ 2 個体に対し、エージェント集合から任意に 1 つのエージェントを選び、そのエージェントが参照する木を交叉に使用する木とする。
2. 選択された木 T, T' を参照しているエージェントの集合 $A(T), A(T')$ を各々求める。

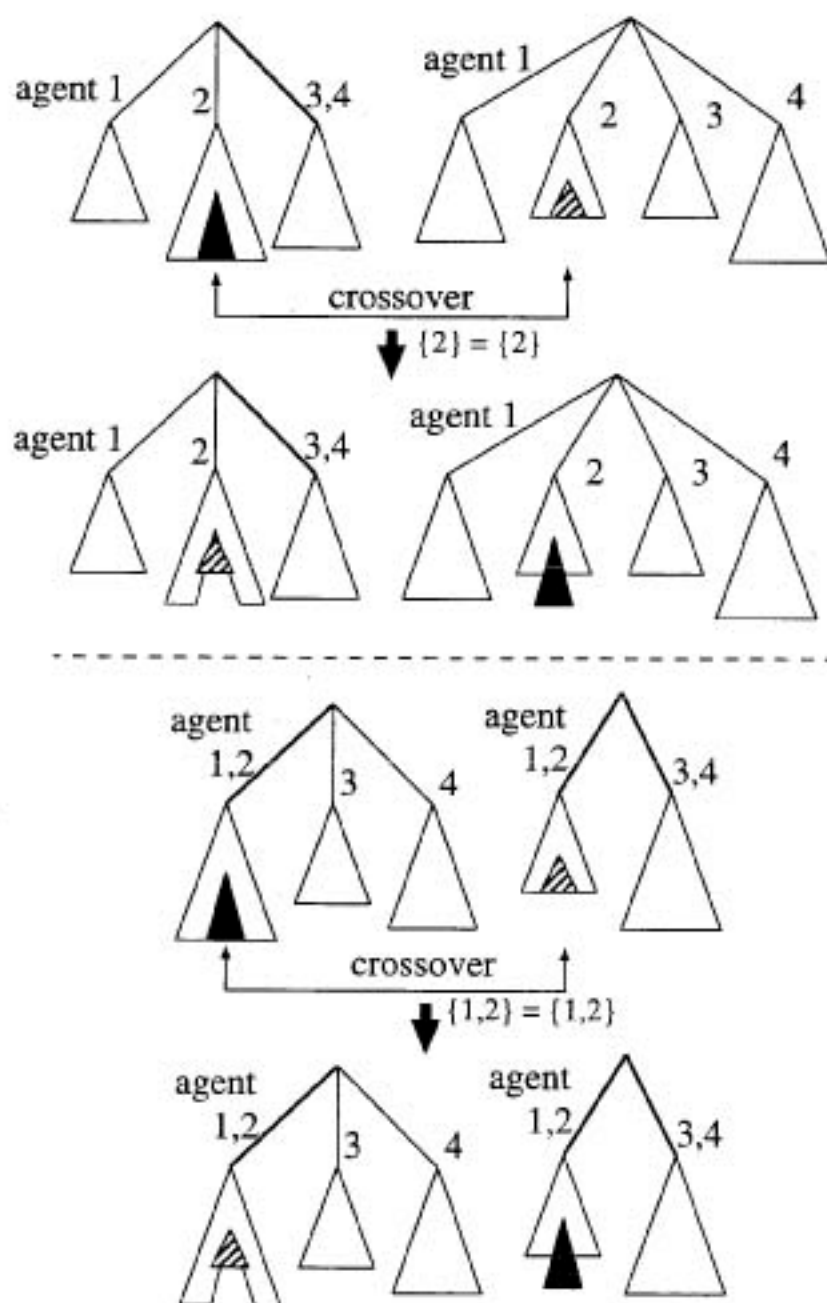


図 3.3: 交叉の例 (type a)

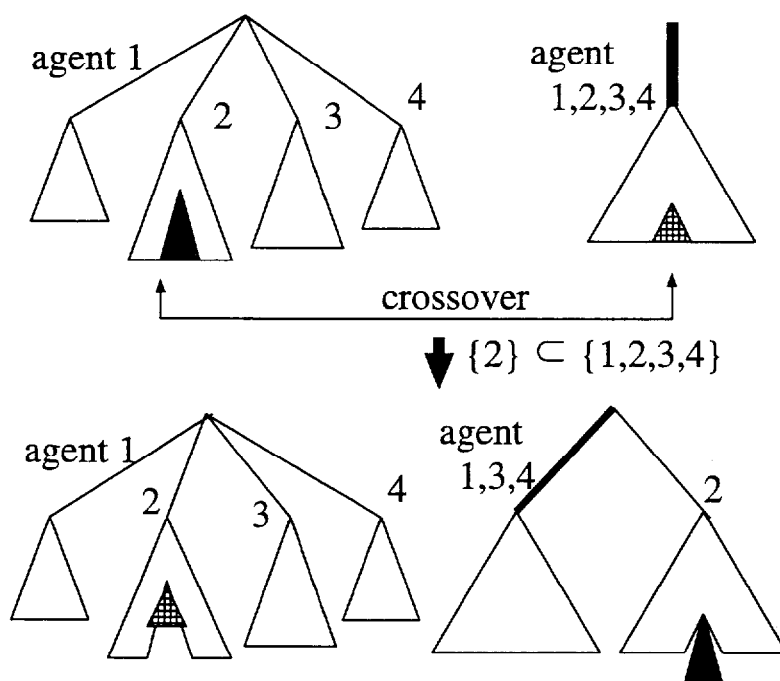


図 3.4: 交叉の例 (type b)

3. 選択された木 T, T' から交叉する部分木を各々任意に決定し、交叉を行なう。
このとき木 T と木 T' を交叉する際に、

(type a) $A(T) = A(T')$ ならば、各個体のグループ構造は変化しない。

例えば、図 3.3 はエージェント 2 が参照している木同士で交叉する場合である。交叉される木を参照するエージェントの集合が等しいため、Restricted Breeding と同様に部分木の交換を行なう。各個体のグループ構造は変化しない。

(type b) $A(T) \subset A(T')$ ならば、 T を持つ個体にグループの分割が起こり、 T' を持つ個体のグループ構造は変化しない。

例えば、図 3.4 はエージェント 2 が参照している木同士で交叉する場合である。交叉する木を参照するエージェントの集合は、左側の個体は $\{2\}$ 、右側の個体は $\{1, 2, 3, 4\}$ であり、左側が右側の個体に含まれる形となっている。左側の交叉される部分木はエージェント 2 に特化したものであり、単純に交叉すると右側の個体のエージェント 1, 3, 4 にまで影響を与え、好ましくないと考えられる。そのため右側の個体で、エージェント 2 が参照している木と同じ木を新たに生成し、その木を参照する新しいグループにエージェント 2 だけを移す。結果として、エージェント 2 が参照する木のみが交叉に使用されるようにグループの分割が起こり、それから部分木の交換が行なわれる。

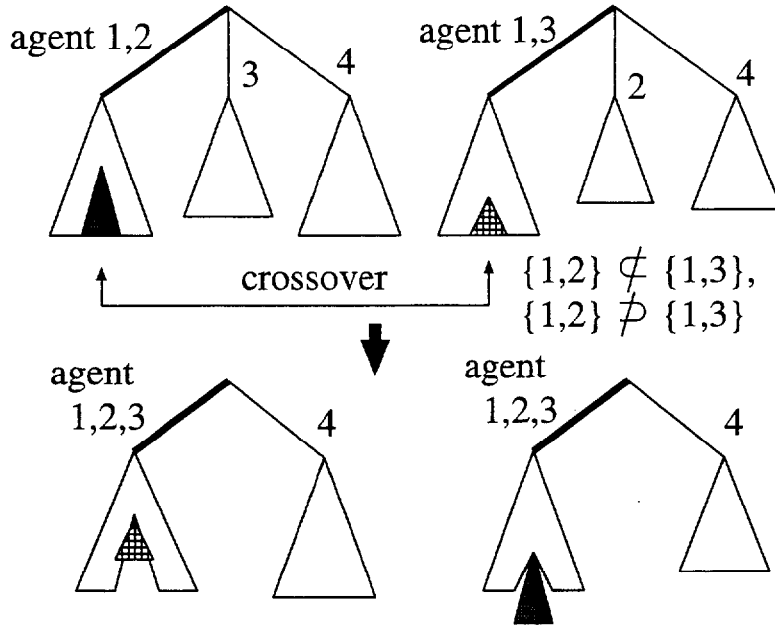


図 3.5: 交叉の例 (type c)

(type c) $A(T)$ と $A(T')$ が包含関係にない場合，どちらの個体も， $A(T) \cup A(T')$ が同一の木を参照するようにグループの統合が起こる．

例えば，図 3.5 はエージェント 1 が参照している木同士で交叉する場合である．交叉する木を参照するエージェントの集合は，左側の個体は $\{1, 2\}$ ，右側の個体は $\{1, 3\}$ であり．この 2 つの集合は包含関係にない．左側の個体はエージェント 1 と 2 が同じ行動規則に従い，右側の個体はエージェント 1 と 3 が同じ行動規則に従って行動しているため，次世代に生成される子はその和集合である $\{1, 2, 3\}$ が同じ行動規則を持つべきであると考え，エージェント $\{1, 2, 3\}$ が同じ木を参照するようにグループを統合した後，交叉を行なう．

3.2.3 グループ突然変異

上述のとおり，ADG では交叉に使用される木を参照するエージェント集合の違いにより，グループの分割や統合が発生する．よって，もし全ての個体と同じグループ構造に収束してしまうと，グループ構造の変化が起こらなくなってしまう．そのため，ADG では局所的な構造への収束を防ぐ目的で，グループ突然変異という遺伝操作を導入する．グループ突然変異では，あらかじめ定めたグループ突然変異率という確率で，エージェントを任意のグループに移す操作を行なう．ただし移動先として選択されたグループが現在所属するグループだった場合は，現在参照している木と同一の木を新たに生成し，その木を参照する新しいグループに

そのエージェントを移す。この操作は交叉によるグループ構造の変化を促進する働きを持つため、次世代の個体集団を生成する際に、交叉の前に実行する。

3.2.4 選択

次世代に子孫を残す親個体の選択には、トーナメント方式を用いる。トーナメント方式は元の集団の中から決められた数の個体(トーナメントサイズという)をランダムに選択し、その中で最も適応度の高い個体を次世代に残す個体として選択する。この過程を決められた個体数が得られるまで繰り返すという手法である。ただし、ADGでは最も適応度の高い個体が複数存在した場合、グループ数の少ない方の個体を選択することにする。この条件により、グループの過度な分割を防止することができる。例えば、Heterogeneous チームの全てのエージェントが同じ内容の行動決定木を参照している場合、このチームの適応度は、全てのエージェントが同じ行動決定木を参照する Homogeneous チームと同じである可能性が考えられる。この場合、Heterogeneous チームが持つ複数の木は余分なものであると考えることができる。このため選択の際に複数の最良個体が存在した場合は、グループ数の少ない方の個体を選択する。

このような枠組で個体集団を進化させることにより、効率的に探索を行ない、問題に対する適切な協調動作を獲得することが期待される。

また、ADGによるグループモデルにおいて、全体が1つのグループにまとまったものが Homogeneous モデルに相当し、逆に各エージェントが別々のグループを形成したものが Heterogeneous モデルに相当する。このことから ADG は、従来のモデル化を内包する、より柔軟なモデルとみなすことが可能である。

3.3 おわりに

本章では、進化により大規模エージェント群を最適化する際の重要な点である、エージェント群のグループ構造に着目し、以下の3点を自動的に獲得する手法 ADG を提案した。

- エージェント群をいくつのグループに分けるか。
- 各エージェントをどのグループに割り当てるか。
- 各グループの行動決定プログラム

ADG におけるグループ構造の最適化は以下のような操作から構成される。

- 子への形質保存, および各エージェントの特殊化という観点から, Restricted Breeding を応用したグループ分割操作
- 子への形質保存の観点からのグループ統合操作
- 個体集団の多様性の維持の観点からのグループ突然変異

一般に, GA や GP などの進化的計算法では, 個体群の多様性の維持, および形質遺伝の観点が重要であるとされる. ADG における遺伝操作も, 進化的手法の設計指針にのっとりしたものとなっている.

次章では, ADG をマルチエージェントによる荷物搬送問題に適用し, その有効性を示す.

第4章 マルチエージェントによる荷物搬送問題へのADGの適用

4.1 はじめに

本章では、ADGの有効性の検証のため、能力の異なる複数のエージェントによる荷物搬送問題へADGを適用する。従来のエージェント群のモデル化としてよく利用されるHomogeneousモデル、Heterogeneousモデルと性能を比較することにより、ADGの有効性の検証を行う。また、エージェント数や問題解決に必要とされる役割分担数が異なる環境へADGを適用し有効性を示す。

4.2 荷物搬送問題の環境設定

役割分担の発生の確認のため、図4.1のようなマルチエージェントの簡単な荷物搬送問題を設定し、本手法を適用した結果を示す。

設定した環境は以下のようなものである。

- 重さの異なる2種類の荷物を配置する。

環境内に荷物の置かれている場所を2箇所設定し、一方には軽い荷物を、もう一方には重い荷物を配置する。各地点には大量の荷物が置かれており、なくなることはないとする。

- 荷物を運び込むゴール地点を1箇所設定する。

また、ゴール地点から各荷物置き場までの距離は等しいとする。

- 搬送能力の異なる複数のエージェントを配置する。

環境内のエージェントの総数は20とする。そのうち番号0,4,8,12,16のエージェントは軽い荷物と重い荷物のどちらでも運ぶことができ、それ以外のエージェントは軽い荷物しか運べないとする。シミュレーション開始時は、全てのエージェントはゴール地点からスタートし、単位時間ごとに各エージェントは自分が参照するGP木を評価し行動を決定する。

- 一定時間内に、できるだけ多くの荷物を運ぶことを目標とする。

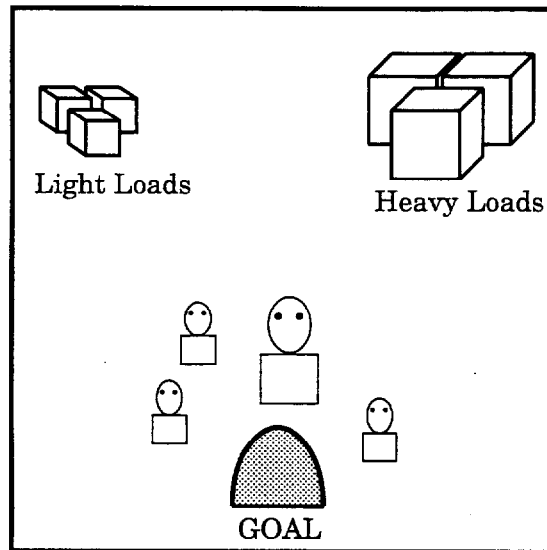


図 4.1: 荷物搬送問題

チームの適応度として、定められた時間 (*Max_Step*) 内にゴール地点に運び込まれた荷物の数に応じて得点を与える。なおエージェントは荷物を持ってゴール地点に着くと自動的に荷物を置き、運び込まれた荷物としてカウントされる。ただし、重い荷物の得点 (*Heavy_Load_Score*) の方が軽い荷物の得点 (*Light_Load_Score*) よりも高いとする。

- 環境は 11×10 の2次元格子平面である。ゴール地点は、座標 (5, 0) のセルに配置される。また軽い荷物置き場は (0, 9), 重い荷物置き場は (10, 9) に設定する。エージェントは上下左右に動くことができ、またエージェントや荷物などの各物体は同時に同じセルを占めることができるとする。

また GP を適用する際の、終端・非終端記号の設定を表 4.1 に、GP や環境に関するパラメータの設定を表 4.2 に示す。木のノードの突然変異では、突然変異率の確率で集団から個体を選択し、選ばれた各個体の中で任意に選んだ 1 つのノードで突然変異が発生するように設定した。

4.3 実験結果と考察

4.3.1 ADG による個体集団の進化と獲得されたグループ構造

以上のような設定で ADG による実験を行なった。図 4.2 にこの実験の結果を示す。この図は最大適応度を世代ごとにプロットしたものである。なおグラフは 10 回の実験の平均値である。今回の実験では、各エージェントは定められた時間内

表 4.1: 荷物搬送問題における終端・非終端記号の設定

記号	引数の数	機能
If_carrying_load	2	エージェントが荷物を持っているかどうかで分岐
If_load_here	2	現在地に荷物があるかどうかで分岐
Move_goal	0	ゴールの方へ向かう
Move_heavy_load	0	重い荷物の方へ向かう
Move_light_load	0	軽い荷物の方へ向かう
Pick_up	0	現在地に荷物があり、現在荷物を持っていない状態で、かつ、それを持つ能力があれば拾う

表 4.2: GP, 環境のパラメータ

個体数	1000
選択	トーナメント +エリート保存
トーナメントサイズ	5
交叉率	0.9
突然変異率 (突然変異が 適用される個体の割合)	0.9
グループ突然変異率	0.01
Max_Step	100
Heavy_Load_Score	5
Light_Load_Score	1

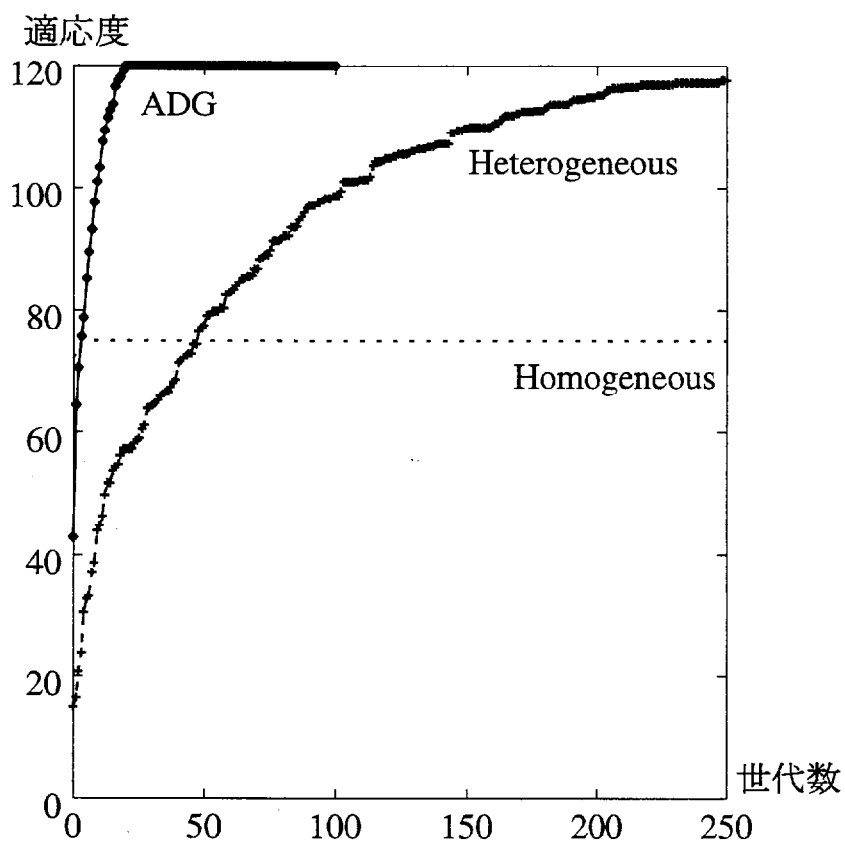


図 4.2: 適応度の変化

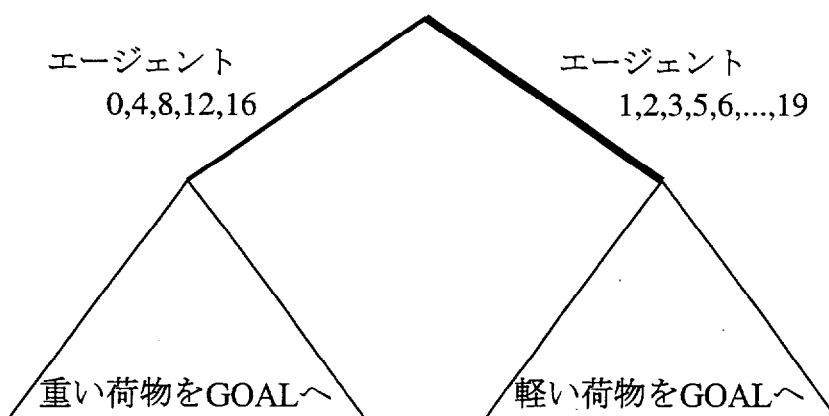


図 4.3: 獲得された木構造

にゴール地点と荷物置き場を3往復することが可能であり、20のエージェントの中で、5つのエージェントが重い荷物を運ぶことができるので、とりうる最大適応度は、 $(5 \times \text{Heavy_Load_Score} + 15 \times \text{Light_Load_Score}) \times 3 = 120$ となる。このグラフから ADG では約 20 世代目で最適な行動規則が得られていることがわかる。

また 10 回の試行全てにおいて、実行後獲得された木構造は、図 4.3 に示すように、あらかじめ重い荷物を運ぶことができるように設定した番号 0,4,8,12,16 のエージェントのグループと、それ以外のエージェントのグループの 2 つに分かれており、各々重い荷物を運ぶものと軽い荷物を運ぶものに役割が分担されていた。本手法で獲得された木構造を見ることにより、必要な役割分担数やどのエージェントが共通の役割を担っているかを知ることができる。このような個々の能力に応じたグループ分けは、前章で挙げた Legion system によるエージェント群の分割比率を最適化する処理では実現できないものであり、本手法がより高度なグループ化のために有効であることが分かる。

また図 4.4 に、ある 1 回の GP 実行時における各世代の個体群の平均グループ数の変化を示す。実行開始時はランダムにグループ分けされた個体群が、次第に最適なグループ数 2 に近付きながら探索を行なっていることが分かる。

4.3.2 従来法 (Homogeneous/Heterogeneous モデル) との比較

次に本手法の有効性を示すため、同じ環境、同じパラメータ設定を用いて、グループの分割や統合の操作を行なわない、従来の Heterogeneous や Homogeneous のみの集団での実験も行なった。このときの適応度の変化も図 4.2 に示されている。なおグラフは各々 10 回の実験の平均値である。Heterogeneous のみの集団では、10 回の試行の内、最大の適応度が得られたのは最短でも 164 世代目であった。本手法に比べ進化に多くの時間が必要になったのは、Heterogeneous の個体ではエージェントの数と同じ 20 個の木構造を最適化する必要があり探索領域が膨大になったためと考えられる。またこの問題では探索に十分な時間をかけることにより、Heterogeneous のみの集団でも ADG と同じ適応度を最終的に得ることができるが、獲得された木構造では同じ働きをするエージェント同士でも別々のプログラムを参照しており、役割分担がどのように行なわれたかを知るには一つ一つのプログラムを解析する必要がある。ADG を使用することによりグループ化や分業の進化などを明示的に確認できることは、本手法の利点の 1 つである。また Homogeneous のみでの集団では、最適な行動規則は得られなかった。これは能力の異なるエージェント集団を同一のプログラムで動作させるため、能力に応じた仕事ができなかったためである。そのため獲得された行動は、能力によらず全てのエージェントが重い荷物を運ぶことを試みるもので、適応度の最大値は、 $5 \times \text{Heavy_Load_Score} \times 3 = 75$ となっている。この実験から、ADG を用いることにより、探索領域を抑えながら、役割分担のような協調動作を実現できることが分かる。

さらに、ADG における交叉によるグループ操作の効果を確認するため、グルー

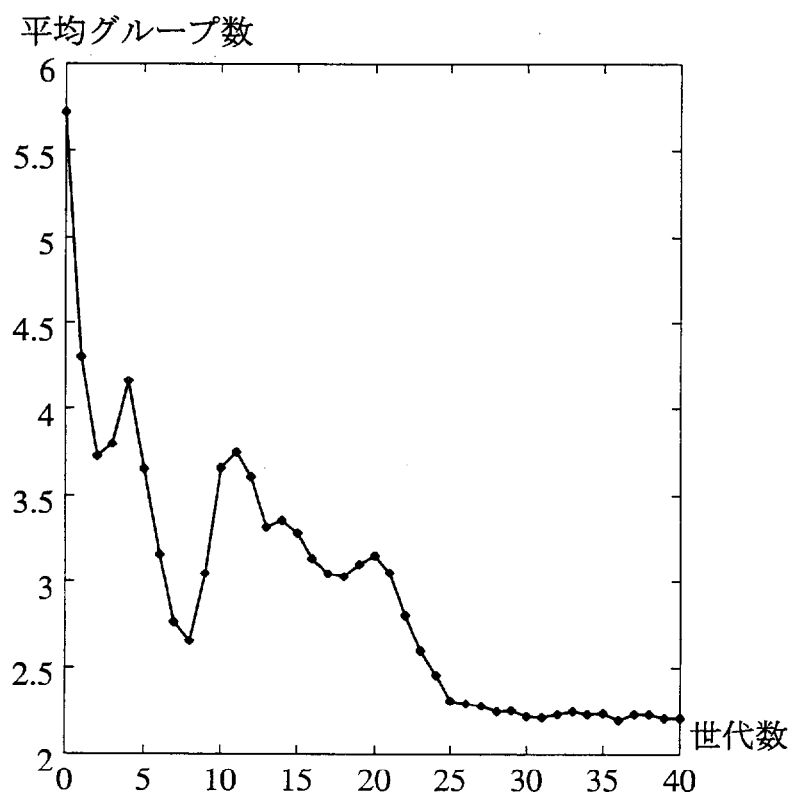


図 4.4: ADG の各世代における平均グループ数の変化

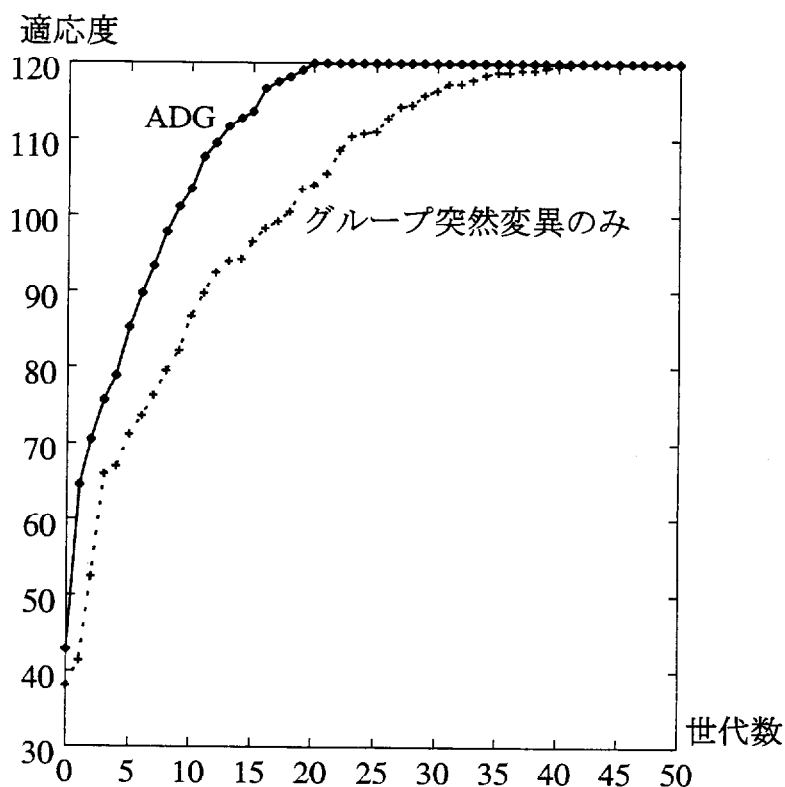


図 4.5: ADG とグループ突然変異のみを用いた手法の比較

ブ構造の変化にグループ突然変異のみを使用する手法との比較を行なった。この手法における交叉は、グループの分割や統合を行わずに部分木の交換のみを行なう Restricted Breeding であり、その他の遺伝操作などについては ADG と同様に行なう。実験はともに表 4.2 に示すパラメータで行なった。図 4.5 にこの実験の結果を示す。なおグラフは各々 10 回の実験を行ないその平均を取ったものである。このグラフから、収束速度において、グループ突然変異のみを用いる手法よりも ADG の方が優れていることが分かる。また ADG では、10 回の試行全てで、2 つのグループからなる最適な構造が得られた。これに対して、グループ突然変異のみを用いた手法では、100 世代終了時に最適な構造が得られたのは 10 回の試行中 1 回だけであり、必要以上にグループ分けされた構造が多く見られた。以上のことから、ADG の交叉による分割や統合が効果的であることが分かる。

4.4 エージェント数や必要な役割分担数が増加した場合の検証

4.4.1 実験環境の設定

前節の実験は、エージェント数 20、最適なグループ数 2 という特定の環境における結果であった。本節では、エージェント数や必要とされるグループ数が増加した場合の ADG の有効性の検証を行う。

必要とされるグループ数を増加させるため、4.2 節で示した実験環境を以下のように変更した。

- 重さの異なる N 種類の荷物と搬送能力の異なる N 種類のエージェントを環境内に配置する。
- N 種類の荷物の重さ w は、軽い方から順に、 $1, 2, \dots, N$ とする。また、重い荷物ほどゴール地点に搬送した際に得られる得点が高いとする。
- N 種類のエージェントの搬送能力 p は、能力の低いものから順に、 $1, 2, \dots, N$ とする。

エージェント群の各搬送能力タイプへの振り分けは、エージェントの番号を N で割った剰余類によって分類した。

- エージェント i の力を $p_i (1 \leq p_i \leq N)$ 、荷物の重さを $w (1 \leq w \leq N)$ とするとき、 $p_i - w \geq 0$ ならばその荷物を搬送可能とする。
- GP の終端記号 `Move.heavyload`, `Move.lightload` を `Move.loadi` ($i = 1, 2, \dots, N$) に変更する。
- 各種類の荷物が配置される地点は、各荷物地点からゴール地点までのマンハッタン距離が同一になるように設定する。また、目標とする荷物地点へ移動中に他の荷物地点の上を通過してしまうことがないように配置を設定する。
この設定は、意図しない荷物を運んでしまうことを防止するためである。これは、使用している GP の関数記号 `if_load_here` が、現在地の荷物の種類を判定できないことから生じる。

このような環境設定では、各エージェントが自分の搬送能力の範囲内で、できる限り重い荷物を担当することが最も効率が良い。ADG を用いることにより、荷物の種類と同じ N 個のグループからなる集団が構成され、各グループの行動規則はその搬送能力に応じた荷物を担当するという解が得られることが期待される。

4.4.2 実験結果と考察

前節で示した環境を用いて、以下の3通りの組合せで実験を行った。

1. エージェント数 20, 荷物の種類 $N = 10$
2. エージェント数 50, 荷物の種類 $N = 5$
3. エージェント数 50, 荷物の種類 $N = 10$

上記1,2,3各々の場合において、ADGを用いてエージェント群を進化させた際の最大適応度の変化、および各世代における平均グループ数の変化を図4.6, 図4.7, 図4.8, 図4.9, 図4.10, 図4.11に各々示す。なお、最大適応度の変化のグラフにおける点線は、設定した環境において、最適な役割分担がなされたときに獲得できる理論上の最高得点を示している。

最大適応度のグラフを見ると、上記1,2,3いずれの場合も、最適な性能が得られていることが分かる。また、各世代における平均グループ数の変化のグラフを見ると、10種類の荷物が存在する環境の結果である図4.7, 図4.11では、進化とともに平均グループ数が10に近付いており、5種類の荷物が存在する環境の結果である図4.9では、進化とともに平均グループ数が5に近付いていることが確認できる。

また実際に、各実験において獲得された最良個体のグループ構造を解析した結果、いずれの実験においても、搬送能力ごとにエージェントがグループ分けされ、各グループがその能力に応じた荷物の搬送に携わっていることを確認した。

これらの結果から、ADGはエージェント数、および必要とされるグループ数が増加した場合も有効に機能することが分かる。今後は、ADGの理論的な解析も必要となる。

4.5 おわりに

本章では、能力の異なる複数のエージェントによる荷物搬送問題へADGを適用し、ADGの有効性の検証を行った。

エージェント群のモデル化として従来使用されてきたHomogeneousモデルやHeterogeneousモデルと比較を行った結果、ADGは、Homogeneousモデルよりも複雑な協調動作が可能であること、Heterogeneousモデルよりも効率的に最適化が可能であることを確認した。また、獲得されたグループ構造を見ることで、エージェント群がいくつのグループに分かれているか、どのエージェントが同じ役割を担っているかを容易に判断することが可能であり、エージェントの行動解析にも有効であることを確認した。

また、ADGをエージェント数や必要とされる役割分担数が異なる複数の環境に適用し、環境に応じた最適なグループ構造を獲得できることを確認した。

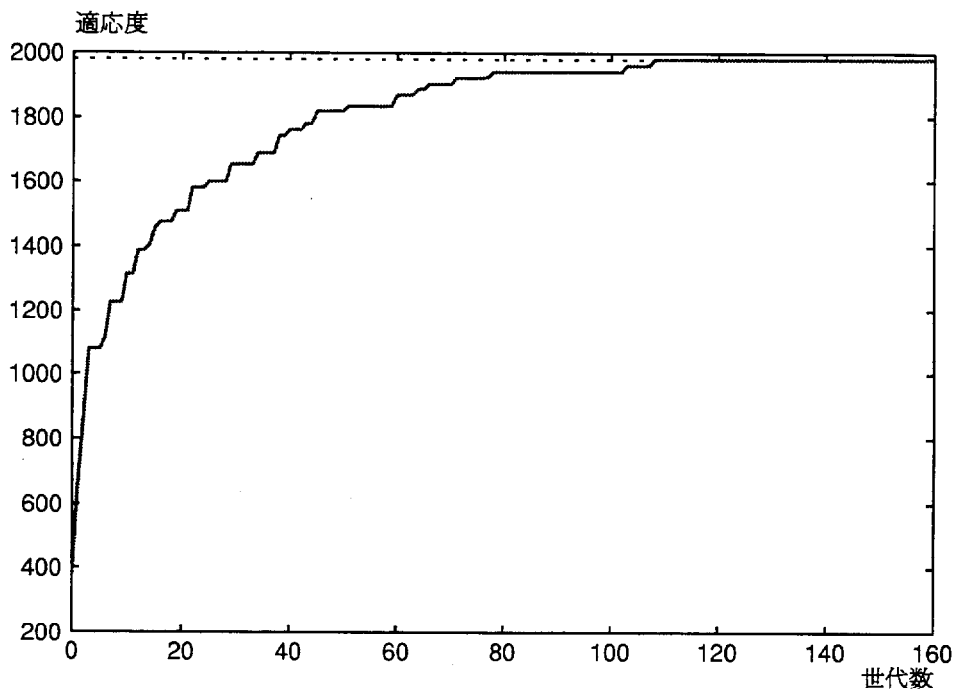


図 4.6: エージェント数 20, 役割分担数 10 の環境における適応度の変化

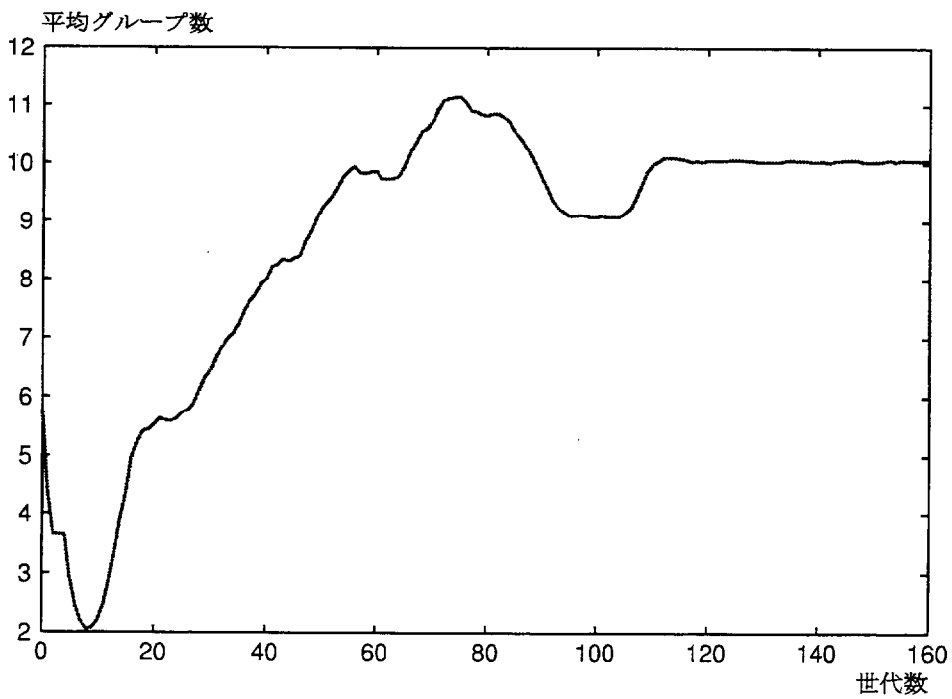


図 4.7: エージェント数 20, 役割分担数 10 の環境における平均グループ数の変化

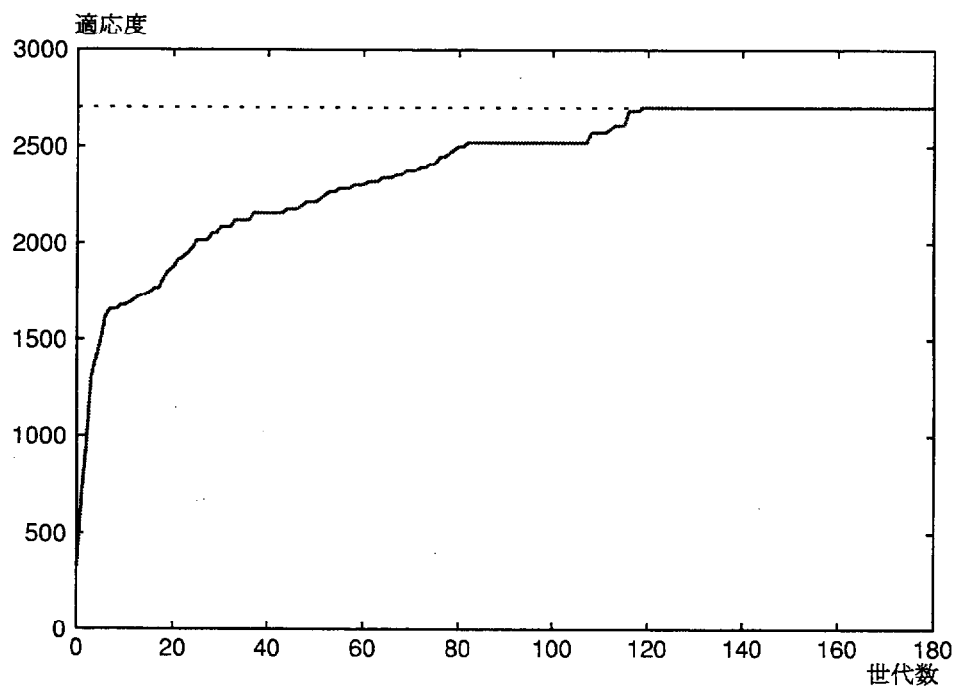


図 4.8: エージェント数 50, 役割分担数 5 の環境における適応度の変化

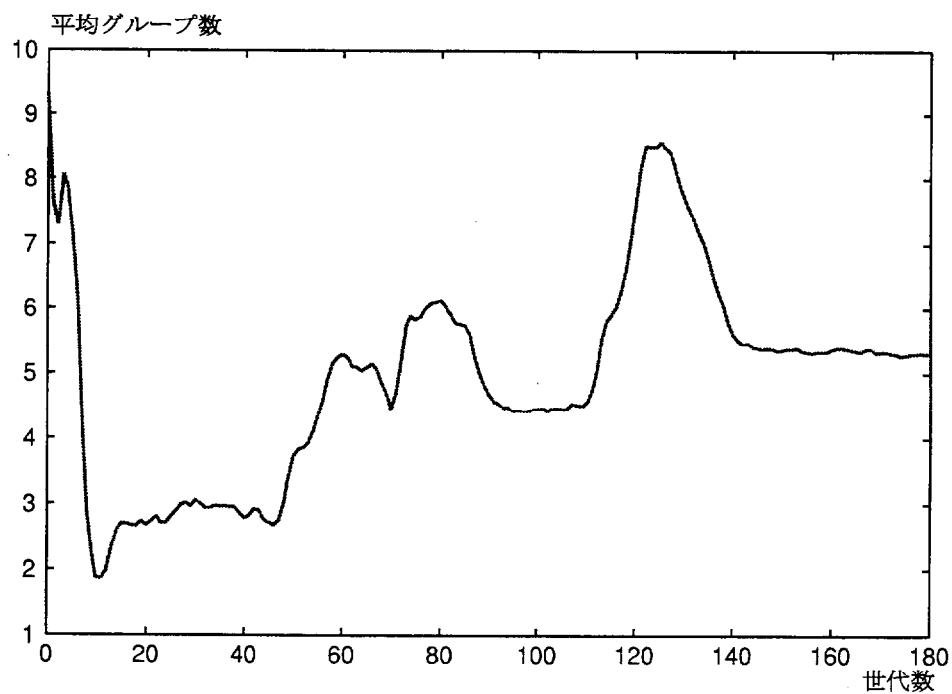


図 4.9: エージェント数 50, 役割分担数 5 の環境における平均グループ数の変化

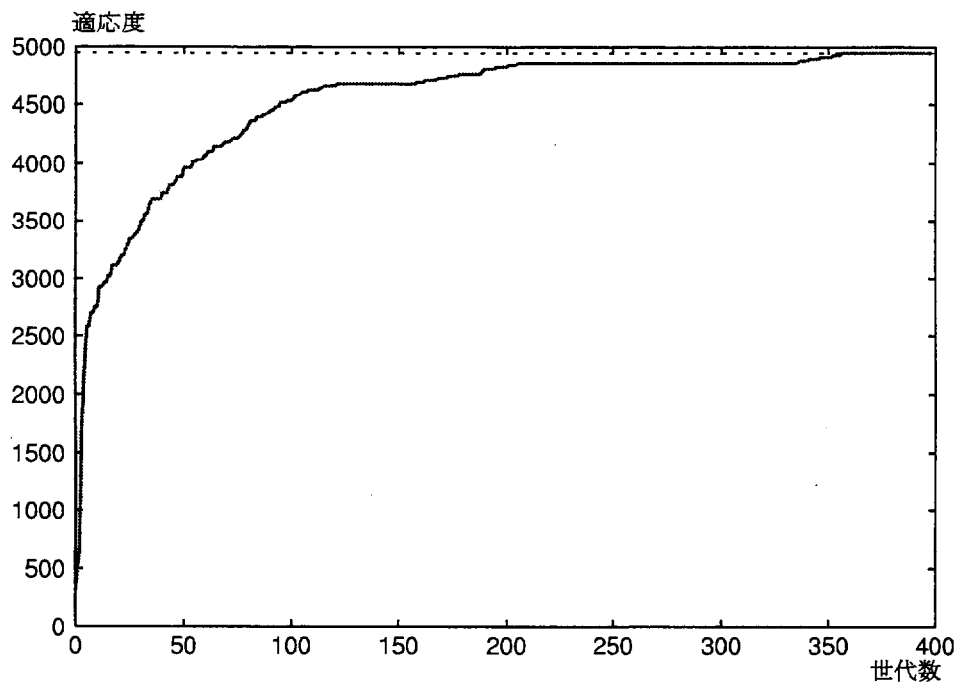


図 4.10: エージェント数 50, 役割分担数 10 の環境における適応度の変化

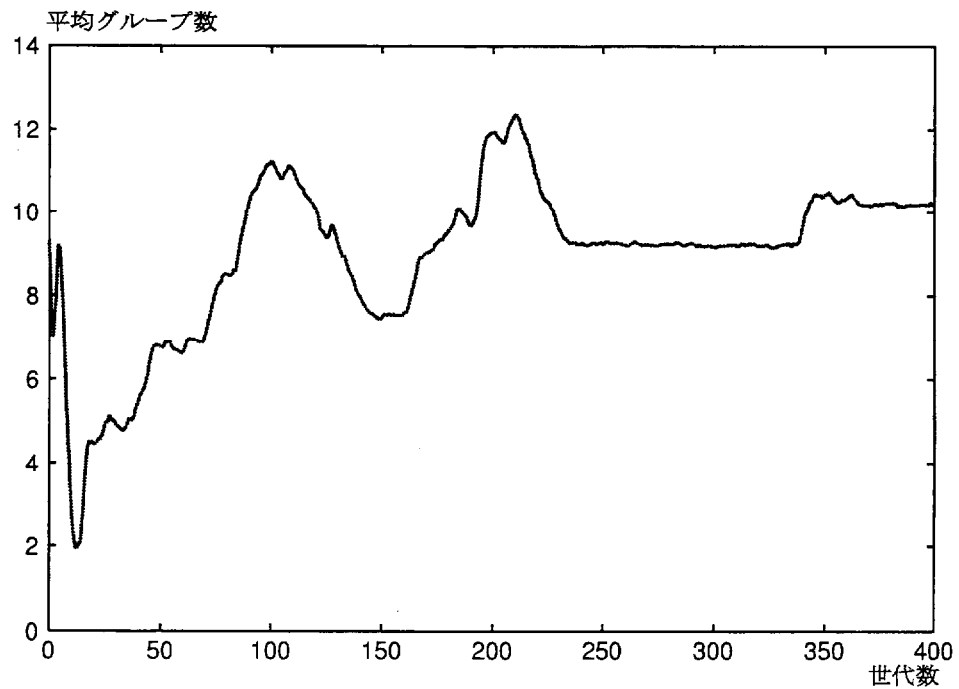


図 4.11: エージェント数 50, 役割分担数 10 の環境における平均グループ数の変化

第5章 タイルワールド環境における マルチエージェント制御への ADGの適用

5.1 はじめに

第4章では荷物搬送問題における実験により、ADGが主に探索領域の抑制の点で効果的であり、また獲得された木構造は、問題解決に必要な分業の理解に役立つことを示した。ただし獲得された行動については、Heterogeneousモデルによるものと同じであり、グループ化の有効性を示すものではなかった。この章では別の環境での実験により、グループ構造の違いによる問題解決の効率への影響という点からADGの有効性について検討する。

5.2 タイルワールドの環境設定

ここではエージェントの協調が必要とされる別の環境でのシミュレーションをADGを用いて行なう。マルチエージェントの標準的なテスト環境として良く知られているものに、タイルワールド [37, 38] がある。タイルワールドは2次元格子上に、エージェント、タイル、ゴール、障害物を配置した、計算機上の仮想的な実験環境である。各物体は1つのセルを占め、エージェントは上下左右に単位時間当たり1セル動くことができる。エージェントに隣接するセルにタイルがある場合、その方向にエージェントが移動することによりタイルを押すことができ、各々が隣接するセルへと移動する。エージェントの目標は全てのタイルをゴール地点へできる限り早く運ぶことである。

ここで扱う実験環境は、このタイルワールドを拡張した問題である。具体的な問題設定は以下のとおりである。環境はトラス状の2次元格子平面であり、その上に複数のエージェントといくつかのタイル、そして1つのゴール地点が配置される。各タイルは1つのセルを占め、他のタイルやエージェントと重なることはないが、複数のエージェントは同一のセルを占めることが可能である。エージェントは上下左右に単位時間当たり1セル動くことができる。エージェントに隣接するセルにタイルがある場合、その方向にエージェントが移動することによりタイルに力を加えることができる。この問題で最も重要な点は、タイルを動かすためには複

表 5.1: タイルワールドにおける終端・非終端記号の設定

記号	引数の数	機能
+	2	2つのベクトルを加える.
-	2	第1引数のベクトルから第2引数のベクトルを引く.
*2	1	ベクトルの大きさを2倍にする.
/2	1	ベクトルの大きさを半分にする.
->90	1	ベクトルを時計回りに90度回転させる.
inv	1	ベクトルを反転させる. 例えば, 入力が $\vec{v}$ のとき $-\vec{v}$ を返す.
ifdot	4	4つの引数 $\vec{x}_1, \vec{x}_2, \vec{x}_3, \vec{x}_4$ に対して, ($\vec{x}_1 \cdot \vec{x}_2 \geq 0$) の場合は $\vec{x}_3$ を, そうでない場合は $\vec{x}_4$ を返す.
if>=	4	4つの引数 $\vec{x}_1, \vec{x}_2, \vec{x}_3, \vec{x}_4$ に対して, ($ \vec{x}_1 \geq \vec{x}_2 $) の場合は $\vec{x}_3$ を, そうでない場合は $\vec{x}_4$ を返す.
Last	0	エージェント(自分)に対するGP木の最後の出力ベクトル. 最初に動くときは0ベクトルを返す.
Goal	0	エージェント(自分)からゴールへのベクトル.
Nearest-Tile	0	エージェント(自分)から最も近いタイルへのベクトル.
Same-Group	0	同じグループに属するエージェントのうち, 自分から最も近いエージェント (ただし同じ位置にいるエージェントは除く)へのベクトル.

数のエージェントが同時に押すことが必要となることである. 各エージェントは, タイル方向に移動することにより, タイルに大きさ1の力ベクトルをかけることができる. タイルにかかる全ての力の合成ベクトルが, ある一定の値 *Power_Radius* よりも大きくなったとき, エージェントはそのタイルをその合成ベクトルの方向に運ぶことができる. つまり, タイルを動かすためには, 複数のエージェントが同じ方向に同時に押すことが必要となる. この問題の目的は一定時間内に, ゴール地点に全てのタイルをできる限り早く運ぶことである. また今回の実験では簡単のため, 障害物の配置やタイル, ゴール地点の確率的な出現消滅という条件は省略した.

この問題でのエージェントの行動の進化にGPを適用するため, 表5.1に示すような終端・非終端記号を使用する. GPの木表現のためにベクトル操作を選び, 終端記号の型と非終端記号の引数, 返り値の型がいずれも2次元ベクトルとなるように設定した. これらの記号は, 文献[22, 23]における記号設定を参考に設定した. シミュレーションでは単位時間ごとに, 各エージェントは自分の参照するGP木を評価し行動を決定する. GP木は常に2次元ベクトルを返し, そのベクトルがエージェントの取る行動に変換される. エージェントの行動は, STAY(その場に留まる), UP(上に移動), DOWN(下に移動), LEFT(左に移動), RIGHT(右に移動)の5つである. GP木の出力である2次元ベクトル $\vec{v}$ から, エージェントの取

る行動への変換は次のように行なわれる。

ベクトルの大きさ $|\vec{v}|$ がパラメータ $Radius$ 以下である場合、エージェントは現在の位置に留まる (STAY)。そうでなければ、ベクトル $\vec{v}$ を上下方向と左右方向の成分に分割し、最大の成分の方向へ1セル移動する。タイルの移動方向も、かかる力の合成ベクトルとパラメータ $Power_Radius$ を用いて同様に決定される。タイルが動かない場合は、その方向に移動しようとしたエージェントはその場に留まることになる。

GP の各個体の適応度は、定められた時間 (Max_Step) 内でいかに効率良く仕事を完了できるかによって評価を行なう。そこで、以下の3つの要素を考慮する。

- ft : 時間内に運び込まれたタイルの数
- t_F : 全てのタイルをゴールへ運び終えるのににかかったタイムステップ数。仕事が完了しなかった場合は、 $t_F = Max_Step$ となる。
- 仕事が完了しなかった場合、タイルをゴールの近くまでどのくらい押すことができたか。

これらの要素を、パラメータ $Bonus$, $Speed$, $Dist$ で各々重み付けする。つまり適応度は次のように計算される。

$$f = Bonus \times ft + Speed \times (Max_Step - t_F) + Dist \times \sum_{t \in LT} \{MD(og(t), Goal) - MD(cr(t), Goal)\}, \quad (5.1)$$

ここで LT はゴールへ運ばれずに残されたタイルの集合である。 $MD(x, y)$ は x と y のマンハッタン距離を表し、また $og(t)$, $cr(t)$ は各々、タイル t の初期位置と現在の位置を表す。よって、 $\{MD(og(t), Goal) - MD(cr(t), Goal)\}$ はゴールへのタイル t の移動距離を表す。この適応度は、各タイルをゴールへ近付けるほど高く、また全てのタイルを運び終えた場合は費やしたステップ数が少ないほど高くなるように設定されており、できるだけ多く、かつ、効率的にタイルをゴールへ運ぶ行動が学習できる [23]。

今回の実験では、10のエージェント、5つのタイル、1つのゴール地点を図5.1に示すように重なりなく配置する。また、 $Power_Radius = 3.0$ 、すなわち、1つのタイルの搬送には3つのエージェントが同じ方向に同時に押す必要があるように設定する。実験で使用したパラメータを表5.2に示す。

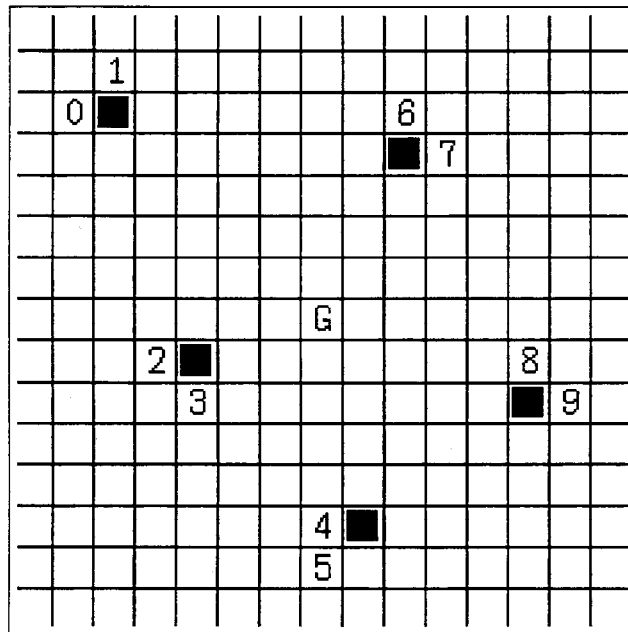


図 5.1: タイルワールドの環境 (初期配置)(0,...,9 :エージェント, ■:タイル, G:ゴール)

表 5.2: GP, 環境のパラメータ

個体数	1000
選択	トーナメント +エリート保存
トーナメントサイズ	5
交叉率	0.8
突然変異率 (突然変異が 適用される個体の割合)	0.9
グループ突然変異率	0.075
<i>Radius</i>	1.0
<i>Power_Radius</i>	3.0
<i>Max_Step</i>	200
<i>Bonus</i>	2500
<i>Speed</i>	80
<i>Dist</i>	100

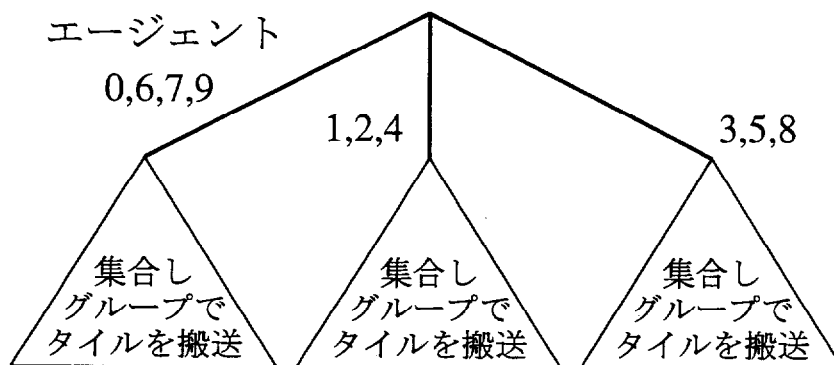


図 5.2: 獲得された木構造

5.3 実験結果と考察

5.3.1 ADG により獲得されたグループ構造と行動の様子

以上のような、環境の下で ADG による実験を行なった。図 5.2 は ADG により獲得された最良個体のチーム構造を示したものである。この図から、このチームは 3 つのグループに分割されており、各グループが 3 つ以上のエージェントを含んでいることが分かる。また図 5.3 は獲得された最良個体による動作の一場面を示したものである。この動作の様子を観測すると、同じグループに属しているエージェントが集団を形成し、各々のグループがタイルをゴール地点へ運ぶというグループ行動が見られた。なお、付録 A に獲得された最良個体の木構造を、付録 B にこの個体による動作の様子を示す。

また図 5.4 に、ある 1 回の GP 実行時における各世代の個体群の平均グループ数の変化を示す。実行開始時はランダムにグループ分けされた個体群が、次第に最適なグループ数 3 に近付きながら探索を行なっていることが分かる。

5.3.2 従来法 (Homogeneous/Heterogeneous モデル) との比較

ADG の効果を確認するため、同じ環境の下で、従来の Homogeneous や Heterogeneous モデルでの実験も行なった。Homogeneous モデルは全てのエージェントが同一グループに属する集団とみなすことができ、一方 Heterogeneous モデルでは各々が別々のグループを構成する集団とみなすことができる。ただし Heterogeneous での実験では公平のため、終端記号 Same-Group ではなく最も近い別グループのエージェント (ただし同じ位置にいるエージェントを除く) へのベクトルを表す Different-Group を用いた。このときの適応度の変化を図 5.5 に示す。今回の実験では、(5.1) 式から分かるように、5 つのタイルを運び終えたときに得られる適応度は、 $Bonus \times 5 = 12500$ 以上であり、より短い時間で効率良くタスクを完

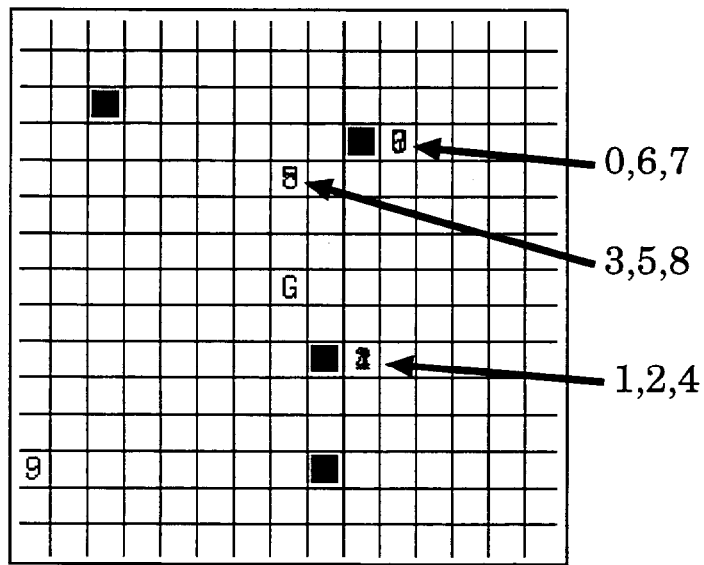


図 5.3: 進化により得られたマルチエージェントの行動

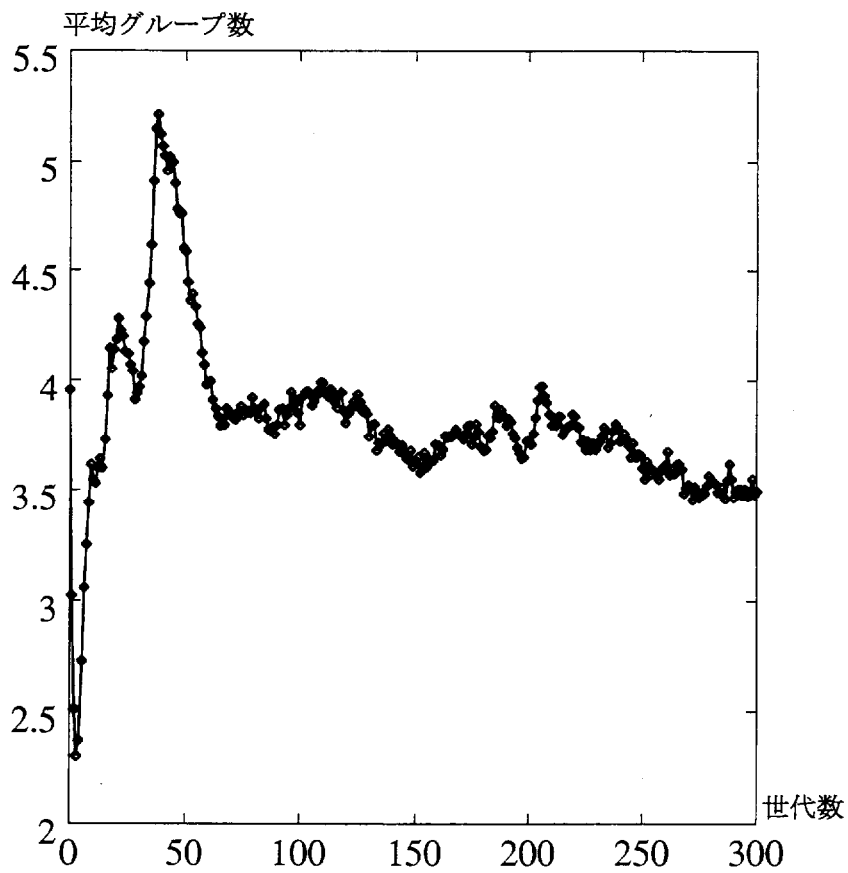


図 5.4: ADG の各世代における平均グループ数の変化

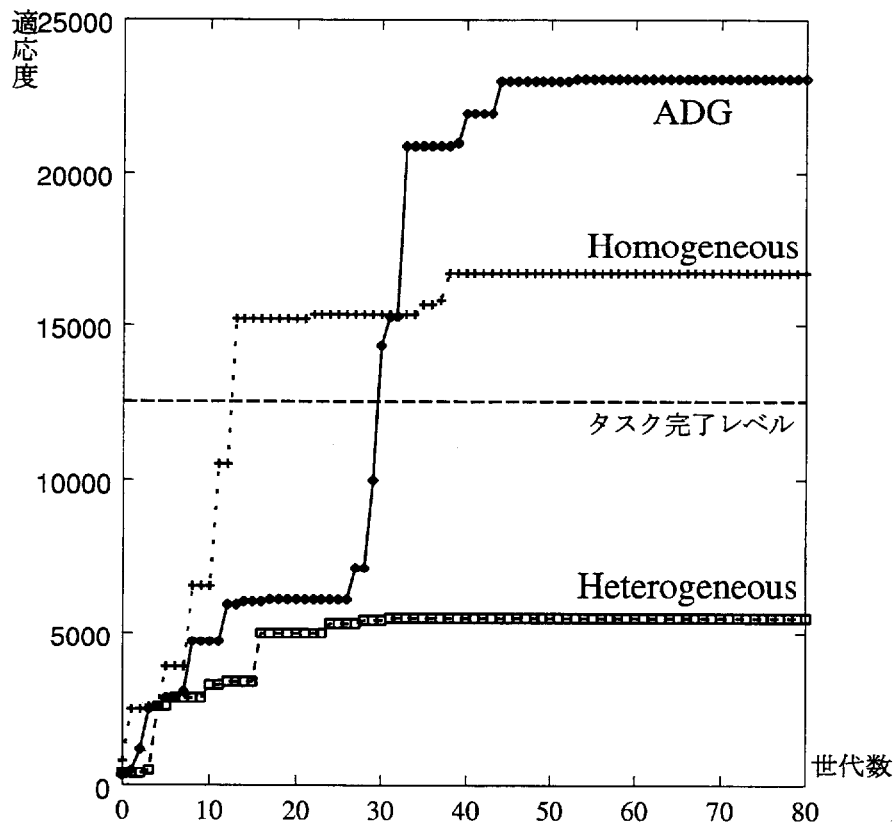


図 5.5: タイルワールドにおける ADG と従来手法との比較

了する個体ほど、さらに高い適応度を得ることができる。Homogeneous 集団では Same-Group により、次第に集団を構成してタイルをゴールへ運ぶ行動が獲得されているが、独立した小集団による並列処理が実現できておらず、効率では ADG に劣っている。また Heterogeneous 集団では、各エージェントの木を最適化するため探索領域が膨大になり、タスクを完了する集団は獲得できなかった。この実験から、効率的な行動を行なうには、集団の適切なグループ化が必要であり、本手法によりそれが実現できていることが分かる。

5.4 おわりに

本章では、グループ構造の違いによる問題解決の効率への影響という点から ADG の有効性について検証を行った。

タイルワールド環境での実験において、ADG ではグループ化により独立した 3 つの小グループが各々問題解決にあたることで、従来の Homogeneous モデル、Heterogeneous モデルよりも高い性能を示した。このことは、Heterogeneous モデルに対する ADG の利点が、効率的な探索により最適化に要する時間を短縮できる

ことだけでなく、より複雑な協調動作を表現できる可能性があることを示唆している。このように、ADGを用いることでグループ化による効率的な問題解決が可能であることを示した。

また、前章での実験と同様に、獲得されたグループ構造を見ることで、エージェント群がいくつのグループに分かれているか、どのエージェントが同じ役割を担っているかを容易に判断することが可能であり、エージェントの行動解析にも有効であることを確認した。

第6章 動的環境への適応のためのグループ構造の変更方法の検討

6.1 はじめに

第4章, 第5章で示した実験では, 獲得されたグループ構造は問題解決中は固定であり, 自分が担当する仕事が完了した場合に, 自ら現在の役割を変更し, 他のグループの仕事を手伝うといったことは考慮していない. 動的な環境下では, 問題の解決途中においてもグループ構造の変更が有効である [39].

本節では, 動的環境への適応を目指して, 以下のようなグループ構造の変更方法を検討した結果について述べる.

- GP における終端記号にグループ移動記号を設定する手法
- 同一のエージェントでも, 内部状態が異なる場合は別のエージェントとみなし, 一括して最適化する手法

6.2 グループ移動記号によるメンバの移動による動的環境への適応

6.2.1 グループ移動記号の設定

本節では, 問題解決途中に図 6.1 のようなグループ間のメンバの移動により, 組織構造を変更し動的環境へ適応する方法について検証する.

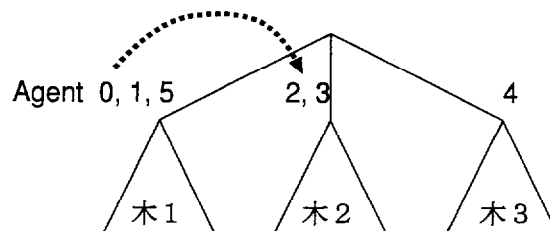


図 6.1: メンバの移動によるグループ構造変更の概念

この実現のため、GPの終端記号にグループ移動記号を導入する。この設定方法としては、以下のようなものが候補として考えられる。

方法1 各グループに所属するエージェントの内、番号が最小のものを選び出し、その大きさを基に、順に、group1, group2, group3, ... と名付ける。また終端記号には、各グループへの移動に対応した Shift_to_group i を用意する。
例えば、図6.1では、左から順に、group1, group2, group3となる。

方法2 エージェント i が所属するグループへの移動記号として、Shift_to_agent i を用意する。

例えば、図6.1では、木2への移動記号として、Shift_to_agent 2, Shift_to_agent 3 が利用できることになる。

方法3 各グループに所属するエージェントの内、番号が最小のものを各々選び出し、そのエージェント番号 i を用いてそのグループ名を group i と名付ける。また終端記号には、各グループへの移動に対応した Shift_to_group i を用意する。

例えば、図6.1では、左から順に、group0, group2, group4となる。初期集団の生成や突然変異の際の新たな木構造プログラムの生成時には、この個体用のグループ変更記号として、Shift_to_group 0, Shift_to_group 2, Shift_to_group 4のみを用意する。

方法1では、例えば、{0,1,2,3}, {4,5}のように2つにグループ分けされた個体を考えた場合、これらのグループ名は順に、group1, group2となる。図6.1の個体と比較した場合、同じgroup2という名前であってもそのグループを実際に構成するエージェント群は全く異なってしまう。そのため、Shift_to_group 2という記号を含んだプログラムを交叉により親個体同士で交換しても、グループ移動に関する形質を適切に子に伝えることはできない。

また、方法2では、上記のグループ移動の形質遺伝に関する問題は避けられるが、エージェント数と同じだけの終端記号を用意する必要があることから、GPの探索領域が膨大になり実際の使用は困難である。

これに対し、方法3では、例えばShift_to_group 2という記号は、エージェント2が参照するグループへの移動といったように、GP個体間でのShift_to_group i は少なくともエージェント i の所属しているグループという共通性を持つ。このように参照に関する形質を保存することができるため、方法3は方法1に比べて優れていると考えられる。また、エージェント数が多くなった場合でも、初期集団の平均グループ数はエージェント数に比べて小さいため、初期個体群の生成時や突然変異による新たな部分木の作成時に用いる終端記号の数を抑えることができ、方法2よりも効率的に探索を行えると考えられる。よって、方法3を採用し、グループ間のメンバの移動による動的環境への適応について検証する。

表 6.1: 動的環境下の荷物搬送問題における終端・非終端記号の設定

記号	引数の数	機能
If_carrying_load	2	エージェントが荷物を持っているかどうかで分岐
If_load_here	2	現在地に荷物があるかどうかで分岐
If_fail	2	前ステップの行動が失敗したかどうかで分岐
Move_goal	0	ゴールの方へ向かう
Move_heavy_load	0	重い荷物の方へ向かう
Move_light_load	0	軽い荷物の方へ向かう
Pick_up	0	現在地に荷物があり、現在荷物を持っていない状態で、かつ、それを持つ能力があれば拾う
Shift_to_group i	0	エージェント i の所属しているグループへ移動

6.2.2 動的環境の設定

動的環境の簡単な例として、4.2節と同様に、ある一定ステップ内にできるだけ多くの荷物を運ぶという荷物搬送問題を設定し実験を行なった。ただし、今回の実験では、重い荷物の方が数が少なく、途中で先に重い荷物がなくなる場合を想定する。

当初の予備実験では、重い荷物がシミュレーション途中で尽きてしまうように、重い荷物の数をある程度小さな一定値に設定した環境を設定し実験を行った。しかし、この場合は、あらかじめ重い荷物を運ぶことができるエージェントの一部を、始めから軽い荷物を運ぶグループに割り当てておくといった固定された構造が獲得された。これは、ADGのエージェント割り当て機能がはたらき、動的なグループ変更が必要ないように、あらかじめ用意された全ての重い荷物を実行ステップ内でちょうど運ぶ終わるのに最適なエージェント数だけ(例えば、力のあるエージェント5つのうち2つだけなど)を、重い荷物を運ぶグループに配置することが可能であったためである。

このため、以降の実験では、世代ごとに用意する重い荷物の数をランダムに設定し、ある世代ではシミュレーション中に重い荷物が尽きることがなく、また別の世代では、早いステップで重い荷物が尽きてしまうといった環境を使用した。

また、この実験で使用したGPの終端・非終端記号を表6.1に示す。エージェントは自分が取った直前の行動が、実行できたかどうか判断することが可能であり、If_failは前ステップの行動が失敗したかどうかで分岐する関数である。今回の実験では次の4つの場合が行動の失敗となる。

1. 軽い荷物しか運べないエージェントが重い荷物に対して、Pick_upを実行した場合。
2. 重い荷物がなくなっているのに、Move_heavy_loadを実行した場合。

この場合、方向が分からずその場にとどまることとする。

3. 荷物がない地点で、Pick_up を実行した場合。

4. 既に荷物を持っているのに、Pick_up を実行した場合。

上記の3,4に関しては、行動規則を最適化することにより完全に除去できるが、1に関しては、エージェントは自らの搬送能力を判断するための記号を持っていないことから、能力に応じて異なる木を参照するように適切なグループ化が行なわれなければならない。また2に関しては、シミュレーション中に、参照する行動規則の変更が必要になる。

6.2.3 実験結果と考察

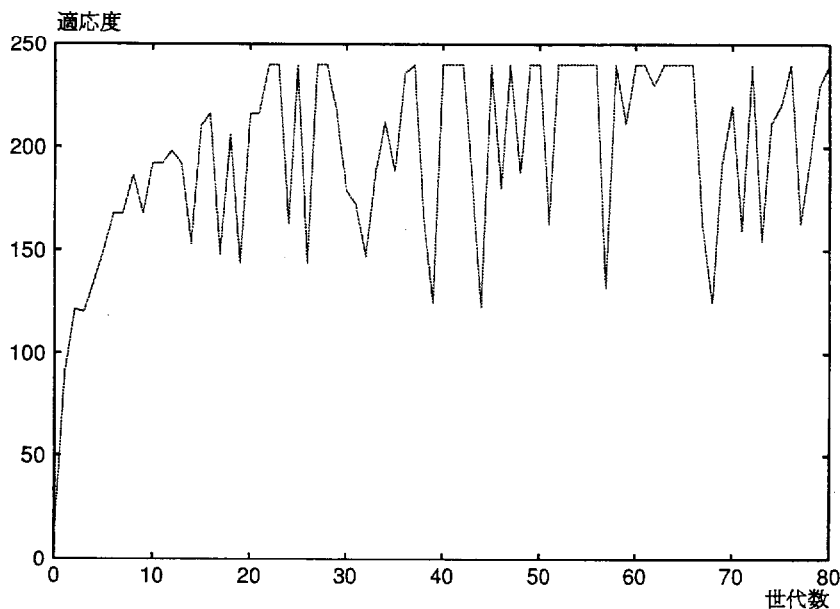


図 6.2: 最大適応度の変化

前節で示した環境に ADG を適用した。この実験での各世代における最大適応度と平均グループ数の変化を図 6.2, 図 6.3 に各々示す。動的環境においても、高い得点をあげるように個体集団が進化し、2つのグループからなる構造に変化していくことが確認できる。なお、図 6.2 では、50 世代以降も世代ごとに、最大適応度の揺れが見られるが、これは各世代においてあらかじめ用意された重い荷物の個数が異なるためであり、エージェント群の行動は同一であることを確認した。

この終盤の最良個体の木構造を図 6.4 に示す。エージェントの搬送能力別に 2 つのグループに分かれ、能力に応じた仕事を行っていることが分かる。また、重い

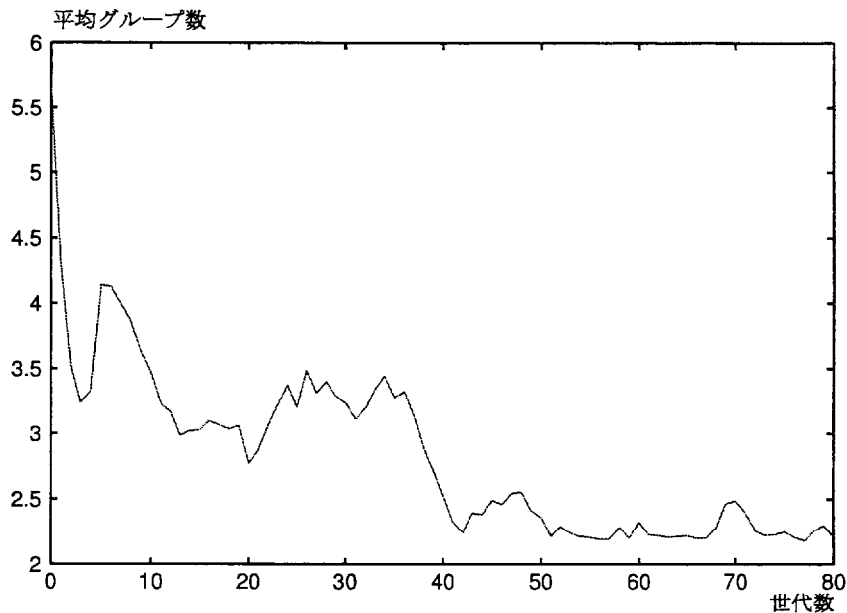


図 6.3: 各世代の平均グループ数の変化

- どちらの荷物も運ぶことができるエージェントは、木 0 を参照

木 0: Agent={0, 4, 8, 12, 16}

```
(If_carrying_load Move_goal (If_load_here
  Pick_up (If_fail Shift_to_group_1
    Move_heavy_load )))
```

- 軽い荷物しか運べないエージェントは、木 1 を参照

木 1: Agent={1,2,3,5,6,7,9,...,19}

```
(If_carrying_load Move_goal (If_load_here
  Pick_up Move_light_load ))
```

図 6.4: 獲得された木構造 (簡単化したもの)

荷物が尽きた場合は、その搬送を行っていたエージェントは、Move_heavy_loadの実行に失敗し、その結果軽い荷物を運ぶグループへ移動するという行動を獲得している。

このように、グループ移動記号を用いることにより、組織構造の変更が必要な動的環境へも適応できることを確認した。

6.2.4 グループ移動記号に関するまとめ

本節では、グループ移動記号を用いたグループ構造の変更方法について検討し、実験によりその有効性を示した。今後も、より効率的なグループ移動記号の設定方法について検討する必要がある。また、グループの再編が必要となる場合は、自らの行動の失敗の他にも、他グループのエージェントの故障や応援要請など、問題により様々である。今後はその判断基準の自動生成についても検討を行う必要がある。

6.3 内部状態ごとの行動最適化による動的環境への適応

6.3.1 内部状態を考慮したエージェントモデル

本節では、問題解決途中の組織構造の変更が可能になるように、内部状態ごとに行動を最適化し動的環境へ適応する方法について検討する。

例えば、荷物搬送問題で重い荷物を担当している単一のエージェントにおいて、順調に仕事を実行している時と、重い荷物が尽きて行うべき仕事が見つからず行動の失敗が続いた時とでは、異なる内部状態を持つとみなし、それぞれの状態に応じた行動規則を最適化することを考える。

ここでは、内部状態 j を持つエージェント i を一つのエージェントとみなし、 $A_{i,j}$ と表記する。物理的に存在するエージェントの数を N_a 、各エージェントにおいて考慮する内部状態の数を N_s とすると、総数 $N_a \times N_s$ のエージェントを ADG を用いて最適化することになる。

なお、以下では、物理的に存在する行動主体を「ロボット」と表記し、同一のロボットであっても内部状態の違いにより区別された行動主体を「エージェント」と表記することとする。

6.3.2 動的環境の設定

本手法の検証に使用した環境は以下のとおりである。

- 3種類の荷物と、搬送能力の異なる3種類のロボットを環境内に配置する。

ロボットは15体とする。各ロボットは内部状態を表す状態変数を持ち、この状態変数の取り得る値は0,1,2の三段階とする。

よって、この場合は、 $15 \times 3 = 45$ のエージェントの最適化を行なう。

また、各ロボットの搬送能力 p は以下のように設定する。

- 力の強いロボット ($p = 3$) = {ロボット 0, 3, 6, 9, 12}
- 普通のロボット ($p = 2$) = {ロボット 1, 4, 7, 10, 13}
- 力の弱いロボット ($p = 1$) = {ロボット 2, 5, 8, 11, 14}

また、3種類の荷物の重さ w は以下のように設定する。

- 重い荷物 : $w = 3$
 - 普通の荷物 : $w = 2$
 - 軽い荷物 : $w = 1$
- ロボット i の力を $p_i (1 \leq p_i \leq 3)$, 荷物の重さを $w (1 \leq w \leq 3)$ とするとき、 $p_i - w \geq 0$ ならばその荷物を搬送可能とする。
 - GP の終端・非終端記号には、表 6.2 に示す記号を用いる。
 - 作業開始時は各ロボットの状態変数は0に初期化され、作業を失敗した場合に自動的に増加されることとする。以下の4つの場合が作業の失敗に相当する。
 1. 自分の能力以上の荷物に対して荷物に対して、Pick_up を実行した場合。
 2. なくなった種類の荷物に対し、Move_*_load を実行した場合。
 3. 荷物が無い地点で、Pick_up を実行した場合。
 4. 既に荷物を持っているのに、Pick_up を実行した場合。
 - 重い荷物、普通の荷物はシミュレーション途中で消滅するように設定する。各 GP 個体の評価 (シミュレーションの実行) は、全 300 ステップであり、重い荷物は 100 ステップ、普通の荷物は 200 ステップの時点で消滅することとする。

表 6.2: 動的環境下の荷物搬送問題における終端・非終端記号の設定

記号	引数の数	機能
If_carrying_load	2	エージェントが荷物を持っているかどうかで分岐
If_load_here	2	現在地に荷物があるかどうかで分岐
Move_goal	0	ゴールの方へ向かう
Move_heavy_load	0	重い荷物の方へ向かう
Move_normal_load	0	普通の荷物の方へ向かう
Move_light_load	0	軽い荷物の方へ向かう
Pick_up	0	現在地に荷物があり、現在荷物を持っていない状態で、かつ、それを持つ能力があれば拾う

6.3.3 実験結果と考察

前節で示した環境に ADG を適用した。この実験での、各世代における最大適応度と平均グループ数の変化を図 6.5, 図 6.6 に各々示す。また獲得された最良個体の木構造を図 6.7 に示す。図 6.7 では、木構造プログラムを表す三角形の上側にあるエージェント群は状態変数 0, すなわち実行開始時のグループ構造を表しており、三角形の下側には内部状態が変化した状態のエージェントを記述してある。

この図から、例えばロボット 0 に関しては、以下のことが分かる。ロボット 0 は初期状態 ($A_{0,0}$) では、重い荷物の搬送を担当する。重い荷物が消滅した場合、Move_heavy_load の実行に失敗し、内部状態値が 0 から 1 へと変化する。よって、ロボット 0 は $A_{0,1}$ が参照している普通の荷物を搬送するグループへと移ることにより行動を変化させる。さらに普通の荷物が消滅すると、Move_normal_load の実行に失敗し、内部状態値が 1 から 2 へと変化する。よって、ロボット 0 は $A_{0,2}$ が参照している軽い荷物を搬送するグループへと移ることにより行動を変化させる。

なお、この最良個体において、例えばロボット 2 は初期状態から常に軽い荷物の搬送に従事するように行動の最適化がなされており、作業の失敗を経験することがない。そのため、 $A_{2,1}$ などは利用されることはない。図 6.7 では、このようなエージェント群については省略し、実際に利用されるエージェントのみを記述している。

このように、内部状態ごとに行動を最適化する方法では、GP の木構造のレベルでは、静的なグループ構造を利用しながらも、エージェントの実際の行動レベルでは、動的な環境に対応することが可能である。このような方法は、6.2 節で述べたグループ移動記号を木構造中に埋め込む方法と比較して、以下のような利点を持つと考えられる。

- 多数のエージェントの最適化が可能な ADG の利点の活用できる。
- 初期状態で同一の木を参照するエージェントでも、別々の木へ移る動作が可能であり、より多様な行動を生みやすい。

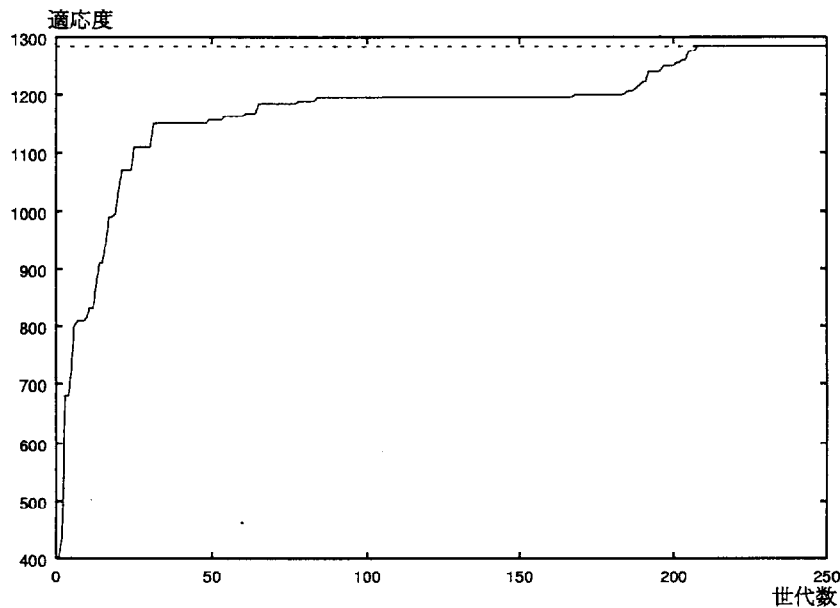


図 6.5: 最大適応度の変化

6.3.4 内部状態ごとの行動最適化に関するまとめ

本節では、内部状態ごとの行動最適化により動的環境に適応する方法について検討し、実験によりその有効性を示した。ただし、今回の実験では、内部状態として用意する状態数、および内部状態が変化する条件をあらかじめ設定しておいた。今後は、GP の記号設定に内部状態を操作する記号を追加し、適切な内部状態数、および内部状態が変化する条件を自動的に獲得する方法についても検討する必要がある。

6.4 おわりに

本章では、動的環境への適応を目指して、以下のような手法を用いたグループ構造の変更方法について検討し、実験によりその有効性を示した。

- GP における終端記号にグループ移動記号を設定する手法
- 同一のエージェントでも、内部状態が異なる場合は別のエージェントとみなし、一括して最適化する手法

ただし、現時点では、グループ構造の変更のためのエージェントの状態設定やエージェントの行動失敗に関する認識能力など、個別の問題に依存し人手の設計によるところも大きいのが現状である。今後、より汎用的なグループ構造変更の枠組について検討を行う必要がある。

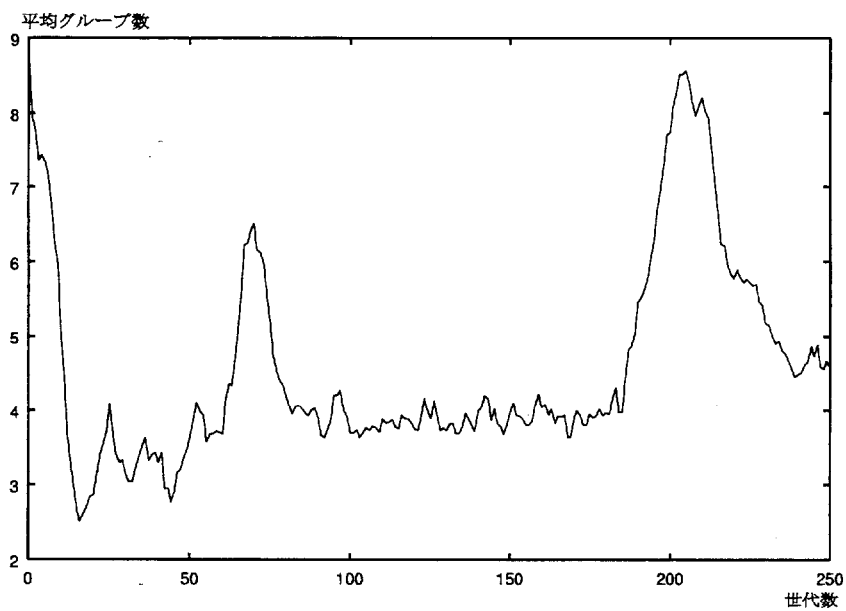


図 6.6: 各世代の平均グループ数の変化

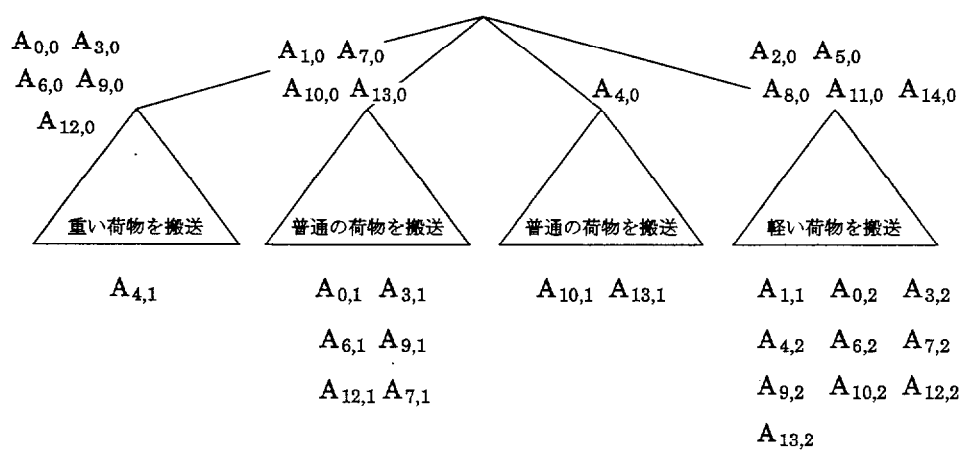


図 6.7: 獲得された木構造

第7章 ADGを用いた人工株式市場の構築と解析

7.1 はじめに

実社会における株式市場は、各投資家が様々な思惑のもとに行動し、取り引きなどの相互作用を通じてマクロな株価変動が現れる大規模なマルチエージェントシステムである。本章では、提案手法 ADG を人工市場研究に応用し、ADG がエージェント群の構造解析のツールとして有効であることを示す。また、現実的な株価変動を示す人工株式市場を構築し、それを利用して株価変動メカニズムの解析を行った結果について述べる。

7.2 人工市場研究の背景

従来の経済学では、市場に関する理論を数学的に展開するため、市場に存在する投資家に強い仮定を設定してきた。その代表的なものとして、合理的期待仮説や効率的市場仮説がある [40, 41, 42]。合理的期待仮説は、投資家を以下の特徴を持つ合理的経済人として仮定する。

目的合理性 自らの効用 (満足度) を最大化するという目的に対し、最も効果的な行為を選択する。

完全情報 行為に関する全ての選択肢とそれによって得られる結果、また不確実状況では全ての事象の生起確率を知っている。

決定の独立性 意志決定に際して、他の主体の行為から何らの影響を受けない。

完全な識別能力 事象のいかなる微小な変化も識別できる。

このような仮定は、市場に存在する全ての投資家が全て合理的に行動し、その予測に違いはなく均質的であるということを意味する。また、効率的市場仮説は、この合理的経済人の競争の結果により市場価格は全ての情報を瞬時に適切に反映するという仮説である。これは、突発的なニュースの発生にも価格は速やかに新たな水準へと調整され、独り抜け駆ける利益が得られるチャンスは、市場には一切残されていないことを意味している。

このような理想的な投資家や市場のモデル化は、自然科学的な経済理論の展開に大きく貢献してきた。しかし、1980年代以降、アメリカの株価の急騰やブラック・マンデーと呼ばれる大暴落、日本のバブルなどのように、従来の理論では説明が困難な現象がいくつも発生した。これは、従来の経済理論が、過度に理想的で非現実的なモデルを仮定していることや、投資家の個性や心理的な面を考慮していないことが原因である [42, 41, 43]。このような現状から、より現実的な市場のモデル化を目指した人工市場研究と呼ばれるアプローチが近年注目を集めている。次節では、従来研究とともに本研究のアプローチについて述べる。

7.3 本研究における市場研究へのアプローチ

近年では、計算機内に取引に関する戦略を持った仮想エージェントを作成し、これらのエージェント群の取引行動によって起こる現象を観察することで実際の世界の経済現象の解析に応用する人工株式市場モデルの研究が行われている [44, 45, 46, 47, 48, 49]。

従来の多くの人工株式市場モデルにおいて、各エージェントは過去の変動等のデータから利益が上がる戦略を学習し、取引を行っている。これらの研究の特徴とその問題点、および本研究の方向性を次に示す。

- 「預貯金の利子と株の配当の比較を行うモデルが多い」

従来モデルの多くは常に利子と配当の大小関係を考慮に入れた予測を立て、利子に対してどの程度利益を得ることができるかを前提として株式の売買を行っている。しかし実際の市場では、キャピタルゲイン(売買によって得られる差額)を目的として取引を行う投資家が多く存在しており、常に利子と配当の比較を行うことが適当であるとは考えられない。

そこで本研究では、利子と配当を比較しないモデルで、複雑な変動を獲得できる人工株式市場の構築を行う。

- 「市場を判断するルールやその基準は市場設計者が与えている」

従来研究においては、投資家の予測のための判断材料の選択、およびその重視度を classifier system や遺伝的アルゴリズムにより自律的に獲得する手法もあるが、それらを用いて実際の取引を判断するルールおよび予想価格の決定方法は市場設計者が与えていることが多い。このような前提では、取引の判断方法の多様性の影響などを認識することができないという問題が起きる。従って、本研究ではエージェントの戦略決定に自由度を持たせるために、遺伝的プログラミング (Genetic Programming; GP) を用いる。

- 「価格変動自体について評価を行っていない」

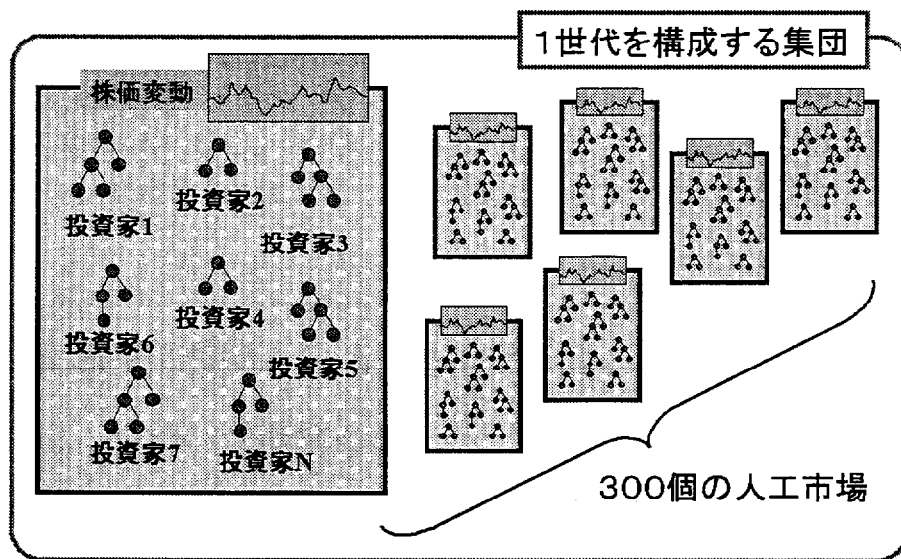


図 7.1: GP による人工市場の最適化の概念図.

進化的計算法を用いた人工株式市場の研究では、利益が上がるようなエージェントの戦略の学習、およびそれらの相互作用の結果生じる現象の解析を行うものが多い。その一方で、エージェント群の取引行動の結果として得られる市場の評価、つまり株価の変動について言及しているものは少ない。

本研究では、市場の構築にあたってエージェントの行動や利益は考慮に入れず、エージェント群により構成される人工市場の株価変動が、実際の市場の変動の特徴に類似するように最適化を行う。すなわち、図 7.1 に示すように、エージェント群全体からなる 1 つの市場を GP における 1 個体とし、現実的な株価変動を示す市場ほど適応度が高いものとして進化させる。

この手法は、人工社会のエージェント群を現実の社会指標に一致させるように進化させ、その結果得られたエージェント群を解析することにより、その社会現象の発生要因を解明するという逆問題的なアプローチである。文献 [50] では、このアプローチが電子コミュニティのネットワーク構造分析に適用され、マクロ社会現象の創発の解明手段として有効であることが示されている。本研究では、社会指標として株価変化率に関する統計量を採用し、人工市場における株価変動が実際の市場の変動に似た動きを見せるとき、市場に存在するエージェント群はどのような振る舞いをしているかを解析し、株価変動メカニズムの解明を目指す。

また、一般に、市場には多種多様な投資家が存在するが、GP を用いて各自のプログラムが異なるような不均質なエージェント集団を生成する場合、エージェントの数が多いと探索領域が膨大になり、最適化が困難になる。ADG を用いること

により、エージェント群をグループ化し探索領域を抑えて効率的に最適化することができる。また、得られたグループ構造はエージェント群の役割分担の理解のためにも重要である。本論文では、これまでGPを用いて行われてこなかった、異なる役割を担うエージェントグループの相互作用という観点から株価変動をとらえ、ADGにより獲得されたグループ構造を利用して市場の株価変動メカニズムの解析を行う。

7.4 本研究における人工株式市場の概要

7.4.1 GP を用いた戦略形成

株価変動、あるいは市場を構成する投資家の戦略をGPの木構造を用いて表現する研究はこれまでも報告されている [51, 52]。これらの研究では、GPの非終端記号に用いた三角関数に株価を代入するなどして複雑な戦略表現を可能にしている。しかし、2つの変化率の乗算や三角関数への株価や株価変化率の代入は、市場の動きを解析する上で、その戦略の意味付けが困難となる。本研究では市場の解析を行いやすいようなGPの記号設定を用いる。また、単一の株を扱い、扱うデータは株価変化率とする。株価変化率とは、時刻 t における株価 $Price(t)$ を用いて

$$Change(t) = \frac{Price(t) - Price(t-1)}{Price(t-1)} \quad (7.1)$$

と計算される。なお、本論文で扱う株価変動は日次データであるとする。

各エージェント i は自分の持つ予測木から予測変化率 $c_i(t)$ を獲得し、 $c_i(t)$ と現在の株価 $Price(t)$ から、明日の予測株価 $p_i(t+1)$ を以下の式に従って計算する。

$$p_i(t+1) = (1.0 + c_i(t)) * Price(t) \quad (7.2)$$

7.4.2 市場の構成

構築する人工株式市場の構成を以下に示す。

- エージェントが a 人、株が $a/2$ 個存在する。
- エージェントの保有できる株の最大数を1個とする。すなわち、エージェントの状態は、株を {持っている, 持っていない} の2通りである。
- エージェント i は、過去の変動を利用した予測木から予測変化率を計算し、それに基づいて予測価格 (p_i) を決定して市場に提示する。
- 市場は全員の予測を集計してエージェント i が

- 株を持っている場合,
株が p_i 円以上ならば売る
- 株を持っていない場合,
株が p_i 円以下ならば買う

という取引行動を取らせるようにする。すなわち、この予測価格 p_i は、エージェントが売買に参加する際の取り引き限度額を表しており、売り手ならばその値以上で、買い手ならばその値以下であれば取り引きに応じることになる。つまり、各投資家は、自分が予測する将来価格よりも、今高く売れるならば売り、安く買えるならば買うという動作を行う。

市場では、その日の全てのエージェントの売り注文と買い注文を突き合わせて売買契約を成立させる。具体的には、売り手側は安い希望価格を提示したエージェントを優先的に、買い手側は高い希望価格を提示したエージェントを優先的に取引に参加させ、売買の組を作る。買い手側の希望価格が売り手側の希望価格を上回る、あるいは、一致したときのみ取引が成立する。そして市場価格は最後に成立した売買の組の希望価格の平均値として決まる。

この方法は一般的には板寄せ方式と呼ばれるもので、市場価格は最大と取引数量が実現される価格、すなわち需給が一致する均衡価格となる citeUmart.

この市場を一定期間取引させることによって、株価の変動を獲得する。

7.4.3 投資家群のモデル化

実際の市場を構成する投資家の予測戦略は様々である。人工市場を構成するエージェント群についても、Homogeneous モデルではなく、Heterogeneous モデルのような同一の状況下でも各自が異なる戦略をとることが可能であるようなモデル化が必要とされる。

このように不均質なエージェント群のモデル化が必要とされる一方で、一人一人のエージェントの戦略が全て異なり、同じ戦略をとる者は全く存在しないとは考え難い。過去のトレンドを重視し、前日までの変動と同様の傾向が続くと予測する順張りエージェントや、これまでのトレンドと逆の方向へ株価が動くと予測する逆張りエージェント、過去の変動に関係なくランダムに予測をするエージェントなど予測戦略のタイプは大きくいくつかに分けられると考えられ、それらをタイプごとに一括して最適化できれば、探索領域の増大を抑えることができ効率的である。この場合、GP の一個体が市場の再現に必要なだけの複数の戦略木を持ち、各々の木は複数のエージェントにより参照される形となる。ただし、このようなモデル化を行うには、いくつかの戦略を用意し、どのようにエージェントを配分するかをあらかじめ決める必要があり、実際の適用は困難である。この解決策として、本研究では自動グループ構成手法 ADG を用いて最適化を行う。ADG を

表 7.1: 株価変化率に関するパラメータ.

パラメータ	正規乱数	実際の市場
自己相関	0.0	0.05 ~ 0.1
二乗変化率の自己相関	0.0	0.4 ~ 0.6
$\text{Prob}\{ x < \sigma\}$	0.683	0.75 ~ 0.8
$\text{Prob}\{ x > 3\sigma\}$	0.003	0.01 ~ 0.02

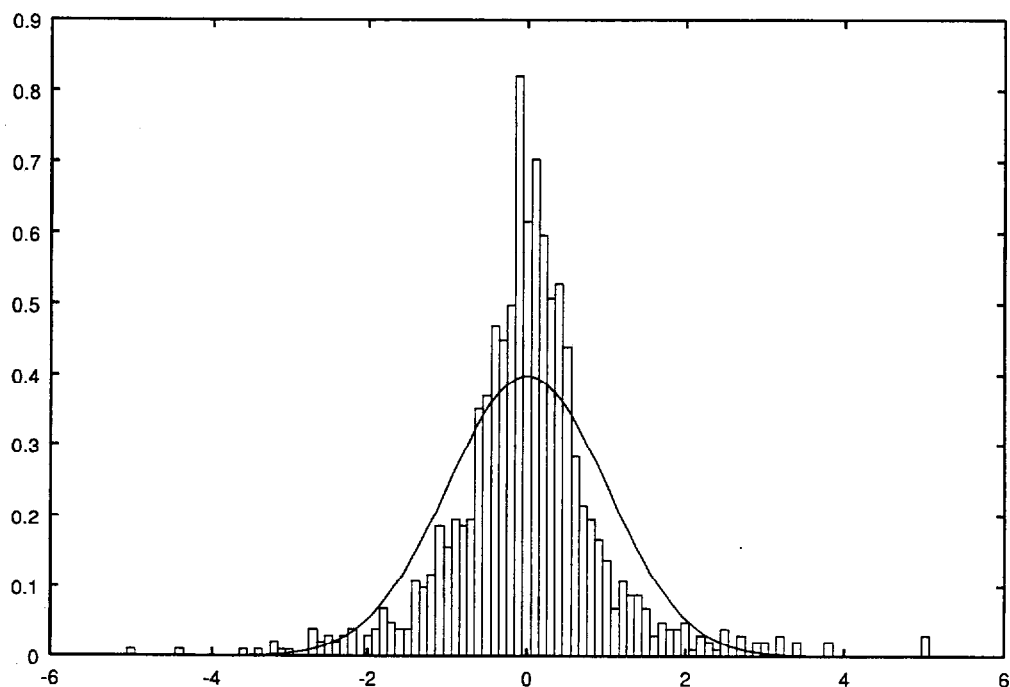


図 7.2: 基準化変化率のヒストグラム (TOPIX: '88~'93).

用いて、実際の市場に近い変動を見せる人工市場が獲得できたとすれば、そのときの GP 個体を見ることによって、人工市場内の戦略タイプ分類とその構成比を解析することが可能になる。

7.5 人工株式市場の構築

7.5.1 人工市場の現実性の評価方法

一般に株式市場の株価変化率について、表 7.1 のような特徴があることが知られている [53, 54, 55, 56].

つまり、株価変化率自体についてほとんど相関性は見られないが、その変化率を二乗して得られる変動幅に関する時系列についてはかなり高い相関性が見られる。

表 7.2: 株価変化率に関するパラメータ (その 2) .

パラメータ	実際の市場
変化率の平均	-0.0003 ~ 0.0003
変化率の標準偏差	0.010 ~ 0.015
最大パワースペクトル	0.001 以下

このことは、株価には、激しい変動の後には激しい変動が、小さな変動の後には小さな変動が続く傾向があることを表している。また、株価変化率の標準偏差 σ を用いて、株価変化率の絶対値が σ より小さくなる確率、 3σ より大きくなる確率を調べると、ともに正規分布よりも高い値をとる。この分布を詳しく見るため、実際の市場における変化率を平均 0.0、標準偏差 1.0 に正規化した基準化変化率のヒストグラム、および標準正規分布のグラフを描いたものを図 7.2 に示す。この図からも分かるように、正規分布と比較すると、高い頂点、厚い裾野 (High peak and Fat tail) を持つことが分かる。

また、(1) 式で定義した変化率に関して、過去の実際の株価変動の統計量を調べたところ、表 7.2 の結果を得た。平均値に比べて標準偏差が大きいこと、また、著しい周期性を持たないことが分かる。

そこで本研究では、木構造の予測を持つエージェント集団から構成される市場を構築して、これらのエージェントに株の売買を行わせて得られた株価の変動が表 7.1、表 7.2 の実際の市場のパラメータの条件を満たすことを目標に最適化を行う。すなわち、GP の個体は各々異なる市場を表し、各市場について表 7.1、表 7.2 に示したパラメータを計算し実際の市場との二乗誤差の重み付け線形和をとったものを適応度とする。この誤差を小さくすることで、現実的な株価変動を示す人工市場の構築を行う。

7.5.2 GP の終端・非終端記号の設定

人工株式市場を構成する各エージェントの予測の木に用いる記号を表 7.3 に示す。終端記号 RND ではランダムな予測をすることとし、正規分布に従った乱数を返す。今回の実験では、平均 0.0、標準偏差 0.05 の正規分布を用いた。ただし、市場内に存在するランダム予測を同じものとして扱い市場の解析を行いやすくするため、過去の変動とは無関係の記号である RND に対しては、rev, *2, /2 操作は行わないこととする。また終端記号 if>big_change における Cth には、実際の市場の株価変化率の標準偏差の 2 倍にあたる 0.02 を用いた。

表 7.3: 人工株式市場における終端・非終端記号設定 ($arg0, arg1, \dots$ は非終端記号の引数を表す.)

記号	引数	機能
if>	4	if ($arg0 > arg1$) return $arg2$ else $arg3$
if>big_change	2	if ($ CHG > C_{th}$) return $arg0$ else $arg1$
rev	1	$-arg0$
*2	1	$2 \times arg0$
/2	1	$arg0/2$
CHG	0	前日からの変化率
RND	0	正規分布に従う乱数
ZERO	0	0

表 7.4: 市場および GP のパラメータ

エージェント数	20
株の数	10
取引期間	1024
GP 個体数	300
選択	トーナメント +エリート保存
トーナメントサイズ	5
交叉率	0.9
突然変異率 (突然変異が 適用される個体の割合)	0.9
グループ突然変異率	0.05

7.6 実験結果と考察

7.6.1 獲得した人工市場とその株価変動

実験で使用した市場および GP のパラメータを表 7.4 に示す. このような設定の下で, ADG を用いて GP 個体集団を進化させた結果, 表 7.1, 表 7.2 に示した市場パラメータを満たすように最適化された最良個体を得られた. この市場における株価の例を図 7.3 に, このときの株価変化率, および二乗変化率の時系列を図 7.4, 図 7.5 にそれぞれ示す. この人工市場の株価時系列における統計量は表 7.5 のようになっており, 最適化に用いた条件をほぼ満たしている. また, 図 7.5 から分かるように, 部分的に大きな変動が続いたり, 小さな変動が続いたりする期間が存在し, 実際の市場と同様の変動が見られた. 基準化変化率のヒストグラムに関しては, 図 7.6 のように, 高い頂点, 厚い裾野を持ち, 実際の市場に近い分布を示した.

表 7.5: 人工市場の株価変化率に関する統計量.

パラメータ	人工市場
自己相関	0.10
分散の自己相関	0.41
$\text{Prob}\{ x < \sigma\}$	0.78
$\text{Prob}\{ x > 3\sigma\}$	0.019
変化率の平均	0.0003
変化率の標準偏差	0.010
最大パワースペクトル	0.0003

7.6.2 人工市場におけるグループ構造の解析

市場パラメータを満たし、前節で示したような株価変動を見せる人工市場におけるエージェント群には様々な構成が見られた。それらのエージェント群のグループ構造、および予測戦略の例を図 7.7 に示す。市場統計量を満たし現実的な株価変動を見せる市場に見られる特徴として、次のようなものが挙げられる。

- 図 7.7 における市場 1 の予測木 1 や市場 2 の予測木 1 を参照するグループのように、直前の変動と同一あるいはその 2 倍を予測値とするエージェントが存在する。これらのエージェント群は、前日に価格が上がったときに次も上がると予測し、価格が下がったときに次も下がると予測する、いわゆる順張りの戦略をとる。これらのエージェントが変動に過剰に反応することが、二乗変化率の自己相関が高くなったことの 1 つの要因であると考えられる。
- 図 7.7 における市場 1 の予測木 2 や市場 2 の予測木 3 を参照するグループのように、直前の変動幅が大きい場合に、その変動の 0.25 倍を予測したり、価格変化無し (0 予測) と予測するエージェントが存在する。これらのエージェント群は、前日に大きな変動が発生しても今日はあまり変化しないであろうと予測する、いわゆる控え目な戦略をとる。これらは大きな変動を抑制する働きを持つと考えられる。

これらの結果から、現実的な株価変動は、順張り予測を行うエージェント群、控え目な予測を行うエージェント群、ランダム予測を行うエージェント群の 3 つのグループの相互作用によって、形成できることが分かる。

次節以降では、この典型的な形を表している図 7.7 の市場 1 を利用し、株価変動メカニズムの解析を行う。

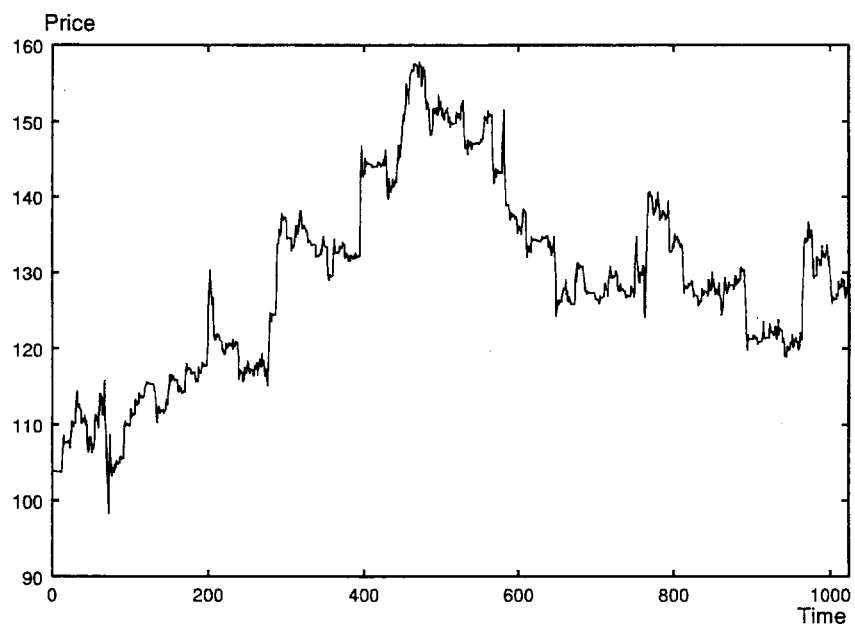


図 7.3: 人工株式市場の株価の推移.

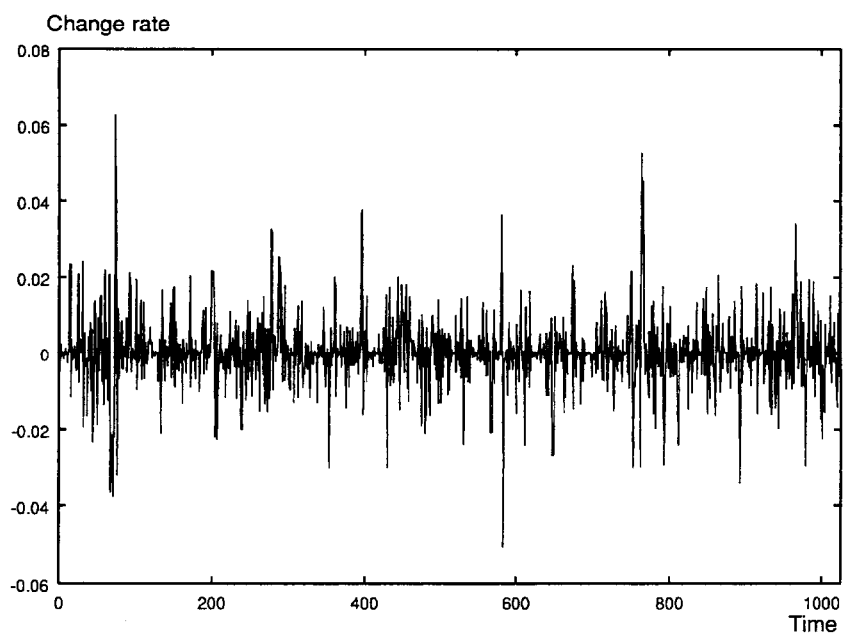


図 7.4: 人工株式市場の株価変化率の推移.

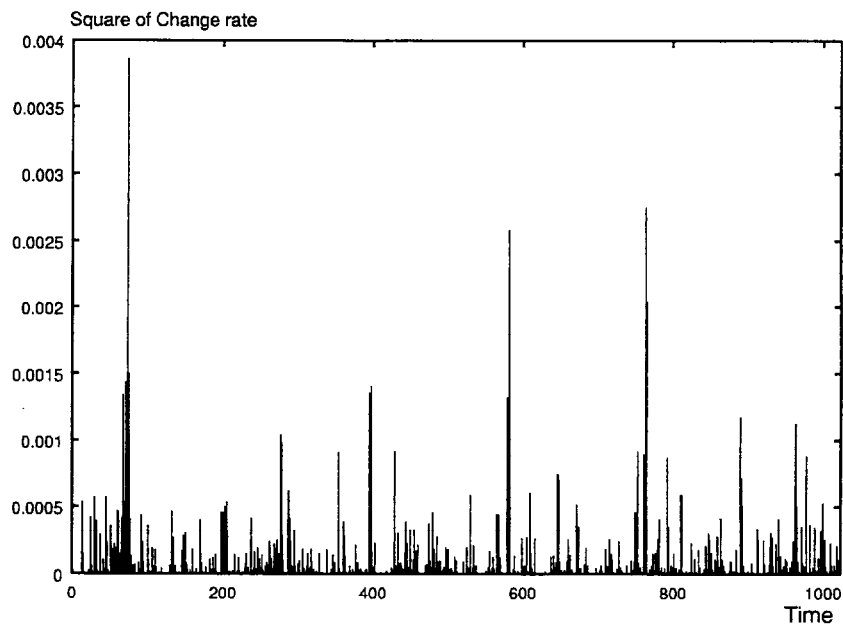


図 7.5: 人工株式市場の二乗変化率の推移.

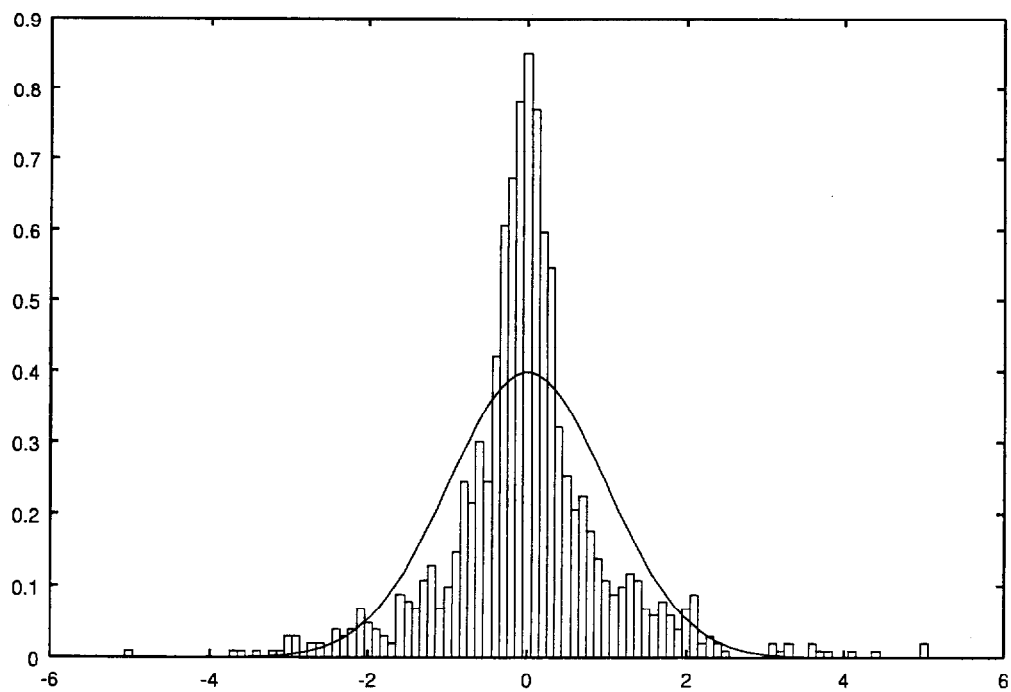


図 7.6: 人工市場における基準化変化率のヒストグラム.

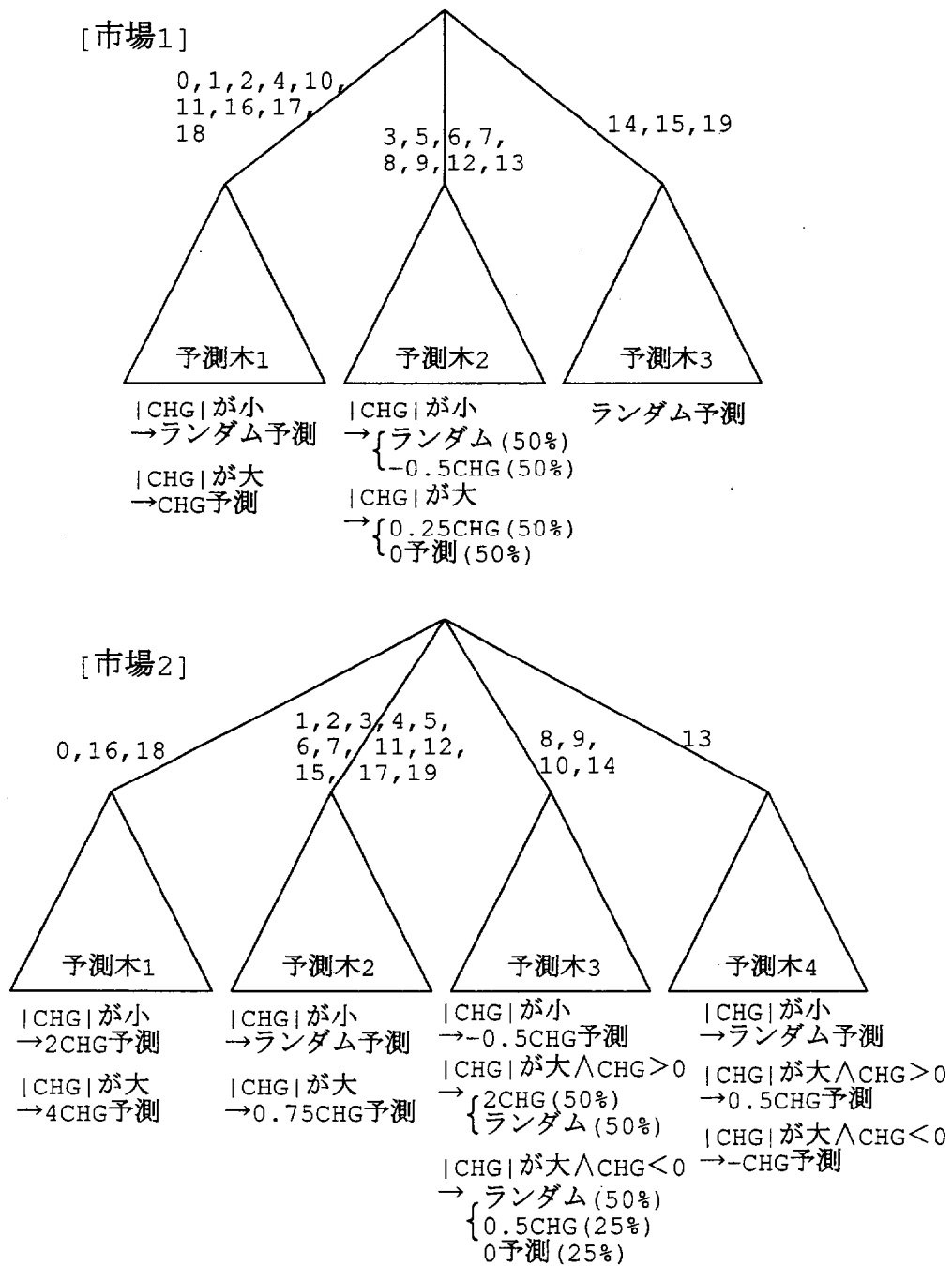


図 7.7: パラメータをほぼ満たす市場の行動分析例.

表 7.6: 各設定における株価変化率に関する統計量.

パラメータ	Case 1	Case 3
自己相関	-0.02	-0.19
二乗変化率の自己相関	0.05	0.08
$\text{Prob}\{ x < \sigma\}$	0.67	0.88
$\text{Prob}\{ x > 3\sigma\}$	0.004	0.03

7.6.3 各グループの役割の検証

各グループの役割を検証するため、図 7.7 に示した市場 1 のエージェント群から、各グループを取り除いて取り引きを行わせ、その様子を観察した。ただし、市場を構成するエージェント数を同一に保つため、取り除いたグループに所属していたエージェントは、エージェントの構成比を維持するように残りのグループに移動させることとし、以下の 3 つの場合について取り引きを行わせる。

Case 1 予測木 1, 2 を除去し、それらのグループに所属していたエージェント群を全て 3 のグループへ移動。(全員ランダム予測)

Case 2 予測木 2 を除去し、そのグループに所属していたエージェント群を、予測木 1, 3 のグループへ移動。

Case 3 予測木 1 を除去し、そのグループに所属していたエージェント群を、予測木 2, 3 のグループへ移動。

まず Case 2 の市場では、株価は指数関数的に増加し現実的な株価変動は得られなかった。この結果から、Case 2 で除去された予測木 2 のグループは大変動を抑える効果をもつことが分かる。また Case 1, Case 3 におけるシミュレーションにより得られた株価変化率の統計量を表 7.6 に示す。全員が正規分布に従うランダム予測を行うエージェント群からなる Case 1 の市場では、その株価変動の変化率の分布も正規分布となり、市場の特徴である高い頂点、厚い裾野を持つような分布は再現できなかった。また、Case 3 の市場では株価変動の二乗変化率の自己相関が減少した。Case 3 における株価および二乗変化率の時系列の例を図 7.8, 図 7.9 にそれぞれ示す。ランダム予測の影響により大変動も見られるが、これは一時的なものであり大変動が続く期間は見られないことが分かる。このことから、予測木 1 は過去の変動に過剰に反応し大変動を持続させることで、二乗変化率の自己相関を高める効果を持つことが分かる。

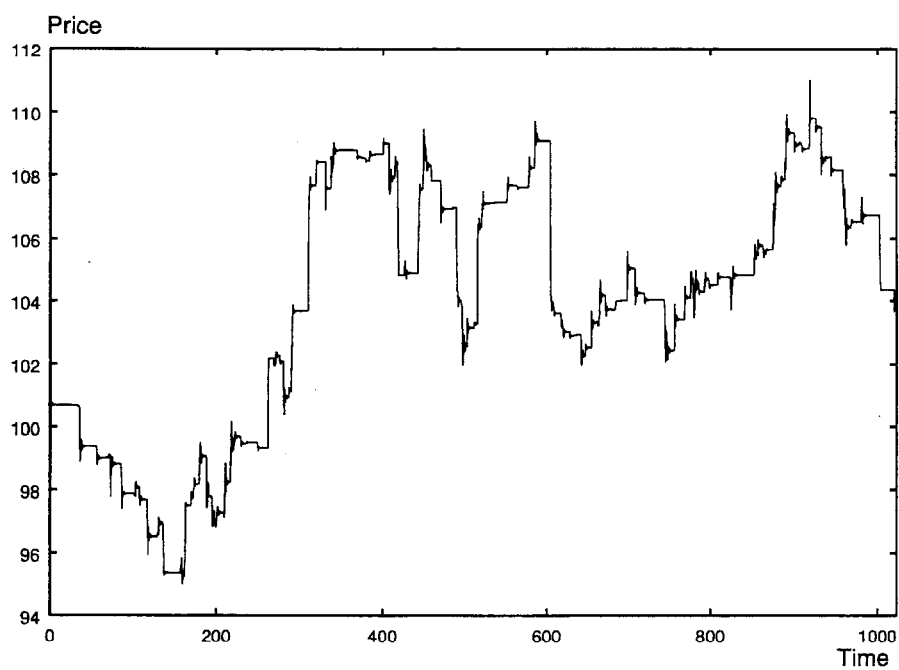


図 7.8: Case 3 における株価変動.

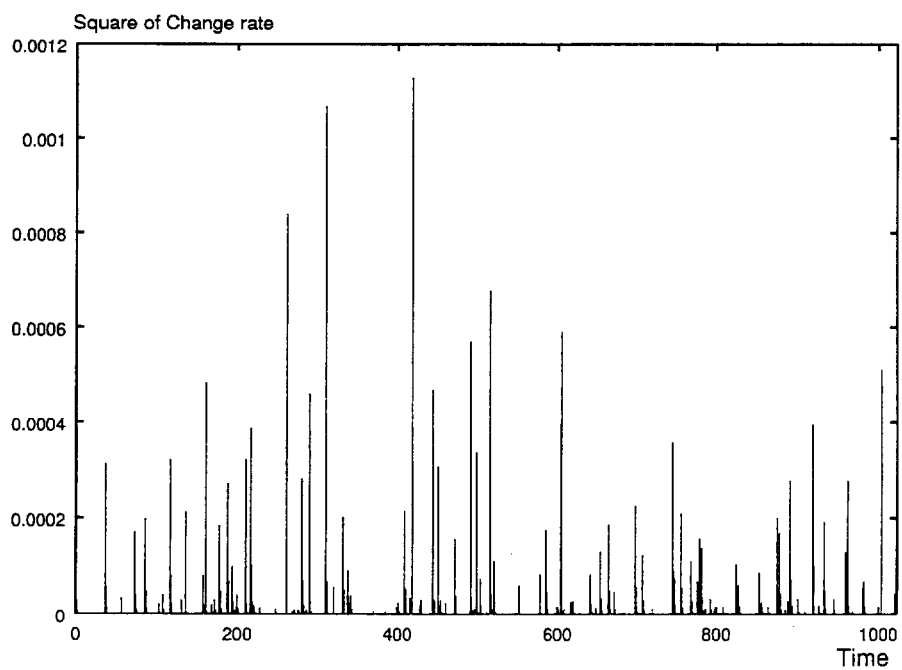


図 7.9: Case 3 における二乗変化率の推移.

7.6.4 株価の大変動の発生メカニズムの解析

図 7.7 に示した市場 1 において、株価の暴落時の変動を取り上げ、そのときの各エージェントの取引の様子を観察した。その結果、株価の持続的な大変動の発生のメカニズムは、以下の 3 つのフェーズから構成されていることが確認できた。

1. 突発的な大変動の発生
2. 大変動の持続
3. 大変動が収まり、株価が安定へ

この一連の流れを、図 7.10 に示した人工市場の株価変動における 880 日目から 884 日目に発生した株価の暴落の際の取引を例に挙げて説明する。また、付録 C に、この時の各エージェントの取引の様子を示す。

突発的大変動の発生の要因

図 7.7 の市場 1 において、各エージェントの行動を見ると、前日の変動幅が小さい場合は、どのグループもほぼランダム予測戦略をとり、グループ間での戦略に差は見られない。そのため、株価変化率が小さな値で安定している期間では、売り手側・買い手側双方に、各予測戦略グループのエージェントが混在することになる。これは、付録 C における 877 日～879 日の取引の様子からも確認できる。

ただし、このような状況下で、880 日目に大幅な下落が発生している。これは、一部のエージェントがランダム予測によって、偶然に大幅な変化率を予測し、その取引が成立したことが原因である。付録 C における 880 日目の取引から、その様子が確認できる。

大変動の持続の要因

図 7.7 の市場 1 において、各エージェントの行動を見ると、前日の変動幅が大きい場合は、予測木 1 を参照するエージェント群は順張り予測戦略をとり、予測木 2 を参照するエージェント群は控え目な予測戦略をとる。よって、前節のような原因で一旦大変動が発生すると、次の日から市場のエージェント群は各自のグループに応じた個性的な戦略をとるようになる。なお、暴落時においては、順張り予測は引続き価格が下がり続けるであろうと予測することに相当し、控え目な予測は今後はそれほど価格は下がらないと予測することに相当する。

市場では一般に、売り手側は安い価格でも売ろうとする順張りエージェントが優先される。また、買い手側は今後の変化を控え目に予測するエージェントの方が、順張りエージェントよりも高めに希望価格を提示するため、控え目なエージェントの方が優先される。このため、取引可能なエージェントの組合せは、ランダ

ム予測エージェントを除くと（売り手側，買い手側）＝（順張り，順張り），（順張り，控え目），（控え目，控え目）の3通りである。

特に，順張り予測エージェント同士の取引は，どちらも大幅な安値で取引に応じるため，大変動を維持する効果を持つ。この取引が連日発生することで，大変動が持続する。このことが，株価変動の変動幅の相関が高いこと，および株価変化率の分布が厚い裾野を持つことの原因であると考えられる。

このように大変動を維持しつつも，売り手側は順張り予測エージェントが優先され，買い手側は控え目予測エージェントが優先されることから，日が経つにつれて，売り手側には控え目予測エージェントが集中し，買い手側には順張り予測エージェントが集中するようになる。

この順張り同士の取引，および各戦略タイプのエージェントの一方の側への集中の様子は，付録Cにおける881日～883日の取引の様子から見る事ができる。

大変動期から価格の安定へ移行する要因

前節のような取引が続くと，最終的には，売り手側には控え目予測を行うエージェント全員が集中し，買い手側には順張り予測を行うエージェント全員が集中する。この株価下落時においてこのような状況になると，売り手側の希望取引価格よりも買い手側の希望取引価格の方が安いいため，取引が成立しない状況に陥る。結果として，取引が発生しないため前日の株価から変化しないという状況，もしくは，ランダム予測同士の取引しか成立せず株価変動が小幅な値動きに留まる状況となる。

一旦株価の変化率が小さくなると，翌日の取引からは，順張り，控え目といった個性的な戦略をとることはなくなり，また全員がランダム予測のような動きをすることになる。これによって株価が安定する。これが大変動期から価格の安定期へ移行する要因であると考えられる。

大暴落発生メカニズムのまとめと現実への関連付け

前節までの結果をまとめると以下のようなになる。

獲得した市場においては，最初の突発的な価格の大幅な下落は，ランダム予測エージェントの取引によるものであった。しかし，その後に引続き起こる下落は，順張り予測エージェントおよび控え目な予測エージェントの相互作用により発生している。大幅な下落時に見られる典型的な取引の様子は以下のようなものであった。

1. 株価の安定期

どのグループもランダム予測の傾向を持ち，グループ間での大きな取引の差は見られない。

2. ランダム予測を行うエージェントが大きな変化率を予測し、その取引が成立することにより大幅な下落が発生

この大変動を受けて、翌日の取引では、各エージェントは順張り予測戦略、あるいは控え目な予測戦略という特徴的な傾向を持つようになる。

3. 順張りエージェント同士の取引により大変動が持続。

順張り同士の取引により大変動を維持しながらも、次第に売り手側には控え目なエージェントが集中し、買い手側には順張りエージェントが集中していく。

4. 売り手側、買い手側それぞれへ、各取引戦略グループの集中が完了すると、取引が成立しない状況に陥り、株価は停滞する。

5. 一旦変動が収まると、どのグループもランダム予測の傾向を持つようになる。その結果、株価変動が安定する。

逆に、大幅な株価上昇の局面では、順張り予測同士の取引により大変動を維持しながらも、順張りエージェントは売り手側に、控え目エージェントは買い手側に次第に集中していく。その結果、最終的には取引が成立しない状況に陥り、株価はそこで安定する。

このように、一旦ランダム予測の影響により大変動が発生すると、各予測戦略グループが、買い手側、売り手側へと集中していきながらも、順張りエージェント同士の取引が行われる過程により大変動が持続される。このことが、株価変動が厚い裾野 (Fat tail) および二乗変化率の相関の高さ (大変動がしばらく続く傾向) を示す原因であると推測できる。

この過程は、要人発言のような過去の変動によらない突発的なニュースにより大変動が発生し、それが生み出したトレンドに追随しようと投資家群が順張り戦略をとることにより価格の高騰・暴落が発生する状況に相当すると考えることもできる。

7.7 おわりに

本研究では、計算機内に木構造で表される戦略をもつエージェント群を作成し、これらの株式取引の結果得られる株価の変動が実際の株式市場の変動の特徴に類似するように最適化することを目指した。そこで人工株式市場を構成するエージェントを対象とした効率的な戦略形成とその解析を目的として、自動グループ構成手法 ADG を適用した。そしてエージェント群からなる人工株式市場の最適化を行い、実際の市場に似た複雑な変動を見せる市場を得た。解析の結果、各々の役割を担ったグループの相互作用により、二乗変化率が高い自己相関を示すこと、および株価変化率の分布が厚い裾野を持つことを示した。

しかし、本研究におけるエージェントの予測戦略は、過去の変動のみに基づいたものであり、過去の変動を重視する順張りグループや、大変動を抑制するグループがどういった要因から発生したかについては触れていない。このような群行動はどういったメカニズムから生まれるのか。リスク回避や収益を上げている他のエージェントへの同調といったエージェントの心理的側面も含めた市場の最適化が今後の課題となる。

また、実際の市場を再現するには現在与えているパラメータの最適化のみでは不十分である。今後は、経済物理学 (econophysics) [54, 57] の研究成果などを参考に、人工市場と現実市場の株価変動の類似度の測定方法の確立が重要となる。また、現状では市場の設定やエージェントの利用できる GP の終端・非終端記号設定などは非常に単純化したものであり、より現実的で複雑な戦略をとることが可能な市場設定で実験を行う必要がある。

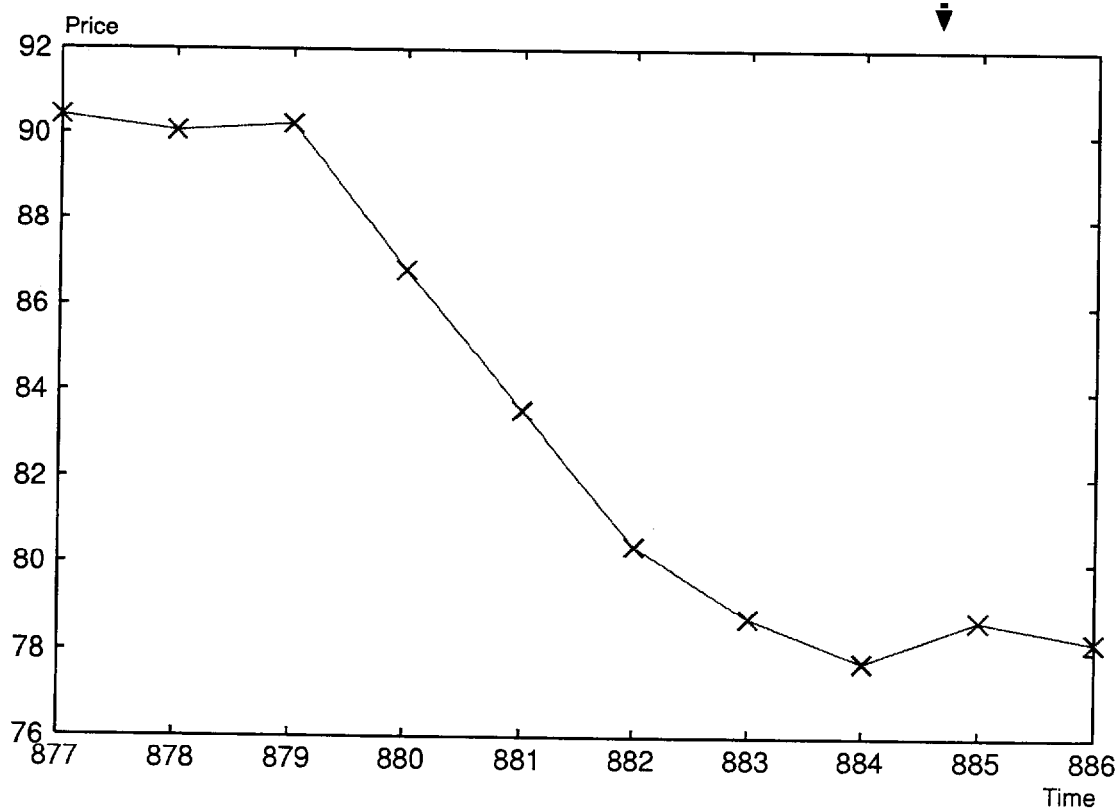
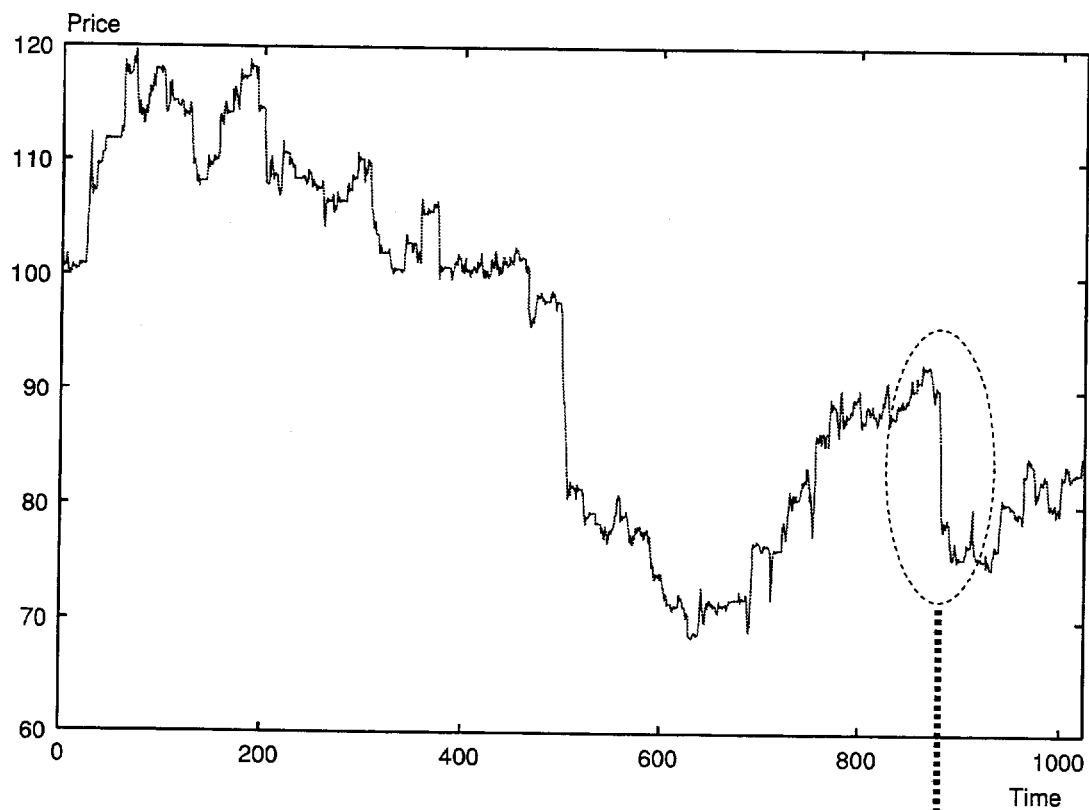


図 7.10: 人工市場の株価変動の例.

第8章 結論

8.1 まとめ

本研究の成果として、以下の4点が挙げられる。

1. マルチエージェントに協調行動を行わせる際に重要となるグループ構造および各グループのプログラムを、進化的に生成する手法 ADG を提案したこと。
2. マルチエージェント群の代表的な問題領域である荷物搬送問題やタイルワールド環境に提案手法 ADG を適用し、有効性を検証したこと。
3. 動的環境への適用を目指して、ADG のグループ構造を利用した組織構造の変更方法について提案し、有効性を検証したこと。
4. 実社会における大規模マルチエージェントシステムである株式市場の解析に提案手法 ADG を適用し、ADG が複雑系の解析のツールとして有効であることを示すとともに、市場の株価変動メカニズムの解明のためのモデルを構築したこと。

以下で、各々の成果を総括し、今後の課題について述べる。

8.1.1 ADG の提案について

協調行動を行うマルチエージェントシステムにおいては、相互作用などを考慮する必要があるため、人手による設計は非常に困難であり、進化などにより各自の行動を自律的に獲得することが望まれる。

本論文では、進化により大規模エージェント群を最適化する際の重要な点であるエージェント群のグループ構造に着目し、以下の3点を自動的に獲得する手法 ADG を提案した。

- エージェント群をいくつかのグループに分けるか。
- 各エージェントをどのグループに割り当てるか。
- 各グループの行動決定プログラム

ADG におけるグループ構造の最適化は以下のような操作から構成される。

- 子への形質保存, および各エージェントの特殊化という観点から, Restricted Breeding を応用したグループ分割操作
- 子への形質保存の観点からのグループ統合操作
- 個体集団の多様性の維持の観点からのグループ突然変異

一般に, GA や GP などの進化的計算法では, 個体群の多様性の維持, および形質遺伝の観点が重要である. ADG における遺伝操作も進化的手法の設計指針にのっとったものとなっている.

8.1.2 マルチエージェント問題への適用による ADG の有効性の検証について

マルチエージェント群の代表的な問題領域である荷物搬送問題やタイルワールド環境に提案手法 ADG を適用し, その有効性の検証を行った. エージェント群のモデル化として従来使用されてきた Homogeneous モデルや Heterogeneous モデルと比較することにより, ADG が以下の利点を持つことを確認した.

- Homogeneous モデルよりも複雑な協調動作が実現可能であるとともに, Heterogeneous モデルよりも効率的に最適化が可能であること.

荷物搬送問題において, Homogeneous モデルではエージェントの能力差に応じて異なる行動をとることができないため ADG より低い性能に留まり, 一方, Heterogeneous モデルでは進化に ADG より多くの時間を要した.

- グループ化による効率的な問題解決が可能であること.

タイルワールド環境において, ADG ではグループ化により独立した3つの小グループが各々問題解決にあたることで, 従来モデルよりも高い性能を示した.

- 問題解決に必要な役割分担数を, 獲得されたグループ構造から明示的に確認できること.

荷物搬送問題およびタイルワールド環境において, 獲得されたグループ構造を見ることで, エージェント群がいくつのグループに分かれているか, どのエージェントが同じ役割を担っているかを容易に判断することが可能であった.

8.1.3 グループ構造の変更方式の検討について

問題解決途中においてもグループ構造の変更が必要とされるような動的環境を想定し, グループ構造の変更方式の検討を行った.

具体的な変更方式として、以下の2つの手法を提案し、動的環境下における荷物搬送問題において有効性を検証した。

- GPにおける終端記号にグループ移動記号を設定する方法
- 同一のエージェントでも、内部状態が異なる場合は別のエージェントとみなし、一括して最適化する方法

8.1.4 人工株式市場の構築について

現実社会におけるマルチエージェントシステムである株式市場を取り上げ、ADGを用いて株価変動メカニズムの解析を行った。この実験に関する成果として以下の点が挙げられる。

- 逆問題のアプローチの有効性を示した。

現実的な株価変動というマクロ現象を再現するエージェント群を進化的に生成し、それらエージェント群のミクロな行動を解析することによりマクロ現象の創発のメカニズムを解明するという手法を用いた。この逆問題のアプローチが、多様な行動主体からなる社会システムの解析に有効であることを示した。

- ADGが複雑系の解析のためのツールとして有効であることを示した。

ADGの利点の1つに、問題解決に必要な役割分担を、獲得されたグループ構造から明示的に確認できる点がある。株式市場のようなグループ構造が未知である社会システムの構築にADGを利用し、獲得されたグループ構造を解析することで新たな知見を得た。このように、ADGは複雑系の解析のためのツールとしても有効であることを示した。

- 株価変動メカニズムの解明。

1. 不均質な投資家群からなる市場モデルの提案。

従来の経済学では、全て合理的で均質な投資家群を想定していた。本研究では、現実的な株価変動が、順張り予測、控え目な予測、ランダム予測という3種類のエージェント群の相互作用により構成できることを示した。

2. 現実の市場の特徴である変動幅の相関が高い点、および変化率の分布が厚い裾野を示す点についての検証。

ADGで獲得された、順張り予測、控え目な予測、ランダム予測というグループ構造を利用して解析を行った。現実の市場に見られる、変動幅が高い相関を持ち、変化率の分布が厚い裾野を示すというマクロな現象

を、エージェント群の取引行動の相互作用というミクロな観点から説明した。

8.2 今後の課題

今後は、エージェントの数、必要な役割分担数、協調の形態などが異なる様々な問題に対して本手法を適用しその有効性を示すとともに、ADGの最適解への収束性やグループ分割・統合操作の効率の理論的な解析を行なう必要がある。

またGPでは、エージェントの基本行動要素として終端・非終端記号を与え、それらを組み合わせることにより環境に適応したプログラムを獲得するが、現状では、その記号設定は設計者の手に委ねられており、普遍的な設計方法は存在しない。これはGPの問題の1つである。一般には、基本行動要素をあまりにプリミティブなものにすると、進化に非常に時間がかかるか、進化の結果、高度な機能が発現されない場合が多い。逆に、基本行動要素を高度なものにすれば、高度な作業の実現が期待できるが、天下りの設計との指摘を受けるし、実現できる作業の一般性に難が残る [58]。使用する基本行動要素を、処理目的に応じて自律的に設定する方法論を確立することは今後の大きな課題である。

一方で、4章や5章の実験で示したように、同一の記号集合を用いても、交配戦略 (ADG, Homogeneous, Heterogeneous) の違いにより性能に差が現れた。交配戦略と記号集合の設定には密接な関係があると考えられる。今後は、GPにより実現できる作業の一般性を高めるとともに、ADGによるグループ構造を有効に活用できるような終端・非終端記号の設定方法についても検討する必要がある。特に、グループ構造の動的な再編は重要な課題である。6章でその方式について検討したが、グループ構造の変更の手段として、エージェントの状態設定やエージェントの行動失敗に関する認識能力といったような個別の問題に依存したものを利用しており、人手の設計によるところが大きいのが現状である。今後、より汎用的なグループ構造変更の枠組について検討を行う必要がある。

以上で述べたような、進化的手法の改善、およびそれに代わる新たな手法の提案という基盤技術の研究とともに、ここで得た手法を実社会へ応用していくことも重要である。本論文で提案した手法は、マルチエージェントの行動制御、および複雑系の解析に有効なものである。実社会におけるマルチエージェント領域は、マルチロボット環境はもちろん、7章で扱った株式市場、あるいは企業組織の運営、社会心理学など幅広く及ぶ。進化的なアプローチから得られた新たな知見を実社会へ活かしていくことも今後の課題である。

謝辞

本研究を進めるにあたり終始多大なるご指導を頂きました横浜国立大学の長尾智晴教授に深く感謝致します。また、著者を院生として受け入れて頂き、研究や学生生活へ貴重なご助言とご指導を頂きました長橋宏教授に深く感謝致します。また、本論文をまとめるにあたり、貴重なご助言とご指導を頂いた内川恵二教授、小林隆夫教授、伊東利哉教授、山口雅浩助教授に深く感謝致します。また貴重なご意見・ご指摘を頂きました長尾研究室の皆様に深く感謝致します。

参考文献

- [1] 油田信一. 複数の自律移動ロボットの協調行動. 日本ロボット学会誌, Vol. 10, No. 4, pp. 433–438, 1992.
- [2] 新井民夫, 太田順. 複数移動ロボット系の計画. 日本ロボット学会誌, Vol. 10, No. 4, pp. 444–449, 1992.
- [3] Alexis Drogoul and Jacques Ferber. From tom thumb to the dockers: Some experiments with foraging robots. In *From animals to animats 2*, pp. 451–459. MIT Press, 1993.
- [4] 長田正編. 自律分散をめざすロボットシステム. オーム社, 1995.
- [5] 伊藤宏司編. 知の創発. NTT 出版, 2000.
- [6] 佐々木貴宏, 所真理雄. 進化的エージェント集団の動的環境への適応. コンピュータソフトウェア, Vol. 14, No. 4, pp. 33–46, 1996.
- [7] John R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.
- [8] Thomas Haynes, Roger Wainwright, Sandip Sen, and Dale Schoenefeld. Strongly typed genetic programming in evolving cooperation strategies. In L. Eshelman, editor, *Genetic Algorithms: Proceedings of the Sixth International Conference (ICGA95)*, pp. 271–278, Pittsburgh, PA, USA, 15–19 July 1995. Morgan Kaufmann.
- [9] 倉林大輔, 太田順, 新井民夫, 野口克行. 不均質な行動原理を有する移動ロボット群の協調作業. 計測自動制御学会論文集, Vol. 35, No. 11, pp. 1377–1384, 1999.
- [10] J. H. Holland. *Adaptation in Natural and Artificial Systems*. The Univ. Michigan Press, 1975.
- [11] D. E. Goldberg. *Genetic Algorithms in search, optimization, and machine learning*. Addison Wesley, 1989.

- [12] 伊庭斉志. 遺伝的アルゴリズムの基礎. オーム社, 1994.
- [13] 安居院猛, 長尾智晴. ジェネティックアルゴリズム. 昭晃堂, 1993.
- [14] 長尾智晴. 最適化アルゴリズム. 昭晃堂, 2000.
- [15] 北野宏明編. 遺伝的アルゴリズム. 産業図書, 1993.
- [16] 原章, 長尾智晴. 遺伝的アルゴリズムを用いた画像中の任意方向を向いた顔の候補領域の抽出. 日本印刷学会誌, Vol. 36, No. 1, pp. 42–49, 1999.
- [17] 伊庭斉志. 遺伝的プログラミング. 東京電機大学出版局, 1996.
- [18] 伊庭斉志. 進化論的計算の方法. 東京大学出版会, 1999.
- [19] 伊庭斉志. 遺伝的プログラミング入門. 東京大学出版会, 2001.
- [20] 青木紳也, 長尾智晴. 木構造状画像変換の自動構築法 ACTIT. 映像情報メディア学会誌, Vol. 53, No. 6, pp. 888–894, 1999.
- [21] 伊庭斉志. 遺伝的プログラミングとロボット. 日本ロボット学会誌, Vol. 15, No. 5, pp. 663–666, 1997.
- [22] Sean Luke and Lee Spector. Evolving teamwork and coordination with genetic programming. In John R. Koza, David E. Goldberg, David B. Fogel, and Rick L. Riolo, editors, *Genetic Programming 1996: Proceedings of the First Annual Conference*, pp. 150–156. MIT Press, 1996.
- [23] Hitoshi Iba. Emergent cooperation for multiple agents using genetic programming. In Hans-Michael Voigt, Werner Ebeling, Ingo Rechenberg, and Hans-Paul Schwefel, editors, *Parallel Problem Solving from Nature IV, Proceedings of the International Conference on Evolutionary Computation*, Vol. 1141 of *LNCS*, pp. 32–41. Springer Verlag, 1996.
- [24] Hitoshi Iba. Multiple-agent learning for a robot navigation task by genetic programming. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pp. 195–200. Morgan Kaufmann, 1997.
- [25] Thomas Haynes, Sandip Sen, Dale Schoenefeld, and Roger Wainwright. Evolving a team. In E. V. Siegel and J. R. Koza, editors, *Working Notes for the AAAI Symposium on Genetic Programming*, pp. 23–30, MIT, Cambridge, MA, USA, 10–12 November 1995. AAAI.

- [26] Thomas Haynes and Sandip Sen. Crossover operators for evolving A team. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, pp. 162–167. Morgan Kaufmann, 1997.
- [27] Terence Soule. Voting teams: A cooperative approach to non-typical problems using genetic programming. In *GECCO-99: Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 916–922. Morgan Kaufmann, 1999.
- [28] David Andre. The evolution of agents that build mental models and create simple plans using genetic programming. In L. Eshelman, editor, *Genetic Algorithms: Proceedings of the Sixth International Conference (ICGA95)*, pp. 248–255. Morgan Kaufmann, 1995.
- [29] Hitoshi Iba and Makoto Terao. Controlling effective introns for multi-agent learning by genetic programming. In *GECCO-2000: Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 419–426. Morgan Kaufmann, 2000.
- [30] Sean Luke. Genetic programming produced competitive soccer softbot teams for robocup97. In John R. Koza, Wolfgang Banzhaf, Kumar Chellapilla, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max H. Garzon, David E. Goldberg, Hitoshi Iba, and Rick Riolo, editors, *Genetic Programming 1998: Proceedings of the Third Annual Conference*, pp. 214–222. Morgan Kaufmann, 1998.
- [31] Kazuo Miyashita. Job-shop scheduling with genetic programming. In *GECCO-2000: Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 505–512. Morgan Kaufmann, 2000.
- [32] 宮下和雄. GP によるスケジューリングヒューリスティクスの生成. 信学技報 [人工知能と知識処理], Vol. 99, AI99-12, pp. 9–15, 1999.
- [33] Shinya Masunaga and Tomoharu Nagao. An action control model of agents based on evaluation of group behaviors. In *1997 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 3343–3348. IEEE Press, 1997.
- [34] 梶永慎哉, 長尾智晴. 隣接遺伝子間結合強度を用いたマルチエージェントの行動最適化. 信学技報, Vol. 97, No. COMP97(1998-3), pp. 1–8, 1998.

- [35] Josh C. Bongard. The legion system: A novel approach to evolving heterogeneity for collective problem solving. In *Genetic Programming: Third European Conference(EuroGP2000)*, pp. 25–37. Springer-Verlag, 2000.
- [36] Josh C. Bongard. Reducing collective behavioural complexity through heterogeneity. In *Artificial Life VII: Proceedings of the Seventh International Conference on Artificial Life*, pp. 327–336. The MIT Press, 2000.
- [37] M.E. Pollack and M. Ringuette. Introducing the tileworld: Experimentally evaluating agent architectures. In *Proceedings of the 8th National Conference on Artificial Intelligence*, pp. 183–189. MIT Press, 1990.
- [38] 大沢英一, 沼岡千里, 石田亨. サーベイ : 分散人工知能小問題集. マルチエージェントと協調計算 I, 日本ソフトウェア科学会 Multi Agent and Cooperative Computation '91, Vol. 41, No. 4, pp. 179–216, 1992.
- [39] 川村隆浩. マルチエージェントシステムにおける動的な組織構造変更方式の提案. 第5回マルチエージェントと協調計算ワークショップ (MACC'95) On line Proceedings, <http://www.csl.sony.co.jp/person/nagao/macc95/>, 1995.
- [40] E. F. Fama. Efficient capital markets: A review of theory and empirical work. In *Journal of Finance*, 25, pp. 383–417, 1970.
- [41] 田中泰輔. マーケットはなぜ間違えるのか. 東洋経済新報社, 1995.
- [42] Andrei Shleifer. *Inefficient Markets: An Introduction to Behavioral Finance*. Oxford University Press, 2000.
- [43] Lars Tvede. *THE PSYCHOLOGY OF FINANCE*. John Wiley & Sons, Ltd., 1999.
- [44] 和泉潔, 植田一博. 人工市場入門. 人工知能学会論文誌, Vol. 15, No. 6, pp. 941–950, 2000.
- [45] 佐藤浩, 久保正男, 福本力也, 廣岡康雄, 生天目章. 人工市場のシステム構造. 人工知能学会論文誌, Vol. 15, No. 6, pp. 974–975, 2000.
- [46] Koichi Kurumatani, Y. Koyama, Takao Terano, Hajime Kita, Akira Namatame, Hiroshi Deguchi, Yoshironi Shiozawa, and H. Matsubara. U-mart: A virtual stock market as a forum for market structure analysis and engineering. In *Proceedings of the Fifth Joint Conference on Information Sciences*, Vol. 2, pp. 957–960, 2000.

- [47] 高階知巳, 渡辺成良. マルチエージェントシステムの学習過程の定量化手法—人工株式市場モデルでの評価—. 計測自動制御学会論文集, Vol. 35, No. 12, pp. 1609–1616, 1999.
- [48] Shareen Joshi, Jeffrey Parker, and Mark A. Bedau. Technical trading creates a prisoner's dilemma: Results from an agent-based model. In *Santa Fe Institute working paper*, pp. 98–12–115, 1998.
- [49] Yorikazu Takao, Isao Ono, and Norihiko Ono. Constructing approximation models based on agent-based simulations by genetic algorithms. In *ICCIMA 2001: Proceedings of the International Conference on Computational Intelligence and Multimedia Applications 2001*, pp. 231–235. IEEE Computer Society, 2001.
- [50] 倉橋節也, 南潮, 寺野隆雄. 逆シミュレーション手法による人工社会モデルの分析. 計測自動制御学会論文集, Vol. 35, No. 11, pp. 1454–1461, 1999.
- [51] Hitoshi Iba. Bagging, boosting, and bloating in genetic programming. In *GECCO-99: Proceedings of the Genetic and Evolutionary Computation Conference*, pp. 1053–1060. Morgan Kaufmann, 1999.
- [52] S.-H. Chen and T.-W Kuo. Towards an agent-based foundation of financial econometrics: An approach based on genetic-programming artificial markets. In *GECCO-99: Proceedings of the Genetic and Evolutionary Computation Conference*, pp. pp.966–973. Morgan Kaufmann, 1999.
- [53] 山根裕明, 長尾智晴. 進化的計算法を用いた人工株式市場の構築. 信学技報 [人工知能と知識処理], Vol. 99, No. 718, AI99-100, pp. 9–16, 2000.
- [54] Rosario N. Mantegna and H. Eugene Stanley. *An Introduction to Econophysics*. Cambridge University Press, 2000.
- [55] 佃良彦. 金融時系列. 計測と制御, Vol. 39, No. 7, pp. 454–460, 2000.
- [56] Kiyoshi Izumi and Kazuhiro Ueda. Phase transition in a foreign exchange market — analysis based on an artificial market approach. *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, Vol. 5, No. 5, pp. 456–470, 2001.
- [57] 佐藤彰洋, 高安秀樹. 統計物理から見た人工市場. 人工知能学会誌, Vol. 15, No. 6, pp. 958–965, 2000.
- [58] 太田順, 新井民夫. 群ロボットシステムと創発. 日本ロボット学会誌, Vol. 15, No. 5, pp. 693–697, 1997.

付録

付録 A タイルワールドでの最良個体の木構造

エージェント 0, 6, 7, 9 は, 木 2 を参照

木 2:

```
(+ Nearest-Tile (->90 (inv (if>= (inv Last )Goal (*2 Goal ))(inv (if>= (->90 (+
(if>= (inv Last )(->90 (+ (->90 (if>= (ifdot (- (*2 Last ))(+ (+ Same-Group (+ (ifdot
Nearest-Tile (if>= (->90 Nearest-Tile ))(ifdot (ifdot Last Same-Group Nearest-Tile Last
))(*2 Last ))(- Same-Group (ifdot (/2 (->90 Goal ))(inv (if>= Nearest-Tile Goal Same-
Group Nearest-Tile )))Nearest-Tile (->90 (/2 (->90 (ifdot (- Nearest-Tile Goal )Goal
Goal (/2 Same-Group ))))))))Same-Group )Same-Group Last )(if>= (*2 (->90 Same-
Group ))(- (*2 (+ Last (->90 Same-Group )))Last )Same-Group (ifdot (+ (+ Nearest-
Tile Goal )Goal )Same-Group (*2 (+ (inv Nearest-Tile ))(- Same-Group Goal ))) (if>=
(ifdot Same-Group (- Goal Last ))(- Same-Group Nearest-Tile ))(+ Same-Group Last ))(+
Goal (/2 Same-Group ))Nearest-Tile Last )))Nearest-Tile (ifdot Last Last Same-Group
Nearest-Tile )))Goal ))(/2 (if>= Last Nearest-Tile Nearest-Tile Goal ))(inv Goal ))(+
(inv Same-Group ))(inv Same-Group )))Nearest-Tile Goal (*2 (if>= Goal (ifdot (/2 Last
))(ifdot (+ Goal Goal )Last (inv Nearest-Tile ))(ifdot (->90 (inv Same-Group ))(ifdot
(/2 Nearest-Tile )Nearest-Tile Nearest-Tile (+ Goal Same-Group ))(/2 (/2 Nearest-Tile
))Same-Group ))Same-Group (ifdot (inv Last ))(/2 Same-Group ))(/2 Goal ))(- (inv (/2
(/2 (/2 (ifdot (->90 Same-Group ))(*2 (if>= (- (if>= (ifdot (if>= Last Goal Goal
Last ))(/2 Nearest-Tile ))(*2 Same-Group ))(if>= Same-Group Same-Group Goal Same-
Group ))(+ (inv Last )(->90 Same-Group ))(inv (*2 Goal ))Nearest-Tile )Nearest-
Tile ))(- Last Same-Group ))(- Goal Goal )(ifdot Last Last Same-Group Same-Group
))) (if>= Goal Goal Same-Group (ifdot (/2 Last ))(->90 Last )Last (inv Goal ))(inv
Same-Group ))))))Nearest-Tile )))Same-Group Nearest-Tile ))))(->90 Last )))(*2 Goal
)(inv Same-Group ))(*2 Goal ))(ifdot (*2 (*2 (if>= (*2 (- (->90 Same-Group ))(-
Same-Group (if>= Last Nearest-Tile Nearest-Tile Same-Group )))Same-Group ))(/2
(ifdot Same-Group Goal Goal (- (+ Nearest-Tile (*2 Nearest-Tile ))(+ (->90 Goal
))(*2 Goal )))))(inv (+ Nearest-Tile Last ))(ifdot (inv Same-Group ))(inv Same-Group
)Nearest-Tile Last ))))Last (->90 Nearest-Tile )(ifdot (ifdot Last Last Nearest-Tile (+
Same-Group (inv (inv Same-Group ))))(->90 (/2 Nearest-Tile ))(- Goal (/2 (->90 (-
(if>= Same-Group (ifdot (- Last Same-Group ))Same-Group Same-Group (- Last Same-
Group ))Nearest-Tile Goal )Nearest-Tile ))))(/2 Goal ))Same-Group Nearest-Tile ))))
```

エージェント 1, 2, 4 は, 木 0 を参照

木 0:

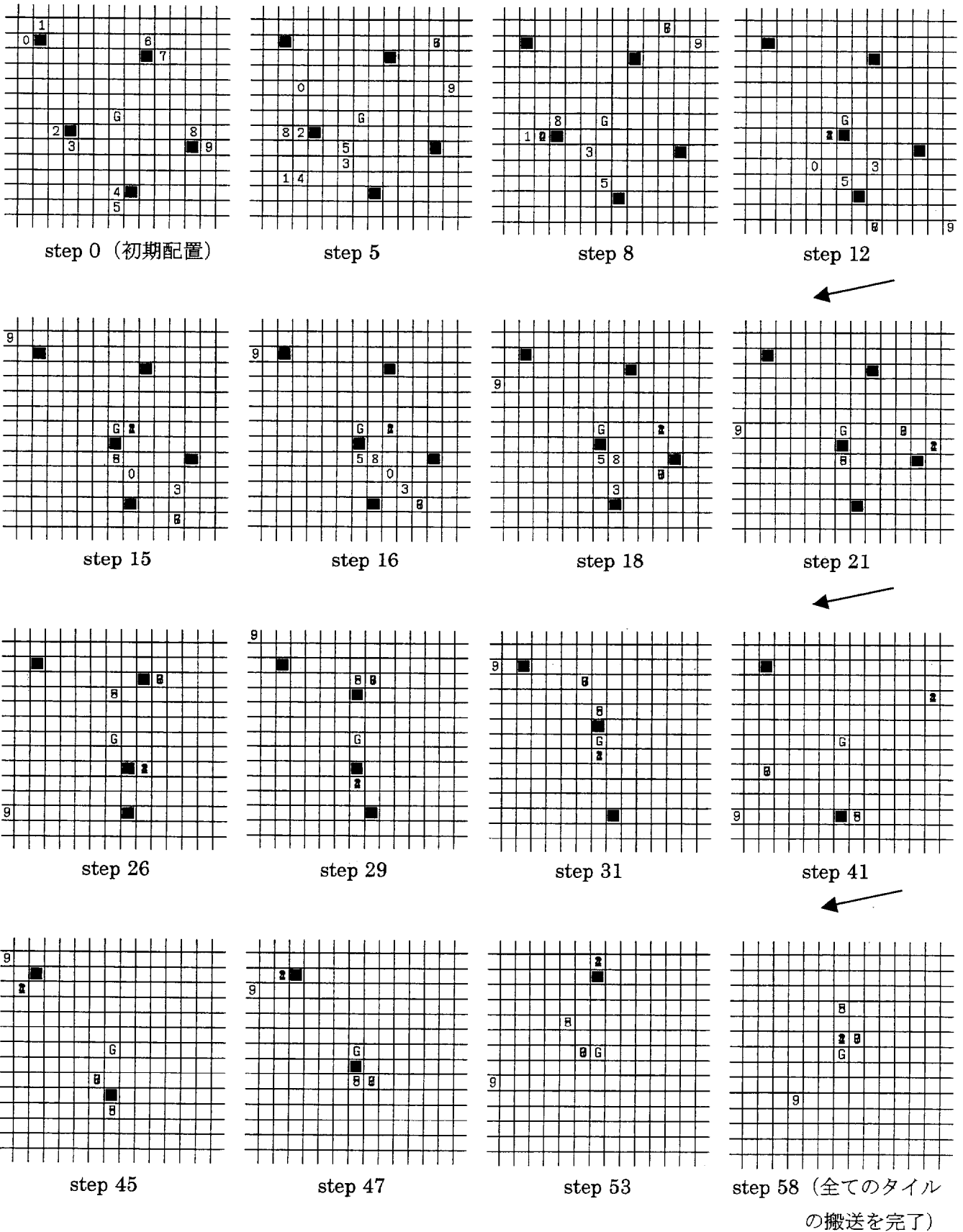
```
(+ (->90 (+ Same-Group Nearest-Tile ))(ifdot Nearest-Tile (- (if>= (->90 (if>=
(/2 Last ))(inv (if>= (/2 (+ (inv Goal ))(ifdot (if>= Nearest-Tile Same-Group Last
Last ))(->90 Goal ))(/2 Nearest-Tile ))(if>= Last Goal (if>= (inv (inv Nearest-Tile
)))(ifdot (->90 Same-Group ))(ifdot Goal Nearest-Tile Last Same-Group )Last Same-
Group ))(/2 (+ Same-Group Last ))(/2 Last ))Goal )))))(inv Goal )(+ Nearest-Tile (if>=
Last Nearest-Tile (- Nearest-Tile Goal )(*2 Last )))Same-Group ))(ifdot Last (- (inv Goal
)Goal ))(inv Nearest-Tile ))(->90 Goal ))(- Same-Group (inv Same-Group )))(if>= (+
(ifdot (ifdot Last Same-Group Same-Group (*2 Goal ))(if>= (->90 (if>= (- (+ Goal
Same-Group ))(/2 Same-Group ))Goal (if>= (inv (+ Goal (inv (inv (inv Same-Group
)))))(+ (ifdot Same-Group Goal Last Goal ))(if>= Goal Same-Group Goal Nearest-Tile
)))(+ Nearest-Tile (if>= (/2 (if>= Last (+ Last Same-Group )Same-Group (+ Goal
Goal )))Same-Group Last Nearest-Tile ))(+ Last Goal ))(ifdot Nearest-Tile Nearest-
Tile (*2 (if>= Last Same-Group Last Goal ))(->90 (->90 Nearest-Tile )))))(- (inv (/2
Last ))(*2 (- (inv Nearest-Tile ))(ifdot (->90 Last ))(+ (ifdot Same-Group Same-Group
(if>= Goal Nearest-Tile (+ Same-Group Goal ))(inv Nearest-Tile ))(+ Last Same-Group
))Goal ))(ifdot (- Same-Group Same-Group ))(/2 (+ Last (ifdot Goal (inv (->90 (- (- Goal
Same-Group ))(->90 Last )))))(inv Last )Last )))(if>= Goal Same-Group Last Nearest-
Tile )Same-Group )Same-Group ))))Goal Last ))(->90 (- (*2 (if>= (- (->90 Last ))(/2
(if>= Goal Last Goal Last )))Goal (+ Goal Nearest-Tile )Nearest-Tile ))Last ))Same-
Group )Last )Nearest-Tile Nearest-Tile (->90 Last ))Nearest-Tile Last )Goal )Last
(ifdot Last Last (+ (if>= (+ Last Last ))(+ (+ Same-Group Last )Goal )Nearest-Tile
Nearest-Tile ))(+ Nearest-Tile Same-Group ))(ifdot (- (/2 (inv (*2 Nearest-Tile )))(/2
(+ (->90 (+ Nearest-Tile Last ))(*2 (*2 Last ))))Last Same-Group (/2 (->90 Last
))))))
```

エージェント 3, 5, 8 は, 木 1 を参照

木 1:

```
(+ (->90 (+ Same-Group Nearest-Tile ))(ifdot Nearest-Tile (- (if>= Nearest-Tile
(if>= (+ (+ (ifdot Nearest-Tile (->90 (inv (+ (->90 Same-Group ))(inv (ifdot (if>=
(/2 Nearest-Tile )Goal (->90 (*2 (+ Same-Group Nearest-Tile )))(ifdot (inv Nearest-
Tile ))(if>= Goal (*2 Nearest-Tile ))(inv (/2 Goal ))(ifdot (->90 Same-Group ))(if>=
Last Goal Last Same-Group ))(->90 Last )Last ))(- (*2 (if>= Same-Group Last Nearest-
Tile Same-Group ))(->90 (+ Goal Last )))(ifdot (if>= (if>= Goal Same-Group Goal
Nearest-Tile ))(- Goal Nearest-Tile )Last Goal ))(- (+ Goal Goal )Nearest-Tile ))(- (/2
Same-Group ))(inv Last ))(inv (/2 Goal ))))Goal Last Last ))))Same-Group (->90 (-
Same-Group (*2 Last ))))(->90 (+ Last Last ))(/2 (+ Goal (/2 (*2 (*2 (+ (if>=
(- Last Goal ))(+ Last (- Last (inv (+ Goal Last )))))(if>= Nearest-Tile Same-Group
Last Last ))(ifdot Goal Last Same-Group Last ))Same-Group )))))(+ (/2 Last ))(+
Nearest-Tile Same-Group ))(inv Same-Group ))(*2 Last ))Nearest-Tile Last )Goal )Last
(ifdot (inv (+ (->90 (->90 Last ))(inv (ifdot Same-Group Nearest-Tile Last (inv Same-
Group ))))Last (+ (if>= (ifdot (ifdot (if>= Nearest-Tile Last Same-Group Same-Group
))(->90 (/2 (/2 (/2 (if>= Last Same-Group Last Last )))))(- (inv (- (if>= Nearest-
Tile Goal Last Last )Nearest-Tile ))Same-Group ))(inv Goal ))Last (inv (->90 Last
))Nearest-Tile )Same-Group Nearest-Tile Nearest-Tile ))(+ Nearest-Tile Same-Group
))Nearest-Tile ))))
```

付録B 進化により得られたマルチエージェントの行動例



■ : 3人でなければ運べないタイル.

G : ゴール (タイルの収集地点)

数字はエージェントの通し番号 (同一地点にいるものは数字が重なって表示されている.)

付録C 人工市場における株価暴落時の取引の様子

図 7.10 に示した株価暴落時におけるエージェント群の取引の様子を以下に示す。

Day 877 ; 前日 \$89.6932 --> 今日 \$90.4168 (変化率: 0.0081) ; 取引数 4
[取引内容]

売り手側			買い手側		
\$希望取引限度額 (予測変化率)			\$希望取引限度額 (予測変化率)		
A_11(Trend)	\$ 85.1141 (-0.0511)	<----->	A_ 2(Trend)	\$100.57 (0.1212)	
A_ 0(Trend)	\$ 87.6386 (-0.0229)	<----->	A_17(Trend)	\$ 91.6470 (0.0218)	
A_ 9(Modest)	\$ 89.8616 (0.0019)	<----->	A_15(Random)	\$ 91.4910 (0.0200)	
A_12(Modest)	\$ 89.8616 (0.0019)	<----->	A_13(Modest)	\$ 90.9719 (0.0143)	
A_ 3(Modest)	\$ 89.8616 (0.0019)	↑最後の取引により	A_18(Trend)	\$ 89.6704 (-0.0003)	
A_ 7(Modest)	\$ 89.8616 (0.0019)	その日の株価が決定	A_ 4(Trend)	\$ 88.1213 (-0.0175)	
A_ 8(Modest)	\$ 89.8616 (0.0019)		A_ 1(Trend)	\$ 86.0306 (-0.0408)	
A_16(Trend)	\$ 94.9201 (0.0583)		A_10(Trend)	\$ 85.6186 (-0.0454)	
A_14(Random)	\$ 97.2786 (0.0846)		A_ 5(Modest)	\$ 84.4376 (-0.0586)	
A_19(Random)	\$ 98.0214 (0.0929)		A_ 6(Modest)	\$ 83.6537 (-0.0673)	

※ Trend: 順張りグループ, Modest: 控え目グループ, Random: ランダム予測グループ

Day 878 ; 前日 \$90.4168 --> 今日 \$90.0521 (変化率: -0.0040) ; 取引数 7

A_19(Random)	\$ 86.0277 (-0.0485)	<----->	A_ 4(Trend)	\$ 99.8357 (0.1042)	
A_16(Trend)	\$ 86.3358 (-0.0451)	<----->	A_ 0(Trend)	\$ 96.5326 (0.0676)	
A_14(Random)	\$ 86.8722 (-0.0392)	<----->	A_10(Trend)	\$ 94.5793 (0.0460)	
A_ 2(Trend)	\$ 89.6032 (-0.0090)	<----->	A_ 9(Modest)	\$ 92.1458 (0.0191)	
A_13(Modest)	\$ 90.0521 (-0.0040)	<----->	A_ 5(Modest)	\$ 90.0521 (-0.0040)	
A_ 3(Modest)	\$ 90.0521 (-0.0040)	<----->	A_ 6(Modest)	\$ 90.0521 (-0.0040)	
A_ 8(Modest)	\$ 90.0521 (-0.0040)	<----->	A_12(Modest)	\$ 90.0521 (-0.0040)	
A_17(Trend)	\$ 94.6006 (0.0463)		A_ 1(Trend)	\$ 89.7890 (-0.0069)	
A_ 7(Modest)	\$ 95.0537 (0.0513)		A_18(Trend)	\$ 89.2201 (-0.0132)	
A_15(Random)	\$ 97.8924 (0.0827)		A_11(Trend)	\$ 89.1127 (-0.0144)	

Day 879 ; 前日 \$90.0521 --> 今日 \$90.2337 (変化率: 0.0020) ; 取引数 5

売り手側			買い手側		
A_15(Random)	\$ 88.2070 (-0.0205)	<----->	A_14(Random)	\$101.6512 (0.1288)	
A_10(Trend)	\$ 88.2790 (-0.0197)	<----->	A_16(Trend)	\$ 94.2823 (0.0470)	
A_17(Trend)	\$ 88.9180 (-0.0126)	<----->	A_ 2(Trend)	\$ 92.7891 (0.0304)	
A_ 9(Modest)	\$ 90.2337 (0.0020)	<----->	A_ 8(Modest)	\$ 90.2337 (0.0020)	
A_12(Modest)	\$ 90.2337 (0.0020)	<----->	A_ 3(Modest)	\$ 90.2337 (0.0020)	
A_ 5(Modest)	\$ 90.2337 (0.0020)		A_18(Trend)	\$ 90.2128 (0.0018)	
A_ 6(Modest)	\$ 90.2337 (0.0020)		A_ 1(Trend)	\$ 90.1403 (0.0010)	
A_ 7(Modest)	\$ 90.2337 (0.0020)		A_13(Modest)	\$ 89.4308 (-0.0069)	
A_ 4(Trend)	\$ 90.3455 (0.0033)		A_19(Random)	\$ 89.0127 (-0.0115)	
A_ 0(Trend)	\$ 92.9137 (0.0318)		A_11(Trend)	\$ 83.9478 (-0.0678)	

Day 880 ; 前日 \$90.2337 --> 今日 \$86.8311 (変化率: -0.0377) ; 取引数 3

売り手側			買い手側		
A_7(Modest)	\$ 79.6717 (-0.1171)	<----->	A_18(Trend)	\$ 93.9810 (0.0415)	
A_4(Trend)	\$ 81.5175 (-0.0966)	<----->	A_17(Trend)	\$ 92.9242 (0.0298)	
A_14(Random)	\$ 82.6714 (-0.0838)	<----->	A_13(Modest)	\$ 90.9908 (0.0084)	
A_16(Trend)	\$ 89.5531 (-0.0075)		A_15(Random)	\$ 89.3845 (-0.0094)	
A_8(Modest)	\$ 90.1427 (-0.0010)		A_19(Random)	\$ 89.2090 (-0.0114)	
A_6(Modest)	\$ 90.1427 (-0.0010)		A_1(Trend)	\$ 87.5248 (-0.0300)	
A_5(Modest)	\$ 90.1427 (-0.0010)		A_12(Modest)	\$ 87.3958 (-0.0315)	
A_2(Trend)	\$ 90.4032 (0.0019)		A_10(Trend)	\$ 85.7641 (-0.0495)	
A_3(Modest)	\$ 95.4147 (0.0574)		A_11(Trend)	\$ 85.6243 (-0.0511)	
A_0(Trend)	\$102.1046 (0.1316)		A_9(Modest)	\$ 85.3560 (-0.0541)	

Day 881 ; 前日 \$86.8311 --> 今日 \$83.5567 (変化率: -0.0377) ; 取引数 5

売り手側			買い手側		
A_17(Trend)	\$ 83.5567 (-0.0377)	<----->	A_19(Random)	\$ 89.8805 (0.0351)	
A_16(Trend)	\$ 83.5567 (-0.0377)	<----->	A_9(Modest)	\$ 86.8311 (0.0000)	
A_18(Trend)	\$ 83.5567 (-0.0377)	<----->	A_7(Random)	\$ 86.0124 (-0.0094)	
A_0(Trend)	\$ 83.5567 (-0.0377)	<----->	A_12(Modest)	\$ 86.0124 (-0.0094)	
A_2(Trend)	\$ 83.5567 (-0.0377)	<----->	A_1(Trend)	\$ 83.5567 (-0.0377)	
A_8(Modest)	\$ 86.0124 (-0.0094)		A_4(Trend)	\$ 83.5567 (-0.0377)	
A_13(Modest)	\$ 86.8311 (0.0000)		A_11(Trend)	\$ 83.5567 (-0.0377)	
A_3(Modest)	\$ 86.8311 (0.0000)		A_10(Trend)	\$ 83.5567 (-0.0377)	
A_5(Modest)	\$ 86.8311 (0.0000)		A_15(Random)	\$ 80.5363 (-0.0725)	
A_6(Modest)	\$ 86.8311 (0.0000)		A_14(Random)	\$ 77.5917 (-0.1064)	

Day 882 ; 前日 \$83.5567 --> 今日 \$80.4059 (変化率: -0.0377) ; 取引数 2

売り手側			買い手側		
A_19(Random)	\$ 78.7452 (-0.0576)	<----->	A_15(Random)	\$ 89.6877 (0.0734)	
A_1(Trend)	\$ 80.4059 (-0.0377)	<----->	A_10(Trend)	\$ 80.4059 (-0.0377)	
A_9(Modest)	\$ 82.7690 (-0.0094)		A_4(Trend)	\$ 80.4059 (-0.0377)	
A_6(Modest)	\$ 82.7690 (-0.0094)		A_0(Trend)	\$ 80.4059 (-0.0377)	
A_13(Modest)	\$ 82.7690 (-0.0094)		A_2(Trend)	\$ 80.4059 (-0.0377)	
A_12(Modest)	\$ 83.5567 (0.0000)		A_11(Trend)	\$ 80.4059 (-0.0377)	
A_5(Modest)	\$ 83.5567 (0.0000)		A_17(Trend)	\$ 80.4059 (-0.0377)	
A_3(Modest)	\$ 83.5567 (0.0000)		A_18(Trend)	\$ 80.4059 (-0.0377)	
A_8(Modest)	\$ 83.5567 (0.0000)		A_16(Trend)	\$ 80.4059 (-0.0377)	
A_7(Modest)	\$ 83.5567 (0.0000)		A_14(Random)	\$ 79.6440 (-0.0468)	

Day 883 ; 前日 \$80.4059 --> 今日 \$78.7353 (変化率: -0.0208) ; 取引数 1

売り手側			買い手側	
A_10(Trend)	\$ 77.3739 (-0.0377)	<----->	A_19(Random)	\$ 80.0967 (-0.0038)
A_ 8(Modest)	\$ 79.6479 (-0.0094)		A_14(Random)	\$ 79.4044 (-0.0125)
A_ 6(Modest)	\$ 79.6479 (-0.0094)		A_ 4(Trend)	\$ 77.3739 (-0.0377)
A_ 5(Modest)	\$ 79.6479 (-0.0094)		A_ 2(Trend)	\$ 77.3739 (-0.0377)
A_12(Modest)	\$ 80.4059 (0.0000)		A_ 0(Trend)	\$ 77.3739 (-0.0377)
A_13(Modest)	\$ 80.4059 (0.0000)		A_ 1(Trend)	\$ 77.3739 (-0.0377)
A_ 3(Modest)	\$ 80.4059 (0.0000)		A_17(Trend)	\$ 77.3739 (-0.0377)
A_ 7(Modest)	\$ 80.4059 (0.0000)		A_18(Trend)	\$ 77.3739 (-0.0377)
A_ 9(Modest)	\$ 80.4059 (0.0000)		A_11(Trend)	\$ 77.3739 (-0.0377)
A_15(Random)	\$ 81.9326 (0.0190)		A_16(Trend)	\$ 77.3739 (-0.0377)

Day 884 ; 前日 \$78.7353 --> 今日 \$77.7525 (変化率: -0.0125) ; 取引数 1

売り手側			買い手側	
A_15(Random)	\$ 75.1164 (-0.0460)	<----->	A_14(Random)	\$ 80.3886 (0.0210)
A_ 7(Modest)	\$ 78.3263 (-0.0052)		A_ 4(Trend)	\$ 77.0994 (-0.0208)
A_12(Modest)	\$ 78.3263 (-0.0052)		A_ 2(Trend)	\$ 77.0994 (-0.0208)
A_13(Modest)	\$ 78.3263 (-0.0052)		A_ 0(Trend)	\$ 77.0994 (-0.0208)
A_ 9(Modest)	\$ 78.7353 (0.0000)		A_ 1(Trend)	\$ 77.0994 (-0.0208)
A_ 5(Modest)	\$ 78.7353 (0.0000)		A_10(Trend)	\$ 77.0994 (-0.0208)
A_ 3(Modest)	\$ 78.7353 (0.0000)		A_17(Trend)	\$ 77.0994 (-0.0208)
A_ 8(Modest)	\$ 78.7353 (0.0000)		A_18(Trend)	\$ 77.0994 (-0.0208)
A_ 6(Modest)	\$ 78.7353 (0.0000)		A_11(Trend)	\$ 77.0994 (-0.0208)
A_19(Random)	\$ 82.3972 (0.0465)		A_16(Trend)	\$ 77.0994 (-0.0208)

Day 885 ; 前日 \$77.7525 --> 今日 \$78.6804 (変化率: 0.0119) ; 取引数 5

売り手側			買い手側	
A_ 8(Modest)	\$ 73.7951 (-0.0509)	<----->	A_ 4(Trend)	\$ 82.5334 (0.0615)
A_ 9(Modest)	\$ 76.6940 (-0.0136)	<----->	A_15(Random)	\$ 81.9615 (0.0541)
A_12(Modest)	\$ 77.5072 (-0.0032)	<----->	A_16(Trend)	\$ 80.6321 (0.0370)
A_ 5(Modest)	\$ 78.2378 (0.0062)	<----->	A_ 1(Trend)	\$ 80.4503 (0.0347)
A_ 3(Modest)	\$ 78.2378 (0.0062)	<----->	A_ 2(Trend)	\$ 79.1231 (0.0176)
A_ 6(Modest)	\$ 78.2378 (0.0062)		A_17(Trend)	\$ 77.6314 (-0.0016)
A_13(Modest)	\$ 79.5953 (0.0237)		A_11(Trend)	\$ 76.2360 (-0.0195)
A_14(Random)	\$ 80.0641 (0.0297)		A_18(Trend)	\$ 75.2503 (-0.0322)
A_ 7(Modest)	\$ 81.6826 (0.0505)		A_10(Trend)	\$ 72.1705 (-0.0718)
A_19(Random)	\$ 82.9579 (0.0669)		A_ 0(Trend)	\$ 71.8172 (-0.0763)

Day 886 ; 前日 \$78.6804 --> 今日 \$78.2109 (変化率: -0.0060) ; 取引数 6

売り手側			買い手側		
A_16(Trend)	\$ 72.6434 (-0.0767)	<----->	A_12(Modest)	\$ 82.2254 (0.0451)	
A_ 1(Trend)	\$ 74.8744 (-0.0484)	<----->	A_17(Trend)	\$ 80.8716 (0.0279)	
A_ 4(Trend)	\$ 75.2859 (-0.0431)	<----->	A_11(Trend)	\$ 80.8101 (0.0271)	
A_15(Random)	\$ 77.0473 (-0.0208)	<----->	A_ 9(Modest)	\$ 78.4130 (-0.0034)	
A_13(Modest)	\$ 78.2109 (-0.0060)	<----->	A_ 5(Modest)	\$ 78.2109 (-0.0060)	
A_ 7(Modest)	\$ 78.2109 (-0.0060)	<----->	A_ 3(Modest)	\$ 78.2109 (-0.0060)	
A_ 6(Modest)	\$ 79.4324 (0.0096)		A_ 8(Modest)	\$ 78.2109 (-0.0060)	
A_19(Random)	\$ 80.1326 (0.0185)		A_ 0(Trend)	\$ 75.2750 (-0.0433)	
A_ 2(Trend)	\$ 83.8541 (0.0658)		A_10(Trend)	\$ 74.7588 (-0.0499)	
A_14(Random)	\$ 86.8106 (0.1033)		A_18(Trend)	\$ 73.3143 (-0.0682)	

付録D 本研究に関する発表文献

論文

- 原 章, 長尾智晴: “自動グループ構成手法 ADG によるマルチエージェントの行動制御”, 情報処理学会論文誌, Vol.41, No.4, pp.1063 – 1072 (2000)
- 原 章, 長尾智晴: “遺伝的アルゴリズムを用いた画像中の任意方向を向いた顔の候補領域の抽出”, 日本印刷学会誌, Vol.36, No.1, pp.42 – 49 (1999)

投稿中論文

- 原 章, 長尾智晴: “自動グループ構成手法 ADG による人工株式市場の構築”, 情報処理学会論文誌 (条件付採録, 再投稿中)
- 山根裕明, 原 章, 長尾智晴: “株価変動の進化的最適化に基づく人工市場の構築”, 情報処理学会論文誌 (条件付採録, 再投稿中)
- 福寄雅洋, 原 章, 長尾智晴: “不完全知覚問題解決のための時系列依存分類システム (TCS) の提案”, 電気学会部門誌 (電子・情報・システム) (条件付採録, 再投稿中)

国際会議

- Akira Hara and Tomoharu Nagao: “Construction and analysis of artificial stock markets using ADG; Automatically Defined Groups”, Proc. of The 2001 International Conference on Computational Intelligence and Multimedia Applications (ICCIMA'01), pp.220-225 (2001)
- Akira Hara and Tomoharu Nagao: “Emergence of cooperative behavior using ADG; Automatically Defined Groups”, Proc. of The Genetic and Evolutionary Computation Conference (GECCO-99), pp.1039-1046 (1999)
- Akira Hara and Tomoharu Nagao: “ADG; Automatically Defined Groups for multi-agent cooperation”, Proc. of The Second Japan-Australia Joint Workshop on Intelligent and Evolutionary Systems, pp.91-98 (1998)

研究会

- 原 章, 長尾智晴: “エージェント群のグループ構造を利用した市場モデルの構築と解析”, 電子情報通信学会「知能と知識処理」研究会 (2002 年 1 月発表予定)
- 原 章, 長尾智晴: “自動グループ構成手法 ADG を用いた人工株式市場の構築”, 電子情報通信学会技術研究報告, Vol.100, No.529, AI2000-53, pp.73 – 78 (2000)
- 原 章, 長尾智晴: “マルチエージェントの協調のための最適なチーム構成の自動獲得” 電子情報通信学会技術研究報告, Vol.98, No.296, AI98-36, pp.23 – 30 (1998)
- 福寄雅洋, 原 章, 長尾智晴: “時系列情報を扱うことのできる分類システムの研究”, 電子情報通信学会技術研究報告, Vol.98, No.296, AI98-38, pp.39 – 46 (1998)
- 原 章, 長尾智晴: “G A を用いた画像中の任意方向の顔領域の抽出”, 電子情報通信学会技術研究報告, Vol.97, No.262, HCS97-12, pp.37-44 (1997)
- 原 章, 佐藤泰介: “遺伝的アルゴリズムを用いたスレッドの協調実験”, 電子情報通信学会技術研究報告, Vol.97, No.63, AI97-9, pp.63-70 (1997)

全国大会

- 原 章, 長尾智晴: “エージェント群のグループ構造を利用した株価変動メカニズムの解析”, 第 6 回進化経済学会大阪大会 (2002 年 3 月発表予定)
- 原 章, 長尾智晴: “人工株式市場におけるエージェント群の行動分析”, 電子情報通信学会総合大会 (2002 年 3 月発表予定)
- 原 章, 長尾智晴: “自動グループ構成手法 ADG を用いた株価変動モデルの構築”, 電子情報通信学会総合大会, D-8-3 (2001)
- 原 章, 長尾智晴: “グループ構造変更によるマルチエージェントの動的環境への適応”, 電子情報通信学会総合大会, D-8-13 (2000)
- 原 章, 長尾智晴: “自動グループ構成手法 (ADG) による集団行動の創発”, 電子情報通信学会総合大会, D-8-16 (1999)
- 原 章, 長尾智晴: “マルチエージェントの協調のための最適なチーム構成の自動獲得”, 電子情報通信学会・システムソサイエティ大会, D-8-8 (1998)

- 原 章, 長尾智晴: “静止画像からの任意方向の顔領域の抽出”, 電子情報通信学会・システムソサイエティ大会 D-8-13 (1997)

訪問記事

- 片岡寛明, 原 章, 平津大輔: “映像現場訪問記 (株) エム・アール・システム研究所”, 映像情報メディア学会誌, Vol.55, No.1, pp.86-88 (2001)
- 原 章, 石田伯仁, 金田道明, 今井繁: “ソニーメディアワールド訪問”, 映像情報メディア学会誌, Vol.52, No.6, pp.821-824 (1998)

新聞記事

- “値動きリアルな仮想市場”, 日本経済新聞 2002年1月18日 (金) 朝刊 (2002)
- “複数のロボット自動で仕事分担 東工大が制御プログラム技術”, 日経産業新聞 2000年6月28日 (水) 朝刊 (2000)

その他

- 片岡寛明, 原 章, 長尾智晴: “エージェントの行動制御用プログラムの進化的生成について”, 文部省科学研究費 特定領域研究「ソフトウェア発展」A04 班研究集会資料 (1999)
- 原 章, 長尾智晴: “マルチエージェントにおける役割分担の創発について”, 文部省科学研究費 特定領域研究「ソフトウェア発展」A04 班研究集会資料 (1998)