

自動グループ構成手法を用いた 遺伝的ネットワークプログラミングによる マルチエージェントの役割分担獲得

中村 貴志[†] 村田 忠彦^{††,†††}

本論文では、自動グループ構成手法 (Automatically Defined Groups : ADG) を適用した遺伝的ネットワークプログラミング (Genetic Network Programming : GNP) を提案し、マルチエージェントの協調行動学習に適用する。GNP では、ネットワーク構造によりエージェントの行動規則を表現する。従来の GNP では、単一のネットワークですべてのエージェントを制御する Homogeneous モデルの構造が用いられていた。しかし、Homogeneous モデルでは、役割分担などの複雑な協調行動の獲得は難しい。複数のエージェントを制御するには、個々のエージェントに異なった行動規則を与える Heterogeneous モデルも考えられるが、複雑な行動規則を同時に学習するのは困難である。そこで、エージェント群をグループ化し制御を行う ADG 手法が提案されている。ADG 手法では、グループ単位で行動規則を持たせるため、最適化する行動規則の数を抑えることができ、探索効率を向上させている。本論文では、ADG のグループ化の枠組みを GNP に適用した手法を提案する。能力が異なるエージェントに対して異なるタスクを与える荷物搬送問題と、能力が均一なエージェントに対して異なるタスクを与える荷物搬送問題を用いた計算機実験から、提案手法によって複数のエージェントが役割分担し協調行動が生成されることを示す。

Genetic Network Programming with Automatically Defined Groups for Role Assignment Among Multiple-agents

TAKASHI NAKAMURA[†] and TADAHIKO MURATA^{††,†††}

In this paper, we propose a Genetic Network Programming (GNP) Architecture using Automatically Defined Groups (ADG) to a multi-agent problem where the cooperation of agents is required. GNP is one of evolutionary methods inspired from Genetic Programming (GP). While GP has a tree architecture, GNP has a network architecture with which an agent works in the virtual world. In GNP with ADG, each agent is assigned to a group according to its role to complete some task of a cooperative problem. We consider two types of problems in this paper: one problem is to assign an appropriate role to each agent according to its ability, and the other is to assign a proper role to each agent with the same ability. While the first problem has the specific conditions as for the ability of an agent, the latter has ones. We show the effectiveness of the proposed algorithm through computer simulations on the two problems.

1. はじめに

単純な規則の組合せにより、様々なタスクを行うことができるマルチエージェントシミュレーション (MAS) は、多様な問題の解決策として期待されている。MAS

の対象とする問題に応じて、即時的な条件判断を行う手法や、行動の流れを表現する手法が研究されており、複数のエージェントを制御するための手法として、遺伝的プログラミング (Genetic Programming : GP) が提案されている¹⁾。GP では、木構造により即時的な条件判断や行動が表現される。下位部分木を順次実行する ProgN ノードを用いれば、行動の流れを表現することも可能であるが、端末まで到達するとルートノードに戻るため、完全に連続的な行動を表現してはいない。

これに対し、連続的な行動を表現するために、遺

[†] 関西大学大学院総合情報学研究科
School of Informatics, Kansai University Graduate School

^{††} 関西大学総合情報学部
Faculty of Informatics, Kansai University

^{†††} 関西大学政策グリッドコンピューティング実験センター
Policy Grid Computing Laboratory, Kansai University

伝的ネットワークプログラミング (Genetic Network Programming : GNP) が提案されている²⁾⁻⁸⁾。GNP では、簡単な機能を有したノードをネットワーク状に接続することで行動規則を表現する。これにより、条件判断や行動の流れを表現することができ、過去の行動や判断を暗黙的に利用することができる。そのため、センサの限界により異なる状態を同じ状態であると認識してしまう部分知覚問題に対しても、状態に応じた最小限の条件判断で行動を決定することができる。さらに、ループ構造を表現できることや、ノードの再利用性が高いことから、部分的なネットワークで局所的な機能を表現することも可能となる。また、GP を拡張した適応型 GP-オートマトンとの比較⁹⁾において、GP では実際には使用されないノードを多く含む決定木を生成するが、GNP のネットワーク構造ではノード数が少ない構造を生成できることが示されている。つまり、GNP ではコンパクトな行動規則が表現可能であり、可読性が高いといえる。

エージェントの制御を行う他の手法としては、クラシファイアシステム¹⁰⁾がある。クラシファイアシステムでは、状態を条件部に行動を結論部に表現した If-Then ルールの集合を修正していくことで、エージェントの一連の行動を獲得する。しかし、各ステップですべての知覚を判断に用いることや、獲得した規則の可読性が低いという問題点がある。ほかに、GP を拡張して部分知覚に対応することを目的とした遺伝的オートマトン GAUGE¹¹⁾が提案されている。GAUGE は、GP をオートマトンの構造に拡張し、内部状態を定義することで、部分知覚による騙し問題に対応している手法である。しかし、状態推移の分岐方法や、ルートノードと終端ノードを反復するという部分で GNP とは異なっている。

エージェント集団を 1 つの個体によって制御する際には、主に Homogeneous モデルと Heterogeneous モデルの 2 つの方法が考えられる。複数のエージェントを単一の行動規則により制御する Homogeneous モデルは、単純な構造という長所があるが複雑な協調行動の表現が困難である。前述の GNP では、このモデルが採用されている。一方、各エージェントを異なる行動規則により制御する Heterogeneous モデル¹²⁾⁻¹⁴⁾は、複雑な行動表現が可能になるが、すべてのエージェントの行動規則の最適化に時間がかかるという欠点がある。なお、前述の文献 6) の手法は、ヘテロマルチエージェントシステムと名付けられているが、この手法では環境の変化に対応するために複数のネットワークを使用しているので、本論文における Heterogeneous モ

デルとは異なっている。

これらのモデルの欠点を克服するため、エージェントをいくつかのグループに分割し、グループ単位の役割分担を GP により表現する自動グループ構成手法 (Automatically Defined Groups : ADG) が提案されている¹⁵⁾。ADG では、エージェント群をグループ化することによって、最適化する行動規則の数を抑えている。また、行動規則とともに、グループ構造も遺伝的操作の中で獲得していく。

本論文では、ADG のグループ化の枠組みを GNP に適用した手法として、GNP with ADG を提案し、エージェントの役割分担を試みる。計算機実験により、GNP が GP よりエージェント行動規則の記述力が優れていることを示す。さらに、エージェントの能力差の有無による役割分担の生成に関する実験を行い GNP の有用性を示す。

以下に本論文の構成を示す。2 章で、従来手法である GNP と ADG について述べる。3 章において、提案手法である GNP with ADG を説明する。4 章で、荷物搬送問題による実験結果と考察を示し、5 章でまとめと今後の方針について述べる。

2. 従来手法

2.1 遺伝的ネットワークプログラミング (GNP)

2.1.1 基本構造

GNP⁴⁾ は、If-Then ルールを木構造によって表現する GP に着想を得て、ネットワーク構造によってルールを表現する手法である。GP と同様の最小単位の機能を持ったノードを有向リンクによってネットワーク状に接続することで行動規則を表現する。その構造を図 1 に示す。GNP では、このようにネットワークで表現された行動規則を 1 つの個体として取り扱う。

ネットワークを構成するノードには、スタートノード、判定ノード、処理ノードの 3 種類が存在する。スタートノードは、プログラムの開始位置を表す。判定

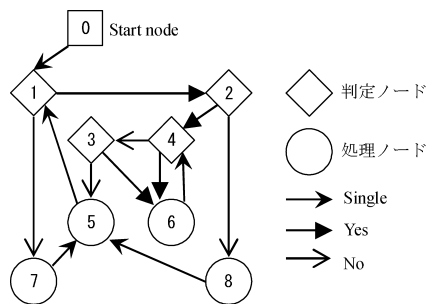


図 1 GNP におけるネットワーク構造の例
Fig.1 Example of network structure in GNP.

node 0	0	0	1				
node 1	1	0	7	2			
node 2	1	0	8	4			
node 3	1	1	5	6			
node 4	1	1	3	6			
node 5	2	0	1				
node 6	2	0	4				
node 7	2	1	5				
node 8	2	1	5				

node i	ノード部			リンク部			
	NT_i	FID_i	C_{i1}	C_{i2}	—	C_{ij}	—

図 2 GNP における遺伝子型の構造
Fig. 2 Structure of chromosome.

ノードは定められた条件判定を行い、その判定結果に従って次の遷移先を分岐する。そのため、分岐の数に応じたリンクを持つ。処理ノードは、定められた処理を行い、所持している 1 つのリンクに従い、次のノードへ実行を遷移させる。この判定ノードと処理ノードは、GP における非終端ノードと終端ノードにそれぞれ対応している。

次に、図 1 の個体を表現する遺伝子型を図 2 に示す。ノード遺伝子は、プログラム内で識別するためのノード識別番号が与えられ、内容はノード部とリンク部からなる。ノード部には、ノードの種類 (NT_i)、機能の番号 (FID_i) があり、リンク部には接続先 (C_{ij}) がリンクの数だけ存在する。 NT_i には、種類に応じた番号を指定し、スタートノード、判定ノード、処理ノードをそれぞれ $NT_i = 0, 1, 2$ とする。 FID_i には、ノード関数を決定する番号が指定される。つまり、判断ノード ($NT_i = 1$) であり判断ノード関数が J 種類あるときは、 FID_i は、0 から $J - 1$ の値をとる。処理ノード ($NT_i = 2$) であり処理ノード関数が P 種類ある場合は、 FID_i は 0 から $P - 1$ の値をとる。 C_{ij} には、リンクの接続先のノード識別番号が入る。たとえば、図 2 の node1 は 0 番目の判定ノード関数を行うことを表し、node7 と 2 に接続する。

2.1.2 制御の実行

GNP での制御は、ネットワーク内のノードの活性化を遷移していくことで処理される。ここで、ノードの機能を実行することをノードが活性化するという。制御は、スタートノードから開始し、ノードが持つリンクに従って制御を遷移していく。このとき、各ノードに制御が遷移したら、ノードを活性化させ、判定や処理を行う。

図 1 では、スタートノードである node 0 から実行が開始され、node 1 へ遷移する。次に、node 1 の活性化処理として判定ノード関数の 0 番目の関数 ($NT_1 = 1$,

$FID_1 = 0$) を実行する。たとえばここでは判断が “No” だったとすると、次に node 7 へ制御が遷移することになる。そして、node 7 で処理ノード関数の 1 番目の関数 ($NT_7 = 2$, $FID_7 = 1$) が実行されてエージェントが行動を行い、1 ステップが終了する。次のステップからは、前回の node 7 が接続している先の node 5 から同様の操作が行われる。実行時のシミュレーションの 1 ステップはノードの活性化回数に従って終了することとした。本論文の実験においては、1 つの処理ノードを活性化するか、5 つの判定ノードを活性化することで 1 ステップを終了する。GNP では、次のステップで前回のステップに続くノードから開始することにより、過去の行動や判断をふまえた処理が行われる。

2.1.3 遺伝的操作

ネットワークの構造は、交叉と突然変異の遺伝的操作によって変更される。本論文では、各ノードはリンクを自分自身へ接続しないこととし、ノード部は変更せずリンクの接続先のみを変えることとした。そのため、集団中のすべての個体において、ノード識別番号が等しいノードのノード部を同一の内容とした。

● 交叉

交叉操作では、一様交叉によってノードを入れ替える。まず、個体群から親個体となる 2 個体を選び、交叉率に従って交叉する。交叉を行う場合、親個体中の各ノードをそれぞれ等確率で決定する。そして、選択されたノードを 2 個体間で交換する。なお、このときノードの内容は全個体で同一にしているため、リンクのみが変更されることになる。

● 突然変異

突然変異操作では、まず、1 つの個体を選び、個体内に存在するすべてのリンクに対して、突然変異率に従ってリンク先をランダムに変更する。

2.2 自動グループ構成手法 (ADG)

ADG¹⁵⁾ は、複数のエージェントに役割分担をさせることを目的とした手法である。つまり、Homogeneous モデル^{2)~8)} で困難であった役割分担の学習と、Heterogeneous モデルで問題となっていた環境内のエージェント数の増加による探索効率の悪化に対応している。ADG では、1 個体内にいくつかの GP 木を持ち、その中の同じ木を参照するエージェントの集合をグループと定義している。そして、エージェント群のグループ構成と、グループごとの役割に対応する木構造をまとめたものを、遺伝的操作における 1 個体としている。つまり、行動規則とともに、グループの構造も遺伝的操作によって獲得される。

3. GNP with ADG

原らは、ADGの枠組みを用いてGPによる役割分担を表現している¹⁵⁾。本論文では、この枠組みを用いて、GNPによる役割分担の獲得を目指す。行動規則としてGPの木構造の部分に代わって、GNPのネットワークを用いる。遺伝的操作の1個体は、グループの数に応じたネットワークと、エージェントのグループ構成からなる(図3)。すなわち図3では、1~6のエージェントが{1, 3, 4, 6}と{2, 5}の2つのグループに分かれ、グループごとに異なるネットワークを参照している個体を表している。

3.1 ADGモデルにおける遺伝的操作

3.1.1 グループ突然変異

グループ突然変異は、ADGにより提案された操作である。この操作は、グループ構造を突然変異的に変更するために行われる。また、交叉操作によってもグループ構造が変更されるが、個体間のグループ構造が収束してしまった場合に、交叉操作では構造が変化しなくなってしまう。それに対処するためにも、グループ突然変異操作を行う。処理の流れを以下に示す。

- (1) 図3のようなGNP with ADGの1個体に対して、グループ突然変異率によって、操作を行うかどうかを決定する。
- (2) 操作を行うのであれば、次へ進む。そうでなければ、この個体に対するグループ突然変異操作を終了する。
- (3) 1つのエージェントを任意に選択する。
- (4) 任意に選んだグループに、(3)で選択したエージェントを移動させる。
- (5) 選んだグループが、元のグループと同じであれば、そのエージェントだけが所属するグループを新たに作り、そのグループの行動規則を、元のグループと同じものにする。

3.1.2 交叉

次に、グループ突然変異の操作を行った個体に、交

叉の処理を行う。交叉操作は2段階の操作を行う。まず、グループ構成の変更を行い、両個体のグループが所有するネットワークに対してGNPの交叉操作を行う。処理の流れは以下のとおりである。

- (1) 交叉対象となる2個体を選択する。
- (2) 1体のエージェントを任意に選択する。
- (3) 2個体内で、(2)で選んだエージェントの所属するグループの構成を比較する。
- (4) 比較したグループの構造に応じて、両グループの構成が等しくなるように、次のいずれかの操作を行う。
 - (a) 2つのグループ構成が同等の場合、それぞれのグループが所有するネットワークに対して(5)を行う。
 - (b) 片方のグループにもう一方のグループが包含されている場合、包含しているグループに対して分割を行う。
 - (i) 包含しているグループを、包含されているグループと同じエージェント群と、それ以外のエージェント群とで2つのグループに分割する。
 - (ii) 分割したそれぞれのグループには、元のグループと同じネットワークを持たせる。
 - (iii) グループ構成が等しくなったグループどうしのネットワークに対して(5)を行う。
 - (c) 選択されたグループの構成が同等でも包含関係にもない場合、選択された2つのグループの和集合のエージェント群が1つのグループとなるように統合を行う。
 - (i) それぞれの個体において、和集合のエージェント構成となるように選択されたグループへ他のグループからエージェントを移動させる。
 - (ii) エージェントの所属変更により、エージェントがいなくなったグループは消滅する。
 - (iii) グループ構成が等しくなったグループどうしのネットワークに対して(5)を行う。
- (5) グループ構成が等しくなったグループが参照するネットワークに対してGNPの交叉操作を行う(2.1.3項参照)。

たとえば、(b)の場合は以下のようなグループ分割を行う。いま、エージェント1を基準に親個体Aと親個体Bの中からそれぞれ $GroupA = \{0, 1, 2, 3\}$, $GroupB = \{1, 2\}$ のグループが選ばれたとする。このとき、 $GroupA \cap GroupB$ となっているため、(b)の分割操作を行うこととなる。まず、包含しているグループである $GroupA$ を、 $GroupA' = \{1, 2\}$, $GroupA'' = \{0, 3\}$ の2つのグループに分割する。

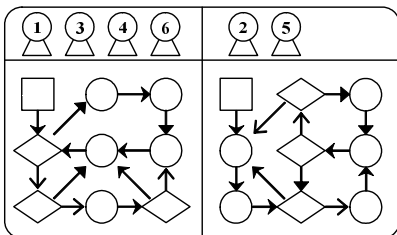


図3 GNP with ADGの1個体の例

Fig. 3 Example of an individual in GNP with ADG.

このとき、これらのグループのネットワークは、元々の $GroupA$ と等しいネットワークを与える。そして、分割された $GroupA'$ と $GroupB$ が参照するネットワークを交叉する。

一方、(c) の場合では以下のようなグループ統合を行う。同様に、エージェント 1 を基準として親個体 A と親個体 B から $GroupA = \{0, 1, 3\}$, $GroupB = \{1, 2\}$ が交叉対象となったとする。 $GroupA \neq GroupB$ かつ $GroupA \not\supseteq GroupB$ かつ $GroupA \not\subseteq GroupB$ であるため、(c) の統合操作を行う。まず、 $GroupA$ と $GroupB$ の構成を、 $GroupA \cup GroupB = \{0, 1, 2, 3\}$ となるように各個体内でエージェントの移動を行う。つまり、親個体 A のグループ群が $GroupA = \{0, 1, 3\}$, $Group\bar{A} = \{2, 4\}$ であったなら、このグループ構成を $GroupA' = \{0, 1, 2, 3\}$, $Group\bar{A}' = \{4\}$ と変更する。同様に、親個体 B 内で $GroupB$ を $GroupB' = \{0, 1, 2, 3\}$ に変更する。そして、 $GroupA'$ と $GroupB'$ が参照するネットワークを交叉する。

3.1.3 突然変異

個体内のすべてのネットワークに対して、GNP の突然変異操作を行う。つまり、個体内のすべてのネットワークに含まれるすべてのリンクに対して、以下の操作を行う。

- (1) 突然変異率に従って、操作の可否を決定する。
- (2) リンクの接続先をランダムに決定する。

3.2 GNP with ADG のアルゴリズム

本手法では 3.1 節の ADG モデルの遺伝的操作を用いる。全体のアルゴリズムは、以下のとおりである。

(1) 初期個体群生成

個体群サイズ n の個体集団を生成する。初期個体は、ランダムなグループ構成を生成するとともに、各グループが保持するネットワークをランダムに生成する。このとき、すべてのネットワーク内で、ノード構成は統一し、リンクの接続のみをランダムに決定する。

(2) 評価

すべての個体を問題に適用し適応度を算出する。ADG では、適応度が等しい個体はグループ数が少ないほうを優越させる。これにより、グループ数が少なくなるように進化をさせている。

(3) 選択

全個体中の最良の 1 個体をエリートとして保存する。次に、すべてのエージェントから、トーナメント選択により、遺伝的操作を行う個体群を集団 P として $(n-1)$ 個体を選択する。

(4) 遺伝的操作

遺伝的操作は、(3) での集団 P 内のすべての個体に対して、グループ突然変異、交叉、突然変異の操作を順次施す。

(5) 新個体群の生成

保存したエリートと、(4) の操作を行った後の集団 P をあわせたものを、新個体群とする。

(6) 評価

(2) と同様に評価を行う。

(7) 終了条件判定

あらかじめ指定した終了世代まで到達したのならば、実験を終了する。そうでなければ、(3) に戻る。

4. 実験と結果

本論文では、原らが用いた荷物搬送問題に提案手法 (GNP with ADG) を適用して検証を行う。この問題では、エージェントの行動規則は If-Then 形式で記述することが可能である。GNP では、このような If-Then 形式の問題に対して適用することができ、本手法でも同様の形式の問題に適用することが可能である。

最初に、エージェントによって能力差の存在する問題を用いて、GP と GNP の違いを示す。次に、同様の問題において、Homogeneous モデル、Heterogeneous モデル、提案手法の性能比較を行い、提案手法での役割分担の生成を検証する。さらに、エージェントに能力差のない場合においても、適切なグループ分けが行えるのかどうかを試みる。

4.1 実験 1: 能力差による役割分担

4.1.1 荷物搬送問題の問題設定

● 概要

エージェントは、2 カ所の荷物の置き場からゴールへ、荷物を運搬する。2 カ所の荷物置き場には、それぞれ重い荷物と軽い荷物の 2 種類の荷物が配置される。また、エージェントには、重い荷物を持てるかどうかの能力差をつける。

● 環境設定

11×10 のグリッドワールドで、ゴールの位置は (5, 0) とした。2 カ所の荷物置き場は、それぞれ、ゴールからの距離は等しくなるように、重い荷物の置き場は (0, 9)、軽い荷物の置き場は (10, 9) としている。どちらの荷物も置き場に無限に存在し、実験の間はなくなる。エージェントは、グリッド上で、上下左右に 1 マスずつ移動し、複数のエージェントが同一のマスに存在することができる。

表 1 実験 1 で用いるノード関数
Table 1 Node functions in Experiment I.

ノード関数名	リンク数	意味
If_Carry_Load	2	荷物を運んでいるか
If_Load_Here	2	ここに荷物があるか
Move_Goal	1	現在荷物置場に居るか
Move_Heavy_Load	1	ゴールへ向かって移動
		重い荷物置き場へ
		向かって移動
Move_Light_Load	1	軽い荷物置き場へ
		向かって移動
Pick_Up	1	足元に荷物があり, その荷物を持つことが 可能なら荷物を拾う

● 目的

エージェントは、最初にゴール地点で荷物を持たない状態から行動を開始し、指定された時間 (*Limit_Time*) 内に、より多くの荷物をゴールに運び込むことを目的としている。適応度は、ゴールに運び込んだ重い荷物の得点 (*HL_Score*) と軽い荷物の得点 (*LL_Score*) の総得点となる。ゴールに運び込んだ重い荷物の数を *HL_Count*、軽い荷物の数を *LL_Count* とした場合、適応度は以下の式により計算される。

$$Fitness = HL_Count \times HL_Score + LL_Count \times LL_Score$$

● 設定

環境内には、2種類の能力差のあるエージェントを 20 体配置する。エージェントを生成し、5 体の Agent_H は重い荷物でも軽い荷物でも持つことができ、残りの 15 体の Agent_L は軽い荷物しか持つことができないこととした。エージェントは、一度に 1 つの荷物を持つことができ、ゴールにたどり着くことで、自動的に荷物を置くことができる。

GP や GNP で用いるノード関数を表 1 に、実験で用いたパラメータを表 2 に示す。本論文では、それぞれの手法による個体表現の違いを比較するために、個体表現に用いるノード数を等しく設定した。つまり、GP の木は、最大の深さを 5 とし、一方の GNP のネットワークは、スタートノード 1 つと各ノード関数を 5 つずつとした。これにより、どちらの手法も利用可能なノード数を最大で 31 個とした。この問題では、ゴールと荷物置き場との間の往復に 30 ステップかかり、制限時間の 100 ステップ以内には、3 往復することが可能である。したがって、最高適応度は以下の計算により 120 になる。

$$MaxFitness = (HL_Score \times 5 + LL_Score \times 15) \times 3$$

表 2 実験パラメータ
Table 2 Parameter settings.

個体数	200
トーナメントサイズ	5
エリート保存個体数	1
交叉率	0.9
突然変異率	0.01
グループ突然変異率	0.9
終了ステップ数	100
重い荷物の点数	5
軽い荷物の点数	1

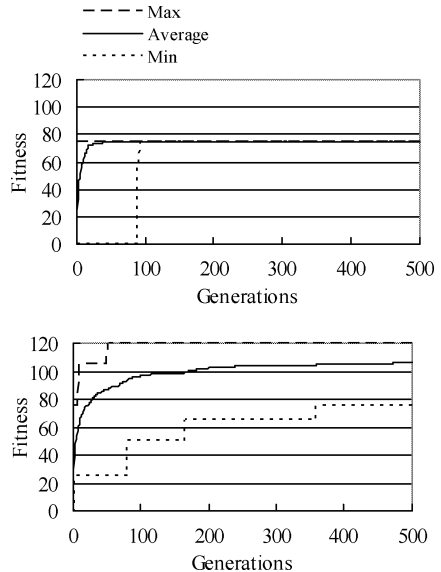


図 4 Homogeneous における GP (上) と GNP (下) の 100 回の実験での適応度の変化

Fig. 4 The fitness of GP (top) and GNP (bottom) with homogeneous in 100 trials.

4.1.2 実験結果と考察

まず、GNP と GP との比較を行った。エージェントのモデルは、Homogeneous モデルとした。GP におけるノードは、GNP と同様に表 1 に従い、パラメータも表 2 と同様にした。100 回の実験での、GNP と GP での各世代の最高適応度の平均値の推移を図 4 に示す。また、それぞれの手法で獲得した最良個体の例を、図 5 に示す。この図では、不要なノードとそのリンクを削除してある。図 4 より、GP の最高適応度が 75 であったことが分かる。この実験では、Homogeneous モデルであるため、Agent_H も Agent_L も同一の行動規則に従う。そのため、エージェントの能力に関係なく、重い荷物を拾いに行くことになり、重い荷物の分の得点しか得ることができない。これに対して GNP では、最高適応度の 120 を獲得することができた。図 5 で GNP のネットワークでは、まずすべて

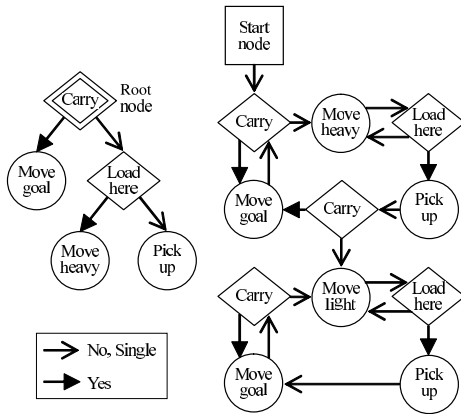


図 5 GP (左) と GNP (右) での最良個体の例
 Fig.5 Acquired programs in GP (left) and GNP (right).

のエージェントが重い荷物置き場へ向かって荷物を拾う行動を行う。このとき、軽い荷物のみを持つことができる Agent_L は、重い荷物を拾うことができないので、続く If_Carry_Load の判定により、次からは軽い荷物を運ぶように行動を切り替えている。この事例では、Pick_Up という行動の結果を判断することで、エージェント自身の能力を判別し、後の行動の選択を行っている。これに対し GP の木構造では、その場における条件判断を行うことしかできず、GNP のように行動を切り替えることはできない。また、このネットワークは、ゴールと重い荷物置き場の間を往復する行動と、ゴールと軽い荷物置き場の間を往復する行動を別々に表現している。これにより、エージェントを制御する方法を容易に読み解くことができる。

次に、従来手法 ADG (GP with ADG) と提案手法である GNP with ADG, Homogeneous モデルの GNP, Heterogeneous モデルの GNP の 4 つのモデルの比較を行った。それぞれの手法での各世代の最高適応度の変化を図 6 に、提案手法におけるグループ数の変化を図 7 に示す。それぞれ 100 回の試行の平均である。Homogeneous モデルでは、100 回の試行中で 1 試行のみ最高適応度を獲得したが、Heterogeneous モデルでは 1 試行も得ることはできなかった。それに対し提案手法では 450 世代で、従来の ADG 手法では 100 世代の時点で 100 試行すべてにおいて最高適応度に到達した。Homogeneous モデルと Heterogeneous モデルのどちらの手法でも理論的には最高適応度を獲得できるが、探索に時間がかかると考えられる。一方、ADG モデルの 2 手法では、少ない世代数でエージェントの能力に応じた 2 グループを生成し、グループごとに荷物置き場へ向かう役割分担が獲得できた。

図 6 から従来手法である GP with ADG 手法の方

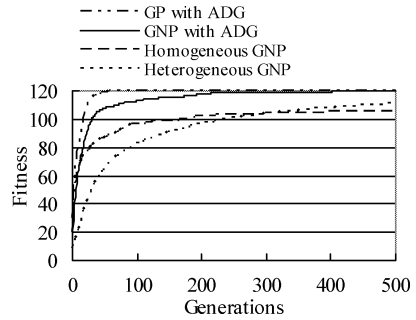


図 6 GP with ADG, GNP with ADG, Homogeneous GNP, Heterogeneous GNP による 100 回の実験での適応度の変化
 Fig.6 Comparison of the average fitness in 100 trials among four methods.

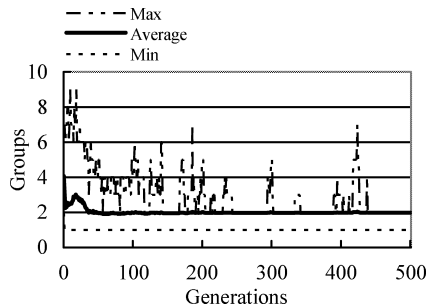


図 7 提案手法でのグループ数の変化
 Fig.7 The change of the number of groups among proposed method.

が、提案手法の GNP with ADG と比べて、最適解を早く探索できていることが分かる。これは、GNP では生成できる解のバリエーションが多いため、局所解が多く存在するためであると考えられる。獲得された個体の構造を比較するため、両手法により最終世代で獲得された最適個体に含まれるノード数を表 3 に示す。GNP での値は、スタートノードから到達可能なノード数を示している。また、この実験での ADG モデルは最終的に 2 グループに収束しているので、最大ノード数が 62 となる。この表より、獲得された個体に含まれるノード数は GNP with ADG の方が少ないことが分かる。これは、GNP では不要なノードの増加をおさえて少ないノードによる行動規則が生成できることを示している。

4.2 実験 2：能力を同等にしたときの役割分担

次に、エージェントの能力を同等にした場合の実験を行う。すなわち、明確にグループを分ける能力差などの要因がない状態で、問題に適したグループ構成を獲得することができるかを検証する。

4.2.1 問題設定

荷物とエージェントが各々 1 種類だけとなるように

表 3 GNP with ADG と GP with ADG の最良個体に含まれるノード数

Table 3 The number of nodes in GNP with ADG and GP with ADG.

	Max.	Ave.	Min.	St. dev.
GNP with ADG	40	24.7	13	6.1
GP with ADG	62	47.8	36	5.8

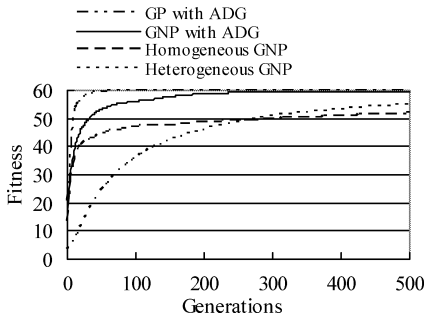


図 8 実験 2 での各モデルによる適応度の変化

Fig. 8 Comparison of the average fitness among four methods for Experiment II.

した以外は，実験 1 と同様の設定で実験を行った．2カ所の荷物置き場を荷物置き場 1，荷物置き場 2 とし，環境上に同種類の荷物のみが存在することとした．これに応じて，ノード関数の荷物置き場へ向かう 2つのノード (Move_Heavy_Load , Move_Light_Load) を，それぞれの荷物置き場へ向かう Move_First_Load , Move_Second_Load とした．また，エージェントは，20 体すべてが荷物を持つことができるものとした．そして，荷物置き場に配置する荷物を，それぞれ 30 個ずつとして，合計で 60 個の荷物を環境上に配置した．決められた時間内には，3 往復しかできないため，多く荷物を運ぶには，2カ所の置き場を分担して回る必要がある．これにより，能力の差ではなく，問題の設定に応じた役割分担の生成が必要となる．

4.2.2 結果と考察

100 回の試行での結果を，図 8 に示す．Homogeneous モデルと Heterogeneous モデルの 2つのモデルは，実験 1 と同様に，500 世代の間に平均値で最高適応度を獲得することはできなかった．一方，提案手法は 100 回の中で 98 回，最高適応度を獲得していた．最高適応度を獲得した個体 (図 9 参照) では，エージェントが 10 体ずつの 2つのグループが形成され，それぞれが別々の荷物置き場を往復する行動を行っていた．これは，提案手法によって問題の構造に応じた役割分担を形成できたことを示している．また，GNP with ADG と GP with ADG における個体に含まれるノード数を表 4 に示した．図 8 から GP with ADG

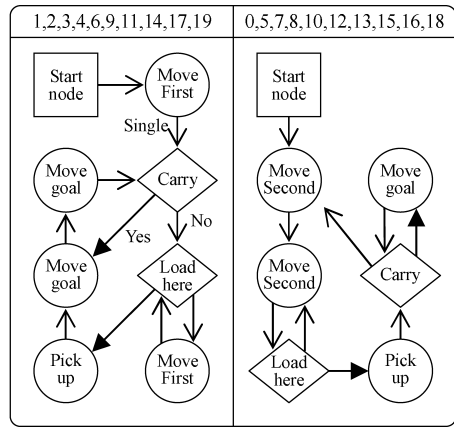


図 9 実験 2 における GNP with ADG での最良個体の例
Fig. 9 Acquired program in GNP with ADG for Experiment II.

表 4 実験 2 における各 ADG 手法の最良個体に含まれるノード数

Table 4 The number of nodes in two methods for Experiment II.

	Max.	Ave.	Min.	Stdev
GNP with ADG	44	23.4	8	7.0
GP with ADG	62	51.5	30	6.9

の方が解の探索が早いことが分かるが，表 4 から個体中には行動には不要なノードが多く含まれていることが分かる．そのため，GP により生成された行動規則には必要以上のルールが存在し，実際に利用されるルールの判別が難しくなっていると考えられる．

5. まとめ

本論文では，ネットワーク構造により行動規則を表現する GNP に，グループ化を行う ADG の枠組みを適用した GNP with ADG 手法を提案した．本手法を荷物搬送問題に用いることで，その有効性を検証した．GP と GNP の比較により，GNP による行動規則の可読性の高さを示した．また，提案手法と Homogeneous モデル，Heterogeneous モデルとの比較により，提案手法における ADG の枠組みにより，役割分担を適切に行えることを示した．特に，エージェントに能力差がある場合にも，ない場合にも提案手法役割分担に応じたコンパクトな行動規則を獲得した．

今後の研究として，図 5 に示したように，GNP は 1つのネットワークによっても，複数の役割を表現するための局所的な機能を独立に表現することが可能である．1つのネットワークで複数のエージェントが状況に応じて異なる行動をするように制御する手法を考察していきたい．

謝辞 本研究の一部は，文部科学省社会連携研究推

進事業（平成 17 年度～平成 21 年度）による私学助成を得て行われた。

参 考 文 献

- 1) Koza, J.R.: *Genetic Programming*, The MIT Press (1992).
- 2) 平澤宏太郎, 大久保雅文, 片桐広伸, 古月敬之, 村田純一: 蟻の行動進化における Genetic Network Programming と Genetic Programming の性能比較, 電気学会論文誌 C, Vol.121, No.6, pp.1001-1009 (2001/6).
- 3) 間普真吾, 平澤宏太郎, 古月敬之, 村田純一: 遺伝的ネットワークプログラミングのオンライン学習, 電気学会論文誌 C, Vol.122, No.3, pp.355-362 (2002/3).
- 4) 片桐広伸, 平澤宏太郎, 古月敬之, 村田純一: Genetic Network Programming とそのマルチエージェントシステムへの応用, 電気学会論文誌 C, Vol.122, No.12, pp.2149-2156 (2002).
- 5) 片桐広伸, 平澤宏太郎, 古月敬之: ノード数可変型 Genetic Network Programming, 電気学会論文誌 C, Vol.123, No.1, pp.57-66 (2003).
- 6) 平澤宏太郎, 大久保雅文, 古月敬之, 村田純一, 松家裕子: Genetic Network Programming によるヘテロマルチエージェントシステムの構成, 電気学会論文誌 C, Vol.123, No.3, pp.544-551 (2003).
- 7) 江口 徹, 平澤宏太郎, 古月敬之: Genetic Network Programming を用いた共生学習進化型マルチエージェントシステム, 電気学会論文誌 C, Vol.123, No.3, pp.517-526 (2003).
- 8) 中越 洋, 間普真吾, 平澤宏太郎, 古月敬之: マクロノード付遺伝的ネットワークプログラミング, 電気学会論文誌 C, Vol.124, No.8, pp.1619-1625 (2004).
- 9) Murata, T., Nakamura, T. and Nagamine, S.: Performance of Genetic Network Programming for Learning Agents on Perceptual Aliasing Problem, *Proc. IEEE Int Conf. System, Man & Cybernetics*, Vol.3, pp.2317-2322 (2005).
- 10) Holland, J.H., Holyoak, K.J., Nisbett, R.E. and Thagard, P.R., 市川ほか (訳): *インダクション*, 新曜社 (1991).
- 11) 片岡寛明, 原 章, 長尾智晴: 遺伝的オートマトン GAUGE, 情報処理学会論文誌, Vol.44, No.12, pp.3232-3241 (2003).
- 12) Luke, S. and Specter, L.: Evolving Teamwork and Coordination with Genetic Programming, *Proc. Genetic Programming 1996*, pp.141-149 (1996).
- 13) Iba, H.: Emergent Cooperation for Multiple Agents Using Genetic Programming, *Parallel Problem Solving from Nature IV*, pp.32-41 (1996).
- 14) Iba, H.: Multiple-Agent Learning for a Robot Navigation Task by Genetic Programming, *Proc. Genetic Programming 1997*, pp.195-200 (1997).
- 15) 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).

(平成 17 年 10 月 3 日受付)

(平成 18 年 3 月 2 日採録)



中村 貴志

1981 年生。2004 年関西大学総合情報学部総合情報学科卒業。2006 年関西大学大学院総合情報学研究科知識情報学専攻修士課程修了。遺伝的ネットワークプログラミング, マルチエージェントに関する研究に従事。



村田 忠彦 (正会員)

1971 年生。1997 年大阪府立大学大学院工学研究科電気・情報系専攻博士後期課程修了 (年限短縮 2 年)。博士 (工学)。足利工業大学工学部経営情報工学科助手, 講師を経て, 2001 年より関西大学総合情報学部助教授。2005 年より関西大学経済・政治研究所政策グリッドコンピューティング実験センター長を兼任。IEEE SMC Society において Soft Computing Technical Committee Chair, Japan Chapter 副支部長, Associate Editor を務める。また, 2007 年開催の Intl. Conf. on Evolutionary Multi-Criterion Optimization の Program Chair をはじめとして, 多数の国際会議のプログラム委員を務める。遺伝的アルゴリズムによる多目的最適化, 強化学習, マルチエージェントシミュレーション, グリッドコンピューティングに関する研究に従事。1997 年システム制御情報学会奨励賞受賞。日本知能情報フジ学会, IEEE, ACM 等の会員。