動的再構成可能ロボットシステムに関する研究*

(第12報, 遺伝的セル生成アルゴリズム(GCPA)に基づく 自己組織システム)

川内 陽志生^{*1}, 稲 葉 忠^{*1}, 福 田 敏 男^{*2}

A Study on Dynamically Reconfigurable Robotic System (12th Report, A Self-organizing System Based on a Genetic Cell Production Algorithm)

Yoshio KAWAUCHI, Makoto INABA and Toshio FUKUDA

Self-organizing systems and distributed systems have recently been studied by many researchers. The authors have studied a cellular robotic system called CEBOT as a dynamically reconfigurable robotic system. CEBOT consists of many sorts of functional units, which have simple functions, knowledge, and a database. CEBOT is required to perform many given tasks in various environments by reconfiguring the structures of both hardware and software. If CEBOT does not possess a proper subsystem, called a module, in its own system, it must create a module by combining cells. In the process of the cell production, we apply the genetic algorithms (GA) to the CEBOT. We propose the algorithm called the genetic cell production algorithm, (GCPA) in this paper. We also propose a performance index based on the theory of entropy in order to estimate the performance of distributed systems. To assess the GCPA and the estimation method, we conducted an experiment using a robot manipulator controlled by self-organizing knowledge system based on the GCPA.

Key Words: Robotics, Mechatronics, Self-organizing System, Self-Evolution, Cellular Robotic System (CEBOT), Genetic Algorithm, Genetic Cell Production Algorithm (GCPA)

1. は じ め に

著者らは今までに多種多様な環境下でさまざまなタスクを処理するロボットとしてセル構造化ロボット(以後 CEBOT)を研究, 報告してきた。CEBOT はシステム全体をセルと称するある単機能を有する要素に分解し, それらセルが環境, タスクに応じて結合の組合せを変化させシステム全体の形態を動的に再構成可能な生物的なロボットシステムを目指している。これまでにその基本概念, 自動結合, 分離実験, セル間通信, 最適形態化理論, 知識の最適配分, マスタセル決定法等について報告してきた^{(1)~(9)}。

生物組織がその形態, 性質を自己組織化していくように CEBOT においてもハードウェアとともにソフトウェアも自己組織的, 自己進化的であることが望まれる。

本報では分散システムの中に新規のサブシステムを生成し, それを自らの構造に組入れる自己組織システムについて報告する。2章では自己組織システムの概

念を CEBOT との関係において論じ, ある分散システムのモデルについて述べる。3章では自己組織システムの動作の基本アルゴリズムについて論ずる。自己組織化においては自己増殖が問題の一つと考えられるが新規サブシステムの生成の過程で生物の進化, 遺伝をモデル化し遺伝アルゴリズム (Genetic Algorithm) と呼ばれるアルゴリズムの手法を取り込んだ遺伝的セル生成アルゴリズム (Genetic Cell Production Algorithm, GCPA) を提案する。(自律分散, 自己組織化についてはパターン⁽¹⁰⁾⁽¹¹⁾の自己発生⁽¹⁰⁾⁽¹¹⁾や分散計算機環境⁽¹²⁾などの研究がなされているがハードウェア, ソフトウェアともに自己組織的, 自己進化的であるロボットシステムに関する研究はあまりなされていない。また GA についてもその研究の多くが組合せ最適化⁽¹³⁾等の探索問題に応用されており, ロボットシステムの自己進化性への応用は見当たらない。

4章ではサブシステム間が定量化された結合の強さで表される分散システムをエントロピーで評価する手法を提案し, 2章で記述されるモデルを用いてこの評価法の妥当性を論じる。そして5章では3章で提案されたアルゴリズムの動作を確かめるためにモデル中のサブシステムをハードウェアに対応付け, 未知の特性を持つ対象物を操作する過程において学習, 推論を行

* 原稿受付 平成4年3月5日。

^{*1} 正員, 東洋エンジニアリング(株)応用技術部 (●275 習志野市茜浜2-8-1)。

^{*2} 正員, 名古屋大学工学部 (●464-01 名古屋市中種区不老町1)。

うロボットマニピュレータについて述べる。そのロボットマニピュレータを用いて未知の対象物を操作させる過程における知識システム構造の変化を4章で定義される評価関数を用いて解析, 評価し実験結果とともに報告する。

2. 自己組織システム

2.1 ハードウェアの自己組織化 CEBOT の基本概念は環境, タスクに応じた最適形態に自らの形態を変えるロボットシステムである。多くのセルが無関係に混沌と混在している状態から, 環境, タスクに適応するためにあるセル間の関係が強まりモジュールと呼ばれるセルの集合体が生成される (図1参照)。

図1はその概念図であり環境タスクに応じてシステム内で自律的にいくつかのセルが集まりサブシステム(モジュール)が生成される様子を表している。著者らはハードウェアにおけるモジュールとは, 「セルが複数個結合されて一まとまりの機能を構成した結合可能な結合面を一つ以上有する知能を持った機能単位」と定義した⁽⁴⁾。システムに与えられるタスクの種類によってはある機能単位がしばしば用いられる場合, タスクが与えられるごとにセルレベルまで分解せず有用なモジュールをサブシステムとして有しているほうがセル間の通信量また分離, 結合の際の消費エネルギー減少の点で有利である。

2.2 ソフトウェアの自己組織化 ハードウェア同様に CEBOT はそのソフトウェアも自己組織的であることが望まれる。ソフトウェアの自己組織化もハードウェアと同様に知識等の基本要素の散在した状態から環境, タスクによって各要素間の関係が変化し, 密な関係を有する要素同士が集まり, より複雑なタス

クを遂行できるモジュール(知識モジュール)を作成する。このようないくつかの知識要素結合させる概念はミンスキーの K-Line 理論にも示されている⁽¹⁴⁾。ハードウェアの自己組織化と同様にシステム内に生成され, 残される知識モジュールは環境, タスクに依存する。以上の点でソフトウェアの自己組織化もシステム内の要素間の関係に着目しどのようにサブシステムとしてのモジュールが形成されまた分解されるのかという点ではハードウェアの自己組織化と同様に考えられる。

2.3 自己組織システムのモデル ハードウェアの自己組織化もソフトウェアの自己組織化もシステムを構成する基本単位の動的な離散集合という点では同じであるという立場から, 以下に本論文で対象とする自己組織システムのモデルについて述べる。

図2に本論文で用いる自己組織システムのモデルを示す。本システムは階層化された要素(Cell またはセル)の集合体である。図2に示されるように, 各セルは自分の属している層の上下層内の任意のセルと結合しており, 同一層内のセル間では結合はないものとする。各結線には結合係数(Connection weight)と称するある正の値が振り分けられており, 図3に示されるようにあるセルの下位層のセルとの結合係数を総和が

図1 ハードウェアの自己組織化の概念図

図2 自己組織システムのモデル

Connection weight: A positive number assigned to a connection between knowledge-cells
 $w(i,j), w(i,k), w(i,l), w(i,m)$

Connection intensity: A normalized connection weight

$$\frac{w(i,j)}{W}, \frac{w(i,k)}{W}, \frac{w(i,l)}{W}, \frac{w(i,m)}{W} \\ W = w(i,j) + w(i,k) + w(i,l) + w(i,m)$$

Knowledge: A vector composed of connection intensity

$$\text{Cell-}i[\frac{w(i,j)}{W}, \frac{w(i,k)}{W}, \frac{w(i,l)}{W}, \frac{w(i,m)}{W}]$$

図3 知識セルと結合係数, 結合強度, 知識の定義

1 になるように正規化したものを結合強度 (Connection intensity) と呼ぶ。セルは自らの結合強度を知識 (Knowledge) の一つとして記憶しているものとする (図3参照)。したがってシステムの状態は層の数、各層内のセルの数および各セルの有する知識 (結合強度) で記述される。各層内のセルの数および層の数は制限されないものとする。

3. 遺伝的セル生成アルゴリズム (Genetic cell production algorithm, GCPA)

図2に示されるモデルでは環境や与えられたタスクを満足するために必要なサブシステムがシステム内に存在しない場合は新しいサブシステムを作成する必要がある。そのためにどのようにして新しい要素を生成するかの方法が問題となる。そこで新しいセル生成の過程において生物の遺伝、進化をモデル化した遺伝アルゴリズム [Genetic algorithm (以下 GA)] 的手法を取り込んだ遺伝的セル生成アルゴリズム [Genetic cell production algorithm (以下 GCPA)] を提案する。GA は近年組合せ最適化問題⁽¹³⁾ などに対して応用されている探索法の一つでありその基本動作は以下に示される⁽¹⁵⁾。

1. 遺伝子に対応して対象物を符号化する (Coding)。
2. それぞれの符号列より計算される評価関数 (Fitness function) を設定し計算する。
3. より高い評価関数値を有する遺伝子 (符号列) が選択される (Reproduction)。
4. 選択された遺伝子間で交差が行われる (Crossing-over)。
5. ある確率で符号が変えられる (Mutation)。

以上のステップの3, 4, 5を一世代として以後何世代も繰返すことでより大きな評価関数値を有する遺伝子を選択していくことがGAの基本的考え方である。GAは生体系の適者生存をモデル化している。CEBOTもさまざまな変化する未知のタスク、環境に適応していく必要がある。そこでこのGAの考え方を図2, 3で示されるシステムの新しいセル生成に応用し、新たに遺伝的セル生成アルゴリズムを提案する。GAにおいては遺伝子およびFitness functionを決めねばならない。GCPAでは遺伝子として図3で定義されているセルの知識を用いる。Fitness functionには式(1)で定義される正解度 (Correctness) を用いる⁽⁸⁾⁽⁹⁾。

$$C_p(i) = \sum_j \{w(i, j) \times C_q(j)\} \quad \dots\dots\dots (1)$$

ここで $C_p(i)$, $C_q(j)$ はそれぞれ層 p に属するセル i , と層 q に属するセル j の有する正解度。 $w(i, j)$ は

セル i , セル j の間の結合強度を表す。システム内の任意のセルに正解度の初期値が与えられればすべてのセルの正解度は式(1)を用いて計算できる。正解度とは0以上1以下の値で、入力データとセルとのマッチングの度合を表すものである。この正解度の利用例はすでに報告してきた⁽⁸⁾⁽⁹⁾。

ここで提案するGCPAは以下の手順に従う。

1. 遺伝子としてセルの有する結合係数の列を対応させる (Coding)。
2. 各知識セルの有する正解度 (Correctness) を計算する (Fitness function)。
3. 正解度で重み付けされたルーレットを用いて新しい知識セルの候補1を選択する (Reproduction-1)。
4. 同じルーレットを用いて交差用の候補2を選択する (Reproduction-2)。
5. 交差位置 n をランダムに決め交差を行い交差後の候補1を新しい知識セルの有する知識とする (Crossing-over)。 (以上図4参照)。

表1 GAとGCPAの比較

Natural	GA	GCPA
gene	character	connection weight
chromosome	string	knowledge
phenotype	parameter set (quantity of systems)	ability of a system (quality of systems)
	fitness function	correctness

図4 遺伝的セル生成アルゴリズム

図5 システムの基本動作

表1にGAとGCPAまた生物システムとの対応関係を示す。GAは経路の探索や既存のシステムの効率的な組合せに応用されることが多いのに対し、GCPAは新しいサブシステムの生成に用いられておりサブシステムを生成するなどシステム全体の質を変化させている。したがってGAがシステムの「量」的なものを変化させるのに対しGCPAはシステムの「質」的なものを取扱っていると言える。図5に本システムの基本動作を示す。図5のボックスはそれぞれ以下に説明される処理を表す。

Process 1: 入力データとしていくつかのセルの正解度を設定する。

Process 2: システムへの入力に適応するサブシステムを探索する。システム内にある場合はProcess 4, ない場合はProcess 3を実行する。サブシステムの探索の一方法は既に報告している⁽⁸⁾⁽⁹⁾。

Process 3: 図4で示されるGCPAに基づいて新しいセルを生成する。

Process 4: セル間の結合係数を変化させる。

Process 5: セル, モジュールの数を再構成する。

Process 4, 5についてもその一方法を既に報告している⁽⁸⁾⁽⁹⁾。

4. サブシステム間の結合関係に 注目したシステムの評価法

前章のモデルではシステムの状態は層の数, 層内のセルの数, そして各セル間の結合強度の集合で記述される。しかしこれらすべてを明らかにしてもシステム全体の評価をすることは困難である。そこで図2, 3で表されるシステムを評価するための評価関数を設定する。ここではその評価関数に情報理論におけるエントロピーの概念を用いる⁽¹⁰⁾。情報理論におけるエントロピーは一般に以下の式で表される。

$$E = \sum_{x=1}^N \left\{ p(x) \times \log_n \frac{1}{p(x)} \right\} \quad \dots\dots\dots (2)$$

ここで $p(x)$ は事象 x の生起確率。

これは生起確率 $p(x)$ を有する N 個の事象を表記するのに必要な情報量を意味する。式の底 (n) が2の時はすべての事象を表記するのに必要なビット数を表す (以下 $n=2$ として考える)。全体のエントロピーの値は生起確率に偏りがあるほうが小さくなる。要素数が同じながらも要素間の結合強度の偏りによって値が変化することが本システムの評価関数としてエントロピーの概念を用いる理由である。そこで図6(a)に示されるセル- U_i (Cell- U_i) の有するエントロピーを式(2)の生起確率 $p(x)$ を各結線間の結合強度に置き換

えることで式(3)を用いて定義する。図6(a)のシステムの場合 Bottom Layer (セル数= n) のセルはより下位層のセルと結合していないためセルの有するエントロピーは零である。したがってシステム全体のエントロピーはセル- U_i の有するエントロピーに等しい [式(3)参照]。

$$E(U_i) = \sum_{k=1}^n \left\{ p(U_i, L_k) \times \log_n \frac{1}{p(U_i, L_k)} \right\} \quad \dots\dots\dots (3)$$

次に図6(b)で示されるシステムのエントロピーについて以下のように考える。図6(b)は図6(a)中の Upper Layer と Bottom Layer (セル数= n) の間に Middle Layer (セル数= m) があり, Middle Layer の Cell- M_j は Bottom Layer のセル (l_j 個) と結合していると仮定すると式(4)が成立する。

$$\sum_{j=1}^m l_j = n \quad \dots\dots\dots (4)$$

Upper Layer 内の Cell- U_i と Middle Layer 内の Cell- M_j の結合強度を $q(U_i, M_j)$, Cell- M_j と Cell- L_k 間の結合強度を $r(M_j, L_k)$ と表記する。Upper Layer のセルから Bottom Layer の各セルに到る結線上の結合強度の積は図6(a)と(b)の場合で同じであると仮定すると以下の式(5)が成立する。

$$q(U_i, M_j) \times r(M_j, L_k) = p(U_i, L_k) \quad \dots\dots\dots (5)$$

またどのような l_j に対しても式(5)が成立するように $q(U_i, M_j)$ は式(6)を満たすものとする。

$$q(U_i, M_j) = \sum_{k=1}^{l_j} p(U_i, L_k) \quad \dots\dots\dots (6)$$

(a) 2層よりなる分散システムの例

$$p(U_i, L_k) = q(U_i, M_j) \times r(M_j, L_k)$$

(b) 3層よりなる分散システムの例

図6 各分散モデルとシステムのエントロピー

以上の仮定のもとシステム全体のエントロピーを以下の式で定義する。

$$E(\text{system}) = \sum_{i=1}^n \{\text{path}(L_i) \times E(L_i)\} \quad \dots\dots (7)$$

ここで $E(\text{system})$ はシステム全体のエントロピー、 $E(L_i)$ は式(2)で定義されるセル- L_i の有するエントロピー、 $\text{path}(L_i)$ は最上位の知識セルからセル- L_i に到るまでの結合強度をすべて掛け合せたもの。式(7)の妥当性を確認するため Bottom layer のセル- L_1 からセル- L_n を含む斜線で示されるサブシステムのエントロピーを計算する。図6(a)の場合のサブシステムの評価関数値 [E [hatched area of 図6(a)]] は式(3)の n に l_1 を代入した値である [式(8)参照]。

$$E[\text{hatched area of 図6(a)}] = \sum_{k=1}^{l_1} \left\{ p(U_i, L_k) \times \log_n \frac{1}{p(U_i, L_k)} \right\} \quad \dots\dots (8)$$

図6(b)の場合の評価関数値 [E [hatched area of 図6(b)]] は式(7)の定義式より式(9)で表される。

$$\begin{aligned} E[\text{hatched area of 図6(b)}] &= q(U_i, M_1) \\ &\times \log_n \frac{1}{q(U_i, M_1)} + q(U_i, M_1) \\ &\times \sum_{k=1}^{l_1} \left\{ r(M_1, L_k) \times \log_n \frac{1}{r(M_1, L_k)} \right\} \\ &\dots\dots\dots (9) \end{aligned}$$

式(9)の第1項は Upper layer の Cell- U_i の有するエントロピーの一部、第2項は Middle layer のセル- M_1 の有するエントロピー。式(5)より式(9)の第2項 [$E(M_1)$] は以下のように変形できる。

$$\begin{aligned} E(M_1) &= q(U_i, M_1) \times \sum_{k=1}^{l_1} \left\{ \frac{p(U_i, L_k)}{q(U_i, M_1)} \right. \\ &\times \log_n \frac{1}{r(M_1, L_k)} \left. \right\} \quad \dots\dots\dots (10) \end{aligned}$$

$$\begin{aligned} E(M_1) &= \sum_{k=1}^{l_1} [p(U_i, L_k) \times \{\log_n q(U_i, M_1) \\ &- \log_n p(U_i, L_k)\}] \quad \dots\dots\dots (11) \end{aligned}$$

$$\begin{aligned} E(M_1) &= \sum_{k=1}^{l_1} \left\{ p(U_i, L_k) \times \log_n \frac{1}{p(U_i, L_k)} \right\} \\ &+ \sum_{k=1}^{l_1} \{ p(U_i, L_k) \times \log_n q(U_i, M_1) \} \quad \dots\dots (12) \end{aligned}$$

式(6)を用いると式(12)の第2項は式(13)のように変形できる。

$$\begin{aligned} \sum_{k=1}^{l_1} \{ p(U_i, L_k) \times \log_n q(U_i, M_1) \} \\ = q(U_i, M_1) \times \log_n q(U_i, M_1) \quad \dots\dots\dots (13) \end{aligned}$$

したがって式(12)、式(13)の右辺を式(9)に代入することで図6(b)の斜線部で示されるサブシステムの評価関数値は式(14)で表される。

$$E[\text{hatched area of 図6(b)}] = q(U_i, M_1)$$

$$\begin{aligned} &\times \log_n \frac{1}{q(U_i, M_1)} \\ &+ \sum_{k=1}^{l_1} \left\{ p(U_i, L_k) \times \log_n \frac{1}{p(U_i, L_k)} \right\} \\ &+ q(U_i, M_1) \times \log_n q(U_i, M_1) \\ &= \sum_{k=1}^{l_1} \left\{ p(U_i, L_k) \times \log_n \frac{1}{p(U_i, L_k)} \right\} \quad \dots\dots (14) \end{aligned}$$

図8の(a)、(b)では最上位の Cell- U_i と最下位層の Cell- L_1 から Cell- L_n までの関係は式(5)で保存されているため評価関数値は同じになるはずである。定義式 [式(7)] より求めた両ケースの評価関数値は式(8)と式(14)に示されるように同じ値をとる。したがってシステム全体のエントロピーを式(7)で定義する妥当性が示される。この式(7)で定義される評価関数はサブシステム間の結合関係に注目しておりサブシステムの質などには無関係である。したがって本評価手法は CEBOT のみならず、サブシステムが定量化されたある関係を有しながら存在するその他の一般的システムの評価にも用いることが可能である。

5. 実験装置の構成および実験

本報で論じている分散システムは図2、3に示される構造を有している。各セルはあるサブタスクを処理するための、サブシステムとして考えられ、それらセルが効率的に組合されることで、より高度なタスクが処理される。そこでセルの効率的組合せ、生成の動作を確かめるために、システムの動作のある特別な場合として以下に示す実験装置を用いた実験を行った。

5.1 実験装置のシステム構成 実験装置はタスクを処理する「ロボットマニピュレータ」とコンピュータからタスクをセットされる「タスクボード」からなる(図7参照)。ロボットマニピュレータは三菱電機

図7 実験装置の外観図

(株)製のムーブマスタを用い、エンドエフェクタ部に6軸の力トルクセンサを取付けた。タスクボードは各々番号をつけられた9個のジョイスティックから構成され、すべてのジョイスティックは上下左右と4方向に可動である。マニピュレータ、タスクボードともにコンピュータに接続されておりコンピュータよりタスクボードに対してあるタスクが埋め込まれる。本実験においてはタスクを以下のように定義する。

タスク：ある決められた順番で決められた番号のジョイスティックを決められた方向に動かすこと。

「ステップ1においてNo.2のジョイスティックを右に動かし、ステップ2でNo.5を上方向に動かす。」ことがここで述べられるタスクの一例である。またマニピュレータの制御も同じコンピュータを用いて行われているがどのようなタスクが与えられたかの通信は行われていない。またマニピュレータはタスクの終了をコンピュータからのタスク終了信号を受け取ることで認識する。

5.2 CEBOTと実験装置との対応関係 本実験ではタスクボードに与えられたタスクを処理していくうちに3章で記述したアルゴリズムに従って新しい知識セルを生成し、また結合係数を変化させるを行った。ここではタスクはジョイスティック (Element) とそのジョイスティックの操作方向 (Operation) の組合せで記述される。これらタスクの記述法、データ構造は既に発表している⁽⁸⁾。そこでジョイスティックのアドレスや動作方向をサブシステムと見立てて図2で示されるセルに対応させ、ジョイスティックと動作方向の組合せをそれらセルの集合で記述するものとする。図2の知識セルをCEBOTのハードウェアのセルと置き換えて考えればセル間の結合係数を変化させることは、ある機能を有するセル間の関係を強めたり弱めたりすること、すなわち2章で論じたようにモジュールを作成したり分解することに対応する。

本実験では図2、3で定義、モデル化されているシステムの特別な場合として4層構造のシステムを考える⁽⁸⁾⁽⁹⁾。各層は下位より第1層、第2層、第3層、第4層と呼ばれる。タスクはジョイスティックと動作方向の番号の組合せで記述されるため、既存知識セルの第1層 (Layer-1) を可動方向を記述する層とし、その上位に第2層 (Layer-2) を作成し、ジョイスティックの番号を表す層とする。それらの組合せによりなるタスクを記述する層は第3層 (Layer-3) にまたタスクの集合を第4層 (Layer-4) のセルに割り当てる。表2に分散知識システムと実験装置の対応関係を示す。また図8に実験におけるマニピュレータ、タスクボード、

GCPA の関係を示す。オペレータは操作すべきジョイスティックのアドレスをコンピュータとGCPAに基づく自己組織システムに与える。コンピュータは表3および次節で定義される G_{ij} や H_i で記述されるあるモデルを有しており、そのモデルを用いてタスクを生成しタスクボードに設定する。自己組織システムでは与えられたジョイスティックに対応するセルの正解度が1に設定され図4、図5に示されるアルゴリズムに従って動作列が生成されマニピュレータを用いてその動作をタスクボードに対して行う。マニピュレータの行った操作があらかじめ設定されたタスクに等しい場合のみ信号が自己組織システムに送られる。自己組織システムは多くの試行を通じてモデルを自らの中に作成していく (図8参照)。

実験の初期においては自己組織システム内の知識セルは等しい強度で結合しており、また第3、4層のセルの数も少ないため操作列生成も順列組合せ的に手あたり次第にタスクボードを操作するのでタスクボードに設定されたタスクを処理するのに時間がかかる。しかし何度でもタスクボードを操作するうちにタスクボー

表2 分散システムモデルと実験装置の対応関係

Knowledge cells	Experimental equipment
Layer-4	Complicated tasks
Layer-3	Given tasks
Layer-2	Address of joy-sticks
Layer-1	Operational direction

表3 実験時のモデルの設定値

H_1	H_2	H_3	H_4				
7	84	29	31				
G_{11}	G_{12}	G_{13}	G_{14}	G_{31}	G_{32}	G_{33}	G_{34}
0.871	0.043	0.043	0.043	0.043	0.043	0.871	0.043
G_{21}	G_{22}	G_{23}	G_{24}	G_{41}	G_{42}	G_{43}	G_{44}
0.043	0.871	0.043	0.043	0.043	0.043	0.043	0.871

図8 実験におけるGCPAと実験装置の関係

ドの特性, 例えばジョイスティック-1は1の方向に倒すと正解が得られやすいといったことを学習していきだんだんと正解のタスクを得るまでの時間(操作数)の減少が期待される。以上のことを以下に示す実験において確認する。

5.3 実験手順 実験は以下の手順に従って行われた。

- 1: タスクボード上の各ジョイスティック(i)の動作方向(j)が正解である相対度数(G_{ij})を与える。
- 2: 各ジョイスティック(i)にある正の定数(H_i)を与える。
- 3: 正解のタスクをタスクボードにセットする。

正解のタスクとはどのジョイスティックをどの方向にどのステップで動かせば良いかで記述される。各ジョイスティックの正解の方向は G_{ij} で、またジョイスティックを操作する優先順位は H_i を用いて確率的に決められる。例えば $G_{ij}=0.8$ の場合ジョイスティック- i は8割の確率で j の方向が正解の方向として選ばれる。しかし必ずしも j 方向だけに倒せば良いわけではなく2割は他の方向が正解となる。これら正解の方向は各試行ごとに変化する。

4: ロボットマニピュレータにそのタスクを遂行するために動かすジョイスティックの番号を知らせる。

これは図5の Process 1 に対応する。

5: タスクを遂行するまで(タスク完了の信号がコンピュータからくるまで) マニピュレータはジョイスティックをさまざまな方向へ動かす。

6: タスク遂行の際マニピュレータは知識構造を変化させまた新しい知識セルを作成して与えられたタスクボード上のタスクをより効率よく処理するために知識の自己組織化を行う。

図5の Process 2 から Process 5 を実行。

7: 3にもどる

以上の手順を行わせ、ジョイスティックを操作した総数(以下操作回数)を記録していく。

図9 実験結果(試行回数と操作数の関係)

入力データはジョイスティック番号として1-2, 2-3, 3-4, 1-4の4種類をシステムに与えた。簡単のため正解タスクは2ステップと仮定した。 G_{ij} , H_i の初期値は表3に示される。

5.4 実験結果 前節で述べた方法にしたがって実験を行い、マニピュレータが入力データを得てからタスクボードに対して正しい操作をしてコンピュータからタスク終了信号を受信するまでに何ステップ、ジョイスティックを操作したかを記録しグラフにした(図9参照)。

図9は横軸に試行回数、縦軸に操作回数をとっている。試行回数が少ない時はタスクボードの特性をつかんでおらずランダムにジョイスティックを操作するため操作回数が多くなっている。試行回数が増えるごとに知識セルが増し知識セル間に密な関係と粗な関係ができる。これはタスクボード上の各ジョイスティックと正解になりやすい操作方向が密に関係しあうことに対応する。これはCEBOTにおいては与えられるタスク、環境によって、特定のセル同士の結びつきが強まり、モジュールを形成しやすくなることに対応している。試行回数が増加していくとその情報を用いてタスクボードを操作するため、操作回数が減少する。

図9中で試行回数が上に飛び出ているところは表3の G_{ij} の低い確率のところが正解タスクとして選択された場合で、新しいセルを作成している時である。新しいセル生成の際の操作回数も減少の傾向がみられ本システムが学習を通して与えられたタスクボードの特性を学び、より効果的にセルを生成していると言える。

5.5 評価 図9はマニピュレータのジョイスティック操作数を記録したものである。その際にも知識システムはセル数、結合強度が変化している。図9からではマニピュレータにインストールされている知識システムの評価は明らかでない。そこで4章で定義し

図10 評価結果

た評価関数を用いて、本実験中に変化しているシステムを定量的に評価した (図 10 参照)。

図 10 の横軸は試行回数を、縦軸は学習後のシステムの評価関数値を学習前のシステムの評価関数値で割ったものである。学習前のシステム状態とはすべてのセルが等しい結合強度で結合している状態を意味し、学習後のシステムとは対象物を操作する過程で各セル間の結合強度が変えられた状態を意味する。ここで学習とはある動作がなされ、それが成功した場合にその動作を表記するために用いたセル間の結合強度を強め、失敗した場合に弱めることを意味する⁽⁸⁾。

図 10 より試行を重ねるごとにセルの関係に密なところと粗なところが形成されていることがわかる。実験の最初は各セル間の結合強度は等しく設定されていたため等結合をしたシステムの評価関数との比は 1 であったのが、試行を重ねるごとに結合強度が変化したためである。評価関数の減少傾向からこの実験を通じて GCPA に基づく自己組織システムの有効性が示される。

6. ま と め

本論文では CEBOT のハードウェアおよびソフトウェアの自己組織化の考察から自己組織化モデルを提案し以下のことについて報告した。

(1) 多くのサブシステムの結合関係で記述されるシステムの評価の一方法としてエントロピー理論を応用した評価関数を定義した。

(2) 新しいサブシステム作成の際に生物の適者生存の原理を応用した遺伝的セル生成アルゴリズム (GCPA) を提案した。

(3) GCPA で作動する自己組織システムの応用例として未知の対象物の特性を学習するシステムを作成しそれをロボットマニピュレータと連結させた。

(4) 実験結果および解析結果より GCPA に基づく本システムの有効性を示した。

以上セル構造システムの自己組織化およびその一応用法について論じた。GCPA はサブシステム間の結合強度を操作するものである。したがってサブシステムを CEBOT のハードウェアのセルと見立てたり、またソフトウェアのセルと見立てることでハードウェア、

ソフトウェアの自己組織化への応用が可能であると考ええる。

文 献

- (1) 福田・中川, セル構造を有する自己組織化ロボット—その基本概念とセル間の粗接近制御及び形態決定法について—, 電学論 C, 107-11(1988), 1019-1026.
- (2) 福田・中川, 動的再構成可能ロボットシステムに関する研究(第 1 報, セル間の自動接近, 結合, 分離制御実験), 機論, 55-509, C(1989), 115-118.
- (3) 福田・中川, 動的再構成可能ロボットシステムに関する研究(第 2 報, セル構造マニピュレータの最適形態化), 機論, 55-516, C(1989), 2009-2106.
- (4) 福田・川内・マーチン・浅間, 動的再構成可能ロボットシステムに関する研究(第 3 報, セル構造化ロボット CEBOT の認識, 通信システム), 機論, 56-523, C(1990), 709-716.
- (5) 福田・川内・浅間, 動的再構成可能ロボットシステムに関する研究[第 4 報, 分散知能システムとしての Cellular Robotics (CEBOT) の通信量と知識量による解析と評価], 機論, 56-532, C(1990), 151-158.
- (6) 福田・川内・原, 動的再構成可能ロボットシステムに関する研究(第 6 報, 分散知能システムにおける知識の自己再配分の一方法), 機論, 57-536, C(1991), 236-243.
- (7) 福田・植山・ほか 5 名, 動的再構成可能ロボットシステムに関する研究(第 7 報, ネットワークのエネルギー評価方法に基づくマスターセル決定法), 機論, 57-537, C(1991), 234-242.
- (8) 福田・川内・原, 動的再構成可能ロボットシステムに関する研究(第 8 報, セル構造化ロボット (CEBOT) の知能システムとしての, 推論, 学習を行う自己組織形分散知識システム], 機論, 58-545, C(1992), 208-216.
- (9) 福田・川内・原, 動的再構成可能ロボットシステムに関する研究(第 9 報, 自己組織化分散知識システムの定量解析及び最適形態), 機論, 58-546, C(1992), 196-203.
- (10) 向井・宇野・鈴木, 自律分散制御による多足歩行パターンの自己組織, 第 3 回自律分散システム・シンポジウム講演集, (1992-1), 25-30.
- (11) 湯浅・伊藤, 自律分散システムとその歩行パターン発生器への応用, 計測自動制御学会論文集, 25-2(1989), 1091-1097.
- (12) 神余・竹垣, 自己調整を行う分散計算機環境, 文献(10)の 179-182 ページ.
- (13) 柴田・福田・新井・小菅, Genetic Algorithm を用いた移動ロボットの最適行動計画, 日本ロボット学会第 9 回学術講演会予稿集, (1991-11), 589-599.
- (14) Minsky, M. (安西訳), 心の社会, (1990), 112-113, 産業図書.
- (15) David E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning", (1989), 1-23 Addison-Wesley Publishing Company Inc.
- (16) アブラムソン(宮川訳), 情報理論入門, (1949), 12-52, 好学社.