

木構造表現された論理回路への 進化論的設計の適用

2K-10

千葉慎二

仙台電波工業高等専門学校

1. はじめに

近年、進化論的設計手法によりハードウェアそのものを変更し、目的のハードウェアを構成する"進化するハードウェア(EHW: Evolvable Hardware)"の研究が進められており、適応速度の高速化、新たな論理回路設計手法として期待されている[1]。

本研究では、論理回路を2入力1出力の基本セル(入力のANDかOR、その出力をD-F/Fに出力するか否か等を設定可能)をノードとした、部分木の共有、順序回路でのループ構造の表現も可能な木構造で表現し、そのような木構造表現された論理回路の進化論的設計手法について報告する。

2. 論理回路の木構造表現

論理回路は、図1に示す2入力(入力信号1、2)1出力(出力信号)の"ノードセル"を複数組み合わせた形で表現する。ノードセルは、2入力の論理積(論理和)の出力をD-F/Fにバッファリングするか否かの機能を持つ。ノードセルの構造と接続状態によって、様々な回路表現が可能となる。

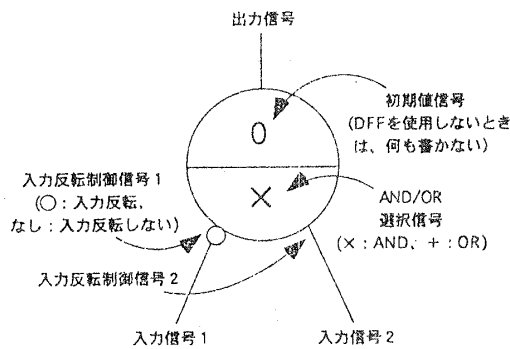


図1: ノードセルのシンボル

ノードセルの組み合わせによって、1ビット出力論理回路は図2のように木構造で表現される。

回路を構成する各ノードセルにユニークな番号(ノード番号)を付け、その番号の明示が必要な場

合はノードセルの左上に明記する。ノードセルで回路を構成するとき、あるノードセルの出力を2つ以上のノードセルの入力とすると、出力元のノードセルをリンクルート、出力先のノードセルの入力をリンクリーフと呼ぶことにする。このような接続を許すことで、回路構成に使用するノードセル数を少なくすることができる。ただし、リンクによって組み合わせ回路によるループができると回路が不安定となる可能性があるため、そのようなリンクは禁止している。また、木構造のリーフについては、回路の入力端子(入力リーフ)、リンクリーフの2種類がある。入力リーフについては、入力端子を番号(入力端子番号)で区別し、その入力端子番号で表す。リンクリーフは、対応するリンクルートのノード番号を円で囲った記号で表現する。回路の出力端子は、木構造のルートにあるノードセルの出力とする。

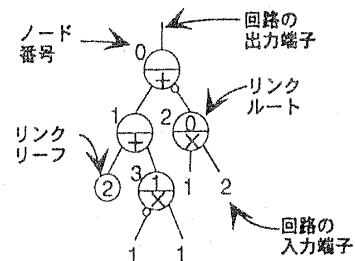


図2: ノードセルによる回路例

3. 遺伝的操作

ノードセルによる木構造に対して、突然変異、交叉を行なう。

突然変異は、任意のノードセルに対して、構造変化、それ以降の部分木の削除、部分木の付け足しのいずれかを行なう。

交叉は、2入力ノードセルによって表現された木構造を、連続した同一構造のノードセルを1つの多入力ノードセルに変換し、論理的に定数と見なせる部分は定数で置き換えた木構造に対して行なう(図3)。上記の変換をした任意の2つの木構造に対して、任意のノードセル以降の部分木の交換を行なう。

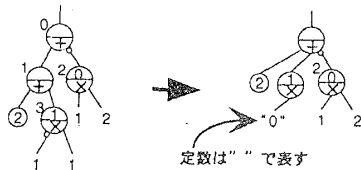


図3：多入力ノードセルによる木構造への変換

また、遺伝的操作の影響が染色体の広範囲に及ばないようにするため、リンクルートに対しての遺伝的操作を行う場合、対応する任意のリンクリーフにリンクルート以降の部分木をコピーし、コピー先を新たなリンクルートに更新してから行うようにしている。

4. 迷路探索回路の進化論的設計

本研究の適用例として、迷路探索回路の設計を行なった。迷路は34×34の2次元格子とし(図4)、その迷路内のゴールを探索するロボットの行動をシミュレーションする。ロボットは自身の位置に隣接する前方、右、左の格子が壁か否かを識別するセンサ(3ビット)、左右の駆動輪を回転させるか否かのアクチュエータ制御信号(2ビット)を備えている。アクチュエータ制御信号によって、シミュレーションの1ステップで、1格子分前進(左右駆動輪回転)、90度右回転(左駆動輪回転)、90度左回転(右駆動輪回転)の3つの行動をとれる。なお、左右の駆動輪が停止している状態は禁止状態としている。ロボットは上記のセンサ、アクチュエータを利用してゴールを探索するのであるが、その探索アルゴリズムは与えられておらず、進化論的な設計手法で獲得する。

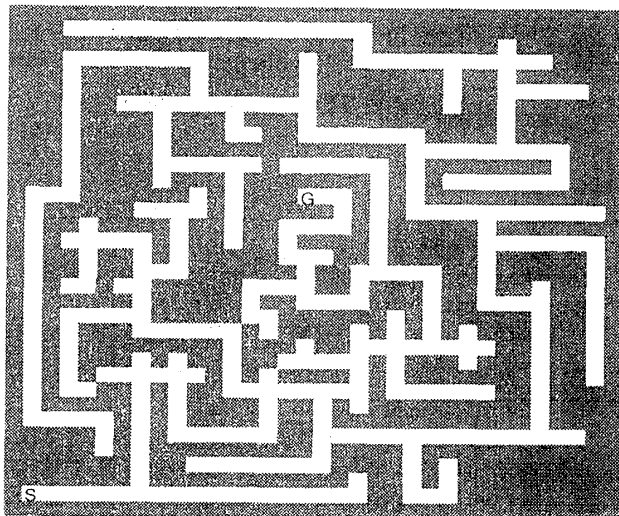


図4：探索する迷路 (s:スタート、G:ゴール)

ロボットの迷路探索アルゴリズムを表現する3入力2出力の論理回路を染色体とし、個体の総数は

400とした。交叉確率は0.6、突然変異確率は1ノードセル当たり0.05とし、エリート戦略として次世代に上位ランクから0.05%の染色体をそのまま残した。また、染色体を構成するノードセル数の上限を10とし、それ以上の規模の回路は作成しないようにした。

各ロボットは最大1000ステップのシミュレーションを行ない、その染色体を評価する。染色体の評価は、1000ステップ以内にゴールした場合は評価値を1、壁に衝突した場合は0.2、禁止行動をとった場合は0.1、1000ステップに達した場合は、スタートからゴールを結ぶ最短距離に含まれる分岐点をチェックポイントとし、それらチェックポイントの通過数に比例して0.2~1の評価値を与えた。

各染色体のランクを、評価値、ノードセル数、D-F/F数で決める。それら3つの値には評価値、ノードセル数、D-F/F数の順に優先順位があり、”評価値が大きく、ノードセル数が少なく、D-F/F数が少ない”染色体ほど上位のランクに位置するとした。次世代の染色体の親の選択には、このランクによるルーレット選択を行なった。これにより、たとえ評価値が1の染色体であっても冗長な回路を表現する染色体は淘汰されていくことになる。

表1は、乱数の初期値を変えて10回シミュレーションした結果である。正解(評価値=1)を示す染色体は、平均して28.2世代で出現しており、最適解(評価値=1、かつ、ノード数、D-F/F数が収束)を示す染色体は、平均して55.2世代で出現した。

表1：迷路探索回路の進化論的設計結果

試行 No.	正解			最適解		
	世代	ノード数	D-F/F数	世代	ノード数	D-F/F数
1	4	4	2	5	2	1
2	32	9	4	43	2	1
3	4	3	1	5	2	1
4	52	7	3	73	2	1
5	26	6	2	46	2	1
6	5	8	3	48	2	1
7	35	4	2	99	2	1
8	55	7	2	93	2	1
9	43	6	1	84	2	1
10	26	3	1	56	2	1
平均	28.2	5.7	2.1	55.2	2	1

5. おわりに

ノードセルを基本とした木構造で表現された論理回路に進化論的設計を適用し、その有効性を示した。

[1] X. Yao, T. Higuchi Evolvable Systems: From Biology to Hardware, Lecture Notes in Computer Science 1259, pp.55-78, Springer Verlag, 1997.