

C-002

免疫システムアーキテクチャとその評価

Architecture for immune system

尾関 剛[†] 井上 勇氣[†] 香川 憲市[†] 吉川 雅弥[†] 寺井 秀一[†]
 Tsuyoshi Ozeki[†] Yuki Inoue[†] Kenichi Kagawa[†] Masaya Yoshikawa[†] Hidekazu Terai[†]

1. まえがき

ナーススケジューリング問題 (以下 NSP と略記する) とは、医療機関において看護婦にできるだけ良い労働条件を提供するために、勤務負荷のバランスを取りながら勤務シフトを決定する問題である。NSP は互いに競合する複数の評価項目を持つ多目的性と、様々な重要度を持ち、互いに干渉し合う多くの制約条件を持つ多重制約性の2つを併せ持つ問題である[1]。このような組み合わせ最適化問題を解くためのアルゴリズムの1つとして、免疫アルゴリズム (Immune Algorithms: 以下 IA と略記する) [2],[3]が知られている。IA は、生物の持つ自己防御機構である免疫システムを模倣した最適化手法であり、多様性のある抗体産生機構や自己調節機構、記憶学習機構などがある。しかし、非常に多くの計算処理時間を要するため、ソフトウェアで実行する場合、高速化には限界があり、専用ハードウェアの開発が期待される。そこで本論文では、高速な免疫応答を実現する専用プロセッサを提案する。

2. 免疫アルゴリズム

生物は有害な抗原に対する自己防衛機構を持っており、それを免疫系という。免疫系では、病原菌やウイルスなどの抗原が体内に侵入してくると、それらを排除するのに有効な抗体を選択的に産生する。しかし、抗原を排除し、抗体が体内で異常に増えすぎた場合には、その抑制を行う。また、免疫系において1度免疫が成立すると、免疫応答は長い間保持される。よって、2回目以降の免疫応答は、より強く起こる場合が多い。これらの抗原に対する抗体の振る舞いを、工学的に模倣したアルゴリズムが IA である。組み合わせ最適化問題に IA を適用する事を考えた場合、抗原が対象問題であり、抗体がその解に相当する。図1にIAのフローチャートを示す。

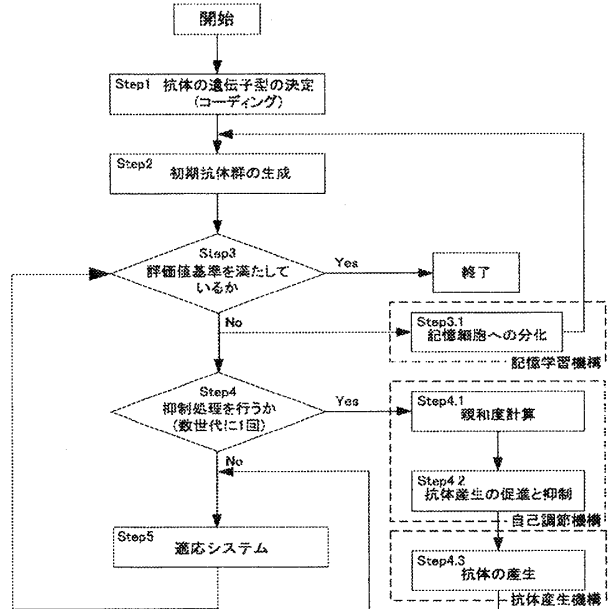


図1. IAのフローチャート

図2. コーディング

3. IAのハードウェア化

3. 1. コーディング

NSP は、看護婦 i の j 日の勤務 X_{ij} を要素とする行列を決定する問題である。 X_{ij} は基本的な勤務パターンである、日勤・準夜勤・深夜勤・休日のいずれかである。これらを日勤 00, 準夜勤 01, 深夜勤 10, 休日 11 とし抗体のコーディングを行った。図2に本研究で用いたコーディングを示す。看護婦数を可変とし、最小看護婦数を16人、最大看護婦数を48人とした。また、対応日数は31日固定とし、32日目の列には各抗体のIDを格納した。

3. 2. 制約条件と評価手法

本研究で採用した制約条件を表1に示す。ここで(1)~(5)の条件を絶対制約条件とした。また、制約条件の取り扱いとして、以下の2つの手法を考案した。

- (i)制約条件をペナルティーとして評価関数に取り込む。
- (ii)制約条件を崩さないように適応システムに工夫を加える。

全ての制約条件を評価関数に取り込むと、絶対制約条件を満たさない実行不可能な解が大多数を占めてしまう。従って、制約条件の中でも、絶対制約条件 ((1)~(5)) に対しては(i)の手法を適用せず、(ii)の手法によって条件を満たすようにし、その他の制約条件に対しては(i)の手法を用いた。

ここで図2のコーディングを、1ヶ月(31日)のスケジュールを日ごとに見た縦方向の制約と、看護婦ごとに見

[†]立命館大学 理工学研究科

[†]Graduate School of Science and Engineering, Ritsumeikan Univ.

表1. 制約条件

制約条件の分類	制約条件の内容
人に関する制約	(1) 婦長・主任は夜勤をしない (2) 夜勤者はあるレベル以上
1日ごとの制約	(3) 日勤者数 ≥ 必要日勤者数 (4) 深夜勤者数 = 必要深夜勤者数 (5) 半夜勤者数 = 必要半夜勤者数
勤務状態の前後関係に関する制約	(6) 基本夜勤パターンを守る (7) 連続夜勤日数を守る (8) 夜勤パターンと夜勤パターンの間は少なくとも1週間あける (9) 日勤→深夜勤の組み合わせは禁止 (10) 半夜勤→日勤の組み合わせは禁止 (11) 半夜勤→深夜勤の組み合わせは禁止 (12) 休日の前日に半夜勤は禁止 (13) 休日の翌日に深夜勤は禁止
各看護婦の1月分のスケジュールに関する制約	(14) 公休日数 = 指定公休日数 (15) 夜勤日数が看護婦の中で均等化されているか

た横方向の制約に分ける。表1では、(1)~(5)が縦方向の制約、(6)~(15)が横方向の制約になる。縦方向の絶対制約条件に対しては、予め制約条件を満たした抗体群を発生させ、適応システムにおいても縦方向の制約を満たす処理を行うことで、絶対制約条件を満足させるようにした。一方、横方向の制約に対しては、制約違反があるごとに評価値に反映させた。

本研究では、評価方式はペナルティ方式をとった。したがって、評価値の小さな抗体の方が優良抗体である。評価方法は(6)~(15)の制約条件を、6つの評価項目に分割して評価を行い、各抗体の評価値は6つの評価項目からのペナルティの線形和で算出する。

3. 3. 適応システム

免疫系は、遺伝子の再構成によって多様な抗体を発生することで、生体内に侵入する抗原に対応させる。これは、遺伝的アルゴリズム (以下、GA と略記する) と同じように免疫系が抗体遺伝子の再構成による多点探索処理を行うことを示している。そこで、本研究での適応システムでの処理は、選択・交叉・突然変異の3つの処理で構成する。

3. 4. 自己調節機構

自己調節機構は、親和度計算、期待値計算、抗体産生の促進と抑制を行うことで実現する。

まず、親和度計算では、解集団の中の抗体同士の親和度(類似度)を考え、解集団の中に自分と類似した抗体がどの程度存在するかを計算する。抗体同士の親和度は、それぞれの抗体で対応する格子を順に比較し、一致する格子の数が予め設定しておいた閾値を超えた場合に類似していると考えられる。ただし、ここでは抗体に多様性をもたせるため、類似している2抗体のうち、評価値の低い抗体の親和度に1を加えることにした。

次に求めた親和度を用いて全抗体に占める各抗体の濃度を求める(式①)。

$$\text{濃度} = \text{各抗体の親和度} / \text{全抗体数} \dots \text{①}$$

この濃度は、ある種類の抗体が免疫系全体に占める割合を表している。最後に濃度を用いて期待値を求める(式②)。

$$\text{期待値} = 1 / \text{濃度} \cdot \text{評価値} \cdot \dots \text{②}$$

但し、評価値はペナルティ方式なので、優良抗体ほど評価値(ペナルティ)が低い。この期待値が予め定められた期待値の閾値を超えない抗体は、淘汰される。

3. 5. 抗体産生機構

解集団として抗体数を保つため、期待値計算によって淘汰された抗体に代わる新たな抗体をランダムに産生する。但し、この場合も表1の(1)~(5)の絶対制約条件を満たした抗体を発生させる。

4. 回路構成

以上述べたアーキテクチャに基づく免疫システムエンジンの回路ブロック図を図4に示す。

ここでは、図4でのデータフローを図1のフローチャートに沿って説明する。

Step1のコーディングは3.1で述べたので省略する。

Step2の初期抗体群の生成は、外部の初期抗体生成モジュールで行う。初期抗体データは評価回路とデータパススイッチへ送る。評価回路へ送った抗体データは各抗体ごとに評価値を計算し、選択回路に送る(Step3)。その際、評価値基準を満たした抗体が存在すれば、処理は終了する。一方、データパススイッチへ送った抗体データは、そのまま現代メモリに格納する。ここで、Step4の抑制処理を行うかどうかの判断を迫られるが、この場合のデータフローは後で述べる。

Step5の適応システムでは、全ての抗体の評価値を選択回路に送信したら、選択回路はその評価値データをもとに、現代メモリから抗体を選択し、交叉回路、突然変異回路の順に抗体を送る。そして、データパススイッチを経て、次世代メモリに格納する。その際、抗体データを評価回路へも送り、再び全ての抗体の評価値を計算し(Step3)、選択回路へ送る。基本的な進化は以上の処理の繰り返しでありGA処理と呼んでいる。また、GA処理では毎世代、最優良抗体を、データパススイッチから記憶細胞(メモリ)へ格納する(Step3.1)。

予め設定しておいた世代数のGA処理が終了すると、システムは抑制処理に移る。この時、GA処理は完全に停止

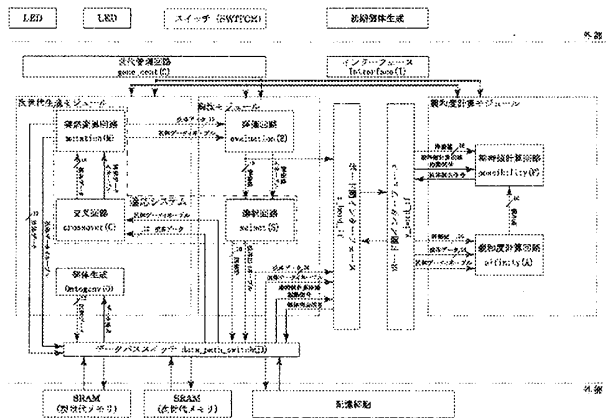


図4. 免疫プロセッサのブロック図

する。抑制処理に入ると、その世代の抗体データと評価値を、データパススイッチからボード間インターフェースを通り、親和度計算回路へ送る。親和度計算回路では、

全ての抗体の親和度を求め(Step4.1)、その値を期待値計算回路へ送る。期待値計算回路では、各抗体の親和度をもとに、それぞれの抗体の期待値を計算する(Step4.2)。期待値計算回路はデータパススイッチへ、各抗体の期待値データをもとに、消去すべき抗体に対し消去信号を送る。ここで、選択回路が再び起動し、抗体がメモリから読み出される。しかし、読み出された抗体が消去の対象になっていれば、その抗体を消滅させ、代わりに、新抗体産生回路から、新たな抗体を読み出し(Step4.3)、交叉回路へ送る。

5. パイプライン処理

今回、メモリアクセスに着目したパイプラインを構成した。図5は看護婦人数32人の場合でメモリアクセスを逐次処理で行った場合の図である。メモリアクセスに着目するとアクセスを行っていない時間が多く、無駄が多いことが分かる。そこで、メモリアクセスに着目し、パイプラインを構成した(図6)。今回、抗体データに評価値を持たせなかったことにより、突然変異回路を出たデータは分岐され、評価と書き込みを並列に行っている。

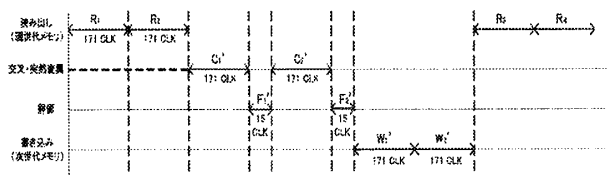


図5. 逐次処理

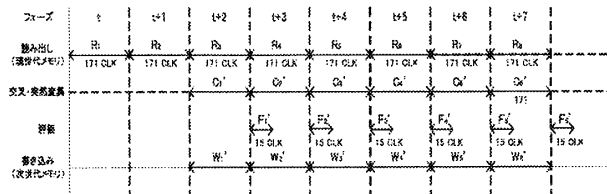


図6. メモリアクセスに着目したパイプライン

6. 評価実験

各回路のゲートサイズを表2に示す。本研究では、論理検証ツールは Cadence 社の NC-Verilog を、論理合成ツールは、Cadence 社の Synplify を使用した。また、FPGA ボードは Altera 社の APEX EP20K600E と APEX EP20K400E を、メモリは三菱電機マイコン機器社の MU200-XR を想定し、設計を行った。

ここでは、看護婦数を 32 人、世代数を 10000 に固定し、免疫における抑制処理の間隔と扱う抗体数を変えてソフトウェアとの速度比較実験を行った。ソフトウェアについては C 言語でプログラミングし、PC 上 (CPU : PentIV 2.4GHz, OS : Vine Linux, メモリ : 1GHz) で実行した。表3に、ハードウェアを動作周波数 30MHz で動作させたときの処理時間、およびソフトウェアに対する速度比を示した。抗体数を増やして頻繁に抑制処理を行うと、

表2. 各ブロックのゲートサイズ

ブロック名	Logic Resources
交叉回路	357/18640 (2.15%)
突然変異回路	1238/18640 (7.44%)
評価回路	1744/18640 (10.48%)
選択回路	948/18640 (5.7%)
データパススイッチ	1282/18640 (7.58%)
新抗体産生回路	7080/18640 (42.55%)
親和度計算回路	22586/24320 (92.67%)
期待値計算回路	513/24320 (2.11%)

表3. 処理時間と性能比(世代数:10000)

処理時間の間隔	抗体数	ハードウェア (30MHz)		ソフトウェア		ソフトウェアを基準とする速度比(倍)
		総クロック数	CPU時間(秒)	CPU時間(秒)		
200世代ごと	128	221555362	7.30	115.56	15.83	
	256	442070194	14.47	238.88	16.21	
	512	906719128	30.22	520.85	17.24	
	1024	1817602066	60.59	1237.10	20.42	
100世代ごと	128	222700362	7.42	119.48	16.10	
	256	446560634	14.89	249.50	16.76	
	512	924418778	30.81	605.00	19.64	
	1024	1883935166	62.90	1612.44	25.68	
50世代ごと	128	224991862	7.50	126.73	16.90	
	256	450541534	15.18	292.17	19.25	
	512	909818078	32.0	789.14	24.66	
	1024	2016601366	67.22	2349.88	34.96	

親和度計算のパターンが一気に増えるので、ソフトウェアでは処理時間が大幅に増加した。

一方、今回提案したハードウェアの場合、抗体数を増やしても抑制処理間隔を縮めてもあまり急激な処理時間の増加は無く、平均で 20.3 倍ソフトウェアよりも早くなり、提案アーキテクチャの有効性を確認した。

7. まとめ

本論文では、高速な免疫応答を実現する専用エンジンアーキテクチャを提案した。提案アーキテクチャでは、親和度計算の並列化や抗体産生機構におけるパイプライン処理などを実現した。また NSP を対象とした評価実験によりその有効性を確認した。

今後は、PC から各種データを設定するための GUI の作成などを行い、FPGA への実装を行う予定である。

文献

- [1] Arthur, J.L. and Ravindran, A.: A Multiple Objective Nurse Scheduling Model, AILE Transactions, Vol.13, No.1, pp.55-60, 1981.
- [2] Wei-Dong SUN, Zheng TANG, Hiroki TAMURA, and Masahiro ISHII "An Artificial Immune System Architecture and Its Applications" IEICE TRANS. FUNDAMENTALS, VOL.E86-A, NO.7 pp1858-1868, JULY 2003
- [3] D.Dasgupta, ed., Artificial Immune Systems and their Applications, Springer Verlag, 1999.