

S-system を用いた GA による遺伝子ネットワーク予測

菊地 進一^{*,**} 富永 大介^{*} 有田 正規^{*} 富田 勝^{**}

^{*} 産業技術総合研究所 生命情報科学研究センター

^{**} 慶應義塾大学 先端生命科学研究所

概要

従来法である S-system 系でタイムコースをフィッティングする目的のみに GA を利用する手法では、偶然にしか骨格構造が得られない。本論文では、その評価関数に、冗長なパラメータを削除する項を加える手法を提案する。パラメータの絶対値和の項を加えるだけの簡単な改良によって、骨格構造が従来法の 5.4 倍の収束率、具体的には 43% の割合で得られることを示す。

Genetic Networks Prediction using S-system Optimized by GA

Shinichi Kikuchi^{*,**} Daisuke Tominaga^{*} Masanori Arita^{*} Masaru Tomita^{**}

^{*} CBRC, National Institute of Advanced Industrial Science and Technology (AIST)

^{**} Institute for Advanced Biosciences, Keio University

Abstract

A conventional method to fit S-system parameters to given time-courses using GA rarely succeeds in obtaining skeletal structures. In this paper, We propose a method which add an additional term which eliminates wasteful parameters to the conventional value function. Using our simple improvement, addition the sum of the absolute values of parameters, 5.4 times better success rate, 43%, is attained in terms of obtaining skeletal structures.

1 はじめに

一般に、化学反応系、特に生体内の化学反応系や遺伝子ネットワークは、複雑な動的挙動を示す非線形システムである。このシステムの構成要素（状態変数）の定常状態値を求めたり、システムに含まれるパラメータの感度解析などを行おうとする場合、まず最初に対象となる反応系の数理モデルを作成し、そのモデルを使ってコンピュータシミュレーションを行う方法がよく用いられている。しかし、数理モデルを作成するためには、その反応系内で行われている個々の反応の詳細、つまり反応形式や速度定数などが既知でなければならない。従って、これらが未知である反応系に対して、動的解析を行おうとする場合には、まずその反応系の様々な動的挙動を観察し、そこから反応系の内部を推定していく必要がある。しかし、これには、多くの実験や労力に加え、経験的な推測を必要とする。観察された動的挙動から直接、反応速度式を求めることは一般に困難である。

このような問題に対し、任意の化学反応系について、反応形式や化学量論式が明らかでないとき、あるいは反応系での反応種の間での相互作用が明確でないときでも、S-system[1][2] という記述法を用いることで数理モデルが構築できることが知られて

いる。これは、式 (1) で表される形式の連立微分方程式であり、広く用いられている一般質量作用則 (Generalized Mass Action law, GMA) の生成過程、分解過程をそれぞれ 1 本のパスで近似したものである。

$$\frac{dX_i}{dt} = \alpha_i \prod_{j=1}^n X_j^{g_{ij}} - \beta_i \prod_{i=1}^n X_j^{h_{ij}} \quad (i = 1, 2, \dots, n) \quad (1)$$

ここで、 X_i を状態変数と呼ぶ。この形式は特に化学反応系に特化したものではないが、ここでは化学反応系を表すことを目的とする。すると、状態変数は系内の物理量（反応種の濃度、圧力、温度、pH など）に相当し、式 (1) の左辺は状態変数の時間変化（変化速度）を表す。 n は状態変数の個数、 i, j ($i, j = 1, 2, \dots, n$) は各変数の添字である。式 (1) の右辺第 1 項は X_i の増加速度、第 2 項は減少速度を表す。 α_i, β_i は反応速度定数に、 g_{ij}, h_{ij} は反応次数に相当する。

式 (1) は、反応種 X_i の生成過程と分解過程にすべての反応種 X_j ($j = 1, 2, \dots, n$) が関与していると仮定する全結線モデルである。もし、 X_i の生成過程（もしくは分解過程）に X_j が関与していない場合、 g_{ij} （もしくは h_{ij} ）の値は 0 ということになる。

これらの S-system の $2n(n+1)$ 個のパラメータ $(\alpha_i, \beta_i, g_{ij}, h_{ij})$ は、観測される状態変数 X_i のタイムコースから推測しなければならない。

我々は、このような S-system のパラメータ最適化の問題に対する探索アルゴリズムとして、遺伝的アルゴリズム (Genetic Algorithm, GA) [3] による手法を提案してきた [4][5][6]。

しかし、これまでの手法は、主にデータのフィッティングに注目していたので、骨格構造を得られる可能性は偶然にすぎず、得られる遺伝子ネットワークはブラックボックスとなっていた。そこで、本研究では、遺伝子ネットワークのパラメータ推定に際し、冗長なパラメータを削除する方法を提案する [7]。これによって、単に時系列を再現するだけでなく、骨格構造が得られ、観測時系列のみからの反応経路の予測が可能になる。

以下、本論文では、2章で遺伝的アルゴリズムによる S-system のパラメータの推定法について説明する。3章で骨格構造を得るための本手法の提案をする。4章で本手法の実験結果、5章で考察、6章でまとめを述べる。

2 GA による S-system の最適化

式 (1) で表される S-system のモデルには、非常に多数の実数パラメータ $(\alpha_i, \beta_i, g_{ij}, h_{ij})$ が含まれており、その総数は、状態変数の個数を n とすると、 $2n(n+1)$ 個である。

式 (1) から数値解法によりモデルの動的挙動を計算する際、これらのパラメータは相互に影響を与えあうため、逐次的に決定していくことはできず、すべてを同時に決定する必要がある。

また、決定すべきパラメータの個数は $O(n^2)$ であり、共役勾配法などの解析的な方法では、パラメータの個数が 10 個程度になると最適化がほぼ不可能になってくる。

さらに、この問題の場合は、目的関数は「モデルの動的挙動と実験で得られた挙動がどの程度近いか」を表すものであり、両者の間の二乗誤差などの一般的なものを採用すると、目的関数の勾配を高精度で計算するのは困難である。

本研究では、(1) 目的関数が非常に多くの局所解をもっているすなわち、異なるパラメータの組み合わせが、よく似た、またはほとんど同じ動的挙動を示すことが多い、(2) 従来の手法では、多数のパラメータを同時に決定することは難しい、という2点から GA を採用する。

特に、GA の中でも、関数最適化に際し、学習速度が向上するとされている実数 GA [8] を用いる。これは、遺伝子へのコード化として、従来のバイナリ表現や Gray 表現ではなくて、そのまま 1 つの実数値を用いる手法である。

個体の定義としては、1 つの個体で 1 つのモデルを表すようにした。すなわち、1 つの個体は、 $\alpha_i, \beta_i, g_{ij}, h_{ij}$ からなる $2n(n+1)$ 個の実数値パラメータの組で表現される。遺伝子表現には実数そのものを用いている。

従来の時系列を再現する目的の評価関数としては、次の相対的な二乗誤差和 E が用いられる。

$$E = \sum_{i=1}^n \sum_{t=1}^T \left(\frac{X_{i,cal,t} - X_{i,exp,t}}{X_{i,exp,t}} \right)^2 \quad (2)$$

ここで、 $X_{i,exp,t}$ は時刻 t における状態変数 X_i の実測値、 $X_{i,cal,t}$ は時刻 t における X_i の S-system モデルから計算される値、 n は状態変数の数、 T は実験値のサンプリングポイントの数である。GA における各個体の評価値 (適応度) は、この E の逆数 $F (= 1/E)$ として計算される。つまり、最適化ではこの F を最大化する個体を探索していくことになる。本論文で用いる評価関数の改良については、3章で述べる。

この個体に対し、以下に示すシンプル GA を適用する。

(1) 初期化

P 個の個体を生成し、個体中の各パラメータに探索範囲内の一様乱数を与える。

(2) 評価

各個体について、式 (2) に従って、適応度 F を計算する。適応度が最大の個体 (エリート個体) は別に保存しておき、以下に述べる (3) から (5) の遺伝的操作は適用しない。すなわち、エリート戦略である。エリート個体の適応度が十分に高いと判断できる場合には、その個体を最適解として最適化を終了する。また、世代交替の回数が打ち切り回数 G_{max} まで達した場合にも同様に終了する。

(3) 淘汰

個体集団中から、各個体の適応度に従った確率で 1 つの個体を選び出すという作業を $(P-1)$ 回行って、 $(P-1)$ 個の選びだされた集団と 1 つのエリート個体で個体集団を置き換える。この $(P-1)$ 個の選び方は、ランキング選択を用いる。ここで、ランキング選択とは、個体 i を選択確率

$$p_i = \frac{1}{n} \left(\eta^+ - (\eta^+ - \eta^-) \frac{i-1}{n-1} \right) \quad (3)$$

によって淘汰する戦略である。

(4) 交叉

新しい集団から等しい確率で2個体(親)を選び出し、交叉によって1つの個体(子)を生成する。例えば、子個体の g_{11} を決定するには、両親のどちらかの g_{11} の値をそのままコピーする。どちらからコピーするかは等しい確率で選ぶ。このコピー作業を子に含まれるすべてのパラメータに対して行う。子を $(P-1)$ 個生成した時点で個体集団を子の集団で置き換える。

(5) 突然変異

すべての値に対して、一定の確率 m で突然変異によって値を変更する。変異は、平均0、分散 d の正規分布乱数を加える操作によって実現される。もし値が探索範囲をこえる場合にはやり直す。

分散 d は最適化開始時は初期値 d_1 に設定されるが、エリート個体の適応度が G_1 世代の間変化しなくなったときに $d_2 (< d_1)$ に変更される。それでもエリート個体の適応度が G_2 世代の間変化しなかった場合は $d_3 (> d_1)$ に変更する。さらに G_3 世代の間変化しなかった場合はもとの d_1 に戻す。なお、適応度に向上がみられた場合は分散をもとの d_1 に戻す。

突然変異率 m は、 G_k 世代の間変化しなかった場合に $k (> 1)$ 倍する。ただし、上限となる m_{max} に達した場合は固定する。なお、適応度に向上がみられた場合はもとの m_0 に戻す。

この突然変異の作業を $(P-1)$ 個の子のすべてのパラメータに対して行う。

(6) (2) に戻る。

以上には、最適化の条件を決定すべきパラメータが数多く存在するが、これらの適切な値は最適化される関数によって異なるので、経験的に設定する。

3 GAによる経路予測

2章で述べたアルゴリズムが従来法の概要である。従来法の欠点として、時系列の再現性だけに着目しているために、複数の局所解に収束してしまう。また、それらの解の性質として、骨格構造が得られることは少なく、得られたとしても偶然であるという点が挙げられる。

そこで、本研究では、時系列の再現性だけでなく、遺伝子ネットワークの骨格構造を得ることを目的とする。これによって、未知経路予測が可能になり、実用上でも非常に有用な手法となることが期待される。

最適化アルゴリズムには2章の戦略を用いる。改良するところはただ1点で、評価関数の設定である。

本手法では、以下の誤差評価関数を用いる。

$$E = \sum_{i=1}^n \sum_{t=1}^T \left(\frac{X_{i,cal,t} - X_{i,exp,t}}{X_{i,exp,t}} \right)^2 + c \sum_{i=1}^n \sum_{j=1}^n (|g_{ij}| + |h_{ij}|) \quad (4)$$

c は2つの評価項の間のバランスをとる重み係数である。右辺第1項は従来法と同じ評価項、第2項はモデルパラメータ g_{ij}, h_{ij} の絶対値和である。すなわち、骨格構造を求めるための実装としては、評価関数にこの和を加えるのみで実現される。

ここで、この評価関数のもう意味について考える。右辺第1項は従来の時系列の再現性を表す項である。それに対して、右辺第2項はパラメータの冗長度を表す項と言える。この項は、絶対値和が小さいほど評価がよいということになるので、ほとんどのパラメータの値は0の方がよいという評価関数となっている。従来法は可能性を消さないために全結合を仮定しているが、実際のネットワークはほとんどが疎結合であり、前提条件では冗長なパラメータ表現がなされている。パラメータの値が0ということはその状態変数の与える影響がないということであるから、グラフ表現した場合の結線が切られることを意味する。この第2項はそれを数式で実現した項である。

また、正則化の概念からとらえると、式(4)の右辺第2項はパラメータの絶対値和であり、これはラプラス正則化項 [9] に相当している。同様に、二乗和を用いた項も考えることができ、これはガウシアン正則化項 [10] に対応している。厳密にはガウシアン正則化項ではパラメータ値は完全に0にはなり得ない。そこで、本研究では絶対値和であるラプラス正則化項を用いている。

ただし、本研究は、正則化項を用いた場合に骨格構造を得られるかどうかを調べるケーススタディである。問題によっては二乗和項を用いた方がよい可能性もあり、各種パラメータと同様に柔軟に選択した方がよいと考えている。

実際に用いるアルゴリズムとしては、次の2段階にわかれる。

(1) 前半として、 G_f 世代の間、式(4)を用いて最適化を行い、ある程度の学習精度を保ちながら骨格構造を得る。

(2) 後半として、 G_s 世代の間、式(2)を用いて最適化を行い、得られた骨格構造から、それをくずさない範囲で、学習精度を上げていくという戦略を取る。このとき、ある閾値 S より小さい値のパラメータは復活しないように0に固定してしまう。

表 1: S-system の動的挙動を与えるパラメータ

i	α_i	g_{i1}	g_{i2}	g_{i3}	g_{i4}	g_{i5}	β_i	h_{i1}	h_{i2}	h_{i3}	h_{i4}	h_{i5}
1	5.0	0.0	0.0	1.0	0.0	-1.0	10.0	2.0	0.0	0.0	0.0	0.0
2	10.0	2.0	0.0	0.0	0.0	0.0	10.0	0.0	2.0	0.0	0.0	0.0
3	10.0	0.0	-1.0	0.0	0.0	0.0	10.0	0.0	-1.0	2.0	0.0	0.0
4	8.0	0.0	0.0	2.0	0.0	-1.0	10.0	0.0	0.0	0.0	2.0	0.0
5	10.0	0.0	0.0	0.0	2.0	0.0	10.0	0.0	0.0	0.0	0.0	2.0

この手法と第2章で述べた戦略によって、骨格構造が得られ、さらに収束率が高く、精度のよい解が得られることを次の章で示す。

4 実験結果

以上のアルゴリズムの有効性を検証するために、ケーススタディとして、図1の比較的小規模の遺伝子ネットワーク [11] の相互作用推定の実験を行った。

これは2つの遺伝子（遺伝子1と遺伝子4）から構成される典型的な遺伝子相互作用系である。 X_1 は遺伝子1から生成される mRNA、 X_2 はそれから生成される酵素タンパク質、 X_3 は X_2 が触媒する誘導タンパク質である。 X_4 は遺伝子4から生成される mRNA、 X_5 はそれから生成される制御タンパク質である。遺伝子1の mRNA (X_1)と遺伝子4の mRNA (X_4)の合成過程に着目すると、誘導タンパク質 X_3 から正のフィードバックと、制御タンパク質から負のフィードバックがそれぞれかかっていると仮定している。

図1の系を式(1)の S-system 表記でモデル化し、表1で表される数値パラメータで表現した。

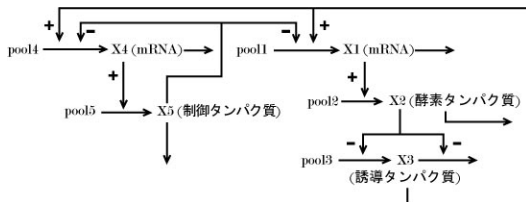


図 1: 実験で用いた遺伝子ネットワーク

実験では5変数のタイムコースを初期値を変えて5セット、計25本のタイムコースを与えた。初期濃度は、表2に示す5セットである。

与えるタイムコースの例として、Trial 1とTrial 4の2組を図2と図3に示す。他のタイムコースも表1と表2の値を用いれば計算可能である。

表 2: 実験に用いた5セットの初期濃度

Trial	X_1	X_2	X_3	X_4	X_5
1	0.70	0.12	0.14	0.16	0.18
2	0.10	0.70	0.14	0.16	0.18
3	0.10	0.12	0.70	0.16	0.18
4	0.10	0.12	0.14	0.70	0.18
5	0.10	0.12	0.14	0.16	0.70

これらのグラフをみてもわかるように、Time=0.5になると同じ濃度に収束する強安定なシステムである。収束領域を外れた場合は発散する。

この25本のタイムコースを与えて、本手法の有効性を確かめる実験を行った。ここでは、ケーススタディとして、最も重要である X_1 と X_4 の合成過程に関する相互作用を明らかにできるかを調べた。ちなみに、分解過程は自然に自己分解していくシステムであるので、推定は行っていない。すなわち、本実験で最適化するパラメータの対象は、 $\alpha_1, g_{11}, g_{12}, g_{13}, g_{14}, g_{15}, \alpha_4, g_{41}, g_{42}, g_{43}, g_{44}, g_{45}$ の12個である。本実験では、他のパラメータの値は既知とし、表1の値に固定した。

本実験の最適化の条件は、 $n = 5, T = 500, P = 100, G_{max} = 20000, \alpha_i \in [0, 20.0], g_{ij} \in [-3.0, 3.0], \eta^+ = 2.0, \eta^- = 0.0, G_1 = 10, G_2 = 5, G_3 = 5, \alpha_i$ について $d_1 = 4.0, d_2 = 0.5, d_3 = 20.0, g_{ij}$ について $d_1 = 1.2, d_2 = 0.15, d_3 = 6.0, m_0 = 0.01, G_k = 2, k = 1.01, m_{max} = 0.5, G_f = 10000, G_s = 10000, S = 0.1, c = 0, 0.0625, 0.125, 0.25, 0.5, 1.0, 2.0$ である。これらの値は試行錯誤的に決定した。また、学習中、簡略化のために、積分計算には通常通り500点を発生させるが、誤差の計算には、500点すべてではなく、10点おきの50点を取り出してフィッティングさせた。また、式(4)の右辺第2項であるパラメータの絶対値和の項に関しても、上記の g_{ij} である10個のパラメータについてのみ計算するものと

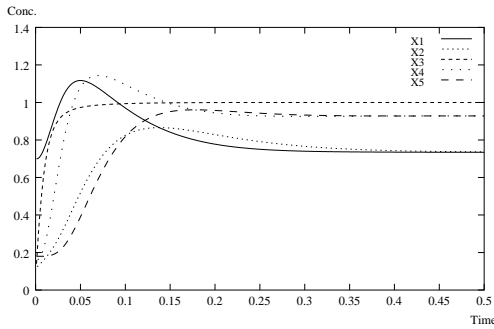


図 2: Trial 1 のタイムコース

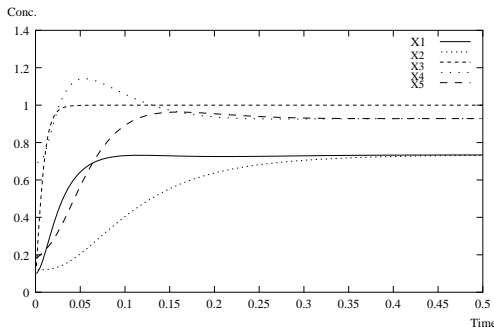


図 3: Trial 4 のタイムコース

した。

実験の結果、最もよい個体では表 1 に示す値と全く同じ値が得られた。表 3 に骨格構造へ収束した割合を示す。ここでの収束率の計算結果は、各々乱数の初期値を変えた 100 回の試行による。

収束率 A は、 $F > 8.0$ を満足する解に収束した割合である。この $F > 8.0$ という条件は、25 本のタイムコースに対して 1 点あたりの相対誤差が 1% 未満であることを示す適応度である。同じく、収束率 A+ は、 $F > 32.0$ を満足する解に収束した割合である。これは相対誤差が 0.5% 未満であることを意味する。ここでの F は式 (2) の意味での適応度である。

$F > 8.0$ を満たす解はほとんど骨格構造が得られている。ただし、一部に局所解として、 ± 0.1 や ± 0.2 程度の値をもった微少リンクが残存することがある。 $F > 32.0$ を満たす解はそのようなことはほとんどなく、骨格構造のみのリンクでモデルを表現するのに成功する水準である。すなわち、収束率 A が従来の学習成功率、収束率 A+ が骨格構造を得る意味での収束率を意味する。ここで、 $c = 0$ は右辺第 2 項を計算しないことに相当するので従来法の結果である。

表 3 の結果より、骨格構造解に収束する割合が本手法を用いることによって向上していることがわかる。また、骨格構造を目指す評価関数を与えたことによって、先見的な知識である「遺伝子ネットワー

表 3: 構造化パラメータ c による収束率の変化

c	収束率 A	収束率 A+
0	8%	0%
0.0625	32%	16%
0.125	38%	18%
0.25	43%	25%
0.5	19%	14%
1.0	3%	3%
2.0	6%	2%

クは疎結合である」という情報を与えたことになっている。これが、骨格構造を得るだけでなく、収束率を向上させたり、評価値の高い解を得たりする能力を向上させている理由である。すなわち、本手法は骨格構造を得るための手法であると同時に、収束率を向上させる手法であることがわかる。従来法と比較して、収束率 A は最適な係数 $c = 0.25$ で 5.3 倍になっている。収束率 A+ に関して比較すると、従来法ではまったく収束しなかったのに対し、本手法では $c = 0.25$ で 25% の収束率を得ている。これは骨格構造を得るという上で、有用な性質である。

表 3 からわかるように、 c の値が大きすぎるとすべてのパラメータ推定値が 0 になってしまい収束率が低下する。逆に、 c の値が小さすぎると、従来法に近づき本手法の効果が小さくなってしまい収束率が同等になる。このことから、 c の値には何らかの最適な値が存在するものと予想される。

また、本手法は、より高速に本質的な構造へと探索していく力が加わっているために、学習速度を向上させる手法ととらえることもできる。学習速度の考察に関しては、収束率の表から推測はできるが、詳しい比較実験などは稿を改めて報告したい。ちなみに、計算機環境は、産総研 CBRC の Magi Cluster を用いている。1040CPU, Pentium III 933MHz のクラスタマシンである。本手法は C 言語を用いて開発した。1 回の試行時間は 6 時間程度である。

5 考察

本実験では、2 変数のみマスクして推定を行った。その結果、本手法の有効性がわかった。ここで、本実験の結果が得られるまでの過程で、考察すべき内容について本章で述べる。

まず、与える時系列のセットが少ないとまったく収束しなかった。本実験の 5 変数の系に対して、5 変

数それぞれのタイムコースを1本ずつでは収束しなかった。これはあまりに自由度が高く収束に至らなかったと考察できる。そこで工夫として、初期濃度を変えることによって、隠れたダイナミクスは変わらない複数の(本実験では5セットの)タイムコース群を用意した。これはネットワーク予測の大規模化に際する戦略となりうる。すなわち、初期濃度を変えてなるべく他のセットと違った挙動を示す時系列を与えることによって拘束条件を増やしている。

ただし、この戦略の問題として、学習時間がその分だけ線形に増加する。また、実際のバイオ実験系への応用を考えたときに、それだけのピュアなデータを複数個も得ることは一般に困難であり、情報学的な手法の1つに終わってしまう。学習時間はさほど非現実的なものではないので、マシンパワーの問題として現在でも十分解決できる。しかし、ピュアなタイムコースを何本も必要とするという点は、実際のバイオ系への応用を考えたとき、少なくとも現時点では改良すべき問題である。よって、本手法の現時点での位置づけとしては、これだけのタイムコースが用意できれば予測可能であるという手法となっている。

また、上記の問題とも関連してくるが、マスクする状態変数の数を増やすと同様にうまくいかなかった。そしてこれは、同じようにタイムコースを増やすと解決した。詳しい実験結果は稿を改めたいが、概要を述べると、計算の難しさが増してしまい、拘束条件をより多く必要とする。

すなわち、今後の課題としては、拘束条件を追加するための条件として、タイムコースの追加以外の手法を考えることが挙げられる。具体的には、遺伝子ネットワークの先天的な知識を疎結合であること以外に何も利用していないので、そのような先天的な知識表現を組み込むことで解探索の拘束力を高める手法が考えられる。

6 おわりに

本論文では、従来法である S-system 系でタイムコースをフィッティングする目的のみに GA を利用する戦略を用いると、偶然にしか骨格構造が得られないことを指摘した。そして、その評価関数に冗長なパラメータを削除する項を加える手法を提案した。パラメータの絶対値和の項を加えるだけの簡単な改良によって、骨格構造が従来法の 5.4 倍の収束率、具体的には 43% の割合で得られることを示した。本実験はケーススタディとしての簡単なネットワーク予測であったが、与えるタイムコースを増やせばネット

ワークの予測規模を大きくすることができる可能性がある。また、先天的な知識表現との組み合わせを用いることによって、さらに規模を大きくすることが期待できる。本手法は小規模なネットワークの詳細に強い手法であり、大規模な遺伝子ネットワーク予測という点から考えると、各モジュールを担当する手法である。このような大規模化には、先に提案した AIGNET[6][12] との併用が有効であると考えている。これによって、本手法が実際の遺伝子ネットワーク予測への応用に役立つことを期待している。

参考文献

- [1] M.A. Savageau, "Biochemical system analysis : a study of function and design in molecular biology", Addison-Wesley, 1976.
- [2] E.O. Voit, "Computational analysis of biochemical systems", Cambridge University Press, 2000.
- [3] D.E. Goldberg, "Genetic Algorithms in search, optimization and machine learning", Addison-Wesley, 1989.
- [4] 富永大介, 岡本正宏, "逆問題 (inverse problem) 解決のための遺伝的アルゴリズムを用いた多次元非線形数値最適化手法の開発", 化学工業論文集, vol.25, no.2, pp.220-225, 1999.
- [5] D. Tominaga and M. Okamoto, "Nonlinear numerical optimization technique based on a genetic algorithm for inverse problems : towards the inference of genetic networks", Proceedings of the German Conference on Bioinformatics, pp.127-140, 1999.
- [6] M. Okamoto, D. Tominaga, N. Koga, Y. Maki, S. Watanabe and Y. Eguchi, "Towards a system for the inference of large scale genetic networks : efficient numerical optimization algorithm based on genetic algorithm for inverse problem", 情報処理学会 数値モデル化と問題解決シンポジウム, pp.29-37, 2000.
- [7] S. Kikuchi, D. Tominaga, M. Arita and M. Tomita, "Pathway finding from given time-courses using genetic algorithm", Genome Informatics, 2001 (to appear).
- [8] 小野 功, 山村雅幸, 喜多一, "実数値 GA とその応用", 人工知能学会誌, vol.15, no.2, pp.259-266, 2000.
- [9] P.M. Williams, "Bayesian regularization and pruning using a Laplace prior", Neural Computation, vol.7, pp.117-143, 1995.
- [10] D.J.C. Mackay, "Probable networks and plausible prediction — a review of practical Bayesian methods for supervised neural networks", Network: Computation in Neural Systems, vol.6, pp.469-505, 1995.
- [11] M.A. Savageau, "Rules for the evolution of gene circuitry", Proceedings of the Pacific Symposium on Biocomputing, pp.54-65, 1998.
- [12] Y. Maki, S. Watanabe, Y. Eguchi, D. Tominaga and M. Okamoto, "AIGNET : a system that infers large scale genetic networks", Genome Informatics, pp.253-254, 1999.