社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2007－MPS－67 (27)
2007－BIO－11 (27)
2007／12／21

Product unit based neural networksを用いた遺伝子ネットワークのS-systemモデル推定手法の提案

村田裕章† 越野亮††
三田村公智† 木村春彦†
†金沢大学 ††石川工業高等専門学校

概要 本研究ではニューラルネットワークの一種である Product Unit based Neural Network (PUNN) を用いて，遺伝子間の制御関係を表現する遺伝子ネットワークを推定する Product Unit based Neural Network モデル (PUNN モデル) を提案する．PUNN モデルでは微分方程式を解くのではなく，PUNN を学習させることで，微分方程式で記述される S-system モデルの遺伝子ネットワークの推定を行う．評価実験の結果，S-system モデルの遺伝子ネットワークを推定する従来手法に比べ，同程度の性能を保持しつつも，約 160 倍の高速化を実現できたことを示す．

Inference of S-system models of the genetic networks using the product-unit-based-neural-networks.

HIROAKI MURATA †, MAKOTO KOSHINO ††, MASATOMO MITAMURA † and HARUHIKO KIMURA†
†KANAZAWA UNIVERSITY ††ISHIKAWA NATIONAL COLLEGE OF TECHNOLOGY

Abstract In this study, we proposed the method of inference of genetic networks which expresses the regulation of genes. The proposed method, which named Product-Unit-based-Neural-Network model (PUNN model), does not solve the differential equations, learns the genetic networks using PUNN and estimates the S-system model of genetic networks which describes the differential equations. The experimental results show PUNN model is 160 times faster than the previous method which estimated S-system model of genetic networks while maintaining equivalent performance to the previous method.

1. はじめに

生体内では様々な遺伝子が互いに制御し合い，発現量を調節することで生命活動の維持が行われている．そのような制御関係は遺伝子ネットワークと呼ばれ，新薬の開発や病気の治療に応用できることから，遺伝子ネットワークの解析や推定を行う研究が行われている．特に DNA マイクロアレイ技術の進歩により，細胞全体レベルでの遺伝子発現量の解析が可能になってきており，これらの情報を用いた遺伝子ネットワーク推定に注目が集まっている．

遺伝子発現量の時系列データから，遺伝子間の制御関係（遺伝子ネットワーク）を推定することは，遺伝子ネットワーク推定問題と呼ばれ，与えられた遺伝子発現量の時系列データを再現する数理モデルを求めることが目的となる．本研究では数理モデルとして，微分方程式モデルの一種である S-system モデル [1] とニューラルネットワークモデル (NN モデル) [2] に注目する．

S-system モデルは化学反応系を記述可能な一般化質量作用則を簡略化したモデルであり，詳細な構造が明らかになっていない遺伝子ネットワークの記述に有効であると考えられる．しかしながら微分方程式を繰り返し解く必要があり膨大な計算量を必要とする．NN モデルは，微分方程式を遺伝子発現量と発現変化量の関係式と見なし，階層型ニューラルネットワーク (階層型 NN) の学習により，その近似関数を求めることで遺伝子ネットワーク推定を行う．これにより微分方程式を解く必要がなくなり，大幅な計算量削減に成功している．しかしながら求められた近似関数がブラックボックス化しているために，遺伝子間の詳細な制御関係の推定は難しく，関係性の有無および正負の三種類の関係しか特定できてていない．

そこで本研究では NN モデルにおいて階層型 NN ではなく，Product Unit based NN (PUNN) [3] を用いて遺伝子ネットワーク推定を行う PUNN モデルを提案する．PUNN では，入力を遺伝子発現量，出力を発現変化量とすることで入出力関係を S-system モデルと同じ微分方程式で定義できる．そのため PUNN モデルでは結合荷重と S-system モデルの間に明確な対応関係があり，S-system モデルと同程度の詳細な遺伝子間の制御関係を求めることができる．評価実験の結果，S-system モデルと比べ得られた最良パラメータは良くならなかったが，得られるパラメータの平均値は S-system モデルよりも良くなり，実行速度に関しては約 160 倍の速度向上を実現できたことを示す．

2. 従来研究：微分方程式モデルを用いた遺伝子ネットワーク推定方法

2.1 微分方程式モデル

微分方程式モデルを用いると，N 個の遺伝子から成る遺伝子ネットワークは式 (1) で表現される．

$$\frac{dX_i}{dt} = G_i(X_1, \cdots, X_N) \quad (i = 1, \cdots, N) \qquad (1)$$

ここで X_i は遺伝子 i の発現量，$\frac{dX_i}{dt}$ は遺伝子 i の発現変

化量（転写速度に相当）を示す．G_i は $\frac{dX_i}{dt}$ を定義する関数であり，微分方程式モデルを用いる遺伝子ネットワーク推定において，この G_i を求めることが目的となる．しかしながら一般形で表現される G_i を求めることは難しく，線形関数などの仮定を設けることで G_i を求めることが多く，微分方程式モデルにおいても様々なモデルが提案されている．本研究ではそれらのモデルの中で S-system モデル [1] と NN モデル [2] に注目する．

2.2 S-system モデルを用いた推定

S-system モデル [1] は一般的な化学反応系を記述可能な一般化質量作用則を簡略化したモデルであり，遺伝子の制御関係を正確に記述できると考えられている．S-system モデルでは遺伝子 i の発現量 X_i の生成過程と分解過程に，システムを構成しているすべての遺伝子の発現量 $X_j(j=1,2,\cdots,N)$ が関与すると仮定する全結線モデルであり，関数 G_i $(i=1,\cdots,N)$ を次の関数で表現している．

$$\frac{dX_i}{dt} = \alpha_i \prod_{j=1}^{N} X_j^{g_{i,j}} - \beta_i \prod_{j=1}^{N} X_j^{h_{i,j}} \tag{2}$$

なお $g_{i,j}$，$h_{i,j}$ はそれぞれ X_i の生成過程，分解過程に関与する X_j の相互作用係数，α_i，β_i はそれぞれ X_i の生成項，分解項に乗じる係数である．S-system モデルを用いることで，N 個の遺伝子から成る遺伝子ネットワークを 2N(N+1) 個の S-system パラメータ α_i，β_i，$g_{i,j}$，$h_{i,j}$ で表現できる．また $g_{i,j}$ と $h_{i,j}$ はキネティックオーダー（kinetic order）と呼ばれている [6]．

しかしながら遺伝子ネットワーク推定問題の問題空間は $2N(N+1)$ 次元であるため，既存の最適化手法を用いて大規模な遺伝子ネットワークを推定することは困難である．そのため問題をいくつかのサブ問題（各遺伝子に相当）に分割する方法が提案されている [4]．遺伝子 i に相当するサブ問題は次のように定式化される．

$$f_{c,i} = \sum_{t=1}^{T} \left(\frac{X^c_{cal,i,t} - X^c_{exp,i,t}}{X^c_{exp,i,t}} \right)^2 \tag{3}$$

$$\frac{dX_i}{dt} = \alpha_i \prod_{j=1}^{N} Y_j^{g_{i,j}} - \beta_i \prod_{j=1}^{N} Y_j^{h_{i,j}} \tag{4}$$

$$Y_j = \begin{cases} X^c_j & \text{if } j = i \\ \hat{X}_j & \text{if } j \neq i \end{cases} \tag{5}$$

ここで $X^c_{cal,i,t}$ は微分方程式 (4) を解くことで得られる時刻 t における遺伝子 i の発現量を示し，$\hat{X}_j$ は実験的に得られた時系列データから補間することで求める．このようにサブ問題に分割することで，$2N(N+1)$ 次元の遺伝子ネットワーク推定問題を，N 個の $2(N+1)$ 次元のサブ問題に変換できる．

S-system モデルは自由度が高いため複数の解が存在する可能性があり，また一般的に与えられる遺伝子発現量の時系列データには実験誤差が含まれるため，正確な S-system パラメータを求めることは困難である．そこで遺伝子ネットワークの結合は疎であるという特徴や，統計モデルの良さを評価する赤池情報量基準（Akaike's Information Criteria : AIC）を目的関数に導入する手法が提案されている [5),6]．

2.3 ニューラルネットワークモデル（NN モデル）を用いた推定

微分方程式モデルでは，一般に微分方程式を繰り返し解く必要があるため，多くの計算量を必要とする．そこで式 (1) を微分方程式ではなく，関数 G_i と捉えることで，関数近似を行い遺伝子ネットワーク推定を行う手法が提案されている [2]．この手法では関数近似の際に 3 層ニューラルネットワークを使用しており，ニューラルネットワークモデル（NN モデル）と呼ばれている [2]．しかしながら NN の入出力関係として関数 G_i を求めるため，ブラックボックス化されてしまい，どの遺伝子がどの遺伝子を制御しているのかを知ることは難しい．そのため感度解析を用いて結合荷重からそれらの情報を抽出することで，関係性の有無および制御の正負の情報を得ている．

3. 提案手法：Product Unit based NN モデル（PUNN モデル）を用いた推定方法

本研究では NN モデルと同様に，微分方程式を解かずに遺伝子ネットワーク推定を行いつつも，より詳細な遺伝子間の制御関係を求めることができる Product Unit based Neural Network モデル（PUNN モデル）を提案する．PUNN モデルでは詳細な制御関係の記述に S-system モデルを用い，NN の一種である Product Unit based NN (PUNN) [3] を用いることで，制御関係（S-system パラメータ）と結合荷重との間に明確な対応関係を設定できる．

3.1 Product Unit based NN（PUNN）

Product Unit based NN（PUNN）[3] は入出力関係が総乗で定義される Product Unit を用いた NN であり，中間層は Product Unit が用いられ，出力層では入出力関係が総和で定義される一般的なユニット（Summation Unit と呼ぶ）で構成されている．k 番目の中間層ユニットの出力 y^H_k および唯一の出力層ユニットの出力 y^O_1 は，それぞれ式 (6) および式 (7) となる．

$$y^H_k = \prod_{j=1}^{N} (X_j)^{w^{IH}_{j,k}} \tag{6}$$

$$y^O_1 = \sum_{k=1}^{n_H} w^{HO}_{k,1} y^H_k - \theta^O_1 \tag{7}$$

ここで入力層のユニット数を N，中間層のユニット数を n_H，出力層のユニット数を 1 とし，X_j は j 番目の入力層ユニットの入力値，$w^{IH}_{j,k}$ は j 番目の入力層ユニットと k 番目の中間層ユニット間の結合荷重，$w^{HO}_{k,1}$ は k 番目の中間層ユニットと出力層ユニット間の結合荷重，θ^O_1 は出力層ユニットの閾値パラメータを示す．

3.2 Product Unit based NN モデル（PUNN モデル）

PUNN を次の設定にすることで，遺伝子ネットワークの S-system モデルを表現する PUNN モデルを定義することができる．入力層のユニット数を遺伝子数 N，中間層のユニットを 2 個，出力層のユニットを 1 個に設定し，j 番目の入力層ユニットと 1 番目の中間層ユニット間の結合荷重 $w^{IH}_{j,1}$ を $g_{i,j}$，j 番目の入力層ユニットと 2 番目の中間層ユニット間の結合荷重 $w^{IH}_{j,2}$ を $h_{i,j}$，1 番目の中間層ユニットと出力層ユニット間の結合荷重 $w^{HO}_{1,1}$ を α_i，2 番目の中間層ユニットと出力層ユニット間の結合荷重 $w^{HO}_{2,1}$ を $-\beta_i$，出力層のユニットの閾値パラメータ $\theta^O_1 = 0$ とする．この PUNN の入出力関係は，式 (8) で表され，この式は S-system モデルにおける遺伝子 i の発現量に関する微分方程式と同じである．

入力ベクトル $\mathbf{X_t} = (X_{1,t},\cdots,X_{N,t})$ に対して $y_t(=\frac{dX_{i,t}}{dt})$ を出力する PUNN モデルの学習の際に用いる誤差

関数として一般的な式（9）ではなく式（10）を用いる．これは細かな誤差にまで注目し学習することで，より y_t に近い出力を得られるようにするためである．なお本研究ではPUNNの学習の際に一般的な最適化手法を用いることから，PUNNモデルの誤差関数を目的関数と呼ぶ．

$$y_1^o = \alpha_i \prod_{j=1}^{N} X_j^{g_{i,j}} - \beta_i \prod_{j=1}^{N} X_j^{h_{i,j}} \tag{8}$$

$$e_{P,i} = \sum_{t=1}^{T} (y_1^o(\mathbf{X_t}) - y_t)^2 \tag{9}$$

$$E'_{P,i} = \begin{cases} -\infty & \textbf{if } e_{P,i} = 0 \\ \log(e_{P,i}) & \textbf{if } 0 < e_{P,i} < 1 \\ e_{P,i} & \textbf{if } 1 \leq e_{P,i} \end{cases} \tag{10}$$

3.3 クラスタリングを用いたペナルティ項

目的関数に「遺伝子ネットワークの結合は疎である」という特徴を導入する方法の一つに，キネティックオーダー数を制限するペナルティ項を導入する方法がある．従来までは，それらの制限は固定（定数パラメータ）されていたが，本研究では動的に制限が変化するペナルティ項を導入する．基本的な考えは，キネティックオーダーを絶対値を基準に，二つのクラスに分け，絶対値が小さい方のクラスに属するキネティックオーダーを0にするようにペナルティを与えるというのもである．クラスタリングを用いてキネティックオーダーを値が0であるクラス K_{c0}，絶対値の小さなクラス K_{c1}，絶対値の大きなクラス K_{c2} に分け，K_{nz} を K_{c0} の要素数とすると，クラスタリングを用いたペナルティ項は以下のようになる．

$$P_{P,i} = \begin{cases} \infty & \textbf{if } K_{nz} < 2 \\ 0 & \textbf{if } K_{nz} = 2 \\ \sum_{x \in K_{c1}} x + (2N - K_{nz}) & \textbf{else} \end{cases} \tag{11}$$

N は遺伝子数であり，2N はキネティックオーダー数を示す．また生成過程と分解過程のバランスによって遺伝子発現量が決まるとする S-system モデルを再現するためには，非零のキネティックオーダーが2個以上必要であると考えペナルティ項の設計を行った．

3.4 動的目的関数

ペナルティ項が付加されている目的関数の問題を解く際には，しばしば本来の目的関数の項とペナルティ項にそれぞれ動的に変化する係数を乗じることで，効率的に最適化を行う方法が用いられ，本研究でもそのように動的に変化する係数を導入する．ネットワーク推定開始時刻を0として終了時刻を t_{end}，現在の時刻を t とすると，PUNNモデルの目的関数 $E_{P,i}$ は式（12）となる．

$$E_{P,i} = c_P(t) \times E'_{P,i} + (1 - c_P(t)) \times P_{P,i} \tag{12}$$

$$c_P(t) = \left(1 - (1 - \frac{t}{t_{end}})^2\right) \times 0.4 + 0.5 \tag{13}$$

4. 最適化手法

PUNNモデルではPUNNを学習することで，遺伝子ネットワーク推定を行うため，PUNNの学習精度はPUNNモデルの推定精度に大きな影響を与える．文献[7]によると，NNの学習に良く用いられる傾き降下法（Gradient Descent : GD）はPUNNの学習には不向きであると言われている．そのため遺伝的アルゴリズム（GA）やPSO（Particle Swarm Optimization ; 粒子群最適化）などの進化的計算手法を用いてPUNNの学習を行う研究がなされている[7]．そこで本研究では学習時間と学習精度において共に良い結果が得られているPSOを用いてPUNNモデルの推定を行う．

また本研究では「遺伝子ネットワークの結合は疎」という特徴を解候補の初期化時に導入することで，より効率的な初期化を行う．具体的には初期化するキネティックオーダー数 N_{Init} を 2〜N の範囲で乱数により決定する．2個以上に設定した理由は3.3節で述べた理由と同じである．また同じ遺伝子から生成過程と分解過程の影響を同時に受けるとは考え難いため，N_{Init} を 2〜N の範囲で初期化するようにした．従来研究において，ローカルサーチとしてキネティックオーダを対象としたHill Climbing Local Search（HCLS）が提案されている[5]．本研究ではこれに加え，与えられた解候補に±5％の誤差を付加した新たな解候補を n_{L1} 個生成し，与えられた解候補より評価値が更新された場合のみ，解候補を修正するアルゴリズムを導入し，対象をすべてのパラメータに拡張する．

5. 評価実験

5.1 実験の設定

PUNNモデルの性能を評価するために，従来研究でよく用いられている5つの遺伝子から成る遺伝子ネットワークを用いて評価実験を行う．実験に用いる遺伝子発現量の時系列データは，文献[4]に従って準備し，実験条件も文献[4]と同様に設定した．比較対象は出力結果が異なるNNモデルは用いずに，S-systemモデルのみとした．

またS-systemモデルとPUNNモデルではパラメータの目的関数が異なるため，両者の目的関数値を比較することに意味はない．そのため実験結果および考察の際には，目的関数値ではなくパラメータ誤差（目標パラメータとの絶対誤差）を用いる．なお粒子数は共に60個体とし，n_{L1} は20とした．実験環境はAMD Athlon(tm) 64 X2 Dual Core Processor 3800+ 2.01GHz, 1.00GB のコンピューターを用いた．

5.2 遺伝子発現データの変化量の求め方

PUNNモデルでは与えられた発現変化量を教師信号として学習し遺伝子ネットワーク推定を行うため，使用する発現変化量に含まれる誤差とPUNNモデルの推定能力に何からの関係があると考えれる．そこで本研究では以下の方法で求めた精度の異なる3つの発現変化量を用いて評価実験を行う．（1）与えられた遺伝子発現量を補間して求める方法，（2）より細かい遺伝子発現量データが得られると仮定し，差分法を用いて求める方法，（3）生物学的実験により発現変化量が得られると仮定し，その発現変化量を用いる方法の三つの方法を用いる．それぞれの発現変化量を用いたPUNNモデルをPUNNモデル（補間），PUNNモデル（差分），PUNNモデル（正確）と呼ぶ．なお評価実験の際には，（2）は遺伝子発現量データを11ポイントに間引く前の発現量データを用い，（3）は正確な遺伝子発現量データと目標S-systemパラメータを用いて発現変化量を求める．

5.3 実験結果

各モデルを用いて遺伝子ネットワークを推定する実験を30回試行し，その結果を表1に示す．実行時間は一回の試行に要した時間であり，1つのサブ問題ではなく，すべてのS-systemパラメータを推定するために要した時間を示す．評価回数とはパラメータを評価した回数であり，目的関数の計算回数を示す．誤差はパラメータ誤差を示

表 1 各モデルにおける遺伝子ネットワーク推定結果
Table 1 The results of inference of genetic networks in each model.

移動回数	モデル	実行時間 [s]	評価回数	誤差 (平均)	誤差 (標準偏差)	誤差 (最小)
500	S-system モデル	19891.73	1518420	6.2197	14.7249	0.0571
	PUNN モデル (補間)	123.43	1251817	3.2900	1.6494	1.1332
	PUNN モデル (差分量)	132.08	1249353	2.3724	2.0382	0.2476
	PUNN モデル (正確)	123.38	1253127	2.1277	1.6807	0.2361
1000	PUNN モデル (補間)	181.15	2092374	2.6717	1.5272	1.1284
	PUNN モデル (差分量)	181.66	1947153	1.5281	1.8697	0.0701
	PUNN モデル (正確)	173.98	2035241	1.5609	1.8026	0.0624
5000	PUNN モデル (補間)	667.95	9034026	1.8170	0.8323	1.1867
	PUNN モデル (差分量)	571.29	7398901	0.3022	0.8187	0.0118
	PUNN モデル (正確)	562.43	6890762	0.6055	1.3121	0.0000
10000	PUNN モデル (補間)	1317.66	18064577	1.6636	0.4853	1.1972
	PUNN モデル (差分量)	1102.35	14045943	0.2909	1.2666	0.0118
	PUNN モデル (正確)	1017.19	12804644	0.2889	0.8469	0.0000
50000	PUNN モデル (補間)	6301.80	90134959	1.5278	0.1054	1.2124
	PUNN モデル (差分量)	4727.22	60929488	0.0289	0.0317	0.0118
	PUNN モデル (正確)	4728.00	59369760	0.0190	0.0401	0.0000

表 2 従来研究において推定された遺伝子ネットワーク
Table 2 The estimated genetic networks in the previous works.

著者名 (参考)	実行環境	実行時間 [s]	誤差
Kimura[4]	Pentium 3, 1GHz	17640	2.463
Noman[5]	Pentium, 1.7GHz	18000	0.724

※各文献に記載されている最も良いパラメータの誤差を示す.

しており，30 回試行中の平均値，標準偏差，最良値を示している．なお本評価実験では実行時間の上限値を 5 時間（各サブ問題を 1 時間以内で解く）としており，移動回数 500 回以上において，S-system モデルは時間内に解くことが出来なかったため省略した．

6. 考　　察

移動回数 500 回において S-system モデルと PUNN モデルの実行時間を比べると，S-system モデルでは約 20000 秒，PUNN モデルは約 120 秒となっており，約 160 倍速いことが分かる．これは PUNN モデルではパラメータの評価時に微分方程式を解かないためだと考えられる．

パラメータ誤差について見てみると，平均値，標準偏差共に PUNN モデルのほうが小さくなっており，平均的に良いパラメータを求めていることが分かる．しかしながら最小値で比較すると，PUNN モデル（補間）よりも S-system モデルのほうが良いことが分かる．しかしながら表 2 に示した従来研究で求められたパラメータと比較すると，PUNN モデル（補間）で求められたパラメータでも十分良いパラメータと言える．

PUNN モデルにおいて，使用する発現変化量に含まれる誤差と求まるパラメータの関係について見てみると，発現変化量に含まれる誤差が少なくなるほど良くなっており，特に移動回数が 5000 回以上の PUNN モデル（正確）の場合では，誤差なく目標パラメータを求めている．PUNN モデルでは与えられた発現変化量を教師信号として学習し，教師信号に含まれる誤差までも学習してしまうために，与えられる発現変化量に含まれる誤差が少ないほど良いパラメータを求められたと考えられる．

7. おわりに

本研究では微分方程式モデルの遺伝子ネットワーク推定問題において，PUNN を用いることで微分方程式を解くことなく，S-system モデルの遺伝子ネットワークを推定できる PUNN モデルを提案した．評価実験の結果，S-system モデルと比べ得られた最良パラメータは良くならなかったが，平均値，標準偏差ともに PUNN モデルのほうが良くなっており，平均的には PUNN モデルのほうが良いパラメータを求められたことを示した．実行時間に関しては約 160 倍の速度向上に成功したことを示した．

今後の予定としては，与えられた遺伝子発現量の時系列データから，より正確な発現変化量の算出方法を検討するとともに，与えられる遺伝子発現量の時系列データに誤差が含まれる場合や，より大規模な遺伝子ネットワーク推定問題に PUNN モデルを適用し，PUNN モデルの有効性を検証したいと考えている．

参 考 文 献

1) M. A. Savageau, "Biochemical systems analysis. II The steady state solution for an n-pool system using a power-law approximation," Journal of Theoretical Biology, Vol. 25, pp. 370 – 379. (1969)
2) 木村周平, 園田克樹, 山根総一郎, 松村幸輝, 畠山眞里子, "遺伝子ネットワークのニューラルネットワークモデル同定法の提案," 情報処理学会研究報告 バイオ情報学, Vol. 2007, No. 21, pp. 79 – 84. (2007)
3) R. Durbin and D. Rumelhart, "Product units: a computationally poweful and biologically plausible extension to backpropagation networks," Neural Computation, Vol.1, No.1, pp.133 – 142. (1989)
4) S. Kimura, M. Hatakeyama and A. Konagaya, "Inference of S-system Models of Genetic Networks from Noisy Time-series Data," Chem-Bio Informatics Journal, Vol. 4, No. 1, pp.1-14. (2004)
5) N. Noman and H. Iba, "Inference of Genetic Networks using S-system: Information Criteria for Model Selection," Proc. of GECCO-2006 , pp.263-270. (2006)
6) S. Kikuchi, D. Tominaga, M. Arita, K. Takahashi and M. Tomita, "Dynamic modeling of genetic networks using genetic algorithm and S-system," Bioinformatics, Vol. 19, No. 5, pp.643 – 650. (2006)
7) A. Ismail and A. P. Engelbrecht, "Global Optimization Algorithms for Training product Unit neural networks, " IEEE Computer society, Vol. 1, pp. 132 – 137. (2000)