

《小特集》

バーチャル大腸菌の基本設計

大 竹 久 夫*・滝 口 昇*

ABSTRACT *Escherichia coli* is the best-characterized free-living organism. We here describe the first-version of Virtual *E. coli* system (VEST), a novel software package for knowledge-based simulation of the whole-cell system of *E. coli*. The VEST provides the user with a powerful tool for understanding the software of *E. coli* through knowledge-based qualitative simulation.

1. はじめに

生物化学工学 (Biochemical Engineering) という学問分野がある¹⁾。主に微生物や高等動植物の培養細胞などを用いて、医薬品や食品などの有用物質を、高い収率で生産するための技術と方法論について研究する学問分野である。この分野において、今からもう 30～40 年も以前に、酵素反応や微生物増殖の動力学モデルが盛んに研究されていたことは、あまり知られていないようだ²⁾。古典的な Monod の式はもとより、ありとあらゆる酵素反応モデルが提案され、連立微分方程式に組入れられて、コンピュータシミュレーションが行われている。当時はゲノム情報はもとより、遺伝子の構造や機能についても、詳しい情報が存在しない時代であった。それでも、遺伝子発現のフィードバック制御を考慮に入れた、微生物細胞内酵素濃度の振動現象など、先駆的なシミュレーションも数多くなされている³⁾。

しかし、遺伝子工学全盛の時代がやって来ると、酵素反応や微生物増殖の数理モデル構築と、コンピュータシミュレーションへの取組みは、時を追うように衰退していった。今振り返ってみると衰退の最大の原因は、未定パラメーターの問題であったように思われる。細胞内で生起する反応を、詳細にモデル化すればするほど、パラメーターの数が増える。パラメーターの多くは事実上測定不可能であり、たとえ精製した酵素などを用いて実測したとしても、そのまま細胞内での反応に当てはまる保証はない。いくら精巧な数理モ

デルを組立てても、パラメーターの値が決まらなければ、定量的なシミュレーションは実行できない。苦肉の策として、計算結果を実験結果に一致させようと、パラメーター値の最適探索も試みられた³⁾。しかし、その試みも結局のところ、曲線当てはめの域をでるものではなかった。最近になって、シミュレーション生物学への期待を、再び耳にするようになった。敢えて言えば、生命現象のシミュレーションとは、パンドラの箱の様なものである。不用意に箱の蓋を開けてしまうと、未定パラメーター問題という厄介な魔物が現れる。

本稿では、筆者らが取組んでいるバーチャル大腸菌 (Virtual *E. coli* System, 略称 VEST) の基本設計について述べる。筆者らは、上で述べた生物化学工学における先駆的な取組みとその苦い経験から、いきなり連立微分方程式を組立てて、定量的なシミュレーションを実行するつもりはない。他方、ゲノム関連の諸データを百科事典よろしく登録したデータベースも、既に良いものが多数ある。以下に紹介する VEST の基本設計では、大腸菌という舞台の上で、沢山の構成要素によって演じられる、生命活動劇のシナリオを描くことに力点が置かれている。沢山の構成要素が複雑に絡み合う生命現象のシミュレーションでは、しっかりとシナリオこそ、なによりも重要であると考えからである。

2. 大腸菌のシナリオ

大腸菌は、大きさ約 2～4 ミクロンの腸内細菌であり、これまで分子生物学やバイオテクノロジーの研究材料として広く利用されてきた⁴⁾。大腸菌 K-12 MG1655 株のゲノムの大きさは約 4.6 Mb (1 Mb は百万塩基対) あり、1997 年にその全塩基配列の決定がなされている⁵⁾。当初、コンピュータ解析の結果か

Basic Design of Virtual *E. coli*. By Hisao Ohtake and Noboru Takiguchi (Dept. of Molecular Biotechnology, Graduate School of Advanced Sciences of Matter, Hiroshima University).

*広島大学大学院先端物質科学研究科分子生命機能科学専攻

ら、蛋白質をコードする領域が88%, rRNA および tRNA をコードする領域が0.8%, 非コーディング繰り返し配列が0.7%, そして残りの約11%の領域が遺伝子の転写調節などに関わっていると推定された。コンピュータ解析はまた、大腸菌 K-12 MG1655 株のゲノム上には、4,288 個の遺伝子が存在すると推定している。しかし、そのうち機能がわかっている遺伝子の割合は、コードしているポリペプチドのアミノ酸配列の相同性などから推定できるものを含めても、全体の約62%に過ぎなかった。なお、大腸菌 K-12 MG1655 株のゲノムや遺伝子に関する情報は、米国ウイスコンシン大学の大腸菌ゲノムプロジェクトのホームページ (<http://www.genetics.wisc.edu:80/>) や国立遺伝研の遺伝子データベース (<http://www.ddbj.nig.ac.jp/>) などアクセスすることができる。根気さえあれば、これらのデータベースから必要な情報をダウンロードして、自分で使いやすいデータベースを作成することも容易である。

ところで、多少とも複雑に動く人工物を構築しようと思えば、ハードウェアとともにソフトウェアの設計が必要であることは、工学者なら誰でもよく承知していることだろう。生命体ほど複雑なシステムともなれば、ハードウェアとソフトウェアの両面から、設計されて然るべきであろう。ここで、筆者らが定義する生命体のソフトウェアとは、生命体がシステムとして生存し続けるために必要な、アルゴリズム(方法と手順)の総体のことである⁶⁾。生命体は、常に厳しい生存競争に晒されており、競争に打ち勝って生存し続けなければならない。生命体には、生存し続けるという先験的な目的があり、それを達成するために、長い進化の過程で築かれた、方法と手順(アルゴリズム)が備わっているはずである。もし、明確な方法と手順が備わっていなければ、生命体は世代ごとに同じ試行錯誤を繰返さなければならなくなる。生命体のハードウェアが、遺伝子を基本単位として説明されるように、生命体の振る舞いの総てを、アルゴリズムを基本単位として説明したいものである。

これから便宜上、大腸菌のソフトウェアの基本単位であるアルゴリズムを、より一般的な言葉を使って機能ルールと呼ぶことにしよう。言うまでもなく、この機能ルールの全体が、大腸菌のソフトウェアである。さて、大腸菌と言えども、多数の機能ルールが存在すると予想されるが、いくつ存在するかはまだ誰にもわからない。そこで筆者らは手始めに、大腸菌の機能ルールを、図1のように大きく3つのグループに整理して、抽出することとした。すなわち、①環境からの

ストレスの有無に関わらず、細胞が増殖したり生命活動を維持したりするのに必要なハウスキーピングルール、②様々なストレスに応答し、危機を回避するために使うストレスレスポンスルール、および③危機回避はもはや困難となり、ひたすら耐えて生き延びるために使うサバイバルルールである。

ハウスキーピングルールには、生体物質の合成や活動エネルギーを得るための、代謝などに関する機能ルールが含まれる。ハウスキーピングルールの多くは、代謝マップなどとしてよく研究されており、主要な機能ルールの抽出には文献⁷⁾などを利用することができる。ストレスレスポンスルールには、エネルギー源となる有機物の枯渇、リンや窒素などの栄養物の欠乏、pH、浸透圧や温度の変化、酸化、毒物や紫外線など有害なストレスに応答するための機能ルールが分類される。筆者らは、主に文献⁴⁾の記載を参考に、図1に示した10種類のストレスについて、主要なストレスレスポンスルールの抽出を行っている。大腸菌がこれらの機能ルールを駆使しても、ストレスによる被害を避け難くなったとき、サバイバルルールが呼び出される。大腸菌の場合、サバイバルルールの呼び出しには、緊縮応答シグナル物質である (p)ppGpp の増加や、シグマ因子の σ^{70} から σ^{38} への切り換えなどが関係すると言われている。筆者らはサバイバルルールについても文献⁴⁾の記載などを参考に抽出を試みているが、基礎研究があまり進展していないために、まだ詳細を書き上げる段階には至っていない。なお、文献を参考に機能ルールの抽出が曲がりなりにも可能であるのは、大腸菌の生物学的知見が例外的によく蓄積されているからである。同様な方法を他のモデル生物に適用することは、まだとても困難であろう。もっとも、大腸菌と言えども機能ルールの抽出には、分子生

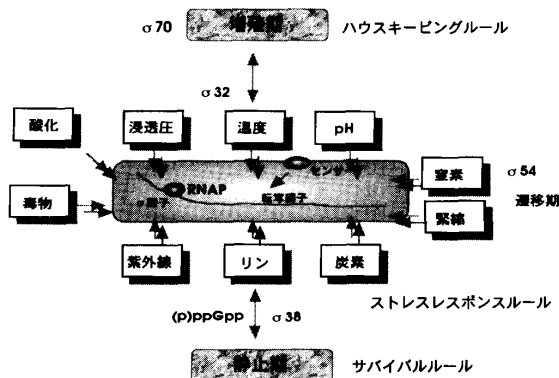


図1 大腸菌の機能ルール

物学に関する広範な知識と経験が必要であり、誰にでも容易にできる作業でもない。

筆者らは、大腸菌の機能ルールを視覚的にも理解しやすくするため、流れ図という表現方法を採用している（図2）。自作の Algorithm Draw と呼ぶプログラムは、大腸菌の機能ルールを流れ図に書き表すとともに、コンピュータに登録するために使うことができる。Algorithm Draw は、テキストベースで入力された機能ルールを、自動的にアルゴリズムの流れ図として表示するばかりでなく、各ステップで必要とされる遺伝子情報等も、流れ図の中に登録することができる。これにより、後で述べる機能ルールの実行段階におい

て、必要な遺伝子を DNA データベースから呼び出すことが可能となる。また、研究の進展により機能ルールの書換えも当然必要となろう。したがって、Algorithm Draw はテキストベースの改訂により、流れ図が簡単に書換えられるよう設計されている。

前にも述べたように、大腸菌にいくつの機能ルールが存在するかは、誰にもわからない。したがって、登録した機能ルールの数から、大腸菌シナリオの完成度を知ることは難しい。筆者らは、大腸菌ゲノム上の 4,288 個の遺伝子のうち、登録した機能ルールによって 1 度でも呼び出された遺伝子の割合を調べ、これをシナリオの完成度の目安としている。このため、Gene Array と呼ぶプログラムにより、4,288 個の遺伝子をゲノムチップよろしくモニター画面に並べ、呼び出された遺伝子の色を変えて表示するようにしている。

3. VEST の基本構成

機能ルールの多くは、互に関連し複雑に絡み合っているから、これらを統合して管理するツールが必要である。この統合管理ツールこそ、筆者らが構築中の VEST そのものである。VEST 構築の基本的考え方を図3に示す。まず、仮想環境（VEnv）に生息する大腸菌は、栄養物濃度、pH や温度などの環境情報を、分子センサーなどを使い受容する。もちろん、環境情報の受容と細胞内シグナル伝達そのものにも、多くの

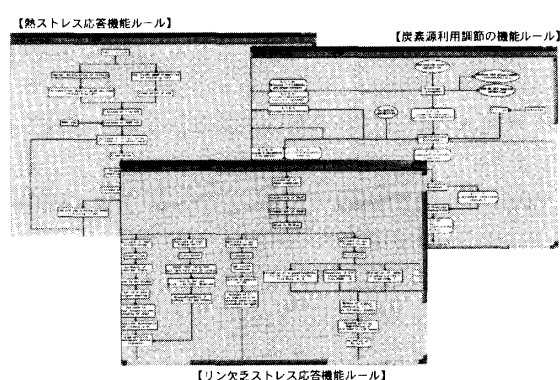


図2 大腸菌機能ルールの流れ図

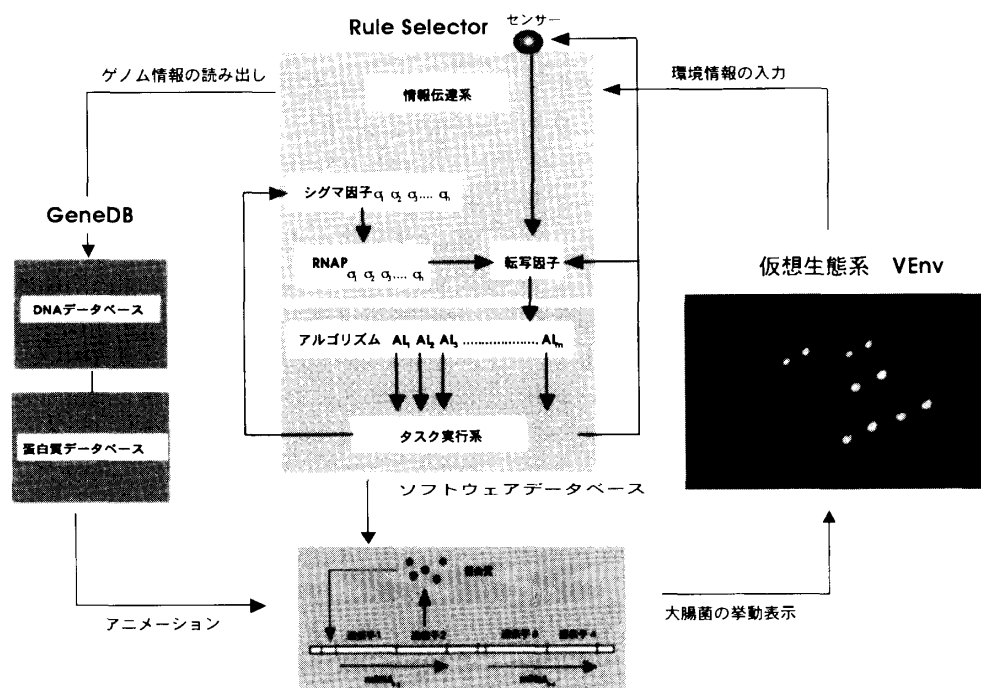


図3 VEST 構築の基本的考え方

機能ルールが存在する⁴⁾。読み込まれた環境情報は、Rule Selector と呼ばれる機能ルール選択プログラムに送られ、いま必要とする機能ルールが呼び出される。筆者らは、Rule Selector の実体として、大腸菌の遺伝子転写装置を念頭に置いている。大腸菌は環境に応答して、必要な遺伝子を選択的に発現する。ソフトウェアの視点から見れば、これは大腸菌が必要な機能ルールを選択する手続きに他ならない。したがって、Rule Selector では、RNA ポリメラーゼコア酵素 (RNAP)、主要シグマ因子、および活性化した転写因子の組み合わせにより、環境に応答して必要な機能ルールが選択される。

機能ルールは、ソフトウェアデータベースに登録されている。Rule Selector は必要な機能ルールを、このデータベースから選択する。機能ルールが選択されると、その実行段階において、DNA データベース (GeneDB) から必要な遺伝子情報が呼び出される。もちろん、大腸菌では必要な遺伝子がまず転写され、蛋白質に翻訳されて初めて、ストレス応答が表れる。しかし、VEST においては、主役はあくまで機能ルールであり、遺伝子や蛋白質はシナリオ実行のための部品に過ぎない。機能ルールが実行される様子は、モニター画面に、アニメーションとして表示される。最後に、機能ルール実行の結果は、再び仮想環境へフィードバックされ、大腸菌の行動に反映されることとなる。

4. VEST による定性的シミュレーション

VEST を使えば、大腸菌の振舞いを定性的にシミュレーションすることができる。ユーザーはまず、VestM を起動する (図 4)。VestM の画面左隅のウィンドウには、仮想環境 (VEnv) のコントロールパネルが表示される。ユーザーは、このパネルを使い、仮想環境の環境条件を任意に設定することができる。例えば、コントロールパネルから温度を選択すると、パネル下のスライダーは自動的に、大腸菌の至適温度である 37°C を示す。ユーザーがスライダーを動かして、37°C を越えた温度にセットすると、Rule Selector が温度の異常を感知して、熱ストレスに応答するための機能ルールを選択する。選択された機能ルールは、実行段階において必要な遺伝子を GeneDB から次々と呼び出す。ここで呼び出される遺伝子は Gene Array と呼ぶプログラムにより、大腸菌のゲノムチップを模したアレイ上に、緑色のスポットとしてディスプレイ表示することができる (図 5)。機能ルールが実行される様子は、画面中央に位置する上下 2 つのパネルに表示される。上側のパネルには、Task Tracer と呼ば

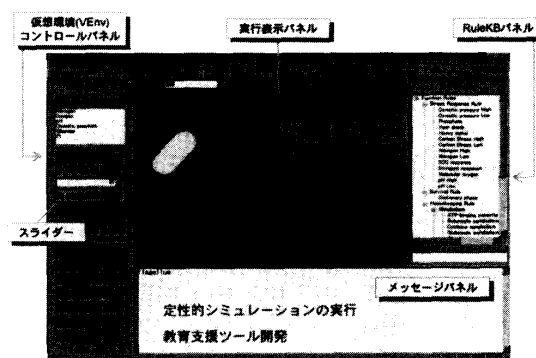


図 4 VEST のユーザーインターフェイス VestM 画面

れるプログラムにより、アルゴリズム実行の様子がアニメーションにより示される。また、下側のパネルには Algorithm Viewer と呼ばれるプログラムにより、アルゴリズムのどこが実行中であるか、表示されるようになっている。

VEST による定性的シミュレーションでは、細胞内で生起する反応が機能ルールにしたがって、アニメーションにより表示される。もちろん、反応の速度などは実際に計算されたものではなく、もっともらしくシナリオに記載されているに過ぎない。それでも、反応の因果関係を、環境との関わりの中で理解するためには、定性的シミュレーションでも十分に効果的である。また、定量的シミュレーションに先立って、定性的シミュレーションを行えば、シナリオの妥当性や完成度をチェックすることもできよう。もちろん、定性的シミュレーションには、未定パラメーター問題など存在しない。

Vest にはさらに、大腸菌に関する様々な学習支援機能が付けられている。VestM の画面右にある RuleKB パネルには、大腸菌の機能ルールがツリー状に表示されている (図 4)。ユーザーが任意の機能ルールを選択すると、Algorithm Viewer が開き、機能ルールを流れ図の形式で見ることができる。また、Algorithm Viewer の中で、流れ図の任意のステップをクリックすると Gene Array が開き、そのステップで呼び出される遺伝子情報を見ることができる。さらには、Gene Array を直接起動し、ゲノムチップを模したアレイ上の任意の遺伝子を右クリックすると、その遺伝子が破壊されたこととなり、機能ルールへの影響を調べることができる。この他、VestM のメイン画面の中央上側パネルには、大腸菌の電子顕微鏡写真等も表示されるようになっている。詳しい情報を知りたいと思えば、画像を順に選択して行けば、説明が表示され

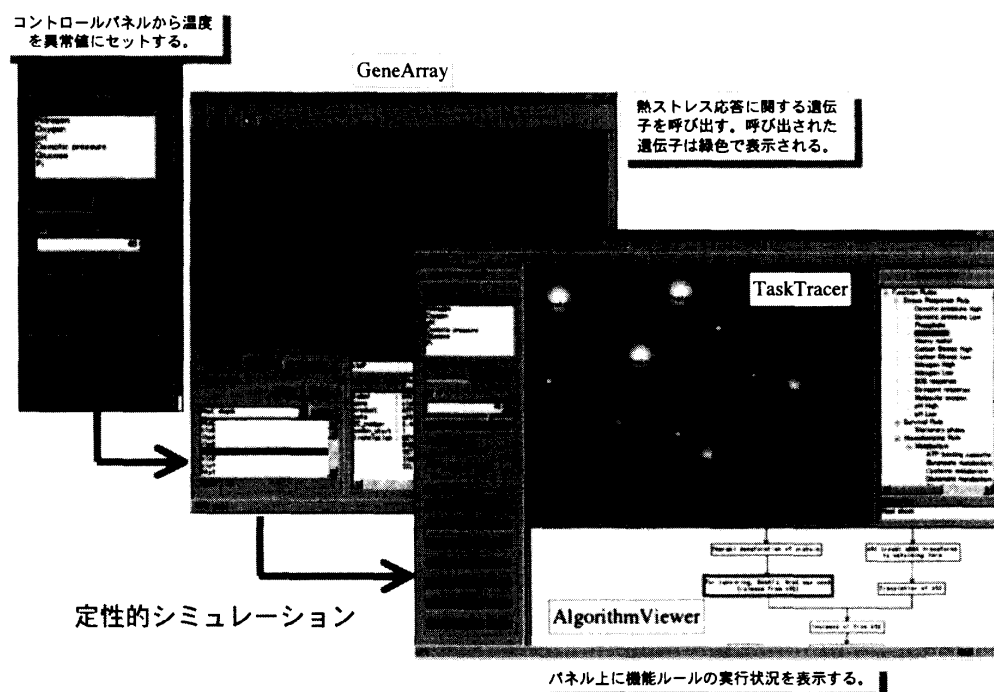


図5 熱ストレスにตอบสนองして機能ルールが実行される様子

るようになっている。

5. VESTによる定量的シミュレーション

これまで VEST は、大腸菌というシステムの振舞いを、定性的にシミュレーションすることに、重点を置いて構築されてきた。前にも述べたように、定量的なシミュレーションでは、どうしてもパラメーターの値によって、計算結果がとんでもなく影響されてしまう。この問題は、大腸菌の振舞いを詳細にシミュレーションしようとするほど、深刻なものとならざるをえない。例えば、蛋白質の細胞内濃度の変化を、連立微分方程式を用いて記述しようとする、細胞の分裂速度まで考慮せねばならなくなる。これは、極めて困難な作業であると言わざるをえない。今のところ VEST では、大腸菌の走化性⁹⁾や熱ショック応答⁴⁾など、細胞の増殖速度の影響を考慮せずとも計算可能な機能ルールに限り、定量的なシミュレーションが実行できるようになっている。ユーザーは、VestM から SimEng と呼ぶプログラムを起動させることにより、大腸菌が誘引物質に集積する様子や、熱ショックにตอบสนองして σ^{32} やシャペロンなどの細胞内濃度が、時間的に変化する様子を、定量的にシミュレーションすることができる。詳細は文献⁹⁾などに譲るが、いずれの場合にも蛋白質等の細胞内濃度が、連立微分方程式で書

き表わされており、数値解法を用いてシミュレーションが実行されるようになっている。

ところで、走化性の様に単純な機能ルールであっても、いざ連立微分方程式に書くとすると、15 個ものパラメーターが必要となる⁹⁾。誘引物質が存在しない場合に、細胞の運動方向が片寄らないことなどの拘束条件を考慮しても、まだ 12 個の独立パラメーターが残る。大腸菌の走化性では、これら独立パラメーターの多くが、精製された蛋白質を使って実測されている¹⁰⁾。しかし、実測値を用いてシミュレーションを行っても、大腸菌はなかなか誘引物質に集積してくれない。結局のところ、もっともらしいシミュレーション結果を得るには、パラメーターの値を最適探索するしかない。そのため VEST には、実測値ではなく最適探索したパラメーターの値が、ユーザーのために用意されている。

大腸菌の走化性についてはまた、強いロバスト性が認められるという報告がある¹¹⁾。このロバスト性とは、パラメーター値を大きく変動させても、大腸菌が誘引物質へ集積するという計算結果には、殆ど影響がないことを意味している。パラメーター値が変われば、細胞内における反応パターンは大きく変わるはずである。それでも結果に影響がないのであれば、細胞内反応のパターンには、計算上いくつもの解があるこ

とになってしまう。これでは、もっともらしいシミュレーション結果を得たとしても、それが真に生命現象を再現した結果であるのか、いつまでも疑問は残る。事実、SimEngにより大腸菌の走化性と熱ショック応答の定量的シミュレーションを行ってみると、もっともらしい結果を得るには、かなり作想的にパラメータの値を選ばなければならないことがわかるであろう。

たとえ多くの仮定を前提としながらも、定量的シミュレーションが必要となる場合も、もちろんあることだろう。今後、生命現象の定量的シミュレーションを行うためには、未定パラメータ問題は、どうしても避けて通ることはできない。ひょっとすると、この古くて新しい問題は、シミュレーション生物学における、最も重要な研究課題のひとつであるかもしれない。筆者らはこの問題への対応策として、次のような方法を考えている。

① とくに必要がない限り、定性的シミュレーションで済ませる。シミュレーション生物学への期待を、よくよく分析してみると、アニメーションで間に合うところはかなりある。しっかりとシナリオがあれば、コンピュータグラフィックスを駆使して、いくらでもアニメーションは提供できる。

② 細胞内で生起する反応系を可能な限りモジュール化する。例えば、解糖系、TCA サイクルや電子伝達系などのエネルギー代謝系などでは、格別の理由がない限り、個々の中間代謝産物の濃度変化まで、いちいち把握する必要はないだろう。これらの反応系は、それぞれモジュールに纏めてしまい、入出力だけをチェックするようにすればよい。多数の反応系も、少数のモジュールに纏めてしまえば、未定パラメータの数は大幅に減少する。

③ デジタル/アナログ混在回路モデルを採用する。細胞内シグナル伝達系などは、つまるところ遺伝子発現のためのスイッチに過ぎない。これらは、連立微分方程式に表現する必要はなく、ON/OFF のデジタルスイッチとして扱えば、簡略化することができる。筆者らは、広島大学の岩田穆教授との共同研究において、大規模デジタル/アナログ混在回路の設計技法¹²⁾を用いて、大腸菌のシミュレーションモデルを構築する試みを行っている。この技法を用いれば、多数の反応を比較的少数の階層的モジュールに纏めることができ、モジュールレベルで定量的なシミュレーションを実行できる。

6. おわりに

VEST は、大腸菌をシステムとして理解するための、新しいコンピュータプログラムである。生命科学には、生命体システムをハードウェアとソフトウェアに分けて、理解する習慣はない。しかし、いくらハードウェアを調べても、生命体をシステムとして理解することが難しいことは、シミュレーションに取り組んでみれば、実感としてよく理解できることだろう。VEST を使えば、大腸菌に関するあらゆる情報を、機能ルールというシナリオを軸に、コンピュータに登録して管理することができる。もちろん、大腸菌の全遺伝子の約 30% の遺伝子の機能が未だ不明であり、細胞機能もまだ完全に明らかになっているわけではないから、総ての機能ルールが VEST に登録されているわけではない。大腸菌の分子生物学の発展と歩調を合わせて、VEST もまた今後更新されて行かねばならないことは、言うまでもない。

最後に、VEST のプログラムは、VisualC++ 6.0SP3 を用いて開発されており、作動させるための標準環境としては、Windows98, Pentium 200 MHz, メモリー 64MB 以上が必要である。また現在、東京工業大学の山村雅幸助教授らにより、VEST の JAVA への移植が進められており、いずれインターネットを通して公開される予定になっている。

参 考 文 献

- 1) S. Aiba, A. E. Humphrey, N. F. Milis: Biochemical Engineering, University of Tokyo Press (1973)
- 2) 合葉修一, 永井史郎: 生物化学工学反応速度論, 科学技術社 (1975)
- 3) J. E. Bailey, D. F. Ollis: Biochemical Engineering Fundamentals, McGraw-Hill (1977)
- 4) F. C. Neidhardt et al. (eds.): *Escherichia coli and Salmonella* 2nd ed., ASM Press (1996)
- 5) F. R. Blattner et al.: Science, 277, 1453/1462 (1997)
- 6) 大竹久夫: 生き物たちのソフトウェア, 共立出版 (1998)
- 7) 日本生化学会編: 細胞機能と代謝マップ, 東京化学同人 (1997)
- 8) 黒田章夫, 加藤純一, 大竹久夫: 蛋白質核酸酵素, 41, 146/152 (1996)
- 9) 諸星知広, 辻 敏夫, 大竹久夫: 計測自動制御学会論文誌, 34, 1731/1738 (1998)
- 10) D. Bray, R. B. Bourret: Mol. Biol. Cell, 6, 1367/1380 (1995)
- 11) N. Barkai, S. Leibler: Nature, 387, 913/917 (1997)
- 12) 桜井 至: Design Wave Magazine, 123/132 (1999)