

ナ賞を受賞したラズボロフはこの分野に大きな進歩をもたらした。

4. 理論と現実とのギャップ

$P \neq NP$ 問題などは、もし $P = NP$ という形で決着するならば実際面への大きなインパクトがありうるのだけれど、私の予想は $P \neq NP$ なので、解けても実用上の意味は乏しい（しかし理論的には非常に自然で基本的な、よい問題である）。

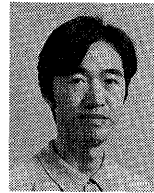
逐次アルゴリズムの研究は、実用的にも意味が明らかなオイシイところはすでに食べ尽くされた感がある。しかしノイマン型のアーキテクチャが滅びることはなく、だれかが新しい問題領域を掘り起こせばまた新しい展開がみられるであろう。

近似アルゴリズムや確率アルゴリズムについては、実用的にも重要なのに、啓蒙活動が不足しているように思う。たとえば OR の大規模な問題については、アルゴリズム論的に厳密に解こうとするよりは、分割統治法や問題の特性を利用する経験的なヒューリスティクスの活用が必要になると思われるが、そのへんの理論的整備が足りないせいか、分かりやすいよい解説書が見当たらない。理論屋だけで論文を生産していると置き去りにされそうなどころでもあり、「理論に強い実務家」の手をお借りしたいものである。

スーパー・コンピュータ向きのアルゴリズムについては、一般的な理論化はまだできておらず、個別的なノウハウが蓄積されている段階である。これは並列処理システムにもいえることであるが、実用的なアーキテクチャと理論屋が好んで使うモデルとのギャップが大きく、理論が現実に対応しにくい状況にあるように思う。

ギャップがあることは「埋める余地がある」挑戦の状況で、実用を意識する理論屋にとってはおもしろい時期にいる、ともいえる。だから実際面に明るい方も、理論をバカにせず、新しい理論を建設するのに協力されるとよい、と思う。またそのためにも理論屋は「やさしい理論を提供する」努力をすべきである。5年もたてば忘れられる「深い定理」より、皆の理解を助ける「基本的な定義」のほうが重要なのだが、と私は思う。

(平成5年5月24日受付)



野崎 昭弘 (正会員)

1959年東大理学部卒業。計算機科学の基礎理論、特に計算量の理論と多値論理を専攻しているほか、言語理論や人工知能にも関心がある。電子情報通信学会、数学会、ACM、EATCS など各会員。

人工ニューラルネットワークにおける 理論と実際のギャップ†

熊 沢 逸 夫††

1. はじめに

「いくらやってもうまくいかないよ。君の理論、どこか間違っているんじゃない？」

「どこどこ…。あれ、ここに括弧が一つ抜けているんじゃないか。自分の間違いを人のせいにしちゃいけないな。」

シミュレーションで何かを検証しようとする

とき、こうしたやりとりをよく見かけるものである。

理論ではうまくいくとされているもの、あるいは論文でうまくいくと報告されているものを、実際に自分で試してみるとうまくいかないことがある。多くの場合、その原因は自分の理解の不足やプログラムのバグにある。それでもたまたま数週間、あるいは数カ月要して、あらゆるバグを取り除いたと思っても、期待した結果が求まらないことがある。そのために参照した理論や論文を疑いたくなるようであると、客観性と再現性を第一と

† What is Most Required of Theoretical Neural Networks to be More Biological and Practical by Itsuo KUMAZAWA (Department of Computer Science, Tokyo Institute of Technology).

†† 東京工業大学情報工学科

する科学にとって深刻な事態であると言える。

この種のトラブルは新しい研究分野によくみられるものである。たとえば、常温核融合の初期のデータの真偽は依然として闇の中である。情報の分野では、人工ニューラルネットワークがこの種の問題を多く含んでいると思う。ある人がうまくいったとする手法をほかの人が追試したときに、ちょっとした条件の違いによって同じ結果が得られないことがままあるのである。近似や誤差の集積が問題となる数値計算を除けば、厳密なアルゴリズムに基づく論理的計算の世界ではこの種の問題は起こり得るはずがない。

人工ニューラルネットワークやその学習手法の動作を解析する場合には残念ながら論理は通用しないのである。そこでは大規模なシステムの構成要素が互いに複雑に絡まりあい、また依存しあって動作する。あらゆる初期条件が指定されても、実際にシステムを動作させたときに、それがどのように振る舞い、将来どのように状態を推移させていくか予測することはほとんど不可能に近いのである。現在、多くの人々は、この不明朗で不安定な手法を、不安におびえ、腫れものにもでも触れるような気持ちで使用しているのが実情ではないだろうか。

本稿では、人工ニューラルネットワークに以上のような問題点があることを念頭に置いて、この分野で理論と実際とのギャップを示す二つの話題、すなわち、

- 人工ニューラルネットワークと生体ニューラルネットワークのギャップ。
- 理論的に主張されている人工ニューラルネットワークの有用性とその実際の有用性との間のギャップ。

についてお話ししたいと思う。後々この二つのギャップが同じところに端を発しており、しかもその端というのは上に指摘した「人工ニューラルネットワークの不明朗さと不安定さ」に結び付くことが明らかになるのであるが、始めのうちは両者を分けて論じようと思う。

2. 人工ニューラルネット (理論) と生体ニューラルネット (実際) のギャップ

理論の実効性について議論する前に、理論が実際に忠実であるかどうか、つまり、人工のニュー

ラルネットワークが生体のニューラルネットワークにどれだけ近いかということについて考察しておきたい。

「ニューロンは単独で CPU 並みの高度な機能をもつ。」

これは筆者がある講演会で電総研の松本元氏から伺ったことである。ニューロンを単純な素子として扱う従来 (現在) のアプローチに慣れきっていた筆者にとってショッキングな言葉であった。しかしながら、ビデオで高速度撮影されている影響も幾分あるかもしれないが、培養液中でアメーバのように振る舞い、神経線維を伸ばしていくニューロンの姿を見たとき、この言葉の意味を実感した。

培養液の中で、ニューロンはあたかも一つの生き物のように振る舞っていたのである。生き物と機械の本質的な違いはどこにあるのであろうか。一言でいえば、生体と機械の違いは「ホメオスタシス」という言葉に集約できよう。日本語では一言でうまく言い表せないが、「恒常性」、あるいは「自律調整」という言葉がこれに当たろう。こう言ってもとても分かった気持ちになれないのでもう少し具体的に説明したい。

およそ生き物と名のつくものならば何でもよい。たとえば昆虫やバクテリアについてしてみると、これらの生き物は、その生存を助けようとする外部の特別なお膳立てがなくとも、適当な環境中に放っておけば、勝手に生きながらえ繁殖していく。生き物は外部環境の大きな変化に対応し、体温など自己の内部環境を一定に保つ能力を有しているのである。それが「ホメオスタシス」、すなわち恒常性である。

一方で、現存するどのような巧妙なロボットも、それが課せられた機能を果たすためには、細部まで厳密に指定された特殊な作業環境を必要とする。たとえば、歩行ロボットは少々地面の条件が変わると転んでしまう。産業用ロボットは、整然と組織化された生産ラインの中で規定どおりの部品が与えられたときのみ正常に働くことができるが、ほんのわずかに規定外の要素が加わると途端に機能しなくなる。この意味で機械は生き物に比べて「堅く」、融通が効かないのである。

ニューロンが一つの生き物のように見えたということは、ニューロンが単独で恒常性、あるいは

自律調整機能を有していることを意味している。現在、ほとんどすべての人工ニューラルネットワーク理論で、ニューロンは、機械的に一定の規則で入力を出力に変換する受動素子として扱われている。まさしく機械の部品としての扱ひである。もちろん人工ニューロンでも長期的には学習によってその特性は変化し得るが、自律調整という意味からは、周囲の環境に依存してもっと短時間に、そしてより多岐にわたり特性が変化することが望まれる。生体のニューロンでは、細胞膜のイオン透過性は常々変化し、自律調整されている。さまざまな化学物質がその透過性を変化させるのに関係していると言われている。ニューロンは「生きた」能動素子であり、さまざまなルートを通じて与えられる情報を用いて、自律的に自己の特性を変化させているのである。

こうしてみると、一寸の狂いもない部品が数千、数万と設計図どおりに間違いなく組み合わせられることで初めて機能し得る機械、あるいはほんのわずかなバグによって全体の機能が麻痺してしまうプログラム、こういったものになる新しい人工システムを求めて始められた人工ニューラルネットワークの研究ではあったが、現在の人工ニューラルネットワーク理論は、この目標に沿ったものになっていないように思われる。

その原因は、システムを「機械的な」受動素子としての人工ニューロンに還元しようとしている点にあると思う。ニューロンに仕様どおりの固定した特性を要求し、その規格に基づき全体を組み立て、目的の機能を実現しようとするのでは、現在の機械やプログラムの設計法となんら変わるところがない。そのような方針に従う限り、得られるものは「堅い」「機械」や「プログラム」の一種に過ぎないのではないだろうか。

生き物も、原子のレベルでは物理法則に従っているのかもしれないが、少なくともニューロンのレベルには高度な自律調整機能が存在することが必要であって、それがなくば脳のように大規模なシステムは安定に動作しえないのではないか。わずかなニューロンしかもたない線虫や単一細胞の微生物が、現在の人工ニューラルネットワークよりもはるかに生き物らしく見えることは、機械的な素子であってもそれを多数組み合わせれば生き物らしいシステムが得られるのだとする現在の研

究方針に何か誤りがあることを示唆しているように感じられる。

3. 理論的に主張されている人工ニューラルネットワークの有用性（理論）とその実際の有用性（実際）との間のギャップ

「Hopfield が論文の中で言っていることは虫がよすぎる。実際にシミュレーションしてみるとそんなにうまくいくもんじゃないよ。」

これは Hopfield の論文が出て間もないころ、視聴覚情報研究会 (AVIRG)* の討論会で、東京理科大学の上坂先生から聞いたお話である。世の中でニューラルネットワークに対する関心が高まりつつあるころであったから、大変驚いたことを記憶している。聞くところによれば、上坂先生は、Hopfield がニューラルネットワークを巡回セールスマン問題に応用した方法を、大学の学生たちに計算機実習の課題として与えてみられたそうであるが、手法中のパラメータの調整が大変で、最適解を求めることのできた学生はわずかだったそうである。

細かな話になってしまうが、この Hopfield の手法には各種のパラメータが含まれる。ニューロンの特性を表す入出力関数（シグモイド関数）の傾きと飽和の仕方を規定するゲインと呼ばれるパラメータ、減衰率を決める時定数、そして一番厄介なのは目的関数中の各項の重み付けを決めるパラメータである。巡回セールスマン問題というのは、平面上に配置した複数の都市をセールスマンが最短で一回りする順序を求める問題であるが、恐るべきことにこの最後のパラメータの値は、問題ごとに、つまり都市の配置が変わるたびに決め直さなければならない。試された方はよくご承知と思うが、Hopfield の手法はこれらのパラメータの値にきわめて敏感で、その設定に誤ると意味のない目的関数の極小点に捕まったり、目的関数の最小点が正しい解に対応しなくなったりするのである。

いくら理論的に人工ニューラルネットワークで最適化問題を解けるといっても、解くために問題ごとにパラメータの調整に苦勞するのは、人工ニューラルネットワークの実際上の有用性は少な

* 通常の講演会形式の研究会と異なり、研究者同士が本音をぶつけ合うカジュアルな形式の研究会である (1993年7月時点の連絡先は代表幹事の玉川大学工学部情報通信工学科 森見徳氏)。

くなってしまう。元の最適化問題よりも人工ニューラルネットワークのパラメータの調整（パラメータの最適化）のほうが難しいなどということになったら笑話にもならないのである。

ところでパラメータ調整の問題はシステムが大規模化するにつれて深刻になると予想される。大きなシステムで内部に含まれる膨大な数のパラメータを適切な値に設定することはとても難しいのである。小さな人工ニューラルネットワークは通常の計算機上で簡単にシミュレートできてしまい、ノイマン型計算機上の一つのソフトウェアとしての意味しかもたない。面白みがあり、また新しい可能性をもたらしてくれるのは、大規模な人工ニューラルネットワークであるのに、今の方法論ではそのような大規模なシステムを安定に動かすことなどとてもできないのである。それに対して現在のノイマン方式のソフトウェアは、一つの要素の障害が致命傷となることは確かに問題であるが、注意して構成すればいくらかでも大規模化することができ、プログラマが意図したとおりの正確に動いてくれる。

4. おわりに

生体というものは多数の要素と膨大な数のパラメータからなる想像を絶する大きさのシステムである。どうしてこのように大規模なシステムが環境の種々の変化の中で正常に機能することができるのであろうか。この謎が解けるまでは、少なくとも高い信頼性が要求される場面では、われわれは、動作を把握できる小さな人工ニューラルネットワークの範囲の応用に留まらなければならないであろう。パラメータをなんとか調整して大規模な人工ニューラルネットワークを動かしてみても、それは微妙な使用条件の変化により予想もしない振舞いをみせることになるかもしれないのである。

ニューラルネットワークの一研究者としては大変残念なことであるが、これまでに提案されてきた人工ニューラルネットワークの大規模なものは、もちろん筆者自身によるものも含めてであるが、いずれも上述の意味でいつ暴発するともされない爆弾を抱えており、安心して使えるものはほとんどない。よく使用されている誤差逆伝搬法でさえ、どのような場合に学習がうまくいき、

どのような場合にうまくいかないかということはまったく未解明であり、仮にこの学習法によって、訓練に用いたデータに対して正しく動作するシステムを形成できたとしても、未知の入力に対してそれがどう振る舞うかということは予測できないのである。

現状では、通常のソフトウェアに対抗して人工ニューラルネットワークが成果を上げているのは、確実に正確な結果が得られなくとも、ある程度良い結果が得られれば十分な分野、あるいは、得られる結果がときたま不正確となってもそれが致命傷となることのない応用分野である。たとえば、パターン認識、最適化などの分野がそれに相当する。しかしながら、これらにしても、入力や使用条件はシステムが正しく動くことのできる範囲に強く制約されている場合が多い。

理論は実際から学ばなければならないとよく言われる。生体も100%信頼できるわけではないが、それが環境や条件の変化に柔軟に対応できるきわめて安定なシステムであることは間違いのないことである。もしも人工ニューラルネットワーク研究の目的の一つが、これまでの機械にはありえなかった、安定性と柔軟性を備えた人工の機械を人類が初めて(?)手中にすることにあるのだとしたら、われわれはこれまで以上に強く生体に目を向けていかなければならない。生体から学ぶことによって、パラメータの自律調整の謎が解けたときに、初めて人工ニューラルネットワーク理論は真に実際的なものとなることができるのであろう。ニューロンを機械の部品としてではなく、それ自身「ホメオスタシス」の機能をもつ生きた細胞として捉えることはそのための第一歩なのかもしれない。

参考文献

- 1) 本稿の問題定義の種とされる Hopfield の論文は次を参照してほしい。
J. J. Hopfield and D. W. Tank: "Neural Computation of Decisions in Optimization Problems", *Biol. Cybern.*, 52, pp. 141-152 (1985).
- 2) 生体のニューロンの振舞いについては、次の文献が分かりやすい。
松本 元: "生体とゆらぎ", *ヒューマンサイエンス*, 3, 生命現象のダイナミズム, 中山書店, pp. 89-116 (1984).
- 3) 単細胞生物の巧妙な振舞いは、次の文献により実感できる。

大沢文夫：“微生物の行動”，ヒューマンサイエンス，3，生命現象のダイナミズム，中山書店，pp. 37-60 (1984).

- 4) 脳神経系の膨大なフィードバックによる調整機能は次の文献に詳しい。

R. グラニット：“目的を持つ脳”，海鳴社(1978).

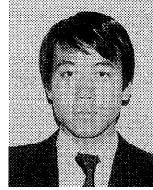
- 5) 部品化によるシステム構成法，要素化によるシステム解析法に対する批判については次の文献を参照してほしい。

アーサー・ケストラー編著：“還元主義を越えて”，工作舎 (1984).

- 6) 大脳皮質の構築，コラム間，領野間の線維連絡については次の文献を参照してほしい。

ノーマン・D. クック：“ブレイン・コード”，紀伊國屋書店 (1988).

(平成5年4月7日受付)



熊沢 逸夫 (正会員)

1959年生。1981年東京工業大学工学部電気電子工学科卒業。1986年同大学院博士課程修了。同年同大学工学部情報工学科助手。1990年同学

科助教授。パターン認識，信号画像処理，ニューラルネットワークの研究に従事。工学博士。電子情報通信学会，国際ニューラルネットワーク学会各会員。E-mail: kumazawa@cs.titech.ac.jp.

並列プログラミング†

上 田 和 紀††

正しく，しかも効率のよい並列プログラムの作成には，記述の道具であるプログラミング言語と，並列化の方法論が重要な役割を果たす。本稿では，この二つの側面において，理論と実際がどう関わっているかを，個人的経験をまじえながら考えてみたい。

1. 並列プログラミング言語とその処理系

ひと昔前まで，プログラミング言語とそれを実行する計算機アーキテクチャとの間のセマンティック・ギャップは，次第に減少するものと期待する向きが多かった。たとえば高級言語マシンの研究は，この方向を追求するものであった。

しかし，最近の RISC や並列計算機のめざましい進歩と普及は，この期待とは逆の方向をいくものであったと言える。これは，むしろ歓迎すべき方向であったと思う。なぜなら，プログラミング言語と計算機アーキテクチャそれぞれに対するわれわれの理解を深め，さらには両者をつなぐ方法論の進歩をもたらしたからである。

この動きの背景には，コンパイラ技術の近年のめざましい進歩がある。これは，理論と実際がよくかみ合っている分野といえよう。すでに逐次言

語については，人間が容易に想像できないような機械語を最適化コンパイラが出すようになった。レジスタ割付けのようなスケジューリング問題は，本来計算機が得意とする問題であったわけである。さらに，RISC や並列計算機の出現は，文や命令のスケジューリング技術をも大きく進展させた²⁾。これらのスケジューリングは，単に最適化問題を解けばよいというものではない。むしろ，最適化問題を切り出すためのプログラム解析技術がきわめて重要である。より詳細で，より大域的な解析がとりいれられるようになった結果，高度な最適化が可能になったわけである。

このように，既存のプログラミング言語を所与のものとして，自動並列化を含む最適化をしようとする試みの一方で，新たな並列プログラミング言語の設計も盛んである。これは，より簡潔で汎用性の高い理論を軸として，並列処理へのエレガントな方法論を構築しようとするものである。しかし，既存の言語のもつ巨大な文化と慣性に対抗するためには，強固な理論的基盤と実用性とを同時に示す必要がある。

筆者は長年，並行・並列プログラミング言語 GHC および KL1¹⁾の研究開発にたずさわってきた¹⁰⁾が，そこで感じたことは，言語設計やその後の発展に際して，処理系作成，記述力，理論的扱いの三者を，バランスよく考えることの重要性で

† Gaps Between Theory and Practice: Parallel Programming by Kazunori UEDA (Department of Information and Computer Science, Waseda University).

†† 早稲田大学理工学部情報学科