



Title	Hyper-Exponential Evolution of AI Systems
Author(s)	中島, 秀之; 津田, 一郎
Citation	人工知能学会論文誌, 40(3), E-084_1-E-084_4 https://doi.org/10.1527/tjsai.40-3_e-o84
Issue Date	2025-05-01
Doc URL	https://hdl.handle.net/2115/98678
Rights	ここに掲載した著作物の利用に関する注意 本著作物の著作権は人工知能学会に帰属します。本著作物は著作権者である人工知能学会の許可のもとに掲載するものです。ご利用に当たっては「著作権法」に従うことをお願いいたします。
Type	journal article
File Information	40_40-3_E-084.pdf



AIシステムの進化速度は指数関数を超えている

Hyper-Exponential Evolution of AI Systems

中島 秀之
Hideyuki Nakashima

札幌市立大学
Sapporo City University
president@scu.ac.jp, <http://www.fun.ac.jp/~nakashima/welcomej.html>

津田 一郎
Ichiro Tsuda

(同 上)
i.tsuda@scu.ac.jp, <https://www.scu.ac.jp/research/news/profile/ichiro-tsuda/>

keywords: singularity, computational complexity, exponential, AI

Summary

Ray Kurzweil argues that by 2045, the Singularity will arrive, at which point the computing power of computers will surpass that of humanity. His claim is based on Moore's Law, which states that technological advancements increase chip density, thereby shortening signal transmission distances and increasing transmission speeds, ultimately resulting in faster computations. However, this argument does not account for the programs written on these chips. It assumes that if program size remains fixed while chip density increases, computations will automatically become faster. Now, suppose the physical size of the chip remains unchanged. When the density increases, the size of the programs it can store increases. Furthermore, a program that is N times larger can handle more than N times the amount of information. If computations rely solely on local communication on the chip, the speed increases by a factor of $\sqrt{N}$, while the computational capacity becomes N times larger. If computations require global communication, on the other hand, the speed remains unchanged, aligning with Kurzweil's prediction. In the case of vector computations, such as those used in GPUs for Transformers operating on SIMD (Single Instruction, Multiple Data) architectures, computation is localized. This implies that not only does the speed increase, but the complexity of the program must also increase.

The central argument of this paper is that the overall improvement in speed would not merely be exponential, as Kurzweil claims, but rather exponential of the exponential. Although we do not explore this further, our argument should apply to a broader range of parallel computing architectures.

1. あ ら す じ

Kurzweil[Kurzweil 05] は 2045 年にはコンピュータの計算能力が人類のそれを超えるシンギュラリティがやってくると主張した。その根拠となっているのがムーアの法則である。技術の進歩によりチップの集積度（面積あたりの素子数）が上がり、それに反比例して信号の伝達距離が短くなり、伝達速度が上がるので、結果として計算が高速になるというものである。この議論はチップに書かれるプログラムには言及していないが、プログラムが固定であると仮定し、それを格納するチップが小さくなると計算が高速になるという主旨である。

ならばチップの大きさが変わらなければどうだろう？集積度が N 倍になるとプログラムの大きさは N 倍にできる。しかも N 倍の大きさのプログラムは N 倍以上の情報を扱える。計算が局所的通信のみで実行されるなら、

速度は $\sqrt{N}$ 倍^{*1}になったまま、 N 倍の大きさの計算ができる。計算が大局的通信を必要とするなら、速度は元のままである。この場合は Kurzweil の予測通りになる。ところが、Transformer に使われている GPU のベクトル計算が SIMD(Single Instruction Multiple Data) であるとするると計算は局所的であり、SIMD は、演算ユニットの独立性が高く、ユニット間の通信が少なく、メモリアクセスが局所的になりやすいという特徴を持っている。そして、Transformer の中心となる attention 機構では、離れた場所に格納されたデータ間の依存関係は存在せず、大局的通信が不要である。そうすると、速度が N 倍になるだけでなく、プログラムの複雑性も増すはずである。

したがって、全体的速度向上は Kurzweil の主張する指数関数ではなく、指数関数の指数関数になるのではないかというのが本論文の主張である。

*1 通信距離は面積の平方根で減少する。

2. 計算の複雑度

計算科学では計算の複雑度に関してさまざまな尺度が考案されている。一概に複雑度を定義はできない。出力の予測不可能性（この定義も難しい）で計るとというのが一般的である。

- コルモゴロフ複雑性

あるビット列を出力する最小のプログラムの長さによってそのビット列の複雑さを定義する。 n ビットの完全にランダムな系列はいかなるプログラムを用いてもそれ以上圧縮することはできないため複雑さは n である。それに対して、010101... と 01 列を繰り返す 2 周期列を出力するプログラムは、例えば Repeat 01 n times のような形のものになるだろう。このプログラムで最もビット数が多いのは自然数 n の 2 進表現である。すなわち $\log n$ のオーダーであり出力ビットに比べ圧縮されている。

- プログラムの長さ

最小化されていないプログラムは複雑に見えるが、さまざまな工夫を凝らして可能な限りプログラムを圧縮し、実行可能なプログラムを最小化できればコルモゴロフの複雑性と同義である。しかしながら、コルモゴロフの複雑性は結果を出すためのプログラムの最小性を問うており、現実のアルゴリズム開発にそのままでは適応できない。ある与えられた問題の複雑性クラスが P, NP などによって計算量（時間的・空間的）が異なるので、あるアルゴリズムによって複雑性クラスに応じて決まる計算量より小さく抑えることができるならば、実効的なアルゴリズムを発見したことになる。この意味で、プログラムの長さによって複雑さを計ることは現実的な意味を持つ。

- パラメータ数

2024 年現在の生成 AI の典型である GPT-4 のパラメータ数は 100 兆個と言われている。基本的には Transformer を学習させるのに必要なニューラルネットの結合数と閾値の数がパラメータ数となる。パラメータ数が多いほど生成 AI は複雑になり、より広範な関数を表現できるようになるという意味で計算能力は増大する。他方でパラメータ数が多くなると学習にかかる時間が増大したり（パラメータ数にほぼ比例？）、過学習を防ぐための別の工夫が必要になるなど、より良いハードウェアとソフトウェアが必要になってくる。すなわち、チップの集積度が上がることでより多くのパラメータ数を持つニューラルネット学習が可能になる。

しかし、ここで問題にしているのは学習速度ではなく、学習済みネットワークの計算速度である。Transformer は基本的にはベクトル演算を行っているので、GPU 上でパラメータ数には依存しない速度で計算できるものと考えられる。

知能の複雑性はこれとは異なるか？ Kurzweil が問題にしているのは知的計算能力である。次章で述べるように Kurzweil は AI プログラムの複雑性の議論を避けて、人間の脳の構造をそのままアップロードしてシミュレーションを行うという前提で計算している。つまり、脳が行っていることが何であれ、それをそのまま、しかも脳より高速にシミュレートできれば、人間の知能以上のものが実現できるという議論を行っている。

しかしながら、現状の生成 AI はそのような計算原理に基づいていないし、実際問題として生成 AI が今後どのように進化するかという方が問題である。つまり、計算方式次第では生成 AI の進化は Kurzweil の予測より速いのではないか。本論文では緻密なモデルによる議論ではなく、定性的な議論を行い、この主張を示唆するに留める。

3. ムーアの法則とシンギュラリティ

Kurzweil は 2045 年にはコンピュータの計算能力が人類のそれを超えると主張した。その根拠となっているのがムーアの法則である。技術の進歩によりチップの集積度（面積あたりの素子数）が上がり、それに反比例して信号の伝達距離が短くなり（正確には距離は面積の平方根に比例する）、伝達速度が上がるので、結果として計算が高速になるというものである。

ムーアの法則に関してはかなり以前からその加速の限界が来ると言われているが、さまざまな新しい技術^{*2}の導入により、チップの計算速度の指数関数的加速は維持されている。

Kurzweil の議論を見ておきたい。[Kurzweil 16] 第 3 章「シンギュラリティの期日を見極める」節からの引用：

2030 年代の初めには、1000 ドルで約 10^{17} cps のコンピューティングが買えるだろう（おそらく、ASIC を使い、インターネット経由で配信されているコンピューティングを取り入れると 10^{20} cps あたりになる）。今日でも、年間 10^{11} ドル（1000 億ドル）以上をコンピューティングに使っており、2030 年には控えめに見ても 10^{12} ドル（1 兆ドル）に増えるだろう。よって、2030 年代の初めには、毎年、 10^{26} から 10^{29} cps の非生物的なコンピューティングを生産していることになる。これは、おおよそ、現存しているすべての人間の生物学的知能の容量として見積もった値に等しい。

容量ではわれわれ自身の脳と同等だといっても、われわれの知能に占めるこの非生物的部分は、脳よりもさらに強力になるだろう。なぜなら、人

^{*2} たとえば 3D 化は高速化技術の候補である。
<https://xtech.nikkei.com/atcl/nxt/column/18/02957/092500002/>

間の知能がもつパターン認識能力と、機械がもつ記憶と技能を共有する能力や正確な記憶能力とが合体するからだ。非生物的部分はつねに最高の性能を発揮する。この点は、今日の生物的人間の特性と大きく異なる。現在での生物的人類文明の能力は 10^{26} cps あるとしたが、これは十分に活用されていない。

だが、この 2030 年代初めのコンピューティングの状況は、シンギュラリティではない。まだ、われわれの知能を根底から拡大するまでには至らないからだ。しかし、2040 年代の中盤には、1000 ドルで買えるコンピューティングは 10^{26} cps に到達し、1 年間に創出される知能（合計で約 10^{12} ドルのコストで）は、今日の人間のすべての知能よりも約 10 億倍も強力になる。

ここまでくると、確かに抜本的な変化が起きる。こうした理由から、シンギュラリティ—人間の能力が根底から覆り変容するとき—は、2045 年に到来するとわたしは考えている。

以上は計算速度だけの計算だが、中身のソフトウェアに関しては人間（脳だけではなく内臓なども）をスキャンしてアップロードする方式が提案されている。

4. 生成 AI の進化速度

生成 AI のコアは Transformer [Vaswani 17] である。これは確率的言語モデルで、単語を超高次元ベクトルで表現し、単語間の遷移確率の計算を基本にしたシステムである。単純化して言えば、部分的に生成された単語列から、次の単語を予測するものである。ただし、生成された単語列を同等に扱うのではなく、attention 機構を巧みに導入し、文脈依存の処理ができるようになっている。

一般に文章の処理においては、遠距離の参照などの依存関係が多発する。従来の RNN (Recurrent Neural Network) による処理ではこの依存関係の処理のために大きなデータ空間を必要としていた。これを局所的な処理で済ませるために、Transformer では文全体ではなく、必要部分だけに絞込むという attention 機構が導入された。つまり、attention の導入により、入力文全体の大域的依存関係を見ることなく、局所計算で済むというのが Transformer の長所である。attention 機構の基盤としても使い得るといふ現代版ホップフィールドネットワーク^{*3} [Ramsauer 21] は連想記憶回路であり、ネットワークの状態と記憶の内積 (SIMD 型並列計算ができる) を結合の重みとすることで、何に attention をかけるかを効率良く決めることができ、その結果記憶容量をノード数の指数関数で増加させることができる。

チップの高密度化により、この処理がどのように加速

されるかの見積もりが本論文の核心である。なお、以下の議論は大局的通信を必要としない並列計算一般に適用できるが、それが並列計算の世界でどの程度一般的なのかは本論文の域を超えるので議論しない。

計算におけるプログラムの長さはコルモゴロフの意味で計算の複雑さを与える。ここでは、計算の複雑さの観点で可能なプログラムの数を計算する。つまり、より大きな（複雑な）プログラムが従来と変わらぬ速度で計算できるならば、それは計算速度が加速したのと同じ効果をもたらす。

また、チップにはプログラムだけではなくそのデータとなるパラメータも格納されるわけだから、パラメータ数の増大という意味での複雑度の増加も起こる。

チップの集積度が N 倍になったとしよう。集積度に伴い論理回路、記憶回路もともに N 倍に集積される。論理演算用にもともと L ビットの計算容量があったとして NL ビットに増加すると、可能な計算量 C は

$$C(N) = \sum_k \binom{NL}{k} = (1+1)^{NL} = 2^{NL}$$

と推定され、 $d \equiv 2^L$ において、 $C(N) = d^N$ となり、指数関数で増加する。計算の複雑さを対数で評価すると、 $\log 2^{NL} / \log 2^L = N$ 倍になる。記憶回路についても L ビットの空間が NL ビットの空間に増加したと考えると、同様に kd^N (k は定数) で増加する。

このようにチップの集積度が N 倍になると、可能な計算量は 2^N に比例した増加率、すなわち N の指数関数で増加すると推定される。ムーアの法則はチップの集積度が時間に関する指数関数で増加するというものなので、 $N \propto 2^t$ (t は時間。ムーアの法則の場合は年単位) と表される。したがって、チップの集積度が N 倍になると、可能な計算量は $2^{(2^t)}$ に比例して増加する。すなわち、時間に関する指数関数の指数関数で増加すると推定される。

生成 AI の進化に必要なプログラムの複雑さがここで示した単純な計算量そのものであるとは言えないだろうが、上記の方向で加速する可能性はあると思う。少なくとも Transformer は各局所のブロックを attention による（連想記憶によって）非連続的に処理できるという観点で、計算の局所性と計算の複雑性の増加が可能になっている。

5. ま と め

Kurzweil の議論は計算速度の向上だけを扱っており、計算の中身の複雑性には言及していない。しかし、「知能」を問題とするからには複雑性に注目すべきである。複雑な計算が可能になればそれは進化と言える。

ここでは計算量に着目し、それが指数関数的に増えることを指摘した。速度の指数関数的増加と併せると、指数関数の指数関数となる。議論の要点は、集積度の上がった

*3 2024 年度ノーベル物理学賞

チップを元の大きさで使用することによって、処理できるプログラムの複雑度を上げることである。この場合、チップの演算速度の高速化を持つためには計算が局所的でなければならない。生成 AI に用いられている Transformer の attention 機構はその条件を満たしているので、集積度の向上 × プログラムの複雑度の向上、によって上記の結論が導ける。

なお、一般の並列計算も、計算が局所的であれば同じ議論が適用できるはずである。

謝 辞

匿名の査読者からは有益なコメントをいただいたので、特に記しておく。

◇ 参 考 文 献 ◇

- [Kurzweil 05] Kurzweil, R.: *The Singularity Is Near: When Humans Transcend Biology*, The Viking Press (2005)
- [Kurzweil 16] Kurzweil, R. 著, (小野木明恵, 野中香方子, 福田実共訳): シンギュラリティは近い [エッセンス版]—人類が³生命を超越するとき, NHK 出版 (Kindle) (2016)
- [Ramsauer 21] Ramsauer, H., *et.al.* : Hopfield Networks is All You Need, arXiv:2008.02217v3 [cs.NE] (2021)
- [Vaswani 17] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I.: Attention Is All You Need, arXiv:1706.03762[cs.CL] (2017)

〔担当委員：我妻 広明〕

2024 年 8 月 14 日 受理

—— 著 者 紹 介 ——



中島 秀之(正会員)

札幌市立大学学長。1971 年東京大学入学。1978 年 MIT 留学（1 年間）。1983 年東京大学大学院情報工学専門課程修了（工学博士）。同年、電子技術総合研究所入所。2001 年産業技術総合研究所サイバーアシスト研究センター長。2004 年公立ほこだて未来大学学長。2016 年東京大学先端人工知能学教育寄付講座特任教授、2018 年より現職。著書『Prolog』（産業図書、1983）, 『知能の物語』（ほこだて未来大学出版会、2015）他。



津田 一郎(正会員)

札幌市立大学 AIT センター特任教授。九州工業大学情報工学部助教授、北海道大学大学院理学研究科数学専攻教授、中部大学創発学院教授、院長などを歴任後 2024 年 4 月より現職。北海道大学名誉教授、中部大学名誉教授。専門：複雑系脳科学、カオス力学系などを含む数理科学（応用数学）