

発表論文：

Mankowitz, D.J., Michi, A., Zhernov, A. *et al.*

Faster sorting algorithms discovered using deep reinforcement learning.

Nature **618**, 257–263 (2023).

<https://doi.org/10.1038/s41586-023-06004-9>

研究目的・概要

ソートアルゴリズムは、データの集合を昇順や降順などの一定の規則に従って整列させる、最も基礎的なアルゴリズムの一つである。ソートアルゴリズムを含む基盤的なアルゴリズムの多くは何十年にもわたり改良が重ねられ、人によるさらなる最適化が見込めない段階まで達していると考えられている。

そこで本研究では、ゼロから効率的なソートアルゴリズムを構築する深層強化学習エージェントである AlphaDev を開発した。一般に、Python などのプログラミング言語で書かれたプログラムを実行する際、アルゴリズムの内容はアセンブリ言語と呼ばれる、コンピューターが理解できる機械語と 1 対 1 に対応した言語にコンパイルされる。筆者たちはアセンブリ言語レベルでなら、ソートアルゴリズムに最適化の余地が残されていると仮説を立て、「アルゴリズムの最適化」という課題を **AssemblyGame** と呼ばれる 1 プレイヤー用のゲームに落とし込んだ (図 1)。

AssemblyGame ではプレイヤーがアセンブリ言語で書かれた指令をターンごとに書き加えながらソートアルゴリズムを構築し、アルゴリズムの精度と計算時間に応じて報酬を受け取る。AlphaDev は深層学習、強化学習、モンテカルロ木探索の 3 つの手法の組み合わせた手法を用いて累積報酬和の期待値が最大となるような方策を学習する。

AlphaDev を用いて固定の要素数を含むリストを整列させる **Fixed Sort** と、一定の要素数以下のリストを整列させる **Variable Sort** の 2 つのアルゴリズムの改良に成功した。AlphaDev が構築した **Sort 5** アルゴリズム (5 つの要素を含むリストの整列) は、既存のアルゴリズムに比べて計算時間が 70%早いことが示された。また、AlphaDev が構築した **Variable Sort 5** アルゴリズムは、既存のものに比べて全く違うアプローチが取られていた。(図 2)

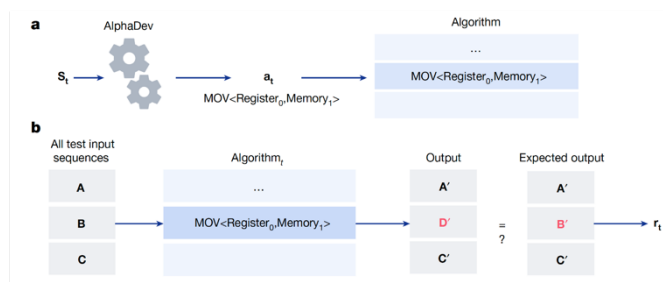


図 1

先行研究と比べて何がすごい？技術やアプローチのキモはどこ？：

ゼロからソートアルゴリズムを構築することができる深層強化学習モデルを開発した。Transformer encoderを含む深層学習ネットワークを用いてアセンブリ言語で書かれたアルゴリズムを数値ベクトルに変換し、ソートアルゴリズムの最適化を一プレイヤーゲームの枠組みに落とし込んだのがアプローチのキモである。AlphaDevの学習に使われた強化学習、深層学習、モンテカルロ木探索の3つを組み合わせた学習手法はチェスや囲碁に特化したAlphaZeroの学習手法の拡張であり、目新しいものではない。

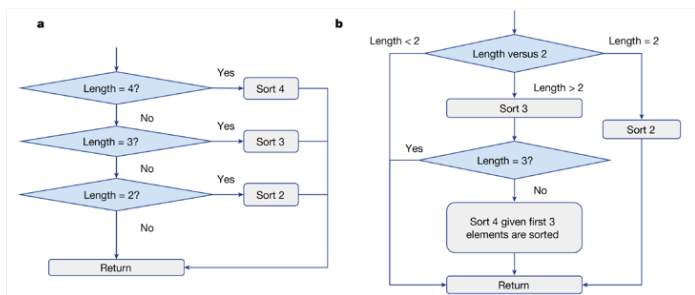


図 2

どのようにこの手法・仮説の有効性を検証したのか？

高精度かつ計算時間が短い Fixed Sort アルゴリズムと Variable Sort アルゴリズムを構築するように AlphaDev を訓練し、実際に構築されたアルゴリズムを 100 のマシン上で実行することによってその精度と計算時間を測り、既存のアルゴリズムより優れているかを検証した。

その他、議論した内容 (ネガティブコメントや limitation もあれば)：

- C++などの高水準言語で書かれたアルゴリズムではなく、よりマシン語に近いアセンブリ言語で書かれたアルゴリズムを強化学習を用いて直に改良したのは印象的である。
- マージソートなどの一般的に使用されているソートアルゴリズムは、大きなリストを多数の小さなリストに分割し、それら整列し統合することで全体のリストの整列を行う。本研究で Fixed Sort 5 や VarSort 4 など、要素数が少ないリストの整列を目的としたアルゴリズムの効率化が行われたことは、マージソートなどのさらなる高速化に繋がる。
- AlphaDev の開発により Fixed Sort 3 アルゴリズムのさらなる改良化に成功したが、内容は無駄な指令を 1 行分割したのみである。もし、全く新しいアルゴリズムが発見されていたらよりインパクトが大きかった。

この研究をさらに発展させるとしたら：

- ソートアルゴリズムのような、何十年にもわたり改良が続けられてきたアルゴリズムにもまだ改良の余地が残されているのなら、他の一般的なアルゴリズムもまだまだ最適化が可能であることが推測される。この研究をさらに発展させるとしたら、AlphaDev を用いて文字列探索などの他のアルゴリズムの改良が可能かどうかを試してみたい。