

# マルチ GPU 環境における 機械学習ハイパーパラメータの自動チューニング(1)

多部田敏樹<sup>†</sup> 藤家空太郎<sup>†</sup> 藤井昭宏<sup>†</sup> 田中輝雄<sup>†</sup> 加藤由花<sup>††</sup> 大島聡史<sup>†††</sup> 片桐孝洋<sup>†††</sup>

工学院大学<sup>†</sup> 東京女子大学<sup>††</sup> 名古屋大学<sup>†††</sup>

## 1. はじめに

本研究は自動チューニング(AT)の研究である。プログラムの実行性能の要因をパラメータ化し、適切なパラメータを推定して性能向上を図る。しかし複数のパラメータを同時に推定するには時間がかかる。パラメータの増え方が僅かでも組み合わせの増え方が急激で、探索する範囲が急増するためである[1]。

我々は、複数のパラメータを同時に推定する手法として、多次元のパラメータ空間における反復次元探索[2]のAT機構[3]を提案している。この手法はパラメータ設定と実行性能の実測を繰り返して、自動的に推定を行う。

一方ロボット制御分野では、歩行者経路予測アプリケーションに機械学習を利用した研究が報告されている[4]。機械学習プログラムには、実行性能に影響を及ぼすハイパーパラメータが複数存在する。

本研究は機械学習プログラムにATを適用し、ハイパーパラメータの組み合わせを推定する。実測1回に30分を要する機械学習プログラムは、我々のAT機構でも推定を終えるに数日かかる。そこでマルチ GPU 環境を利用し、推定に必要な実測を並列化することで、推定時間を削減する。

## 2. ATの機械学習プログラムへの組み込み

### 2.1 AT機構と機械学習プログラム

文献[4]における研究では、歩行者の移動経路を予測する予測器を、機械学習により構築する。図1にATと機械学習の位置付けを示す。AT機構は機械学習プログラムに対して、パラメータを設定、実行性能データを取得する。実行性能はパラメータを設定するたびに実測する。ここで本研究はマルチ GPU 環境を利用することにより複数回の実測を並列化する。

Auto-Tuning of Hyperparameters of Machine Learning on Multi GPU Environment (1)

Toshiki Tabet<sup>†</sup>, Sorataro Fujika<sup>†</sup>, Akihiro Fujii<sup>†</sup>, Teruo Tanaka<sup>†</sup>, Yuka Kato<sup>††</sup>, Satoshi Ohshima<sup>†††</sup>, Takahiro Katagiri<sup>†††</sup>

<sup>†</sup>Kogakuin University, <sup>††</sup>Tokyo Woman's Christian University, <sup>†††</sup>Nagoya University

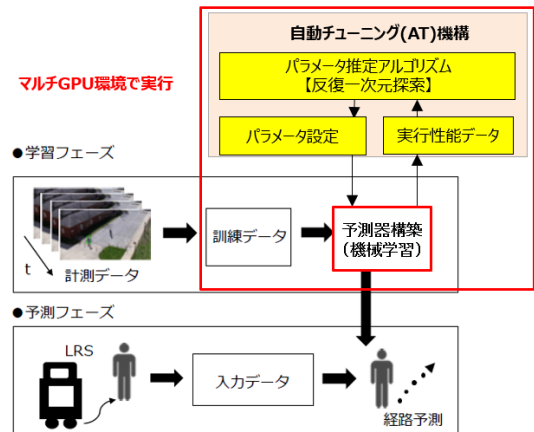


図1 ATと機械学習の位置付け (文献[4]より)

### 2.2 パラメータ推定手法「反復次元探索」

各パラメータのパターンを、離散点からなるパラメータ空間とする。反復次元探索とは、方向探索と次元探索を繰り返す手法である[2]。探索する方向には、各パラメータの軸と複数のパラメータを組み合わせた斜め方向からなる。方向探索を初めは各パラメータ軸のみとし、徐々に斜め方向を追加して範囲を広げることで、探索した点を活用しながら探索を進める。

従来は逐次的に1点ずつ探索する(従来手法)。本研究は提案手法として複数の探索点を並列に探索する。この際に、方向探索と次元探索をまとめて並列化する。反復次元探索における並列化対象を図2に示す。

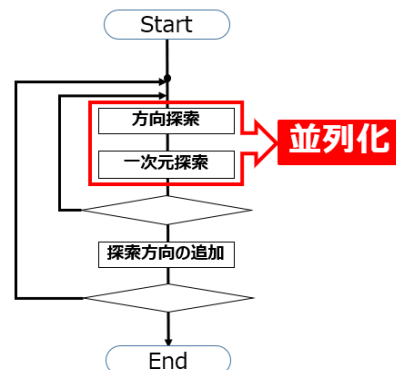


図2 反復次元探索における並列化対象

### 3. 機械学習ハイパーパラメータ推定

歩行者経路予測アプリケーションで利用する機械学習プログラム[4]に、提案手法を適用する。この機械学習プログラムには、5 パターンの値を取り得るハイパーパラメータ 6 種類 (表 1) が存在し、15625 パターンの組み合わせが存在する。実際の歩行者の経路と予測器が予測した経路の誤差を評価指標値として、評価指標値が小さいハイパーパラメータの組み合わせを推定する。

名古屋大学のスーパーコンピュータ「不老」Type II サブシステム[5]を実行環境として用いる。リソースグループは cx-share とした。最大 50 のジョブを同時に実行でき、1 ジョブで 1GPU 使う。1 ジョブで学習 1 回分を割り当てる。

従来手法と提案手法の推定結果を表 1 に示す。推定を終えるまで従来手法は約 279 時間(11.6 日)、提案手法は約 12.3 時間かかり約 23 倍高速化した。探索回数は従来手法で 553 回、提案手法で 761 回となった。提案手法は探索方向すべてを一度に探索するため、探索回数が増加したと思われる。評価指標値について推定前は 12.1 となっていた。今回の評価で従来手法は 0.89、提案手法は 0.81 に削減した。

図 3 に提案手法の並列度を示す。提案手法は、機械学習プログラムを合計 761 回実測している (赤線の区切りでそれぞれ並列化されている)。探索 102 回目までは、各パラメータ軸方向のみで探索しており探索点が 4 回移動した。探索範囲が小さいため並列度が低い。探索 102 回目以降は、複数のパラメータを組み合わせた斜め方向にも拡大し、探索点が移動しなかった。探索範囲が大きくなったため 50 以上の並列度を抽出した。実稼働の並列度は最大 50 なので、各ノード上で並列化単位ごとに 1~5 回、全体は 20 回分相当が実行される。

今回の評価では各パラメータ軸の範囲のみで十分に探索が行われていた。探索範囲を初めは抑えておくことによって、探索に必要な実測を減らすことができる。

表 1 従来手法と提案手法の推定結果

|               | 従来手法   | 提案手法    | 推定前    |
|---------------|--------|---------|--------|
| 所要時間          | 279 時間 | 12.3 時間 |        |
| 探索回数          | 553 回  | 761 回   |        |
| 評価指標値         | 0.89   | 0.81    | 12.1   |
| Rnn size      | 512    | 512     | 128    |
| Grad clip     | 9      | 10      | 10     |
| Learning rate | 0.003  | 0.01    | 0.003  |
| Decay rate    | 0.9    | 0.95    | 0.95   |
| Dropout       | 0.3    | 0.3     | 0.5    |
| Lambda param  | 0.0002 | 0.0001  | 0.0005 |

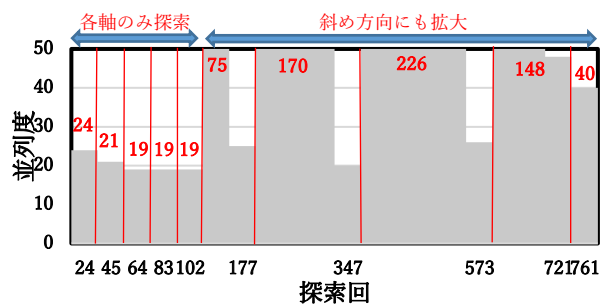


図 3 提案手法の並列度

### 4. おわりに

本研究では従来の反復次元探索を改良し、マルチ GPU 環境を利用して、機械学習ハイパーパラメータ推定に適用した。従来手法に比べて提案手法は、所要時間を 279 時間(11.6 日)から 12.3 時間まで削減した。

今後の課題として、実行する並列化環境ごとのアルゴリズムの柔軟な最適化が挙げられる。今回は「不老」Type II サブシステムのマルチ GPU 環境に合わせて AT 機構を適用したが、他の計算機環境においても検証する必要がある。

### 謝辞

本研究の一部は JSPS 科研費 JP18K19782, JP18K11340 の助成および名古屋大学 HPC 計算科学連携研究プロジェクトの支援による。

### 参考文献

- [1] T. Katagiri, Auto-tuning for The Era of Relatively High Bandwidth Memory Architectures: A Discussion Based on an FDM Application, Proceedings of IEEE IPDPSW2018.
- [2] 望月大義, 藤井昭宏, 田中輝雄, “ソフトウェア自動チューニングにおける複数同時性能パラメータ探索手法の提案と評価”, 情報処理学会論文誌コンピューティングシステム, Vol.11, No.2, pp.1-16.
- [3] Toshiki Tabeta, Naoto Seki, Akihiro Fujii, Teruo Tanaka, Hiroyuki Takizawa, “An Optimization Technology of Software Auto-Tuning Applied to Machine Learning Software”, Poster Session on HPCAsia2020.
- [4] Rina Akabane, Yuka Kato, “Pedestrian Trajectory Prediction Using Pre-trained Machine Learning Model for Human-Following Mobile Robot”, 2020 IEEE International Conference on Big Data.
- [5] <http://www.icts.nagoya-u.ac.jp/ja/sc/overview.html#type2> (2021/1/8).