

# 不老上での自動チューニング機構の並列と 機械学習プログラムに適用時の安定性の検証



藤家 空太郎, 藤井 昭宏, 田中 輝雄(工学院大学)

加藤 由花(東京女子大学)

大島 聡史(名古屋大学)

# 研究背景

- 自動チューニングの研究

- ▶ 性能パラメータを自動で最適化

- 性能パラメータ：プログラムの性能（消費電力，実行時間等）に影響を及ぼすパラメータ

- ▶ 対象プログラムの実測(1回)と実測値の評価の繰り返し

- 対象プログラムが1回の実測に時間がかかる場合，  
チューニングには膨大な時間が必要

- ▶ 本研究の対象の機械学習プログラムの場合

- 1回の実測：30分（簡易）
    - 推定完了：12日

# 研究目的

## 1. 自動チューニング機構の推定に必要な時間を削減

- ▶ 推定の際に必要な実測を並列化
- ▶ 「不老」Type II サブシステムの活用

## 2. 自動チューニング機構で求めた結果を安定させる

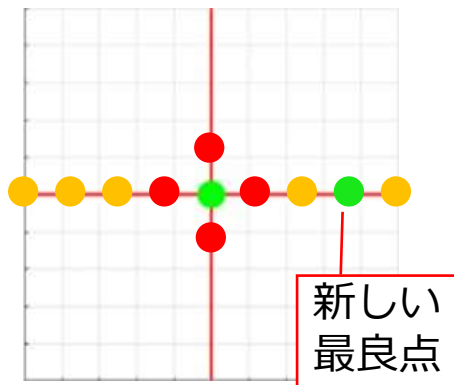
- ▶ 推定した性能パラメータを複数回追加で実測
- ▶ 実測結果の分布や平均を調べる

## 反復次元探索

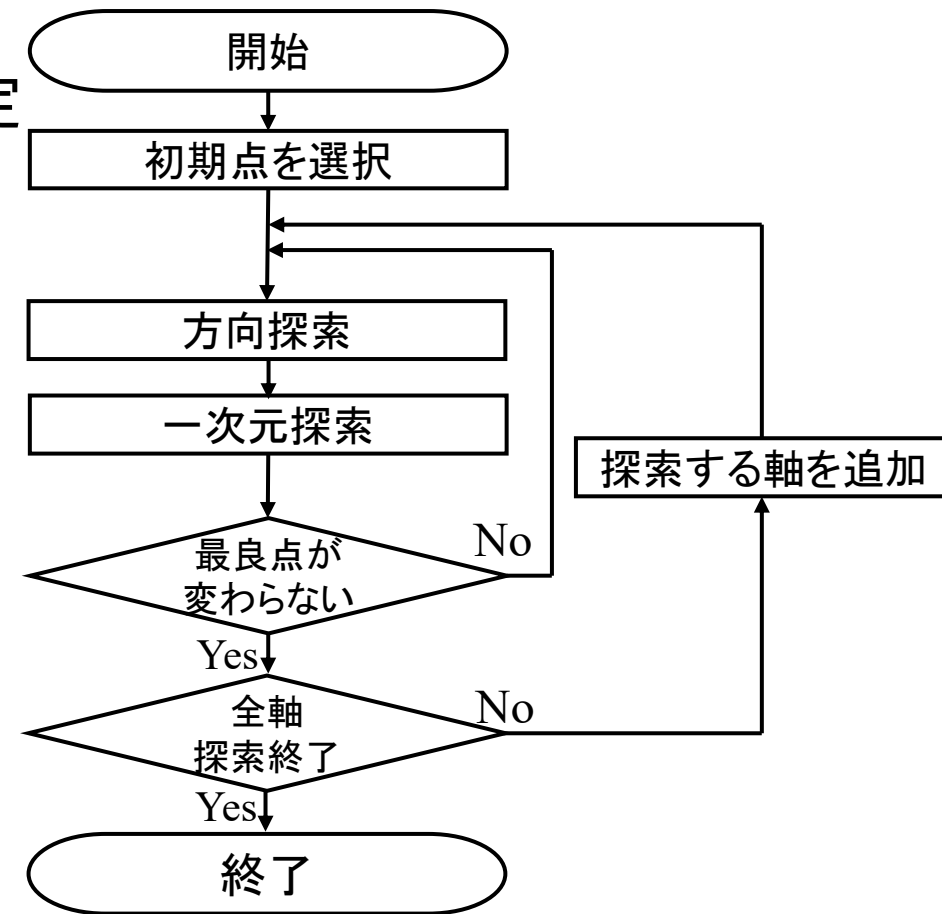
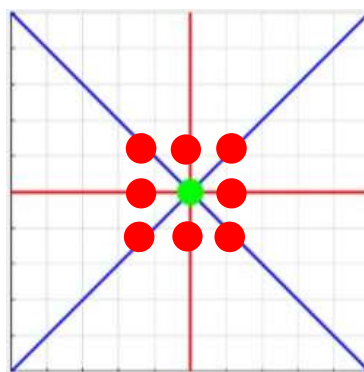
- 複数( $n$ )の性能パラメタにおける各パラメータの取りうる値から構成される $n$ 次元パラメータ空間を設定
- パラメータ空間で方向探索と次元探索を繰り返す
- 最初は探索方向を絞り、順次追加

▶例) $n = 2$ の場合:

2軸のみの方向探索  
と次元探索

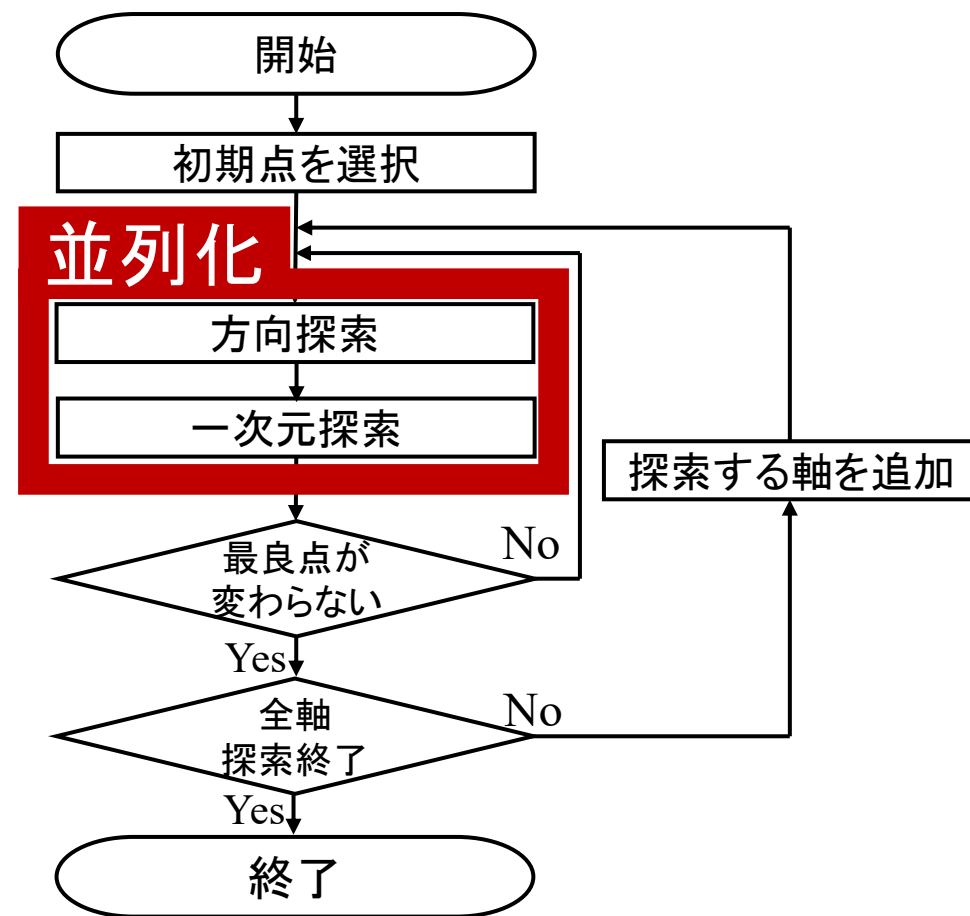
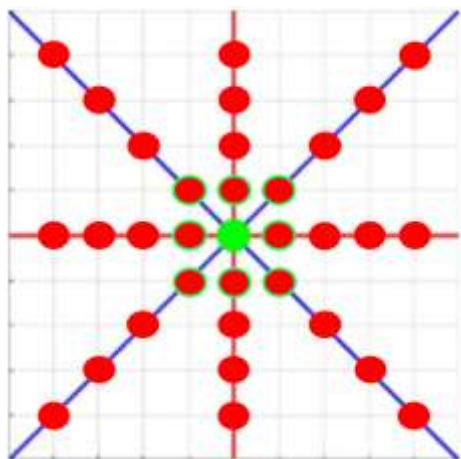


2軸を含む方向探索  
に探索する軸を追加



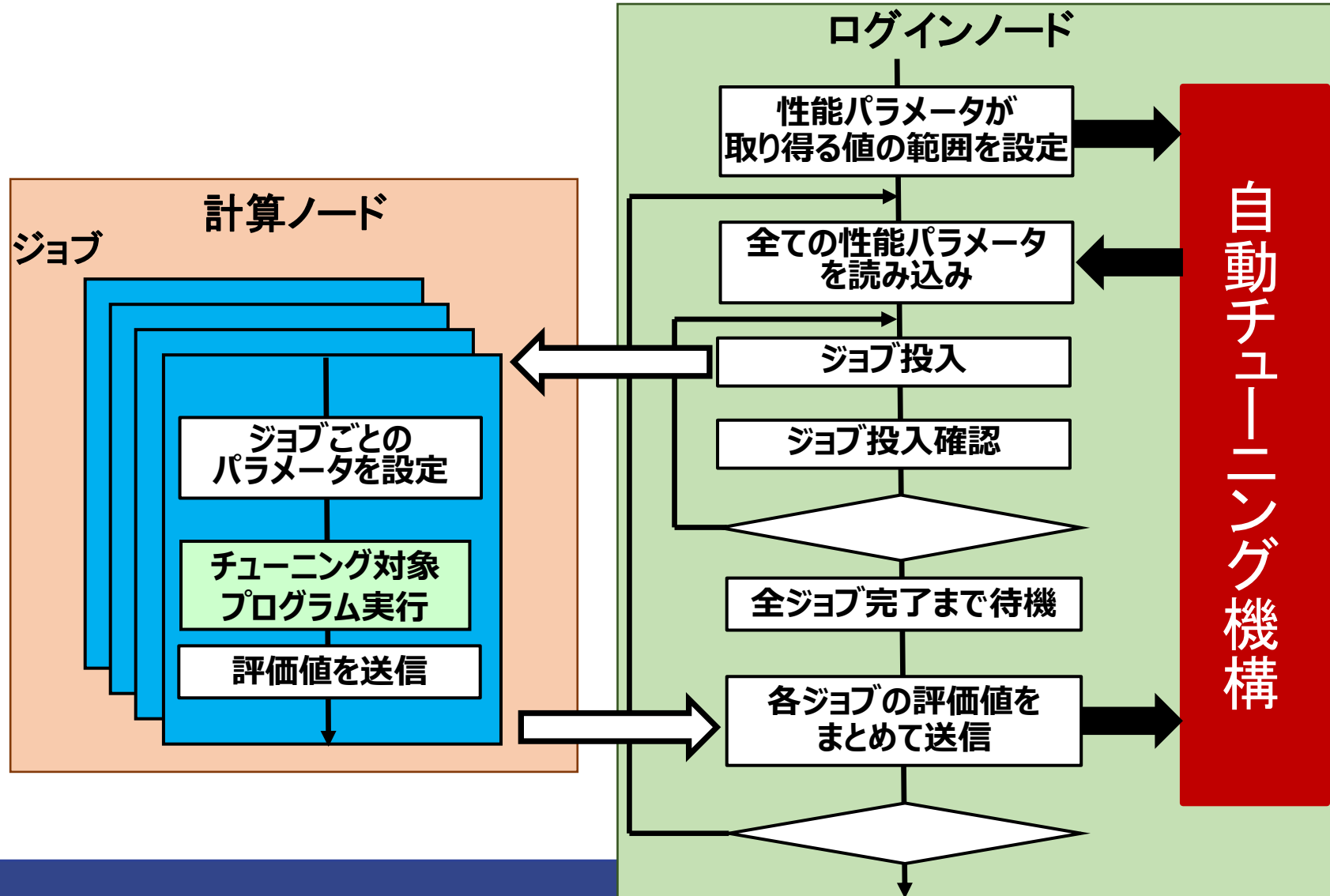
## 並列化

- 複数のジョブで同時に実測
  - ▶ それぞれのジョブが別の性能パラメータを実測
- 方向探索と一次元探索をまとめて行う





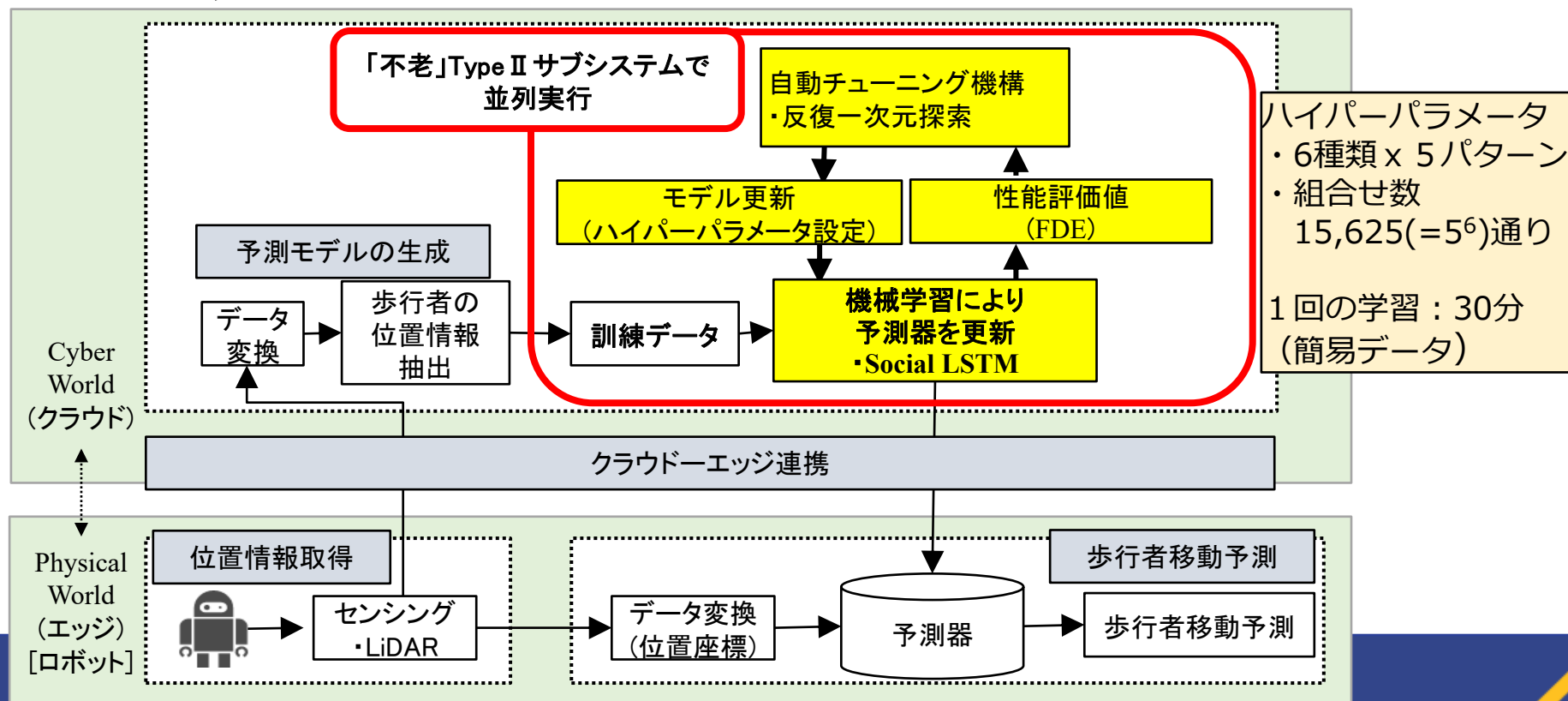
# 不老を用いたジョブ実行



# 対象とする機械学習プログラムと問題設定

## 歩行者経路予測用アプリケーション

- ▶ ロボットが歩行者に追従するように歩行者の経路と到着地点を予測
- ▶ 予測した到達地点と実際の到達地点の誤差であるFDE (Final Displacement Error) [m] で評価(小さい程良い)





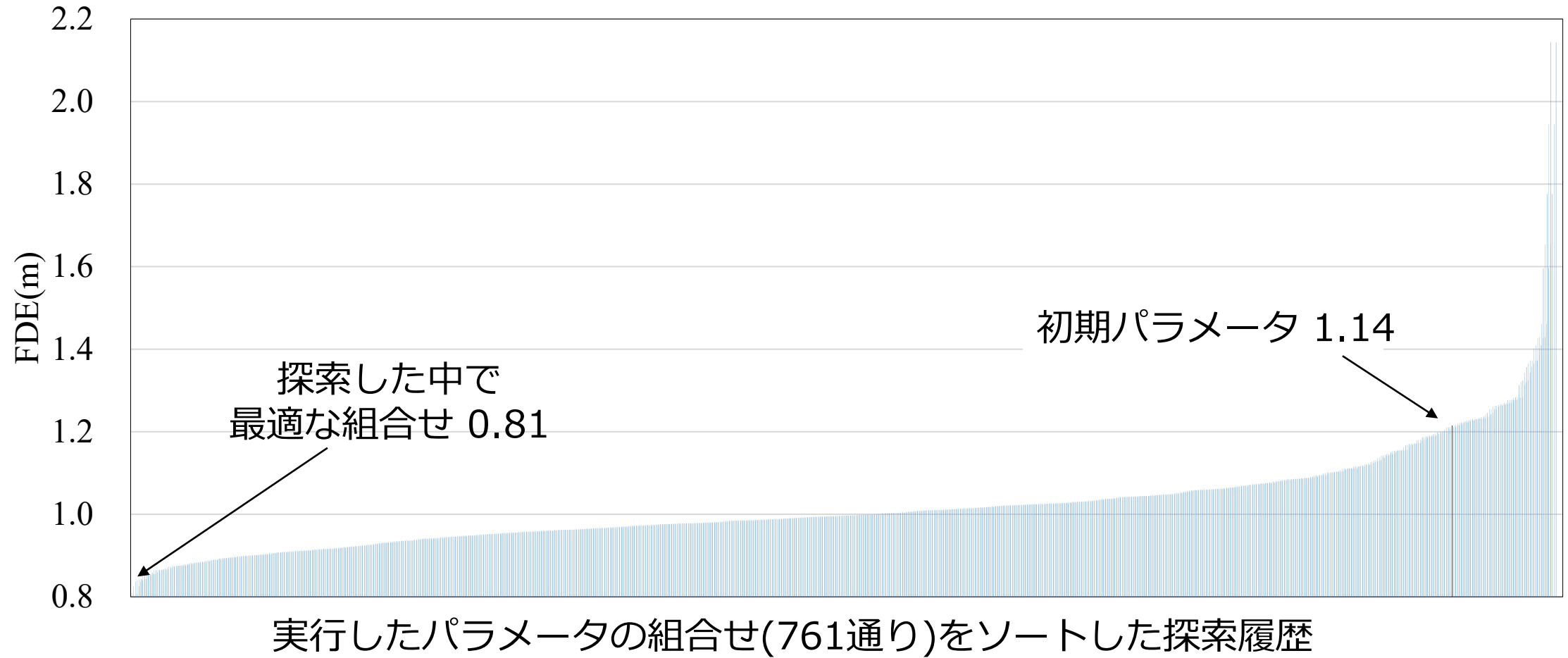
## 逐次・並列での結果の変化

・ FDE : Final Displacement Error [m]

	実行時間	探索回数	FDE(m)
逐次実行	279時間	553回	0.89
並列実行	12.3時間	761回	0.81
初期パラメータ	( 経験的に最良として使われている ハイパーパラメタの組合せ )		1.14

- FDE: 逐次実行より減少
- 推定時間: 並列化によって22.7倍高速化
- 探索回数: 並列化によって1.38倍に増加
  - ▶ 複数の方向を同時に探索した分, 探索範囲が拡大

# 探索履歴 (昇順にソート)

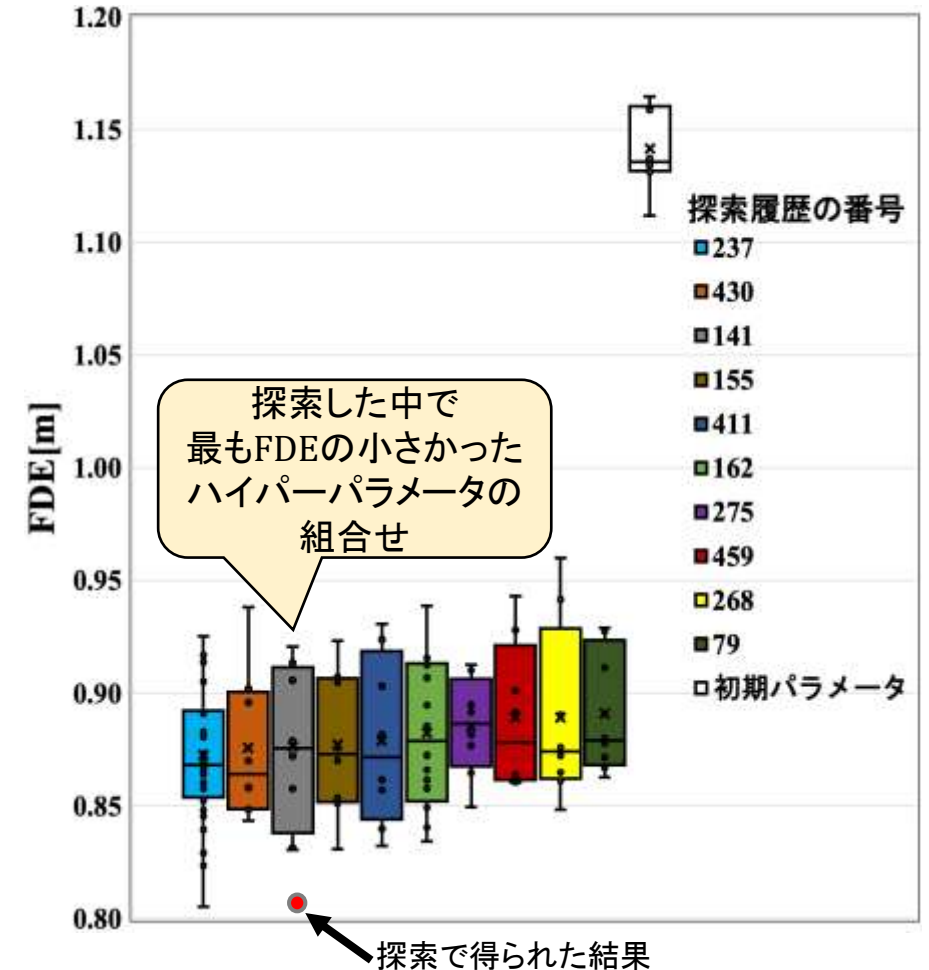


## 探索結果の安定性の検証

- 各箱ひげは、探索によって得られた性能の良い上位10通りのハイパーパラメータの組合せと初期パラメータ\*

\*自動チューニングを行う前、  
最良として使われていたハイパーパラメータの組合せ

- 推定したハイパーパラメータは初期パラメータと比べ優位な値で推移



# まとめ

- 行ったこと
  - ▶ 自動チューニングの探索時の実測を複数ジョブに分けて並列化
  - ▶ 推定結果をもとに複数回追加で実測
- 結果
  - ▶ 自動チューニングの推定完了までの時間を279時間から12.3時間まで短縮
  - ▶ 現在使われているハイパーパラメータよりも安定して良い結果となるハイパーパラメータの推定に成功
- 今後の課題
  - ▶ 自動チューニング機構のライブラリ化
  - ▶ 統計手法の導入
  - ▶ 探索回数減少のための手法の検討(枝刈り)