

AI技術“Maisart”による機器制御の高性能化

穂山利貞*
Toshisada Mariyama
増井秀之†
Hideyuki Masui
穂苅寛光†
Hiroaki Hokari

太田 佳†
Kei Ota
森本卓爾†
Takuji Morimoto

Sophisticated Equipment Control by Artificial Intelligence Technology "Maisart"

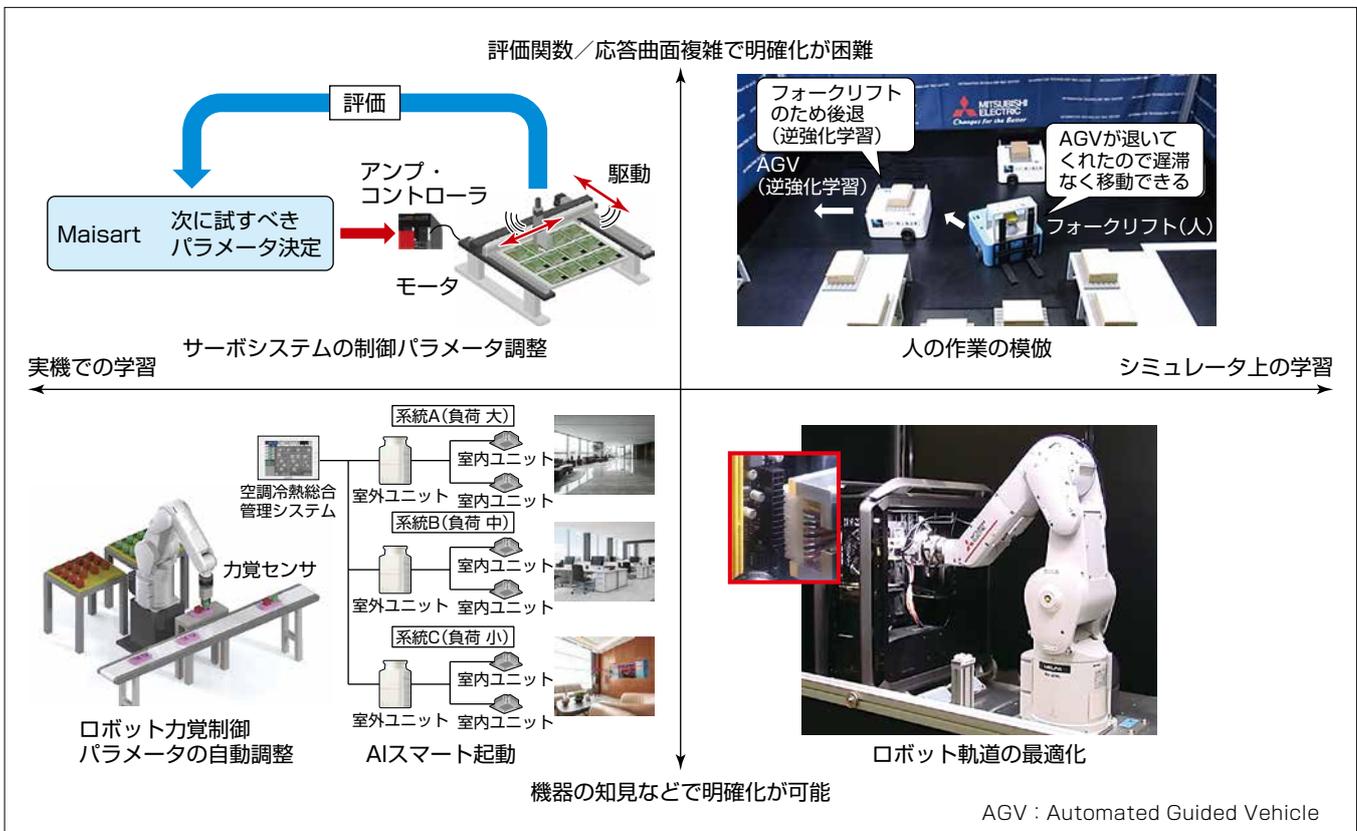
要 旨

三菱電機では、機器の課題に応じてAI(Artificial Intelligence)技術Maisartを開発することで、機器制御の高性能化を実現している。高精度なシミュレータの有無、評価関数や応答局面の複雑さによって用いる手法が変わる。

高精度なシミュレータが利用できない場合、制御分野でAIを用いる利点の一つは、機器の生のデータを使用することで最適な制御を実現できることである。その場合、実機でデータを収集するため、できる限りデータ収集量を少なく済ませる必要がある。例えば、機器の知見を用いることでパラメータの探索空間を小さくする手法や、バイズ最適化のような効率よく良解を求めるアルゴリズムの導入が考えられる。この事例として、空調機器の予冷・予熱運転

の高機能化技術、産業用ロボットの力覚制御パラメータ調整によるタクトタイム短縮技術、サーボシステムでのパラメータチューニングによるタクトタイム短縮技術を開発した。

一方、高精度なシミュレータが利用できる場合には、AIを用いることによって、より高性能な制御を導き出すことが可能になる。代表的な手法として深層強化学習や逆強化学習などが挙げられるが、どちらもシミュレータ上で膨大な試行錯誤によって最適解を求めるため、アルゴリズムの工夫で計算時間を短縮することが求められる。この事例として、産業用ロボットの軌道を短時間で自動生成する技術や、無人搬送車の制御ルールを人の運転ログデータを基に獲得する技術を開発した。



“Maisart”の機器制御への適用事例

横軸に、学習としてシミュレータを用いたか、実機を用いたかを示し、縦軸に評価関数の設計が容易か困難かを示し、当社のAI技術Maisartの機器制御への適用事例をマッピングした。右上にいくほど課題としての難易度が上がる。難易度の高い課題に対しては、シミュレータを駆使してAI手法を高度化することで課題を解決している。左下の事例は、機器の知見を用いることでデータ収集コストを下げ、課題を解決している。

1. ま え が き

AI技術は2010年代前半までは画像識別技術を中心に発展してきたが、2010年代後半からは制御分野での利用が活発に検討されるようになった。本稿では、当社が開発したAI技術Maisartの制御分野での取組み事例について述べる。AIによる制御を考えた場合、解きたい問題の難易度、シミュレータの有無などによって手法が大きく異なる。そこで、事例を紹介しながら、開発目的に応じたAI技術について事例を引用して述べる。

制御分野では、シミュレータの有無が手法選択の大きな分岐点となる。まず、シミュレータを用いずにAIを用いる場合は、試行錯誤を伴わないケースと、伴うケースの二つに大きく分けられる。試行錯誤を伴わない場合には、既に最適な、又は目的となる軌道・設定が設計されており、データを大量に収集して学習することで自動化を実現することになる。いわゆる教師あり学習が主に用いられる。試行錯誤を伴う場合、主に実機そのものを用いてデータを取得することになる。試行錯誤の回数を減らし、機器を故障から守る手段を用意することが必要である。そのため、設定するパラメータ数が少なく、実機を動かした後に、その動きを評価する指標が事前に作り込まれている必要がある。これらの取組みを2章で述べる。

一方、シミュレータが用意できる場合、シミュレータ上で様々な動きの再現が可能になる。そのため、調整するパラメータ数が大きい場合、入力次元が多い場合、出力を時系列的に扱う場合などにも使用できる。ただし、膨大な計

算時間を使えることを前提にしているが、実用上は計算時間に制約がかかるため計算時間の短縮が求められる。これらの取組み事例を3章で述べる。

2. 機器で取得したデータを用いるAI

2.1 AIスマート起動

シミュレータを用いず、実機そのものを用いて未知の最適な制御方法を学習する場合には、試行錯誤の回数を減らす必要がある。この実現には様々な方法が存在するが、一つには機器や制御対象の環境に由来する知見を用いて、学習データ数を増やしてしまう方法がある。図1に示す空調機器の予冷・予熱運転のための最適な起動時刻の学習部分がそれに当たる。この最適な起動時刻の自動設定機能“AIスマート起動”では、数日分(数回分)の起動時の実運転データ及びそこから増やしたデータを用いて、空調機器の使用環境の特性を学習させる。これを利用して、システムを立ち上げた数日間は、得られた数少ない実データから最適な起動時間を決定する。一方、実データが十分にたまってくると、実データだけをそのまま用いて学習し、最適な起動時間を決定する。Maisartのこのような工夫によって、試験設備で、学習開始から3日分(3回分)のデータで、適切な起動時間を予測できるようになった。当社ビル用マルチエアコンの最上位モデル“グランマルチ”に接続することで、この機能を活用できる空調冷熱総合管理システム“AE-200J”は2020年7月中旬に発売予定である。

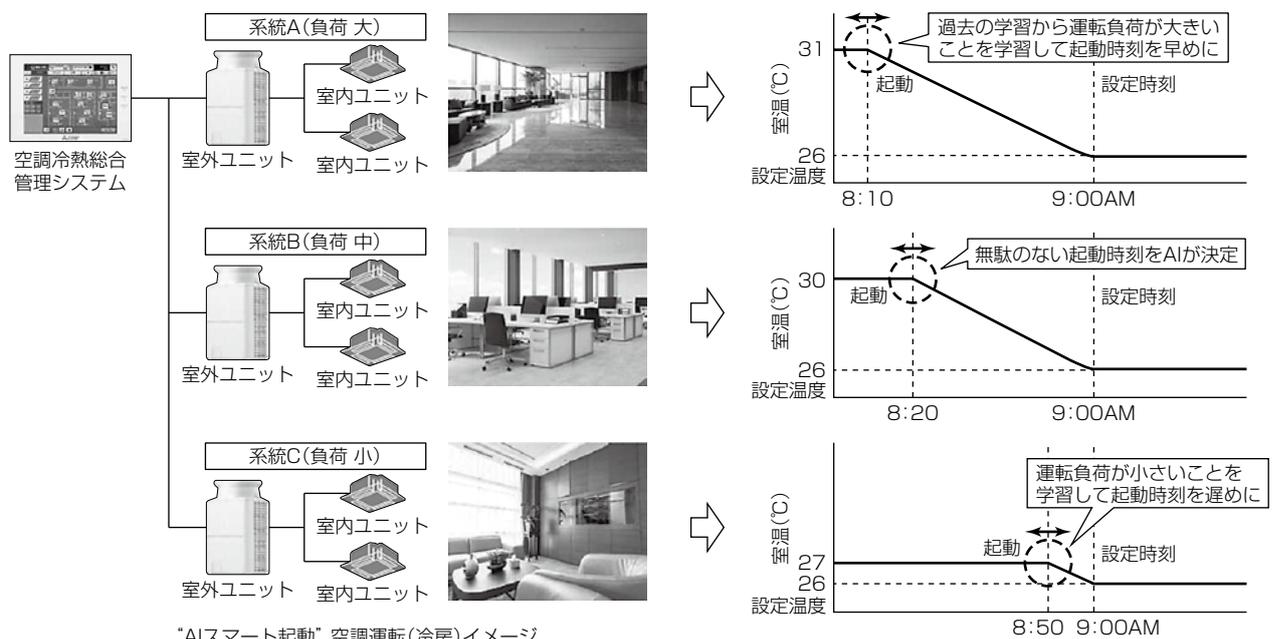


図1. Maisartによって最適な起動時刻を自動的に決定する“AIスマート起動”

2.2 力覚制御パラメータの自動調整

熟練技術者が“機器の知見”を豊富に持つ場合、重要な制御パラメータを特定し、実機での試行錯誤を経て調整が可能である。図2に示す産業用ロボットに部品の挿入動作を行わせる事例で、当社は、ロボットの制御パラメータ調整に関わる“機器の知見”に基づいて自動調整するMaisartを開発した。Maisartによって自動的に速度パターンや力覚制御パラメータが短時間の試行錯誤によって調整され、ある実験条件ではコネクタ挿入作業のタクトタイムを従来比60%削減できた。一般に、挿入作業を高精度にシミュレータ上で再現することは難しいため、シミュレータによるパラメータ調整は非常に困難である。そのため、実機を用いたパラメータ調整作業が必要になるが、この開発手法では少ない試行錯誤での自動調整を実現している。これらの機能は、当社産業用ロボット“MELFA FRシリーズ”の機能拡張オプション“MELFA Smart Plus”で“力覚センサ拡張機能”を有効化することで使用できる⁽¹⁾。

2.3 サーボシステムの制御パラメータ調整

機器や環境の特性上、取得データから機器特性の予測が難しい場合には、先に述べた手法は使えない。このような場合には、ベイズ最適化などの勾配情報を必要としない最適化手法が有効である。ベイズ最適化は、過去の試行結果からパラメータに対応した評価値の応答曲面を予測し、最適だと予測されるパラメータを試行する手法である。ここで評価値とは、機器の性能などの試行したパラメータに対する良し悪(あ)しを数値化したものである。図3はサーボモータを用いた位置決め制御システムを示している。この課題では、できる限り早く目標位置にヘッドを移動させたい一方で、振動を一定のしきい値以内に抑える必要がある。この例では、振動と位置決め時間を基に評価指標を定義し、ベイズ最適化を用いて最適な制御パラメータを求めている。その結果、今回の実験例では、従来であれば熟練技術者が1週間程度を要していた制御パラメータの調整が、1日程度で達成できるようになった。また、位置決め時間を最

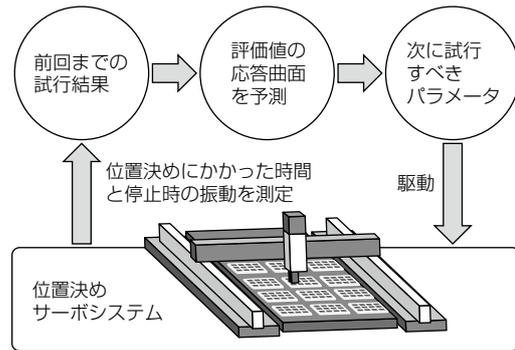


図3. AI位置決めサーボシステムの制御パラメータ調整

大20%短縮できた。このような課題を解く上で重要な点は、実機で動かす以上、試行錯誤的な動作を行える回数は非常に限られている点である。後述する深層強化学習などは、膨大な試行錯誤数が必要になるため、シミュレータを用いず実機で実現することは極めて困難である。今回のような実機による試行錯誤を行うためには、決定する制御パラメータの次元数を抑える(例：10次元以下)などの工夫が必要である。

3. シミュレータを利用した学習

3.1 ロボット軌道の最適化

シミュレータが存在する場合、シミュレータ上で多量の試行錯誤が可能になる。ただし、シミュレーション時間が無制限に確保できるというわけではない。多くの場合、現状行われている人手による設計時間との対比になる場合が多く、アルゴリズムとして計算時間の短縮が重要な要件になる。図4に示す課題は、障害物が存在する状況でロボットアームを障害物と干渉することなく、最適な軌道を生成する課題である。この課題では、人手で軌道を設定する場

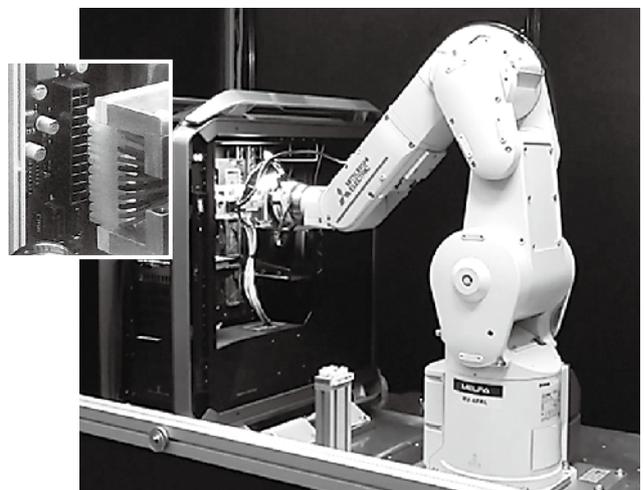


図4. 6自由度ロボットを用いた干渉回避タスク

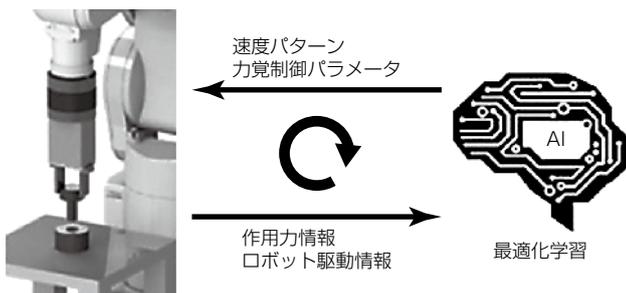


図2. ロボット力覚制御パラメータの自動調整

合、通常約2日(15時間)程度の時間を要する。自動化によるエンジニアリングコスト削減のため1.5時間を目標時間と設定してアルゴリズムを開発した。

今回開発したアルゴリズムは、課題を段階的に分けることで計算時間を短縮している⁽²⁾。大きく二段階に分けている。第1段階ではRRT(Rapidly-exploring Random Tree)という経路を算出するアルゴリズムを用いて参照軌道を算出している。RRTは最適な経路の保証、また速度や加速度などを考慮しないが、高速に解を算出できる。第2段階ではこの第1段階の経路に近い軌跡をたどり、タクトタイムが短く、加減速がロボットの物理的特性の範囲内に入るような軌道を生成する。このアルゴリズムによって、第1段階では1.00秒かかっていた動作が、0.44秒にまで短縮されている。また、計算時間は、今回の課題に対して目標である1.5時間以内に収めることが可能である。ちなみに、このような工夫を凝らさずに強化学習を事前に行った場合、72時間の計算でも解を算出できなかった。

3.2 人の作業の模倣

3.1節では強化学習の実例について述べたが、この節では逆強化学習を用いた実例について述べる。強化学習を用いるためには報酬関数と呼ばれる行動結果の良好度合いを定量化した関数が必要であるが、その報酬関数の設計が非常に困難な場合がある。例として、自動車の運転を考えた場合、前の車と適切な車間を取り、急ぎ過ぎず、他の交通参加者への安全に配慮しながら運転をするのが好ましい運転であると考えられるが、これを表現した報酬関数の設計は極めて困難である。“お手本となる動きは存在するが、評価関数が設定しきれない”といった状況下で現在注目されているのが逆強化学習といわれる手法である。

この節では、AGVと呼ばれる工場内で走行する無人搬送車を対象に、人のような運転を学習させる技術を述べる⁽³⁾。現状では、無人搬送車は設計者によって、防護範囲の設定、防護範囲に障害物を検知した場合の動作などの制御ルールをあらかじめ設計している。逆強化学習では、人がAGVを運転したとき、その運転データから人の考える報酬関数を推定することで、人のような動作を模倣することを可能にする手法である。図5に表示されたタスクに対し、人が操作したログデータを用いて逆強化学習でAGVに学習させた。その結果、従来であれば人手によって設計した制御ルールをAIがログデータから獲得することで、AGVとフォークリフトの経路干渉の機会が減り、テストシステム全体で30%程度の作業効率の改善ができた。

教師あり学習もこれを実現できるが、教師用のデータを



図5. 逆強化学習の応用例(人との協調動作)

大量に用意する必要が出てくる。教師あり学習は発生し得る状況のデータを収集し尽くし、制御の最中に発生する制御誤差が発生した場合の補正用のデータを準備する必要がある。逆強化学習は、推定した報酬関数に基づいてシミュレータ上で試行錯誤することで、上記課題を回避する特性がある。先の例で評価した場合、教師あり学習と比較して1/2~1/10程度のデータでより良い性能が得られている。このように、報酬関数が設計しづらい一方、教師となるログデータが存在し、シミュレータが利用できる場合には、ログデータのルールを自動的に獲得する逆強化学習も今後有力な選択肢になっていく。

4. むすび

制御分野でのMaisartの開発事例をユースケースとして紹介し、課題に合わせたAIの選定方法を述べた。本稿では触れてはいないが、開発の現場では、今回述べたようなシミュレーションなどの机上検討の前にサンプルデータによるデータ解析、シミュレーション後での実機検証試験に非常に多くの時間を費やしている。どのステージも重要であり、検証を重ねる必要がある点を付け加える。

今後、AIの制御分野進出に関しては、シミュレーションと実環境との差、AIの信頼性、品質保証、説明性といった課題が数多くあり、これらを解決していくことがAIの成功の鍵を握っていると考えられる。

参考文献

- (1) 三菱電機産業用ロボット機能拡張オプションMELFA Smart Plus <https://www.mitsubishielectric.co.jp/fa/products/rbt/robot/pmerit/msp/forcecensor.html>
- (2) Ota K., et al.: Trajectory optimization for unknown constrained systems using reinforcement learning, IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 3487~3494 (2019)
- (3) Matsunaga S., et al.: Learning to Drive via Generative Adversarial Imitation Learning with Top-down View, IROS, IEEE (submitted)