

1. 目次

1	背景.....	2-2
1.1	ロボティクス分野について	2-2
1.2	強化学習について	2-2
	Behavior Cloning.....	2-3
	専門家データ融合 : Dataset Aggregation	2-3
	逆強化学習	2-3
1.3	ロボットアームの先行研究について	2-4
1.4	ダイレクトドライブについての先行研究	2-5
1.5	力制御について	2-7
1.6	本論文の位置づけ	2-7
2.	手法と環境構築	2-8
2.1	実験に使用したロボットアームについて	2-8
2.2	モータドライバの制御	2-11
2.4	モータドライバの交換	2-13
2.5	制御プログラムの設計	2-15
3.	実験	3-16
3.1	学習データの収集	3-16
3.2	学習手法について	3-17
4.	結果	4-19
5.	結論と考察	5-20
1.7	ティーチング時に発生する外力の問題について	5-20
6.	謝辞	6-21
7.	参考文献	6-22

1 背景

1.1 ロボティクス分野について

現代の生産業や医療業界などの産業は、生産性を向上させ品質の高い製品を提供するために、手作りでの製品の生産からエンジニアリングを活用した自動化への流れが急速に加速している。これらのアプリケーションでは正確な物体操作やアーム操作を実現し、安定して目的を達成することが求められる。また、このようなシステムにおいて、操作の柔軟性は非常に重要な観点を占める。このような現場においては、センサを用いたフィードバック制御が重要であり、近年は画像センサを機械学習の手法を使ったアプローチが多くみられる。

そのなかでも、多リンク系のロボットアームはその汎用性や人間と似た動作をしやすといった特徴から非常によく好まれている。ここで多リンク系のロボットとは、ロボットにおいて回転する軸である関節どうしをつなぐ剛体をリンクと呼び、これが直列につながっているものを指す。そこで、本論文では機械学習を用いてより汎用的でロバストな物体操作を行うためのロボットアームの制御方法についての研究が必要である。

1.2 強化学習について

機械学習とは、経験からの学習により自動で改善するコンピューターアルゴリズムのことを指す。機械学習にも様々な種類があり、一般的には強化学習、教師あり学習、教師なし学習、模倣学習などが挙げられる。強化学習は、エージェントが環境において複数のアクションを実行し、その際に環境から受け取る報酬を最大化するほうに学習する手法である。教師あり学習とは既知の正解の入出力データを与え、それを教師データとして学習する手法であり、教師なし学習は正解データなしに与えられたデータから、その中に含まれる構造や法則といった情報を抽出する手法のことである。

ロボット分野においても機械学習を用いてより人の介入を減らそうとする動きがある。強化学習や物体認識といった教師なし学習の技術を使い、問題のタスクについてシステムが最適なモデルを設計し、安定した動作を実現するためのネットワーク構造またはアルゴリズムを使われることが増えている。このように機械学習をつかうことにより、プログラミングなどで動作を明示的に示さずとも、将来のタスクやアクションをシステムが自動的に決定することができる。

そのような機械学習の一例として模倣学習がある。模倣学習とは、人の手本となる行動データを真似するようにエージェントに学習させることを模倣学習という。模倣学習の入力データは完全な正解ではなくノイズやランダム性を含むデータセットであるが、その中から特定の行動を取り出し意図した動作をできるようにパラメータチューニングすることが必要である。

模倣学習には以下のような種類がある.

Behavior Cloning

教師が作成したデータを理想とするポリシーとし, 正解データのポリシーとエージェントのポリシーの差分を表す関数の値をできるだけ小さくするようにパラメータを学習することで, 専門家のポリシーに近づける手法である.

行動クローニングが初めて提案されたのは, Dean A. Pomerleau らによる道路追従のタスクを目的とした論文である [1]. 路面を撮影したカメラ画像を基に, 三層のニューラルネットワークを用いて, 一定の制約のもとで訓練することで実際の道路に効果的に追従することが確認できている. もう一つの行動クローニングが上手く活用された例として, 航空機の飛行シミュレーションが挙げられる [2]. このシステムは航空機を飛行させる際に, 環境の乱気流などの様々な条件下において目標とするフライトを達成したことを述べている.

そして, この行動クローニングの技術を利用してロボットアームを制御しようという試みが近年行われており, その内容は 1.3 節で示す

専門家データ融合 : Dataset Aggregation

Dataset Aggregation (DAgger) [3] と呼ばれる模倣学習のアルゴリズムは, **Policy Iteration** と呼ばれる方策反転を行い, 学習済みのすべての方策が訪れた状態を新たな専門家データとして扱う手法である. これによって, 何回な模倣学習問題であるベンチマークのシーケンスラベリング問題で以前のアプローチを上回ることを実証している. しかし, 誤り伝播と呼ばれる問題点が指摘されている. これは, **Policy Iteration** の影響で, ある一つの行動がその後の学習のデータとして残り続けるという問題点である.

逆強化学習

逆強化学習(**Inverse Reinforcement Learning; IRL**) [4]とは, エキスパートの行動データを用いて報酬関数を推定し, その報酬関数に基づいてエージェントが行動を選択する手法である. そのなかでもニューラルネットワークを用いて学習するものとして, **Deep IRL** [5]や **Adversarial Inverse Reinforcement Learning (AIRL)** [6] **GAIL** [7]とは, エージェントは, エキスパートの行動が最適なものであると考えられるような報酬関数を学習しようとするため, ただティーチングデータをまねるよりもより良い結果となることがある

1.3 ロボットアームの先行研究について

ロボットとは、人間に似た動作機能を有している機械のことを指す。近年はロボットによる産業自動化が非常に激しく進んでおり、システムの自動化において重要な役割をいんあっている。しかし、まだ完全に人間の介入無く自動化を行うには正確性やエラー検知などの問題もあり、さらなる技術の発展が必要である。そのなかでも、機械学習や物体検知といった研究分野の登場、発展により、問題に対し自律的に行動するロボットの設計、制御が望まれている。

過去にも強化学習や模倣学習を用いたロボットアームの制御についてはいくつもの研究がなされている。本論文では「ペグインサーション」をタスクとして設定しているため、似たような動作を行う他リンク機構の制御を扱った論文をいくつか先行研究として取り扱う。

模倣学習を使いロボットアームの操作を行う研究例としては以下のようなものが挙げられる。

先行研究（学習方法略称または著者名）	シミュレーション	入力	タスク
HA-PASTIL を用いた模倣学習 [8]	シミュレーション	F/T ジョイントトルクセンサ	ピッキング、物体の持ち上げ
Simon Stepputtis, 2022 [9]	シミュレーション	F/T ジョイントトルクセンサ	物体の組み立て
SAFARI [10]	シミュレーション, 現実のロボットを含む	画像シーケンス	オブジェクトを動かす
VIOLA [11]	シミュレーション, 現実のロボットで実演	画像シーケンス	ピッキング
Heecherol Kim [12]	現実のロボットで実演	視線画像	バナナの皮をむくタスク

Table .1 Some papers about imitation learning with robotic arm

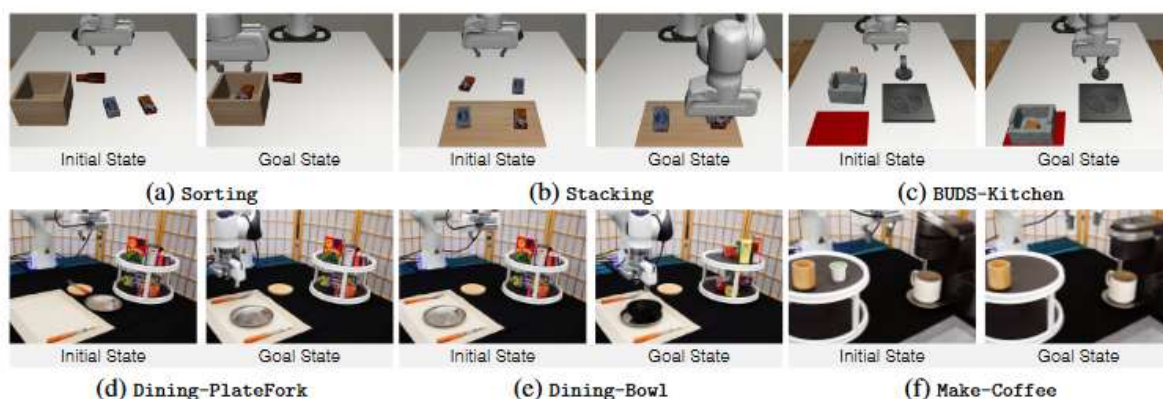


Fig .1 Some example tasks of VIOLA [11]

また、近年は Implicit Behavior Cloning [13]といった手法も注目されるようになっている。

このように複数の手法が模倣学習を使って複数のタスクを上手くこなせていることが分かっているが、複雑なタスクを実際のロボットに動作させる課題においても一定の成果が出ている研究は画像や動画といった情報を使ったものが多く、トルクセンサ等といった体制感覚のみで動作しているものは少数であった。

1.4 ダイレクトドライブについての先行研究

産業用ロボットにおいて、動作教示（ティーチング）方式によるロボット制御が非常に多くの場面で使われている。動作教示方式とは、模倣学習のように専門家が実際にロボットを動かすことで作業内容をシステムに記憶させ、その内容をロボットが再現することで問題を解決する駆動方式を指す。この際、ダイレクトドライブモータを使用したロボットは動作教示方式において非常に有効である。

ダイレクトドライブ（方式）とは、関節の出力とモータとの間にギア等を用いた減速機構を使わずに、入出力を軸で直接つなげて駆動する方式である。これにより、ギアのバックラッシュといったガタの発生が少なくなる、減速機におけるエネルギーロスや騒音の発生を抑制できる、力制御に向いているなど、様々なメリットがある。ダイレクトドライブを用いない一般的なロボットアームでは、高剛性なため外部から力を加えて動かすことが難しいが、ダイレクトドライブ方式はその性質上、ロボットの関節を外部から直接動かすことが簡単のため、動作教示がしやすく、模倣学習の相性が良いこともわかっている。

ダイレクトドライブの多関節ロボットアームに関する研究の一例として、機械技術研究所で1977年に開発されていた7軸のマニピュレータが挙げられる[14]。この研究では、油圧方式のダイレクトドライブ方式を用いて高精度の位置決め制御を実現できることがわかった。

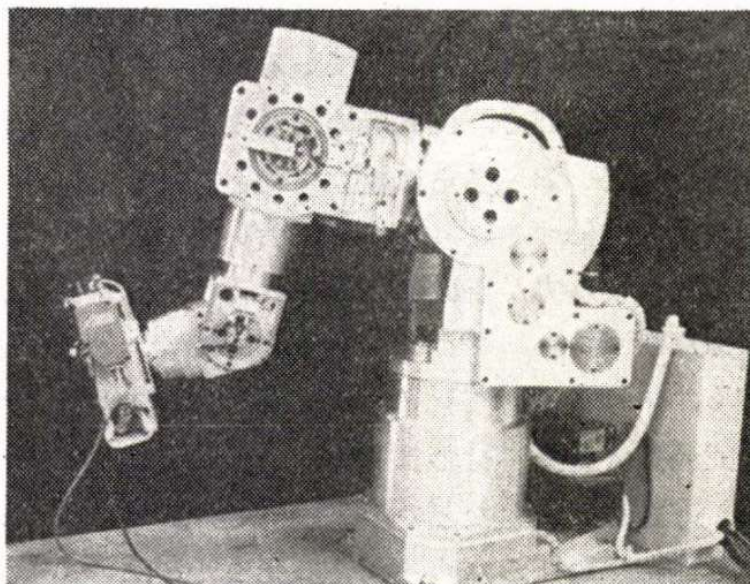


Fig. 2 6-axis Direct Drive manipulator in 1977 [13]

また三軸程度の低自由度の機構におけるダイレクトドライブの研究はいくつか行われていたが、[15][16]、電動のモータを関節に用いたダイレクトドライブ多関節ロボットアームの研究で6軸以上の関節をもつロボットにおける研究例は小数であった。いくつかの例としては、根本の二関節のみ能動重力補償機構を搭載することで7軸のロボットアームのエンドエフェクタが目標軌道を動かせるようにする研究[17]等がある。これは、ダイレクトドライブは減速比を用いないため、出せる関節トルクが一般的な機構に比べて非常に小さいという欠点からくるものであった。しかし、近年になって急激にモータが出せるトルクが大幅に向上したため、次に示すようないくつかの研究が行われるようになった。例えば昨今ではダイレクトドライブモータを歩行ロボットの足回りとして使用する例がいくつか見られる[18]。これらの事例はいずれも目標位置追従誤差を最小化するような位置制御で動いているものが多く、そのフィードバックを高速で行うことによって高い安定性を実現するシステムである。もしくは、力センサやトルクセンサを用いずにダイレクトドライブのインピーダンス制御を行う研究例などもあり[19]必要なトルクを関節角速度や角加速度から推定することで算出して制御できることなどが分かっている。

1.5 力制御について

ペグ挿入などの外界との接触を伴う動作を実現するにあたって、位置だけではなくロボットが受ける、または各関節が出力する力やトルクの情報を活用することは重要である。この際に、従来の位置制御とは異なり、フィードフォワードの関節トルクによってロボットの運動を制御しようという手法を力制御という。力制御では、関節に取り付けたトルクセンサの値や電流値をもとにフィードバックループを構成し、制御を行う。また力制御はその特性上、外部と接している際にその外力を制御することが可能であり、今回の様に力を調整する必要があるタスクの実現に向いていると考えられる。

力制御の例としてインピーダンス制御やアドミッタンス制御が挙げられる。インピーダンス制御 [20] は、ロボットが特に外部と接触している時に、特定の力や力の変化に対して制御された反応を提供するための制御手法である。この手法は、ロボットが外部環境やユーザーとの相互作用に適応し、柔軟かつ安全に動作できるようにするために使用される。

また、ダイレクトドライブ方式は力制御においても利点になりうるのではないかと考えられる。力の伝達が直接行える上に、減速機の摩擦などに影響されない緻密な力制御ができるのではないかと仮説からである。

1.6 本論文の位置づけ

以上のことにより、外界との接触を含むタスクを柔軟に実現するには、

- 模倣学習による適切な方策の決定
- ダイレクトドライブによる力制御

が必要不可欠であると考え。手先が外部に触れる必要があるタスクにおいて、位置制御では手先の制御誤差によって機械に大きな負荷が発生しうるため相性が悪く、力制御と機械学習を組み合わせることで動作の柔軟性を獲得することが必要不可欠である。このような力制御と模倣学習を組み合わせた研究はシミュレータ上でのロボットの学習タスクにおいては数多くの事例があるが、ダイレクトドライブを用いた実機という一つのシステムで完結した研究例はあまり見受けられなかった。そのため、ダイレクトドライブや力制御、模倣学習といった要素がタスクの達成率や動作の柔らかさにどれほど寄与するかを調べる必要があり、本論文はそれを目標としている。

そして、その指標としてペグ挿入を行い、それぞれの要件について評価することが目的である。

2. 手法と環境構築

実験では、下図のような6軸の直列リンク機構を持ったロボットアームを制作し、それに対して基本的図形のペグ挿入タスクを模倣学習で学習させた。

2.1 実験に使用したロボットアームについて

ロボットのハードウェアについては、以下のような6軸のダイレクトドライブ方式の多関節マニピュレータのロボットアームを用いた。なお、このロボットのハードウェアは、日本ビジネスシステムズ株式会社から東京大学工学部機械情報工学科 知能システム情報学研究室に提供いただいたものを使用しており、本研究ではソフトウェア及び回路モジュールを改良し、パラメータチューニングなどを変更したものを使用している。

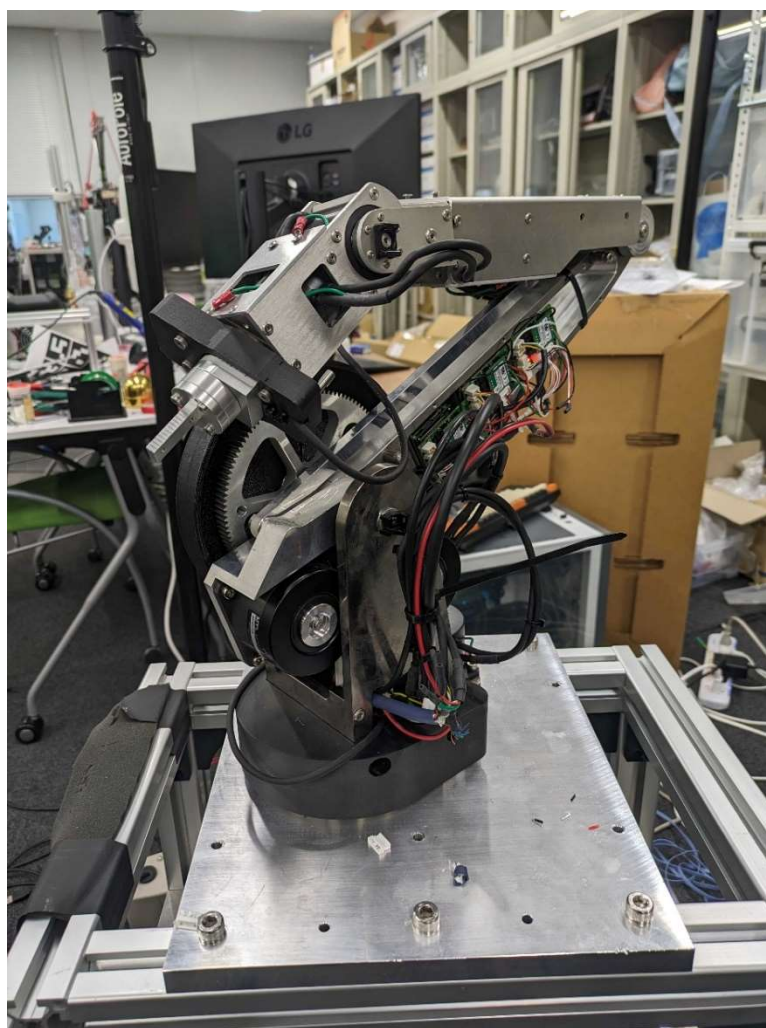


Fig. 3 Direct Drive 6-axis robotic arm in this research

ロボットは肩関節がピッチヨーの3軸, 肘関節が1軸, 手先の関節がピッチロールピッチの3軸で計6軸となっている. 先端の位置と姿勢を考えると6自由度あるので, 冗長性のない至適の設計である. 主な部品は以下の様な構成となっている.

	型番など	個数	備考
関節モータ	LAC16D04 3 MDH(12) 40E	3	肘関節より先の部分に取り付けられている.
関節モータ	LAC17E03 3 MD 4018 20B	3	肘関節より根本の部分に取り付けられている
制御基盤(モータドライバ)	MAN-G-SOLTWIIG 5002	6	
6 軸力覚センサ	Leptrino の PFS シリーズ	1	エンドエフェクタにつけられたフォーストルクセンサ
制御コンピュータ		1	C++のコードによって UDP 通信でモータドライバと接続し, 関節の情報を読み取ったり関節トルクの情報を送ったりしている
モータドライバ 制御モジュール	Maestro Multi Axis Controller	1	制御 PC から送られてきた信号をそれぞれのモータドライバにつなげる

Table .2 Main parts of the robotic arm

て

肘関節から先の部分の3関節と肩関節の3関節は使われているモータがことなり、上記で示した表のような構成となっている。これは、よりトルクを出す必要があり、設計において物理的スペース似余裕のある肩関節部分により高出力なモータを取り付けるという目的であると考えられる。



Fig .4 Maestro Multi Axis Controller (Image from Elmo Motion Control website)

そして、末端部分には6軸のフォーストルクセンサ（力覚センサ）が搭載されており、XYZ 方向それぞれの力とモーメントを読み取ることができる。

そして、各モータとモータドライバ、コントローラは以下の様に接続されている。。

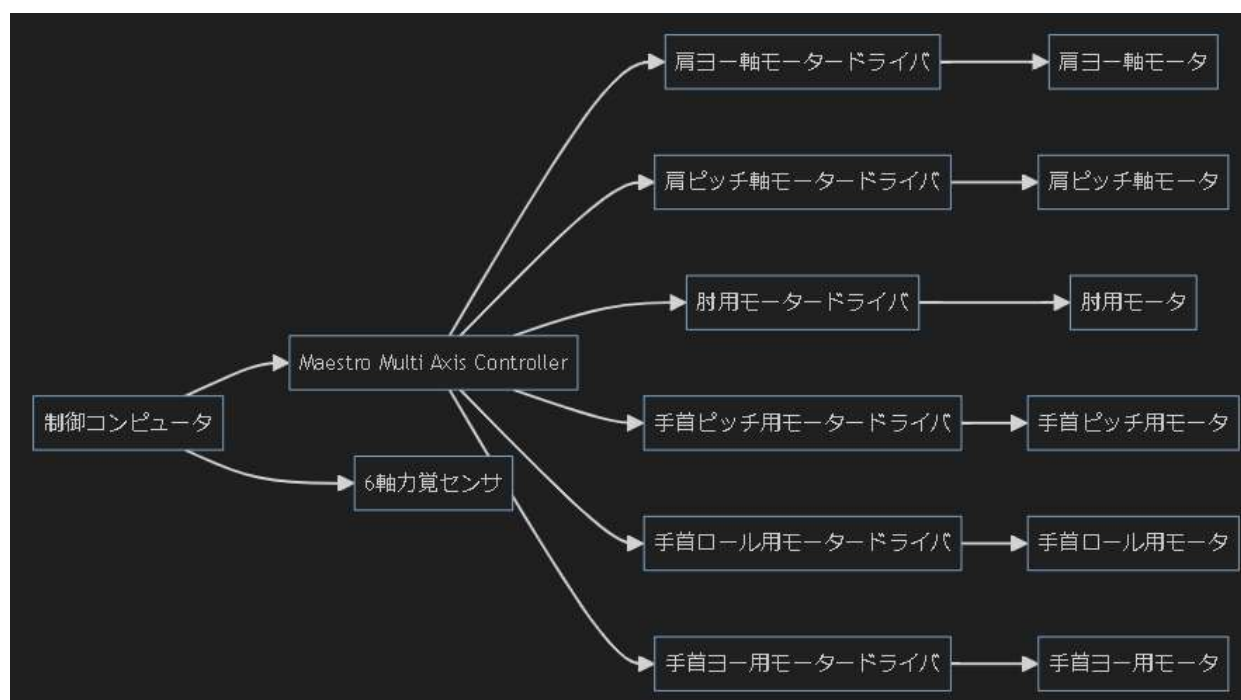


Fig .5 Main architecture of the robotic arm system

制御コンピュータは、C++で書かれたプログラムによって、UDP 通信で Maestro Multi Axis Controller と通信を行い、それぞれのモータドライバに信号を送ることで、モータを制御している。ここでは、

Maestro Developer Studio のライブラリで提供された API を用いて通信を行っている。そして、Maestro Multi Axis Controller は、それぞれのモータドライバに信号を送ることで、モータを制御している。ここでは、EtherCAT 通信を用いて約 1000Hz で高速に通信している。

t

それぞれのモータドライバと Maestro Multi Axis Controller には電源が供給されており、それぞれ専用の 24V 電源と 12V 電源が用意されている。また、制御コンピュータには、5V 電源が供給されている。

また、制御コンピュータは、6 軸力覚センサからの情報を読み取っている。ここでは制御コンピュータの USB 端子に変換器を通して、6 軸力覚センサを接続している。

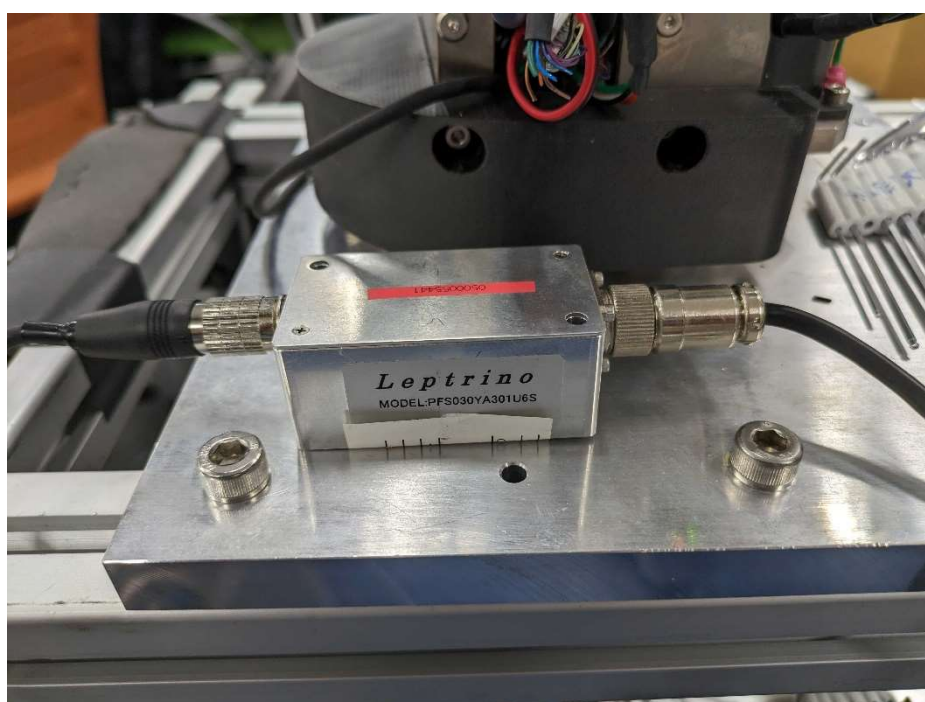


Fig 5 6-axis force / torque sensor

2.2 モータドライバの制御

それぞれのモータは PID 制御を用いてコントロールしている。PID 制御とは古典的なフィードバック制御の一種で計算量が少なく高速に動作する一般的な制御である。比例制御（P 制御）、積分制御（I 制御）、微分制御（D 制御）を意味し、それらのフィードバック功を加えたものである。

K_p, K_d, K_i の 3 つのパラメータを持つ

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$$

P 制御では目標値と現在地の偏差に K_p をかけたものを制御量とし、D 制御では目標値との差の時間微分に K_d を比例定数としてかけたもの、I 制御では時間積分したものを制御量とする。また、制御電流が想定以上に大きくなるのを防ぐため、出力電流に 1000mA のリミットを書けている

モータのキャリブレーションを行うために下画像の様に二つのモータとモータドライバを繋げた簡易的なシステムを作り，ここでそれぞれのパラメータ調整を行った。

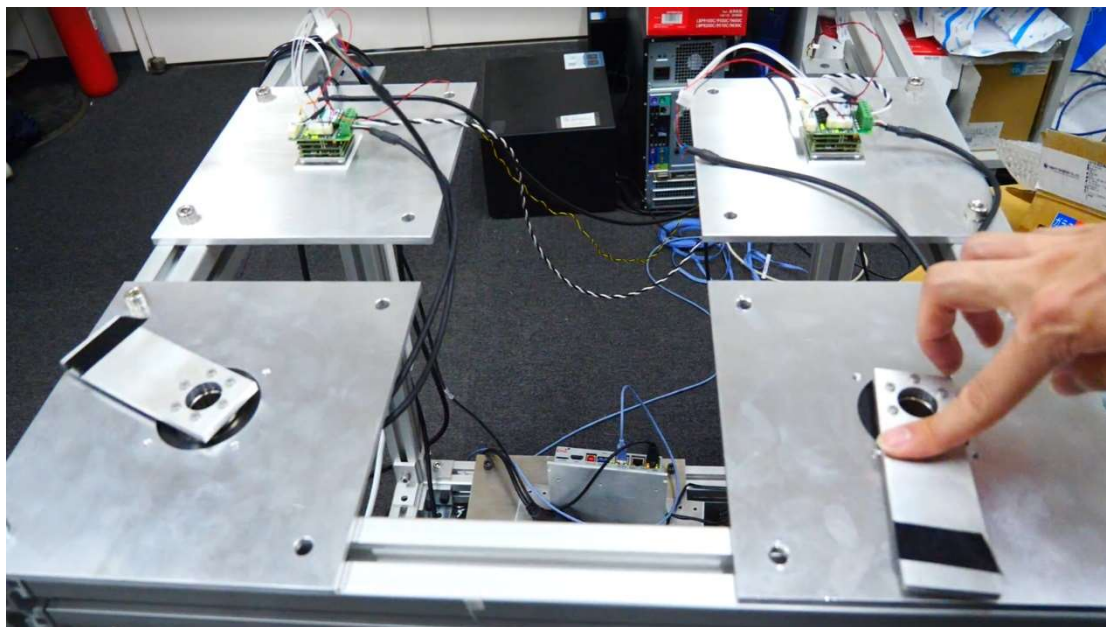


Fig 6 Experiments and PID parameter tuning on motor drivers

以下の画像はその際に一方のモータを手で動かし，もう一方をその回転量を送ることで追従させる制御をしたときの，それぞれのモータの回転角のグラフである。



Fig. 7 PC monitor during PID parameters tuning

このように、二台のモータが互いに追従しあっていることがわかる。また、一方を抑えた状態から

2.4 モータドライバの交換

本来のモータドライバ基板では、モータドライバを統合する基板と制御 PC のプログラムの間で最大数秒程度のラグがあり、リアルタイムにロボットを動かすことが困難であった。その為、モータドライバを交換し EtherCAT 通信で高速に Maestro Multi Axis Controller と通信することを可能にした。

交換部は下図において四角の枠で囲まれた二か所である。それぞれ肘関節より先のモータドライバ（黄色部分）と肩関節と肘関節のモータドライバ（水色部分）であり、それぞれ 3 つのドライバを直列でつなぎ直した。

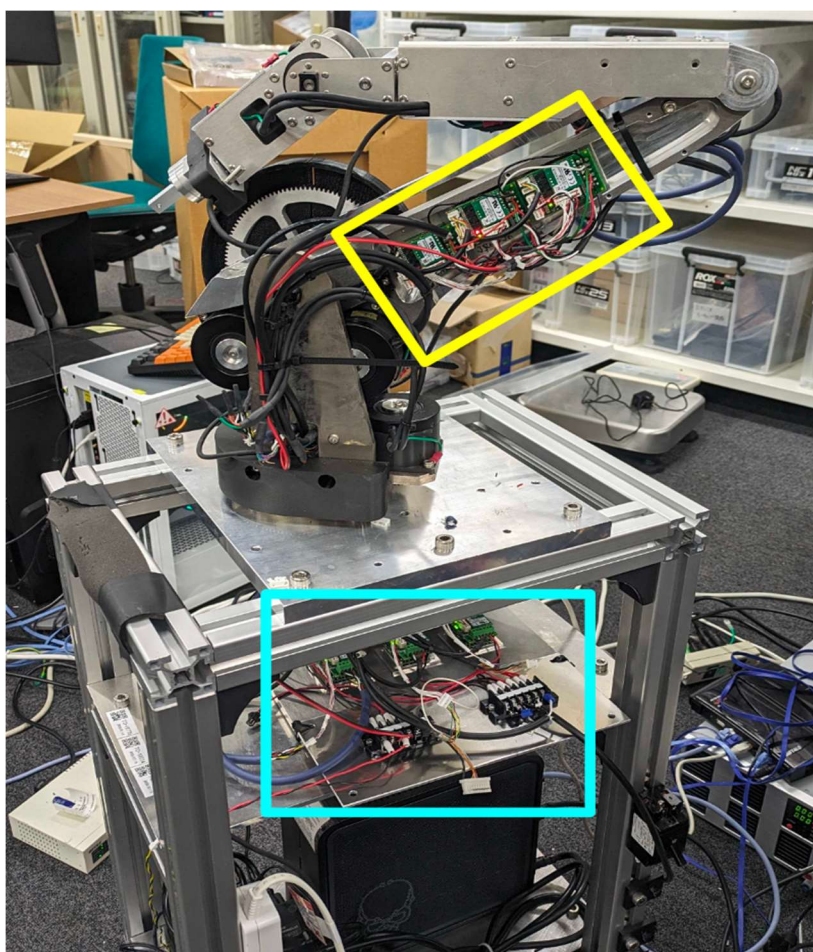


Fig. 8 Motor driver circuits exchanged for better performance

肩の部分の配線は以下ようになっており、ロボット中央の内部を通して配線されている。

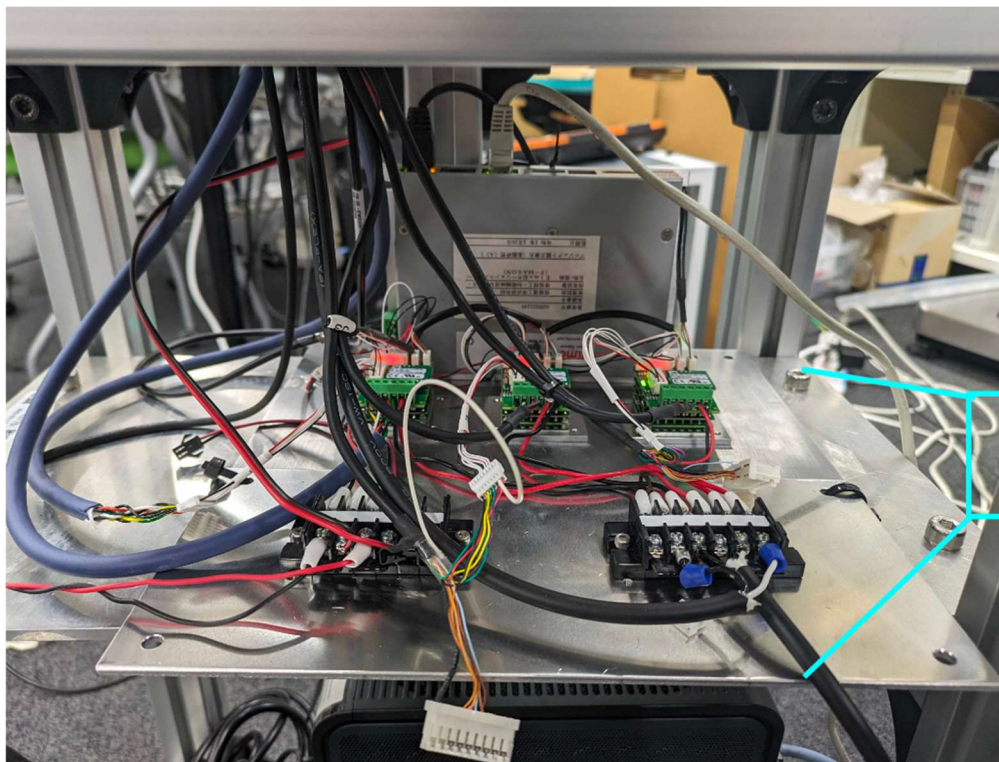


Fig. 9 Motor drivers for shoulder actuators

その後、6つのモータドライバに同時に接続し、それと同時にPCとも modbus で通信するプログラムを C++ で書き直し、Maestro Multi Axis Controller にインストールすることで、約 1kHz で高速に制御PCのプログラムとそれぞれのモータドライバが通信できるようなシステムを作成した。

2.5 制御プログラムの設計

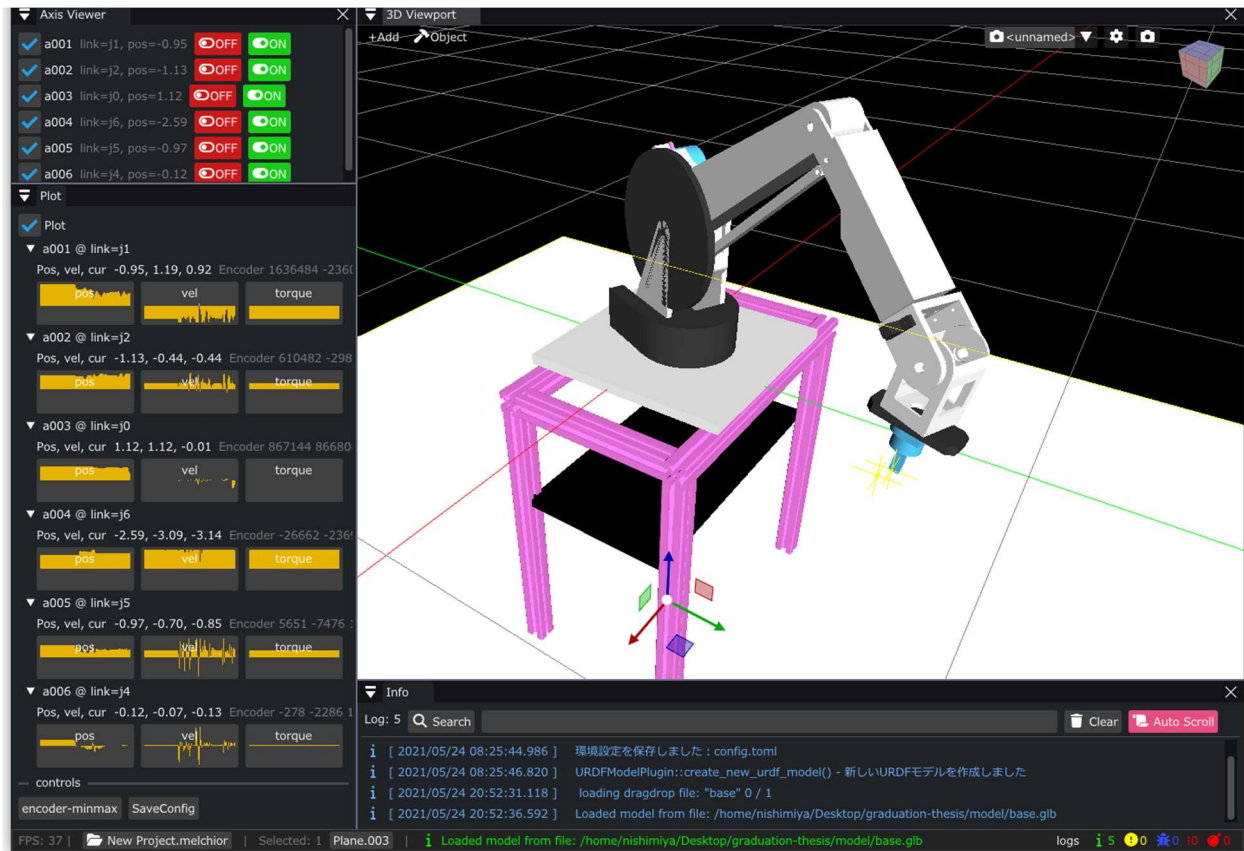


Fig. 10 C++ program to collect imitation learning dataset

制御プログラムはC++を用いて実装し、OpenGLを用いた独自のロボットの状態を可視化するアプリケーションを作成した。これにより、GUI環境から各関節等のデータの表示やモータのON/OFF、モータドライバのPID制御のパラメータ調整や再設定等ができるようになり、より効率的に学習データを集められるようになった。

3. 実験

3.1 学習データの収集

学習には、以下の様な円柱形と直方体の二種類のペグを力覚センサの先端に取り付けたものを使用している.それぞれのペグの設計図は以下のようにになっている.

直方体のペグ先端の一辺の長さ, 円柱形のペグの直径はともに 17mm, 高さ 10mm のペグをそれぞれ Fusion 360 で設計し, 3D プリンタで ABS 樹脂を加工して作成した.

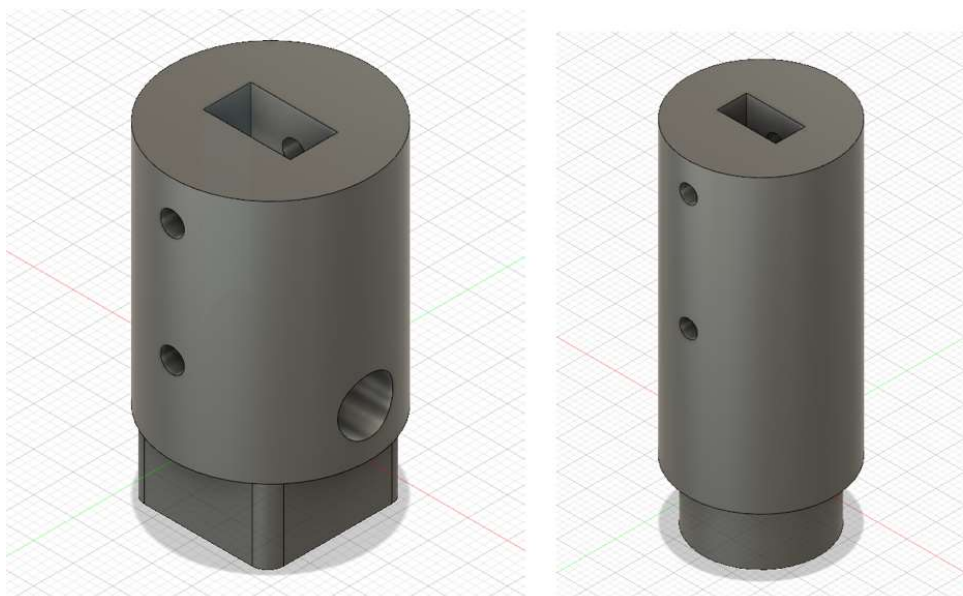


Fig. 11 Peg insertion CAD models

上図の様に, 力覚センサの先端部分にペグのアタッチメントをネジで固定し, ペグに加わる力を直接測定できるようにした.

そして, 挿入先は厚さ 5.5mm の MDF 板をレーザーカッターで加工したものを使用しており, 以下の様な設計図で加工した. そして両端をネジでロボットアームの土台に Fig.12 の様に固定し, ペグ挿入の台として使用した

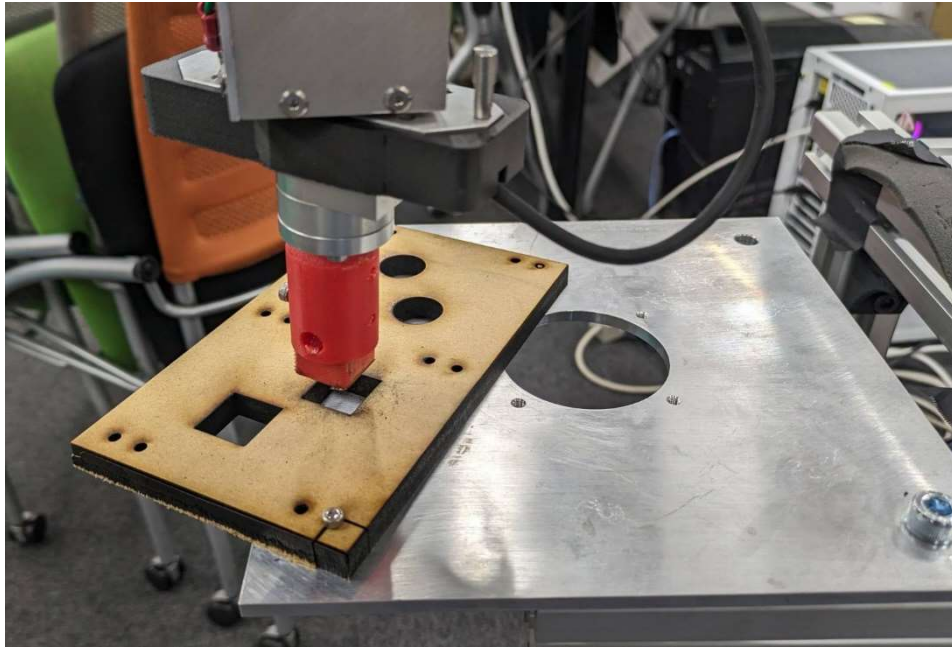


Fig. 12 Peg and work attached to force sensor or robot base

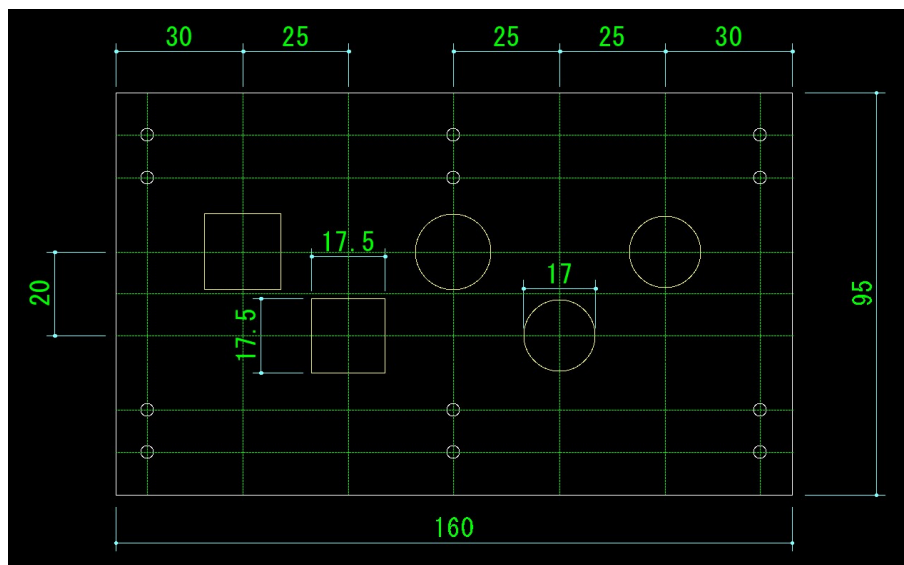


Fig. 13 blueprint for peg insertion attachment

3.2 学習手法について

以下の様な全結合層のネットワークを作成し学習を行った。

レイヤ	入力	出力
入力層		18
全結合層 1	18	512
ReLU	512	512
全結合層 2	512	256
ReLU	256	256
全結合層	256	6

Table .2 Network structure of the imitation learning system

円柱、直方体のペグともに 200 件ずつペグ挿入のサンプル動作を人の手によって教示しながら前述したソフトウェアを使ってロボットの状態（6 関節の角度と角速度、出力トルクの計 12 パラメータ）と力覚センサのセンサ出力（XYZ 方向の力とトルクで 6 パラメータ）を入力とした。

また、データは 1000Hz で取得し、5 ステップごとの平均をとりその値をネットワークへの入力とした。そして出力としてその次の 20 ステップ（50ms 後）の位置を学習するようなデータセットを作成した。

また、これらの値を正規化した後のデータに、平均 0 分散 0.1 の正規分布ノイズを加えたものを水増しデータとして学習させた。

尚、学習における諸パラメータは以下の通りである

- 損失関数：平均二乗誤差関数
- 学習率：0.1
- バッチサイズ：32
- エポック数：50

4. 結果

円柱と直方体のそれぞれのペグ挿入タスクにおいて上記で示したようなモデルでそれぞれ学習を行った. その際のモデルの学習時の損失関数は以下ようになった.

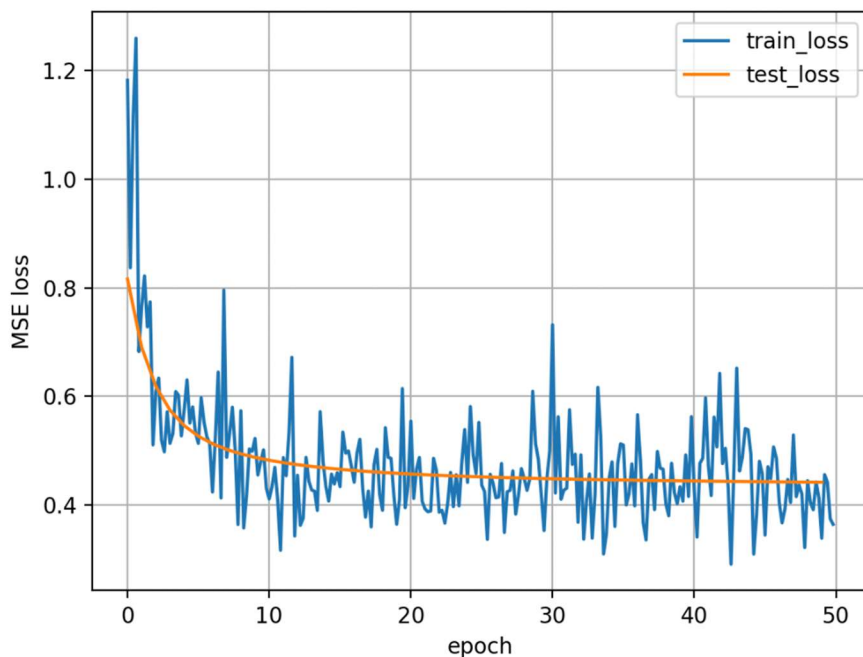


Fig. 12 MSE loss during the imitation learning of insertion behavior for box pegs

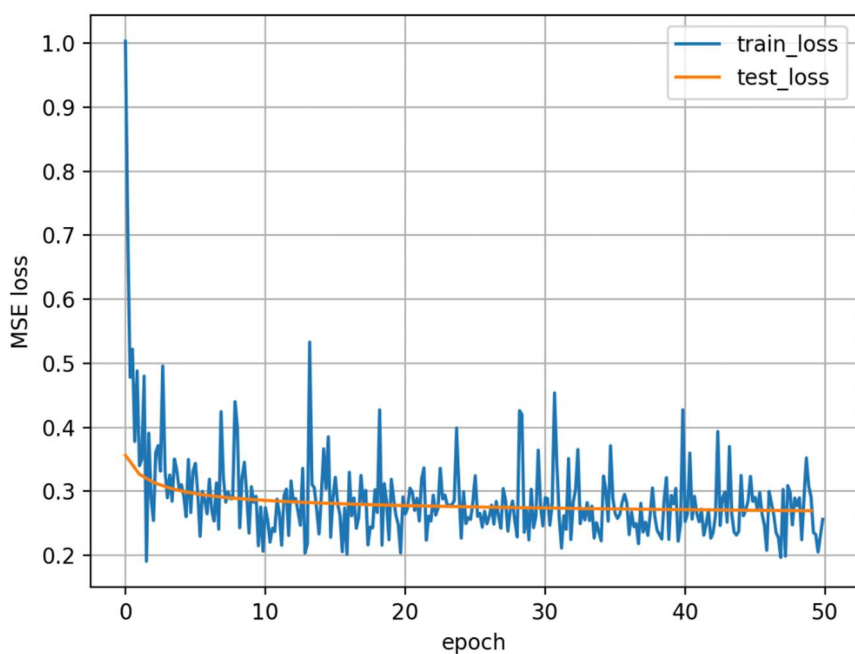


Fig. 13 MSE loss during the imitation learning of insertion behavior for cylinder pegs

上のグラフより、ティーチングした内容において一定の学習効果があり学習の収束がみられる。最終エポックにおける直方体のペグのティーチングデータのロスとテストデータの損失関数の値はそれぞれ 0.39 と 0.46、円柱ペグにおける損失関数の値は 0.27 と 0.29 であった。

5. 結論と考察

本論文では、第二節でダイレクトドライブの多間接ロボットアームにおける、模倣学習を使ったタスク実行手法について議論し、それに対する先行研究やその概要について述べた。それを元に実際の 6 軸ロボットアームを用いて模倣学習のプラットフォームを作成し、それを用いて学習を行った。

このような実験から、ペグ挿入のタスクにおいて深層模倣学習がある程度の有効性を持つ可能性があると考えられる。しかし、学習結果のモデルを用いて実機を動作させるところまでいかなかったため、どれほどの効果を持つのかは今後の課題としてあげられる。

1.7 ティーチング時に発生する外力の問題について

完全な力制御を行うためのデータセットを作るには、ティーチングを行うロボット自体に触れてはならない。ティーチング時に意図せず触れている部分があればそこから余計な力が発生し、関節トルクや外力そのものが計算不可能になるからである。そのような状況を排除するため、当初はバイラテラル制御を用いて学習データを集める予定であった。

しかし、バイラテラル動作をさせること自体が難しく、本論文では直接ロボットアームに触れてティーチングを行うという手法をとった。単純な手法としては、二台のロボットがお互いの関節角を送りあい、相手の関節角を目標値としてフィードバック制御しあうというものがある。

しかし、この手法ではティーチングするアームを動かすために従来の二倍近くの力が必要なことや、制御の不安定性など様々な問題が挙げられる。またはシステムをモデル化し、重力補償制御を行いながら力感覚をフィードバックするという手法も考えられるが、さらに実装が困難であったためこの点も今後の課題として取り上げる。このように、力制御をするにあたっては、ティーチング時に遠隔でロボットアームを操縦するシステムの開発が求められる。

6. 謝辞

本研究は、東京大学大学院情報理工学系研究科知能機械情報学専攻國吉康夫教授、また大村吉幸先生のご指導のもとで行われました。国吉教授と大村先生から毎週のサブグループミーティングやウィークリーミーティングで非常に多くのフィードバックをいただき、非常にお世話になりました。研究ならではの発想の仕方や進め方に対するアドバイスなど、幅広くご指導いただき非常に感謝しています。キムヒョルさん、宮ノ原優斗さん、涇澤諒さんなどは特に、ミーティングや普段の実験中からアドバイスをいただきありがとうございました。

そして、同じく学部四年生の平野聖剛さん、黄子馨さん、椋山天平さん、山田紘大さんから、普段から研究自体の相談などは勿論のこと、研究自体のモチベーション維持しあい、様々なことで相談しあうなど、お互いを高めながら研究できる環境を提供してくださったことに感謝しています。

また、本研究で使用したロボットのもととなったハードウェアは **JBS (Japan Business systems)** 日本ビジネスシステムズ株式会社からご提供受けたものを使用しています。

また、本研究で使われたプログラムの一部は金井嵩幸さんが残してくださったコードを一部使用しています。

最後に、常日頃からお世話になっている家族や友人の皆様に感謝を申し上げて謝辞とさせていただきます。

7.参考文献

- [1. **PomerleauADean.** Alvin: An autonomous land vehicle in a neural network. Technical report, Carnegie MelonD. A. Pomerleau. Alvin: An autonomous land vehicle in a neural network. Technical report, Carnegie Melon.: Univ. Pittsburgh, PA. Artificial Intelligence and Psychology., 1989.
2. **Isaacand Sammut, ClaudeAndrew.** Goal-directed Learning to Fly : MACHINE LEARNING-INTERNATIONAL, 2003.
3. **Stephane RossGordon, Drew BagnellGeoffrey.** A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning. : Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, PMLR 15:627-635, , 2011.
4. **RussellS.** Learning Agents for Uncertain Environments. : Proc. of 16th International Conference on Machine Learning, 278/287, 1998.
5. **Markus WulfmeierOndruska, Ingmar PosnerPeter.** Maximum Entropy Deep Inverse Reinforcement Learning. Oxford Robotics Institute, 2016.
6. **Justin FuLuo, Sergey LevineKatie.** Learning Robust Rewards with Adversarial Inverse Reinforcement Learning. : ICLR, 2018.
7. **ErmonHo and StefanoJonathan.** Generative Adversarial Imitation Learning. 出版地不明 : Advances in Neural Information Processing Systems, vol.29, pp.4565-4573, , 2016.
8. **NielsLander.** Interactive Imitation Learning for Force control: Position And Stiffness Teaching with Interactive Learning. : Delft University of Technology, 2021.
9. **Simon StepputtisBandari, Stefan Schaal, Heni Ben AmorMaryam.** A System for Imitation Learning of Contact-Rich Bimanual Manipulation Policies. 出版地不明 : International Conference on Intelligent Robots and Systems (IROS),, 2022.
10. **JohnsDi Palo and EdwardNorman.** SAFARI: Safe and Active Robot Imitation Learning with Imagination. : 5th Annual Conference on Robot Learning, 2020.
11. **Zhuand Joshi, Abhishek and Stone, Peter and Zhu, YukeYifeng.** VIOLA: Imitation Learning for Vision-Based Manipulation with Object Proposal Priors.: 6th Annual Conference on Robot Learning (CoRL), 2022.
12. **Heecheol KimOhmura, Yasuo KuniyoshiYoshiyuki.** Robot peels banana with goal-conditioned dual-action deep imitation learning. : arXiv, 2022.
13. **Pete FlorenceLynch, Andy Zeng, Oscar Ramirez, Ayzaan Wahid, Laura Downs, Adrian Wong, Johnny Lee, Igor Mordatch, Jonathan ThompsonCorey.** Implicit Behavioral Cloning. 2021.
14. 新井 健生智昭, 橋本 亮一, 中野 栄二矢野. 7自由度人間腕形ダイレクトドライブ・マニピュレータの開発.: 日本ロボット学会誌 pp.27-35, 1987.
15. 長谷川 勉尚士, 高瀬 國克末広. 環境モデルと作業スキルの統合におけるロボット作業システム.: 日本ロボット学会誌 pp.66-74, 1991.

-
16. **SUEHIRO TKTAKASE.** DD マニピュレータ ETA-3 の開発と二次デジタルフィルタによるサーボ剛性の強調. : Proceedings of the 15th International Symposium on Industrial Robots, Vol.1 pp.479-486 , 1985.
17. 新井 健生智昭, 橋本 亮一, 中野 栄二矢野. Development of a direct-drive human-like manipulator. : 日本ロボット学会誌 5 (1), 27-35, 1987.
18. **Design and Control of a Open-Source, Low Cost, 3D Printed Dynamic Quadruped Robot . 2021.**
19. 舘二泰輔, 荒井 裕彦, 西澤 昭一郎, ホセ・フェリペ ペラエス・ポロ榊. カセンサを用いないダイレクト・ドライブ・マニピュレータのインピーダンス制御. : 日本ロボット学会誌 pp.172-184, 1989.
20. **N. Hogan. Impedance Control: An Approach to Manipulation: Part I-Theory,. : Journal of Dynamic Systems, Measurement and Control, vol. 107, pp. 1-7, 1985.**