

# ヒューマノイドロボットのモーション作成支援システムの開発

— MonoMotion: Design your robots' motion without the hassle —

## 1. 背景

ヒューマノイドロボットを個人で製作し、その製作手法を公開している人はとても少ない。また、ヒューマノイドロボットを製作したいが、技術情報の不足から製作を断念する人が多く見受けられる。ヒューマノイドロボット製作のハードルが高い原因としては、第一にロボット本体を製作することが非常に高度な技術を要すること、そして第二に、ヒューマノイドロボットの制御が難しいことが挙げられる。本クリエイターは、独自のヒューマノイドロボットの製作過程をブログや Web ページで公開したり、プログラムや 3D モデル、制御基板の設計情報などを GitHub で公開したりすることで、前者の問題の解決に努めてきた。しかし、後者の問題については未だ解決策がない状況が続いていた。

## 2. 目的

本プロジェクトでは、ロボット製作者にその制御の難しさを感じさせないようにするための取り組みを行った。サーボモータで構成された様々なロボットに対して思い通りの動作を生成するための支援フレームワーク(Reficere, BlendMotion, flom, trainer, servoarray)を作成し、自ら使用すると同時にその取り組みを社会に認知させ実際にロボット制作者を支援するために展示会(Maker Faire Tokyo 2018 及び World Maker Faire New York 2018)への出展等の広報活動をした。

## 3. 開発の内容

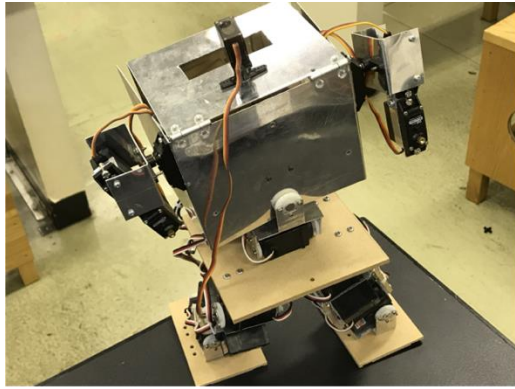
本プロジェクトで開発したフレームワーク“MonoMotion”は大きく分けて以下の 3 つの部分に分割される。括弧内はそれぞれのソフトウェアの名前である。

1. ロボットモデルを作る(Reficere)
2. モーションを作る(BlendMotion, flom)
3. 実機で動かす(trainer, servoarray)

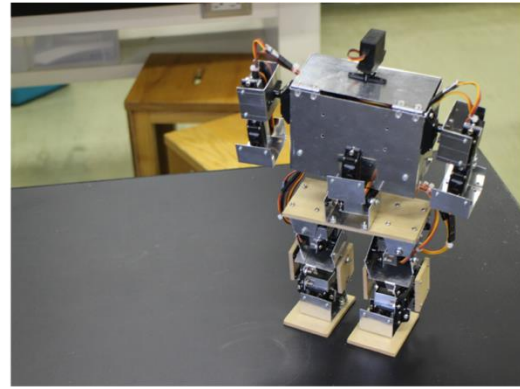
開発したハードウェア(本プロジェクト期間以前の機体を改良したものを含む)は以下の 4 つである。

1. YamaX 4.2
2. YamaX 5.0
3. YamaX 6.0
4. YamaX 7.0

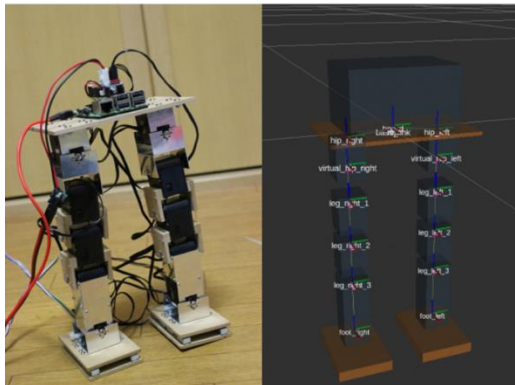
それぞれの外見は図 1 の通りである。YamaX 6.0 と YamaX 7.0 について、写真の右側に作成したロボットモデルのイメージを示す。



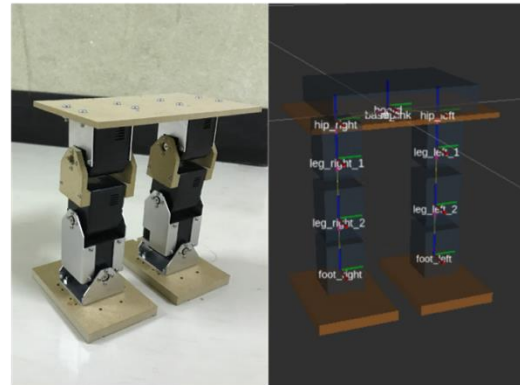
YamaX 4.2



YamaX 5.0



YamaX 6.0



YamaX 7.0

図 1. 製作したロボット

### 3.1. ロボットモデルを作る(Reficere)

本フレームワークでは自作したロボットのモデルとして、その URDF (Unified Robot Description Format) ファイルを使う。しかし、これまでロボットのモデルを作るにあたって必要な、パーツの長さや重さ、重心位置、関節の位置や方向などのパラメータを測定するには、多大な手間がかかっていた。特にロボットのパーツ(リンク)の重量測定は分解の必要があり、リンク内での重心位置の計測は単純には不可能であった。

Reficere はロボットを分解することなく短時間に、各リンクの重量計測、及びリンク内での重心位置の計測をできるようにするソフトウェアである。特徴としてはロードセル(荷重変換器)を使用し、ロボット全体の重心測定と重量測定ができるセンサ(重心センサ)を安価に製作し、システムに活用したことが挙げられる。図 2 に Reficere を使ったパラメータ測定の様子を示す。

Reficere の詳細は以下に記した。

- プロモーションビデオ: <https://www.youtube.com/watch?v=nECBQznYx9g>
- 解説サイト: <https://monomotion.netlify.com/technology/reficere>

### 3.2. モーションを作る(BlendMotion, flom)

Reficere で作成したロボットモデルを用い、視覚的にロボットのモーションを作成するために BlendMotion というソフトウェアを開発した(図 3)。

BlendMotion は 3DCG ソフトウェアの Blender のアドオンとして作成した。また、モーションを作成した後にそれを保存する手段として、ファイルフォーマット兼ライブラリの flom を制作した。



図 2. Reficere を使った測定の様子

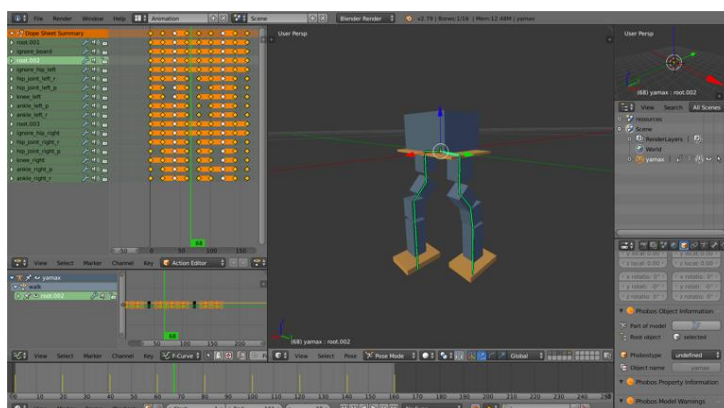


図 3. BlendMotion でモーションを作成している様子

### 3.3. 実機で動かす (trainer, servoarray)

BlendMotion で作成したモーションは物理シミュレートをして作成したものではないため、それをそのまま実機で動かしても思い通りになるとは限らない。そこでシミュレーションで学習を行い、モーションの調整することになる。これを DFO (Derivative-free optimization) により自動で行うソフトウェア、trainer を開発した(図 4)。

また、学習によって修正されたモーションを実機で動かすためのソフトウェア、flom-runner を開発した(図 5)。モーションは flom に出力されているため、これは flom のモーションを実機で再生するソフトウェアと言える。さらに、ロボット間の違いを吸収するために、サーボモータ抽象化ライブラリ servoarray を開発し、flom-runner で使用している。

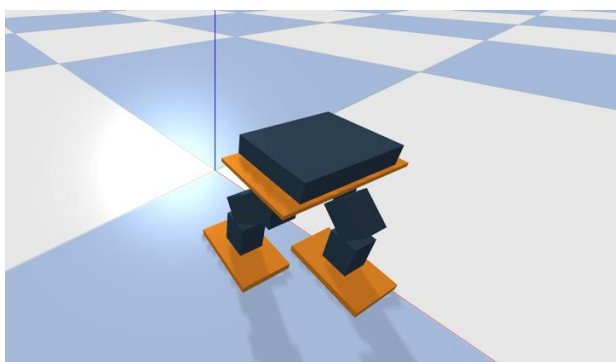


図 4. シミュレーションでの学習

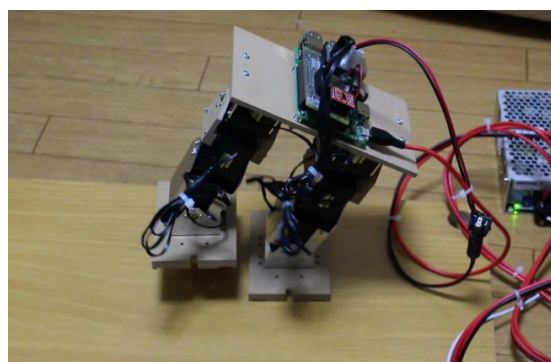


図 5. 実機での再生

これらを使用することで、BlendMotion で作ったモーションが学習によって最適化され、実機で思い通りに動かすという目標が達成されたことになる。

## 4. 従来の技術(または機能)との相違

MonoMotion は従来のロボットフレームワークと比較して、以下のような特徴がある。

- ホビーユーザを想定
  - 比較的安価な手作りのロボットに対しても理論上使用可能である。
  - ユーザが直感的にモーションを作成できる。
- オープンソース
- フレームワークを適用可能なロボットが非限定的
  - 自作の複数のロボットに適用可能であることが実証されている。
- フレームワークの各コンポーネントが疎結合
  - 各種ライブラリ (Reficere, flom, servoarray など) は、MonoMotion に限らず広い用途に使用できる。

## 5. 期待される効果

本フレームワークにより、ホビーユーザ層に存在していたロボット制御への障壁を緩和できたと考えられる。そして、先述の Maker Faire などでの広報活動の結果、ホビーユーザ層に、ヒューマノイドロボットを作る意欲はあるものの作り方や動かし方の情報不足により作ることが困難な人が多数いることが判明した。MonoMotion はホビークラスにおいて、ロボット開発の初心者がロボットを製作することへの支援、及びこれまでもロボットを作ってきた人たちの制御に関わる工程の簡略化が期待される。さらに、これまでホビーロボット開発界隈が閉鎖的であった背景として、使用する技術のハードウェアへの依存が大きいことを考えると、ハードウェアへの依存性の低い汎用化された MonoMotion を使うことによって、ホビーロボット界隈での情報交換が活発になることも期待される。

## 6. 普及(または活用)の見通し

MonoMotion はホビークラスを念頭に制作されたオープンソースのフレームワークであるため、具体的な利用者数は予測不可能である。そのため、展示会の出展数や本クリエイターの参加経験等から利用規模を概算する。

本クリエイターも出展した Maker Faire Tokyo 2018 では、ロボティクスのカテゴリでの出展は 102 件あった。Maker Faire Tokyo が日本最大規模の展示会であることと、趣味として製作したロボットの出展が比較的多いことを考慮すると、日本全体では数百人規模のユーザが見込まれると考えられる。

また、ロボット教室等での使用がある場合、利用規模はさらに大きくなると考えられる。

## 7. クリエータ名(所属)

小川 広水(東京都立小石川中等教育学校)

山名 琢翔(東京都立小石川中等教育学校)

(参考)関連 URL

- <https://monomotion.netlify.com>
- <https://github.com/MonoMotion>
- <https://github.com/Y-modify/YamaX>