

# ROS を用いたモジュラーホームロボットのための オフライン動作生成システムの開発

根井 学, 畠山 直登, 関戸 佐知, 山本 明弘, 佐藤 大祐, 金宮 好和 (東京都市大学)

## Development of an Offline Motion Generation System for a Modular Home Robot Utilizing ROS

Manabu Nei, Naoto Hatakeyama, Sachi Sekido, Akihiro Yamamoto,  
Daisuke Sato, and Yoshikazu Kanamiya (Tokyo City University)

This work describes the design of an offline teaching and playback system for a modular home robot, based on ROS. The system is characterized by its flexibility, which is needed for accommodating various designs obtained by combining different modules.

### 1 緒言

近年, 床面清掃のような単一の家事支援機能を持つロボットが普及している. しかし, これらのロボットは単一の機能しか有していないため, ユーザの様々な家事代行ニーズに答えることができない.

そこで我々は独自の開発コンセプトを基にモジュラーホームロボットを開発している. モジュラーホームロボットは複数のモジュールによって構成されるロボットである [1][2]. このロボットのモジュール構成はユーザの要求する家事作業, 実行する作業によって動的に変更する. これにより様々な家事作業を実現する. このためには, ロボットへ動作教示, 動作のシミュレーションが必要不可欠であり, 本稿では ROS を用いてモジュラーホームロボットのモデリングを行い, オフライン動作生成システムを構築する.

### 2 モジュラーホームロボット

Fig. 1 に示す我々が構築したモジュラーホームロボットのプロトタイプは, グリッパ, アーム, ホイール, 深度センサ, 音声認識, 視覚センサモジュールで構成されており, アームモジュールは七つのジョイントモジュールから構成される. アームモジュールはベースと先端で対称な構成となっており, 両端部が同じ機構である. そのため, アームの固定具を壁や天井に取り付けることにより, アームが独立で移動することが可能である.

### 3 モジュラーホームロボットの表現

現在, 我々は ROS[3] を用いてソフトウェアの開発を行っている. ROS はロボットのプログラムを記述するためのソフトウェアフレームワークで, 世界中の開発者が共有するオープンソースのソフトウェアである. ROS では RViz というプラグインベースの可視化ツールが公開されており, これを用いてシミュレーション環境を構築する.

#### 3.1 モジュールの表現

ROS においてロボットのモデルは Unified Robot Description Format (URDF) ファイルで扱うことが可能である. URDF はロボットのリンクおよびジョイント情報が Extensible Markup Language (XML) 形式

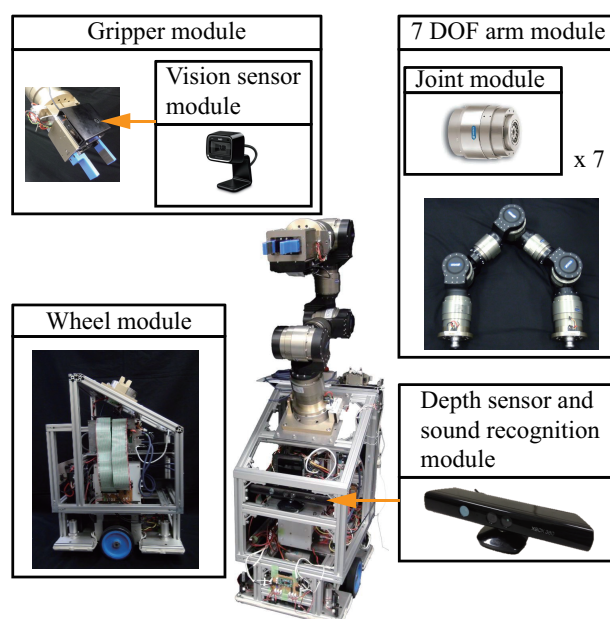


Fig. 1 Modular home robot.

で記述される. これにより, 単一のロボットがモデリング可能となっている.

しかし, 我々が扱うロボットはモジュールの構成が動的に変化し, 複数のロボットが存在する状況がある. これを ROS のロボットモデルでモジュラーホームロボットを表現することは困難である. そこで, モジュールごとにロボットモデルとし, それを統括することでモジュラーホームロボットを表現する.

#### 3.2 tfを用いたモジュール構成の動的切り替え

複数のモジュールのロボットモデルを RViz 内で統括するために tf[4] と呼ばれる ROS のパッケージを用いる. tf とは座標系のツリー構造を構築し, これを管理するためのパッケージである. このツリー構造は座標系間の親子関係とその相対位置情報により構成される. このとき, 子座標系は親座標系を複数保有することができない. また, tf は指定した任意の二つの座標系間の相対位置の取得, ある点群データの座標変換などの座標変換ライブラリとしての機能も有する.

ロボットモデルの座標系を tf で管理するためには joint\_state\_publisher, robot\_state\_publisher と呼ばれ



Fig. 2 Interactive Marker.

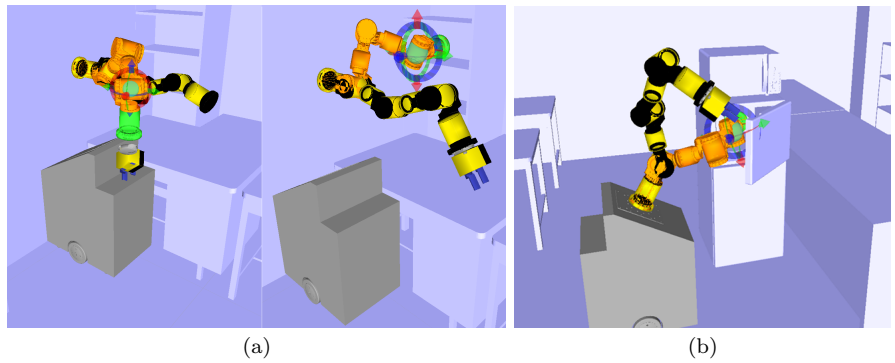


Fig. 3 Teaching motions for each task: (a) the arm reattaches the gripper and (b) approaches the refrigerator as a mobile manipulator.

るパッケージを用いる．我々はこれをモデルごとに用いることでモデルを tf の管理下に置き，モデルの親子関係，相対位置をコントロールする．例えば，Fig. 1 に示すモバイルマニピュレータのような構成とする場合，世界座標系から順にホイール，アーム，グリッパモジュールの親子関係となる．

これらを用いて，モジュールの親子関係とその相対位置をコントロールすることにより，モジュラーホームロボットの特徴の一つであるモジュール構成の切り替えをシミュレータで扱うことが可能となる．

#### 4 オフライン動作生成システムの構築

ROS では MoveIt! と呼ばれる運動計画のためのフレームワークが存在する．MoveIt! は経路計画法，衝突判定，順逆運動学計算のライブラリなどが統合されており，アーム，モバイルベースなどの障害物回避を含めた動作の生成が可能である．経路計画法は一般化座標系内の経路を取得する手法であり，MoveIt! ではロボットモデルの関節空間に対して行っている．

経路計画法は搬送作業などによく用いられるが，我々は様々な家事作業を行うことを考えている．家事作業においては物体の操作もする必要があるため，デカルト空間で動作指令を生成する必要がある．ロボットの動作をデカルト空間で生成することが可能なソフトウェアは ROS に存在しない．

これらのことから我々はデカルト空間上で動作を生成するシステムを構築した．

##### 4.1 Interactive Marker (IM) を用いた

###### 視覚的な動作の教示

ROS では IM と呼ばれるユーザインタフェースライブラリを提供している．IM の外観を Fig. 2 に示す．これは今回開発した動作生成システムにおいても使用しており，Fig. 3 に示すアームモジュールに付いている．IM を用いることでマウス操作により画面上のロボットに対し，視覚的に手先位置を指示することが可能となる．

##### 4.2 アームモジュールの動作生成

アームモジュールの動作生成の流れを以下に示す．

1. 任意の位置に IM を移動
2. IM の位置を登録
3. 1~2 を繰り返し，制御点のリストを取得

4. スプライン関数を用いて滑らかな手先の軌道を生成

5. ヤコビの擬似逆行列を用いて関節角度軌道を算出

ここでスプライン関数はパラメトリックスプラインを用いているため，手先軌道が多価関数となることが可能である．また，このスプライン関数では制御点を通過するようにした．

これらの処理により，デカルト空間上に任意の手先の軌道を生成し，変換することでロボットへの動作指令値を生成する．

##### 4.3 RViz のプラグインの開発

RViz はプラグインと呼ばれる動的ライブラリを追加/削除することで，機能をカスタマイズすることが可能である．そこで，今後はモジュールごとに特化した動作生成インタフェースのプラグインを開発する必要がある．これにより，モジュールの構成に応じてモジュール情報の可視化，教示インタフェースをプラグインとして切り替えることで汎用的な動作生成システムを構築する．

#### 5 結言

ROS を用いてモジュラーホームロボットのためのシミュレーション環境，動作生成システムを開発した．しかし，アームモジュールの動作生成しか開発できていないため，ホイールモジュールなどのアクチュエータを持つモジュールの動作生成機能を拡張する．

#### 参考文献

- [1] 宮澤友規，上野修平，久米修平，金宮好和，佐藤大祐，“多機能生活支援ホームロボットを実現するためのミニマリストモジュラーシステムコンセプトの提案”，ROBOMECH, 1A1-O06, 2007.
- [2] Takahito Tsuchiya, Yohei Shiraki, Sachi Sekido, Akihiro Yamamoto, Daisuke Sato and Dragomir N. Nenchev, “Modular home robot system based on the MMM concept-design instance with detachable symmetric arm module,” Advanced Intelligent Mechatronics (AIM), 2013 IEEE/ASME International Conference on, 2013, pp. 280–285.
- [3] M. Quigley B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, A. Ng, “ROS: an open-source Robot Operating System,” ICRA workshop on open source software, 2009, Vol. 3, No. 3.2.
- [4] Tully Foote, “tf: The transform library,” Technologies for Practical Robot Applications (TePRA), 2013 IEEE International Conference on, 2013, pp. 1–6.