

# モデル駆動型開発ツールを使用した AR マーカーに基づく自律航行クアッドコプターの位置制御

柴原 大賀<sup>†</sup> 佐藤 未来子<sup>†</sup> 大江 信宏<sup>†</sup> 久住 憲嗣<sup>‡</sup>

東海大学情報通信学部組込みソフトウェア工学科<sup>†</sup> 九州大学工学部電気情報工学科<sup>‡</sup>

## 1 はじめに

近年、マイコンやセンサの製造技術の進歩に伴い、手ごろな価格で小型の移動ロボットやドローンが入手可能になっている。教育現場にもロボットが導入されており、筆者も enPiT2[1] という文部科学省の実践教育カリキュラムにおいてロボット教材を用いた Project Based Learning (PBL) 実践教育を受講した[2]。本講義では、教材ロボットの使い方や制御方法、UML を用いたモデルベースのソフトウェア開発手法などを学び、その後、情報処理学会 SIGEMB 主催の ESS ロボットチャレンジ 2019 (以下、ESSRC) で、AR マーカーに基づく自律航行クアッドコプター制御の精度を競う競技に取り組んだ[2]。PBL では、UML を用いたソフトウェア開発経験者が、UML でモデル設計を行い、その情報を共有しながらプログラム開発を分担した。しかし、クアッドコプターの位置を精度よく制御できず、正しい位置に着陸することができなかった[3]。

ESSRC 開催後、筆者は情報処理学会 SIGEMB 共催の APRIS ロボットチャレンジ 2019 (以下、APRISRC) [4] で使われたモデル駆動開発ツール (以後 MDD ツール) を用いて、あらためてクアッドコプターの航行制御のプログラムを設計し直した。その結果、ESSRC の際に課題となったクアッドコプターの航行位置の制御、読み取った AR マーカーの ID 情報を別ロボットの制御に応用するための通知機能などを実現できた。

本稿では、PBL を実践する際に、UML 設計を参照してコードを開発する方法と、MDD ツールを用いて動作検証しながら開発する方法の利点欠点などを実体験に基づいて考察する。

## 2 課題コースと規定動作

図 1 に本研究で扱った課題コースと規定動作を示す。また、図 2 に本研究のシステム構成を示す。クアッドコプター (DJI 社製 Tello) をルータとして制御 PC と Wi-Fi 接続し、Tello に対して制御コマンドを送信して制御する。Tello には

Autonomous Quadcopter Position Control Based on AR-markers Using the Model-Driven Development Tool.

<sup>†</sup>Taiga Shibahara, Mikiko Sato, Nobuhiro Ohe, Tokai University.

<sup>‡</sup>Kenji Hisazumi, Kyushu University.

カメラが備わっており、画像を制御 PC へ送信することができる。また、Tello を制御する制御 PC1 と、別ロボットの制御を想定して AR マーカーの ID 情報を受信するための PC2 は、Tello とは別の LAN へ接続する。

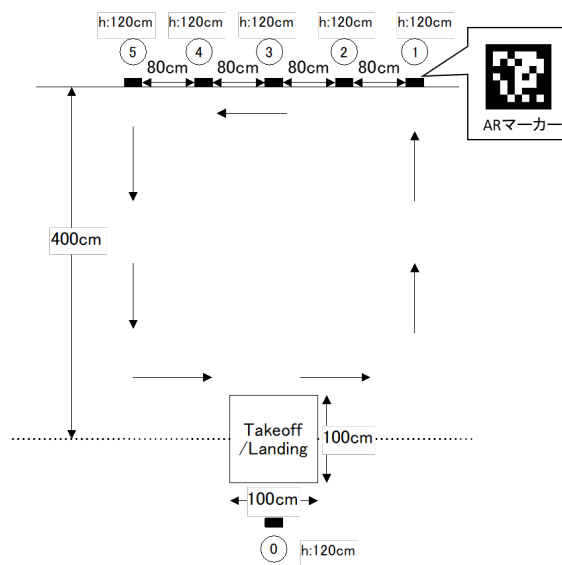


図 1 本研究で設定した課題コースの概略 (クアッドコプターは、離着陸地点から AR マーカー①の高さ 120cm まで飛び立ち、4m 先に設置されている 5 つの AR マーカー①～⑤をカメラでとらえ、AR マーカーから ID 情報を読み取る。その後、元の離着陸地点へ戻る。読み取った 5 つの AR マーカーの ID 情報を、別のシステムへ通知する。)

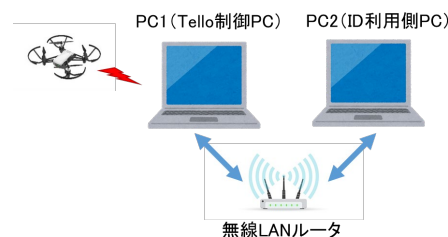


図 2 本研究のシステム構成図

## 3 モデル駆動開発ツールによる設計開発

本研究では、MDD ツールとして BridgePoint を使用した。本 BridgePoint には、Tello と制御 PC 間の通信、AR マーカーの認識、Tello の基本航行などを扱うための基本クラスが、APRISRC 向けに用意されており、プログラム開発者はそのクラスを応用して、ステートマシン図で Tello の

制御プログラムを設計・検証できる。

図 3 に、2 章で示した規定動作を BridgePoint で設計・検証した状態マシン図の一部と、AR マーカーを用いた航行制御のコード例を示す。本設計開発では、Tello の状態遷移を UML で記述していき、各状態での Tello の動きをプログラムした。例えば、`7.AR\_Control\_repeat` の状態で AR マーカーの情報を受け取ると、次の状態 `6.AR\_control` へ遷移し、`6.AR\_control` ではソースコード例に示すように、Tello から見える AR マーカーの相対位置情報 `param` を用いて、Tello の航行位置を調整している。AR マーカーの正面で ID を検出するまで `6` と `7` を繰り返して姿勢制御し、ID 情報をイベントとして、次の状態 `8.AR\_Catch` へ遷移する。`8` では PC1 から PC2 への ID 通知を実施した後、次の AR マーカーへ移動するための処理 `9.timer` へ遷移し、再度 `7` に移行して姿勢制御を行っている。

#### 4 開発方法に対する考察

ESSRC では、クラス図、状態遷移図、シーケンス図でモデル設計を行い、その情報を共有して Python で Tello の動作をコーディングした。Tello の航行制御の流れをこれらの UML で示すことにより、プログラム開発者はテストケースの生成、コード開発を行ったが、自律航行課題における、AR マーカーの情報で行動制御すると

いう詳細設計をモデルでは伝えられなかった。

BridgePoint を用いて開発した場合、プログラム開発者にとって Tello の動作シーケンスを整理しやすく、設計で不足している点、機能追加が必要な箇所などを判断しやすいという利点があった。一方で、MDD ツールの使い方を十分に把握できなかったため、処理をモジュール化できず、状態数が 16 個と多くなり、見通しが悪くなった。より洗練されたコード開発のためには、MDD ツールの特徴を習得し、モデル化およびコード開発を多く実践する必要があると感じた。

#### 5 まとめ

本稿では、UML からプログラムを生成するモデル駆動開発ツールを用いて実践した AR マーカーの位置に基づく航行制御の実践内容と、体験した開発方法の違いを考察した。AR マーカーの ID 通知処理などを利用して、別ロボットとの連携などの複雑なシナリオ実践にも取り組みたい。

#### 参考文献

- [1] enPiT, <http://www.enpit.jp> (Jan. 6. 2020 閲覧)
- [2] 大江信宏他, “ESS ロボットチャレンジ 2019”, 情処, 組込みシステムシンポジウム 2019 論文集, pp.96--102 (2019).
- [3] 秋山慈音他, “ESS ロボットチャレンジ 2019 チームポスター概要—チーム開発におけるモデルを用いた情報共有の実践—”, 組込みシステムシンポジウム 2019 論文集, pp.103--104 (2019).
- [4] APRIS2019 Robot Challenge, [http://www.sigemb.jp/APRIS/2019/?page\\_id=35](http://www.sigemb.jp/APRIS/2019/?page_id=35) (Jan. 6. 2020 閲覧)

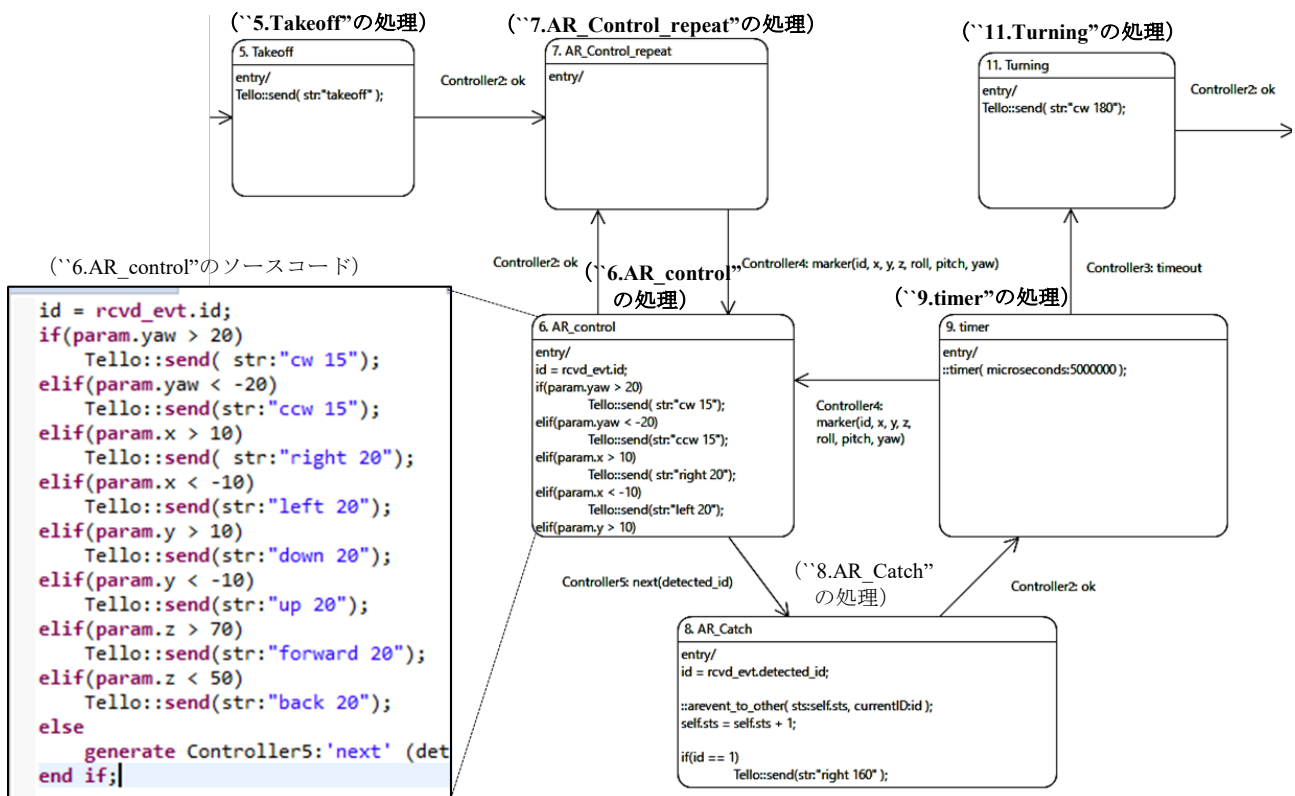


図 3 本研究のシステム構成図