

地域性の強い非管理型 IoT アプリケーションのための エージェントフレームワークの提案

富井 駿介^{†1} 伊藤 優樹^{†2} 阿部 亨^{†3,†2} 菅沼 拓夫^{†3,†2}

^{†1} 東北大学工学部電気情報物理工学科 ^{†2} 東北大学大学院情報科学研究科

^{†3} 東北大学サイバーサイエンスセンター

1 はじめに

センサや無線通信技術の進展に伴い、IoT アプリケーションの普及が進んでいる。本研究では IoT アプリケーションを 2 つの観点から分類する。一つは、地域性の強弱、すなわち、センサデータが生成される場所と利用される場所との地域依存性の度合いの観点である。もう一つは、管理性の強弱、すなわち、センサ、センサデバイス、サービス、データ等の管理性の度合いの観点である。これらの観点に基づいた IoT アプリケーションの分類を図 1 に示す。横軸が地域性、縦軸が管理性の軸となっており、楕円内にはそれぞれの領域での代表的な IoT アプリケーションの例を示している。

上記のような IoT アプリケーションの分類において、それぞれの領域でのアプリケーションの設計・開発に適するアーキテクチャやプラットフォームの構成が存在するが、管理性が弱く地域性が強い右下の領域については、研究が不十分である。このような領域向けのプラットフォームとして、自律分散型のマルチエージェントシステムが適すると考えられるが、センサデバイスなどの限られた計算資源のデバイス等を組み込むことが可能なエージェントフレームワークは未整備である。

本研究では、管理性が弱く地域性が強い IoT アプリケーションの開発の効率化・高度化を目的とする。この領域に適したプラットフォームとして、自律分散型のマルチエージェントシステムを提案する。具体的には、アンライセンス系 LPWA の P2P 通信と、エージェント間協調プロトコルを適用し、限られた計算資源上でも動作するエージェントフレームワークを提案する。

本発表では、当該エージェントフレームワークの概要と、設計・実装の基本方針について述べる。

2 関連研究

本章では、既存のエージェントフレームワークと、地域性の強い IoT アプリケーション向けプラットフォームに関する既存研究を示す。

既存のエージェントフレームワークとして

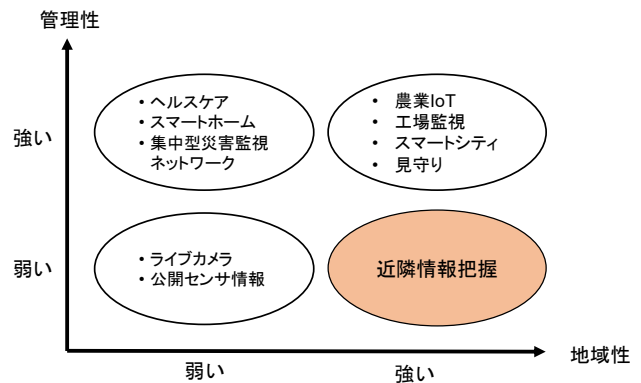


図 1: IoT アプリケーションの分類

Python Agent Development(PADE) [1] が存在する。PADE は、Python で構築されたマルチエージェントプラットフォームである。FIPA 規格に準拠しているため、他フレームワークのエージェントとも協調動作が可能である。論文 [1] では、PADE を利用した開発事例として電力網の自動普及システムの例が示されている。

地域性の強い IoT アプリケーション向けプラットフォームとして、IFoT [2] がある。この研究では、既存のクラウド一極集中型アーキテクチャのプラットフォームでの、通信帯域の逼迫やクラウド型の計算・蓄積資源の枯渇などの問題の根本的な解決を目指し、地域で生成された IoT データ流を地域で処理して利活用する、地産地処をコンセプトとした情報処理基盤として IFoT を提案している。

同様のアプリケーション向けプラットフォームとして、I-BoT [3] がある。この研究では、エッジコンピューティングにおいて、ノートパソコンなどの個人用デバイスを Unmanaged Edge Device(UED) として利用することを目的としている。UED の計算能力の不均一性、UED の利用可能性の不確実性、UED を共有する複数のタスク間の干渉などの問題を解決するオーケストレーションスキームとして I-BoT を提案している。

これらの既存研究は、地域性の強い IoT アプリケーションに関する研究ではあるが、クラウド型やエッジ型のアーキテクチャで実装されており、管理性が弱い場合については考慮されていない。よって、本研究が着目する管理性が弱く地域性が強い IoT アプリケーションには不向きである。

An agent framework for localized and unmanaged IoT applications
Shunsuke TOMII^{†1}, Yuki ITO^{†2}, Toru ABE^{†3,†2}, and Takuo SUGANUMA^{†3,†2}

^{†1}Department of Electrical, Information and Physics Engineering, Tohoku University

^{†2}Graduate School of Information Sciences, Tohoku University

^{†3}Cybercience Center, Tohoku University

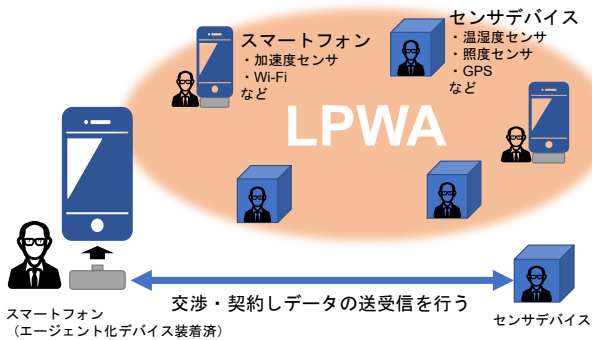


図 2: 提案の概要図

3 提案

本研究では、管理性が弱く地域性が強い IoT アプリケーションのためのプラットフォームを提案する。こうした領域のプラットフォームの要件としては、まず、利用可能な非管理のセンサ、デバイス、サービス等を動的に組み合わせて、アドホックにアプリケーションを実現する機能が必要である。また、地域内において、非管理の構成要素間で、必要十分な通信が行える通信機能が必要である。

そこで本研究では、上記の要件を満たす、マルチエージェント型のプラットフォームを提案する。まず、アプリケーションの非管理の構成要素を自律分散したエージェントとみなし、エージェント協調プロトコルによって、センサやサービス間で高度な交渉をすることでシステムを動的に組み上げる、エージェントフレームワークを構成する。また、エージェント間の通信には、基地局やサービス登録なしで広域無線通信が可能なアンライセン系 LPWA の P2P 通信を利用する。提案の概要を図 2 に示す。スマートフォンなどにエージェント化デバイスと呼ばれるハードウェアを装着することで、LPWA 通信範囲内のエージェントと通信を行う。エージェントはコントラクトネットプロトコル等を用いて契約交渉を行い、近隣エージェントから適切なセンサを選択しデータの送受信を行う。

4 実装・実験

エージェント化デバイスには、マイコン (ESP32) を使い、LoRa モジュール (ES920LR2) を搭載する。スマートフォンなどのデバイスとは USB で接続し、シリアル通信を行う。マイコン上では、実装環境として MicroPython を使用し、PADE を基にしたエージェントフレームワークを実装する。エージェント化デバイス上には限られた計算資源でも動作する軽量のエージェントを実装する。PADE では TCP 通信と、Python ライブラリである Twisted を用いた並行処理を行っているが、本実装では、LPWA 通信と、MicroPython のライブラリを用いた並行処理を実装する。また、エージェント間協調プロトコルとして、FIPA 規格に準拠したコントラクトネットプロトコルを実装し、既存フレームワークとの互換性を持たせる。

```
[agent_initiator](20230105 10:50:47) PROPOSE message received
[agent_initiator](20230105 10:50:47) PROPOSE message received
[agent_initiator](20230105 10:50:47) Analyzing proposals...
[agent_initiator](20230105 10:50:47) Analyzing proposal 1
[agent_initiator](20230105 10:50:47) Power Offered: 15.0
[agent_initiator](20230105 10:50:47) Analyzing proposal 2
[agent_initiator](20230105 10:50:47) Power Offered: 10.0
[agent_initiator](20230105 10:50:47) The best proposal was: 15.0 VA
[agent_initiator](20230105 10:50:47) Sending REJECT_PROPOSAL answers...
[agent_initiator](20230105 10:50:47) Sending ACCEPT_PROPOSAL answer...
[agent_initiator](20230105 10:50:47) INFORM message received
```

(a) agent initiator

```
[agent_participant_A](20230105 10:50:47) CFP message received
[agent_participant_A](20230105 10:50:47) REJECT_PROPOSAL message received
```

(b) agent participant A

```
[agent_participant_B](20230105 10:50:47) CFP message received
[agent_participant_B](20230105 10:50:47) ACCEPT_PROPOSE message received
```

(c) agent participant B

図 3: コントラクトネットプロトコルの動作

エージェント化デバイス上での実装に先立ち、予備実装として、仮想環境の MicroPython 上の PADE にコントラクトネットプロトコルを実装し、動作確認実験を実施した。なお、仮想環境上の実装では、TCP を用いた通信を行っている。図 3 に実験結果を示す。図 3 より、コントラクトネットプロトコルによる協調動作が実現できていることが確認できた。

今後は、エージェント化デバイス上への実装を進め、LPWA 通信を用いた動作実験を実施していく。

5 おわりに

管理性が弱く地域性が強い IoT アプリケーションの開発の効率化・高度化を目的として、LPWA 通信とマルチエージェントの協調プロトコルを適用したエージェントフレームワークを提案した。今後は、エージェント化デバイス上での実装を進め、エージェント化デバイスとセンサデバイスを用いた動作確認実験を実施していく。

参考文献

- [1] Melo, L. S., Sampaio, R. F., Leão, R. P. S., Barroso, G. C. and Bezerra, J. R.: Python-based multi-agent platform for application on power grids, *International Transactions on Electrical Energy Systems*, Vol. 29, No. 6, p. e12012 (2019).
- [2] 中村優吾他: IoT データ流を実時間で分散処理するための IoT デバイス向け共通ミドルウェアの設計と評価 (知的環境とセンサネットワーク), 信学技報, Vol. 116, No. 509, pp. 247-254 (2017).
- [3] Suryavansh, S., Bothra, C., Kim, K. T., Chiang, M., Peng, C. and Bagchi, S.: I-BOT: Interference-Based Orchestration of Tasks for Dynamic Unmanaged Edge Computing, *CoRR*, Vol. abs/2011.05925 (online), available from (<https://arxiv.org/abs/2011.05925>) (2020).