

IFoT の考えに基づいた IoT データ流利用アプリケーションの開発

坂口 僚¹ 藤田 茂¹ 今野 将²

概要：現在の IoT プラットフォームは、遠隔地のクラウドサーバにデータが集中するような構成でなされている。この構成では、取り扱われる IoT デバイスの台数増加に比例してトラフィックが一極集中する。今後数千億～数兆台規模に IoT デバイスタ数が増加した場合、クラウド側で膨大なデータを集めきれず、処理しきれない恐れがある。このような状況を考慮するなかで、データ自身を一箇所に蓄積せず、流通させ、即時的な利用をすべきであるという情報流技術の考え方がある。情報流の利活用には、IFoT (Information Flow of Things) という IoT プラットフォームが提言されており、各 IoT デバイスを計算資源として、クラウドに依存しない構成として示されている。本稿では IFoT の考え方に基づき、管理機器としての計算機、25 台の Raspberry Pi、異種かつ複数台のセンサによる、IoT データ流を利用したアプリケーションを開発するための準備として、実験を行ったので報告する。

1. 概要

現在の IoT プラットフォームは、図 1 のように遠隔地のクラウドサーバにデータが集中するような構成で示される。この構成では、取り扱われる IoT デバイスの台数増加

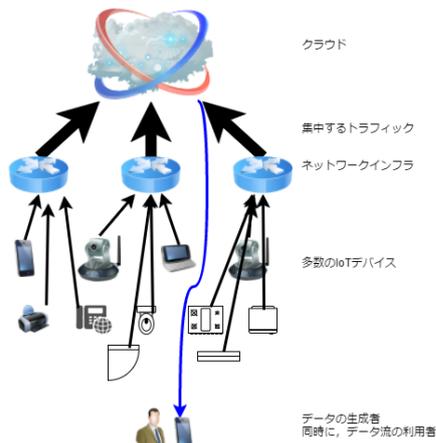


図 1 クラウドにデータが集中する構成の IoT システム

に比例してトラフィックが一極集中する。今後数千億～数兆台規模に IoT デバイスタ数が増加した場合、クラウド側で膨大なデータを集めきれず、処理しきれない恐れがある。

このような状況を考慮するなかで、データ自身は蓄積せ

¹ 千葉工業大学情報科学研究科情報科学専攻, Graduate School of Computer Science, Chiba Institute of Technology

² 千葉工業大学先進工学部知能メディア工学科, Department of Advanced Media, Chiba Institute of Technology

ず、流通させ、即時的な利用をすべきであるという情報流技術の考え方がある。情報流の利活用には、IFoT (Information Flow of Things) という IoT プラットフォームが提言されており [1], 各 IoT デバイスを計算資源として図 2 のようにクラウドに依存しない構成で示される。更に、IFoT ミドルウェアとして実装、研究が進められている [2]. そのな

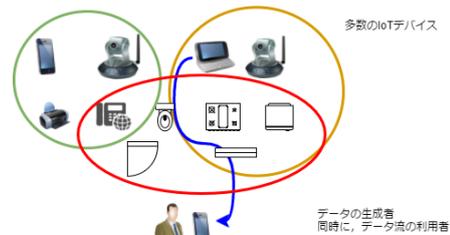


図 2 ローカルでデータが流通する構成の IoT システム

かで、我々が実際に IFoT の考えに基づいて温度計測アプリケーションを構築し、9 台の Raspberry Pi による実装、評価を行った [3]. この実験を発展させ、本研究では IFoT の考え方に基づき、管理ノードとしての計算機、25 台の Raspberry Pi、異種かつ複数台のセンサによる、IoT データ流を利用したアプリケーションを開発した。この開発のなかで、我々は 25 台の IoT デバイス間でのセンサデータの、一極集中による通信帯域への影響の有無の検証を行った。以下、2 章で想定するアプリケーションの開発例について述べ、3 章では本研究の基になっている IFoT に関連した研究を示す。4 章では通信帯域への有無の検証の中で

の実験の内容と結果を示し、5章では4章での結果を踏まえて、まとめとして今後の展望を述べる。

2. アプリケーションの開発例

本研究では、アプリケーションの開発に関して、具体的な実装環境を想定する必要がある。本章では、実装開発の検討も視野に入れた上で、具体例としてのユースケースを示す。想定する環境下で、開発が考えられるアプリケーションの例を4種類、以下に挙げる。

2.1 空調の指標獲得

冷暖房が取り付けられた室内環境を考える。何らかの指標に基づき、制御がなければ、空調制御下でも室内の温度が均質に保てておらず、暑い空間と寒い空間ができてしまう。このような室内環境に人間が左右されてしまい、快適に過ごせないことが想定される。そこで、温湿度センサや照度センサから得られたセンサデータを用いて、センサノード側の計算資源によって、PMV(快適指数)といった指標を獲得する。センサデータの即時的な分析により、また即時的な室内環境の向上につなげられることが期待される。

2.2 講義室内の環境状況把握

大学において、講義が行われる講義室内の環境を考える。受講生が快適に講義を受けられるようにしていくにあたっては、空調制御の上、さらに講義を行う上で、教壇に立った講師の声の聞こえ方や板書の見え方を考慮する必要がある。この時、受講生1人1人の席に対しての音の聞こえ方を、マイクによる音声データを収集し、板書への見え方をカメラによる動画データを集めて、受講生の座席からの見え方や聞こえ方を把握する。この収集結果から、板書の配置の変更や音響設備の改善を行う指標を獲得できる。

2.3 詳細な鉄道運行情報の獲得

詳細な情報の求められる鉄道の運行状況を考える。センサデータを用いる場合、運転車両に加速度センサやGPSからのセンサデータを用いて、車両の現在位置を詳細に確認することが可能である。そのようにしてより詳細な運行状況を提示することで、利用者側から見れば質の高い運行情報を提供することができる。

2.4 学生食堂の混雑度状況把握

大学の設備として利用される、学生食堂のような環境を考える。学生食堂では座席数が制限され、利用する学生の数によっては席の空き状況も目視では把握しづらい。そこで、カメラからの動画データによる動体検知や、人の動きを解析すること、空席の把握により、食堂内の混雑度状況を把握することができる。さらに、席毎の空き状況を食堂入り口に設置することにより、特に混雑時には空席数の

具合を確認することが可能となる。

3. 関連研究

本章では、本研究での基となっている、IFoTの概念に関連した情報流の考え方や、実装研究、またIFoTの研究自身からの関連研究としての技術について示す。

3.1 情報流技術の考え方

センサデータは、実世界の状況を示すことが可能であり、従来は、ストレージに蓄積して分析をかける活用方法である。しかし、センサデータは複雑で雑多であり、膨大な量が存在する。文献[4]では、このような状況を考慮した時、データは蓄積するのではなく、高次の利用のためにリアルタイムに流通させるべきであることを主張している。このような手法をとることで、蓄積では間に合わない、IoTデータ流の活用を促進できる。本研究は、このような情報流の考え方にも基づいていく。

3.2 IFoT ミドルウェア

IFoTの概念に基づいた実装研究として、IFoTミドルウェアの開発、実装が進められている。IFoTミドルウェアは、図3のような動作環境上で動作する。一連の処理の

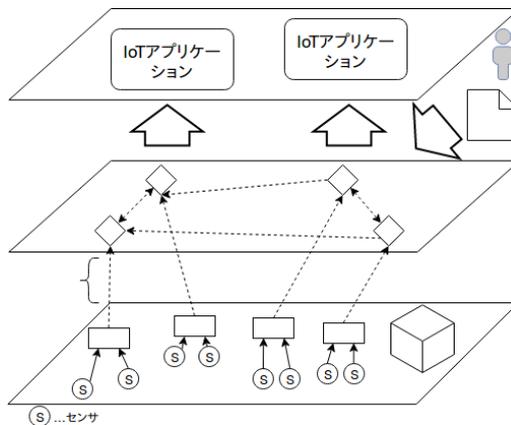


図3 IFoTミドルウェアの動作環境(文献[2]を基に作成)

流れは、図3に従って、以下の通りとなっている。

- (1) アプリケーション管理者側でレシピを作成、読込
- (2) IFoTマスタノードが、読込まれたレシピによってIFoTスレーブノードに処理タスクを分担
- (3) センサノードからの生データより、データ流の発生する位置情報やデータの種類を記したメタデータを付加した一次IoTデータ流を作成
- (4) 受け取った処理タスクの内容に従ってIFoTスレーブノード側が、目的に応じた高次IoTデータ流を生成、流通

このような動作でセンサデータが処理され、要求に応じてより高価値なIoTデータ流が生成される仕組みとなって

いる。

先行する研究報告では、クラウドへデータを収集することなく、小規模な IFoT ミドルウェアのプロトタイプとして 6 台の IoT デバイスと管理マネージャ上に実装して、ダミーのセンサデータを用いることで動作検証を行っている [5]。また、動画データに注目し、予め IoT デバイス間で動画を解析しておくことで、クラウドに送信するデータ量を減らし、クラウドへの依存を減らすことを目的としたシステムの実装を行っている [6]。この研究では、実際の動画データに着目して、実装が進められている。さらに、顔画像の IoT データ流に対して並列分散処理を行い分析をかけた際の処理動作の並列化効率が得られたことを実証している [2]。他にも、地域情報流の高次利用に着目した上で GAIFoT(Geographically distributed Ambient interface utilizing local Information Flow of Things) の提案 [7] がされている。

本研究では、一種類のアプリケーションに特化して、センサデータを目的に応じた、他の仕組みによって作られた IoT データ流を利用していきよう、開発を進めていく。

4. 評価実験

本章では、25 台の IoT デバイスを用いたなかで、トラフィック集中下でのセンサデータの流通の検証を目的として、実験を行った結果、考察を示す。

4.1 実験下の環境

本実験では、各デバイスがセンサデータを生成し、データを送受信し合える環境を想定してネットワークを構築する。各デバイス間と管理機器はアドホックネットワークによる無線通信で相互に接続する。この時、1 台のデバイスをサーバ側に見立てて、他 24 台をクライアント側に見立てて、ネットワークを、図 4 のような構成にして実験を行う。IoT デバイスには、Raspberry Pi を 8 台、Raspberry

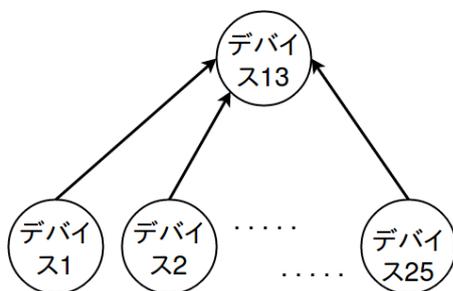


図 4 各デバイスのネットワーク

Pi2 を 1 台、Raspberry Pi3 を 16 台用いる。各デバイスと管理機器の仕様は表 1 に示す。

4.2 実験内容と結果

本実験では、UDP 通信による 32byte のデータを、仮想

表 1 各デバイスと管理機器の仕様

デバイス 1 から 8	
使用機器	Raspberry Pi
OS	Raspbian
CPU	ARM1176JZF-S 700MHz
メモリ	512MB
デバイス 9	
使用機器	Raspberry Pi2
OS	Raspbian
CPU	ARM Cortex-A7 900MHz
メモリ	1GB
デバイス 10 から 25	
使用機器	Raspberry Pi3
OS	Raspbian
CPU	ARM Cortex-A53 1.2GHz
メモリ	1GB
管理機器	
使用機器	Think pad x100e
OS	Ubuntu 12.04
CPU	AMD Athlon(tm) Neo Processor MV-40 1.6GHz
メモリ	2GB

的なセンサデータとして扱い、クライアント側としての役割を持つ各デバイスが生成し、サーバ側としての役割を持つデバイスに送信する。サーバ側のデバイス側が、各デバイスからのセンサデータを全て受信した時、各デバイスでのセンサデータの平均値を求める。このように、かかった一連の処理手順による処理時間がどのような結果になるかを観測する。図 5 に処理の流れと、対象とする処理時間を示す。サーバ側の役割を担ったデバイス (デバイス 13) か

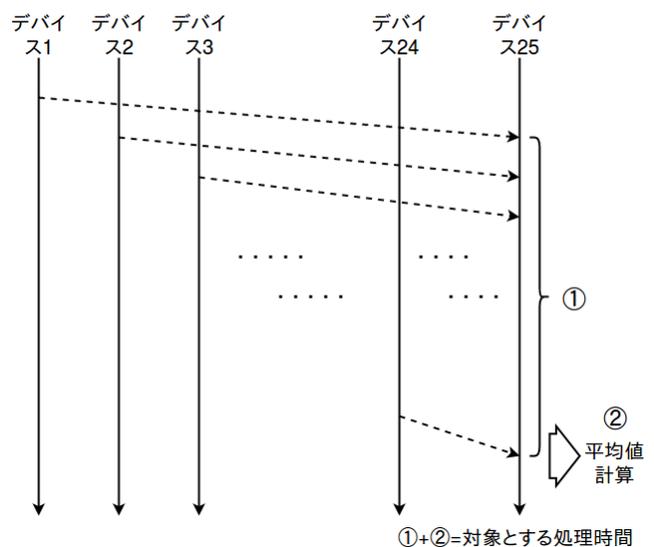


図 5 各デバイス毎の送受信処理の流れ

ら、他の各デバイスに ping を 10 回送信した際の RTT の結果を表 2 に、一連の処理を 50 回試行した内の処理時間の最小、平均、最大の結果を表 3 に示す。

表 2 各デバイスの ping による RTT

デバイス番号	最小 [ms]	平均 [ms]	最大 [ms]	平均偏差
1	4.264	9.322	26.887	7.148
2	4.586	11.565	29.553	8.166
3	4.297	6.325	15.819	3.342
4	4.454	8.944	32.291	8.189
5	2.125	7.445	13.314	3.370
6	4.230	6.397	10.964	2.575
7	3.920	5.625	14.127	2.907
8	4.399	24.626	174.517	50.171
9	4.624	7.586	18.432	4.194
10	5.759	19.846	39.686	10.890
11	6.745	10.472	15.406	3.121
12	7.541	11.681	19.552	3.653
13	0.020	0.073	0.097	0.023
14	7.293	14.863	31.320	7.211
15	7.403	10.841	15.324	2.733
16	7.493	19.093	83.068	21.979
17	7.519	23.311	113.180	30.412
18	2.343	25.677	134.959	37.439
19	1.239	21.618	117.507	32.572
20	6.067	19.821	54.535	14.792
21	3.517	10.556	17.234	4.443
22	7.797	14.515	40.496	9.441
23	7.255	27.581	169.278	47.509
24	7.555	22.890	115.019	32.231
25	7.677	32.097	196.442	55.418

表 3 一連の処理時間結果 (32Byte データ)

最小 [s]	平均 [s]	最大 [s]
0.507	1.873	5.255

4.3 考察

32Byte のデータ量での送受信処理の範囲では、24 台のデバイスの通信の際に順次受信処理を行えば、平均値計算の処理が、表 3 に示す程度の処理時間で実行可能であることが分かった。

5. 終わりに

今回、本研究では、25 台の IoT デバイス間での、センサデータの一極集中による通信帯域への影響の有無の検証を行った。今後は、屋内または屋外での実験を通じて、アプリケーションの開発を進めていく。

参考文献

- [1] Yasumoto, K., Yamaguchi, H. and Shigeno, H. Survey of Real-time Processing Technologies of IoT Data Streams, *Journal of Information Processing*, Vol. 24, No. 2, pp. 195–202 (online), DOI: 10.2197/ipsjip.24.195 (2016).
- [2] 中村優吾, 水本旭洋, 諏訪博彦, 荒川豊, 山口弘純, 安本慶一: 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, 研究報告ユビキタスコンピューティングシステム (UBI), Vol. 2017, No. 40, pp. 1–8 (2017).
- [3] 坂口僚, 藤田茂: IFoT の考えに基づいた温度計測アプリケーションの Raspberry Pi への実装と評価, 研究報告マルチメディア通信と分散処理 (DPS), Vol. 2017, No. 17, pp. 1–6 (2017).
- [4] 安本慶一, 山口弘純: モバイル時代のサービスを支える技術: 5. 多数のデータストリームを実時間で融合・編纂し利活用するための次世代「情報流」技術-情報流キュレーション基盤実現に向けた課題抽出と取り組み-, 情報処理, Vol. 55, No. 11, pp. 1281–1287 (2014).
- [5] 中村優吾, 諏訪博彦, 荒川豊, 山口弘純, 安本慶一, 他: 多様な IoT データストリームをクラウドレスで分散処理するミドルウェアの設計, 研究報告高度交通システムとスマートコミュニティ (ITS), Vol. 2015, No. 22, pp. 1–8 (2015).
- [6] 中村優吾, 諏訪博彦, 荒川豊, 山口弘純, 安本慶一, 他: ローカル環境での効果的な動画画像解析を実現する分散処理システムの提案, 研究報告高度交通システムとスマートコミュニティ (ITS), Vol. 2016, No. 32, pp. 1–7 (2016).
- [7] 中村優吾, 松田裕貴, 荒川周造, 金平卓也, 安本慶一: 地域情報流の高次利用を促す地域分散アンビエントインタフェース (GAIFoT) の提案, 研究報告システム・アーキテクチャ (ARC), Vol. 2015, No. 5, pp. 1–4 (2015).