

## IoT データ向け分散データベースの評価用環境の構築の検討 Environment of Distributed Database Systems for IoT Data

根上聡太<sup>†</sup>武本充治<sup>†</sup>爰川知宏<sup>†</sup>

Souta Negami Miciharu Takemoto Tomohiro Kokogawa

東京国際工科大学 工科学部 情報工学科<sup>†</sup>International Professional University of Technology in Tokyo  
Department of Information Engineering, Faculty of Engineering

### 1. はじめに

近年 IoT 製品の普及が高まり、インターネットに接続されているデバイスは約 500 億台接続されている。そのため、複数の IoT デバイスから測定したデータをインターネットに介してデータベースに送るデータ量が膨大になり、データベースの保管や品質の低下が伴う課題がある。そうした IoT デバイス増加に伴い、IoT デバイスから測定したセンサーデータを蓄積する IoT データ向けデータベースの実現に向けた研究が様々な形で提案されている。しかし、その提案されてきているものは実際に測定した IoT データをもとに評価している物は少なく、乱数を用いて評価を行っている。そのため、IoT データ向けデータベースの実現のための評価機構は不十分ではないのかと考えた。

我々は IoT データを用いた Skip Graph[1]のアルゴリズム性能評価するシミュレーション・可視化システムである optisn を提案する。本稿では、IoT データ向けデータベース実現のための、IoT データを用いた Skip Graph シミュレーション・可視化システムである optisn の実現方法について述べる。

### 2. IoT データの蓄積方式

IoT システムにおいて製品に組み込むセンサーなどの IoT デバイスによって IoT データが測定されている。IoT データをインターネットに介してクラウドあるいはオンプレミスのデータベースサーバーに集められている。そのような IoT データは、データを測定するシステムが稼働し続ければ、センサーよりデータは測定され、データベースサーバーにデータが蓄積され続ける。測定した IoT データを多く蓄積するため、IoT データ向けデータベースは様々な提案がされている。その中で、IoT データ向けに提案されている内の 1 つである InfluxDB[2]は、時刻情報を主キーとした時系列データベースの 1 つである。センサーデータやログデータなどの時刻情報を不随したデータを扱うのに長けている特徴を有している。また、IoT データ向けデータベースの実現に Skip Graph[1]を用いる研究もある。研究としては Skip Graph の単一ピアに 1 つのキーではなく、複数のキーを保持可能とさせる Multi-key Skip Graph[3]、ノードが担当する複数のキーを範囲キーとし、範囲検索を可能とさせる Range-Key Skip Graph[4]があり、Skip Graph を用いる分散管理データベースの研究において、Skip Graph のアルゴリ

ズムの性能評価は乱数を用いたデータを用いて行われているものが多数である[3][4]。つまり、IoT データを用いて評価をされているものは少ない。

そこで我々は、実際に IoT デバイスから測定した IoT データを用いた Skip Graph のアルゴリズム性能評価を行う。その IoT データは、実際の建物内に多数の IoT デバイスを設置し、CO2 濃度、温度、湿度を測定し、Influx データベースに蓄積する[2]ものである。

### 3. sgsim の概要

複数条件、範囲検索が可能である特長を具備することが可能[3][4]な Skip Graph[1]を用いて、IoT デバイスより測定したデータを活用し、シミュレーションを行う。Skip Graph シミュレーターは、sgsim[5]を用いる。sgsim は Skip Graph のノード間通信をシミュレーションするだけでなく、その結果を可視化する。また、sgsim は、git hub 上で MIT ライセンスにより公開がされており、Python3.9 で実装がされている。

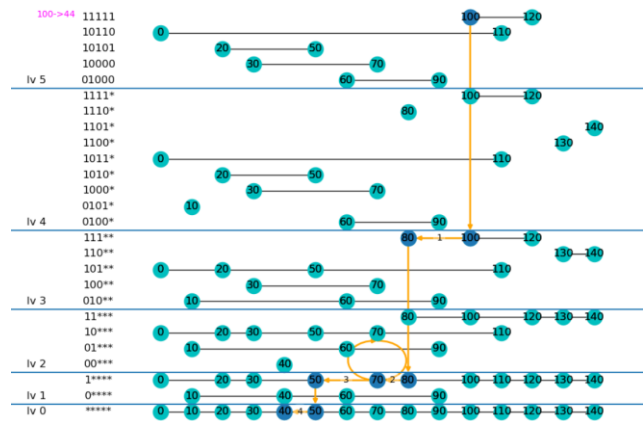


図 1 : sgsim による Skip Graph のシミュレーションと可視化

### 4. Resilient Campus Platform の概要

Skip Graph を評価するために用いる IoT データを測定し、収集する必要がある。そのため我々は、実際の建物内に IoT デバイスを設置し、IoT データを「CO2 センサー等」から測定し、データを蓄積するシステム InfluxDB を用いて実装をした[6][7]。

実際の建物内にある多数のセンサーからデータを集め、電光掲示板やグラフから建物内の状況を可視化するシステムである。SCD30 で測定した CO2 濃度、温度、湿度を M5 Stick C Plus で処理し、その後 Wi-Fi 経由で MQTT データとしてオンプレミスのサーバーに送信している。そのサーバー内で受信した MQTT データを InfluxDB に格納している。Wi-Fi 経由でオンプレミスサーバーに送信している実際の MQTT データを図 2 に示す。また、InfluxDB に蓄積された CO2 濃度データを Visual Studio Code のターミナル上で表示させたものを図 3 に示す。

```
{ "sensor": "C868", "temperature": 24.5, "humidity": 57.3, "co2": 559 }
{ "sensor": "D8B9", "temperature": 24.6, "humidity": 62.9, "co2": 642 }
{ "sensor": "1CEE", "temperature": 28.3, "humidity": 56.6, "co2": 1074 }
{ "sensor": "A0C4", "temperature": 26.1, "humidity": 59.6, "co2": 873 }
```

図 2 : 実際の MQTT データ

```
23 query_api = infclient.query_api()
24
25 while True:
26     try:
27         df = query_api.query_data_frame('
28             from(bucket:"m5kkiput")
29             > range(start:-30s)
30             > filter(fn: (r) => r["_field"] == "co2")
31             > pivot(rowKey:["time"], columnKey:["_field"], valueColumn:"_value")
32             > keep(columns:["_time", "_measurement", "co2"])
33         ')
34         print(df.to_string())
35
36         time.sleep(30)
37
```

問題	出力	ターミナル	デバッグコンソール		
17	_result	17	2023-06-13 11:33:42.461714+00:00	F089	661.0
18	_result	18	2023-06-13 11:33:35.529562+00:00	F4B6	502.0
	result	table	line	measurement	co2
0	_result	0	2023-06-13 11:34:14.085412+00:00	0887	578.0
1	_result	1	2023-06-13 11:34:00.798240+00:00	141D	573.0
2	_result	2	2023-06-13 11:34:10.458299+00:00	181E	665.0

図 3 : 測定されているデータ (CO2、温度、湿度)

## 5. optisn の提案

上記の章より、実際に測定し、収集した IoT データを用いたシミュレーション・可視化システムとして、我々は optisn を検討している。optisn は、Resilient Campus Platform を用いて Wi-Fi 経由より MQTT データとしてサーバーに受信する CO2 濃度、温度、湿度のデータを取り出し、sgsim のシミュレーターに用いる。

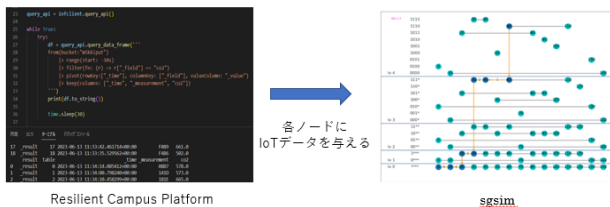


図 4 : optisn

Resilient Campus Platform のデータは Linux 上の InfluxDB に蓄積されており、Flux 言語を用いて指定した時間に受信されたデータを取り出す。取り出したデータに基づいて Skip Graph オーバーレイネットワークを構築する。そのデータへの利用リクエストをクエリとして Skip Graph トポロジ上でルーティングパスをレンダリングし、グラフの可視化を行う。シミュレーターに Resilient Campus Platform のデ

ータを与えた optisn の動作は、Windows PC 上の Python3.9 による動作を想定している。

optisn を実装することにより、IoT データ向けデータベースの実現のための IoT データを用いた Skip Graph のアルゴリズム性能評価をすることができる。つまり、IoT データ向けデータベースの実現のための評価機構を十分にすることができると思われる。

## 6. まとめ

本稿では、IoT データを用いた Skip Graph シミュレーション・可視化システムである optisn について述べた。今後は、Resilient Campus Platform の測定する IoT データの種類を増やすこと、シミュレーションを行うノードを複数のコンピュータで行う拡張を検討し、optisn の実装、Skip Graph などのデータ蓄積機構の評価を行う予定である。

## 参考文献

- [1] Aspnes, J. and Shah, G., "Skip Graphs," 14<sup>th</sup> Annual ACM-SIAM Symposium on Discrete Algorithms, pp.384-393 (2003).
- [2] InfluxDB, <https://www.influxdata.com/>
- [3] 小西ほか, "単一ノードに複数キーを保持可能とする Skip graph 拡張," 情報処理学会論文誌, Vol.49, No.9, pp.3223-3233 (2008)
- [4] 石ほか, "Range-Key Skip Graph による範囲検索可能な大規模分散キーバリューストアの実現," 情報処理学会論文誌, Vol.53, No.7, pp.1850-1862 (2012)
- [5] sgsim: A Simple Skip Graph Simulator and Visualizer (<https://github.com/abelab/sgsim>)
- [6] 爰川ほか, "サステイナブルな教育環境の構築に向けて ~ Resilient Campus Platform ~," 東京国際工科大学紀要, pp.33-44 (2022)
- [7] Kokogawa, T., "Trial of Building a Resilient Face-To-Face Classroom Based on CO2-Based Risk Awareness," International Conference on Information Technology in Disaster Risk Reduction, pp.95-106 (2021)