

高トラフィック時のエッジアーキテクチャの レイテンシ評価方法の提案と評価

M2019SE001 早川 剛生
指導教員 青山 幹雄

1 研究背景

近年コネクテッドカーの普及が進んでいる。このコネクテッドカーにおけるサービスの1つに緊急通報システムがあるが、通報に人手を介しているため通報に時間がかかる。今後の緊急通報システムは通報と同時にドライブレコーダ動画を直接送信する必要があると考える。通報時間の短縮と出動判断の精度向上が期待できる。しかし、通報の最も近くの基地局にデータ送信が集中し、複数の通報が発生した場合は通報のレイテンシが大幅に増加する。しかし、コネクテッドカーはそれぞれ異なるネットワークを使用するため緊急通報システム全体のレイテンシ測定が困難である。

2 課題

本研究では、以下の2点を研究課題とする。

- (1) 複数のネットワークを考慮した通報集中時のレイテンシ測定方法の提案
- (2) 提案方法をあおり運転ユースケースに適用し、その妥当性の評価

3 関連研究

3.1 Publish/Subscribe(Pub/Sub)アーキテクチャ

Pub/Sub アーキテクチャは非同期メッセージ配信アーキテクチャ[4]である。図1に示すように、Broker 経由でデータ送信を行うことで、Publisher と Subscriber との非同期通信を可能にしている。Pub/Sub アーキテクチャを用いた通信プロトコルに、ISO 標準の MQTT がある[6]。

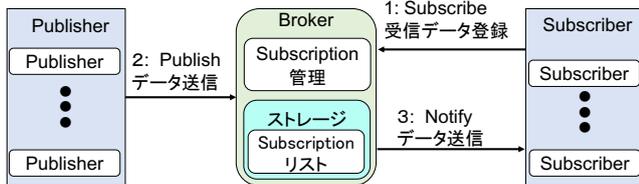


図1 Publish/Subscribe アーキテクチャ

3.2 二重エッジアーキテクチャ

コネクテッドカーではエッジコンピューティング[1][9]技術が活用される。コネクテッドカーとクラウドとの間にエッジサーバを配置し、クラウドとエッジの分散処理、コネクテッドカーとエッジ間の低レイテンシ通信を実現する。

このエッジを活用した二重エッジアーキテクチャ[10]が提案されている。コネクテッドカー内部に車載エッジ、外部に車外エッジを配置することでエッジ間のメッセージ配信処理を分散する。

3.3 緊急通報システムアーキテクチャ

緊急通報システムとは、コネクテッドカーが通報を行うサービスである。図2に示すように、従来の緊急通報シ

テム[7]では、通報車(コネクテッドカー)が接続機関へ接続し、状況を説明した後に救援機関へ通報を行うため、通報時間に課題が残る。先行研究[8]では、通報時間短縮のため車外エッジ(基地局)を介して救援機関や通報現場付近の緊急車両(パトカー)へ位置情報等をまとめた緊急メッセージを直接送信している。

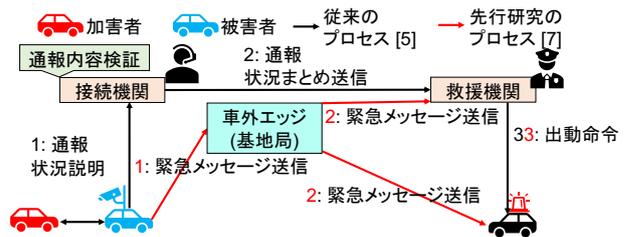


図2 緊急通報システム

3.4 エッジアーキテクチャ評価方法

エッジ間のレイテンシに着目した、プロトタイプ環境によるアーキテクチャ評価方法[5]が提案されている。図3に示す通り、エッジ間で2種類のPub/Sub通信を行い、そのレイテンシの差を測定することで、二重エッジアーキテクチャで適用されているMQTT BridgeというBroker間メッセージ共有機能の性能を評価している。

Pub/Sub アーキテクチャは非同期なアーキテクチャであるが、メッセージ送信区間に区切り、エッジ間でメッセージを往復させることでレイテンシの測定を可能にしている。

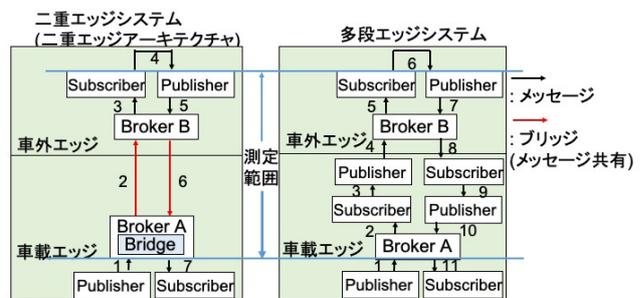


図3 エッジアーキテクチャ評価方法

4 アプローチ

本研究では車外エッジの負荷に着目し、テスト環境下でのエッジアーキテクチャのレイテンシ評価方法を提案する。テスト環境下で車外エッジに通報が集中した場合の通報時間の増加を測定することを目的とする。

通報車から車外エッジまでのレイテンシと、Brokerの処理時間でレイテンシを評価する。図4にアプローチを示す。

通報車疑似ハードウェアは複数の通報車からの通報(緊急メッセージの送信)を疑似し、車外エッジ疑似ハードウェアは車外エッジを疑似する。Brokerの処理時間を測定するため車外エッジ疑似ハードウェアに測定用のSubscriberを配置し、通報車管理プログラムを用いて複数の通報車を生

成する。通報車を、緊急メッセージを送信する Publisher と定義する。生成した複数の通報車から送信を行うことで、レイテンシと処理時間の測定を可能にする。複数のネットワークの疑似にアドホックネットワークを使用する。ルータ等の中継機が必要なく、レイテンシ増加要因が含まれにくいためである。

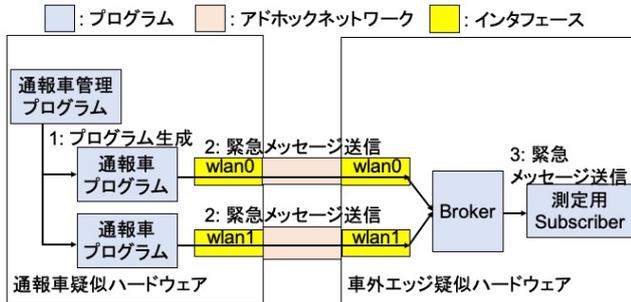


図 4 測定アプローチ

5 提案測定方法

5.1 測定条件

- (1) レイテンシを測定するため、NTP を用いて通報車疑似ハードウェアと車外エッジ疑似ハードウェアの時刻同期を行う。
- (2) 送信データ量の条件: 1, 2 台のコネクテッドカーから通報が発生すると仮定する。使用する通信プロトコルは MQTT とする。1 台から送信される緊急メッセージは、シナリオ 1 では 2.5MB/s(調査した中でのドライブレコーダ動画サイズの平均データ量)、シナリオ 2 では 5MB/s(調査した中でのドライブレコーダ動画サイズの最大データ量)送信すると仮定する。現在 MQTT を用いてドライブレコーダ動画を送信する方法が確立されていないため、MQTT の小量データを軽量に送信できる特徴を利用し、ドライブレコーダ動画を分割し、1 秒間に緊急メッセージを複数回で送信する。
- (3) 受信可能な間隔の条件: この間隔より短くメッセージを送信するとレイテンシが増大していくと考えられる。本研究では、Publisher が送信する間隔 > Broker が受信可能な間隔である必要があると仮定する。これを確かめるため、Publisher が送信する間隔 > Broker が受信可能な間隔となるシナリオ 1 と Publisher が送信する間隔 < Broker が受信可能な間隔となるシナリオ 2 の二つを測定し、比較を行う。緊急メッセージサイズを 0.5MB と定義し、以下の通りに緊急メッセージ送信間隔を変化させ測定を行う。

表 1 送信間隔

シナリオ	シナリオ 1	シナリオ 2
送信間隔[秒]	0.2	0.1

5.2 レイテンシと処理時間の定義

レイテンシ L を表 2 と式(1)のように定義する。通報車が緊急メッセージを Publish し、車外エッジの測定用 Subscriber が受信した時刻までを L とする。

$$L = T2 - T1(\text{ms}) \quad (1)$$

表 2 式(1)の詳細

変数名	
T1	緊急メッセージの送信開始時刻
T2	測定用 Subscriber の受信終了時刻

また、処理時間 $L2$ を表 3 と式(2)のように定義する。緊急メッセージを Broker が受信し、測定用 Subscriber へ送信開始するまでの時間を $L2$ とする。

$$L2 = T4 - T3(\text{ms}) \quad (2)$$

表 3 式(2)の詳細

変数名	
T3	Broker の受信終了時刻
T4	Broker の送信開始時刻

5.3 測定範囲

図 5 に測定範囲を示す。レイテンシ測定範囲は通報車と、車外エッジ内部の測定用 Subscriber の間、処理時間測定範囲は Broker が緊急メッセージを受信してから送信するまでの間とする。

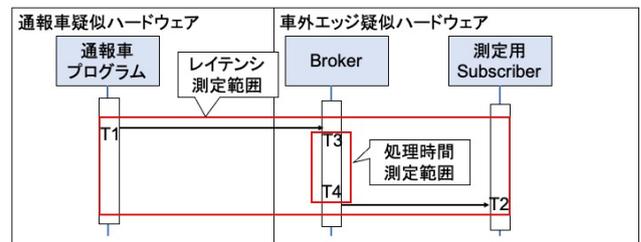


図 5 測定範囲

5.4 測定方法

通報車疑似ハードウェアに測定用の動画データを配置し、通報車に読み込ませる。送信間隔ごとに緊急メッセージを Broker 経由で測定用 Subscriber へデータを送信することでレイテンシ L と処理時間 $L2$ を測定する。通報車はシナリオ 1 とシナリオ 2 のそれぞれの条件で 100 回送信を繰り返す。送信間隔は sleep 関数で設定する。100 回送信したレイテンシ L と処理時間 $L2$ の平均と標準偏差を、通報車が 1 台の場合と 2 台の場合で比較し、評価を行う。通報車が 2 台の場合は、同時刻に通報を開始する。

CPU 使用率を車外エッジ疑似ハードウェアで計測し、異常値が発生する状況になっていないことを確認する。緊急メッセージを通報車が送信し始めてから、測定用 Subscriber が全て受信するまで 1 秒ごとに CPU 使用率を計測する。

6 プロトタイプ

6.1 プロトタイプ実装の目的

以下の目的でプロトタイプを実装する。

- (1) 提案測定方法の妥当性を確認
実際に測定を行い、車外エッジでの、輻輳による影響を測定できるかを確認する。
- (2) 輻輳発生頻度の調査
輻輳の発生頻度を測定し、車外エッジでの輻輳による影響がどの程度であるかを調査する。

6.2 実装環境

プロトタイプの実装環境を表 4 に、実装に使用するソフトウェアの情報を表 5 に示す。Broker(Eclipse Mosquitto[2])

はデータ送受信時刻のログ出力が秒単位のみに対応しており、小数点以下の秒数は出力されない。ログ出力部分のプログラムを修正し、関数を用いて送受信の時刻を配列に格納させ、測定終了後にファイルに書き込みさせることで、ミリ秒単位での処理時間の測定を可能にした。

表 4 実装環境

構成要素	通報車疑似ハードウェア	車外エッジ疑似ハードウェア	NTP サーバ
ハードウェア	Raspberry Pi 4 B	Raspberry Pi 4 B	Raspberry Pi 3 B
OS	Raspbian 10	Raspbian 10	Raspbian 9.9
プロセッサ	Coretex-A72, 1.5Ghz	Coretex-A72, 1.5Ghz	Coretex-A53, 1.2Ghz
メモリ	4G	4G	1G
無線 LAN アダプタ	WN-G300UA	WN-G300UA	

表 5 ソフトウェア情報

構成要素	通報車疑似ハードウェア	車外エッジ疑似ハードウェア	NTP サーバ
Eclipse Mosquitto[2]		1.6.12	
Eclipse Paho[3]	1.4.0	1.4.0	
Python	3.6.5	3.6.5	

6.3 プロトタイプ実装環境構成

実装環境の構成を図 6 に示す。測定前に NTP サーバで通報車疑似ハードウェアと車外エッジ疑似ハードウェアの時刻同期を行う。処理時間は Broker が緊急メッセージを受信した時と送信した時にその時刻を配列に格納させ、指定回数緊急メッセージを測定用 Subscriber へ送信後、ファイルに時刻を書き込むことで処理時間の測定を可能にする。

ラズベリーパイ 4 を使用した理由は、2.4GHz と 5GHz 周波数の両方を使用できるからである。これにより、実験環境に合わせてネットワーク同士が干渉しない周波数を広く選択できる。wlan0 インタフェースを使用するネットワーク 1 では 5,182MHz を、wlan1 インタフェースを使用するネットワーク 2 では 2,452MHz を使用する。

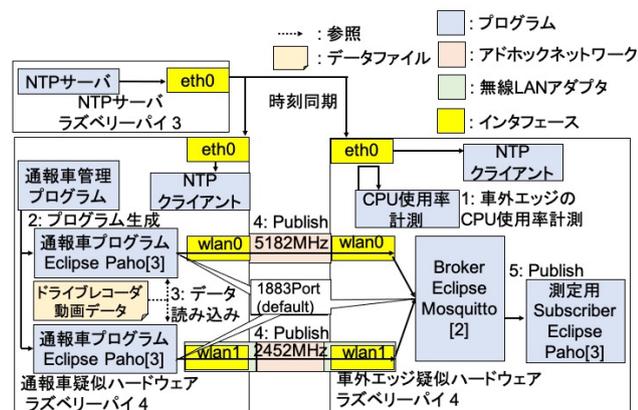


図 6 実装環境構成

7 提案方法の適用

7.1 ユースケース

ユースケースを図 7 に示す。高速道路であおり運転が発生し、1 台もしくは 2 台の通報車から通報が行われると仮定する。通報車から送信されたデータは、全て 1 局の基地局である車外エッジへ送信される。

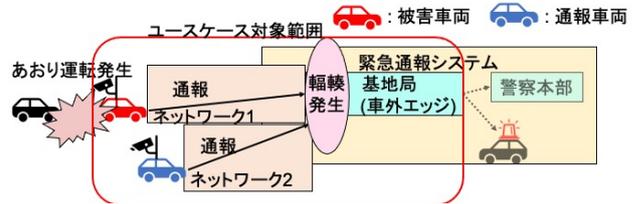


図 7 ユースケース

7.2 緊急メッセージ構成

緊急メッセージのトピック構成を図 8 に示す。緊急メッセージサイズに着目するため、位置情報等のセンサデータを省略し、ドライブレコーダ動画とみなすデータファイルのみを送信する。Video_Message_ID で送信したデータファイル同士の識別を行う。

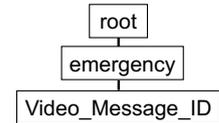


図 8 トピック構成

8 結果と評価

8.1 測定結果

測定結果を表 6, 7, 8 に示す。また、通報車が 2 台の場合の Publisher と Broker の送受信間隔を図 9, 10 に示す。CPU 使用率はシナリオ 1 とシナリオ 2 の全てで平均 2% の整数値で一定であった。

表 6 レイテンシの比較(シナリオ 1)

通報車台数(台)	ネットワーク 1		ネットワーク 2	
	平均(ms)	標準偏差(ms)	平均(ms)	標準偏差(ms)
1	162.85	5.82		
2	164.28	6.12	180.35	14.76
増加率 (2台/1台)	1.01			

表 7 処理時間の比較(シナリオ 1)

通報車台数(台)	ネットワーク 1		ネットワーク 2	
	平均(ms)	標準偏差(ms)	平均(ms)	標準偏差(ms)
1	0.16	0.03		
2	0.17	0.05	0.17	0.05
増加率 (2台/1台)	1.06			

表 8 処理時間の比較(シナリオ 2)

通報車台数(台)	ネットワーク 1		ネットワーク 2	
	平均(ms)	標準偏差(ms)	平均(ms)	標準偏差(ms)
1	0.19	0.05		
2	0.20	0.12	0.17	0.07
増加率 (2台/1台)	1.05			

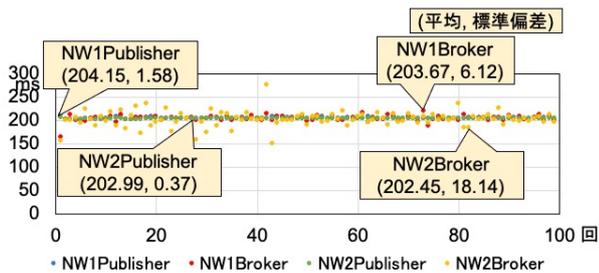


図 9 Publisher と Broker の送受信間隔(シナリオ 1)

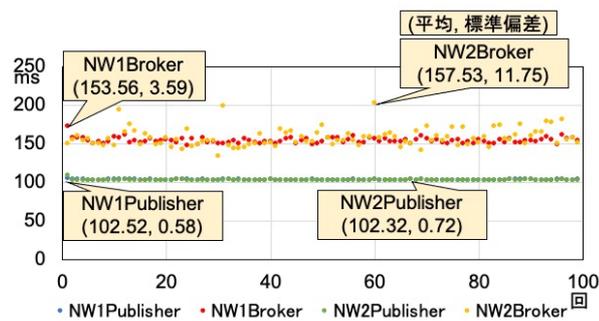


図 10 Publisher と Broker の送受信間隔(シナリオ 2)

8.2 評価結果

シナリオ 1 の評価結果は以下の通りである。

(1) レイテンシの比較

表 6 より、ネットワーク 1 のレイテンシ増加率が平均で 1.01 となっている。これは通報車が 1 台から 2 台に増加したことにより、通報車疑似ハードウェアと車外エッジ疑似ハードウェアの通信による負荷が増加していると考えられる。CPU 使用率は 2% で安定しており、車外エッジでの処理能力が効果を発揮していることが分かる。

(2) 処理時間の比較

表 7 より、ネットワーク 1 の処理時間の増加率が平均で 1.06 となっている。これは、通報車が 1 台から 2 台に増加したことにより、Broker で緊急メッセージの衝突が発生し、処理時間の増加が発生していることが考えられる。CPU 使用率は 2% で安定しており、車外エッジでの処理能力が効果を発揮していることが分かる。

シナリオ 2 の評価結果は以下の通りである。

(1) レイテンシの比較

通報車が 1 台の場合と 2 台の場合の両方で増大しているため省略する。

(2) 処理時間の比較

表 8 より、ネットワーク 1 の処理時間増加率が平均で 1.05 となっている。特に標準偏差に着目すると、通報車が 1 台の時 0.05ms であるのに対し、通報車が 2 台の時 0.12ms であるのは、車外エッジにネットワークの帯域を超えたデータ量を処理することにより、処理時間が不安定となっていると考えられる。CPU 使用率は 2% で安定しており、車外エッジでの処理能力が効果を発揮していることが分かる。

Publisher と Broker の送受信間隔による評価結果は、図 9、10 から分かるように、シナリオ 1 では Publisher の送信間隔と Broker の受信間隔がほぼ同じであり、シナリオ 2 では大きく異なっている。シナリオ 1 ではレイテンシが増大して

いないのに対し、シナリオ 2 ではレイテンシが増大しているため、通報時間の増大を防ぐためには Publisher の送信間隔 > Broker の受信可能な間隔である必要がある。

9 考察

先行研究[5]では、プロトタイプ環境下での、MQTT を用いたエッジ間のレイテンシ測定方法について提案している。Pub/Sub アーキテクチャは非同期なアーキテクチャであるが、メッセージ送信区間に区切り、エッジ間でメッセージを往復させることでレイテンシの測定を可能にしている。

本研究は、プロトタイプ環境で車外エッジにおける通報の集中による通報時間の増加に焦点を当てており、複数のネットワークからの通報を想定している。先行研究と比較し、複数のネットワークからのメッセージ送信を測定できる点が優位である。

10 今後の課題

- (1) 車外エッジの処理が超過した場合における提案測定方法の妥当性評価
本稿で測定した結果は、最大でも CPU 使用率が 2% と車外エッジの負荷が非常に少ない。基地局に通報が集中した場合、車外エッジの処理能力が不足する可能性があり、さらなる負荷をかけた場合における提案測定方法の妥当性を評価する必要がある。
- (2) 3 台以上の通報車からの通報を測定する場合における測定方法の提案
通報車が 3 台以上の場合のレイテンシを測定する際、異なる周波数同士の干渉の発生を考慮した周波数の割り当て方法を考案する必要がある。

11 まとめ

本稿では、テスト環境下での通報車と車外エッジ間のレイテンシ測定方法を提案した。プロトタイプを実装し、あり運転をユースケースとして測定を行った。その結果から、提案測定方法の妥当性の評価を行った。

参考文献

- [1] AECC, General Principle and Vision V. 3.0, Jan. 2020, https://aecc.org/wp-content/uploads/2020/07/General_Principle_and_Vision_January_31_2020.pdf.
- [2] Eclipse Mosquito, <https://mosquito.org>.
- [3] Eclipse Paho, <https://eclipse.org/paho/>.
- [4] P. T. Eugster, et al., The Many Faces of Publish/Subscribe, ACM Computing Surveys, Vol. 35, No. 2, Jun. 2003, pp. 114-131.
- [5] 廣瀬 隼大, 二重エッジシステムの性能評価方法の提案と評価, 2019 年度南山大学理工学研究科修士論文, Jan. 2020.
- [6] ISO/IEC 20922:2016, Information Technology - Message Queuing Telemetry Transport (MQTT) V. 3.1.1, 2016.
- [7] 警察庁 消防庁 国土交通省, 接続機関における自動車からの緊急通報の取扱いに関するガイドライン, <http://www.mlit.go.jp/common/001234105.pdf>.
- [8] 長坂卓朗 他, エッジコンピューティングによる緊急通報システムアーキテクチャ設計方法の提案と評価, 情報処理学会第 82 回全国大会講演論文集, Mar. 2020, pp. 185-186.
- [9] W. Shi, et al., The Promise of Edge Computing, IEEE Computer, Vol. 49, No. 5, May 2016, pp. 78-81.
- [10] 宇野 聡将 他, 二重エッジアーキテクチャ設計方法の提案と MQTT-Bridge を用いたプロトタイプによる評価, 情報処理学会第 81 回全国大会講演論文集, Mar. 2019, pp. 215-216.