

災害時における Publish/Subscribe 通信を用いた ネットワークとアプリケーションの相互制御機構

西山 潤^{1,a)} 多幡 早紀 小林 裕樹^{1,b)} 重野 寛^{1,c)}

受付日 2017年5月8日, 採録日 2017年11月7日

概要: 大規模災害が発生した際には、通信設備に甚大な被害が及ぶことが想定される。災害時は通常時と比較し、通信要求回数が増加するため、ネットワークに流入する通信量がネットワークの許容量を超えることが考えられる。通信量を抑制するためには、ネットワークとアプリケーションの両面から通信制御を行うことが効果的である。しかし、災害時には誤った情報を含む多くの情報が送受信されるため、通信制御を一律に行うのではなく、通信内容を反映した制御を行う必要がある。本論文では OpenFlow によるネットワーク制御と Publish/Subscribe モデルのメッセージング機構に焦点を当てた、ネットワークとアプリケーションの相互制御機構を提案する。提案機構のプロトタイプシステムを実装し、ネットワークの状況に基づくメッセージング制御および低オーバーヘッドの送信制御の観点から提案機構の有用性を示す。

キーワード: Publish/Subscribe モデル, OpenFlow, メッセージング制御, 通信制御

Network-application Interactive Messaging Mechanism for Publish/Subscribe Communication

JUN NISHIYAMA^{1,a)} SAKI TABATA YUKI KOBAYASHI^{1,b)} HIROSHI SHIGENO^{1,c)}

Received: May 8, 2017, Accepted: November 7, 2017

Abstract: The amount of communication data extremely increases compared with normal times at times of disaster. Therefore, the communication traffic is expected to be beyond the marginal capacity of the networks. In order to reduce communication traffic effectively, both the applications and the networks must work on the restriction control. In case that traffics are required to be restricted, it is desirable to reflect the priority of the messages to. We focus on OpenFlow and Publish/Subscribe messaging architecture that messages are published with topics. We propose the mechanism that realizes interaction between a network and applications to control the behavior of each other. We implement the prototype system of the proposed mechanism and show the usefulness of the proposed mechanism from the viewpoint of messaging control responding to the network conditions and low-overhead topic based transmission control.

Keywords: Publish/Subscribe, OpenFlow, messaging control, transmission control

1. はじめに

大規模災害発生時において、被災者に関する情報を迅速に共有することは、救命率を向上させるうえで非常に重要

である。災害時には通信トラフィックは通常時と比べて増加するため、ネットワークの限界容量を超過することが考えられる。これは被災者の安否確認や SNS から発信されるデマ情報等が影響を及ぼしている。通信トラフィックを効果的に削減するためには、ネットワークとアプリケーションの両面から制御をかけることが必要である。しかし、通信トラフィックに一律に制御をかけた場合、トラフィック量が増加している環境下では重要なメッセージの送受信に影響を及ぼすことが想定される。そのため、被災者に関す

¹ 慶應義塾大学大学院理工学研究科
Graduate School of Science and Technology, Keio University,
Yokohama, Kanagawa 223–8522, Japan

a) nishiyama@mos.ics.keio.ac.jp

b) kobayashi@mos.ics.keio.ac.jp

c) shigeno@mos.ics.keio.ac.jp

る情報等の重要度の高い情報は、通信内容に基づいて優先度を考慮する必要がある。

通信内容に基づいた優先制御を実現する方法として、Publish/Subscribe モデル [1] がある。Publish/Subscribe モデルではトピックと呼ばれるメッセージ内容を識別した情報を持つため、トピック情報に基づいたメッセージの選別や制御が可能となる。災害時にはネットワークリソースを考慮し、重要な情報のみを効果的に抽出する必要がある。このことから、Publish/Subscribe モデルは通信トラフィックの観点から、災害通信に適していると考えられる。

ネットワークの状況に応じた柔軟な制御を可能とする通信インタフェースとして OpenFlow [2], [3] がある。OpenFlow はパケットヘッダの任意のフィールドにアクセスすることでトラフィックのフローを分類することが可能である。OpenFlow ではトラフィックのフローを識別する方法として、パケットのヘッダ情報から識別する方法とペイロード情報から識別する方法の 2 種類がある。OpenFlow では、OpenFlow コントローラのみがペイロード情報にアクセス可能である。そのため、通信内容によりメッセージを識別するためには、OpenFlow コントローラへの転送が必要があり、コントローラへの負荷が増大する。

そこで、本論文では OpenFlow によるネットワーク制御と Publish/Subscribe モデルのメッセージング機構に焦点を当てた、ネットワークとアプリケーションの相互制御機構を提案する。提案機構の目的は以下の 2 点である。

- (1) アプリケーションがトラフィックを制御可能とすることによる不要なメッセージの送信の削減
- (2) 災害時における通信内容に応じたネットワークリソースの割当て

提案機構は、OpenFlow を活用した被災者の情報共有を実現する救命情報共有システム [4], [5], [6] に適用することを想定している。提案機構では、主に 2 つの要素を導入する。1 つ目はリアルタイムにネットワークの状態をアプリケーションの挙動に反映させたメッセージング制御であり、2 つ目はアプリケーションからの要求に応じたトピックベースの送信制御である。Publish/Subscribe モデルでは、メッセージはアプリケーションヘッダで表されるトピックに基づき送信される。そのためトピックを解析することで、フローの分離が実現でき、OpenFlow コントローラのパケット転送処理における負荷を軽減することができると考えられる。

以下、2 章で関連研究を示し、3 章で今回提案するシステムについての説明を行い、4 章で提案機構のプロトタイプシステムと評価方式を説明する。そして 5 章で提案機構の有用性を示し、6 章で結論を示す。

2. 関連研究

本章では、不安定なネットワーク環境下における情報共

有に関する研究および Publish/Subscribe モデルについて述べ、本研究のアプローチについて説明する。

2.1 災害時の情報共有

災害医療派遣チームである DMATs (Disaster Medical Assistance Teams) [7] は、災害発生 24 時間以内に被災地に到着することが求められる。しかし、Kitsuta らの報告 [8] では DMATs は被災地到着後、即座に被災者を救助することが難しいと指摘されている。これは、DMATs は被災地到着直後から被災者に関する情報を収集しなければならないのが原因である。そこで、DMAT 間の情報共有を支援し、救助活動の早期開始と被災者の救命率の向上を実現する救命情報共有システム [4], [5], [6] が研究されている。また、日本においては EMIS (Emergency Medical Information System) [9] と呼ばれる病院の稼働状態に関する情報を収集する医療機関向けのシステムも運用されている。これらのシステムは、ネットワークを用いて情報を収集するため、災害時の不安定なネットワーク環境を考慮することに重点が置かれている。

2.2 不安定なネットワーク環境下における情報伝達

2.2.1 Delay/Disruption Tolerant Networking (DTN)

DTN は高遅延かつ頻繁に回線が切断される災害通信の研究で注目されている。Kawamoto ら [10] は、DTN ベースのワイヤレス災害情報システムを提案した。この提案ではメッセージ収集性能を向上させるために、送信者と受信者との位置に応じたメッセージリレーアルゴリズムに基づいて動作する。また、Krug ら [11] は災害時の情報収集において、MANETs (Mobile Ad hoc Networks) の高速展開に焦点を当てている。MANETs は高可用性を備えているため、災害時の状況に適していると考えられている。

2.2.2 Software Defined Networking (SDN)

SDN は、ソフトウェア制御によりネットワーク構成を動的に制御することができる。SDN を実現する代表的な技術として OpenFlow がある。OpenFlow ネットワークでは、OpenFlow コントローラは柔軟なマッチングルールと Packet In メカニズムにより、ネットワーク全体を自由に制御することが可能である。災害時の情報伝達には SDN に基づく研究が行われている。Manic ら [12] は、SDN によるネットワーク仮想化技術を活用した ECS (緊急通信システム) のフレームワークを提案した。また、Sato ら [13] は OpenFlow を導入することで、コグニティブ無線ネットワークシステムを制御し、ネットワークの接続性を向上させる。コグニティブ無線ネットワークでは、通信パラメータは無線ネットワークの利用状況に応じて動的に変更されるため、不安定なネットワーク環境下において効果的に適応する。

2.2.3 移動体基地局

移動体基地局を用いることで、通信設備が被害を受けた地域においても情報の送受信が可能となる。MDRU (Movable and Deployable Resource Unit) [14] もその1つである。Ngo ら [14] は分散画像データベースを用いた複数のMDRUを用いたネットワーク環境下での安全確認アプリケーションを提案している。

また、近年、V2V (Vehicle-to-Vehicle) 通信が活発に議論され、通信機器を搭載した車両が実用化されている。ARIB SD-T109 [15] は、700 MHz 帯を使用した車車間通信の規格の1つである。今回の提案機構は、MDRUと比較してOpenFlowによる車両間の通信およびOpenFlowを用いた積極的なネットワークリソースの管理に特徴がある。

2.3 Publish/Subscribe モデル

Publish/Subscribe モデルは、メッセージの送信者が特定の受信者を想定せずにメッセージを送信する非同期データ配送機構である。メッセージの送信者は Publisher、受信者は Subscriber と呼ばれ、この2つを相互接続するサーバをブローカと呼ぶ。各 Subscriber は受信したいトピックが異なるため、ブローカは Publisher から届いたメッセージをどの Subscriber に送信するのか、逐一判断する必要がある。そのため、Subscriber は必要な情報のみを受信することが可能となる。災害時には、誤りを含む多くの情報が送受信されるため、重要な情報のみを効果的に抽出する必要がある。このことから、Publish/Subscribe モデルは通信トラフィックの観点から、災害通信に適していると考えられている。

2.4 問題点

通信トラフィックを効果的に削減するためには、ネットワークとアプリケーションの両面から制御をかける必要がある。また、災害時にはネットワークへ流入するトラフィック量が増加するため、通信トラフィックに一律に制御をかけた場合、重要なメッセージの送受信に影響を及ぼすことが想定される。そのため、被災者に関する情報等の重要度の高い情報は、通信内容に基づいて優先度を考慮する必要がある。しかし、アプリケーションからのメッセージ内容を反映したメッセージレベルでのネットワーク制御を行うシステムは考慮されていない。

3. 提案

本論文では OpenFlow によるネットワーク制御と Publish/Subscribe モデルのメッセージング機構に焦点を当てた、ネットワークとアプリケーションの相互制御機構を提案する。

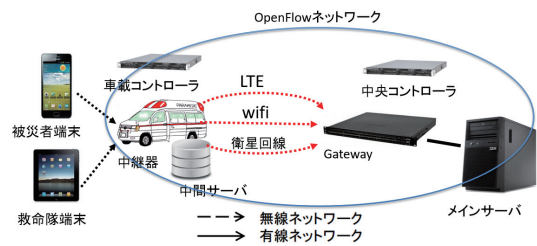


図 1 通信環境

Fig. 1 Communication environment.

3.1 提案環境

救命情報共有システムの全体像を図 1 に示す。このシステムは、災害発生時における救助活動の早期開始および被災者の救命率を向上させることを目的とする。このシステムでは、被災地における通信回線を確保のため、公共車両に中継器を搭載する。中継器は臨時的の基地局として、周囲の端末から発信される要救助者に関する情報を受信し、メインサーバへ情報を中継する。また、情報の宛先であるメインサーバが接続する有線ネットワークは、メインサーバに加えて、関連機関の機器等にも接続されており、このシステムを中心とするネットワークである。Gateway スイッチは、各地から無線ネットワークを用いて送信された情報を受信し、有線ネットワークに送信する。

中継器と Gateway スイッチ間の通信には、複数の無線回線が使用可能である。これは、災害時は回線の状態が不安定であることを考慮し、冗長化を狙ったものである。救命情報共有システムでは、不安定なネットワーク環境下で効率的に回線を使用するために OpenFlow が導入されている。本論文では、回線の優先度にコントローラによる回線の監視結果や外部から与えられる情報を使用し、回線の制御を行うものとする。有線ネットワークおよび中継器と Gateway スイッチ間の無線区間を制御する OpenFlow コントローラを中央コントローラと呼ぶ。また、全回線切断時に中継器の制御を行う車載コントローラと情報を一時的に保持しておく中間サーバが導入されている。

3.2 提案概要

提案機構は、救命情報共有システムの環境下で動作することを想定する。Publish/Subscribe モデルのプロトコルとして MQTT [16] を使用し、被災地を移動する車両に車載ブローカと車載エージェントを導入する。車載エージェントは、車載ブローカと災害対策本部に設置された中央ブローカとの間のメッセージを中継する。

提案機構ではネットワークとアプリケーション間の相互制御を実現する。図 2 に提案機構のブロック図を示す。本機構は主に 2 つのメカニズムで構成される。1 つ目は図 2 に実線で示されているメッセージング制御メカニズムであり、もう 1 点は破線で示されたトピックベースの送信制御メカニズムである。

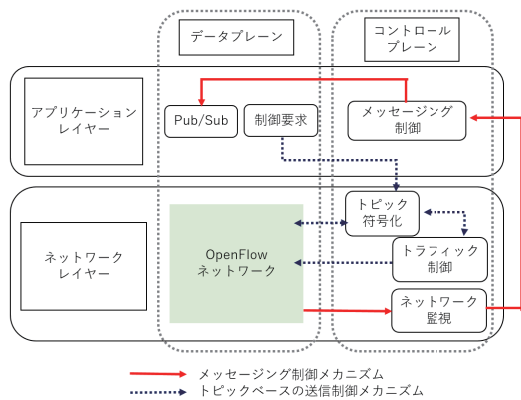


図 2 提案機構のブロック図

Fig. 2 Block diagram of the proposed mechanism.

メッセージング制御は、OpenFlow コントローラがネットワークの状況に応じてアプリケーションを制御するためのものである。車載 OpenFlow コントローラによってネットワークの状態を監視し、その結果をアプリケーションの動作に反映させる。アプリケーションがトラフィックを制御可能とすることで、ネットワーク内を不要なメッセージの送信が削減され、ネットワークの管理が容易になる。

一方、トピックベースの送信制御はアプリケーションがネットワークを制御するためのものである。OpenFlow のデータプレーンでメッセージ内容を識別するために、メッセージのトピック情報をパケットのヘッダに符号化する。そのため、OpenFlow スイッチが OpenFlow コントローラに転送することなく、パケットを識別することが可能となる。したがって、OpenFlow コントローラの処理負荷を大幅に軽減できると考える。また、災害時において、可用性を向上させるために OpenFlow と Publish/Subscribe の両方で冗長化と分散アーキテクチャを導入している。これらの提案された機構により、不要なメッセージの送信を削減し、災害時の限られたネットワークリソースを通信内容に応じて割り当てることが可能となる。

3.3 車載ブローカと車載エージェント

図 3 に提案機構の想定環境を示す。提案機構には 2 種類の Publish/Subscribe ブローカを導入する。1 つ目は救命隊本部に設置された中央ブローカ、もう 1 つは公共車両に搭載された車載ブローカである。中央ブローカは安定したネットワーク環境下での一般的なブローカとして動作する。車載ブローカは被災地において被災者に関する情報を収集する役目を担い、収集したメッセージは最終的に中央ブローカによって収集される。車載エージェントは中央ブローカとの通信が可能に際し、受信したメッセージを中央ブローカに送信する。また、通信が不能な場合には車載エージェントは中央ブローカとの通信が可能となるまで、受信メッセージを一時的に保存する。

車載ブローカと中央ブローカとの間でメッセージの送

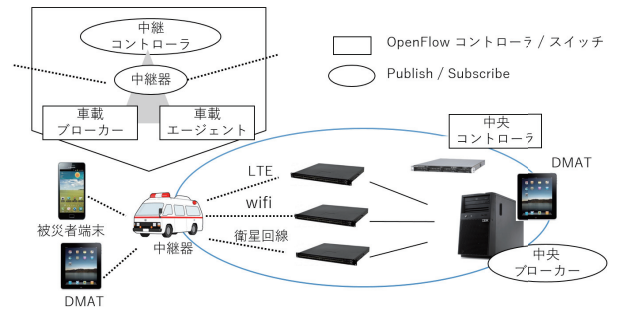


図 3 ネットワークの構成要素

Fig. 3 Network components of the proposed mechanism.

受信を行うために車載エージェントを導入する。これは、Publish/Subscribe モデルではメッセージの送受信にブローカが必要であり、単一障害点となりうるからである。車載ブローカと中央ブローカとを切り離すことにより、中継器は中央ブローカと通信不能な場合においてもメッセージを中継することが可能となる。また、受信したメッセージは車載エージェントを一度経由することにより、メッセージの編集やフィルタリングが可能となる。たとえば、災害時の限られたネットワークリソースを考慮し、被災者から受け取った大量の画像を車載エージェントで圧縮するといったことが可能となる。

3.4 メッセージング制御

車載エージェントは回線の状況に応じて中央ブローカへの送信量を調整する。救命情報共有システムでは、中央コントローラと車載コントローラの両方が中継器と Gateway スイッチとの間の無線回線の状況を監視する。その監視結果に基づき、OpenFlow コントローラは経路情報を更新し、帯域保証等の無線区間におけるトラフィック制御を実現する。車載コントローラは車載エージェントと回線状況を共有し、車載エージェントのメッセージング制御を実施する。

車載コントローラは Keep Alive パケットによる無線回線の接続状況を検出する。回線の切断を検知した場合、車載コントローラは車載エージェントに対してメッセージの送信を一時的に中断させる。また、回線が復旧した場合には、車載エージェントに対してメッセージの送信を再開させる。提案機構では、無線区間におけるトラフィック制御を実現するために、メッセージの送信の開始と停止に加え、ネットワークの輻輳レベルに基づいて車載エージェントからのメッセージ送信量を調整する。

3.5 トピックベースの伝送制御

提案機構では、OpenFlow のデータプレーンがメッセージの識別を可能にするためにメッセージのトピック情報をパケットヘッダに符号化する。トピック情報は車載コントローラでパケットのヘッダに符号化される。そのため、OpenFlow スイッチは OpenFlow コントローラにパケット

を転送することなく、パケットに符号化されたトピック情報に基づいてトラフィックを制御することが可能となる。図 4 にトピック情報をパケットヘッダに符号化する手順を示し、以下に詳細を示す。

- (1) 中継器が MQTT メッセージを受信した際に、中継器は車載コントローラに Packet In メッセージを送信する。一部の TCP ポート番号は事前に MQTT メッセージ用に中継器に登録されており、中継器は TCP ヘッダで MQTT メッセージを識別することが可能となる。
- (2) 車載コントローラは受信したパケットが Publish メッセージであることを確認し、MQTT のヘッダを識別することでメッセージタイプ情報を抽出する。
- (3) Publish メッセージの場合には、MQTT のヘッダからトピック情報を抽出する。
- (4) 抽出されたトピック情報が Publish/Subscribe アプリケーションにおいて制御可能であるかを確認する。本論文では、トラフィックの制御要求は外部アプリケーションによって与えられる。トピック情報にトラフィック制御の必要がない場合、車載コントローラはメッセージに対して処理を加えることなく送信を行う。
- (5) トピック情報をパケットヘッダに符号化する。トピック情報の符号化方式については、OpenFlow コントローラ間で共有する必要がありネットワーク全体で整合がとれている必要がある。
- (6) トピック情報が符号化されたパケットと一致するフローエントリを OpenFlow スイッチに追加する。中央コントローラはメッセージの宛先であるエッジの OpenFlow スイッチに対して、トピック情報を復号化するためのフローエントリの追加の指示を出す。
- (7) 符号化されたパケットを中継器に送信する。符号化されたフィールドはメッセージの宛先ノードが適切に受

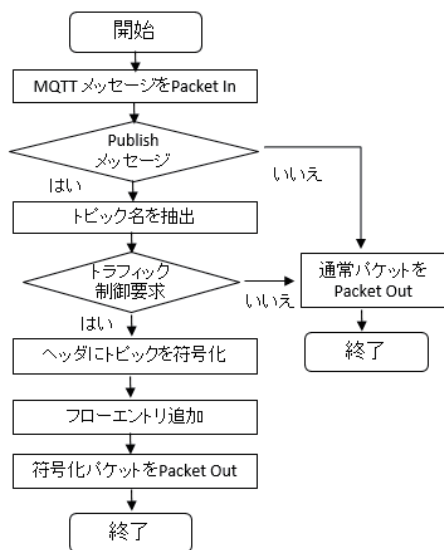


図 4 トピック符号化の流れ
Fig. 4 Flow of topic encoding.

信するように復号化する必要がある。トピック情報をエッジの OpenFlow スイッチで復号化することで、トピックベースのメッセージの送信が可能となる。

3.6 データプレーンの動作

トピックベースの送信制御は OpenFlow スイッチがフローテーブルを参照することで実現される。OpenFlow ネットワークのエッジにおいて、スイッチは符号化されたパケットヘッダを復号化する。OpenFlow スイッチではパケットヘッダの書き換えが可能であるため、トピックの復号化は OpenFlow コントローラを用いることなく実現できる。したがって、提案機構ではトピックベースの送信制御を実現するために、1つのパケットに対して一度の Packet In メッセージのみが必要である。これにより、OpenFlow コントローラの処理オーバーヘッドを大幅に削減できると考えられる。

4. 評価方式

本章では、実装したプロトタイプシステムの概要および提案機構の評価項目と実験概要を示す。

4.1 プロトタイプシステム実装

提案機構の動作確認および評価を行うために、プロトタイプシステムを実装した。図 5 に実装したプロトタイプシステムの各コンポーネントに実装された機能および実験環境を示す。また、表 1 にプロトコルのバージョンおよびハードウェアの仕様等のプロトタイプシステムの詳細情報を示す。プロトタイプシステムでは中継器と Gateway スイッチ間は、ネットワークエミュレータを用いることで無線接続を再現した。

トピック符号化機能およびトラフィック制御機能は車載コントローラに実装されており、メッセージのトピック情報はパケットの TCP ソースポートに符号化される。また、メッセージの各トピック情報は外部アプリケーションから与えられる帯域幅を保証するように動作する。車載エージェントには中間処理機能を実装しており、メッセージの

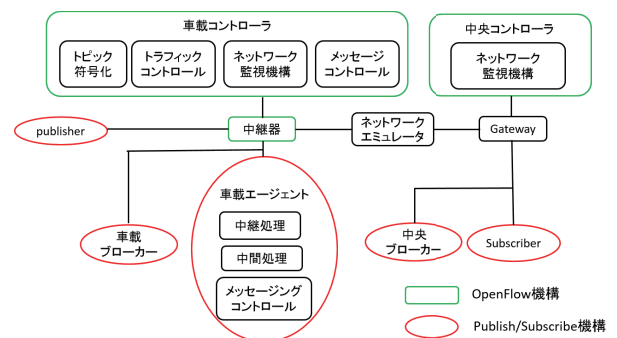


図 5 実験環境の構成図
Fig. 5 Implemented functions.

表 1 実装環境の構成

Table 1 Specifications of prototype system.

Protocol	
OpenFlow	OpenFlow 1.3
Publish/Subscribe	MQTT v3.1
Software	
OpenFlow コントローラ	Ryu ver.3.18 [17]
OpenFlow スイッチ	Open vSwitch ver. 2.0.2 [18]
MQTT ブローカ	Mosquitto ver. 1.4.9 [19]
Hardware	
車載機器 (コントローラ/中継器, ブローカ/エージェント)	OpenBlocks AX3 (ARMADA XP 1.33 GHz, RAM 1 GB, 1000Base-T 4ports)
中央コントローラ	Intel-based server (Xeon E3-1220, RAM 4 GB, 1000Base-T 2ports)
Gateway	Pica P-3297

中間処理として、Publisher からの画像付きメッセージを受信した際にはサムネイルを作成する。

4.2 評価項目

本実験での動作確認および評価項目は以下の3点である。

- (1) リアルタイムにネットワークの状態をアプリケーションの挙動に反映させたメッセージング制御
- (2) アプリケーションからの要求に応じたトピックベースのトラフィック制御
- (3) トピックベースの packets 転送処理によるネットワーク機器の負荷軽減

上記3種類の実験により、ネットワーク状況に基づくメッセージング制御および低オーバーヘッドの送信制御の観点から提案機構の有用性を示す。

4.3 各実験におけるシナリオとパラメータ

4.3.1 動作確認シナリオ1：メッセージング制御

動作確認シナリオ1では、ネットワークの状況を車載エージェントの動作に反映させ、宛先に到達できない不要なトラフィックを削減できているかを確認した。動作確認シナリオ1で使用したシナリオを図6に、パラメータを表2に示す。Publisher はパブリッシュメッセージを定期的に送信し、車載エージェントは車載ブローカからメッセージをSubscribeする。また、車載エージェントは受信したメッセージを中央ブローカへとPublishする。ネットワークエミュレータがパケット損失率の値を操作することで、リンクの切断と復旧を実現した。一定時間内の Keep Alive パケットの受信数に基づいた回線のパケットロス率をもとに優先度の設定を行い、送信間隔を0.9秒、タイムアウト時間を1秒に設定した。

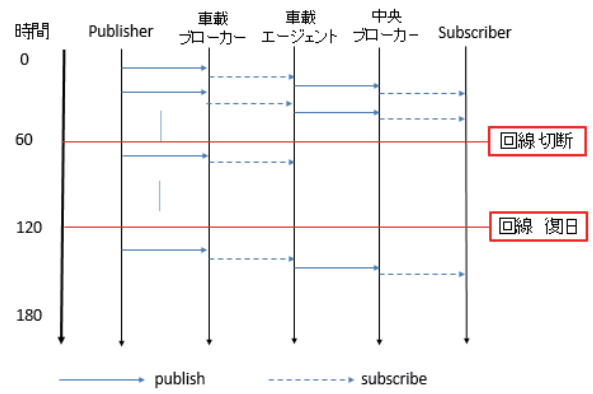


図 6 メッセージング制御の動作確認シナリオ
Fig. 6 Message flow of scenario 1.

表 2 動作確認シナリオ1で用いたパラメータ
Table 2 Parameters in scenario 1.

メッセージサイズ	1.5 [KB]
回線の帯域	100 [Mbps]
KeepAlive パケットの送信間隔	0.9 [sec]
リンク切断検知時間	1.0 [sec]

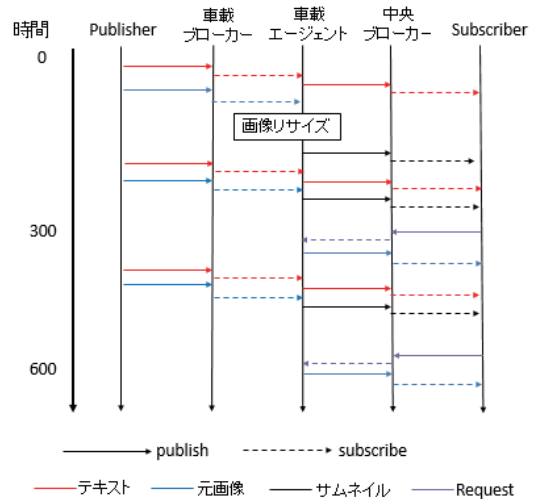


図 7 トラフィック制御の動作確認シナリオ
Fig. 7 Traffic flow of scenario 2.

表 3 動作確認シナリオ2で用いたパラメータ
Table 3 Bandwidth allocation configuration in scenario 2.

コンテンツ	メッセージ サイズ	帯域 (最小) [Mbps]	帯域 (最大) [Mbps]
デフォルト	-	-	10
テキスト	1.5 [KB]	10	30
サムネイル	46 [KB]	30	-
画像	2.1 [Mb]	50	-

4.3.2 動作確認シナリオ2：トラフィック制御

動作確認シナリオ2では、トピックに基づいた各キューへのトラフィックの割当てを確認した。今回使用したシナリオを図7に、パラメータを表3に示す。本実験では10秒間隔でテキストと画像のPublishを行った。また、

Subscriber は受信したサムネイルの元画像にアクセスするためのリクエスト機能を提供する。この機能はメッセージの送信を開始してから 300 秒後に有効になる。

中継器のキューは動作確認用としてキュー 0, 1, 2, 3 の 4 種類を用いた。キュー 1, 2, 3 はそれぞれテキストメッセージ、サムネイル、画像情報用に割り当てられたキューである。キュー 0 に関しては、一般的なスイッチで動作しているデフォルトのキューが設定されている。本実験は、トピックに基づく各キューへのトラフィック割当てを確認するものであるため、帯域幅は決め打ちの値を用いた。また、テキスト情報は、救命情報共有システムで用いられる METHANE 情報を使用し、画像サイズは一般的に普及している携帯端末で撮影した画像を使用した。

4.3.3 評価実験：ネットワーク機器の負荷軽減

評価実験では、OpenFlow コントローラにかかる負荷軽減が実現できているかを確認した。中継器と Gateway スイッチからの Packet In メッセージ量を測定し、トピックベースの転送処理を適応することで Packet In メッセージ量の削減を確認した。

5. 動作確認と評価

本章では、実装したプロトタイプシステムを用いて行った評価結果を示す。

5.1 動作確認 1：メッセージング制御

図 8 にネットワークエミュレータが監視したトラフィックを示す。この図は Wireshark [20] を用いてネットワークエミュレータでパケットをキャプチャした様子である。本実験では、開始 60 秒から 120 秒の間で回線を切断した。その間、図 8 にはリンク接続検出用の Keep Alive パケットのみが示されている。そのため、車載エージェントからの送信は回線が使用できない際には停止している様子が確認される。したがって、ネットワークの状況を車載エージェントの動作に反映できていることを確認した。これにより、ネットワークの状況をアプリケーションの動作に反映させることで、宛先に到達できない不要なトラフィックの量を減らし、ネットワークリソースの節約を実現できることを確認した。

5.2 動作確認 2：トピックベース送信制御

5.2.1 受信データ量

図 9 に Subscriber のトピック別の受信メッセージ量を示す。本実験では、車載エージェントはテキストメッセージと画像メッセージの両方を受信し、Subscriber はテキストメッセージとサムネイルを受信する。車載エージェントは受信した画像に中間処理を施し、サムネイルを作成することで Subscriber が受信するデータ量は大幅に減少させる。Subscriber は受信したサムネイルの元画像を要求するリクエスト機能を保持しており、車載エージェントは過去

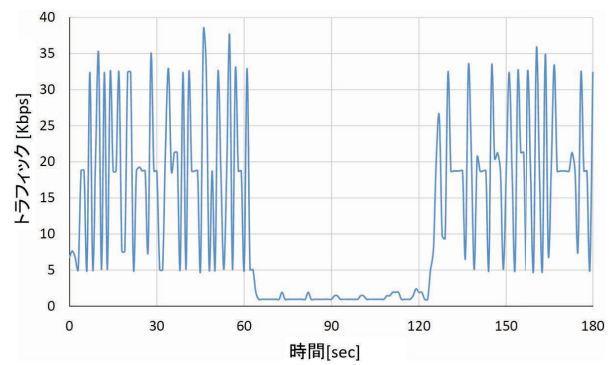


図 8 回線状況に応じたトラフィック制御

Fig. 8 Traffic control based on link conditions.

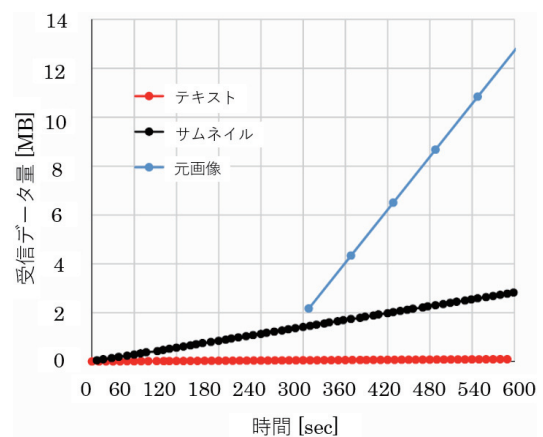


図 9 受信したメッセージ量

Fig. 9 The amount of received messages.

に送信したサムネイルに対応するリクエスト・メッセージを受信すると、そのサムネイルの元画像の送信を開始する。Subscriber は、このリクエスト機能により 300 秒後から受信したサムネイルの元画像を受信している。これにより、Subscriber が要求した情報を取得できていることが確認された。この実験から、車載エージェントで中間処理を加えることにより送信量を節約できることを確認した。

5.2.2 キューごとのトラフィック量

図 10 に、中継器での各キューにおけるトラフィック量を示す。図 9 では各キューに応じたトラフィック量を測定し、図 10 では Subscriber が受信したメッセージ量を測定した。これにより、パケットヘッダ分の違いが確認されるが、キューを通過するトラフィック量は Subscriber が受信したテキスト、サムネイル、画像情報のトラフィック量とほぼ同等であることが読み取れる。また、キュー 0 に割り当てられたパケットは Publish メッセージ以外に用いられた、リンク検出や MQTT トラフィックのための Keep Alive パケット等である。この結果から、提案機構が各メッセージに対して適切なキューが割り当てられ、トピックに応じたメッセージング制御を実現していることを確認した。

図 11 にネットワークエミュレータの帯域幅使用量を示す。各トピックによって最低帯域幅と最高帯域幅を設定し

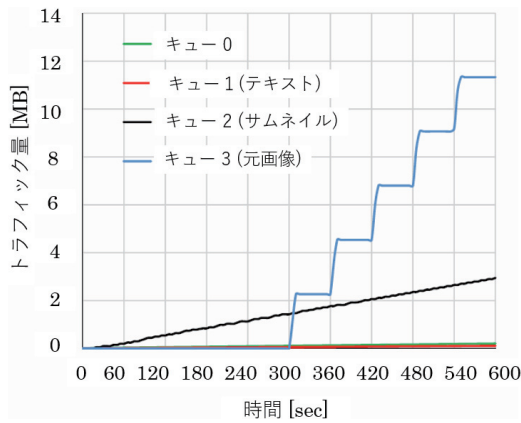


図 10 各キューに応じたトラフィック量
Fig. 10 Traffic value of each queue.

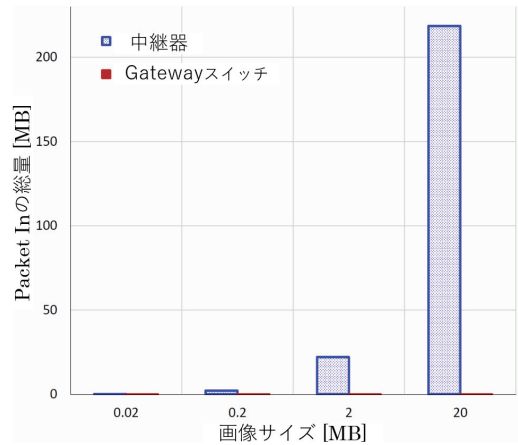


図 12 トラフィック量
Fig. 12 Packet in statistics.

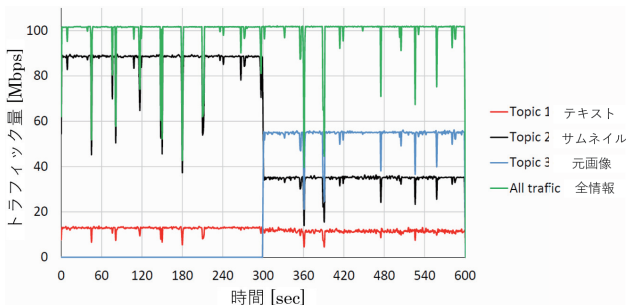


図 11 各キューの帯域
Fig. 11 Bandwidth in each queue.

ており、合計のトラフィック量は回線の許容量を超過しないように制御した。元画像の送信が開始されると、トピック 1 とトピック 2 によって利用される帯域幅が減少し、トピック 1、トピック 2、トピック 3 のフローは、約 12 Mbps、約 35 Mbps、および 55 Mbps の帯域幅を利用している様子が確認された。これらの結果から、提案手法がトピックベースの帯域割当てを実現することを確認した。

5.2.3 評価実験：OpenFlow コントローラの処理負荷

図 12 に、中継器と Gateway スイッチからの Packet In メッセージの量を示す。通常、中継器からの Packet In メッセージの量は画像サイズの増加とともに増加する。プロトタイプシステムでは、Packet In メッセージ量は送信されたメッセージ量と同等となるように、Packet In メッセージを車載コントローラに向けて送信する。一方、Gateway スイッチからの Packet In メッセージ量は、画像サイズに応じた変化が見られなかった。これは、Gateway スイッチが OpenFlow コントローラに転送することなく、パケットを復号しているためだと考えられる。

提案機構では、中継器から車載コントローラへの Packet In メッセージは車両内で完結するため、ネットワーク全体に影響を及ぼすことはない。このため、トピックをパケットに付与し、Packet In メッセージが発生することなくトラフィックの転送を可能としたことは、通信トラフィックが通常時と比べて増加する災害時において、ネットワーク機器の

負荷軽減の観点から有用性があることが理解できる。また、Packet In メッセージや Packet Out メッセージの送受信時間やコントローラでの処理時間を短縮できると考えられ、全パケットを Packet In させる場合と比較し、Subscriber の情報の取得時間において改善効果が見込めると考えられる。

6. おわりに

本論文では OpenFlow によるネットワーク制御と Publish/Subscribe モデルのメッセージング機構に焦点を当てた、ネットワークとアプリケーションの相互動作制御機構を提案した。提案機構では、メッセージング制御とトピックベースの送信制御により、ネットワークとアプリケーション間の相互連携を可能とした。メッセージング制御メカニズムでは、OpenFlow ネットワークが回線状況を監視し、その結果をネットワーク上で動作するアプリケーション動作に反映させる。また、トピックベースの送信制御メカニズムではアプリケーションからの要求に応じてメッセージレベルのトラフィックを制御する。

提案機構のプロトタイプシステムを実装し、提案機構の動作確認と評価を行った。動作確認により、リアルタイムにネットワークの状態をアプリケーションの挙動に反映させたメッセージング制御、および符号化されたパケットが帯域幅の保証の制御に基づいたトピックベースの送信制御の実現を確認した。また、実験からトピックをパケットに符号化することで、OpenFlow コントローラによる負荷を軽減することを可能とした。以上から、提案機構はネットワークの状況に基づくメッセージング制御および低オーバーヘッドのトピックベース送信制御を確認した。

参考文献

- [1] Eugster, P.T., Felber, P.A., Guerraoui, R. and Kermarrec, A.M.: The Many Faces of Publish/Subscribe, *ACM Computing Surveys*, Vol.35, No.2, pp.114-131 (2003).
- [2] Mckeown, N., Anderson, T., Balakrishnan, H., Parulkar,

G., Peterson, L., Rexford, J., Shenker, S. and Turner, J.: OpenFlow: Enabling Innovation in Campus Networks, *ACM SIGCOMM Computer Communication Review*, Vol.38, No.2, pp.69–74 (2008).

[3] Open Networking Foundation: OpenFlow – Open Networking Foundation (online), available from <https://www.opennetworking.org/sdn-resources/OpenFlow> (accessed 2017-01-12).

[4] Tabata, S., Nishiyama, J., Fukui, R., Shimazu, K. and Shigeno, H.: Disaster Information Gathering Mechanism Based on Web Caching and OpenFlow in Unstable Networks, *Proc. 2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA 2016)*, pp.17–24 (2016).

[5] 多幡早紀, 西山 潤, 福井良太郎, 嶋津恵子, 重野 寛: 不安定な通信環境における情報損失を低減する災害情報収集機構, 情報処理学会論文誌, Vol.58, No.2, pp.471–480 (2017).

[6] Nishiyama, J., Tabata, S. and Shigeno, H.: An Efficient Image Gathering Scheme with Quality Control in Disaster, *IPSN DPS*, pp.1–8 (2016) (in Japanese).

[7] U.S. Department of Health & Human Services: Disaster Medical Assistance Team (DMAT) (online), available from <https://www.phe.gov/preparedness/responders/ndms/teams/pages/dmat.aspx> (accessed 2017-01-12).

[8] Eugster, P.T., Felber, P.A., Guerraoui, R. and Kermarrec, A.M.: The Many Faces of Publish/Subscriber, *ACM Computing Surveys*, Vol.35, No.2, pp.114–131 (2003).

[9] Ministry of Health, Labour and Welfare: EMIS – Emergency Medical Information System (online), (accessed <http://www.wds.emis.go.jp/>) (accessed 2017-01-12).

[10] Kawamoto, M. and Shigeyasu, T.: Message Relay Decision Algorithm to Improve Message Delivery Ratio in DTN-based Wireless Disaster Information Systems, *Proc. 2015 IEEE 29th International Conference on Advanced Information Networking and Applications (AINA 2015)*, pp.822–828 (2015).

[11] Krug, S., Begerow, P., Rubaye, A.A., Schellenberg, S. and Seitz, J.: A Realistic Underlay Concept for Delay Tolerant Networks in Disaster Scenarios, *Proc. 2014 10th International Conference on Mobile Ad-hoc and Sensor Networks (MSN 2014)*, pp.163–170 (2014).

[12] Manic, M., Wijayasekara, D., Amarasinghe, K., Hewlett, J., Handy, K., Becker, C., Patterson, B. and Peterson, R.: Next Generation Emergency Communication Systems via Software Defined Networks, *Proc. 2014 3rd GENI Research and Educational Experiment Workshop (GREE 2014)*, pp.1–8 (2014).

[13] Sato, G., Uchida, N. and Shibata, Y.: Performance Evaluation of Software Defined and Cognitive Wireless Network Based Disaster Resilient System, *Proc. 2015 IEEE 29th International Conference on Advanced Information Networking and Applications (AINA 2015)*, pp.741–746 (2015).

[14] Ngo, T., Nishiyama, H., Kato, N., Sakano, T. and Takahara, A.: An Efficient Safety Confirmation Method Using Image Database in Multiple-MDRU-Based Disaster Recovery Network, *IEEE Systems Journal*, Vol.11, No.4, pp.2556–2565 (2015).

[15] Association of Radio Industries and Businesses: 700 MHz BAND INTELLIGENT TRANSPORT SYSTEMS – ARIB STD-T109 Version 1.2 (online), available from <http://www.arib.or.jp/english/html/overview/doc/5-STD-T109v1.2-E1.pdf> (accessed 2017-01-12).

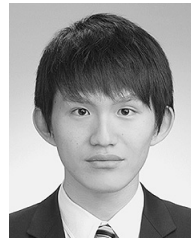
[16] MQTT.org: MQTT (online), available from <http://mqtt.org/> (accessed 2017-01-12).

[17] Ryu SDN Framework Community: Ryu SDN Framework (online), available from <http://osrg.github.io/ryu/> (accessed 2017-01-12).

[18] Linux Foundation: Open vSwitch (online), available from <http://openvswitch.org/> (accessed 2017-01-12).

[19] Eclipse: Mosquitto – An Open Source MQTT v3.1/v3.1.1 Broker (online), available from <https://mosquitto.org/> (accessed 2017-01-12).

[20] Wireshark Foundation: Wireshark – Go Deep (online), available from <https://www.wireshark.org/> (accessed 2017-01-12).



西山 潤 (学生会員)

2016 年慶應義塾大学理工学部情報工学科卒業。現在、同大学大学院理工学研究科修士課程在学中。



多幡 早紀 (学生会員)

2015 年慶應義塾大学理工学部情報工学科卒業。現在、同大学大学院理工学研究科修士課程修了。



小林 裕樹 (学生会員)

2017 年慶應義塾大学理工学部情報工学科卒業。現在、同大学大学院理工学研究科修士課程在学中。



重野 寛 (正会員)

1990 年慶應義塾大学理工学部計測工学科卒業。1997 年同大学大学院理工学研究科博士課程修了。現在、同大学理工学部教授。博士(工学)。情報処理学会論文誌編集委員会, 電子情報通信学会英文論文誌 B 編集委員等を歴任。現在, 情報処理学会理事, 同マルチメディア通信と分散処理研究会主査, Secretary of IEEE ComSoc APB. ネットワーク・プロトコル, ITS 等の研究に従事。著書『ユビキタスコンピューティング』(オーム社), 『情報学基礎第 2 版』(共立出版)等, 電子情報通信学会, IEEE, ACM 各会員。