

## WSNにおける情報収集遅延を考慮したパケット集約法の検討

チャルーンスックブンパシット† 安達 宏一†

† 電気通信大学先端ワイヤレス・コミュニケーション研究センター

〒182-8585 東京都調布市調布ヶ丘 1-5-1

E-mail: †{bounpasith, adachi}@awcc.uec.ac.jp

あらまし 近年, IoT (Internet-of-Things) の発展に伴い無線センサネットワーク (WSN: Wireless Sensor Network) が我々の生活の様々な部分に浸透してきている。WSN は, 産業やヘルスケアなどの広い分野に応用されており, 特に最近では温度や二酸化炭素 CO<sub>2</sub> などの定期観測に利用される事例が多い。IoT 機器用の 920MHz 帯アンライセンスバンドを用いる通信規格として, LoRaWAN (Long Range Wide Area Network) や Wi-SUN (Wireless Smart Utility Network) に代表される LPWAN (Low Power Wide Area Network) が挙げられる。それに対して, IEEE 標準規格の一つとして 920MHz 帯アンライセンスバンドを用いる IEEE 802.11ah (WiFi HaLow™) があり, 特に IoT 向けの通信システムへの適用が期待されている。環境情報の集約では, パケットあたりに伝送されるデータ量は小さいため, データ量に対してヘッダなどのオーバーヘッドが大きくなるという問題がある。IEEE 標準規格には複数データを集約して送信する方法があり, パケットを送信する際に必要となるデータ量に対するオーバーヘッド比率を低減することでより効率的な情報集約が可能となる。本稿では, パケット衝突によるデータ損失ならびにデータのリアルタイム性損失を回避するための手法を提案し, 計算機シミュレーションにより評価を行う。提案法は従来の集約法よりパケットサイズに対するヘッダサイズで定義されるオーバーヘッド率を 2% 程度増大するものの, データ送信成功率を最大約 3% 程度向上できることを示す。

キーワード 情報収集遅延, パケット集約, WSN, IEEE 802.11ah

## Packet Aggregation Scheme Considering Information Gathering Delay for WSN

Bounpasith CHALEUNSOUK† and Koichi ADACHI†

† Advanced Wireless & Communication Research Center, The University of Electro-Communications

1-5-1 Chofugaoka, Chofu, Tokyo, 182-8585 Japan

E-mail: †{bounpasith, adachi}@awcc.uec.ac.jp

**Abstract** In recent years, with the development of IoT (Internet-of-Things), WSNs (Wireless Sensor Networks) have permeated various parts of our lives. LPWAN (Low Power Wide Area Network), represented by LoRaWAN (Long-Range Wide Area Network) and Wi-SUN (Wireless Smart Utility Network), is a promising candidate as a communication standard using the 920MHz unlicensed band for IoT devices. On the other hand, IEEE 802.11ah also uses the 920MHz unlicensed band and is expected to be applied especially for IoT devices. The IEEE standard has a method of aggregating and transmitting multiple data in one packet. This data aggregation can realize efficient information collection by reducing the amount of overhead size required when transmitting data packets. Since, in IoT applications, the data size per packet is relatively small, the ratio of overhead over the packet size becomes large, resulting in inefficient transmission. This paper proposes a method for avoiding data loss due to packet collision while satisfying the delay constraint of each data. The computer simulation results show that the proposed method is able to improve the success of the data transmission ratio by up to 3% at the cost of increase of overhead ratio by 2% compared to the conventional aggregation method.

**Key words** Information Gathering Delay, Packet Aggregation, WSN, IEEE 802.11ah

## 1. はじめに

近年, IoT (Internet-of-Things) の発展に伴い無線センサネットワーク (WSN: Wireless Sensor Network) が我々の生活の様々な部分に浸透してきている [1]. WSN は, 産業やヘルスケアなどの広い分野へ適用されており, 特に最近では温度や二酸化炭素 CO<sub>2</sub> といった物理環境を定期的に観測するため利用される事例が多い [2].

WSN において, 無線通信を介して情報集約局へと送信される観測情報は, 画像や動画などのデータ量と比較して非常に小さい温度や湿度といった情報である. そのような小さいサイズのデータを送受信する IoT 機器用の無線通信には, ライセンスバンドを用いる規格と免許不要のアンライセンスバンドを用いる規格がある. IoT 機器用の 920MHz 帯アンライセンスバンドを用いる通信規格として, LoRaWAN (Long Range Wide Area Network) [3] や Wi-SUN (Wireless Smart Utility Network) [4] に代表される LPWAN (Low Power Wide Area Network) が挙げられる [5]. これらの通信システムは, 免許不要で扱えるため, だれでも簡単に設置・利用することができる.

また, IEEE 標準規格の一つとしてアンライセンスバンドの 920MHz 帯を利用する IEEE 802.11ah (WiFi Halow<sup>TM</sup>) が挙げられる [6]. IEEE 802.11ah は, 特に IoT 向けの通信システムとして期待されている [7]. 802.11ah は, Wi-Fi ベースの通信規格でありつつも, ISM (Industrial Scientific and Medical) 帯である 2.4GHz/5GHz 帯を用いる従来の Wi-Fi と比べて伝送エリアが広い, 置局免許不要で自営設置が可能といった, 他の LPWAN 通信規格と同等のメリットを有している. さらに, IP 通信が可能である点や動画像伝送にも耐えうる数 Mbps のスループットを達成可能である.

物理環境モニタリングでは, 観測毎に得られるデータ量は一般的にかなり小さいものとなるため, 観測データを逐一パケット化して送信した場合, パケット送信に必須となるヘッダがパケットサイズに占める割合が大きくなるため非効率な伝送となってしまふ. IEEE 標準規格の複数個のデータを集約 (Aggregate) して送信する方法 [8] では, 複数個のデータを端末の送信バッファに集約してデータパケットを生成し, 1つのデータパケットで複数のデータを送る方法である. これにより, データパケットを送信する際に必須となるオーバーヘッドに対するデータ量の比を増大することができ, より効率的な情報集約が可能となる. しかしながら, より多くのデータを集約するとパケットサイズが大きくなることにより, パケット衝突の発生する確率が増大する. また, 観測情報のような情報には生起してから情報集約局が受信するまでに達成されなければならない遅延時間制約が存在するため, これらを考慮したパケット集約手法が必要である. 従来の集約法において, 衝突が発生した場合は格納された全てのデータを破棄し, 端末は再び新しいデータの集約を開始し, 集約可能なデータ数まで集約して送信する. それによって, 前までに受信したデータと次に受信されるデータとの遅延が大きくなり, データのリアルタイム性が失われてしまふ.

そこで本稿では, パケット衝突によるデータ損失ならびにデータのリアルタイム性損失を回避するためのデータ集約手法を提案する. 提案法では, データが生起してから情報集約局が

そのデータを受信するまでに許容できる最大許容遅延時間を定義する. 情報集約局で最大許容遅延時間までに受信できなかったデータは, 有効でないデータとして扱われるため, そのデータは失われることとなる. そこで最大許容遅延時間を利用した2つのパケット集約法を検討する. 計算機シミュレーションにより, 提案手法は従来のデータ集約ありの場合と比較して, 送信に成功したデータ数の割合を最大約 3% 程度向上できることを示す.

本稿の構成は以下の通りとなっている. 第2節でシステムモデルを説明した後に, 第3節で従来のデータ集約ならびにパケット伝送について述べる. 第4節で, 提案手法を説明した後に, その計算機シミュレーション結果を第5節で示す. 第6節はまとめである.

## 2. システムモデル

本稿では, 温度や二酸化炭素 CO<sub>2</sub>, 湿度といった環境情報をモニタリングするために  $M_{\text{dev}}$  台の環境センサが通信エリア内に配置されている環境を想定する. 端末  $m$  ( $0 \leq m \leq M_{\text{dev}} - 1$ ) は一定のデータ生成周期  $T_{\text{tr},m} \sim \mathcal{U}(1, T_{\text{max}})$  [min] に従ってデータを生起し, IEEE 802.11ah 規格を利用して情報集約局に観測情報を送信する. ここで,  $\mathcal{U}(x, y)$  は  $x$  と  $y$  の間の整数値をランダムに生成する関数であり,  $T_{\text{max}}$  [min] は最大データ生成周期である.

端末  $m$  が送信するパケットは, 通信規格固有のヘッダサイズとパケットに含まれるデータサイズから構成される. 本稿では, ヘッダとして Physical Header (PHYHDR) と MAC Header (MACHDR) を想定する. パケットサイズは集約するデータ数によって変化する.  $N$  個のデータを集約して送信した場合のパケット長  $T_{\text{pckt},m}(N)$  [bit] は次式で定義する.

$$T_{\text{pckt},m}(N) \triangleq \frac{8 \times L_{\text{ovh}} + \sum_{n=1}^N 8 \times L_{\text{pl},m}(n)}{R_{\text{data}}} \quad (1)$$

ここで,  $L_{\text{ovh}}$  [byte] はパケットのヘッダサイズ,  $L_{\text{pl},m}(n)$  [byte] は端末  $m$  のデータ  $n$  のサイズ,  $R_{\text{data}}$  [bps] は伝送レートを示す.

## 3. データ集約とパケット送信

802.11 規格においてデータを送信する方法として, データが生成されたら即時にパケットを生起して送信する方法と, 複数個のデータを送信バッファに格納し集約した後でパケットを生起し送信する方法がある.

### 3.1 データ集約法 (A-MSDU) [8]

複数データを集約する方法には, A-MSDU (Aggregates MAC Service Data Unit) と A-MPDU (Aggregates MAC Protocol Data Unit) の2つの方法が存在する. 本検討では, A-MSDU の利用を想定する [6]. IEEE 標準規格の A-MSDU では,  $N$  個のデータを集約し, 一つのパケットを構築した後に送信する. A-MSDU では, 端末の送信バッファに格納されたデータサイズが既定サイズに到達するか, 送信バッファ内で最も古いデータが予め設定された遅延時間を越えた場合に送信される. 図1に示すように, 各データに対する送信元アドレス (SA: Source Address) 及び送信先アドレス (DA: Destination Address) とデータのサイズ (Length) も含めてサブフレームとして扱われ

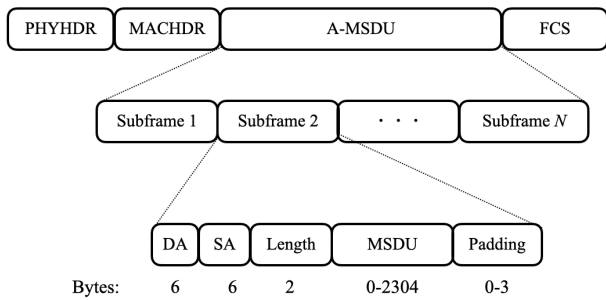


図 1 A-MSDU のパケット集約方法 [9]

る。ここで、送信元及び送信先アドレスとデータのサイズを含んだものをサブフレーム制御といい、送信元と送信先のアドレスはそれぞれ MAC 層の送信アドレスと受信アドレスに一致する。また、サブフレームの最後にある Padding は 0 ~ 3 [byte] となっており、受信側で各サブフレームの区切る箇所がわかるようにするために、サブフレームのサイズを 4 [byte] の倍数になるようにするために挿入される。

### 3.2 集約なし方法と集約あり方法の比較及び問題点

本節では、データ集約なしに送信する場合と、データ集約した後に送信する場合の比較について述べる。本稿では、図 1 に示す PHYHDR と MACHDR をヘッダ、A-MSDU をデータとして、以下のメトリックを考慮する。

- (1) オーバーヘッド率：送信するパケットサイズに対するヘッダサイズの占有率であり、次式のように定義する。

$$R_{\text{ovh},m}(N) \triangleq \frac{L_{\text{ovh}}}{L_{\text{ovh}} + \sum_{n=1}^N L_{\text{pl},m}(n)} \quad (2)$$

- (2) パケット衝突率：複数端末が同一の周波数帯で同時にパケットを送信し、パケットが衝突する確率
- (3) データ損失量：パケット衝突によって損失するデータ数。 $N$  個のデータを集約したパケットが衝突により受信できなかった場合には、 $N$  個のデータ全てが失われるものとする。

#### 3.2.1 オーバーヘッド率

図 2 は、“データ集約なし”と“データ集約あり”，それぞれの方法で  $N = 4$  個のデータを送信する場合を示す。図 2 の (a) に示すように、データ集約なしの場合はデータの生起間隔  $T_{\text{tr}}$  毎にパケットを送信するため、各データの送信にそれぞれオーバーヘッドが必要となる。一方で、図 2 の (b) に示すように、データ集約ありの場合は、 $N$  個のデータを 1 つのパケットに集約して送信するため、1 つのデータに対するオーバーヘッド量を漸近的に  $1/N$  倍に低減することができる。しかし、集約したデータ数分パケット長が長くなるため、他の端末とパケット衝突する確率が高くなる。

#### 3.2.2 パケット衝突率及びパケット損失量

複数の端末が同じ時間で同一周波数帯を利用してパケットを送信した場合、パケット衝突が発生する。データ集約なしの場合、各データを一つのパケットで送信するため、データの生成数が多くなるとパケットの送信数も多くなり、パケット衝突率が高くなる可能性がある。一方で、データ集約ありの場合、複数のデータを一つのパケットで送信するため、各端末の送信パ

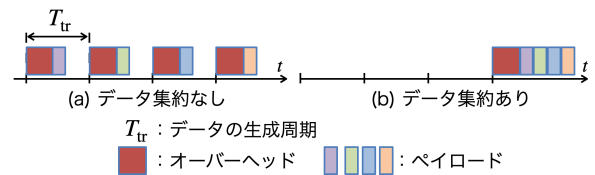


図 2 集約なし・あり方法におけるオーバーヘッド量

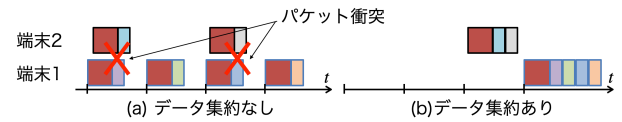


図 3 集約なし・あり方法においてデータ送信によるパケット衝突

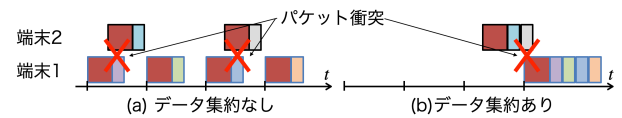


図 4 パケット長の長さによるパケット衝突

ケット数を抑圧することが可能であり、データ集約なしの場合と比較して、パケット衝突確率を低減できると考えられる。

異なるデータの生成周期を持つ端末間のパケット衝突について例を用いて説明する。図 3 に、2 つの端末がシステム内に存在し、端末 1 は  $T_{\text{tr},1} = 1$  [min] 毎、端末 2 は  $T_{\text{tr},2} = 2$  [min] 毎にデータを生成する場合を示す。各端末には固有の送信オフセットが割り当てられているものの、それらが非常に近い場合を考える。データ集約なしの場合、図 3 の (a) に示すように、端末 1 のデータ #1 と端末 2 のデータ #1、端末 1 のデータ #3 と端末 2 のデータ #2 を送信するパケットがそれぞれ衝突してしまう。一方で、データ集約ありの場合、図 3 の (b) に示すように、端末 1 は 2 個のデータを集約してパケットを送信、端末 2 は 4 個のデータを集約してパケットを送信するため、それぞれの端末が送信するパケットの時間が重ならないためパケット衝突を回避できる。しかしながら、複数データを集約して一つのパケットで送信するため、パケット長が長くなり、パケット衝突率が增大する可能性もある。図 4 の例では、端末 1 が 2 個のデータを集約したため、パケット長が長くなり、端末 2 のパケットと衝突してしまっている。

また、パケット衝突が発生した場合、データ集約なしの場合、衝突したパケットに含まれる 1 つのデータのみの損失で済むが、データ集約した場合には、集約された複数データが損失してしまう。図 4 に示す例では、データ集約なしの場合、端末 1 のデータ #1 と #3、端末 2 はデータ #1 と #2 の損失で済む。一方、データ集約した場合、端末 A と B の全てのデータが損失してしまう。

## 4. 提案手法

パケット衝突によりデータが損失したり、データが生起してからそのデータが含まれたパケットが送信されるまで大きな遅延が発生すると、情報のリアルタイム性が失われてしまう。従来のデータ集約法では、パケットに格納可能な最大集約可能データ数  $N_{\text{max}}$  までデータを集約してから送信している。その

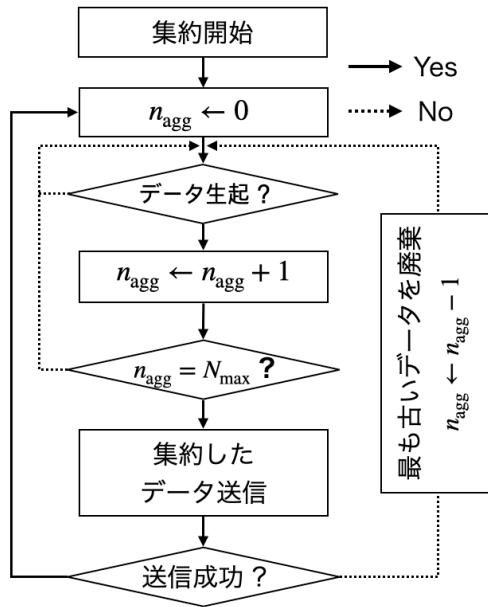


図5 提案法1のアルゴリズム

ため、パケット送信時に衝突が発生した場合、衝突が発生したパケットに格納されている全てのデータが失われることになる。パケット衝突が発生した場合、端末は新しいデータの集約を開始し、再び最大集約可能データ数まで集約して送信する。そこで本稿では、パケット衝突によるデータ損失ならびにデータのリアルタイム性損失を回避するための手法を提案する。

まず、データが生起してから情報集約局がそのデータを受信するまでに許容できる端末  $m$  の最大許容遅延時間  $T_{\text{delay},m}$  を次式のように定義する。

$$T_{\text{delay},m} \triangleq N_{\text{max}} \times T_{\text{tr},m} \quad (3)$$

もし仮に、端末  $m$  のデータが時刻  $t$  で生起してから  $t + T_{\text{delay},m}$  までに情報集約局で受信されなかった場合、そのデータは許容遅延時間を超過したものとし、データとして有効ではないものとする。すなわち、従来のように先頭データの最大許容遅延時間までデータを集約してからパケットを送信し、パケット衝突が発生した場合、その先頭データは失われることとなる。また、3.2.2節で述べたように、データ集約数を大きくした場合には、パケット長が長くなるため、パケット衝突率が増大するため、この影響がより顕著となる。そこで、本稿では、最大許容遅延時間ならびにオーバーヘッド率を考慮したパケット集約法を検討する。

#### 4.1 提案法1

提案法1では、従来のデータ集約法と同じく  $N_{\text{max}}$  個のデータを集約してパケットを送信する。パケット衝突が発生した場合には、衝突パケットに含まれている最新の  $N_{\text{max}} - 1$  個のデータと新たに生起したデータを集約して、新たなパケットとして送信する。これにより、パケット衝突が発生した場合にも、次のパケットが受信成功すれば  $N_{\text{max}}$  個のデータを最大許容遅延時間内に受信することが可能となる。図5に提案法1のアルゴリズム、図6に従来のデータ集約ありの場合と提案法1の比較の例を示す。

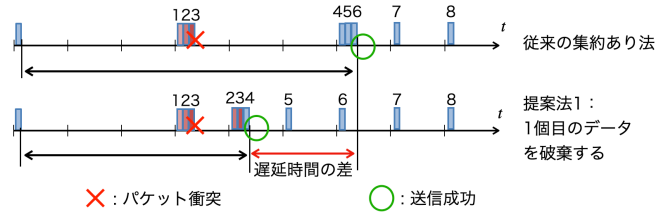


図6 従来の集約あり法と提案法1の比較例

#### 4.2 提案法2

提案法2では、データの最大許容遅延時間とオーバーヘッド率のバランスを考慮したデータ集約を行う。具体的には、ある規範に従って最大集約可能データ数  $N_{\text{max}}$  より少ない数  $N_{\text{agg}} (< N_{\text{max}})$  のデータを集約してパケットを送信する。もし仮に、パケット衝突が発生した場合でも、衝突したパケットに含まれているデータは許容遅延時間内であるため、最初に集約したデータを破棄せずに新たに生起したデータを追加してパケットを送信する。

図7に提案法2のフローチャートを示す。まず端末は集約データ数  $N_{\text{agg}}$  を  $N_{\text{agg}} < N_{\text{max}}$  の条件下で決定する。本稿では初期検討として、 $N_{\text{agg}} = \lceil N_{\text{max}}/2 \rceil$  とする。各端末は、 $N_{\text{agg}}$  個のデータが生起した後に、パケット生成し送信する。もし仮に、パケット衝突が発生した場合には、新たに生起したデータをパケットに追加し  $N_{\text{agg}} + 1$  個のデータを格納したパケットを生成する。パケットが正しく受信された場合には、端末は新たにデータ集約を開始する。一方、パケットが正しく受信されなかった場合には、集約データ数が  $N_{\text{max}}$  となるまで、パケットの生成と送信を繰り返す。もし仮に  $N_{\text{max}}$  個のデータを格納したパケットが正しく受信されなかった場合には、集約した全データを破棄し、最初の  $N_{\text{agg}} = \lceil N_{\text{max}}/2 \rceil$  に戻して、再びデータ集約を開始する。このように、最初に最大集約可能データ数より少ないデータ数で集約して送信することで、パケット長の長さが最大まで集約より比較的短いいため、衝突率を低減できる可能性がある。また、衝突が発生したとしてもすぐに衝突したデータを破棄せずに、もう一回送ることによってデータの遅延時間を抑えることが可能だと考えられる。図8に提案法2の比較の例を示す。

### 5. シミュレーション評価

#### 5.1 シミュレーションパラメータ

計算機シミュレーションにより提案手法の有効性の評価を行う。シミュレーション回数を10000回とし、各シミュレーションにおける観測時間を720 [min]とする。端末数は  $M_{\text{dev}} \in \{10, 50, 100, 200, \dots, 500\}$  とする。各端末の最初の送信タイミングは最大送信時間オフセット  $T_{\text{tr,time}} = 60$  [sec] 以内でランダムに決定し、パケット衝突によって提案手法の有効性を示すため、最大データ生成周期は  $T_{\text{max}} \in \{3, 5, 10\}$  [min] とした。1つのデータサイズは実際の環境センサのデータサイズを想定し、 $L_{\text{pl}} = 80$  [byte] とし、オーバーヘッドサイズは802.11ahのMAC層などのヘッダのサイズを想定し、 $L_{\text{ovh}} = 20$  [byte] とした[6]。また、802.11ahにおける符号率1/2とQPSK (Quadrature Phase Shift Keying) 変調を想定し、伝送レートは  $R_{\text{data}} = 600$  [kbps] とした[6]。3.2.2節で述べたように、

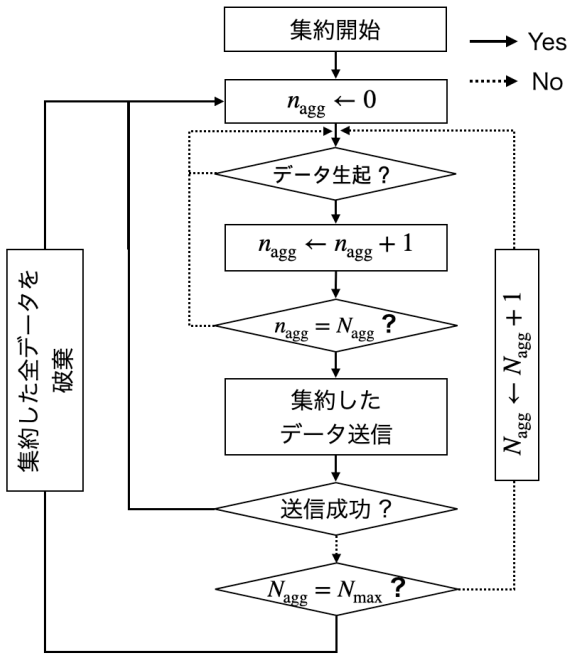


図7 提案法2のアルゴリズム

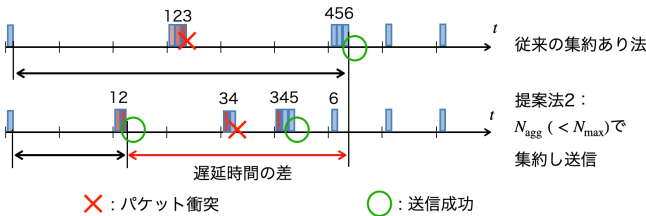


図8 提案法2の例

表1 シミュレーションパラメータ

変数	値
シミュレーション回数	10000 回
シミュレーション時間	720 [min]
センサ端末数 $M_{dev}$	{10, 50, 100, ..., 500}
最大送信時間オフセット $T_{tr,time}$	60 [sec]
最大パケット生成周期 $T_{max}$	{3, 10}[min]
データサイズ $L_{pl}$	80 [byte]
オーバーヘッドサイズ $L_{ovh}$	20 [byte]
伝送レート $R_{data}$	600 [kbps]
最大集約可能データ数 $N_{max}$	{3, 5, 10}

データ集約数が多いほどパケットサイズが大きくなるため、パケット衝突の発生する可能性が高くなる。そのため、最大集約可能データ数を  $N_{max} \in \{3, 5, 10\}$  とした時の特性を取得した。

## 5.2 シミュレーション結果

### 5.2.1 送信に成功したデータ数の割合

図9に、提案法1 (“Proposed 1”)と提案法2 (“Proposed 2”)を用いた場合に得られるネットワーク内の端末数に対する遅延制約時間内に集約成功したデータ数の割合を示す。比較のために、集約を行わず各観測データを一つのデータパケットで送信する場合 (“Conv. w/o Agg.”)と常に  $N_{max}$  個の観測データを一つのデータパケットに集約して送信する場合 (“Conv.

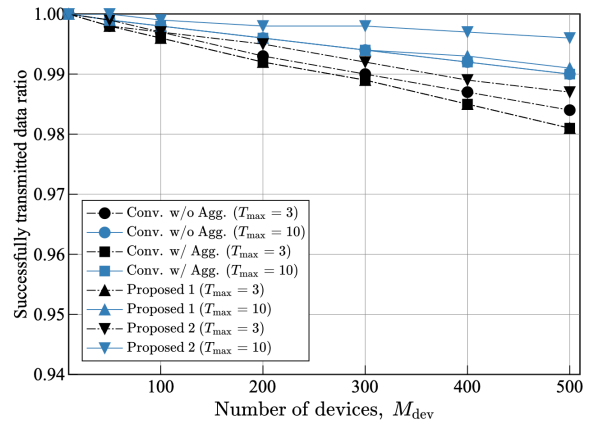


図9 最大集約数 ( $N_{max} = 3$ )

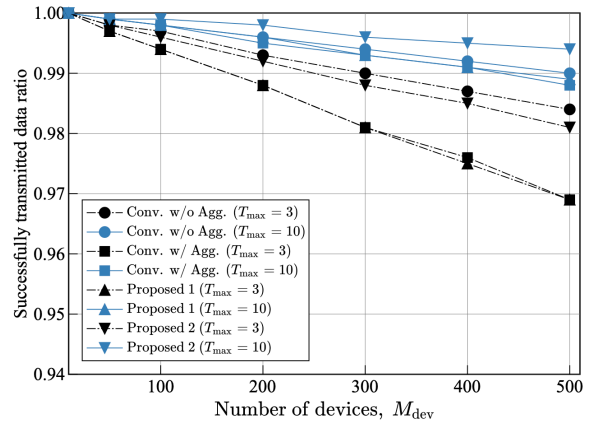


図10 最大集約数 ( $N_{max} = 5$ )

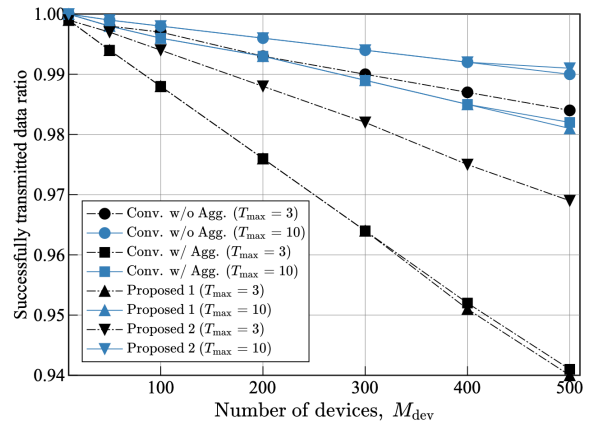


図11 最大集約数 ( $N_{max} = 10$ )

w/ Agg.”)の特性も掲載する。従来の常に  $N_{max}$  個の観測データを集約する手法では、集約を行わない場合と比較して最大データ生成周期  $T_{max}$  に依らず常に集約成功データ数の割合が劣化していることが分かる。これは、以下のような理由による。データ生成周期が一致した端末同士では、パケット衝突が発生する可能性が高くなるが、集約しない場合にはパケット長が短いため、パケット衝突を回避できる可能性が高くなる。その一方で、常に  $N_{max}$  個の観測データを集約する場合には、パケット長が長くなるためパケット衝突の発生する確率が高くなるた

めである。

提案手法 1 を用いた場合、集約を行わない場合と比較して送信成功データ数割合を向上できていないことが分かる。この理由として、従来の常に  $N_{\max}$  個のデータを集約する方法と同様に、パケット長が長くなるためパケット衝突の発生する確率が高くなっていることがある。また、提案手法 1 では、衝突が発生した場合に一番古いデータを破棄して次に生じたデータを集約して送信するため、頻繁な送信が発生する。そのため、データ集約数が多い場合 ( $N_{\max} = 5, 10$ ) にはパケット長が長くなり、かつ頻繁な送信の発生により衝突が発生する確率が高くなるため、従来のデータ集約をしない場合と比較して特性が劣化している。

一方で、提案手法 2 は、常に最大集約可能データ数を集約してパケットを送信する従来法と比較して、最大集約可能データ数ならびに最大データ生成周期に依らず常に優れており、特に端末数が  $M_{\text{dev}} = 500$  の時に従来のデータ集約ありの場合と比較して最大 3% 程度向上できている。これは、最大集約可能データ数より少ない数  $N_{\text{agg}} = \lceil N_{\max}/2 \rceil$  のデータ数を集約して送信することで、従来の  $N_{\max}$  個の観測データを集約する従来法と比較してパケット長が短くなり、パケット衝突を回避することが可能となるためである。また、データ集約を行わない場合と比較して、最大データ生成周期  $T_{\max}$  が大きい場合に、送信成功データ数割合を向上できていることが分かる。これは、データを集約することにより、各端末からの送信パケット数が少なくなるため、パケット衝突が減少したためと考えられる。

### 5.2.2 平均オーバーヘッド率

各手法の最大集約可能データ数  $N_{\max}$  に対する式 (2) で定義される平均オーバーヘッド率を図 12 に示す。図 12 から分かるようにデータ集約を行わない場合、 $L_{\text{pl}}(n) = 80$  [byte] のデータに対して、 $L_{\text{ovh}} = 20$  [byte] のオーバーヘッドサイズを付加してパケットを生成しているため平均オーバーヘッド率は  $R_{\text{ovh}}(1) = 0.2$  となる。一方で、従来のデータ集約を行う場合 (A-MSDU) と提案手法 1 はどの最大集約可能データ数でも平均オーバーヘッド率が最小となっている。これは、常に最大集約可能データ数  $N_{\max} = 10$  個のデータを集約してパケットを生成しているためである。

提案手法 2 は、従来の A-MSDU と提案手法 1 と比較して、オーバーヘッド率が比較的高くなっていることが分かる。これは、最大集約可能データ数  $N_{\max}$  より少ない  $N_{\text{agg}}$  個のデータを集約してパケットを送信しているためである。しかしながら、提案手法 2 は従来のデータ集約なしの場合と比較して大幅にオーバーヘッド率を低減できている。また、5.2.1 で示したように、提案手法 2 は従来のデータ集約なしの場合とほぼ同等もしくはそれ以上の送信成功データ数割合を達成できていることから、提案手法 2 は他の手法と比較して、データの最大許容遅延時間とオーバーヘッドのバランスを取ることができているといえる。

## 6. まとめ

本稿では、WSN における情報収集遅延を考慮し、パケット衝突によるデータ損失ならびにデータのリアルタイム性損失を回避するためのデータ集約手法を提案した。提案手法は最大許容遅延時間を定義し、その最大許容値時間を利用してパケット

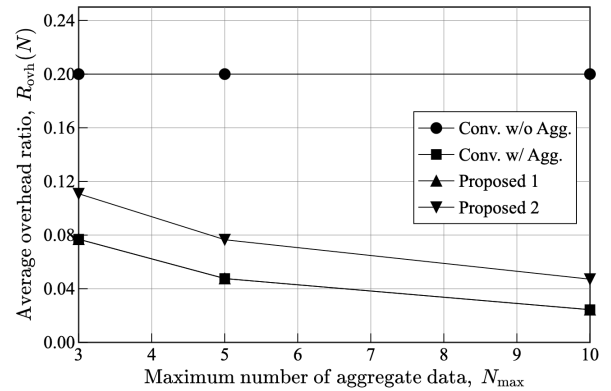


図 12 平均オーバーヘッド率

集約ならびに再送制御を行う。提案手法では、衝突が発生しても集約した全てのデータを保管し、次に生じたデータを追加してパケットを生成するため、多くのデータの最大遅延時間制約を満たすことを可能としつつ、送信成功データ割合を増大できる。計算機シミュレーションにより、提案手法は、従来のデータ集約法と比較してデータに対するオーバーヘッド量を少し増大させるものの、高い送信成功データ数割合を実現できることを示した。

## 文 献

- [1] D. Baghyalakshmi, J. Ebenezer, and S. A. V. Satyamurty, "WSN based temperature monitoring for high performance computing cluster," in Proc. 2011 Int. Conf. on Recent Trends in Information Technol. (ICRTIT), pp. 1105-1110, Jun. 2011.
- [2] M. F. Othman, K. Shazali, "Wireless Sensor Network Applications: A Study in Environment Monitoring System," *Procedia Engineering*, pp. 1204-1210, vol. 41, 2012
- [3] A. J. Wixted, P. Kinnaird, H. Larjani, A. Tait, A. Ahmadi, and N. Strachan, "Evaluation of LoRa and LoRaWAN for wireless sensor networks," *IEEE SENSORS*, 2016, pp. 1-3, Oct. 2016.
- [4] H. Harada, et al, "IEEE 802.15.4g based Wi-SUN communications systems," *IEICE Trans. Commun.*, vol.E100-B, No.7, pp.1032-1043, Jul. 2017.
- [5] O. Georjoui and U. Raza, "Low power wide area network analysis: Can LoRa scale?," *IEEE Commun. Lett.*, vol.6, no.2 pp. 162-165, 2017.
- [6] "IEEE Standard for Information technology – Telecommunications and information exchange between systems - Local and metropolitan area networks–Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 2: Sub 1 GHz License Exempt Operation," in IEEE Std 802.11ah-2016 (Amendment to IEEE Std 802.11-2016, as amended by IEEE Std 802.11ai-2016) , pp.1-594, May 2017
- [7] N. Ahmed, D. De, F. A. Barbhuiya and M. I. Hussain, "MAC Protocols for IEEE 802.11ah-based Internet of Things: A Survey," in IEEE Internet of Things Journal, Aug. 2021
- [8] D. Skordoulis, et al, "IEEE 802.11n MAC frame aggregation mechanisms for next-generation high-throughput WLANs," *IEEE Wireless Commun.*, vol. 15, no. 1, pp. 40-47, Feb. 2008
- [9] T. Selvam and S. Srikanth, "A frame aggregation scheduler for IEEE 802.11n," in Proc. 2010 National Conf. On Commun. (NCC), pp.1-5, Jan. 2010