

# 時空間データ滞留のための滞留データ特性を考慮した 動的送信間隔決定手法の提案

山崎 駿平<sup>†</sup> 野林 大起<sup>††</sup> 塚本 和也<sup>††</sup> 池永 全志<sup>††</sup>

<sup>†</sup> 九州工業大学大学院情報工学研究院 〒820-8502 福岡県飯塚市川津 680-4

<sup>††</sup> 九州工業大学大学院工学研究院 〒804-8550 福岡県北九州市戸畑区仙水町 1-1

E-mail: <sup>†</sup>p232087s@mai.kyutech.jp, <sup>††</sup>{nova,ike}@ecs.kyutech.ac.jp, <sup>†††</sup>tsukamoto@cse.kyutech.ac.jp

あらまし IoT 技術の発展に伴い、様々なアプリケーションデータが生成される。それらのデータの中には、特定の場所や特定の時間でのみ必要とされるデータ（時空間データ）が存在する。先行研究では、「時空間データの地産地消」を実現するための新しいネットワークの構築を目指し、時空間データを広範囲に素早く配信、滞留させるため、車両を用いた時空間データ滞留方式を提案した。この方式は、車両密度により動的に決定されたデータ送信確率に加え、複数の異なるデータを一括して送信することで、空間全体のデータ送信数の削減を実現した。しかし、時空間データは情報発信源ごとに滞留期間、滞留完了までの許容待ち時間が異なる事が予想される。そこで本研究では、先行研究で考慮していなかった滞留期間、及び滞留完了までの許容待ち時間に基づく動的送信間隔決定手法を提案する。そして、シミュレーションにより、目標の滞留完了時間までに滞留エリア内のカバー率を 100% にし、滞留期間後はデータを迅速に廃棄出来る事を確認した。

キーワード 時空間データ、データ滞留、動的送信間隔決定

## Proposal of Dynamic Transmission Interval Control Method by Considering the Requirements for Spatio-temporal Data Retention

Shumpei YAMASAKI<sup>†</sup>, Daiki NOBAYASHI<sup>††</sup>, Kazuya TSUKAMOTO<sup>††</sup>, and Takeshi

IKENAGA<sup>††</sup>

<sup>†</sup> Department of Computer Science and Electronics, Faculty of Computer Science and Systems, Kyushu  
Institute of Technology

680-4 Kawazu, Iizuka-shi, Fukuoka, 820-8502, Japan

<sup>††</sup> Department of Electrical Engineering and Electronics, Faculty of Engineering, Kyushu Institute of  
Technology

1-1 Sensui-cho, Tobata-ku, Kitakyushu-shi, Fukuoka, 804-8550, Japan

E-mail: <sup>†</sup>p232087s@mai.kyutech.jp, <sup>††</sup>{nova,ike}@ecs.kyutech.ac.jp, <sup>†††</sup>tsukamoto@cse.kyutech.ac.jp

**Abstract** With the development and the diffusion of Internet of Things (IoT) technologies, various types of data are generated for IoT applications anywhere and anytime. We defined IoT device data that depends heavily on generation time and location as spatio-temporal data (STD). In the previous works, we have proposed the data retention system using vehicular networks to achieve the paradigm of “local production and consumption of STD.” The system can achieve to provide STDs quickly to users within a specific location by retaining the STD within the location, however, does not consider that each STD has different requirements for the data retention. In particular, the lifetime of the STD and the diffusion time to the whole location directly affect the performance of data retention. Therefore, we propose a dynamic transmission interval control method for the data retention system by considering the requirements. By the simulation evaluation, we found that our proposed method can satisfy the requirements of STD and maintain a high coverage rate in the location.

**Key words** Spatio-temporal Data, Data Retention, Dynamic Transmission Interval Control

## 1. はじめに

IoT(Internet of Things) 技術の発展に伴い、多くの IoT デバイスや新しいアプリケーションが爆発的に増加している。現在のネットワークアーキテクチャは集中管理型で [1], IoT デバイスから生成される膨大なアプリケーションデータを処理するには、高性能 CPU や大容量ストレージを搭載した計算機が必要となる。そのため、現在のネットワークインフラでは指数的に増加するデータトラフィックに対応できないことが予想される。

IoT デバイスが生成するデータの中には、発生した時間や空間に依存するものがあると考えられ、先行研究 [2] ではこのようなデータを時空間データと定義している。例えば、交通情報、気象情報、災害情報、時限的な店舗広告などの情報はその状況がリアルタイムに変化するため、時空間データであるといえる。このようなデータは、インターネットに接続された遠隔のサーバに蓄積するよりも、データが発生したその空間で活用する方が効率的であるケースが存在する。つまり「データの地産地消」を実現することにより、ネットワークインフラが抱える問題を解決することも可能になると考えられる。

既存のネットワークインフラを用いずに「データの地産地消」を実現するには、新しい情報ネットワークの構築が求められる。先行研究ではこの新しい情報ネットワークの構築のため、無線機器搭載の車両から構成される VANET(Vehicle Ad-hoc Network: 自動車アドホックネットワーク) に注目している。車両をデータ配信の中継ノードとすることで、リアルタイムな情報の取得、情報の分散管理による耐障害性の向上、既存ネットワークインフラへの負荷軽減を可能とする。さらに先行研究では、VANET を用いて時空間データを留めて配信する「時空間データ滞留システム」を構築した。また、効果的にデータ滞留を実現するためノード密度運動型データ送信制御を提案している [2] [3]。しかし、実環境で滞留させる時空間データは、滞留期間や滞留完了までの許容時間が異なる事が予想され、先行研究では効果的な滞留が実現できない可能性がある。

そこで本研究では、時空間データに滞留期限や拡散速度などの新たな要求に対応するための動的送信間隔決定手法を提案する。本提案手法では情報発信者が指定する滞留時間内にカバー率 100% を達成するために初期送信間隔を調整し、さらにはデータの滞留期限に応じてデータを削除することで空間内に滞留するデータを抑制する。そして本提案手法の有効性をシミュレーションにより評価する。

2 節では関連研究について述べ、3 節では先行研究で提案された時空間データ滞留システムについて述べる。4 節で提案手法を紹介し、5 節でカバー率やデータ送信数の観点から、シミュレーションにより提案手法の有効性を示す。最後に 6 節でまとめと今後の課題について述べる。

## 2. 関連研究

文献 [4] では、VANET のデータ拡散・共有の問題について議論している。これを踏まえ、文献 [4] は、特定のエリア内で位置に基づく Geocast Routing を使い、情報発信源から全てのノード(車両) に送信するプロトコルを提案している。さらに文献 [5] では、特定のエリア内のノードにデータを送信し、

一定期間保存する方式を提案している。文献 [5] は滞留エリアの位置に依存した geocast message を保持/配信する 3 つの方式、1. server approach, 2. election approach, 3. neighbor approach を提案している。まず 1. server approach では、固定されたサーバがデータ保持の役割を担い、geocast routing protocol に基づき滞留エリアにデータを送信する。サーバは各ノードの位置情報を交換する必要がある、負担が大きくなる可能性がある。また、文献 [6] では SDN (Software Defined Network) を用いた最適な滞留エリア範囲の設定方法が提案されている。サーバが SDN コントローラとなり、周辺ノードの移動情報から最適な滞留エリア半径を決定し送信を行う。次に 2. election approach では、選ばれたノードがデータを保持し、滞留エリアにデータを送信する。これら 2 つの方式は、特定のサーバ/ノードへの負担が大きく、故障によるデータ配信の継続ができない可能性がある。最後に 3. neighbor approach は、移動ノードのみで構成されるシステムである。文献 [7] では自然災害時に、インフラを用いず移動ノード間で災害情報を交換・共有するための実用的なアプリケーションが提案されている。他、文献 [8], Floating Content [9], Locus [10], 文献 [3] などが提案されている。文献 [8] では各ノードのナビ情報の交換により滞留エリアに向かうノードを特定し、効果的にデータを送信する。また Floating Content, Locus の方式では、保持しているデータのリストを各ノードが持ち、隣接ノードとデータリストを交換する。保持していないデータは、隣接ノードに送信要求を行いデータを取得する。このときノードが送信を行うかは、中心からの距離に従う送信確率で決まるため、中心から遠いときデータ取得の減少や、ノードが中心付近に偏っているときにデータ衝突が頻発し得る。

そこで文献 [3] では、関連研究の問題点を解決するため、位置情報に基づくブロードキャストを用いて、滞留エリア内に存在する全ての受信者に、設定された送信間隔でデータを定期的に送信することを目指し、時空間データ滞留システムの構築を行った。そこで次節では、その時空間データ滞留システムの構築にあたって考案された手法の説明を行う。

## 3. 時空間データ滞留システム

本節では、先行研究 [2] [3] で用いられたシステムの前提条件、目標、要件について述べ、さらにデータ送信制御手法について説明を行う。

### 3.1 前提条件

時空間データ滞留システムは、各ノードが GPS を用いて位置情報を取得しているものとする。各ノードは自身の ID を含んだビーコンを一定周期でブロードキャスト送信する。次にデータに関して、ノードはデータをブロードキャスト送信する。データには滞留エリアの情報(中心座標、滞留半径  $R$ )、データ送信間隔  $d$  が含まれているものとする。ノードは自身の位置情報とデータに含まれる滞留エリア情報から、自身が滞留エリアの外か内どちらに位置しているのか判断する。

### 3.2 システムの目標と要件

このシステムの目標は、滞留エリア全体にデータを定期的に拡散し、滞留させることである。よって、システムユーザが滞留エリア内に入ると自動的にデータを受信できる。また、既存ネットワークインフラを使わずデータを分散管理するため、

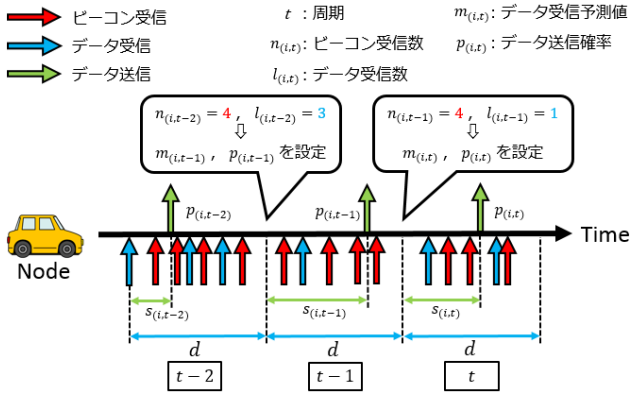


図 1: データ送信確率の設定

サーバの負荷軽減やデータの耐障害性の向上が期待できる。システムの要件として、システムユーザが滞留エリア内に入ると自動的にデータを受信できる確率を示すカバー率を定義している。

$$\text{カバー率} = \frac{S_{DT}}{S_{TA}}. \quad (1)$$

$S_{DT}$  はノードが一定期間内で送信したデータを受信可能な面積、 $S_{TA}$  は滞留エリア全体の面積を示す。

先行研究[2]では、なるべく少ない送信回数で高いカバー率を保つため、ノード密度に応じて送信確率を設定し無駄なデータ送信を抑制する手法を提案している。この手法の説明を 3.3 節で行う。

### 3.3 ノード密度連動型データ送信制御

このシステムでは、無線到達範囲を  $r$ 、滞留エリア中心座標からノードまでの距離を  $Distance$  とし、以下のように送信対象エリアを定義している。

$$\begin{cases} 0 < Distance \leq R + r. & : \text{送信対象エリア} \\ otherwise. & : \text{送信対象エリア外} \end{cases} \quad (2)$$

データを初めて受信した時刻を基準に、データ送信間隔  $d$  を 1 周期として定期的にデータを送信する。このとき、滞留エリア外に存在する一部のノードもカバー率向上のためにデータ送信を行う。なお、実際の送信タイミングは各ノード  $v_i$  ( $i$ : ノードに付与される固有の ID) が  $d$  内でランダムに決定した時刻  $s(i,t)$  に従って送信することで、ノード間の無線チャネルの衝突を抑制する。

次に、全ノードは自身のノード ID を含んだビーコンを  $b$  [s] 間隔で送信する。そうすることで、自身の周りの隣接ノード台数を知ることができる。隣接ノード台数  $n(i,t)$  と、 $t-1$  周期で受け取ったビーコン数を基に、 $t$  周期ごとに送信確率の設定を行う。(図 1) このとき、以下の 2 つの場合を考える。

**case 1:**  $n(i,t-1) \leq 3$

隣接ノード台数が 3 台以下のとき、ノードは必ずデータを送信するために送信確率  $p(i,t) = 1$  と設定する。

**case 2:**  $n(i,t-1) \geq 4$

$p(i,t)$  は隣接ノード台数と周辺のデータ送信状況に応じて計算される。データ送信確率を調整するために  $t$  周期でのデータ受信数の予測値として、 $m(i,t)$  を求める (式 (3))。

$$m(i,t) = \alpha * l(i,t-1) + (1 - \alpha) * m(i,t-1). \quad (3)$$

この  $m(i,t)$  はデータ予測値、 $l(i,t-1)$  は  $t-1$  周期で受信したデータ数、 $\alpha$  は移動平均係数を表している。 $m(i,t)$  と送信間隔  $d$  内で送信されるデータ送信目標値である  $\beta$  と比較して、送信数  $\beta$  個になるよう、式 (4) に従う確率によって調整する。

$$p(i,t) = \begin{cases} p(i,t-1) + \frac{\beta - l(i,t-1)}{n(i,t-1) + 1} & (0 < m(i,t) < \beta) \\ p(i,t-1) & (m(i,t) = \beta) \\ p(i,t-1) - \frac{l(i,t-1) - \beta}{n(i,t-1) + 1} & (m(i,t) > \beta) \end{cases} \quad (4)$$

初期送信確率は、 $\frac{\beta}{n(i,t-1) + 1}$  とする。また、目標値  $\beta$  個になるように確率調整をするが、増減幅が負の場合 ( $p(i,t) = p(i,t-1)$ ) は確率調整は行わない。このとき、データ送信確率の範囲は  $\frac{\beta}{n(i,t-1) + 1} \leq p(i,t) \leq 1$  である。

### 3.4 先行研究の問題点

先行研究では、ユーザが滞留エリアに進入することで、データを得ることができるまでの許容待機時間を設定するために、ノードに送信間隔  $d$  でデータを送信させることで実現した。つまり滞留エリアに入って  $d$  秒間待機すればデータを受信することができるということである。しかしこの許容待機時間に関しては考慮はされていないため、送信間隔  $d$  をどのように設定するかについては言及されていない。

しかし、今後滞留データの種類が増加し、天気や広告 (タイムセール) のような時空間データよりも、データ活用の期限と拡散に要する時間に対して厳しい制約がある情報が滞留される事が予想される。例として、無線資源情報 (利用可能な周波数帯、干渉発生) などが挙げられる。これらの情報に滞留時間が設定されたとき、送信間隔のほうが滞留時間より長く設定されてしまうと ( $d > \text{滞留時間}$ )、滞留エリア全体にデータが行きわたる前に滞留時間を過ぎて、ユーザがデータを取得できないという問題が発生してしまう。よって  $0 < d < \text{滞留時間}$  となるように送信間隔を設定すべきである。

しかし、情報発信源がデータを送信した時刻 ( $t = 0$  秒とする) から時刻  $d$  秒までの  $d$  秒間は、隣接ノード台数が少なく滞留半径が大きいという状況では、滞留エリア全体にデータが広がり切れず、その後データを取得したとしても、コンテンツによっては滞留時間内にそのデータを利用できない可能性がある。

また、時間的制約が厳しい情報が滞留時間後に送信されてしまうと、無駄なデータ送信数の増加やデータ受信者が価値のなくなった情報を利用してしまふなどの問題が発生する恐れがある。これらの問題を踏まえて次節では、滞留完了までの待機時間の最小化、滞留期間終了後の迅速なデータ削除を実現する手法を提案する。

## 4. 提案手法

本提案手法では、3.3 節で述べた、先行研究[2]のノード密度連動型データ送信制御を用いる。そして、カバー率を高く保つために隣接ノード台数に応じて情報発信源が動的にデータ送信間隔を決定する手法を提案する。

### 4.1 本研究における送信間隔

まず、データの生存時間のことを滞留時間  $T_R$  と定義し、この時間以降は時空間データのデータの拡散、及び使用を制限

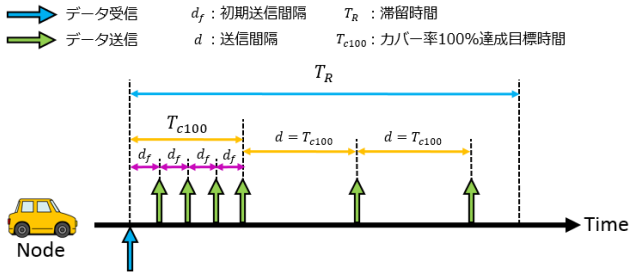


図 2: 滞留時間内のデータ送信間隔

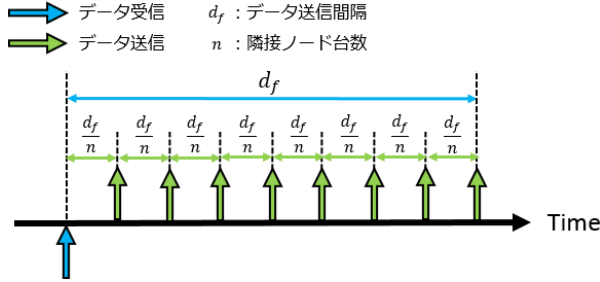


図 3: ランダム時間選択 (\$n=8\$ のとき)

する。また、滞留完了までの許容時間の目標として、カバー率 100% 達成目標時間  $T_{c100}$  を定義する。コンテンツによってこの  $T_{c100}$  に対する要求は異なるため、 $T_{c100}$  の取り得る範囲は  $0 < T_{c100} \leq T_R$  となる。例えば、滞留時間が長い情報は、滞留時間の中で 1 度でもデータを受信できれば、その情報を利用できるので、滞留時間  $T_R$  に対するカバー率 100% 達成目標時間  $T_{c100}$  の割合はある程度大きくても許容可能である。しかし、データの滞留時間  $T_R$  が短い場合、 $T_{c100}$  に大きい値を設定してしまうと、データを受信できてもデータを利用する前に滞留時間  $T_R$  を経過する可能性が出てくる。例として、利用可能な周波数帯は刻々と変化するので滞留時間は短くなり、 $T_{c100}$  秒以内に滞留エリアのカバー率が 100% を達成できなければ、滞留エリア内に存在するノードやユーザがデータを受信し、利用することができない。この問題を解決するには、最初の  $T_{c100}$  秒間の初期送信間隔は、隣接ノード台数や滞留エリアの大きさによって、動的に設定される必要がある。 $T_{c100}$  秒を過ぎた後はデータの送信数を抑えるために、送信間隔を  $d = T_{c100}$  と設定する。本研究では、0 から  $T_{c100}$  の間のみの初期送信間隔を  $d_f (< d = T_{c100})$  と定義し、その決定方法を提案する (図 2)。

## 4.2 動的初期送信間隔決定

### 4.2.1 ランダム時間とホップ数の考慮

先行研究では、送信間隔  $d$  内で送信されるデータ送信目標値を  $\beta$  としてきた。隣接ノード台数を  $n$  とすると、1 回の送信でデータを受信する台数は  $n$  台で、それぞれのノードがデータに含まれている送信間隔  $d$  を確認し、ランダム時間  $s (0 < s \leq d)$  [s] 後に送信を行う。ランダム時間の平均は、 $s \triangleq 0.5d$  となるので、ほぼ  $n$  台のノードが同時にデータを受け取ったと考え、ランダム時間が均等に選択されるとする (図 3)。

滞留エリアをカバーするために最低限必要な送信目標回数は  $\beta$  なので、 $0 < s \leq d$  のうち最初の  $\beta$  回の送信に注目すると、 $\frac{d_f}{n} * \beta$  [s] が有効な 1 ホップまでの時間である。よって、カバー

率 100% 達成目標時間  $T_{c100}$  [s] 以内に可能なホップ数は以下の式で表せる。

$$\frac{T_{c100}}{\frac{d_f}{n} * \beta} = \frac{nT_{c100}}{\beta d_f}. \quad (5)$$

1 回のホップで滞留エリアをカバーできる距離を  $x$  [m] とすると、滞留エリア全体にデータを配信するためには、式 (5) のホップ数を用いて以下の式 (6) で初期送信間隔を設定する必要がある。

$$x \times \frac{nT_{c100}}{\beta d_f} = R$$

$$d_f = \frac{n}{\beta} T_{c100} \times \frac{x}{R}. \quad (6)$$

### 4.2.2 ノード配置の考慮

前提として全てのノードは場所によらず一様に存在し、情報発信源とノードの無線到達範囲  $r$  は同じとする。初期送信間隔  $d_f$  の決定方針として、情報発信源からノードまでの距離が重要となり、以下の 3 つの場合を考える。

- (1) O\_case : 楽観的 (Optimistic) に考える場合
- (2) E\_case : 期待値 (Expected value) が最大の場合
- (3) P\_case : 悲観的 (Pessimistic) に考える場合

まず、(1) **O\_case (楽観的に考える場合)** とは、ノードが情報発信源の無線到達範囲  $r$  上に存在している状態である。このノード配置の場合、常に無線到達範囲  $r$  でデータが伝搬するため、最小ホップ数で滞留エリアの端まで電波が届くことになる。

次に、(2) **E\_case (期待値が最大の場合)** である。ノードは情報発信源を中心とした無線到達範囲  $r$  内に一様に存在しているので、中心からそれぞれのノードまでの距離の期待値は、最大で  $\frac{1}{\sqrt{2}}r$  となる。この長さは、半径  $r$  の円の面積の  $\frac{1}{2}$  倍になる半径である。

最後に、(3) **P\_case (悲観的に考える場合)** とは、期待値が最大になる場合よりも小さくなることを考え、その距離を  $\frac{r}{2}$  とする。この場合、楽観的に考える場合と比べて滞留エリアの端にデータが届くまでに、2 倍のホップ数が必要となる。

以上より、 $T_{c100}$  秒以内に滞留エリアのカバー率 100% を達成するという要求を満たすには、以下の初期送信間隔となる (式 (7))。

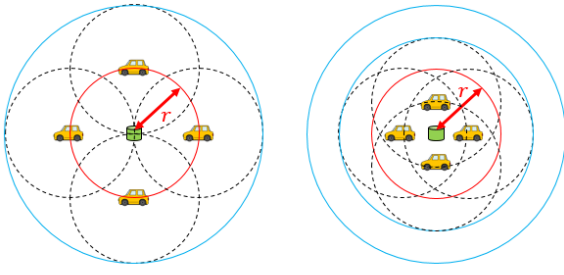
$$\begin{cases} d_{f\_o} = \frac{n}{\beta} T_{c100} \times \frac{r}{R}. & (x = r) \\ d_{f\_e} = \frac{1}{\sqrt{2}} \frac{n}{\beta} T_{c100} \times \frac{r}{R}. & (x = \frac{1}{\sqrt{2}}r) \\ d_{f\_p} = \frac{1}{2} \frac{n}{\beta} T_{c100} \times \frac{r}{R}. & (x = \frac{1}{2}r) \end{cases} \quad (7)$$

なお、 $d_{f\_o}$ 、 $d_{f\_e}$ 、 $d_{f\_p}$  はそれぞれ、O\_case、E\_case、P\_case の場合の初期送信間隔を表している。

### 4.3 滞留時間終了後のデータの扱い

滞留期間が設けられたことにより、データの削除動作が必要である。滞留時間を過ぎてもデータが配信されると、無駄なデータ送信数の増加やデータ受信者が既に価値のなくなった情報を利用してしまふ可能性がある。そこで、ノードはデータを受信すると、データ内に含まれているデータ送信開始時刻  $T_s$  及び滞留時間  $T_R$  を確認する。そして現在の時刻  $T_n$  を見て、 $T_n - T_s > T_R$  であるとデータを破棄する。また、データ送信時にも同じ動作を行い、滞留時間を過ぎていれば送信を中止しデータを破棄する。データ受信時と送信時にこの動作を行うことで無駄なデータの配信を防ぐことができる。





(a) 楽観的に考えたノード配置 (b) 悲観的に考えたノード配置

図 4: ノード配置の考慮

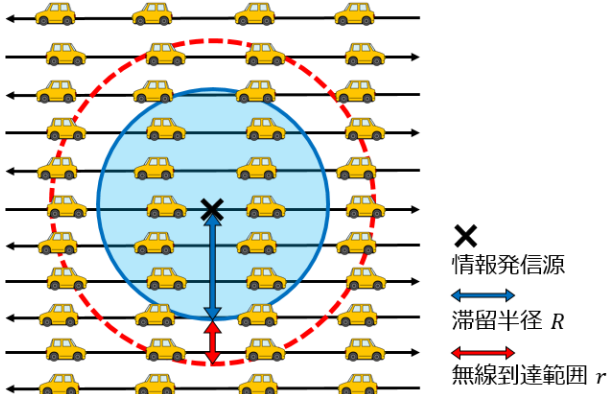


図 5: シミュレーションモデル

## 5. シミュレーション評価

本提案手法の有効性を示すため、シミュレーションにより評価する。5.1 節にシミュレーション環境、5.2 節にその結果と考察を述べる。また比較手法として、滞留時間と  $T_{c100}$  を考慮していない pre\_method を用いる。

### 5.1 シミュレーション環境

ネットワークシミュレータ OMNeT++, 交通流シミュレータ SUMO, IEEE802.11p 実装の Veins を用いて提案手法の評価を行う。図 5 にシミュレーションモデルを示す。ノードのスピードは時速 40 km/h で、レーン間隔は 200 m で固定、ノードの出現間隔は一定とする。また、情報発信源及びノードの無線到達範囲は  $r = 300$  m とする。ビーコン送信間隔は 1 s, 平均移動係数  $\alpha = 0.5$ , 一定期間内の送信目標回数  $\beta = 4$  と設定した。本研究では、情報発信源を同一地点に配置し、滞留半径  $R = 600$  m, 滞留時間  $T_R = 4$  s,  $T_{c100} = 1$  s のデータを 10 個滞留させる。このとき、10 個のデータの送信が同じ時間内で重複しないようにするため、一つのデータ送信（滞留）完了後から 0.5 s 後に次のデータを送信するものとする。またデータ 1 つのサイズは 300 Bytes とする。このとき、隣接ノード台数は  $n \geq 8$  とし、それぞれの台数別の初期送信間隔  $d_{f\_o}$ ,  $d_{f\_e}$ ,  $d_{f\_p}$  (表 1) に加え、pre\_method の送信間隔  $d = 5$  とする。評価指標として、 $T_{c100}$  秒毎のカバー率 (式 (1)), 0 から  $T_{c100}$  秒までの平均送信回数を算出する。

### 5.2 シミュレーション結果と考察

図 6 にノード台数  $n$  毎のカバー率の変化を示す。それぞれ、横軸がデータ送信開始から滞留終了までの時間、縦軸が各初

表 1: 台数別初期送信間隔  $d_f$  [s]

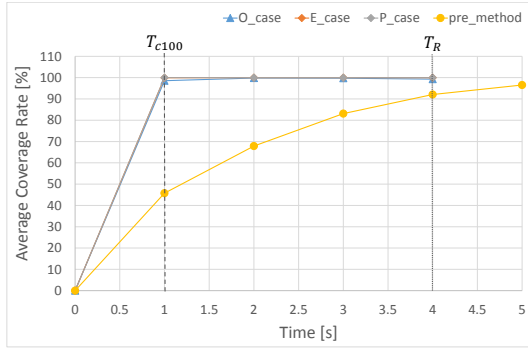
	O_case	E_case	P_case	pre_method
$n=8$	1.000	0.7071	0.5000	5.000
$n=16$	2.000	1.4142	1.000	5.000

期送信間隔ごとのカバー率を示している。これらの結果から、pre\_method は隣接ノード台数が増加するとカバー率が上昇する傾向にあるが、 $T_{c100}$  秒時点でカバー率は、 $n = 8$  のときは 46.87 %,  $n = 16$  のときは 61.73 % と、 $T_{c100}$  の要求を満たせなかった。これは送信間隔を  $d = 5$  と設定しているため、 $d > T_{c100}$  の関係となり、初期送信間隔は  $0 < d_f < T_{c100}$  となるように設定することが必要であることを示している。また、pre\_method は 5 秒時点では 100% に近い値を示しているが、今回滞留させたいデータの滞留時間は  $T_R = 4$  秒であるため、 $T_R$  以降はデータ送信者の要求を超えたデータ滞留であるといえる。

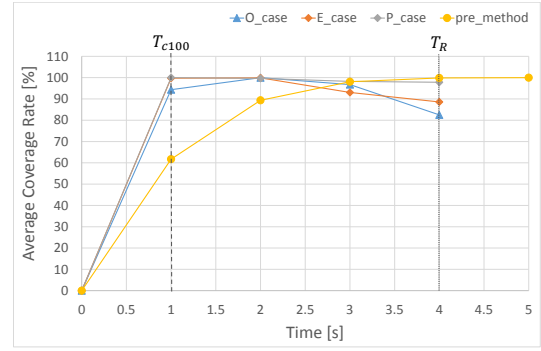
一方で、今回提案した  $d_{f\_o}$ ,  $d_{f\_e}$ ,  $d_{f\_p}$  は、 $T_{c100}$  秒時点でのカバー率が高い値を示している。隣接ノード台数が増加すると滞留エリアの中心から各ノードまでの距離の期待値は  $\frac{1}{\sqrt{2}}r$  に近づくため、ノードの配置は E\_case に近づくことになる。そのため、O\_case ( $d_{f\_o}$ ) では  $n = 8$  のとき  $T_{c100}$  [s] 時点でのカバー率は 99.78% となっているが、隣接ノード台数が増え  $n = 16$  となると、滞留エリアを満たすには  $d_{f\_o}$  では長く、94.34% と低下した。次に  $d_{f\_e}$  は最大 100%, 最小 99.75% となり、 $d_{f\_p}$  は最大 100%, 最小 99.99% となり、隣接ノード台数に関わらず 99% 以上を実現することができた。傾向としては、 $d_{f\_o} > d_{f\_e} > d_{f\_p}$  であるので、初期送信間隔が小さくなるにつれ、カバー率が上昇することが分かる。

次に図 7 は、 $T_{c100}$  秒以内の平均データ送信数を示している。pre\_method は最も送信間隔が大きく、データ送信の機会が最も少ない。よってデータ送信数が少なすぎるため、 $T_{c100}$  秒時点でのカバー率は最大でも 61.73% にしか満たなかった。次に提案手法では、 $d_{f\_o} > d_{f\_e} > d_{f\_p}$  であるため、平均データ送信数に差が出ている。データ送信数が多いとデータ衝突の可能性が大きくなるため、図 7 より O\_case が最も送信数が少ないが、カバー率の観点で見ると、 $T_{c100}$  秒時点で常に 99% 以上は満たせず、特に隣接ノード台数が多い場合は、O\_case では初期送信間隔が長く、要求を満たせなくなると考えられる。E\_case 及び P\_case は、 $T_{c100}$  秒時点で常にカバー率 99% 以上を満たせるが、データ送信数で比べたとき、E\_case のほうが 35.41% 少ないため、提案手法のなかでは E\_case がデータ送信数を抑えつつ要求を満たせる手法だといえる。

図 8 にシミュレーション時間に対するノード送信の有無を 2 値で示す。情報発信源がシミュレーション時間  $T = 1$  でデータを送信し、それから 4 秒間データ滞留が行われている。シミュレーション時間  $T = 5$  で送信が終了し、0.5 秒後に次のデータが送信されているのが分かる。よって、滞留時間  $T_R = 4$  と設定したとき、4 秒以降はデータ滞留を終了することを実現した。



(a)  $n = 8$



(b)  $n = 16$

図 6: カバー率

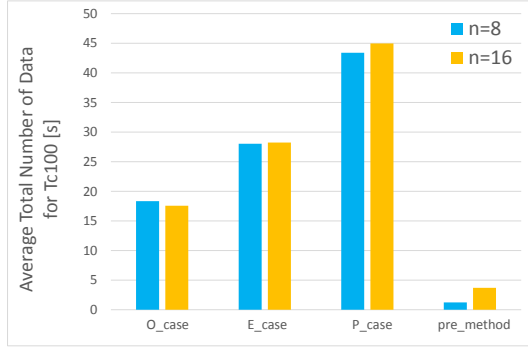


図 7: 手法別平均データ送信数

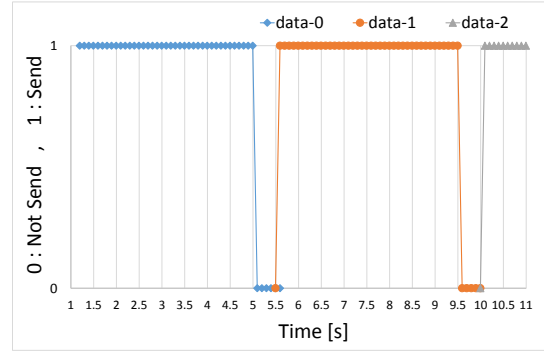


図 8: データ送信の時間変化

## 6. ま と め

本研究では、「時空間データの地産地消」を掲げて時空間データ滞留システムの構築を行ってきた先行研究が、データの滞留時間を考慮していないことに着目した。これによる問題点を解決するため、初期送信間隔の設定や滞留時間後のデータ削除を取り入れ、滞留エリアの要求を満たすことを実現した。しかし、実環境ではノードの配置が様々ではないと考えられ、情報発信源が設定する初期送信間隔では滞留エリアの要求を満たせない可能性がある。今後は、各ノードが隣接ノード台数に応じて初期送信間隔を変更し、場所によってノード密度が変化したときにカバー率を保てるように改善を行う予定である。

## 謝 辞

本研究の一部は、JSPS 科研費 JP18H03234、及び国立研究開発法人情報通信研究機構の委託研究による成果を含む。ここに記して謝意を表す。

## 文 献

- [1] A Survey on Mobile Edge Networks: Convergence of Computing, Caching and Communications,
- [2] 手柴弘樹 他, “時空間データ滞留のための車両ノード密度に応じたデータ送信制御手法の検討”, NS 研究会, 2016 年 6 月.
- [3] T.Higuchi, R.Onishi, O.Altintas, D.Nobayashi, T.Ikenaga and K.Tsukamoto, "Regional InfoHubs by vehicles:balancing spatio-temporal coverage and network load, " InProc.IoV-VoI'16, pp.25-30, 2016.
- [4] F.Li, and Y.Wang, "Routing in vehicular ad hoc networks: A survey, " IEEE Vehicular Technology Magazine, Vol.2, Issue 2, pp.12-22, 2007.
- [5] C.Maihofer, T.Leinumuller, and E.Schoch, "AbidingGeocast: Time-stable Geocast for Ad Hoc Networks, " InProc.

ACM VANET, pp.20-29, 2005.

- [6] A.Maio, R.Soua, M.Palattella, T.Engel and G.Rizzo, "A centralized approach for setting oating content param-eters in VANETs, " 14th IEEE Annual Consumer Commu-nications & CCNC 2017, pp.712-715, Jan, 2017.
- [7] G.Rizzo and H.Neukirchen, "Geo-Based Content Sharing-for Disaster Relief Applications, " International Conferenceon Innovative Mobile and Internet Services in UbiquitousComputing, Advance in Intelligent System and Computing, vol.612, pp.894-903, July, 2017.
- [8] I.Leontiadis, P.Costa, and C. Mascolo, "Persistentcontent-based information dissemination in hybrid vehicu-lar networks, " In Proc.IEEE PerCom, pp.1-10, 2009.
- [9] J.Ott, E.Hyyti, P.Lassila, T.Vaegs, and J.Kangasharju, "Floating Content: Information Sharing in Urban Areas, " In Proc.IEEE PerCom, pp.136-146, 2011.
- [10] N.Thompson, R.Crepaldi and R.Kravets, "Locus: ALocation-based Data Overlay for Disruption-tolerant Net-works, " In Proc.ACM CHANTS, pp.47-54, 2010.