

基幹車両を使ったマルチヘッドクラスタリングの性能評価

M2014SC011 熊澤 秀俊

指導教員: 河野 浩之

1 はじめに

VANET (Vehicular Ad Hoc Network) と呼ばれる車々間のネットワークを作る研究がなされている。移動体通信の中でも高い移動性、地域や時間帯によって大きく変化するノード密度などの VANET 特有の性質が周波数帯域の圧迫やトポロジの急激な変化を引き起こし、安定したネットワークの提供を困難にしている。

これらの問題に対して車両をクラスタリングし、対応を試みる研究 [1] がなされているが、移動性の高いノードによって形成される VANET において CH の離脱などに起因してクラスタが解散し、通信が途絶したりクラスタ再形成のために生まれるオーバーヘッドが問題となるので、可接続性の向上や効率的なクラスタのマネジメントが重要とされている。

そこで我々が提案する CH に対する安定性を評価する指数 RSM (Relative State and Mobility) により CM がクラスタ間でハンドオーバし、各 CM のクラスタ離脱率の低下、ひいてはクラスタ所属率の向上を図る。これにより各ノードにおける可接続性の向上し、さらにクラスタ再形成のためのオーバーヘッドが削減すると考えられる。

この検証にはトラフィックシミュレータ SUMO を用いて都市部におけるモビリティモデルを生成し、ネットワークシミュレータ ns-3 に提案アルゴリズムの追加モジュールを C++14 で実装し、性能を評価する。

本研究論文は全 5 章で構成される。2 章では車々間通信における諸問題を解決すべく提案されたクラスタリングおよびハンドオーバを解説する。3 章では我々が提案する基幹車両を使った MCIH (Multi-head Clustering with Inter-cluster Handover) を説明する。4 章では提案アルゴリズムにもとづいて都市部シナリオにおける車々間通信をシミュレートし、5 章で提案アルゴリズムが示した性質を評価する。

2 先行研究

2.1 クラスタリング

Lo らは一つのクラスタの代表となる MCH (Master Cluster Head) と、それを補助する複数の SCH (Sub Cluster Head) を選出するマルチヘッドクラスタリングを提案した [2]。3.2 節で後述する評価式 RPM (Relative Position and Mobility) を用いて MCH を選出する。この RPM はあるノードが近隣ノード群における位置の平均と速度の中央値に対して相対位置と相対速度を求め、郡中の移動度とする。この RPM が群中で小さいノードが MCH として推薦され、MCH はクラスタ内に SCH を複数配置してクラスタサイズを増大させクラスタの乱立を抑止した。また移動性の高い V2V (Vehicle to Vehicle) のためのクラスタ形成においてモビリティを評価関数に組み込んだことでクラスタの長寿命化を達成した。

一方で MCH の停車などに起因してクラスタが解散すると SCH や CM (Cluster Member) はどのクラスタにも属さない UN (Undecided Node) に戻り、クラスタを再形成するオーバーヘッドが発生する。また CM からみてクラスタ間の移動は提供されておらず、クラスタから離反すれば UN に戻り他のクラスタを探すか MCH となってクラスタを形成しなければならない。

2.2 ハンドオーバ

Sheu らは V2I (Vehicle to Infrastructure) における通信基地局と車両の間のハンドオーバを提案した [3]。図 1 のように車両の基地局に対する状態 $S \in \{11, 10, 01, 00\}$ を信号強度にもとづいて定める。あるノードが基地局から信号を受信したとき閾値以上の信号強度であった場合を $S = 11$ とする。信号強度が閾値に満たない場合は、ノードが基地局に接近しているか離反しているかに着目し、それぞれを $S = 10$, $S = 01$ とする。 $S = 00$ は信号を受信できないときを指す。

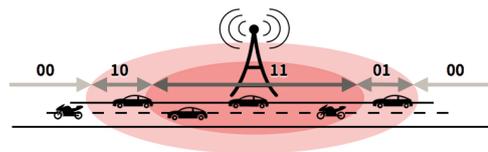


図 1 V2I における状態の定義とハンドオーバ

データの交換するさいは自身の状態をヘッダに加えフラグディングする。あるノード i が他のノード j からのデータを受信したさい、 i の信号強度 S_i が j から受信したデータに記された受信感度 S_j を上回るときノード i はそのデータをフラグディングする。このリレーによりデータが基地局まで伝達される。

2.3 課題

Lo らのマルチヘッドクラスタリングはモビリティを評価関数に組み込むことで安定したクラスタが形成された。また複数の MCH が一定時間接近するとクラスタを併合し領域内にクラスタを効率よく配置した。しかし CM はクラスタ間を移動できず、クラスタから離脱して一度未所属状態にならないと他のクラスタに移動できない。また Sheu らのハンドオーバは車両間の距離や速度を考慮していない。

3 Multi-head Cluster with Inter-cluster Handover

3.1 定義

Sheu らが提案した V2I における車両の通信基地局に対する状態を拡張し、クラスタベース V2V のための状態を

提案する. 任意のノード i のクラスタヘッド l に対する状態 $S \in \{11, 10, 01, 00\}$ を定義する.

- 11: 受信感度が閾値を超える
- 10: 受信感度が閾値未満かつ増大傾向
- 01: 受信感度が閾値未満かつ減少傾向
- 00: 受信不能

全てのノードは Role をもつ. Role は UN, CM, MCH, SCH の 4 種類である. UN はクラスタに属していないノードを指す. また MCH と SCH をまとめて CH (Cluster Head) と呼ぶ.

さらに相対位置 Rel_Dist , 相対速度 Rel_Speed を次のように定義する. この研究では平面上で車両が移動していると仮定 ($D = 2$) する. 位置 P および速度 v と近隣車両の基準値として位置 P_c , 速度 v_c を考える. 前者は相対位置 Rel_Dist の算術平均をとり, 後者は相対速度 Rel_Speed の中央値をとる. m 台の近隣車両と h 台の CH が近隣に存在するとき, それぞれの集合を M, H とする. 2 点間の受信感度は $state$ とする.

$$P = [x_1, x_2, \dots, x_D] \quad (1)$$

$$P_c = \frac{1}{m} \sum_{i=1}^m P_i \quad (2)$$

$$Rel_Dist_{AB} = P_A - P_B \quad (3)$$

$$v_c = Median\{v_i | i \in M\} \quad (4)$$

$$Rel_Speed_{AB} = |v_A - v_B| \quad (5)$$

$$M = \{1, 2, \dots, m\} \quad (6)$$

$$H = \{1, 2, \dots, h\} \quad (7)$$

$$state_{AB} = 11, 10, 01, 00 \quad (8)$$

ここでは受信感度は距離に比例して増減するものと仮定する.

3.2 MCIH の概要

HELLO メッセージ

各ノードは定期的に HELLO メッセージを広告する. HELLO メッセージには自身の IPv6 アドレス, 属する CH の IPv6 アドレス, 位置, 速度, RPM, Role, クラスタサイズで構成されている. RPM は式 (9) で得る.

$$RPM = \alpha \cdot \frac{Rel_Dist_{ic}}{Max\{Rel_Dist_{jc}, j \in M\}} + (1 - \alpha) \cdot \frac{Rel_Speed_{ic}}{Max\{Rel_Speed_{jc}, j \in M\}} \quad (9)$$

近隣に CH がいない場合は, RPM 値が最も小さいノードを MCH として推薦しクラスタを形成する.

基幹車両を使ったクラスタリング

我々の提案では特定の地域を巡回し, 走行時間が長く公共性の高いタクシーやバスなどの基幹車両を専任の MCH

とすることでクラスタの解散を低減する. これにより, 空間内でのクラスタが冗長に配置されるが基幹車両は都市内で過密とならないよう配置されることから, MCH の一局集中はないものと考えられる.

受信ノードが UN であるばあい, エントリテーブル内で評価式 (9) にもとづいて算出される RPM 値がもっとも小さいエントリに対して招待要求を送る. このとき $\alpha = [0, 1]$ は RPM における位置と速度の重みをあわらす. MCH は基幹車両が担当が, 地域や時間帯によっては基幹車両がないばあいが考えられる. このときは Lo らによって提案されたアルゴリズムを使って MCH を選出する.

クラスタ間ハンドオーバ

MCH からの離反にともなう, クラスタから離反し一時的に UN におちいり通信不可となる. この問題を解決するために我々の提案する評価式 (10) を用いてクラスタ間ハンドオーバを実現する. このとき $\beta = [0, 1]$ は RSM における状態と速度の重みをあわらす.

$$RSM = \beta \left(1 - \frac{state_{ki}}{Max\{state_{ki}, k \in H\}} \right) + (1 - \beta) \frac{Rel_Speed_{ki}}{Max\{Rel_Speed_{ki}, k \in H\}} \quad (10)$$

MCH は定期的に HELLO メッセージを送信しており, そのメッセージは周囲のクラスタに属するノードも受信できる. つまり各々のノードは自身の周囲にあるクラスタとの安定性を常に評価できる状況にある.

各ノードは周囲のクラスタに対する RSM 値を記録するテーブルをもつ. 自身が所属するクラスタとの RSM 値が閾値を超えていなければハンドオーバはなされない. しかし, 閾値を超えたばあいテーブル内でもっとも RSM 値が小さいクラスタに移る.

3.3 メッセージ

MCIH では全ノードが定期的に広告する HELLO, クラスタへの登録要求が RGSTREQ (Registration Request) メッセージ, 対する応答が RGSTREP (Registration Reply) メッセージ, クラスタヘッドへの推薦に ELECT メッセージ使う.

Hello メッセージは全ノードから All Nodes Multicast に宛てて発信され, 自身の IPv6 Global Unicast Address, 位置, 速度, RPM, RSM, Role, そして発信者が CH である場合は自身のクラスタに属するメンバの数が記録される. Hello を受け取ったノードはその情報にもとづいて 3.4 節で述べる近隣リストにエントリを登録, または更新する. また時間 T_{out} を超えたエントリはタイムアウトとして削除する.

RGSTREQ メッセージは UN または CM から All Routers Multicast に宛てて発信され, 登録先 CH の IPv6 Address と自身の IPv6 Address が記録される. 受信した CH は登録先アドレスと自身もつアドレスを比較し, 自身宛ての場合はクラスタメンバとして登録し RGSTREP メッセージを送る. 自身宛てでないが CM リストの中に

発信者のアドレスが登録されている場合は、当該エントリを削除する。

RGSTREP メッセージは CH から RGSTREQ に記録されていた発信者 IP アドレスに向けて発信される。UN が受信した場合は、自身のクラスタヘッドとして登録して Role を CM に変更する。CM が受信した場合は、自身のクラスタヘッドを変更する。

ELECT メッセージは UN の近隣に CH に存在しない場合に MCH を推薦するときに UN から UN へ発信される。発信の基準は HELLO メッセージで交換される値にもとづいて RPM が最も小さい UN が推薦される。

ELECT メッセージは SCH を MCH は推薦するさいにも利用し、CM に発信される。受信した CM は SCH への昇格する。

3.4 近隣ノードリスト

IPv6 における Link local multicast が届く範囲を近隣とし、Neighbor List に記録する。Neighbor List のエントリは、IP Address、既知であれば MAC Address、エントリの有効期限、位置、速度、RPM、State、Role で構成される。近隣ノードは Hello メッセージを受信したさいに追加または更新される。有効期限が切れたエントリが見つければその都度削除していく。

4 実験

4.1 実験環境と実装

実験に使用したツールと、実装したプログラムについて説明する。我々は ns-3 に C++ で新たに記述した MCIH モジュールを追加した。

OSM Open Street Map はフリーの地図情報データを作成、共有するプロジェクトである。当研究において OSM から取得した南山大学名古屋キャンパス周辺の地図情報をモビリティシミュレーションに用いた。

JOSM Java OSM Editor は Java で記述された OSM ファイル用のエディタおよびビューアである。

SUMO Simulation of Urban Mobility はオープンソースのモビリティシミュレータである。シミュレーション結果は xml ファイルで出力されるが、同梱されているツール群で各種ファイル形式へと変換できる。

ns-3 Network Simulator 3 は C++ で記述され、Python へのラッパをもつネットワークシミュレータである。

clang clang は LLVM (Low Level Virtual Machine) を使った C、C++、Objective-C、Objective-C++ 用のコンパイラである。MCIH モジュールは clang 3.6.1 でコンパイルした。

実装した ns-3 用の追加モジュールは 6 つにわけて説明する。追加モジュールおよびシミュレーションのシナリオファイルは C++14 で記述されている。

mcih モジュール全体を管理：1416 行

routing-table 経路情報を管理：192 行

packet 各種メッセージのヘッダを定義：844 行

neighbor 近隣リストを管理：592 行

utility モジュール全体で共通するマクロ定義など：137 行

icho 実験シナリオ：2498 行

4.2 地図データ

我々は Open Street Map から取得した南山大学名古屋キャンパス周辺の地図を使った。緯度および経度は 35.1493, 136.9624 である。使用した地図を図 2 に示す。



図 2 JOSM を使った地図の表示と編集

実験の概要

表 1 に示したパラメタにもとづいて、MCIH を RPM のみを用いた CPM (Center-Position and Mobility)[2] と比較する。基幹車両比率は全車両に対して基幹車両が含まれる割合である。

表 1 設定したパラメタ

シミュレーション時間	300s
MAC プロトコル	IEEE802.11p
車両数	30 台
基幹車両比率	0, 10, 20, 30, 40, 50%
ブロードキャスト間隔 (BI)	0.5s
タイムアウト	2 × BI

実験には SUMO (Simulation Urban Mobility) と ns-3 (Network Simulator 3) を使う。

SUMO は地図データとノードが移動する経路を生成し交通流をシミュレートするツールである。地図データは自動生成できるが、Open Street Map の地図も利用できる。我々は SUMO version 0.23 を使った。

ns-3 は様々な通信プロトコルに対応した分散事象駆動ネットワークシミュレータである。C++ を使ってスクラッチで開発され、現在は python でラップされている。我々は ns-3.23 に対して MCIH を C++ で実装した新たなモジュールを追加した。

評価基準

高い移動性をもつ VANET においてクラスタリングが有効 [1] であるが、クラスタを形成するにはノード同士が情報を交換しなければならず各ノードが本来望む通信以外に余計な通信が発生する。これは帯域を圧迫し通信を阻害する可能性を示唆している。MCIH はこの帯域の圧迫を緩和する。これを示すために MCIH が発するパケット数とクラスタの安定性の 2 点に着目して評価する。

パケット数

MCIH が発する HELLO, RGSTREQ, RGSTREP, RESIGN, ELECT のうち定期的に発信する HELLO を除いた 4 種のパケット数を比較する。

クラスタ安定性

基幹車両の増加にともなうクラスタの安定性を RPM で示す。0 に近いほど安定しており、1 に近いほど不安定である。

4.3 実験

OSM から地図を取得し SUMO 用の形式に netconvert を使って変換する。randomTrip, duarouter を使って地図に対応した車両と経路情報を生成し、さらに SUMO を使ってモビリティをシミュレートする。そのシミュレーション結果を ns-3 に適用するにあたり、SUMO は xml 形式で出力するが ns-3 へは tcl 形式で入力するので traceExporter を使って変換しなければならない。変換されたデータと追加モジュールを使い、ns-3 で実験した。

MCIH が送信するパケットのうち、定期的に広告する HELLO を除いた各種パケットの数を図 3 に示した。横軸は基幹車両の比率である。

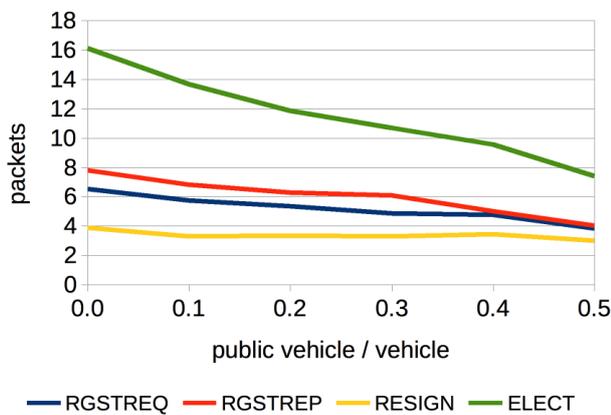


図 3 各種 MCIH パケット数と基幹車両比率

各 CM の所属クラスタの CH に対する RPM を BI ごとに記録し、その平均を図 4 に示した。横軸は基幹車両の比率である。

4.4 考察

図 3 によれば、基幹車両が増えるにつれて各種パケットが減少している。特に CH を推薦する ELECT パケットは最大で 16.1packets から 7.369packets に 54.2% 減少し、基幹車両がクラスタ形成にかかるパケットを減少さ

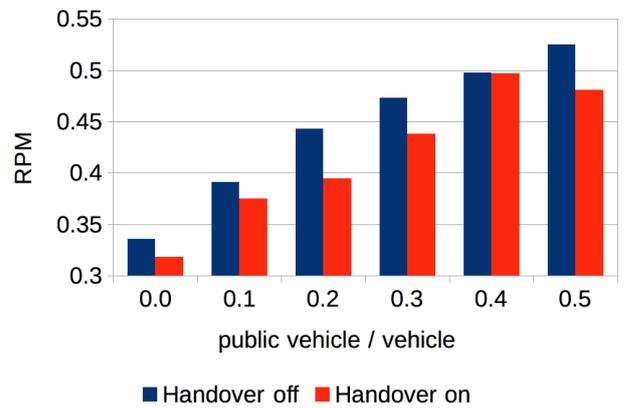


図 4 各 CM の所属クラスタの CH に対する平均 RPM

せたと考えられる。一方で図 4 からクラスタの不安定性を示す RPM 値を増加させている。これは基幹車両が相対位置や相対速度を考慮せず MCH に着任するため、必ずしも安定したクラスタが形成できないことを示している。しかしクラスタ間ハンドオーバーにより CM がより安定したクラスタへ移動するため、基幹車両の比率が 30% のとき、RPM を 0.443 から 0.396 に 10.6% 低減させクラスタの安定性を向上していると考えられる。

5 むすび

V2V のためのクラスタリングには CH の離脱にともなうクラスタの解散およびクラスタ再編成のためのオーバーヘッドという問題があった。これらの問題にたいして基幹車両を使ったマルチヘッドクラスタリングとクラスタ間ハンドオーバーの適用を提案した。OSM の地図と SUMO を使い車両のモビリティファイルを生成し、MCIH と CPM を比較した。結果として、基幹車両を使ったマルチヘッドクラスタリングによってクラスタ形成パケットおよびメンテナンスパケットが減少した。一方でクラスタの安定性が低下したが、クラスタ間ハンドオーバーを適用し安定性を維持した。今後、 α および β を変化させることで、基幹車両増加にともなう不安定化をより抑えられると考える。

参考文献

- [1] Bali, R. S., Kumar, N. and Rodrigues, J. J.: clustering in Vehicular Ad Hoc Networks: Taxonomy, Challenges and Solutions, *Vehicular Communications*, Vol. 1, No. 3, pp. 134–152 (2014).
- [2] Lo, S., Lin, Y. and Gao, J.: A Multi-Head Clustering Algorithm in Vehicular Ad Hoc Networks, *International Journal of Computer Theory & Engineering*, Vol. 5, No. 2, pp. 242–247 (2013).
- [3] Sheu, J., Lo, C. and Hu, W.: A Distributed Routing Protocol and Handover Schemes in Hybrid Vehicular Ad Hoc Networks, *Parallel and Distributed Systems (ICPADS), 2011 IEEE 17th International Conference on*, pp. 428–435 (2011).