V2X 協調型運転システムにおける通信性能のリアルタイム可視化ツール
Real-Time Visualization Tool for V2X Communication Performance in Cooperative ITS

伊藤 彰秀 †　大井 貴晴 †　塚田 学 †　落合 秀也 †　江崎 浩 †
Akihide Ito †　Oi Takaharu †　Manabu Tsukada †　Hideya Ochiai †　Hiroshi Esaki †

研究の要旨

高度交通システム (Intelligent Transportation System) の分野では，協調型運転の研究が盛んになってきている．これらの研究分野では，技術の導入にあたってシミュレーションによって評価をする段階から，フィールド実験での検証する段階へとシフトしてきている．そのため，ネットワーク性能を測定するためのツールが必要となっている．しかし，現状の V2X ネットワーク性能の測定には， ping や iperf と言った古典的ツールを利用することが多い．このようなツールでは，フィールドの地理的特性を反映しておらず，性能評価ツールとして十分であるとは言えない．そこで，本研究ではフィールドの地理的特性を反映しながら，リアルタイムで性能を評価できるツールを提案する．このツールは，先行研究で課題であったリアルタイムに関して機能が追加されている．さらに，実装したアプリケーションを評価するためのフィールド実験を行い，実際に地理的特性の反映された指標が表現されていることを確認した．

1　序論

1.1　研究の背景

高度交通システム (Intelligent Transportation System; ITS) の発達により，交通はより安全に，より快適に行えるようになった．国内では，路車間通信を用いた道路交通情報通信システム (Vehicle Information and Communication System; VICS) や，電子料金収受システム (Electronic Toll Collection System ; ETC) などが既に一般にも広がっている．また，車両に取り付けられたセンサにより衝突を回避する機能や，走行速度を一定に保つような機能をもつ高度な運転者支援システム (Advanced Driver-Assistance Systems; ADAS) を備えた車両も増えてきており，2014 年には世界中で 9 千万台だったのが，2 年後の 2016 年には 1 億 4 千万台もの自動車に ADAS の機能が装備され，更に市場は広がっている [1].

また，ITS において，より多様なアプリケーションを実現するために，道路で収集された情報を，車両と車両，あるいは車両と路側機などとリアルタイムで交換することで，更に高度で効率の良い交通システムを実現しようとする，協調型 ITS という仕組みづくりが進んでいる [2]．これは，各々の車両が自分の状態 (位置・速度など) を周囲に伝える車両間通信 (Vehicle to Vehicle; V2V) だけにとどまらず，通行人が所持する携帯端末と車両との通信 (Vehicle to Pedestrian; V2P)，道路に設置された路側機と車の間の路車間通信 (Vehicle to roadside Infrastructure; V2I) など，車と様々なものを接続する通信 (Vehicle to everything; V2X) を利用する技術方式である．この協調型 ITS の仕組みにより，従来得られていた情報よりも膨大なデータをリアルタイムで交換し合うことができるようになり，より発展した運転支援技術の開発が期待されている．

† 東京大学, The University of Tokyo

近年では，この V2X 技術がシミュレーションによる検証からフィールドにおける検証段階へと移行してきている．その際の通信性能の評価に用いられているのは ping や iperf などの古典的ツールである．しかし，これらの古典的ツールでは，

- フィールドの地理的特性を反映していない．
- 実際の V2X 通信で行われるプロトコルの特性を反映していない．

と言った課題が存在している．

本研究の目的は，V2X 通信の通信状況をリアルタイムに分析するツールを提供することである．また，V2X 通信で実際に用いられるプロトコル上で動作する分析ツールを用意することで，より現実に即した分析を可能にすることを目的とする．

2　協調型 ITS の現状と課題

2.1　自律型自動運転と協調型運転

協調型 ITS は，外部から提供される情報を通信などを使って取得しながら走行するという手法である．これは，例えば他の車両が得た情報のみに限らず，路側機 (Road Side Unit; RSU) などから得られる情報などを含んでいる．協調型 ITS はもともと，車と車 (V2V)，車と路側機 (V2I)，車と歩行者 (V2P) など，車とあらゆる対象 (V2X) との間で通信を行うことによって，車の情報や道路の状態などの交通情報の品質と信頼性を高めることで，交通の効率や安全性を向上させることを目的に発展してきたものであり，欧州の標準化団体である ETSI によって標準化が行われてきた．協調型 ITS を搭載した車両とインフラストラクチャは，相互の通信やインターネット上での通信を可能にすることで，従来では車載センサから得られる情報しかなかったのに対し，協調型ではより広範囲の道路交通に関する情報が得られるようになった．しかし，協調型 ITS を行うための課題がいくつか顕在してきている．例えば，異種混合な車両内外のシステムを接続するためのインタフェースの整備や，V2X 通信が満たすべき厳しい要件を満たすこと，チャネル干渉への対策，法規制や標準化への対応などがある [3]．さらに，歩行者や自転車などの車載器を持たない路上の対象 (Pedestrian) を，どのように協調型 ITS の枠組みに取り込んでいくのかという技術的課題も存在する．

現在主流の自動運転技術は，様々な車載センサを装備させて各自動車が独自に判断をする自律型自動運転技術であり，この技術を利用した自動運転は既に実証実験のレベルまで行われている．自律型自動運転において重要とされるのは，周囲の交通状況の把握であり，認識速度や認識精度と事故の危険性というのは密接に関係している．ゆえに，多様な対象物に対するセンシング技術や認識・検知技術の向上が不可欠であり，様々な研究が行われている．一方で，車載センサから取得できる情報や，車載の計算機での処理性能だけでは限界があり，外部か

ら提供される情報を通信によって取得しながら運転する，協調型 ITS の仕組みを自動運転の枠組みに導入しようとする試みが存在する．V2X を導入することにより，車載センサでは見通せない場所の情報を得ることが可能となり，さらなる安全・効率の向上が見込まれている．

V2X の研究は既にシミュレーションによる評価を行う研究が幾つも存在している．たとえば，Ning らの研究 [4] では車両間通信の遅延に対して評価を行っているが，これは待ち行列理論に基づいた数理的なモデルを仮定してシミュレーションを行うことによってエッジコンピューティングの必要性について強調している．また，Günther ら [5] は V2X 通信を行って隊列走行をすることで，渋滞を解消するにはどの程度の通信機能を有した車両が必要なのかを述べているが，これもシミュレーションの結果に基づいた主張となっている．次のフェーズでは，より多くのフィールドテストと実証実験を行うことが重要となっており，フィールドテストを支援するツールを作成することは，フィールドテストの効率を上げることにつながり，協調型自動運転の発展に貢献することが出来る．

2.1.1 V2X 通信のプロトコル

CAM 協調型 ITS において，道路の利用者である，車両・通行人や路面上に設置された設備との間で，自分の状態を伝えるためのメッセージとして Cooperative Awareness Message (CAM) [6] がある．送る情報の例としては，送信元の GPS やセンサなどによる位置情報・速度・進行方向などがある．このメッセージは ETSI によって標準化されている．CAM は車両が起動してから電源が落ちるまで，常に定期的に送信し続けることを想定している．CAM を定期的に送り合うことによって，安全性・快適性・効率性の向上を図っている．このメッセージは IEEE802.11p(5.9GHz 帯) を使用した近距離での通信を仮定しており，主にブロードキャスト・シングルホップで用いられている．日本国内では，5.9GHz 帯の無断使用は電波法で禁止されているため [7]，本研究では，IEEE 802.11b を用いている．

DENM 上で述べた，定常的に送られることを想定している CAM 以外に，道路の利用者に対して検知されたイベントを通知するメッセージとして Decentralized Environmental Notification Message (DENM) [8] がある．このときのイベントとは，例えば事故や故障などによる車線上への停止，あるいは悪天候や路面凍結などのことを指している [9]．このメッセージは CAM と比べて優先度が高く，より高品質な通信であることが望ましい．

LDM センサーと他の ITS ノードから送られるメッセージ情報を整形して保存する仕組みとして，Local Dynamic Map(LDM) [10] がある．LDM は CAM・DENM などを受け取り，それらをデータベースに格納し，定期的に情報を整理したり，破損している情報を破棄したりする機能を持っている．

2.2 AutoC2X

AutoC2X [11] とは，自律型自動運転ソフトウェアである Autoware が認識する領域を拡張するために，協調型自動運転ソフトウェアである OpenC2X を Autoware に統合したシステムである．

Autoware [12] は Autoware Foundation オープンソースの自動運転ソフトウェアであり，自己位置推定や，障害物検出，車体制御など，自律型自動運転を成立させるために必要な多くの機能が実装されている．Autoware は，2015 年に公開された，世界初の自動運転技術向けの “all-in-one” なソフトウェアであり，既に世界中で 100 社以上の会社で利用されている実績がある [13]．しかし，Autoware はセンサから得た情報を元に計算を行っているため，取得できる情報には限界がある．また，Autoware は自律型自動運転用のソフトウェアであるため，位置情報を共有出来ないという問題点も存在する．

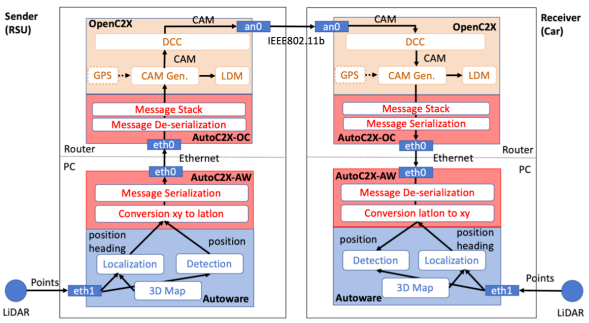

図 1: AutoC2X のモジュール構成 図の出典: [11]

OpenC2X [14] は協調型運転の実証実験を行うためのオープンソースソフトウェアである．OpenC2X は，V2X 通信に用いられるプロトコルとして ETSI で標準化が行われている，GeoNet や Security の部分は未実装であるものの，ETSI 標準である，CAM や DENM，LDM などの機能に関しては実装されている．OpenC2X は，それぞれの機能が独立しており，拡張性や再利用性が高いという利点がある．

Autoware の問題点を解決するために OpenC2X を用いて協調型運転の仕組みを導入したのが AutoC2X である．現在， Autoware に協調型運転の枠組みを追加したソフトウェアは， AutoC2X のみである．具体的なモジュールの構成は図 1 のようになっている．自律型自動運転と協調型自動運転における代表的なソフトウェアを統合した AutoC2X は，今後フィールドテストに利用されていく可能性が高いと判断し，本研究ではこのソフトウェアを用いて実験を行う．

2.3 関連技術・研究

第 2.3 章では，フィールドテストで実際に V2X の通信品質を評価する際に用いられているツールについて言及する．

2.3.1 古典的トラフィック分析ツール: ping, iperf

V2X 通信のフィールド実験を行った研究ではトラフィックの分析に ping や iperf などのツールを採用している．Napolitano ら [15] は，CAM を用いて路上の Vulnerable Road User(VRU，例えば歩行者や自転車など) の位置データの交換を行い，距離があるしきい値を下回ったら DENM を送信するという研究を行っている．この研究は，VRU の場所を固定し，車両が 10〜30km/h の速度で走行したとき，Wi-Fi と LTE 回線とを用いて，車両とエッジサーバ・クラウドサーバとの Round-Trip Time (RTT) の差がどれくらい生じるかについて評価している．また，Kutila ら [16] は，5G ネットワークが既



図 2: AnaVANET が動いているときの画面 図の出典: [17]

に導入されている中国とフィンランドで，LiDAR で取得した点群データを RSU に送信する研究を行っている．この研究では，点群データを送る際の通信の遅延やスループットなどを測定している．いずれの研究も，ping や iperf などのツールを用いてフィールドテストによって評価を行っている．

しかし，ping や iperf などのツールには，

- フィールドの地理的特性を反映していない．
- 実際の V2X 通信で行われるプロトコルの特性を反映していない．

と言った課題が存在している．V2X 通信は比較的高速に移動する通信者同士がフィールド上で通信を行うという性質上，車両の移動特性や，フィールドの遮蔽物による地理的特性などが存在しており，また，実際の V2X 通信で用いられるような BTP/GeoNetworking や CAM，DENM などのプロトコルでデータの転送を行っておらず，ping や iperf などのツールでは上記の問題点に対処できない．

2.3.2 AnaVANET

地理的特性を考慮した V2X 通信分析ツールとして AnaVANET (Analyzer for Vehicular Ad-hoc Networks) [17] がある．図 2 は AnaVANET が実際に動作しているときの図である．このシステムは，V2X 通信のプロトコルである BTP/GeoNetworking の解析を可能としており，AnaVANET は，フィールドテストを行った際，テストデータの後処理・関連する性能指数の生成・性能指数を適切に表示するための視覚化などを行うツールである．フィールドテストを行う際，全ての車載ルータで GPS ログと TCPDUMP ログを取得しておき，そのデータを入力として AnaVANET を実行することで，gnuplot によるグラフ化や， Google マップによる可視化を可能としている．

しかし，このツールはデータをすべて集めた後に，テストデータの処理を行うという性質上，古典的トラフィック分析ツールに存在していたリアルタイム性という利点を失っている．

2.4 現在の課題点

以上の点をまとめると，リアルタイム性を残しつつ，フィールドの地理的特性を反映しているようなツールを作成することが必要であることが分かる．

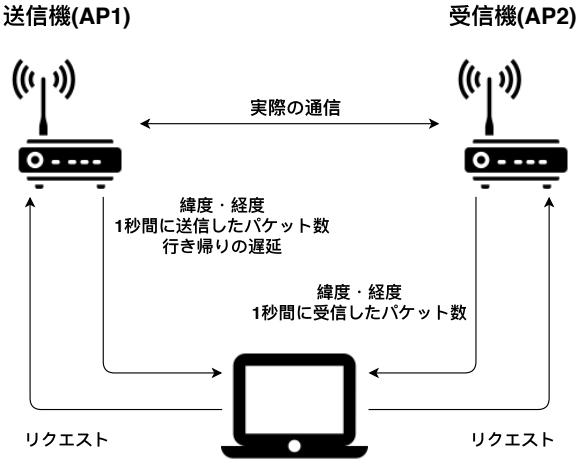

図 3: ネットワーク的な設計

3 設計と実装

本章では，まず本研究で提案する V2X 性能評価システムに要求される要件を整理する．次に，具体的な設計手法について述べ，さらにどのようにそれを実装したのかをまとめる．

3.1 要件

本研究の要件としては，

- 地理的特性を可視化出来る
- リアルタイムに通信状態が見える
- 実際の V2X 通信で用いるアプリケーション上で動く

といったものが挙げられる．

3.2 設計

図 3 にネットワークの設計を示す．

まず，送信機 (AP1), 受信機 (AP2), 性能可視化用のクライアント (PC) を用意する．

クライアントである PC が 1 秒ごとに AP1 と AP2 に対してリクエストを送信し，AP1 は緯度・経度，遅延，1 秒間に送信したパケット数を返す．

AP2 は緯度・経度，1 秒間に受信したパケット数を返す．

PC はそれらの情報から，AP1 と AP2 の距離，packet delivery rate (PDR) を計算し，地図とグラフにプロットを行う．

AutoC2X は本来は車両同士の協調認識を行うためのソフトウェアであるが，遅延とパケットロスの性能評価機能も備わっている．その実装を参考に，それぞれのパラメータの計算を行う．

往復遅延の算出方法 AutoC2X で実装されている遅延の算出方法と同様に，送信したパケットと受信したパケットの ID が一致した際のタイムスタンプの差を採用する．

PDR 算出方法 PDR とは，送信したパケットのうち，正しく受信されたパケットの割合の事を意味する．そのため，PDR を計算するためには，送信側と受信側の送信パケット数を知る必要がある．iperf を用いて PDR を算出する際は，1 秒あたりに送信するパケット数を固定

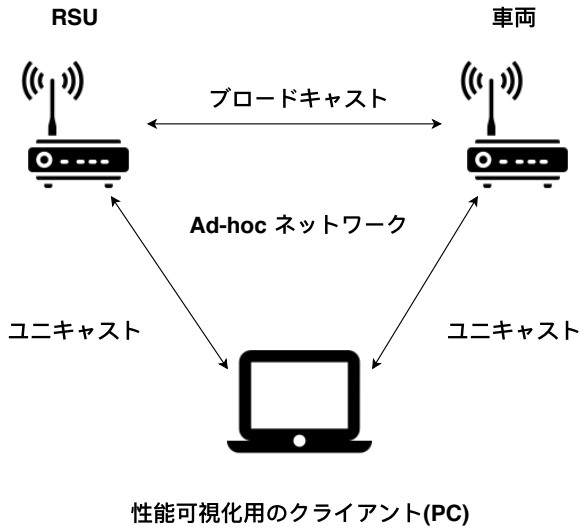

図 4: 実験のネットワーク構成

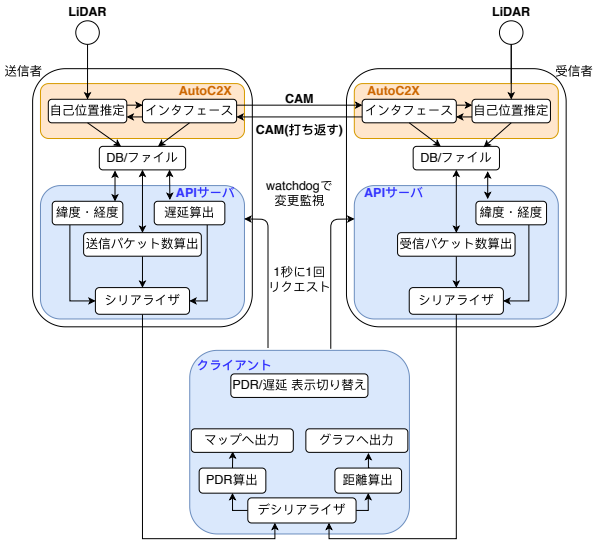

図 5: 作成したツールの実装概要 (青色の部分が実装を行った部分)

し，受信側でパケット数をカウントすることによって計算を行っている．一方で，今回使用する AutoC2X では，実際にアプリケーションを動作させながら計測を行うため，送信パケット数が動的に変化してしまう．このため，AP1 と AP2 が単位時間あたりに送受信したパケットを計測者に伝える必要がある．

PC は AP1・AP2 から受信した値から，PDR を算出する．

3.3 実装

3.3.1 機材・ソフトウェア構成

車載ルータとして，apu4c4 を利用した．これは，一般的には無線通信用のデバイスとして利用されており，本研究では V2X 通信の送受信機として採用した．Wi-Fi モジュールには Intel®Wi-fi 6 AX200 Module を使用した．可視化用の pc では，Ubuntu 18.04 を使用した．ブラウザは google chromeversion 79.0.3945.88 (official build) (64-bit) を使用した．

サーバに関しては Python の軽量な web フレームワークである Flask[1]を用いて実装を行った．

クライアントに関しては，地図上のプロットに Google Maps Platform の Maps JavaScript API[2]を利用した．グラフのリアルタイムプロットには，データ可視化ライブラリである d3.js[3]を利用した．

3.3.2 通信機器の構成

実験で用いる通信機器の関係は図 4 のようになっている．まず，車両と RSU にそれぞれ V2X 通信用のルータを用意し，同一サブネット内の IP アドレスを割り振る．次に，車両と RSU の通信モードを Ad-hoc モードにし，適切にルーティングがなされるようにする．RSU と車両との通信は通常の V2X 通信と同様にブロードキャストで通信を行う．

性能可視化用のクライアント PC は，Ad-hoc モードのネットワークに参加し，RSU・車両それぞれのルータに対してユニキャストで通信をする．

1 https://pypi.org/project/flask/
2 https://developers.google.com/maps/documentation
3 https://d3js.org/

3.3.3 具体的な実装

ここでは，作成したツールの具体的な実装方法について述べる．実装の概要を図 5 に示す．実装はサーバサイドとクライアントサイドに分かれている．

サーバサイド サーバは AP1 と AP2 上で動作する．緯度・経度は Autoware 側の自己位置推定から得られた座標を用いる．AutoC2X では，タイムスタンプ・緯度・経度の情報が csv ファイルとしてルータ上に書き込まれるため，作成される csv ファイルの変更を watchdog[4]で監視し，変更を検出したとき，書き込まれたデータをバッファに溜める．ID の値が一致したパケット同士のバッファ内でのタイムスタンプの差分を計算することによって，遅延を計算する．1 秒間経過するごとにバッファと緯度・経度の値，更にその間送受信されたパケットの数を更新する．このような設計にすることにより，リクエストがクライアントから到達したときに，常に最新の値を返すことが出来る．

クライアントサイド クライアントは 1 秒ごとに AP1，AP2 に対してリクエストを送信し，AP1 と AP2 の距離と，PDR の計算を行う．それぞれの距離を計算したのち，グラフへの出力と，GoogleMap への出力を行う．PDR と遅延の両者を可視化するために，ボタンで両者の表示を切り替えられるよう実装を行った．GoogleMap 上に作成する点については，赤が最大値，青が最小値となるように色を変化させた．具体的には，PDR と遅延について，最大値と最小値を決め，0 から 1 の間に正規化を行い，HSV 色空間上で色相 (hue) を 0° から 240° の間で動かすことで色を変化させている．WebSocket[5]を用いてサーバとソケットを張り，通信を行うという手法も存在したが，今回は AP1,AP2,PC の間の通信が頻繁に切断される可能性が存在していたため，毎秒リクエストを行うという手法を採用した．

4 実験と評価

第 3 章で述べたような実装を元に，フィールド実験を行った．本章では，その結果についてまとめ，考察を加える．

4 https://pypi.org/project/watchdog/
5 https://tools.ietf.org/html/rfc6455

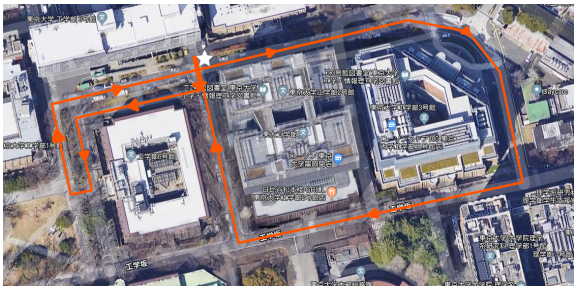

図 6: 実験の走行コース 白い星印の場所に RSU がある



図 7: 実験に使用した RSU



図 8: 実験に使用した車両

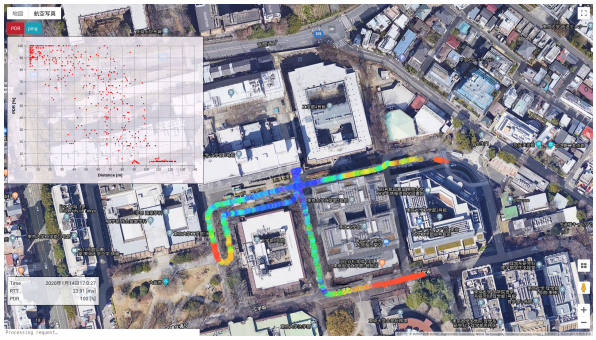

図 9: PDR の可視化

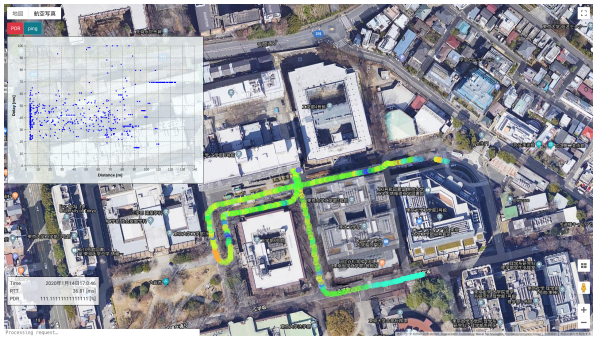

図 10: 遅延の可視化

4.1 実験

2020 年 1 月 14 日にフィールド実験を行った．走行したコースは図 6 のようになる．白い星印の位置に RSU を配置し，矢印に従って車が走行する．実験で使用した RSU 及び車両は，図 7，8 のとおりである．RSU が LiDAR で受け取った周囲からの情報を物体検出し，その情報を CAM にして車両に向けて送信する．車は受信した CAM を RSU に対して打ち返す．この時の通信の遅延と PDR について測定し，可視化を行った．

4.2 実験結果

実験結果は図 9，10 のようになった．左上に表示されているのがグラフであり，横軸は距離 [m] で共通して共通しており，縦軸はそれぞれ遅延 [ms] と PDR[%] を表している．

4.3 分析と評価

図 9 から，距離に応じて PDR が低下していることが分かる．これは，電波強度が距離の二乗に比例しており，電波強度が下がるに従ってパケットロスが増加するためであると推測される．

また，図 10 では，RSU 付近の直線では色の変化が小さいのに対し，見通しの悪い図の左側では，オレンジ色，つまり遅延がが大きくなっている箇所が存在する．このことから，遅延は物理的な障害物の影響を受けやすい可能性があると考えられる．

どちらの図からも，距離が同一であっても値が異なる箇所が存在している．これは遮蔽物の影響が大きいことが理由の 1 つとして考えられる．このように，ping や iperf などの古典的なネットワーク性能測定ツールを用いただけでは判明しなかったであろう，距離だけでは判明しないようなフィールドの特性を反映できていることが確認できる．

ここで，このツールの得失について表 1 にまとめる．作成したツールでは，AnaVANET にはなかったリアルタイム性が追加された．地理的特性や，V2X 通信で用いられるアプリケーションで動作するといった特性は残したままである一方で，送受信したパケットを記録していないという性質上，実験後にその結果を再現したり，あるいはデータを整形したりすることができない．

4.4 議論

図 6 と図 9，10 を見比べてみると，走行したにもかかわらず地図上にプロットがされていない部分が存在していることが分かる．これは，PC と RSU，あるいは PC と車両のどちらか一方または両方との通信が途切れたからであると考えられる．このツールは，RSU と車両の両方と通信が確立されていなければ，距離や PDR が計算できないため，地図やグラフ上にプロットすることができない．したがって，遮蔽物の多い環境では，可視化ツールを乗せた PC を適切な場所に移動させる必要があった．今回の実験でも，RSU と車両のちょうど両者から遮蔽されていない位置に移動しながら計測を行った．しかし，移動しながらであってもネットワークが繋がらなくなる部分が出来てしまい，その部分のプロットが消失してしまっていると推測される．

5 結論

5.1 まとめ

本研究では，現状の V2X 通信における評価ツールの問題点について整理し，新しい V2X 通信の通信性能を評価する支援ツールを提案した．フィールドの地理的特性の反映，リアルタイム性の担保，実際に V2X 通信で用いられるアプリケーション上での動作，などの特徴を有したツールとなっており，フィールド実験を支援することが可能となる．

表 1: それぞれのツールの得失

	ping, iperf	AnaVANET	作成したツール
地理的特性	×	○	○
リアルタイム性	○	×	○
実際のアプリケーション上で動作	×	○	○
実験後のデータ整形	△	○	×

フィールド実験の結果として，遮蔽物により遅延や PDR の値が悪化することがわかった．従来の通信性能の評価ツールであった，ping や iperf では明らかにすることができなかった，フィールドの特性による通信性能への影響も明らかにすることが可能であることを確認した．さらに，デバッグの支援につながるという副次的効果も明らかになった．

5.2 今後の課題

本研究で作成したツールの課題として，第一にネットワークの設計の問題が挙げられる．このツールは，クライアント側が 2 つのサーバに対して，常にネットワーク的に繋がっている必要がある．これは現状のネットワークのアーキテクチャでは，送信するパケットの数を動的に変更しているため，PDR を算出するために両者との接続が必要となっており，これを解決することは困難である．

第二に，データの保存に関する問題が挙げられる．現在のツールでは，サーバが受信したパケットの情報をファイルに書き込むイベントを監視し，最新の値を更新していくという実装を行っているため，実験の状況を再現してみることができない．したがって，実験結果を記録するためには，スクリーンキャプチャなどで動画として保存しておく必要がある．しかし，サーバには送受信したパケットデータがファイル残っているはずであり，それらを使用して同じ状況をクライアント側で再現することは可能なはずであるが，これは今後の課題とする．

参考文献

[1] Self-driving car technology: When will the robots hit the road?, 2017.

[2] COMMITTEE AND THE COMMITTEE OF THE REGIONS A European strategy on Cooperative Intelligent Transport Systems, a milestone towards cooperative, connected and automated mobility. Technical report, 2018.

[3] 東陽テクニカ SPIRENT. 自動車業界は V2X システムをどのように試験すべきなのか？ pp. 1–12, 2018.

[4] Zhaolong Ning, Jun Huang, and Xiaojie Wang. Vehicular fog computing: Enabling real-time traffic management for smart cities. *IEEE Wireless Communications*, Vol. 26, No. 1, pp. 87–93, 2019.

[5] H.Günther, S. Kleinau, O. Trauer, L. Wolf. Platooning at traffic lights. In *2016 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1047–1053, June 2016.

[6] *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service*, 2014.

[7] 電 波 法. https://elaws.e-gov.go.jp/search/elawsSearch/elaws_search/lsg0500/detail?lawId=325AC0000000131. (Accessed on 02/06/2020).

[8] General Principle and Vision White Paper Version 2.1.0・ . Technical report, 2018.

[9] 一般財団法人日本自動車研究所. ITS 協調システムの情報項目の標準化に関する 分析 ・ 検 証 報 告 書 目 次. 2014.

[10] Vehicular Communications. Intelligent Transport Systems (ITS); Vehicular Communications ;. Vol. 1, No. Ldm, pp. 1–42, 2014.

[11] Manabu Tsukada, Takaharu Oi, Akihide Ito, Mai Hirata, and Hiroshi Esaki. Autoc2x: Open-source software to realize v2x cooperative perception among autonomous vehicle. In *The 2020 IEEE 92nd Vehicular Technology Conference (VTC-Fall)*.

[12] S Kato, E Takeuchi, Y Ishiguro, Y Ninomiya, K Takeda, and T Hamada. An open approach to autonomous vehicles. *IEEE Micro*, Vol. 35, No. 6, pp. 60–68, November 2015.

[13] The autoware foundation - an open alliance for autonomous driving technology. https://www.prnewswire.com/news-releases/the-autoware-foundation---an-open-alliance-for-autonomous-driving-technology-300763934.html. (Accessed on 02/03/2020).

[14] florianklinger/openc2x-standalone.

[15] A Napolitano, F Civerchia, K Kondepu, L Valcarenghi, P Castoldi, and Scuola Superiore S Anna. Implementation of a MEC-based Vulnerable Road User Warning System. *2019 AEIT International Conference of Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE)*, pp. 1–6.

[16] M. Kutila, P. Pyykonen, Q. Huang, W. Deng, W. Lei, and E. Pollakis. C-v2x supported automated driving. In *2019 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–5, May 2019.

[17] Manabu Tsukada, Jose Santa, Satoshi Matsuura, Thierry Ernst, and Kazutoshi Fujikawa. AnaVANET: an experiment and visualization tool for vehicular networks. In *9th International Conference on Testbeds and Research Infrastructures for the Development of Networks & Communities (TRIDENTCOM 2014)*, Guangzhou, China, May 2014.