

車車間アドホックルーティングプロトコルGVGrid の実装と評価

濱田 淳 司[†] 孫 為 華[†]
山口 弘 純[†] 楠 本 真 二[†]

近年, 固定基地局を用いない車車間マルチホップ通信技術が注目を集めており, 我々の研究グループにおいても, 車車間マルチホップ通信における位置情報ルーティングプロトコル GVGrid を提案している. GVGrid は車両の移動情報を利用し, 生存時間の長い経路を生成する. 本研究では, GVGrid により構築される経路の品質評価を目的とし, AODV の実装である AODV-UU を用いて GVGrid のプロトタイプ実装を行った. ノード PC からなる小規模テストベッドを構築した上で, 実マップにおける車両の移動を想定して作成したリンク切断シナリオをテストベッド上で再現し, GVGrid 実装および AODV-UU により構築された経路上に ICMP パケットを連続送出することで, スループットの比較実験を行った. その結果, GVGrid はより高いスループットを達成でき, かつプロトコルの制御パケット数が少ないことを確認した.

Implementation of VANET Routing Protocol GVGrid and Its Evaluation

JUNJI HAMADA,[†] WEIHUA SUN,[†] HIROZUMI YAMAGUCHI[†]
and SHINJI KUSUMOTO[†]

In recent days, vehicular ad hoc network communication technology which does not require fixed infrastructure has had much attention. Several routing protocols for vehicular ad hoc networks (VANETs) have been designed so far, and we have also proposed GVGrid, a position-based routing protocol for vehicular ad hoc networks. In this paper, we have implemented the function of GVGrid. Using a scenario which consisted of vehicles' movement on a real map, we have evaluated throughput of ICMP packets which were successively transmitted by the source. The results have shown that GVGrid could achieve more stable communication and less control messages than AODV.

1. ま え が き

我々の研究グループでは, 車車間ルーティングプロトコル GVGrid を提案している. GVGrid は車両の位置, 進行方向, 道路地図を用い, 2 地点間の通信経路をオンデマンドに構築し, 維持する位置情報ルーティングプロトコルである. 領域をグリッド状に分割し, なるべく主要道路に存在し, 同一道路上で同じ進行方向である車両を優先的に中継車両として選択した経路を発見することで, 生存時間の長い経路を実現する. また, 一度構築した経路に対応する道路を記憶しておき, 経路が切断した場合は, 残存する経路を利用しながら, もとの道路上に経路を高速に復元する. 詳細に

については文献^{1),2)}を参照されたい. しかし, GVGrid により構築された経路が, 通常構築される最短経路などと比較しデータの遅延や損失がどの程度軽減されるかについてはまだわかっていない.

本研究では, GVGrid の機能の実装を行い, テストベッド上で AODV により構築された経路とのスループット比較実験を行ったため, それについて述べる. 具体的には, 無線 LAN を備えたノート PC10 台程度の小規模テストベッドを構築し, 実マップにおける車両の移動を想定して作成したリンク切断シナリオを実現するようにした. そのもとで, GVGrid 実装及び AODV 実装により構築された経路上にそれぞれ ICMP パケットを送出し, そのスループットの比較評価を行った. その結果, GVGrid はより高いスループットを達成でき, かつプロトコルの制御パケット数が少ないことを確認した.

[†] 大阪大学大学院情報科学研究科
Graduate School of Information Science and Technology,
Osaka University

2. GVGrid の実装

一般に、テストベッドでの性能評価試験では、テストベッドを構成するノード (PC) の台数により、試験内容が大きく制限される。本研究では主に、GVGrid および AODV により構築された経路が、ノード移動に伴いどのように維持されるか、またその維持過程が通信性能にどのような影響を与えるかについての試験を行う。したがって、GVGrid については、その経路構築プロセスや維持プロセスを直接実行するかわりに、シミュレーション等で得られた情報をもとに、指定されたノード上に強制的に経路を構築・維持するようにした。以下ではこれらについて述べる。

GVGrid の経路構築および維持プロセスの実装は、AODV-UU³⁾ の実装コードをもとに行った。AODV-UU は、スウェーデンの Uppsala University で開発されている AODV プロトコルの実装である。

2.1 経路探索プロセスの実装

GVGrid の経路構築プロセスは、AODV-UU の経路探索プロセスを改造し、外部より指定されたノードを RREQ メッセージの受信先として選択するようにしている。これにより、テストベッドにおいて、予定するノードに経路を構築できるようにしている。

2.2 経路維持プロセスの実装

GVGrid の経路維持プロセスは、(i) 経路の切断を検知したノードが行う、不要なリンクの強制切断処理、(ii) 経路の切断を検知したノードが経路の修復を開始する時に行う、経路維持メッセージの作成および送信処理、(iii) 経路維持メッセージの受信時の処理、および (iv) 経路の修復に関係するノードが行う、ルーティングテーブル更新の処理、からなる。以下では、AODV-UU の実装コードをもとにした、GVGrid の上記の四つの処理からなる経路維持プロセスの実装法について述べる。

不要なリンクの強制切断処理

GVGrid では、経路の切断を検知したノードは、自身が存在するグリッドが道路経路グリッド列に含まれるか否かを判定し、存在しない場合は、自身と接続しているノードとのリンクを強制切断し、通信を終了する。本実装では、経路の切断が発生した場合、経路の切断を検知したノードが強制切断を行うかどうかを外部のシナリオファイルに基づき決定し、実行する。

経路維持メッセージ作成および送信処理

AODV-UU では、経路の切断を検知したノードは、RREQ メッセージを作成してブロードキャスト送信する。これに対し、GVGrid では、自身が存在するグ

リッドが道路経路グリッドに含まれる場合は、RRPR (Route RePair, 以下 RRPR) メッセージの作成を行う。なお、GVGrid の RRPR メッセージと AODV-UU の RREQ メッセージのフォーマットは同一である。GVGrid では、RRPR メッセージの作成後、自身のグリッドの前方グリッドから代替ノードを選択し、自身が記録しているグリッド列を含む RRPR メッセージを代替ノードに送信するが、本実装では RRPR メッセージをシナリオによって指定されたノードへ送信するようにしている。

経路維持メッセージ受信時の処理

AODV-UU では、RREQ メッセージを受信したノードは、保持しているルーティングテーブルを検査し、目的地への経路エントリを保持していない場合は、RREQ メッセージをブロードキャストで転送する。また自身のルーティングテーブル内に目的地への経路エントリを保持している場合は、このノードは切断した通信経路の残存ノードであり、この場合は以下のいずれかの処理を行う。

- (1) 目的地への経路が有効であり、経路エントリの目的地シーケンス番号が受信した RREQ メッセージ内のシーケンス番号より大きい場合、シーケンス番号が大きいほど最新の経路であると考えられることから、このノードから目的地までの経路においては、経路エントリ内の経路が RREQ メッセージ内の経路よりも新しい経路であることがわかる。したがって、RREP メッセージを作成し、経路切断を検知したノードへ送信する。
- (2) 目的地への経路が有効であり、経路エントリの目的地シーケンス番号が受信した RREQ メッセージ内のシーケンス番号より小さい場合は、シーケンス番号が大きいほど新しい経路であると考えられるため、このノードから目的地までの経路については、RREQ メッセージ内の経路が経路エントリ内の経路より新しい経路であることがわかる。したがって、最新の経路を作成するために RREQ メッセージを作成し、ブロードキャストで転送する。
- (3) 目的地への経路が無効であるとき、目的地への新しい経路を作成するために、RREQ メッセージをブロードキャスト送信する。

これに対し、GVGrid は RRPR メッセージ転送時には、シーケンス番号における処理は行わず、目的地方向へのノードを選択してユニキャストで送信する手法をとっているため、AODV-UU を次のように修正

することで、それを実現した。まず、RRPR メッセージを受信したノードは、自身のルーティングテーブルを検査し、目的地への経路エントリを保持していない場合は、RRPR メッセージをシナリオによって指定されたノードへ転送する。また自身のルーティングテーブル内に目的地への経路を保持している場合は、切断した通信経路の残存ノードであるため、以下のいずれかの処理を行う。

- (1) 目的地への経路が有効である場合は、RREP メッセージを作成し、経路切断を検知したノードへ送信する。
- (2) 目的地への経路が無効である場合は、目的地への新たな経路を作成するために、RRPR メッセージをシナリオによって指定されたノードへ転送する。

ルーティングテーブル更新処理

AODV-UU では、経路維持プロセスにおけるルーティングテーブルの更新は、RREQ メッセージ受信時と RREP メッセージ受信時において行われる。RREQ メッセージを受信したノードは、自身のルーティングテーブルを検査し、目的地への経路エントリを保持していない場合は、送信元の情報とその RREQ メッセージを送信した隣接ノードのアドレスを送信元への次ホップとしてルーティングテーブルに記録し、送信元方向の経路を確立する。RREP メッセージを受信したノードは、ルーティングテーブルに目的地への経路エントリが存在するか検査し、目的地への経路エントリが存在しない場合は、目的地の情報と RREP メッセージを送信した隣接ノードのアドレスを目的地方向への次ホップとしてルーティングテーブルに記録し、目的地方向の経路を確立する。また目的地への経路エントリが存在し、かつ経路が無効状態である場合は、経路を有効状態にするなどして経路エントリを更新する。したがって、AODV-UU では双方向の経路の実現に独立した二つの経路エントリを使用している。

これに対し、GVGrid では、RREP メッセージ受信時に、RREP メッセージ内に記録されているノード ID 列から、一つの経路エントリへ自身の前方および後方のノードの情報を記録することにより、一つの経路エントリで双方向の経路を実現しているが、本実装では、AODV のルーティングテーブルと同様に一つの経路エントリで片方向の経路を実現している。GVGrid の経路維持プロセスにおけるルーティングテーブルの更新は、RRPR メッセージ受信時と RREP メッセージ受信時において行われる。RRPR メッセージを受信した場合と RREP メッセージを受信した場合のルー

ティングテーブルの更新処理は、AODV-UU における RREQ メッセージ受信時と RREP メッセージ受信時のルーティングテーブルの更新処理と同じである。

3. テストベッドによる経路品質の評価実験

GVGrid と AODV による経路品質を評価するためのテストベッドを構築した。テストベッドは、ノードの移動性とマルチホップ通信を実験室レベルで実現するために、iptables が持つ MAC 層でのフィルタ機能を使用し、任意のトポロジを擬似的に生成できるようにした。これについては 3.2 節で述べる。このテストベッドを用い、実際の車両の移動状況をもとに、作成した経路構築・維持シナリオにおける GVGrid と AODV のスループットを測定した。

3.1 既存のテストベッドとの位置づけ

MANET のテストベッドとして、MobiEmu⁴⁾、APE⁵⁾、ORBIT⁶⁾ などが知られているが、いずれも汎用のハードウェアを用いた実現が困難である、あるいは汎用の実行環境を提供していない、などの理由から、本研究では汎用の Linux PC のみを用いて容易に実現可能なテストベッドを独自に構築した。

評価用のテストベッドは、IEEE802.11b を備えたノート PC 10 台からなる。各ノート PC に搭載している OS は、Vine Linux 3.2 である。

3.2 MAC 層におけるパケットフィルタ機能

トポロジの実現には、iptables⁷⁾ が備える MAC 層でのフィルタ機能を利用している。これは、指定した MAC アドレスから送信されたフレームを全て破棄する機能である。この機能を利用することにより、実験環境において、任意の 2 ノードが直接通信可能な完全グラフトポロジの状態から任意のトポロジを作成することが可能である。

図 1 を用いて、MAC 層でのフィルタ機能を利用したトポロジ作成の様子を説明する。まず、図 1(a) は、テストベッドを構成する全ノードが、互いの無線範囲に存在するよう、それらのノードを配置する。この状態では、あるノードがブロードキャストを行うと無線範囲内に存在する全ノードにパケットが届く。このもとで、例えば、図 1(c) のようなトポロジを作成したい場合、ノード A からのパケットをノード B 以外が破棄するようにすればよい。よってノード B 以外の各ノードに、ノード A からのパケットを破棄するルールを iptables を用いて実行させる。図 1(b) は、各ノードが図 1(c) のトポロジ作成に必要な経路をフィルタした様子である。これにより、任意のトポロジを擬似的に実現できる。なお、これはすべてのノードが相

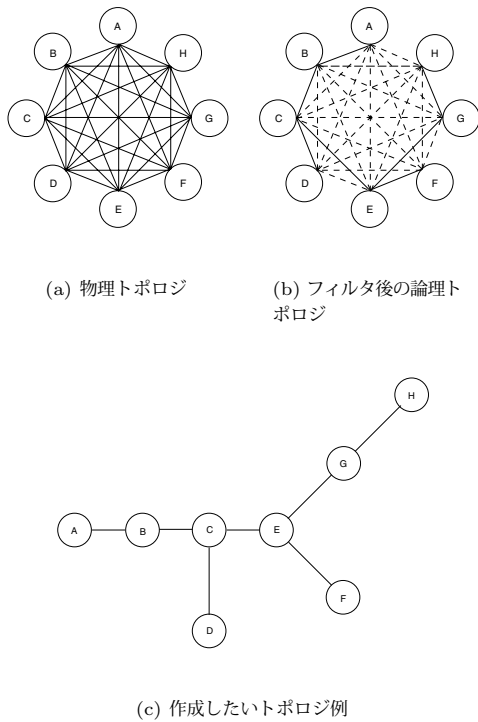


図 1 MAC 層のフィルタ機能を利用した論理トポロジ生成例

互に無線範囲にあるため、実際より多いフレームコンテンションが発生する可能性が高いが、本研究で行う実験ではそれほど大量のフレームを発生しない、クロストラフィックを考慮していない、といった理由のため、その影響は小さいと考えられる。

また本実験では、リンクの切断は、同様に iptables の MAC 層のフィルタ機能を用いてリンク先のノードからのパケットを破棄することで実現している。

3.3 車両の移動シナリオと経路切断状況

本節では、車両の移動シナリオと GVGrid と AODV における経路切断・復元状況について述べる。

車車間通信ではノードの移動が激しいため、ビルなどの障害物を考慮した場合、構築された通信経路が交差する 2 道路上の車両間リンクを含む場合、そのリンクの生存時間は短いことから切断回数が多くなる。また、同一道路を走行する車両間のリンクでも、それらのノードの進行方向が逆である場合はノード間距離の増大によりリンクが切断する確率が高くなる。その一方で、進行方向が同方向であり、かつ速度が同程度である場合はノード間距離はほぼ一定に保たれ、リンク切断が発生する確率は低くなる。

GVGrid では、経路の切断が発生した場合、修復を

開始するノードは、経路が通過する道路として記憶していた道路における隣接グリッドに存在するノードを 1 つ選択する。この際、可能であれば切断された経路の断片を保持するノードを発見し接続することで、残存した経路を積極的に利用し、経路復元を高速に行う。そのため、直線の道路上におけるノード間距離の増大による切断や交差する 2 道路上の車両間リンク切断では、初期の道路経路グリッド上に通信経路を再構築するため、修復前後の経路が大きく変わる可能性は低い。これらの状況をすべて試験できるように、GVGrid のシナリオを作成した。GVGrid の経路の変遷の様子を図 2 に示す。一方で AODV では、切断リンクの上流ノードが RREQ メッセージをブロードキャストしてそのノードから宛先ノードまでの新たな経路を発見する。したがって、修復後は比較的 shortest 経路に近い経路に戻される。AODV の経路の変遷の様子を図 3 に示す。

これらの車両の移動シナリオと経路切断状況は、3 分間のシナリオである。図 2(a) は GVGrid の初期経路、図 3(a) は AODV の初期経路である。

図 2(b) はノード C, D 間のリンクがノード間距離の増大により、図 3(b) はノード D, E 間のリンクが障害物により切断した状態を表している。GVGrid の場合、リンク切断を検知したノード C が隣接グリッドに存在するノード F を選択して経路修復を開始し、残存経路と接続する (図 2(b))。AODV では、リンク切断を検知したノード C が経路修復を開始し、また一部残存する経路を用いた経路が最新の経路であると判断して接続する (図 3(b))。

図 2(c) と図 3(c) は、ノード B, C 間のリンクが交差点における障害物の影響により切断した状態を表している。このとき GVGrid では、ノード B は自身が初期経路を表す道路グリッド列から外れたことを認識するため、接続しているノード A, B 間のリンクを強制切断する。そしてノード A は前方グリッドから代替ノード G を選択して経路の修復を開始し、残存経路と接続する (図 2(c))。これに対し、AODV では、ノード B が経路の修復を開始するが、このときノード F, 目的地 d 間の残存経路は最新の経路ではないと判断し、最新の経路を再構築する (図 3(c))。

図 2(d) と図 3(d) は、送信元 s, ノード A 間のリンクが距離の増大により切断した状態を表している。この場合 GVGrid では、送信元 s は前方グリッドから代替ノード H を選択して経路の修復を開始し、残存経路と接続する (図 2(d))。AODV は、リンク切断を検知した送信元 s が経路修復を開始し、また残存経路

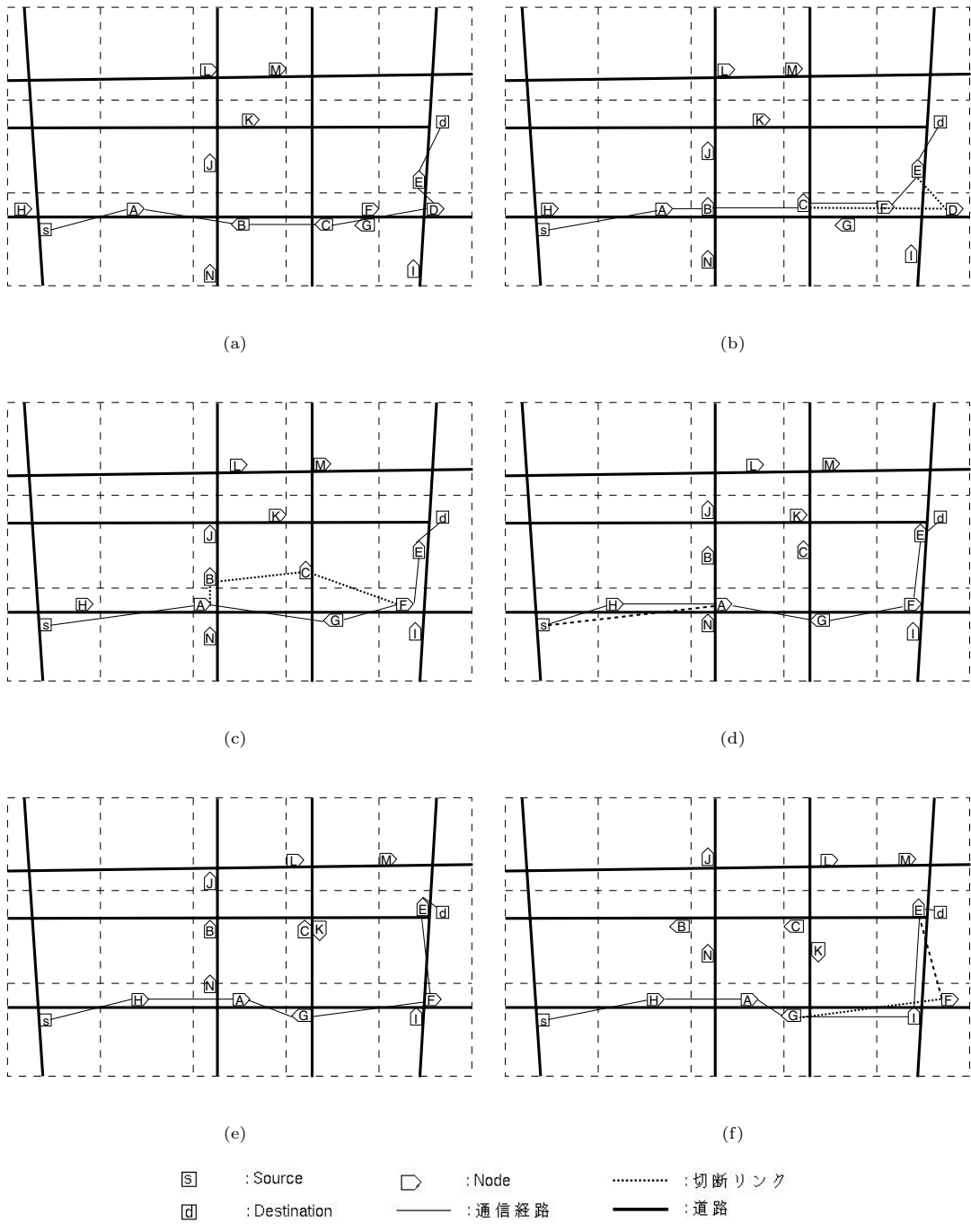


図 2 車両の移動シナリオと GVGrid における経路切断状況

を最新の経路と判断して接続する (図 3(d)).

図 3(e) では, AODV は, ノード J, K 間が障害物により, ノード K, 目的地 d 間のリンクが距離の増大により切断が発生し, リンク切断を検知したノード J が RREQ メッセージのブロードキャストによる

経路修復を開始し, 最新の経路を再構築する. 同時刻において, GVGrid でリンクの切断は発生していない (図 2(e)).

図 2(f) は, GVGrid において, ノード G, F 間のリンクが距離の増大により, ノード F, E 間のリンク

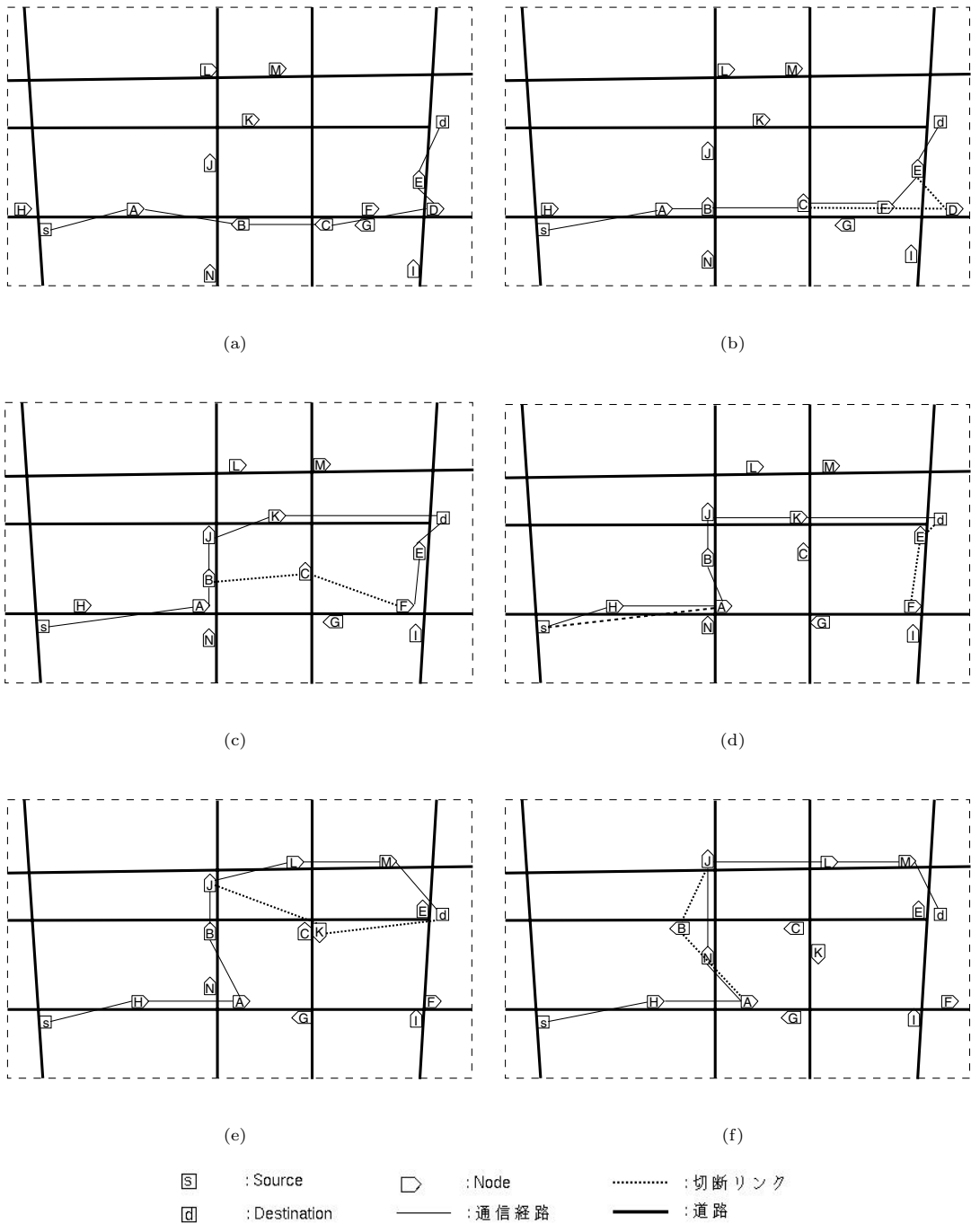


図 3 車両の移動シナリオと AODV における経路切断状況

は交差点における障害物などの影響により切断した状態を表しており、ノード G は前方グリッドから代替ノード I を選択して経路の修復を開始して残存経路と接続する。これに対し図 3(f) では、AODV においてノード A, B 間とノード B, J 間のリンクが障害物に

より切断が発生し、リンク切断を検知したノード A が経路修復を開始し、また残存経路を最新の経路と判断して接続する様子を表している。

3.4 計測ツール

GVGrid と AODV のスループットの計測は、実験

で使用するノードとは別に実験中のパケットをキャプチャするノードが行う(以下パケットキャプチャノード). パケットキャプチャノードは, 無線 LAN の設定をプロミスキャスモードにすることで実現している. プロミスキャスモードとは, 任意の宛先のフレームを傍受することができるモードであり, これを利用することで実験において使用する全てのノードが送信するパケットをキャプチャすることができる. また, キャプチャしたパケットの分析には, パケット分析ツールである Ethereal⁸⁾ を使用した.

3.5 評価結果

ICMP パケットを ping プログラムにより連続送出し, 3 分間の実験シナリオを元にした経路の切断と修復における GVGrid と AODV のスループットを計測した. 図 4(a) は GVGrid, 図 4(b) は AODV のスループットを示している.

経路修復プロセスにおいて, 切断が発生してから経路修復が完了するまでの間に送信されたパケットはバッファリングされ, 経路修復完了後にバッファリングされていたパケットが送信される. そのため, 経路の切断が発生するとデータパケットの送信を停止し, 経路修復のための制御パケットを送信するためスループットは低下し, 経路修復後にはバッファリングしていたデータパケットを送信するため, スループットが一時的に大きくなる.

図 4(b) から, AODV では, 各経路切断の際にはスループットが著しく低下しており, 経路修復後のスループット値は一時的に大きくなっていることがわかる. AODV では, 経路の修復を開始すると最新の経路を構築しようとするため, 部分経路が存在する場合でもその経路が最新でない場合は経路の探索を行う. そのため経路修復にかかる時間が長くなり, 経路の切断を検知したノードにおける ICMP パケットのバッファリング数が増えたためであると考えられる. これに対し図 4(a) より, GVGrid は, 各経路切断の際にはスループットは低下しているが, 低下の度合は低く, 経路修復後のスループット値も AODV ほど大きくない. GVGrid は, 経路の修復において部分経路が存在する場合はその経路を再利用するため, 経路修復にかかる時間が短く, 経路の切断を検知したノードにおける ICMP パケットのバッファリング数が少ないためと考えられる.

以上のことから, AODV と比較して GVGrid では, 経路修復の間のデータのバッファリングが少なく済み, これは各ノードのバッファサイズを小さくできる, あるいはビットレートを上げるなどした場合でもデータ

表 1 制御パケット数と ICMP パケット数

プロトコル名	制御パケット数	ICMP パケット数
AODV	2503	3450
GVGrid	2201	3453

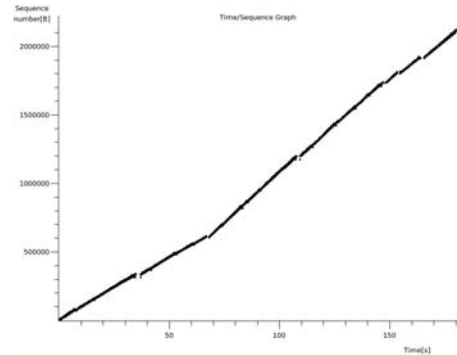


図 5 GVGrid の TCP シーケンス

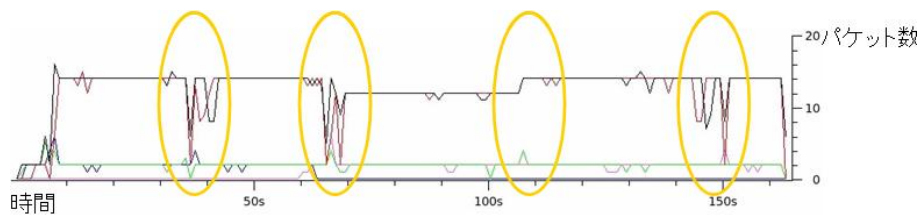
のバッファ溢れが起こる可能性が減り, 安定したデータ送信を行うことができることを示している. また, 切断回数も少ないことから, 通信に与える影響は極めて小さいものと考えられる.

表 1 は, GVGrid と AODV におけるパケット数を示している. GVGrid と AODV は, 同程度の ICMP パケット数を送信しているのに対し, 制御パケット数にはかなりの差が表れている. AODV は経路の切断が発生した場合, RREQ メッセージをブロードキャスト送信して経路の修復を行い, また部分経路が存在する場合でもその経路が最新でないときは, 制御メッセージを利用して経路の探索を行う. これに対し, GVGrid では経路の切断が発生した場合, RRPR メッセージを特定のノードへ送信して経路の修復を行い, また部分経路が存在する場合はその経路を再利用する. AODV と比較して GVGrid で制御メッセージが減った要因は, 経路修復において, AODV では, 残存経路が最新でない場合, 修復された部分経路以外に最新の経路探索においても制御パケットが使用されているが, GVGrid は, 修復された部分経路でのみ制御パケットが使用されているためと考えられる.

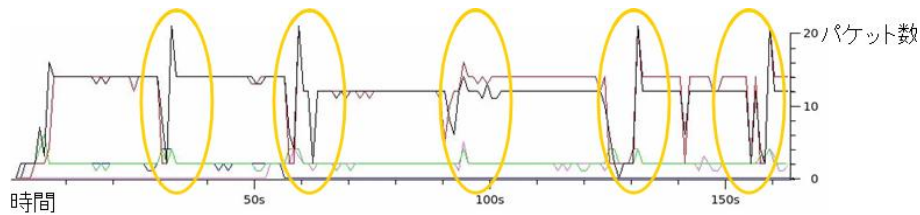
最後に, 同じシナリオを用い, GVGrid による経路上で TCP 通信を行った場合のパケット到着時刻を計測した. 結果を図 5 に示す. 横軸が時刻, 縦軸がその時刻までに受信を完了したシーケンス番号を示す. おおよそスループットの大きな減少も観測されず, スムーズな通信が実現できている.

4. あとがき

本研究では, 研究グループが提案している車車間



(a) GVGrid



(b) AODV

— Source
 — Destination
 ○ 経路修復部分

図 4 GVGrid と AODV のスループット

ルーティングプロトコル GVGrid により構築される通信経路の品質を評価するためにそのプロトタイプ実装を行った。また、ノート PC10 台程度からなる小規模テストベッドを構築し、車両の移動を考慮して作成したリンク切断シナリオにもとづいて経路のリンク切断、復元をテストベッド上で実現し、ICMP パケットを連続送出した上で GVGrid と AODV による経路のスループット評価を行った。その結果、GVGrid は AODV と比較してより高いスループットを達成し、また制御パケット数が少ないことを確認した。

今後の課題としては、テストベッドの機能強化が挙げられる。例えばトラフィックシミュレータから得られる車両位置情報の利用、ストリーミング実験(リアルタイム配信)を通じた動画品質、デバッグ機能の実装などが挙げられる。

参考文献

- 1) 孫 為華, 山口弘純, 楠本真二: 安定性の高い経路を構築する車車間ルーティングプロトコル, 情報処理学会論文誌, Vol.47, No.7, pp.2141-2150 (2006).
- 2) Sun, W., Yamaguchi, H., Yukimasa, K. and

Kusumoto, S.: GVGrid: A QoS Routing Protocol for Vehicular Ad Hoc Networks, *Proc. of 14th IEEE International Workshop on Quality of Service (IWQoS2006)*, pp.130-139 (2006).

- 3) Uppsala University: aodv-uu. <http://core.it.uu.se/core/index.php/AODV-UU>.
- 4) Zhang, Y. and Li, W.: An Integrated Environment for Testing Mobile Ad-Hoc Networks, *Proc. of ACM Int. Symp. on Mobile Ad Hoc Networking and Computing (MobiHoc'02)* (2002).
- 5) Nordstrom, E., Gunningberg, P. and Lundgren, H.: A Testbed and Methodology for Experimental Evaluation of Wireless Mobile Ad hoc Networks, *Proc. of IEEE TridentCom 2005* (2005).
- 6) Ott, M., Seskar, I., Siracusa, R. and Singh, M.: ORBIT Testbed Software Architecture: Supporting Experiments as a service, *Proc. of IEEE TridentCom 2005* (2005).
- 7) : iptables. <http://www.iptables.org>.
- 8) : Ethernet. <http://www.ethereal.com>.
- 9) Perkins, C. E., Belding-Royer, E. M. and Das, S.: Ad Hoc On Demand Distance Vector (AODV) Routing, *RFC 3561* (2003).