車両間通信を用いた事故車両情報の高速伝搬

安藤 洸将^a 北園 優希^a

^a北九州工業高等専門学校

1. はじめに

近年における国内の交通事故発生件数(図 1)は低下の傾向があるが、単純計算でも 1 日に県内で数十件の事故が発生しており、常に交通事故と隣り合わせである事には変わらない。その上近年では総車両台数も増加し(図 1)、道路の単位距離辺りの車両密度が上昇している事が統計より分かっている。交通事故が発生した場合、従来であれば事故状況は発生元付近の人から警察へ伝達、それから交通情報センターへと伝達され、周囲の各車両のナビゲーションシステムへ反映される。上記の場合、情報伝達の連絡網においてかなりのオーバーヘッドが生じる。面倒事を避けようと、事故を警察へ報告しない人なども周りの交通へと大きく影響すると考えられる。車両間通信を利用する事により、一番に情報が必要となる近傍後続車へ高速に事故状況を伝達可能なシステムを実現できる。本システムでは車両の危険行為(急激な速度/加速度変化、無理な割り込み等)を検出し事故を予測する。その予測データは周囲車両とのネットワークで高速に伝達され、事故により車両破損した場合等も事故後に扱えるデータとして保存される。事故発生後は車両間ネットワークを通じて、事故車両から得たデータや近傍の車両の挙動から事故の規模を予測し、後続車へ事故状況を伝達する。

2. システム構成

2.1 構成概要

本システムは周囲の車両が同システムを搭載している事を想定している。本システムは主に、各種センシングデバイスにより事故検出を行うセンサ部(事故検出部)、コンピュータ(マイクロコントローラ)による情報伝達経路の計算を行う処理部、運転者に事故状況を伝達する為の表示器から成る。センサ部では、各種センシングデバイスにより加速度、位置情報といったデータをリアルタイムに取得する。

処理部ではセンサのデータをコンピュータに取り込み、あらゆるパターンに分けてそのデータを処理する。

High-speed Transmission of Information about Car Accident with Communication between Cars

Kosuke Ando^a, Yuhki Kitazono^a

^aNational Institute of Technology, Kitakyushu College

Fig. 1. Transition graph (number of accidents and total vehicles).

センサのデータはパターンに応じて後続車両や近傍の車両へと、RF モジュールを使用して転送される。

表示器のハードウェアは、カーナビゲーションモニターもしくは運転者が車両に取り付けたスマートフォンを想定している。表示器には周辺の事故状況及びを表示し、運転者へと伝搬する。

2.2 事故検出手順

本システムでは 2 段階で車両事故の検出を行う。1 段階目はセンシングデバイスによる事故検出である。前述した通り、車両の危険行為を加速度センサ等のセンシングデバイスを用いて検出する。取得データは加速度センサ、振動センサ、GPS モジュールによる現位置データを考える。加速度センサや振動センサのデータは、短時間での事故判別に用いる。明らかな事故であれば、即座に処理を開始する事が可能と考えられる。対して GPS モジュールによる位置情報データからは、比較的長期間での事故判別に用いる。事故が発生した事が明らかで無い場合、GPS による位置情報から車両の位置や挙動を把握し、事故を推定する。

2 段階目は周囲車両の挙動による事故規模予測である。1 段階目により事故を断定した後、近傍車両の挙動(車両追従の断絶、不審位置での停止)により事故規模を想定する。例えば、事故判定のあった信号機の無い地点で後続車のみ走行を停止していた場合、その地点では走行不可能な事故が発生していると断定できる。事故情

報は後続車両の本システムを転々とホッピングして伝搬される。ホッピングは近傍の通信ノード同士をアドホックで接続しデータを転送する流れを繰り返す事で、遠距離にデータを伝搬する技術である。これを利用し、高速でデータをやり取りする。(図 2) また、危険行為をセンシングデバイスにより検出した際、その車両の取得データを直ぐに近傍車両へと送信する。これは車両破損によりシステムが破壊された場合を想定し、データを分散する RAID システムをネットワーク間で実装したものである。この動作は危険走行が断続的に行われる限り行われ、危険で無い車両だと判断された場合、データを破棄する。

3. 実験

3.1 通信路確立より切断までの時間測定

車両間通信を高速で行う上で必要とされるのは、通信路を確立してからデータを送り終えるまでの一連のプロセスに掛かる処理時間を短縮する事である。その為、実際にどの程度の処理時間を伴うのか計測を行う必要があると考え、実験を行った。通信モジュールには **XBee ZigBee** モジュールを用いた。**WiFi** モジュールの使用も検討したが、コネクションの制約が多く通信確立までに時間がかかると予想する為、**ZigBee** モジュールを用いた。2 台の **ZigBee** モジュールを用意し、離れた位置から電波が届く位置までゆっくり近づき、電波を受信した瞬間 (RSSI を取得できた瞬間) にデータを送信する。10[Byte], 1k[Byte], 10k[Byte] についてそれぞれ転送時間及びデータの整合性を調べた。10[Byte] 送信は、緊急にデータを送信するケースを想定した。1k[Byte] 送信は、1 台辺りの事故情報を送信するケースを想定した。10k[Byte] 送信は、周辺車両の情報を送信するケースを想定した。実験結果より、実現的な時間が観測できたが、通信を行う 2 車両の速度の差や事故の規模が大きいと、データが最後まで伝送できない場合がある事が分かった。

3.2 加速度センサによる衝突検出

センサ部では加速度センサを用いて、車両の危険行為や衝突を検出する。速度に関しては **GPS** による位置情報の変位から導出する。速度と加速度を用い、衝突と非衝突の区別を明確にする。本実験では **Android** 端末を用いた。実システムにおいて、ユーザーにとって身近なスマートフォンを使用することでコストを削減できる。また、スマートフォンは表示器としても使える為、今実験では危険を検出した際に運転者へ通知する機能を実装した。**Android** 端末に **USB** ホストケーブルを装着し、**USB-UART** 変換基板を用いて処理部 (マイクロコントローラ) へとデータを送信する。マイコンには **Renesas** エレクトロニクス社の **RX-62G** を用いた。これは車載機器間の通信に頻繁に用いられる **CAN (Controller Area Network)** 通信の機能を周辺機能として搭載しており、車

Fig. 2. Example of data hopping route.

輛機器に組み込み際に大変都合が良い。本実験では実際に衝突を検出する事が出来た。外乱対策として、被衝突側とのセンサ値比較の機能を搭載した。

4. まとめ

実験より、データ送信のタイミングがこのシステムにおいて最も重要な点である事が分かった本システムに残っている大きな課題として、センサによる事故の誤検知防止や通信経路選択、利便性の向上などが挙げられる。検出性能の上昇に関しては、センサのデータを予めサンプリングしたり周囲のシステムの値と比較したりすることで精度を上昇する事を考えている。利便性の向上としては、今回スマートフォンを用いたと同様に、身近なシステムと連携していく事が望ましいと考える。

文 献

- (1) 警視庁,平成 27 年における交通事故の発生状況,3 月 2016 年.