

VANET におけるネットワークコーディングを用いた 位置依存情報配布の検討

楠嶺生宏[†] 石原進[‡]

[†] 静岡大学工学部

[‡] 静岡大学創造科学技術大学院

1 はじめに

VANET (Vehicular Ad hoc NETwork) において車両が特定位置から発した情報を周辺車両に対してフラッディングで配信する場合を考える。この時、車両密度が高い場合にはパケット衝突によるロスが頻発し、配信が失敗したり他の通信を阻害する恐れがある。また車両密度が低い場合にはパケットの転送経路が存在せず、目的とする範囲への情報配信が出来ないことが予想される。このような条件での情報配信の信頼性を向上するためにランダムネットワークコーディング (RNC) [1] の利用を考える。RNC を適用したアドホックネットワーク上での情報配信に関する研究は、ランダムウェイポイント移動モデルでの有効性は検証されているが [2]、道路構造や車両の移動特性を考慮した場合の効果は十分に検証されていない。

そこで本稿では、VANET 上での位置依存情報のプッシュ型配信において RNC を適用する方法、ならびにその課題について議論する。

2 ランダムネットワークコーディング (RNC)

2.1 RNC の概要

RNC[1] は、中継端末が複数のパケットに対してそれぞれランダムな係数を選択し、ガロア体での線形結合をとることでネットワークコーディングを行う手法である。入力パケット $P_i (i = 1, 2, \dots, N)$ に対し、1 回の符号化処理対象パケット数を N とする。中継端末にはランダムに選ばれた係数ベクトル $C = (C_1, C_2, \dots, C_N)$ が与えられ、受け取ったパケットとの線形演算が行われ、符号化パケット ($p_{out} = \sum_{i=1}^N P_i C_i$) が出力される。 p_{out} は次の中継端末への入力パケットとして用いられる。受信端末に届いた複数の p_{out} パケットは各パケットに付加して送られる係数が線形独立であれば元のパケット P_i を復元可能である。

無線ネットワークでは、一度のパケット送信で隣接端末全てにパケットを受信させることができるため、容易に異なる符号ベクトルによる符号化パケットをネットワーク上に流通させることが出来る。このため、端末の移動によりパケットロスがより発生しやすい状況であったとしても、中継された符号化パケットを収集することで配信データを高い確率で復元できることが見込める。

2.2 VANET における RNC 適用例

VANET における RNC 適用の車々間通信の例として CodeTorrent[2] が挙げられる。

CodeTorrent では、ファイル P を保持するノードがそのメタデータを定期的にブロードキャストする。メタデータに対して関心を持ったノードはそのファイルに対する要求を定期的にブロードキャストし続ける。要求を受け取ったノードは、ファイルを持っていればファイルを N 個の断片に分割し、ランダムに選択される係数を用いて符号化パケットを生成してブロードキャストする。符号化パケットを N 個以上保持したノードは、各符号化パケットで用いられている係数がそれぞれ線形独立である場合、元のファイル P を得る。文献 [2] では、この手法により、VANET 上での RNC を用いたファイル配信が、ユニキャストベースでのファイル要求と応答に基づくファイル配信に比べて有効であることが示されている。

3 VANET での RNC 適用の位置依存情報配信

3.1 前提条件

CodeTorrent が比較的大きな許容遅延の下で音楽データなどの大きなデータを配信することを目指しているのに対し、本研究では、事故や渋滞状況を表す画像、映像などの特定位置に関連付けられている情報 (位置依存情報) を、出来るだけ低遅延で限られた配信範囲 (有効領域) の端末に対して高い配信率で配信することを目的とする。

有効領域は、位置依存情報の生成位置から半径 L 以内の範囲とし、 L は無線通信可能距離に対して十分大

A Study of location dependent information distribution with network coding on VANET
Naruhiko Kusumine[†], Susumu Ishihara[‡]
[†]Faculty of Engineering, Shizuoka University [‡]Graduate School of Science and Technology, Shizuoka University
kusumine@ishilab.net

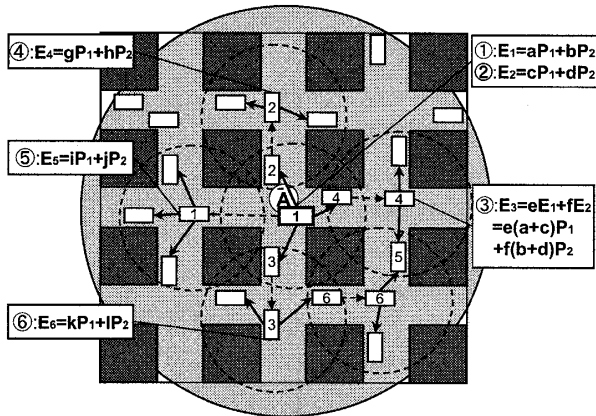


図 1: RNC 適用の位置依存情報配信

きいものとする。各端末で生成される位置依存情報は 1 回のブロードキャストでは全て送信できない大きさであるとする。また各端末は定期的に自らの現在位置を Hello パケットに含めてブロードキャストするものとする。

3.2 位置依存情報配信方法

各端末は、位置依存情報を生成すると、それを N 個の断片に分割して保持し続ける。各端末は、他端末から Hello を受け取ると、自身が保持する情報の断片、あるいは他端末から受信した符号化パケットの生成位置が受信した Hello の送信者の有効領域内であるならば、それらをもとに符号化パケットを生成してブロードキャストする。符号化方法は CodeTorrent と同様である。

符号化パケットを受信、またはオーバーヒアした端末は、他端末から Hello を受信すると、Hello で要求される情報の有効領域内で生成された情報に対する 1 つ以上の符号化パケットをもとに新たな符号化パケットを生成し、ブロードキャストする。

符号化パケットを受信すると、端末は新たに受信した符号化パケットと同一の情報に対するそれまでに保持している符号化パケットをもとに情報を復元する。

図 1 に具体例 ($N=2$) を示す。端末 1 は A 地点において断片 p_1, p_2 からなる位置依存情報 P を生成し、保持している。端末 1 は隣接端末 2 から要求 (=Hello) を受信すると、自身が持つ p_1, p_2 とランダムに選択した係数 (a, b) を線形演算して符号化パケット E_1 を生成してブロードキャストする。また別の隣接端末 4 からも Hello を受信すると、新たに係数 (c, d) をランダムに選択して別の符号化パケット E_2 を生成してブロードキャストを行う。(図中①②)

符号化パケットを受信、オーバーヒアした隣接端末

234 は、符号化パケットを必要以上保持している場合に復号動作を開始する。各符号化パケットに含まれる係数が線形独立である場合、元の情報 p_1, p_2 を復号化する。

復号化が出来た断片、符号化パケットを受信した端末はそれらを保持し、新たに Hello を受信したときにそれらを用いて新たな符号化パケットを生成する。(図中③~⑥) 図中③で端末 4 は受信した E_1, E_2 をもとに新たな符号化パケット $E_3 = eE_1 + fE_2$ を生成し、ブロードキャストしている。

3.3 検討事項

VANET での位置依存情報の RNC による配信の実現上の課題を議論する。

符号化パケットの復号には一定数以上の符号化パケットが必要になる。情報生成直後では、十分な数の異なる係数で符号化した符号化パケットを多数送信しておかなければならない。車両密度が高い場合、Hello の受信率も必然的に高くなるため、多種の符号化パケットを生成することが出来る。この結果、多くの端末が Hello に対して符号化パケットを送るようになるためデータの復号成功率は上がるが、符号化パケットにより通信路が占有され、他の通信を阻害する恐れがある。このため、符号化パケットの過多を防ぐための工夫が必要である。他方、車両密度が低い場合では、Hello を受信する機会が少ないため、積極的に符号化パケットを送信する必要がある。これらの対策として、Hello の受信頻度等に基づいて推測した車両密度、および位置依存情報生成位置と端末間の距離に基づいて符号化パケットの送信を実施するかどうかを決めるということが考えられる。

加えて、Hello での要求に対して適応する情報の符号化パケットを複数所持していた場合、どの符号化パケットを送信するかの判定基準が必要である。またストレージの浪費を防ぐために中継中のデータの破棄を行う基準が必要である。

4 まとめ

本稿では VANET における RNC 適用の位置依存情報のプッシュ型配信方法を示し、実現上の課題と対策を議論した。今後、以上の議論を踏まえて同手法の効果を検証する予定である。

参考文献

- [1] T. Ho, M. Medard, et al. "A random linear network coding approach to multicast," in Proc. of IEEE Trans. Information Theory, vol.52 pp 4413-4430, Oct. 2006
- [2] U. Lee, J. S. Park, J. Yeh, et al. "CodeTorrent: Content Distribution using Network Coding in VANET," in Proc. of MobiShare, Sep. 2006