

doi:10.13756/j.gtxyj.2019.04.012

无线通信技术

# 一种新的滑动窗口网络编码模型与算法研究

桂 超,孙宝林

(湖北经济学院 信息与通信工程学院,武汉 430205)

**摘要:**针对移动自组网(MANET)依靠滑动窗口机制解决传输效率和流量控制问题,文章提出了一种可变长度的滑动窗口网络编码(VLSWNC)算法。利用网络仿真软件 NS2 对 VLSWNC 算法中的数据吞吐量、分组传输率和解码延迟等方面的性能特征进行了仿真实验分析。实验结果表明,VLSWNC 算法可以显著提高 MANET 的性能和可靠性。

**关键词:**移动自组网;网络编码;滑动窗口;解码

**中图分类号:**TN92 **文献标志码:**A **文章编号:**1005-8788(2019)04-0061-05

## Research on A New Sliding Window Network Coding Model and Algorithm

Gui Chao, Sun Bao-lin

(School of Information and Communication Engineering, Hubei University of Economics, Wuhan 430205, China)

**Abstract:** In order to solve the problem of transmission efficiency and flow control by sliding window mechanism in Mobile Ad Hoc Network (MANET), a Variable Length Sliding Window Network Coding (VLSWNC) algorithm is proposed in this paper. The performance of this VLSWNC is studied using NS2 and evaluated in terms of the data throughput, packet transmission rate, and decoding delay when packet is transmitted. Simulations results show that the VLSWNC algorithm can significantly improve the network performance and reliability.

**Key words:** MANET; Network Coding (NC); sliding window; decoding

### 0 引 言

移动自组网(Mobile Ad Hoc Networks, MANET)是由大量具有移动计算及通信功能的移动节点组成的一种多跳移动自组织网络。网络编码(Network Coding, NC)最初是由 Ahlswede 等在文献[1]中首次提出的,其主要目的是解决无线网络中的数据流量传输问题,是一种新的编码概念,理论上已证明了 NC 能够提高无线网络的数据传输效率,但没有给出具体的 NC 实现过程及方法;Jaggi S 等在文献[2]中针对无线网络的有向无环图的边缘节点容量问题,提出了一种随机线性 NC 算法,并给出了 NC 的构造方法,其算法的复杂性为多项式时间复杂度数量级;Yeung R W 等<sup>[3]</sup>讨论了 NC 的发展与应用问题,提出了一种线性 NC 方案,该方案较好地提高了 NC 的编/解码能力和数据吞吐量;文献[4]就 NC 的基本概念、分类、构造方法以及应用领域等方面进行了分析研究,并对转接节点中应用 NC 与非网络编码两种方法进行了比较研究;Aktas T 等<sup>[5]</sup>针对无线网络中的多个单播路由传输协议问题,提出了一个构造简单、有效的无线 NC 算法及相

应的解码算法,该算法降低了编/解码的复杂性;Lin Y 等<sup>[6]</sup>提出了一种无线网络中高效、可靠的分组传输机制,该机制采用多路径路由传输且多个不相交路径上应用 NC 算法,允许在中间节点上对数据分组进行 NC,提高了分组的传输效率;文献[7]提出了一种在无线传感器网络中的 NC 多路径路由算法,该算法提高了无线传感器网络中的数据可靠性和网络负载均衡能力;文献[8]提出了一种 MANET 中基于优先级的分组多路径路由 NC 算法,该算法主要利用模糊控制机制确定优先分组参与 NC。研究结果表明,基于 NC 的无线网络中使用 NC 技术的数据吞吐量和传输效率等要优于传统编码技术。

### 1 可变长度滑动窗口

可变长度滑动窗口与传统的固定滑动窗口不同,其可以自适应地调整滑动窗口的大小,根据数据分组流量的变化和分组到达中间节点和缓存区的消耗能力来调整窗口的大小,从而降低编/解码的处理时间。

可变长度滑动窗口的特点是:滑动窗口中包含

收稿日期:2019-01-16

基金项目:国家自然科学基金资助项目(61572012);湖北省自然科学基金资助项目(2017CFB598,2017CFB677)

作者简介:桂超(1964—),男,湖北武汉人。教授,硕士,主要研究方向为无线网络路由编码技术及信息安全。

通信作者:孙宝林,教授。E-mail: 1392979674@qq.com

一定的所需的冗余信息,以此确定滑动窗口的变化,更好地减少解码错误。然而,由于可移动节点运动越快,解码错误就越多,此时就要减少滑动窗口的长度;当移动节点的速度变缓时,可以适当增加滑动窗口的长度。

可变长度滑动窗口需要两个窗口:发送方持续发送数据分组集,称为发送窗口;接收方持续接收一组连续的数据分组集,称为接收窗口。

1.1 滑动窗口模型

MANET 可以表示成一个无向图  $G=(V, E)$ , 式中:  $V$  为网络中的节点集合;  $E$  为网络中两个节点之间的无线通信链路集合。每个无线通信链路  $e=(a, b) \in E$  表示无线节点  $a$  与  $b$  是直接相连的, 并且假设链路是对称的, 即  $(a, b) \in E, (b, a) \in E$ , 且  $(a, b)=(b, a)$ 。

假设  $P=(P_1, P_2, \dots, P_m)$  为一个有限集; 处理分组集为  $T=(t_{id}, x_1, x_2, \dots, x_n)$ ,  $t_{id}$  为分组集的分组编号,  $x_i \in P, x_i(i=1, 2, \dots, n)$  为有限域的非空子集; 数据分组流  $T=(T_1, T_2, \dots, T_n)$  为一个等待发送的数据分组序列,  $T_n$  为最近的数据分组。假设当前的滑动窗口为  $SW_{n-|w|+1}=(T_{n-|w|+1}, T_{n-|w|+2}, \dots, T_n)$ , 同时也定义了分组集  $T$  中包含  $|w|$  个数据分组,  $|w|$  和  $n-|w|+1$  分别为窗口大小和窗口标识符。当新的数据分组从  $T$  中通过更新机制滑动到新的  $SW$  中时, 窗口最老的数据分组将会从窗口中删除,  $SW$  可以继续向前滑动。

图 1 所示为滑动窗口处理数据分组与窗口滑动机制的示意图, 窗口大小为 5。第一个滑动窗口  $SW_1$  由前 5 个分组组成, 其他滑动窗口  $SW_2$  和  $SW_3$ , 每个窗口都是通过向之前的滑动窗口插入一个新分组并从当前窗口中删除最老的分组而形成的。当前的滑动窗口是最后一个滑动窗口  $SW_3$ 。给定一个最小阈值, 新的滑动窗口生成过程定义为在当前的滑动窗口中较好性能的滑动窗口集合。每个窗口滑动都必须更新分组。

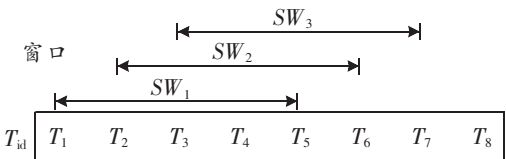


图 1 滑动窗口处理数据流的示意图

1.2 滑动函数

滑动窗口: 设  $D$  和  $R$  为两个有限整数集, 函数  $f: D^n \rightarrow R$  表示长度为  $n$  的函数。用  $\Gamma$  表示滑动窗口的一个滑动操作, 并定义  $f^\Gamma: D^{n+1} \rightarrow R'$ , 且

$f^\Gamma(x)=(f(x_i, \dots, x_{i+n-1}))_{i=1}^t$ 。可以考虑在顺序号  $t=n$  的情况下, 将滑动窗口操作应用于函数  $F_k$ ,  $F_k$  取模 2, 用符号  $F_k^{(j)}$  表明从位置  $j$  开始的长度为  $n$  的窗口中的第  $k$  次变化情况。

定义:  $((\epsilon, \epsilon')-\text{平滑函数})$ ,  $f$  为集合中某点的函数, 并且设定  $\epsilon, \epsilon' \in (0, 1)$ 。  $f$  为  $(\epsilon, \epsilon')$ -平滑函数, 如果  $f$  是一个非负(即:  $f(A) \geq 0$ , 对所有的集合  $A$ ) 非递减(即: 如果  $A \subseteq B$ , 则  $f(A) \leq f(B)$ ) 的平滑函数, 并且多项式有界(即存在  $c > 0$ , 使得  $f(A)=O(|A|^c)$ ) 且所有的集合  $A, B$  和  $C$ , 满足  $f(B) \geq (1-\epsilon)f(A \cup B)$ ,

同时  $f(B \cup C) \geq (1-\epsilon')f(A \cup B \cup C)$ 。(1)

1.3 可变长度滑动窗口

传统的滑动窗口算法使用固定长度的窗口值, 导致算法的效率不高。滑动窗口的长度与算法的计算复杂度成正比例关系, 为了有效地提高滑动窗口的性能, 本文提出了一种可变长度滑动窗口 NC (Variable Length Sliding Window Network Coding, VLSWNC) 算法, 该算法可以根据 NC 节点和解码节点的性能自适应调整滑动窗口的大小。图 2 所示为可变长度滑动窗口的滑动机制, VLSWNC 算法的结构、滑动过程与传统的滑动窗口算法类似, 只是在算法中根据自适应窗口的性能来调整滑动窗口的大小。VLSWNC 算法将自适应地设置每个滑动窗口的长度, 可以避免滑动窗口长度过大或太小。滑动窗口长度将根据网络中的编码与解码节点的性能进行窗口大小的调整, 并使滑动窗口的大小倾向于最佳值。

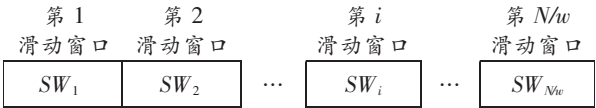


图 2 可变长度滑动窗口的滑动机制

在传统的滑动窗口算法中, 滑动窗口的长度是固定的。本文提出的 VLSWNC 算法中, 对每一个编码和解码窗口来说, 滑动窗口的长度是自适应变化的, 而且每次的迭代取决于向后递归获得的可靠性信息。可变长度滑动窗口长度为  $w$  信息位, 数据分组为  $N$  信息位,  $N$  为  $w$  的整数倍; 定义第  $i$  个向前递归  $(1 \leq i \leq N/w)$  的计算方法, 设当前滑动窗口为第  $i$  个滑动窗口, 则通过向前递归的思想, 采用加权的方法生成当前的滑动窗口长度, 计算公式为

$$SW_i=(1-\alpha) \cdot SW_{i-1}+\beta \cdot SW_{i-2}, \quad (2)$$

式中:  $\alpha, \beta$  为权重因子  $(\alpha+\beta=1, 0 \leq \alpha < 1, 0 \leq \beta < 1)$ ;  $SW_i$  为第  $i$  个滑动窗口的长度。当服务的行为

发生改变后,当前的滑动窗口值  $SW_i$  就会更新,当前的滑动窗口也会更新为新的窗口。

#### 1.4 滑动窗口的概率分配

滑动窗口边界值的确定是根据总数据分组的多少和参加编码的分组数决定的,分组组合过程的下一个阶段是根据周围窗口所包含的分组为每个分组分配一个概率值。

定义  $P(d_{p,w} | s)$  为分组  $d$  在位置  $p$  的相关概率,  $p$  的另一边为使用窗口长度为  $w$  信息位确定的边,得到输入系统  $s$  的返回值为可变长度滑动窗口长度为  $w$  信息位,

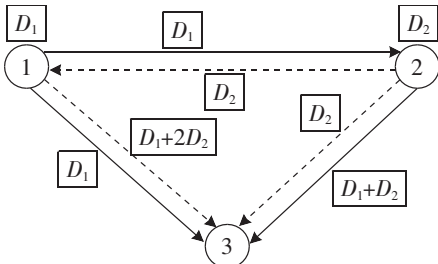
$$P(d_{p,w} | s) = \frac{\sum_{i=h}^k P(d_i | s)}{k - h + 1}, \quad (3)$$

式中:  $k$  为选择的一组加权系数  $(k_1, k_2, \dots, k_m)$  中的最大值;  $h$  为选择的线性回归函数系列的最小值;  $d_i$  为第  $i$  个数据分组。与使用基于每个位置上的数据概率相比,滑动窗口机制可以使滑动窗口中的数据分组数量达到最优。

## 2 NC 机制

### 2.1 NC 过程

VLSWNC 采用了与随机线性 NC (Random Linear Network Coding, RLNC)<sup>[9]</sup> 类似的 NC 策略,但每个时隙中待编码的数据分组数量需要以恒定的速度向前滑动。RLNC 机制采用了分组传输策略,从而能以更少的反馈开销来达到最佳容量。因为 RLNC 的编/解码机制能够提供几乎满容量的性能,因此,它在最近的典型文献[4~5, 10]中非常具有吸引力。此外,它在非二进制伽罗华域  $GF(2^q)$  上工作,可以与 NC 无缝地结合在一起。图 3 所示为在有限域上双节点实现 NC 的处理与传输机制结构图。在信道编码的顶层,使用  $GF(2^q)$  上的 NC 来编码本地以及节点的信息。



注: 1、2、3 分别为网络中的节点;  $D_1$ 、 $D_2$  为节点发送的分组;  $D_1+D_2$  为 NC 后生成的分组; 箭头表示分组传输的方向。

图 3 NC 处理与传输机制

由图可知,NC 机制在每个编码节点上都是时

间不变的。  $D_1$  和  $D_2$  在  $GF(2^q)$  上实现,NC 也在  $GF(2^q)$  中实现,所有的传输分组都遵循独立衰退。

### 2.2 RLNC

RLNC<sup>[9]</sup> 是在一个节点上对一个或多个数据分组进行线性组合的 NC。假设每个分组长度为  $L$  位,每连续  $U$  个比特在域  $GF(2^q)$  上形成了一个数据块,由此可以得到  $L/U$  个数据块。对每个数据块在域  $GF(2^q)$  上进行加法和乘法等相关的线性运算,编码后的数据分组组合成新的数据分组,编码后的分组长度依然是  $L$  位。假设在源节点处产生原始数据  $M$ ,  $M$  可以分为  $n$  个数据分组序列  $M_1, \dots, M_n$ 。在 RLNC 中,线性 NC 系数  $g_1, \dots, g_n$  可以从域  $GF(2^q)$  中产生(见公式(6))。编码的分组可表示为

$$X = \sum_{i=1}^n g_i M_i. \quad (4)$$

一个编码分组并不能携带源数据分组所有的编码信息。解码节点必须要接收到满足解码要求的  $m$  个编码分组 ( $m \geq n$ ) 以及  $(g^1, X^1), \dots, (g^m, X^m)$  系数对,才能恢复出原始的数据包  $M_1, \dots, M_n$ 。换言之,需要解  $m$  个方程和  $n$  个变量,

$$X^v = \sum_{i=1}^n g_i^v M_i, \quad 1 \leq v \leq m, \quad (5)$$

式中:  $X^v$  为解码后产生的分组;  $g_i^v$  为对应的解码系数;  $v$  为分组序列数。

为了编/解码这些信息,需要解对应的线性方程组,且对应的系数向量应该是线性无关的,其  $n \times m$  的矩阵应该也是满秩的,系数矩阵可表示为

$$G = \begin{bmatrix} g_1^1 & \dots & g_n^1 \\ \vdots & \ddots & \vdots \\ g_1^m & \dots & g_n^m \end{bmatrix}, \quad (6)$$

由式(5)、(6)可以解出原始数据分组  $M_1, \dots, M_n$ , 且满足:

$$M = G^{-1} \times X. \quad (7)$$

### 2.3 分析

本节提供了一些简单的边界条件用于验证测试平台的操作和实现。发送和接收分组速率的上限是在假设每个协作数据分组可以并行地在不受干扰的情况下进行通信的条件下设定的,因为这是最好的情况。

滑动窗口的平均大小为  $W$ , 数据包和发送节点的平均数为  $N/W$ , 数据分组以速率  $r$  发送, 且该组合速率  $R_s$  须在  $N$  个节点间分布,

$$R_s \leq \frac{N}{W} \cdot \frac{1}{N} \cdot r = \frac{r}{W}. \quad (8)$$

在  $N$  个节点中,平均  $N/W$  个分组发送,余下的节点以速率  $r$  接收,也就是每个节点的平均接收速率  $\bar{R}_s$  为

$$\bar{R}_s = \left(N - \frac{N}{W}\right) \cdot \frac{1}{N} \cdot r = \left(1 - \frac{1}{W}\right) \cdot r, \quad (9)$$

完成时间的下限是在一次只能有一个节点发送的假设下设定的,因为这是最坏的情况。

假设节点对于发送的数据包具有  $W$  个滑动窗口。经过理论推导,分析 VLSWNC 算法的编/解码复杂度。对于传输控制协议(Transmission Control Protocol, TCP)/NC 算法解码的前向消除部分时间复杂度较高,即  $O(W^3)$ ,随着  $W$  的增加,其平均复杂度也随之增加;VLSWNC 算法解码原始数据分组的时间复杂度为  $O(W^2)$ 。后面的仿真结果也明显支持分析结果。TCP/NC 算法的解码复杂度随着吞吐量的增加而急剧增加,对 VLSWNC 算法而言,接收方解码系数矩阵是稀疏的,因为它具有较少的非零项。因此,VLSWNC 算法能够显著降低目的节点的解码复杂度。

3 仿真实验

3.1 实验参数

本节中将通过计算机模拟对 MANET 中的 VLSWNC 算法进行性能评价。将该算法与延时控制 NC (Delay Controlled Network Coding, DC-NC)<sup>[10]</sup> 及 TCP/NC<sup>[11]</sup> 算法进行比较,利用 NS2<sup>[12]</sup> 仿真实验工具对 VLSWNC 算法的网络吞吐量、丢包率以及解码时延进行性能评价。

在仿真实验过程中,MANET 的无线链路带宽为 5 Mbit/s,分组队列长度为 1 000 个分组,接收窗口大小为 32 个分组。每个分组的长度为 256 字节,每个确认分组的长度为 40 字节,仿真参数如表 1 所示。

表 1 仿真参数

| 参数                   | 数值                                            |
|----------------------|-----------------------------------------------|
| 节点数/个                | 100                                           |
| 网络区域                 | 1 000 m × 1 000 m                             |
| 传输范围/m               | 250                                           |
| 仿真时间/s               | 600                                           |
| 信标周期/ms              | 100                                           |
| 通信模式                 | 固定码率                                          |
| 信号大小尺度/ bytes/packet | 512                                           |
| NC 方案                | RLNC                                          |
| 伽罗华域                 | 8 位                                           |
| 对照路由协议               | DCNC <sup>[10]</sup> , TCP/NC <sup>[11]</sup> |

3.2 仿真结果

图 4 所示为 VLSWNC、TCP/NC 以及 DCNC 3 种算法的网络吞吐量在节点移动速度变化时的比较。由图可知,VLSWNC 算法获得的吞吐量最高,与 TCP/NC 以及 DCNC 算法相比,VLSWNC 算法可以提升 5%~10% 的网络吞吐量。

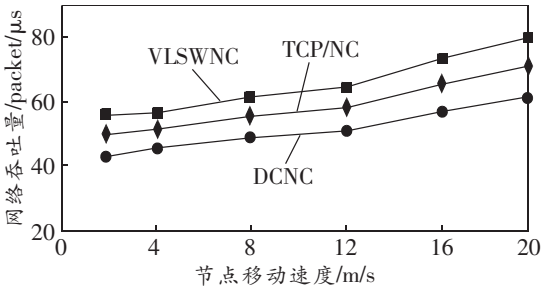


图 4 3 种算法下的网络吞吐量在节点移动速度变化时的比较

图 5 所示为 3 种算法的分组传输率在节点移动速度变化时的比较。由图可知,与 TCP/NC 及 DC-NC 算法相比,VLSWNC 算法可以获得更好的分组传输率。故可变长度滑动窗口可以较好地满足 MANET 分组传输率的要求。

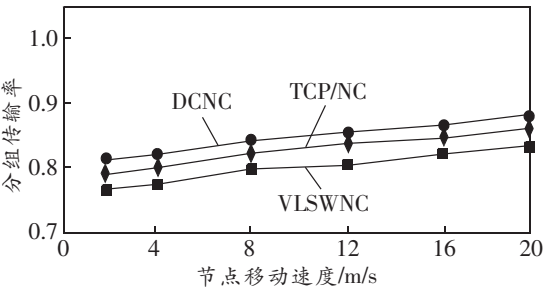


图 5 3 种算法的分组传输率在节点移动速度变化时的比较

解码延迟就是接收节点对接收到的数据分组进行解码的时间。图 6 所示为在 MANET 中节点移动速度变化时 3 种算法解码延迟的比较。由图可知,VLSWNC 算法获得的解码延迟最小。

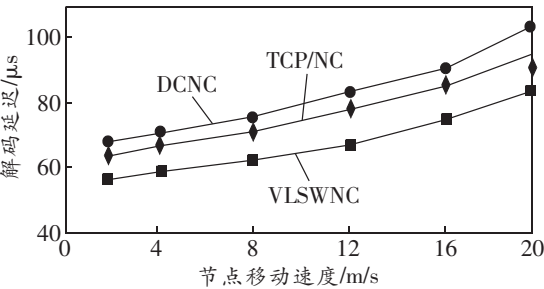


图 6 3 种算法的解码延迟在节点移动速度变化时的比较

## 4 结束语

本文将滑动窗口与 NC 相结合进行研究,提出了一种无线网络 VLSWNC 算法。理论分析表明,VLSWNC 算法能够较好地降低计算复杂性。在 VLSWNC 算法中,应用 NC 和滑动窗口,允许分组在可变长度滑动窗口内进行编码,可以提高网络吞吐量和分组传输率,同时降低解码时延。通过仿真实验表明,VLSWNC 算法能够获得更高的可靠性和吞吐量;VLSWNC 算法在具备较低的编码和解码的同时,提高了 MANET 的性能,改善了 MANET 的网络质量。VLSWNC 算法在无线网络应用中有一定参考价值。

### 参考文献:

- [1] Ahlswede R, Cai N, Li S Y R, et al. Network Information Flow [J]. IEEE Transactions on Information Theory, 2000, 46(4): 1204—1216.
- [2] Jaggi S, Sanders P, Chou P A, et al. Polynomial Time Algorithms for Multicast Network Code Construction [J]. IEEE Transactions on Information Theory, 2005, 51(6): 1973—1982.
- [3] Yeung R W. Network Coding: A Historical Perspective [J]. Proceedings of the IEEE, 2011, 99(3): 366—371.
- [4] Hassan Mohammed A, Dai B, Huang B, et al. A Survey and Tutorial of Wireless Relay Network Protocols based on Network Coding [J]. Journal of Network and Computer Applications, 2013, 36(2): 593—610.
- [5] Aktas T, Ozgür Y A, Aktas E. Practical Methods for

- Wireless Network Coding with Multiple Unicast Transmissions [J]. IEEE Transactions on Communications, 2013, 61(3): 1123—1133.
- [6] Lin Y, Liang B, Li B. SlideOR: Online Opportunistic Network Coding in Wireless Mesh Networks [C]// IEEE INFOCOM, San Diego, CA, USA: IEEE, 2010: 171—175.
- [7] Sun B L, Gui C, Song Y, et al. A Novel Network Coding and Multi-Path Routing Approach for Wireless Sensor Network [J]. Wireless Personal Communications, 2014, 77(1): 87—99.
- [8] Sun B L, Song Y, Gui C, et al. Network Coding-based Priority-Packet Scheduler Multi-Path Routing in MANET Using Fuzzy Controllers [J]. International Journal of Future Generation Communication and Networking, 2014, 7(2): 137—147.
- [9] Ho Tracey, Medard M, Koetter R, et al. A Random Linear Network Coding Approach to Multicast [J]. IEEE Transactions on Information Theory, 2006, 52(10): 4413—4430.
- [10] Qu Y B, Dong C, Chen C, et al. DCNC: Throughput Maximization via Delay Controlled Network Coding for Wireless Mesh Networks [J]. Wireless Communications and Mobile Computing, 2016, 16(1): 137—149.
- [11] Karafillis P, Fouli K, ParandehGheibi A, et al. An Algorithm for Improving Sliding Window Network Coding in TCP [C]// 47th Annual Conference on Information Sciences and Systems (CISS 2013), Baltimore, MD, USA: IEEE, 2013: 1—5.
- [12] VINT Project, UC Berkeley. The Network Simulator — NS2 [EB/OL]. (2019-01-16) [2019-01-16]. <http://www.isi.edu/nsnam/ns/>.