

中继协作多播网络中基于 最小集合覆盖的编码重传方案

王 练^{1,2}, 梁申虎¹, 陈 巧¹, 彭代渊², 梁宏斌²

(1. 重庆邮电大学计算机科学与技术学院, 重庆 400065; 2. 西南交通大学信息科学与技术学院, 四川成都 611756)

摘 要: 为提高单中继协作多播传输效率, 本文提出一种基于最小集合覆盖的分类网络编码重传方案. 该方案充分利用中继节点协作传输的优势, 将接收端的丢包按中继节点的接收状态分为两类, 并按类先后进行编码重传. 在各类丢包内部, 根据对应的状态反馈矩阵寻找编码机会生成编码包, 并将编码包的选择过程抽象为集合覆盖问题, 通过求最小集合覆盖使重传次数逼近最小值. 此外, 在不增加所得前类编码包数的前提下, 利用两类丢包之间的编码机会进一步生成新编码包, 以减少重传编码包数, 从而提高重传效率. 分析与仿真结果表明了该方案的有效性.

关键词: 网络编码; 重传; 调度; 分类; 集合覆盖

中图分类号: TN92 **文献标识码:** A **文章编号:** 0372-2112 (2018)01-0001-07

电子学报 URL: <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2018.01.001

Coded Retransmission Scheme Based on Minimum Set Coverage for Relay-Assisted Multicast

WANG Lian^{1,2}, LIANG Shen-hu¹, CHEN Qiao¹, PENG Dai-yuan², LIANG Hong-bin²

(1. School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China;

2. School of Information Science and Technology, Southwest Jiaotong University, Chengdu, Sichuan 611756, China)

Abstract: A classification network coding retransmission approach based on the minimal set coverage is proposed to improve the single relay cooperative transmission efficiency. In this scheme, to make full use of the advantage of relay-assisted transmission, the lost packets of receivers are classified into two categories according to the packet reception status of relay node after the initial transmission, and then the packets are encoded for retransmission according to the category sequentially. For each classification, according to the corresponding status feed matrix to search for encoding opportunities and generate encoding packets, the encoded packets selecting procedure is abstracted as the set coverage problem, and the number of retransmissions is approached the minimum value by solving the minimal set coverage problem. In addition, to reduce the number of retransmission packets and improve retransmission efficiency, encoding opportunities between the two types of lost packet are further used to generate new encoding packets on the premise of not increasing the current number of encoding packets. The analysis and simulation results show the effectiveness of the scheme proposed.

Key words: network coding; retransmission; scheduling; classification; set coverage

1 引言

网络编码是由 Ahlswede 等人^[1]在 2000 年提出, 其编码融合思想突破传统网络中间节点仅存储转发信息的限制, 使网络吞吐量得到有效提升^[2]. 其中, 机会式网络编码 (Opportunistic Network Coding, ONC) 由于复杂度低和操作性强, 成为了该领域的研究热点.

在无线广播或多播中, 通过单次重传 ONC 包能恢复多个接收端的丢包, 减少重传次数^[3-9]. 其中, 文献[3~6]从编码策略的角度提出了基于 ONC 的广播重传算法, 使重传效率得到提升. 为寻找和利用更多编码机会, 文献[7, 8]分别根据最大匹配和低传输时延寻找编码包. 结合实际传输场景, 文献[9]在不同广播链路状态下, 采用加权方式选择权重较大的丢包结合成编码包.

但该方法仅能保证单次重传性能较优. 文献[10,11]考虑缓存解码的策略,进一步减少了重传次数,该策略要求前项编码包必须被正确接收,且数据包较多时,会消耗更多的缓存和解码时延. 为增加传输距离,并保证传输效率,中继协作传输被引入^[12],调度问题随之成为其关键问题之一. 在此基础上,文献[13,14]提出基于网络编码的中继协作无线多播方案(Network Coding Based Cooperative multicast scheme, NCBC),在重传阶段,优先让中继重传接收端丢失的包. 文献[15]改进 NCBC 的策略,提出在重传时先让源重传丢包的方案(Classification Opportunistic Network Coding Retransmission, CONCR),进一步提升了重传效率. 文献[16]采用最大权重构建了一种高效立即可解码传输方案. 随后,文献[17]提出一种基于包权重的选择调度重传算法(Selective Retransmission-Weighted Packet Scheduling, SRWPS),减少了传输完成时间,但在中继节点的重传成功率明显高于源节点的情况下,没有充分协调好两者之间的重传调度.

针对中继协作传输中编码包生成和重传调度等问题,本文在现有研究基础上,提出一种基于最小集合覆盖的分类网络编码重传方案(Minimal Set Coverage (MSC) based Classification Network Coding retransmission, MSC-CNC). 方案在重传阶段,将接收端的丢包分为:需要源节点重传的 S 类和可用中继代替源节点重传的 R 类. 先重传 S 类,根据该类丢包对应的状态反馈矩阵寻找编码机会生成编码包,并将编码包的选择过程抽象为集合覆盖问题,通过求 MSC 使所需编码数最少. 在不增加所得 S 类编码包数的基础上,充分利用两类丢包之间的编码机会,生成新编码包后重传,进一步减少重传次数,从而提高重传效率.

2 系统模型与相关定义

系统包含一个源节点 S , 一个中继节点 R 和 N 个接收端 $U = \{U_1, U_2, \dots, U_N\}$, 如图 1 所示. 在 R 的协助下 S 将 M 个信息包 $SP = \{P_1, P_2, \dots, P_M\}$ 发送给 N 个接收端. 用 $p_{s,i}$, $p_{r,i}$ 和 $p_{s,r}$ 分别表示 S 到接收端 U_i , R 到接收端 U_i ($i \in \{1, 2, \dots, N\}$), S 到 R 的链路丢包率, 丢包率不随时间变化而改变且满足 $p_{s,i} > p_{r,i}$, $p_{s,i} > p_{s,r}$. 传输为半双工通信, 且每个时隙只传输一个数据包.

传输分初始传输和重传两个阶段. 初始阶段, S 依次广播 M 个原始数据包. 各接收端在收到数据包后, 将接收状态反馈给 S 和 R , R 的接收状态反馈给 S . 假定反馈具有可靠性. 重传阶段, 根据重传方案生成编码包以恢复丢包, 直到各接收端都完全接收数据包.

定义 1 数据包接收状态反馈矩阵 D^U 是由源节点根据各接收端的反馈信息形成的 $N \times M$ 矩阵, N 表示接

收端数, M 表示原始数据包数. 如图 2 所示, 矩阵元素 $D_{i,j}^U = \{0, 1\}$, 其中, “1” 表示接收端 U_i 丢失数据包 P_j , “0” 表示 U_i 成功接收 P_j .

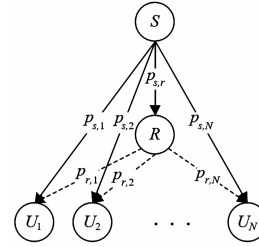


图1 系统模型

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9
U_1	0	0	0	1	1	1	1	0	0
U_2	0	0	0	0	0	0	0	1	1
U_3	1	1	0	0	1	0	1	0	1
U_4	0	0	1	0	0	0	1	0	1
U_5	1	0	1	1	0	0	0	0	1
U_6	0	0	0	0	1	0	0	0	0

图2 状态反馈矩阵实例

定义 2 丢失数据包 P_1, P_2, \dots, P_n ($n \leq M$) 在 D^U 中满足 $\sum_{j=1}^n D_{i,j}^U \leq 1$ ($i = 1, 2, \dots, N$) 的条件, 可生成 ONC 包 $e = P_1 \oplus P_2 \oplus \dots \oplus P_n$, 该编码包在接收端能够立即解码.

定义 3 最小集合覆盖: Q 是一个集合, Q_1, Q_2, \dots, Q_m 是 Q 的非空子集, 且这些子集构成对 Q 的覆盖, 即 $\cup_{k=1}^m Q_k = Q$, 求最小覆盖^[18].

定义 4 用 $G_i = \{a, b\}$ 和 $L_i = \{c, d\}$ 分别表示 U_i 成功接收数据包 P_a 和 P_b , 丢失 P_c 和 P_d , 则所有接收端的丢包集 $L = \{\cup_{i \in \{1, 2, \dots, N\}} L_i\}$, U 和 R 都丢失的包须通过 S 重传, 称 S 类丢包 $S^l = (\cup_{i \in \{1, 2, \dots, N\}} L_i) \cap L_R$; U 丢失而 R 成功接收的包可通过 R 重传, 称为 R 类丢包 $R^l = (\cup_{i \in \{1, 2, \dots, N\}} L_i) \cap G_R$.

3 MSC-CNC 方案

本方案涉及编码包的生成与选择, 以及编码包的传输调度问题. 其核心思想是先对 S 类丢包对应的分布矩阵使用 MSC 方法获取编码包后, 加入可与之编码的 R 类丢包, 得到新的编码包并重传, 直到 $S^l = \emptyset$; 再对剩余的 R 类丢包使用 MSC 方法, 重传所得编码包, 直到所有接收端成功接收所有数据包.

3.1 编码包的生成与选择

将所有接收端丢包构成的集合 L 与所有可编码在一起的丢包组合分别对应为定义 3 中的集合 Q 与 Q 的子集, 从而将编码包选择过程转化为集合覆盖问题. 为避免编码冗余, 要求所求得编码组合中每个丢包只出现一次. 具体步骤如下:

步骤 1 根据反馈信息形成 D^U . 若 D^U 为零矩阵, 表示无数据包丢失, 结束传输; 否则, 去除 D^U 的全零列, 转入步骤 2.

步骤 2 遍历 D^U , 根据定义 2 获取丢包 P_j ($j \in \{1, 2, \dots, M\}$) 对应的后续可编码包集 W_j . W_j 中的元素可单独与 P_j 满足定义 2, 内部元素之间不一定满足.

步骤 3 根据 P_j 与 W_j 寻找以 P_j 为首的可编码组

合 e_j . 再由 $e_j (j \in \{1, 2, \dots, M\})$ 构成所有丢包可能的编码组合集 l . 寻找 e_j 的伪代码如算法 1 所示.

算法 1 寻找 e_j 的伪代码

```

输入:  $P_j, W_j$ 
输出:  $e_j$ 
 $e = \{j\}; e_j = \emptyset;$ 
while  $W_j // W_j$  不为空
     $B = \{e, W_j(1)\}; // W_j(1)$  为  $W_j$  的第一个元素
     $flag = \text{zeros}(1, \text{length}(W_j) - 1) // flag$  为  $1 \times \text{length}(W_j)$  全 0 数组
    for  $k = 1 : \text{length}(W_j) - 1$ 
        if  $W_j(k+1)$  与  $B$  中的所有元素都满足定义 2
             $flag(k) = 1;$ 
        end
    end
    if  $\text{isequal}(flag, \text{zeros}(1, \text{length}(W_j) - 1))$ 
         $e_j = \{e_j; B\}; //$  获得一个编码组合  $B$ 
    else if  $\text{isequal}(flag, \text{ones}(1, \text{length}(W_j) - 1)) // flag$  为全 1 数组
         $e = B;$ 
         $B = \{e, W_j(2)\};$ 
    else
        若  $flag(i) = 1$ , 则  $e_j = \{e_j; B, W_j(i+1)\}; // i \in \{1, 2, \dots,$ 
         $\text{length}(flag)\}$ 
    end
     $W_j = W_j - W_j(1); //$  去除  $W_j(1)$ 
end

```

步骤 4 根据所得的 l 求 MSC, 得出重传编码包集合.

求 MSC 具体操作如图 3 所示. 执行时, 每次选择一个编码组合后, 在剩余编码组合中去除已选择的丢包元素, 即更新 Q 与 Q 的所有子集. $A(P_x)$ 是 Q 的某些子集的序号构成的集合, 这些子集都包含 P_x . 自定义启发函数

$$H(Q_i) = k_1 \cdot \sum_{P_x \in Q_i} |A(P_x)| + k_2 \cdot \text{dec}(Q_i) \quad (1)$$

其中, Q_i 是 l 的一个元素; $|A(P_x)|$ 为集合 $A(P_x)$ 的元素个数, 用以表示 P_x 与其他编码组合的关联度; $\text{dec}(Q_i)$ 表示能从编码组合 Q_i 获得解码恢复的接收端数, 即 D^U 中该编码组合对应列所有“1”的个数; 常数 k_1 和 k_2 满足 $k_1 + k_2 = 1$.

通过上述 4 个步骤所得编码组合为传输编码包. 该方法意在从全局传输的角度最小化重传次数. 因此, 在四个完备策略^[18]都不满足时, 执行启发函数, 优先选择 H 较小的编码组合, 旨在获取能在更多接收端恢复丢包且对全局影响最小的编码组合, 以便关键包通过完备策略选出, 从而更接近集合覆盖的最优解. 在特殊情况下, 会存在 $H(Q_i) = H(Q_j) (i \neq j)$, 且都为最小值, 选择靠前的编码组合, 满足先丢失的包先重传.

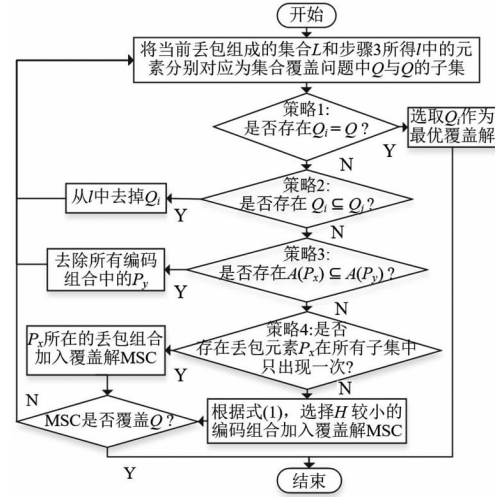


图 3 求 MSC 流程图

3.2 编码包的传输调度

由于 R 能协助传输, 且 $p_{s,i} > p_{r,i}$, 将丢包分为 S 类和 R 类来考虑重传. 在采用 MSC 方法获取编码包的基础上, 通过协调传输, 进一步挖掘两类丢包之间的编码机会, 生成新编码包. 具体步骤如下:

步骤 1 根据 S 类丢包状态反馈矩阵 $D^U (S^L)$, 采用 MSC 的方法得到 S 类编码组合.

步骤 2 向 S 类编码组合中加入可与其满足定义 2 的 R 类丢包. 每次加入一个 R 类丢包, 且一个 R 类丢包不重复添加在不同的编码组合中. 生成新编码包并传输.

步骤 3 接收端和 R 在收到 S 的重传编码包后进行反馈, 更新 D^U, G_R, L_R . 判断 $S^L = \emptyset$ 是否成立. 若成立, 执行步骤 4, 否则, 返回步骤 1.

步骤 4 对剩下的 R 类丢包反馈矩阵 $D^U (R_L)$ 采用 MSC 方法获取重传编码组合. 若所得编码组合中, 存在某个编码组合包含步骤 3 中 R 的不可解码部分, 则直接生成成本步骤中所得编码包并重传. 若存在某个编码包与 R 的不可解码包有部分相同元素, 则先重传 R 缓存的不可解码包.

步骤 5 接收端接收编码包并进行译码和反馈, 更新 D^U . 判断 $L = \emptyset$ 是否成立, 若成立, 则结束传输; 否则, 返回步骤 4.

其中, 步骤 2 的目的是在 S 类编码包中尽量加入可编码的 R 类丢包, 使单个编码包能恢复更多的丢包. 在处理 R 类丢包时, 部分丢包已在 S 类重传时恢复, 就使 R 类丢包重传次数减少. 步骤 3 中, 接收端对收到的编码包可直接解码, R 尽可能解码 (一个编码包包含的 S 类丢包数大于等于 2 时, R 不能完全解码), 保留不可解码部分. R 只要成功接收编码包, 不可解码也视为已经覆盖编码包所包含的原始包, 因为 R 直接重传不能立

即解码的编码包给接收端,也能被接收端立即解码.

MSC-CNC 的主要伪代码如算法 2 所示.

算法 2 MSC-CNC 方案的主要伪代码

```

输入:  $D^U, G_R, L_R$ 
输出:  $D^U$  // 全零矩阵
 $S^L = (\cup_{i \in \{1, 2, \dots, N\}} L_i) \cap L_R$ ;
 $R^L = (\cup_{i \in \{1, 2, \dots, N\}} L_i) \cap G_R$ ; //  $R$  类丢包
while  $S^L$ 
   $e^S = \text{function\_MSC}(D^U(S^L))$ ; //  $S$  类丢包使用 MSC 方法
  for  $i = 1$  : length( $e^S$ ) //  $e^S$  的元素是编码包
    for  $j = 1$  : length( $R^L$ )
      if  $R^L(j)$  与  $e^S(i)$  中的各有元素均可编码
         $e^S(i) = [e^S(i), R^L(j)]$ ; // 将  $R^L(j)$  加入编码包  $e^S(i)$ 
         $R^L(j) = \emptyset$ ;
      end
    end
  end
   $S$  发送  $e^S$  中的各编码包给  $U$  和  $R$ ;
   $U$  和  $R$  进行反馈;
  if  $R$  不能完全解码  $e^S(i)$  // 若接收到,  $R$  能否解码都作出覆盖反馈解码可解部分, 存储不可解码部分到  $R$  的 buffer;
    end
  更新  $D^U, S^L, R^L$ ;
end
while  $R^L$ 
   $e^R = \text{function\_MSC}(D^U(R^L))$ ;
  for  $k = 1$  : length( $e^R$ )
    if buffer( $i$ )  $\subseteq e^R(k)$  且  $e^R(k)$  中其他元素在  $R$  处都单独存在
      生成编码包  $e^R(k)$ ;
    else if buffer( $i$ )  $\cap e^R(k) \neq \emptyset$  且 buffer( $i$ ) 与  $e^R(k)$  互不包含
      先重传 buffer( $i$ );
    end
  end
  更新  $D^U, R^L$ ;
end

```

4 性能分析

4.1 重传效率

用平均每次重传所获得的丢包恢复数表示重传效率 η .

$$\eta = \frac{|L|}{C} \quad (2)$$

$|L|$ 为初始传输后, 接收端的所有丢包数; C 为所有丢包被成功接收所需的重传次数. 采用概率分析, 先考虑 $N = 2$ ($U = \{U_1, U_2\}$) 的情况, 再推广到多接收端. 设 $p_{s,1} \geq p_{s,2}$, 初始传输后, U_1 和 U_2 分别与 R 都丢失的包期望数及其数量关系为 $Mp_{s,1}p_{s,r} \geq Mp_{s,2}p_{s,r}$. S 类丢包重传时, 由于 U_1 的丢包数多, 所有重传包对 U_1 都是有效的, 部

分重传包对于 U_2 是无效的. 当出现 U_2 的无效包时, 加入可编码的 R 类丢包, 不会增加 S 类编码包数, 故该阶段的重传次数由 U_1 的情况决定. 用随机变量 T 表示传输事件, 一个此类包经过 t 次传输后, 被 R 或 U_1 成功接收的概率为

$$p\{T=t\} = p_{s,1}^{t-1}(1-p_{s,1})p_{s,r}^t + p_{s,1}^t p_{s,r}^{t-1}(1-p_{s,r}) + p_{s,1}^{t-1}(1-p_{s,1})p_{s,r}^{t-1}(1-p_{s,r}) = p_{s,1}^{t-1} p_{s,r}^{t-1} (1-p_{s,1}p_{s,r}) \quad (3)$$

则被 R 或 U_1 成功接收需要的期望重传次数为

$$E(T) = \sum_{t=1}^{\infty} t \cdot p\{T_1 = t\} = \frac{1}{1-p_{s,1}p_{s,r}} \quad (4)$$

故 S 类丢包所需的期望重传次数为

$$E(T_s) = \frac{Mp_{s,1}p_{s,r}}{1-p_{s,1}p_{s,r}} \quad (5)$$

对于 R 类丢包, 要确定其期望重传次数, 须先知道在 S 类丢包传输完成时接收端期望丢包数 $E(|L_1|_s)$ 和 $E(|L_2|_s)$. 在开始重传时 $E(|L_1|_0) = Mp_{s,1}$, $E(|L_2|_0) = Mp_{s,2}$, 还需确定传输过程中多少有效包被各端接收. 由式(3)可得出一个包被 U_1 接收的概率为

$$p(U_1 | (R \text{ or } U_1)) = \frac{1-p_{s,1}}{1-p_{s,1}p_{s,r}} \quad (6)$$

由于 S 类传输中会添加可编码的 R 类丢包, 最终 S 类丢包传输完成时 U_1 的期望丢包数为

$$E(|L_1|_s) \leq Mp_{s,1} - \frac{(Mp_{s,1}p_{s,r})(1-p_{s,1})}{1-p_{s,1}p_{s,r}} \quad (7)$$

剩下的 R 类丢包包括 U_1 或 U_2 单独丢失以及两者都丢失的包. 因为一个接收端在一个编码包中只能恢复一个丢包, 则两个接收端单独丢失的包可编码, 而都丢失的只能单独重传. 由式(7)和 $p_{s,1}$ 可得 U_1 处理剩余丢包的期望重传次数为

$$E(T_{R,1}) = \frac{E(|L_1|_s)}{1-p_{r,1}} \leq \frac{Mp_{s,1}(1-p_{s,r})}{(1-p_{r,1})(1-p_{s,1}p_{s,r})} \quad (8)$$

同理可得

$$E(T_{R,2}) = \frac{E(|L_2|_s)}{1-p_{r,2}} \leq \frac{Mp_{s,2}(1-p_{s,r})}{(1-p_{r,2})(1-p_{s,2}p_{s,r})} \quad (9)$$

那么, R 类丢包所需的期望重传次数为

$$E(T_R) = \max\{E(T_{R,1}), E(T_{R,2})\} \leq \max\left\{\frac{Mp_{s,1}(1-p_{s,r})}{(1-p_{r,1})(1-p_{s,1}p_{s,r})}, \frac{Mp_{s,2}(1-p_{s,r})}{(1-p_{r,2})(1-p_{s,2}p_{s,r})}\right\} \quad (10)$$

综上所述, 当 $N = 2$ 时的重传效率

$$\eta_2 = \frac{E(|Z = \{\cup_{i \in \{1, 2\}} L_i\}|)}{E(T_s) + E(T_R)} \geq \frac{M(p_{s,1} + p_{s,2} - p_{s,1}p_{s,2})}{Mp_{s,1}p_{s,r} + \max\{E(T_{R,1}), E(T_{R,2})\}} \quad (11)$$

设有 N 个接收端, 且 $p_{s,k} = \max_{i \in \{1,2,\dots,N\}} \{p_{s,i}\}$, 若每个编码包对 U_k 都有效, 则重传 S 类丢包所需的理论最小传输次数为

$$E(T_S)_{N,\min} = \frac{Mp_{s,k}p_{s,r}}{(1-p_{s,k}p_{s,r})} \quad (12)$$

而当 R 类丢包有所减少后, 还需的重传次数为

$$E(T_R)_N \leq \max_{i \in \{1,2,\dots,N\}} \{E(T_{R,i})\} \quad (13)$$

由此可知, 随着 $E(T_S)_N$ 增加, 若存在 R 类丢包与 S 类编码包满足定义 2, $E(T_R)_N$ 就可能会减少. 具体情况需根据具体的状态反馈矩阵确定. 由式 (12) 和式 (13) 可推出重传效率 η_N . 由于编码可结合的最大丢包数为 N , 当 $M < N$ 时, M 明显能对重传效率产生影响, 加上式 (14) 中 $E(T_S)_N$ 和 $E(T_R)_N$ 并未最终确定, 故 M 不能直接消去.

$$\begin{aligned} \eta_N &= \frac{E(|L| = \{ \cup_{i \in \{1,2,\dots,N\}} L_i \} |)}{E(T_S)_N + E(T_R)_N} \\ &\geq \frac{M(1 - \prod_{i=1}^N (1 - p_{s,i}))}{\frac{Mp_{s,k}p_{s,r}}{1-p_{s,k}p_{s,r}} + \max_{i \in \{1,2,\dots,N\}} \left\{ \frac{Mp_{s,i}(1-p_{s,r})}{(1-p_{r,i})(1-p_{s,i}p_{s,r})} \right\}} \end{aligned} \quad (14)$$

4.2 算法复杂度

MSC-CNC 方案在编码包生成阶段, 根据 $N \times M$ 的状态矩阵确定两两丢包之间的可编码性, 丢包数最多为 M , 则算法时间复杂度为 $O(NM^2)$; 若有 K 个可编码包, 即 MSC 问题中的 K 个子集. 由于单个编码包中的丢包数不超过 N , 用 MSC 方法获取最终编码包时, 策略 1 的时间复杂度为 $O(NK)$; 策略 2 寻找可编码包之间的包含关系, 时间复杂度为 $O(NK^2)$; 策略 3 中计算各丢包属于哪些可编码包, 时间复杂度为 $O(MNK)$; 策略 4 在可编码包中寻找仅出现一次的丢包, 时间复杂度为 $O(MNK)$. 启发函数可基于策略 4 的结果, 并计算各可编码包中的丢包数, 时间复杂度为 $O(K)$. 当所有可编码包均包含两个丢包且不能形成更大维度的可编码包时, K 为最大值 $2(M-2)$; 此外, 最终获得的编码包数不超过 M , 即算法过程最多执行 M 次, 因此, 编码包生成与选择的时间复杂度为 $O(NM^3)$.

MSC-CNC 在生成编码包时进行了分类编码, 原来 $N \times M$ 的矩阵被拆分成 S 类和 R 类对应的两个部分. 就算法整体而言, 编码包生成与选择过程的计算维度被缩小. 因此, 实际时间复杂度是完全小于 $O(NM^3)$ 的. NCBC 和 SR-WPS 方案, 其编码原理与 MSC-CNC 相同, 故寻找编码包的复杂度都为 $O(NM^2)$. 在编码包选择时, NCBC 所给算法单次执行时间复杂度为 $O(NM^2)$ ^[14], 也采用中继和源的分类传输, 最多执行 M 次, 最终复杂度也不超过 $O(NM^3)$; CONCR 方案仅对 NCBC 的传输策略做了调整, 其复杂度与 NCBC 相同;

而 SR-WPS 每次都要通过计算源和中继节点的编码包权重值以决定传输, 且初始反馈矩阵为 $N \times M$, 时间复杂度为 $O(NM^3)$.

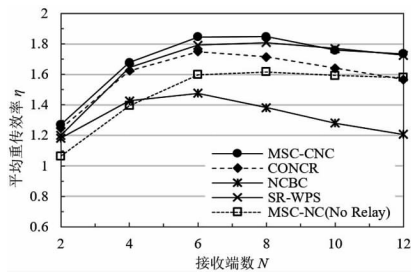
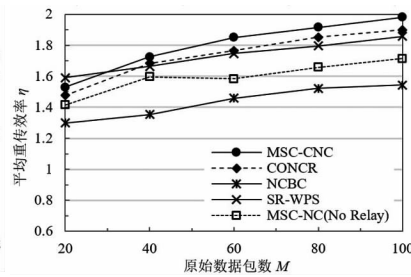
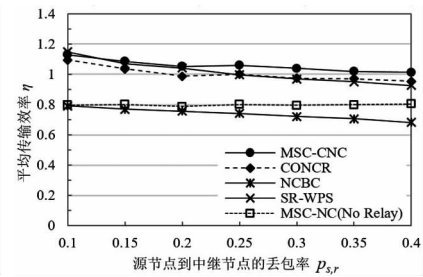
5 仿真实验

本实验将 MSC-CNC 与 NCBC、SR-WPS、MSC-NC、CONCR 方案进行仿真对比与分析. MSC-NC 是在无中继协作情况下, 使用 MSC 方法的方案.

为避免偶然性差异, 采用多次实验求平均值得出实验结果. 设 $k_1 = 0.5$, 图 4 给出了各方案的平均重传效率随 N 变化的情况. 系统参数为 $M = 60, p_{s,r} = 0.2, p_{s,i} = 0.3, p_{r,i} = 0.1 (i \in \{1,2,\dots,N\})$, 仿真结果表明: 与其他中继协作传输方案相比, 在 $N \leq 8$ 的情况下, MSC-CNC 的重传效率不断增加并且最优. 但随着 N 进一步增加性能会有所下降, 是因为 N 不断增加使得数据包随机丢失的空间扩大, 丢失的数据包数就会有所增加, 其他条件未变, R 的接收数据包数相对稳定, 导致 S 类丢包增加, 而 S 到接收端传输成功率不高, 故传输效率下降. SR-WPS 方案在丢包数增加的情况下, 性能也会有所下降, 但其根据编码包权重的选择调度方式受到的影响不明显, 因此, 下降的趋势较缓.

如图 5 为各方案的平均重传效率随 M 变化的对比情况. 其中, $N = 6, p_{s,r} = 0.2, p_{s,i} = 0.3, p_{r,i} = 0.1 (i \in \{1,2,\dots,N\})$. 仿真结果表明, 随着 M 的增加, MSC-CNC 算法仍然具有最优重传效率. 由实验一可知, $N = 6$ 时, MSC-CNC 方案具有明显优势, 在此前提下, 增加 S 发送的原始数据包数, R 的接收数据包数会增加, 由于 R 到接收端的传输可靠性更高, 使得传输的优势增加. SR-WPS 方案性能虽有所增加, 但与 MSC-CNC 方案的差距不断扩大, 主要原因是: 随着 M 的增加, 结合数较大的编码包数量也会有所增加, 传输效率获得提高; SR-WPS 方案通过 S 生成编码包的过程中考虑了每个编码包在 R 处必须可解码, 增加了编解码条件, 导致编码机会减少, 所需重传编码包数增加, 传输效率受到影响. 实际上 R 并不一定要完全接收和解码 S 发送的所有数据包, 只要能协助 S 实现接收端完全接收和解码即可. 而 CONCR 方案在重传阶段虽然也采用分类编码重传, 但没有考虑两类丢包之间的编码机会, 故传输效率要比 MSC-CNC 低.

系统参数为 $M = 60, N = 6, p_{s,i} = 0.5, p_{r,i} = 0.2 (i \in \{1,2,\dots,N\})$ 时, 平均传输效率随 $p_{s,r}$ 变化的情况如图 6 所示. 其结果表明随着 $p_{s,r}$ 的增加, MSC-CNC 算法的性能有所下降, 但总体上仍是最优. R 在初始传输阶段接收的数据包会不断减少, 从而导致存在中继节点协作传输的方案中, R 的作用减小, 数据包重传次数增加, 重传效率下降, 而对不存在中继节点的 MSC-NC 方案并无明显影响.

图4 η 随 N 的变化图5 η 随 M 的变化图6 η 随 $p_{s,r}$ 的变化

6 结论

本文通过对单中继协作传输的分析与研究,提出一种基于MSC的分类ONC重传算法.算法充分利用中继协作传输的优势,将接收端丢包根据中继节点的接收状态进行分类,一方面,在各类丢包内部寻找编码机会并生成编码包,再采用MSC方法得出所需的编码包,从而最小化重传次数;另一方面,通过调度源节点与中继节点的重传,挖掘两类丢包之间的编码机会,生成新编码包,进一步减少重传编码包数.通过对所提算法进行分析和仿真,结果表明本算法的重传效率在相应条件下具有明显的优势.

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] Li Z, Li B, Lau L C. A constant bound on throughput improvement of multicast network coding in undirected networks[J]. IEEE Transactions on Information Theory, 2009, 55(3): 1016 - 1026.
- [3] Rozner E, Iyer A P, Mehta Y, et al. ER: efficient retransmission scheme for wireless LANs[A]. Proceedings of the 2007 ACM CoNEXT Conference[C]. New York: ACM, 2007. 8.
- [4] Nguyen D, Tran T, Nguyen T, et al. Wireless broadcast using network coding[J]. IEEE Transactions on Vehicular Technology, 2009, 58(2): 914 - 925.
- [5] Xiao X, Wang W P, Yang L M, et al. Wireless broadcasting retransmission approach based on network coding[J]. Journal on Communications, 2009, 30(9): 782 - 786.
- [6] 王斌, 王文鼎. 一种基于并行网络编码的无线广播重传方案[J]. 南京邮电大学学报: 自然科学版, 2014, 34(6): 9 - 14.
Wang B, Wang W D. Wireless broadcast retransmission scheme based on parallel network coding[J]. Journal of Nanjing University of Posts and Telecommunications, 2014, 34(6): 9 - 14. (in Chinese)
- [7] 戴彬, 曹志刚, 杨军, 等. 基于匹配理论的无线广播编码重传算法[J]. 软件学报, 2011, 22(11): 2833 - 2842.
Dai B, Cao Z G, Yang J, et al. Coded retransmission algorithm based on matching theory for wireless broadcasting[J]. Journal of Software, 2011, 22(11): 2833 - 2842. (in Chinese)
- [8] 卢冀, 肖嵩, 吴成柯. 基于机会式网络编码的低时延广播传输算法[J]. 电子学报, 2011, 39(5): 1214 - 1219.
Lu J, Xiao S, Wu C K. Opportunistic network coding based delay-sensitive broadcast transmission algorithm[J]. Acta Electronica Sinica, 2011, 39(5): 1214 - 1219. (in Chinese)
- [9] 苟亮, 张更新, 孙伟, 等. 无线网络中基于机会网络编码的加权广播重传[J]. 电子与信息学报, 2014, 36(3): 749 - 753.
Gou L, Zhang G X, Sun W, et al. Weighted broadcasting retransmission based on opportunistic network coding in wireless networks[J]. Journal of Electronics & Information Technology, 2014, 36(3): 749 - 753. (in Chinese)
- [10] 卢冀, 吴成柯, 肖嵩, 等. 基于机会式网络编码的高效广播传输算法[J]. 通信学报, 2012, 33(1): 64 - 70.
Lu J, Wu C K, Xiao S, et al. Efficient broadcast transmission algorithms based on opportunistic network coding[J]. Journal of China Institute of Communications, 2012, 33(1): 64 - 70. (in Chinese)
- [11] Yao Y K, Chen X, Ren Z, et al. Efficient network coding broadcasting retransmission approach based on redundancy avoiding[J]. Systems Engineering & Electronics, 2015, 37(5): 1170 - 1176.
- [12] Zhao B, Valenti M C. Practical relay networks: a generalization of hybrid-ARQ[J]. IEEE Journal on Selected Areas in Communications, 2005, 23(1): 7 - 18.
- [13] Zhi C, Wei C, Fan P, et al. Relay aided wireless multicast utilizing network coding: outage behaviour and diversity gain[A]. IFIP International Conference on Network and Parallel Computing[C]. Shanghai: IFIP, 2008. 358 - 364.
- [14] Fan P, Zhi C, Wei C, et al. Reliable relay assisted wireless multicast using network coding[J]. IEEE Journal on Selected Areas in Communications, 2009, 27(5): 749 - 762.

- [15] Lu L, Xiao M, Rasmussen L K. Design and analysis of relay-aided broadcast using binary network codes[J]. Journal of Communications, 2011, 6(8): 610 – 617.
- [16] Elmahdy A M, Sorour S, Seddik K G. Generalized instantly decodable network coding for relay-assisted networks [A]. Proceedings of the 2013 IEEE 24th International Symposium on Personal Indoor and Mobile Radio Communications [C]. London: IEEE, 2014. 1759 – 1763.
- [17] Gou L, Zhang G, Bian Z, et al. Minimizing completion time for relay-assisted multicast with instantly decodable network coding[J]. IEEE Communications Letters, 2016, 20(3): 434 – 437.
- [18] Guangri Q, Bingrong H, Fei Y, et al. A heuristic function algorithm for minimum set-covering problem[J]. Journal of Software, 1998, 9(2): 156 – 160.

作者简介



王 练 女, 1976 年生于贵州, 博士, 副教授. 现主要从事无线网络编码、网络安全研究.

E-mail: wanglian@cqupt.edu.cn



梁申虎 男, 1989 年生于四川宜宾, 硕士, 现主要从事无线网络编码研究.

E-mail: liang5284569@163.com