

无线网络编码增益感知的低时延路由协议

田贤忠, 朱艺华, 缪得志

(浙江工业大学计算机科学与技术学院, 浙江杭州 310023)

摘 要: 降低无线网络数据包传递时延对实时应用有重要的意义. 该文提出了一种编码增益的计算方法和编码图的简化方法, 并基于此提出了编码增益感知的路由协议 CGAR(Coding Gain Aware Routing). CGAR 利用网络编码增益和无线链路的期望传输次数 ETX(Expected Transmission Count)计算传递一个来自新流的数据包所需要的时间, 并以此作为选路度量, 旨在降低数据传递时延. 仿真试验表明, CGAR 的时延优于 COPE 和 DCAR 协议.

关键词: 无线网络; 网络编码; 时延; 路由

中图分类号: TP393 **文献标识码:** A **文章编号:** 0372-2112 (2013)04-0652-07

电子学报 URL: <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2013.04.005

Wireless Network Coding Gain Aware Routing Protocol with Low Delay

TIAN Xian-zhong, ZHU Yi-hua, MIAO De-zhi

(School of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, Zhejiang 310023, China)

Abstract: It is significant to reduce packet delivery delay for real-time applications in a wireless network. The algorithm of computing networking coding gain(NCG)and the method of simplifying coding graph are presented, which are used in the proposed coding gain aware routing(CGAR)protocol. The NCG and expected transmission count(ETX)of wireless links are used in CGAR to compute the time of transmitting a packet coming from a new flow, which is used as a routing metric in choosing a route so as to reduce packet delivery delay. Simulation results show that the proposed CGAR outperforms the DCAR and COPE protocols in terms of delay.

Key words: wireless networks; network coding; delay; routing

1 引言

数据包传递时延是网络性能的重要指标之一, 对实时应用尤为重要. 在无线网络中, 时延与节点发送数据包时可用的信道容量有关. 一般来说, 一个节点在发送数据包时只能使用已有数据流所占用的信道之外的剩余信道容量. 事实上, 在有限域中进行线性网络编码(Network coding), 可以将多个数据包编码成一个新数据包(简称“编码包”), 而且编码包所包含的字节数不超过参与编码的多个原数据包的最大者. 因此, 在新的数据流与已有数据流可以进行线性编码的条件下, 新流可以与已有流编码在一起并在已有信道中被捎带走, 而不占用额外的信道容量. 这样, 可以提高带宽的使用率、增大吞吐率, 同时可以降低因排队等待空闲信道所带来的时延.

正是出于上述考虑, 本文研究低时延的基于网络编

码的路由协议, 主要贡献与创新之处在于: (1)给出了编码增益的贪婪算法; (2)给出了编码图的简化方法; (3)提出了编码增益感知的路由协议 CGAR, 这种协议以数据包传输时间作为路由选择度量, 充分考虑了网络编码带来的增益, 同时结合了链路的带宽、丢包率对数据传输时延的影响, 以达到降低时延的目的.

2 相关工作

网络编码是由 Ahlswede 等人^[1]提出的, 可有效提高网络的吞吐量和可靠性. 目前, 一些学者把网络编码用于路由协议, 取得了一些研究成果. Katti^[2]等人提出了基于机会的网络编码方法(COPE)并加以实现; Le 等人^[3]和 Guo 等人^[4]提出了多跳网络中节点的编码条件, 并以此找出信源和信宿节点之间的潜在路径和潜在网络编码机会; Yan 等人^[5]提出了速率匹配的编码感知多路径路由协议(RCR), 考虑了不同速率流之间的网络编

码问题;Lhang 等人^[6]综合考虑了网络的吞吐量和编码机会的平衡问题,提出了无线 Ad Hoc 网络中编码和容量感知路由协议(C^2AR);卢文伟等人^[7]提出了利用线性网络编码的节能路由算法.卢冀等人^[8]提出了基于机会式网络编码的低时延广播传输算法,而汪玉等人^[9]则分析了流间网络编码中的侦听管理策略.

传统路由协议一般以最短路径、最少跳数、最少期望传输次数(Expected Transmission Count, ETX)^[10]、最小期望传输时间(Expected Transmission Time, ETT)^[11]等作为选路度量.将网络编码应用到路由算法之后,一些新的选路度量被提出.例如, Ni 等人^[12]从网络编码对提升网络吞吐量的角度定义了期望编码传输次数(the Expected Number of Coded Transmissions, ECX)路由选择度量;Wu 等人^[13]提出了基于马尔可夫性的路由度量;Lhang 等人^[6]提出了 EPT(Estimated Path Throughput)度量,以平衡吞吐量和编码机会;Guo 等人^[4]提出的 FORM(Free-ride Oriented Routing Metric)度量致力于最大限度利用编码机会.

与本文直接相关的是文献[3]提出的 CRM(Coding-aware Routing Metric)度量,它考虑了多条流的编码情况,并用编码图来表示节点中流的编码情况.本文利用网络编码增益和无线链路的期望传输次数,给出了节点在无线链路上传递一个来自新流的数据包所需要的时间,并以此作为选路度量.这个度量考虑了传播时延、节点竞争信道时间等因素.本文所给出的路由算法可以降低数据包的传递时延.

3 编码条件及编码图

本文把从源节点 v_0 到终节点 v_t 传递数据包的路径 $F = v_0 \rightarrow v_1 \rightarrow \dots \rightarrow v_{i-1} \rightarrow v_i \rightarrow v_{i+1} \rightarrow \dots \rightarrow v_t$ 称为从节点 v_0 到节点 v_t 的一条“信息流”(简称为“流”),其中, $v_0, v_1, \dots, v_{i-1}, v_i, v_{i+1}, \dots, v_t$ 表示网络的节点.而且,集合 $U(i, F) \equiv \{v_0, v_1, \dots, v_{i-1}\}$ 和 $D(i, F) \equiv \{v_{i+1}, v_{i+2}, \dots, v_t\}$ 分别称为节点 i 的上游节点集合和下游节点集合.此外,以 $N(i)$ 表示节点 i 的邻居节点集合.

文献[3]给出了寻找编码节点的方法,证明了以下结论(下称“编码条件”).如果两条流 F_1 和 F_2 交汇于节点 i (即节点 i 是流 F_1 和 F_2 的公共节点),则流 F_1 和 F_2 可以在节点 i 进行编码,并且编码包可以被下游节点解码的充分必要条件^[3]:

- ①存在 $v_1 \in D(i, F_1)$, 满足: $v_1 \in N(v_2), v_2 \in U(i, F_2)$; 或者 $v_1 \in U(i, F_2)$;
- ②存在 $v_2 \in D(i, F_2)$, 满足: $v_2 \in N(v_1), v_1 \in U(i, F_1)$; 或者 $v_2 \in U(i, F_1)$.

与文献[3]一样,本文以编码图 $G(V, E)$ 表示经过

某个节点的流是否可以编码.其中, V 是顶点的集合,每个顶点表示进入该节点的一条流; E 是边的集合,每条边表示该边所连接的两个顶点所对应的两条流可进行编码.如图 1, 设 F_1, F_2, \dots, F_6 是进入节点 X 的流,且 F_1, F_2 和 F_3 两两可编码,流 F_4 和 F_5 也可编码,但流 F_6 与其它流不能编码,则节点 X 的编码图可以用图 2 表示,其中, $V = \{F_1, F_2, F_3, F_4, F_5, F_6\}$, $E = \{<F_1, F_2>, <F_1, F_3>, <F_2, F_3>, <F_4, F_5>\}$. 此处,符号 $<F_j, F_k>$ 是图 G 的一条边表示流 F_j 和 F_k 可编码.在编码图 G 中,所有不可编码的流是一个无边相连的孤立点,如图 2 中顶点 F_6 .

利用上述“编码条件”可以构建出各节点的编码图.在编码图中,如果一个图的某个子图满足:任意两个顶点均存在一条边将它们连接起来,则称这个子图为“完全子图”.构成完全子图的所有流可以同时被编码.如图 2, 流 F_1, F_2 和 F_3 构成编码图 G 的一个完全子图(每两个顶点均有一连线).

定理 1 如果节点 X 的编码图中存在一个完全子图,则 X 可以从这个完全子图所涉及的所有入流中各取一个数据包进行编码形成一个编码包,使得在每条流中存在节点 X 的下游节点,这个下游节点可以通过对这个编码包进行解码还原它所在流的信源所发送的数据包,进而使这条流的信宿获得信源所发送的数据包.

证明 当节点 X 有两条入流时,根据“编码条件”,如果两条流在节点 X 处可编码生成一个编码包,则在每条流上存在着 X 的下游节点,它可以利用旁听到的数据包对编码包进行解码获得它所在流的信源所发送的数据包,因此,定理 1 成立.

当节点 X 有三条入流时,如图 3, 设三条入流 F_1, F_2 和 F_3 在编码图中构成一个完全子图,即它们两两可编码.设节点 A_1, B_1, C_1 分别为流 F_1, F_2 和 F_3 的信源节点,发送的数据包分别为 P_A, P_B, P_C , 且节点 X 将这三个数据包编码生成一个编码包 $P_A \oplus P_B \oplus P_C$, 其中,

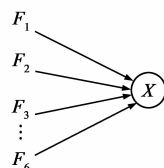


图1 流入节点X的信息流

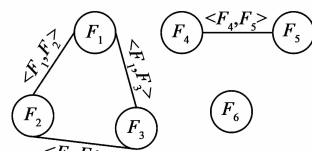


图2 节点X的编码图 $G(V, E)$

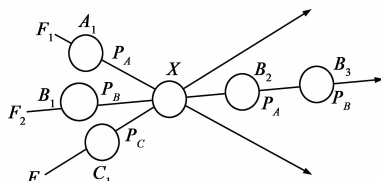


图3 三条可相互编码流共同生成编码包且可解码

\oplus 表示异或操作.因流 F_1 和 F_2 在节点 X 处可编码,由“编码条件”可知,在 F_2 上存在 X 的一个下游节点 B_2 ,它可利用所旁听到的来自流 F_1 的数据包 P_A 进行解码,得出: $P_A \oplus (P_A \oplus P_B \oplus P_C) = (P_B \oplus P_C)$. 然后节点 B_2 把 $(P_B \oplus P_C)$ 转发给下游节点. 此外,因流 F_3 和 F_2 在节点 X 处可编码,由“编码条件”可知,在 F_2 上存在 X 的一个下游节点 B_3 可以利用旁听到流 F_3 的数据包 P_C 对收到的数据包 $(P_B \oplus P_C)$ 进行解码得出: $P_C \oplus (P_B \oplus P_C) = P_B$. 此乃流 F_2 的信源发给信宿的数据包. 同理,流 F_1 和 F_2 信源所发送的数据包也可以被各自的信宿接收.

应用数学归纳法,可以把上述结论推广到任意条入流的情况(留给读者自行证明),因而定理 1 得证.

4 编码增益

以 $F_j(i)$ 表示在 ΔT 时间内进入节点 i 的流,其流量记为 $Q_j(i)$ (单位: bit), $j = 1, 2, \dots, m$. 此外,以 $IF(i)$ 和 $OF(i)$ 分别表示在 ΔT 内流入和流出节点 i 的总流量. 这样,

$$IF(i) = \sum_{k=1}^m Q_k(i) \quad (1)$$

一般来说,在采用网络编码的条件下,流出节点的流量要小于流入节点的流量. 因此,本文将节点 i 在单位时间内因编码而减少的流量称为“节点 i 的编码增益”,记为 $GF(i)$ (单位: bps). 考虑到在 ΔT 时间内,节点 i 所减小的流量为 $IF(i) - OF(i)$. 于是,编码增益可由下式得出:

$$GF(i) = \frac{IF(i) - OF(i)}{\Delta T} \quad (2)$$

由于有限域对线性运算是封闭的,因此,在有限域上将多个数据包进行线性编码生成的编码包所含的比特数不超过参与编码的最大数据包的比特数. 于是,参与编码的数据包越多,带宽越节省. 由定理 1 知,如果一个节点的编码图存在一个完全子图,则在这个完全子图所涉及的多条入流中各取一个数据包可以进行网络编码生成一个编码包,这可以大大增加带宽的利用率. 而且,如果完全子图所包含的节点数越多,编码效率就越高. 因此,在编码图中寻找最大完全子图(即所包含的顶点数最多的完全子图)对提高网络编码的效率是极为重要的. 为此,本文提出下述“贪婪编码增益算法”,这种算法寻找可生成一个编码包的流的最大条数,其步骤如下:

步骤 1 构建编码图 $G(V, E)$: 对于节点 i 的入流 $F_j(i)$, 在图 G 中用一个编号为 j 的顶点与它对应, $j = 1, 2, \dots, m$, 即顶点集合 $V = \{1, 2, \dots, m\}$; 根据上述“编码条件”确定图 G 的边.

步骤 2 采用下述定理 2 对编码图 G 进行精简.

步骤 3 计算 $IF(i) = \sum_{k=1}^m Q_k(i)$, 并令 $OF(i) = 0$.

步骤 4 如果 $V = \Phi$, 则转到步骤 8. 随机取一个顶点 $k \in V$, 找出包含顶点 k 的最大完全子图 $G_0(V_0, E_0)$, 其中, $V_0 \subseteq V, E_0 \subseteq E$.

步骤 5 取图 $G_0(V_0, E_0)$ 中最小流量: $f_{\min} = \min_{k \in V_0} Q_k(i)$, 且令 $OF(i) = OF(i) + f_{\min}$.

步骤 6 将 G_0 中各个顶点的流量减去 f_{\min} , 即 $Q_k(i) = Q_k(i) - f_{\min}, k \in V_0$.

步骤 7 移除编码图 G 中流量为 0 的顶点, 即 $V = V - \{k \mid Q_k(i) = 0\}$, 转到步骤 4.

步骤 8 利用式(2)计算编码增益 $GF(i)$.

步骤 9 算法终止.

下述定理用于简化编码图.

定理 2 在一个节点的编码图 G 中, 如果存在一个顶点, 它的流量不小于与之相连的一些顶点的流量之和, 那么把与它相连的这些顶点去掉之后所得的编码图 G_0 与原编码图 G 的编码增益相同.

由于在节点的编码图中一个顶点对应于这个节点的一条入流, 编码图的一条边表示两条流可以编码, 因此, 定理 2 等价于以下定理.

定理 3 在一个节点的 m 条入流 F_1, F_2, \dots, F_m 中, 如果 F_1 与其余 $m-1$ 条流 F_2, F_3, \dots, F_m 均可以编码, 而且 F_1 的流量不小于 F_2, F_3, \dots, F_m 的流量之和, 那么采用以下两种方案所得的编码增益相等:

方案 1 在入流中去掉其余 $m-1$ 条流 F_2, F_3, \dots, F_m .

方案 2 将 F_1 分别与 F_2, F_3, \dots, F_m 进行编码.

证明 考虑到方案 1 与方案 2 的入流相同, 均为 F_1, F_2, \dots, F_m , 因而入流的流量相同, 于是, 由式(2), 只须证明上述两种方案的出流流量相等即可. 设流 F_j 流量为 $Q_j, j = 1, 2, \dots, m$. 对于方案 1, 流 F_1 不与其它流编码, 因此节点的输出流量为 Q_1 .

对于方案 2, 因为 $Q_1 \geq Q_2 + Q_3 + \dots + Q_m$, 在 F_1 与 F_2 进行编码之后, F_1 的流量中可被流 F_2 编码带走的流量为 Q_2 , 这时节点输出流量为 Q_2 . 同理, F_1 与 F_j 编码之后被带走的流量为 $Q_j, j = 3, 4, \dots, m$. 因此, 流 F_1 与这 $m-1$ 条流编码所带走的总流量即节点采用方案 2 进行编码之后的总输出流量为 $Q = Q_2 + Q_3 + \dots + Q_m$. 编码之后流 F_1 的剩余流量为 $Q_R = Q_1 - Q$, 而且这部分剩余流量不能再进行编码, 因而节点将它直接输出, 输出流量为 Q_R . 这样, 节点的总输出流量为 $Q + Q_R = Q_1$. 于是, 方案 1 与方案 2 的输出流量相等, 定理 3 得证.

5 低时延路由协议

5.1 路由度量

采用网络编码之后,在通过节点 i 的一条新流中,一部分流量可与已有流进行编码而被捎带走,不占用额外的信道容量.这相当于在发送来自新流的数据包时,节点 i 可用的信道容量在原有空闲信道容量的基础上增加了编码增益 $\Delta GF(i)$.也就是说,采用网络编码之后,节点 i 发送来自新流的数据包时可用的信道容量为

$$C(i) = \Delta GF(i) + C_1(i) \quad (3)$$

其中, $C_1(i)$ 表示原有空闲信道容量,可以依文献[14]的方法获得.以下称 $C(i)$ 为节点 i 的“扩充信道容量”(单位:bps).

设 L 为节点发送的数据包长度(单位:bit).于是,在网络编码支持下,节点 i 在信道上传递来自新流的一个数据包所需的时间为 $L/C(i)$ (单位:s).以 $ETX(i)$ 表示节点 i 成功传递一个数据包给接收节点所需要的平均传输次数.这样,节点每传递一个来自新流的数据包所需要的平均时间为

$$\begin{aligned} \Phi(i) &= [(ETX(i) - 1)(\frac{L}{C(i)} + T_0)] \\ &+ [\frac{L}{C(i)} + \delta + T_{ACK} + \delta] + ETX(i) T_C \\ &= \frac{L}{C(i)} ETX(i) + T_0(ETX(i) - 1) \\ &+ T_{ACK} + 2\delta + ETX(i) T_C \end{aligned} \quad (4)$$

其中, T_0 为节点 i 对所发送的帧所设置的定时器值, δ 为传播时延, T_{ACK} 是 ACK 包的发送时延, T_C 是节点每次竞争信道所消耗的时间,其均值按文献[11]方法确定.在式(4)的第 1 行中,方括号项表示,节点 i 在前 $ETX(i) - 1$ 次发送数据包失败(需要在定时器超时之际重发数据包)所消耗的时间;第 2 行方括号项表示,节点 i 在第 $ETX(i)$ 次发送数据包成功收到确认包(即 ACK 包)所消耗的时间;第 2 行最后 1 项表示,节点 i 在 $ETX(i)$ 次竞争信道所消耗的总时间.

在本文的路由算法中,节点 i 在选路时以 $\Phi(i)$ 为路由度量进行.其中,参数 L 、 $ETX(i)$ 和 $C(i)$ 按下述方法计算.

(1) 计算 L .

在 IEEE 802.11 标准中,帧的类型分为三类:数据帧、管理帧和控制帧,其中,帧的最大长度为 2346 Byte(数据帧),帧的最小长度为 34 Byte(控制帧)^[15].不妨设帧的大小符合均匀分布,则其均值为 $(34 + 2346)/2 = 1190$ (Byte).

(2) 计算 $ETX(i)$.

$ETX(i)$ 为节点 i 成功发送一个数据包的平均传输

次数,它与节点传输数据的丢包率有关. De Couto 等人在文献[10]中用广播探测包的方法,在探测包的数据长度为 L_0 字节时,获得节点传输数据的丢包率的近似值如下:

$$p = 1 - (1 - p_f)(1 - p_r) \quad (5)$$

其中 p_f 和 p_r 分别是正向发送探测包和反向发送确认包的丢包率.因此,对于长度为 L 字节的数据包,它可分成 L/L_0 个长度为 L_0 的数据包.于是,传输一个长度为 L 字节数据包的成功概率为 $(1 - p)^{L/L_0}$,即丢包率为

$$p_L = 1 - (1 - p)^{L/L_0} \quad (6)$$

这样,节点 i 传输一个长度为 L 字节数据包的平均传输次数

$$ETX_L = \frac{1}{1 - p_L} = \frac{1}{(1 - p)^{L/L_0}} \quad (7)$$

如前所述, L 符合取值范围为 34 ~ 2346 的均匀分布,故 ETX_L 的期望

$$\begin{aligned} ETX(i) &= E[ETX_L] \\ &= \frac{1}{2346 - 34} \int_{34}^{2346} \frac{1}{(1 - p)^{L/L_0}} dL \end{aligned} \quad (8)$$

由式(8)经计算可得

$$ETX(i) = \frac{L_0[(1 - p)^{-34/L_0} - (1 - p)^{-2346/L_0}]}{2312 \ln(1 - p)} \quad (9)$$

(3) 计算 $C(i)$.

当一条流量为 Q 的新流经过节点 i 时,在 ΔT 时间内它所获得的新增网络编码增益可以用下式计算:

$$\begin{aligned} \Delta GF(i) &= GF_{\text{new}}(i) - GF_{\text{old}}(i) \\ &= \frac{[IF_{\text{old}}(i) + Q] - OF_{\text{new}}(i)}{\Delta T} \\ &\quad - \frac{IF_{\text{old}}(i) - OF_{\text{old}}(i)}{\Delta T} \\ &= \frac{Q - [OF_{\text{new}}(i) - OF_{\text{old}}(i)]}{\Delta T} \end{aligned} \quad (10)$$

其中, $IF_{\text{old}}(i)$ 和 $OF_{\text{old}}(i)$ 分别为在新流流入之前节点 i 的输入流量和输出流量, $IF_{\text{new}}(i)$ 和 $OF_{\text{new}}(i)$ 分别表示在新流流入并参与编码之后节点 i 的输入流量和输出流量.

由式(3)可知,节点 i 的扩充信道容量 $C(i)$ 与 $\Delta GF(i)$ 成正比.因此,有必要分析新流的流量 Q 的变化对新增编码增益 $\Delta GF(i)$ 的影响.当 $Q < IF_{\text{old}}(i)$ 即新流流量小于经过节点 i 的已有流量时,新流的一部分流量可以通过编码被已有流量捎带走,因而随着 Q 增大 $OF_{\text{new}}(i)$ 与 $OF_{\text{old}}(i)$ 相差不大,这样由式(10)可知, $\Delta GF(i)$ 随着 Q 的增大而增大.然而,当 $Q > IF_{\text{old}}(i)$ 时,流量 $Q - IF_{\text{old}}(i)$ 无法和已有流进行编码被捎带走,因而随着 Q 的增大, $OF_{\text{new}}(i)$ 也随着增大,使得 $Q - OF_{\text{new}}(i)$ 保持不变,由式(10)知, $\Delta GF(i)$ 几乎不变.也

就是说,在 $Q > IF_{old}(i)$ 的条件下,加大输入流量 Q 不能产生新的编码增益.

基于上述考虑,在计算节点 i 的新流带来的编码增益时,本文取 $Q = IF_{old}(i)$. 其中, $IF_{old}(i)$ 表示经过节点 i 的已有流的流量之和(新流除外). 这样,由式(3)和式(10),可得:

$$C(i) = \frac{IF_{old}(i) - [OF_{new}(i) - OF_{old}(i)]}{\Delta T} + C_1(i)$$

(11)

5.2 路由发现

CGAR 与 AODV^[16]路由协议类似,寻路过程由信源节点发起,它向邻居节点广播路由请求包 RREQ(Route Request),中间节点向邻居转发 RREQ,信宿(目的节点)在收到 RREQ 之后向信源返回 RREP(Route Reply),信源在收到 RREP 之后,选择一条路径. 在信源与信宿之间有多条路径可选时,信源选择路径上各节点的 $\Phi(i)$ 之和最小的路径.

RREQ 包含域: RREQ 包标识 ReqID、当前节点标识 NID、信源 SRC、信宿 DST、上游节点集 UpSet (即流中信源到当前节点的路径上的节点集合)、上游节点一跳邻居集 UpNbSet 等域. RREP 包含域: RREP 包标识 RepID、ReqID、当前节点标识 NID、信源 SRC、信宿 DST、路径传输时延 Φ 、下游节点集 DnSet (流中当前节点到信宿的路径上的节点)、下游节点一跳邻居集 DnNbSet.

每个节点配置“寻路表”和“流表”,分别记录寻路过程用到的信息和节点中已有流的信息,前者包括域: ReqID、RepID、SRC、DST、 Φ 、UpSet、UpNbSet、DnSet、DnNbSet、NextNodeToSRC(本节点通向信源的下一跳节点)、NextNodeToDST(本节点通向信宿的下一跳节点)、Valid-Time (表示表中所对应的记录的保存时间,单位: ms),后者包括域: FID (流标识)、SRC、DST、UpSet、UpNbSet、DnSet、DnNbSet、 Q (Δt 时间内通过的流量;单位 Mb)、NextHopToSRC(通向信源的下一跳节点)、NextHopToDST (通向信宿的下一跳节点). 例如,对于图 4 中的节点 3,其流表如表 1 所示(其中 Q 值每隔 Δt 时间刷新一次).

表 1 流表

FID	SRC	DST	UpSet	UpNbSet	DnSet	DnNbSet	Q	NextHop ToSRC	NextHop ToDST
1	1	4	1,2	8	4	6	10	2	4
2	4	1	4	6	1,2	8	2	4	2
3	5	8	5,6	4	7,8	1	5	6	7

CGAR 协议的路由发现过程如下:

源节点向邻居节点广播 RREQ, 其中, ReqID 为 RREQ 包的唯一标识, NID 和 SRC 均为源节点编号, DST 是目的节点编号, UpSet 与 UpNbSet 置空. 同时在源节点的寻路表中增加一条记录(即增加一行), 并将上述内

容填入所对应的域中.

中间节点在收到 RREQ 包后,按以下逻辑处理 RREQ 包:

Step1 如果在收到的 RREQ 中,三元组 (SRC, DST, ReqID) 与该节点寻路表的某个记录相同,则丢弃 RREQ,转到 Step 7;

Step2 如果 RREQ 中二元组 (SRC, DST) 与寻路表所记录的对应内容相同,转到 Step3, 否则,转到 Step4;

Step3 如果寻路表中 UpSet 域是 RREQ.UpSet 的子集,则丢弃此 RREQ 包,转入 Step7; 否则,删除寻路表中的对应记录,转到 Step4;

Step4 在寻路表中增加一条新的记录,把 RREQ 包中的 ReqID、SRC、DST、UpSet、UpNbSet 域复制到寻路表,将 RREQ.NID 复制到寻路表的 NextNodeToSRC, FID 域内容自动产生(每次增加 1);

Step5 修改 RREQ 数据包: 把 NID 域修改为本节点编号,将本节点编号添加到 UpSet,将本节点的一跳邻居节点添加到 UpNbSet 域;

Step6 向邻居节点广播 RREQ 包.

Step7 END.

信宿在收到 RREQ 包后,作如下处理:

Step1 如果寻路表中记录的 UpSet 是 RREQ.UpSet 域的子集,则丢弃 RREQ 包,转入 Step4; 否则转到 Step2;

Step2 在寻路表中增加一条记录,把 RREQ 中的 ReqID、SRC、DST、UpSet、UpNbSet 域的数据复制其中,产生 RREP 的唯一标识 RepID 并放入寻路表,自动产生 FID, DnSet 域和 DnNbSet 域均为空,将 RREQ.NID 复制到寻路表的 NextNodeToSRC.

Step3 产生 RREP,其中域 Φ 初始化为 0,并将寻路表的内容复制到 RREP 的对应域中,将 RREP 单播给寻路表的域 NextNodeToSRC 所指的节点;

Step4 END.

中间节点(节点 i)收到 RREP 数据包后,作如下处理:

Step1 初始化 Old_GF = New_GF = 0 (Old_GF 和 New_GF 均为内存变量,分别用于记录该节点的旧编码增益和新编码增益);

Step2 根据三元组 (SRC, DST, ReqID) 在寻路表中找到对应的记录,并将 RREP 的 DnSet、DnNbSet 域中的内容填入其中;

Step3 如果流表中没有流,则取扩展信道容量 $C(i) = C_1(i)$,转到 Step6;

Step4 计算流表中所有流的流量之和 $IF(i)$ (即域

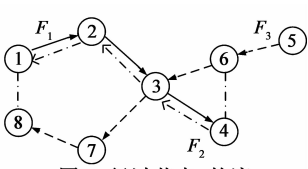


图4 经过节点3的流

Q 的值);

Step5 利用寻路表和流表中 UpSet 和 DnSet 内容,根据“编码条件”判断是否可编码,如果可编码,置 $Old_GF = New_GF$,根据前述“贪婪编码增益算法”计算编码增益 $GF(i)$,赋值 $New_GF = GF(i)$;

Step6 利用 Old_GF , New_GF , 根据式(10)计算新增编码增益 $\Delta GF(i)$;

Step7 按照式(9)计算 $ETX(i)$,进而根据式(4)计算 $\Phi(i)$,将之填入寻路表 Φ 域,并将之累加到 RREP. Φ 域;

Step8 把本节点添加到 RREP. DnSet 域,把本节点的一跳邻居节点添加到 RREP. DnNbSet 域,将 RREP. NID 复制到寻路表的 NextNodeToDST;

Step9 将 RREP 单播给寻路表的域 NextNodeToSRC 所指的节点;

Step10 End.

信源在收到 RREP 之后,就获得到达信宿的一条路径.如果信源收到多个三元组(SRC, DST, ReqID)相同的 RREP 包,则信源选择 Φ 值最小的路径以发送数据包.所选路径中的每个节点收到实际传输的数据包时在流表中加一条记录,并删除寻路表中的所有记录.

6 仿真试验

我们用 MATLAB 进行仿真试验.在 $100m \times 100m$ 的范围内随机产生 25 个节点,每个节点的通信范围均为 35m.探测包的长度 $L_0 = 137B$,其丢包率 p 与节点间的距离相关, ETX 值由式(9)确定, T_c 按文献[11]方法确定.每个节点的初始信道容量在 $1 \sim 54Mbps$ 之间随机产生.其它参数的取值分别为: $L = 1190 B$ (参见前述), $\delta = 0\mu s$ (考虑到数据包传播速度接近光速,传播时延很小,因此,传播时延忽略不计), $T_{ack} = 112\mu s$ (在 IEEE 802.11 标准中,ACK 包的大小占 14 字节,在发送速率为 1 Mbps 时发送时延为该值), $T_0 = 240\mu s$ (定时器时间取略大于两倍 ACK 包发送时延).

分别针对网络中有 1, 2, \dots , 10 条流(随机选取一对节点构成一条流的信源和信宿)的情况,在流的数量(简称“流数”)变化的情况下 CGAR、DCAR、COPE 这三种路由算法吞吐率和时延的仿真结果如图 5 和图 6 所示.

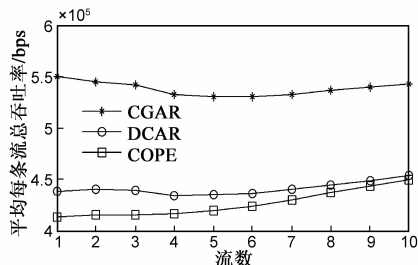


图5 各条流的平均吞吐率

从这两个图中可以看出:(1)对于给定的流数,CGAR 在吞吐量和时延上均优于 DCAR 和 COPE;(2)在网络中流数较小时,随着流数的增大,CGAR 的时延呈上升趋势,但在流数增大到一定程度之后,时延又呈现逐渐下降趋势,其主要原因是随着流数的增大,节点可以将来自多条流的数据包进行编码生成一个编码包,从而提高了效率,节省了时间.

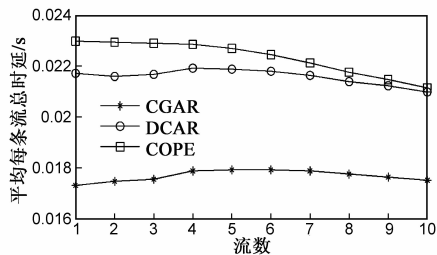


图6 各条流的平均时延

让探测包的成功传输率在 0.82 ~ 1.00 之间变化即丢包率在 0 ~ 0.18 之间变化,得到时延变化的仿真结果如图 7 所示.由此可见:(1)CGAR 在时延上仍然优于 DCAR 和 COPE;(2)CGAR 的时延随数据包成功传输率的增大而减小,这是因为成功传输率大,数据包重传次数减少,时延也相应减小.

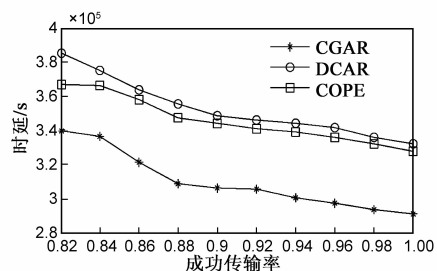


图7 成功传输率与时延关系

在流数变化的情况下,CGAR、DCAR、COPE 路由的路径失效率仿真结果如图 8 所示.所谓路径失效,是指路径上有节点因信道容量不足而无法发送数据包.此外,路径失效率定义为,路由算法所确定的路径总数中失效路径数量所占的比例.由图 8 可见:(1)对于给定的流数,CGAR 的路径失效率低于 DCAR 和 COPE;(2)随着流数的增大,三种路由协议的路径失效率也随着增大,这与直观是一致的,因为各条路径传递数据包的负荷随着流数的增大而增大,从而使带宽不足进而导致路

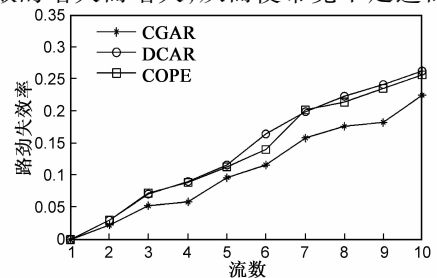


图8 路径失效率

径失效.

7 结束语

本文提出的 CGAR 路由协议利用了网络编码带来的增益,可以让信道携带比传统路由方法更多的数据量,可以降低数据包传递时延和因信道饱和导致的路径失效效率.

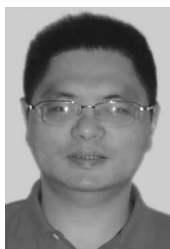
参考文献

- [1] R Ahlswede, N Cai, S Y R Li, et al. Network information flow [J]. IEEE Transactions on Information Theory, 2000, 46(4): 1204 – 1216.
- [2] S Katti, H Rahul, W Hu, et al. XORs in the air: Practical wireless network coding [J]. IEEE/ACM Transactions on Networking, 2008, 16(3): 487 – 510.
- [3] J Le, J C S Lui, D M Chiu. DCAR: Distributed coding-aware routing in wireless networks [J]. IEEE Transaction on Mobile Computing, 2010, 9(4): 596 – 608.
- [4] B Guo, H Li, et al. Analysis of general network coding conditions and design of a free-ride oriented routing metric [J]. IEEE Transactions on Vehicular Technology, 2011, 60(4): 1714 – 1727.
- [5] Y Yan, Z Zhao, B Zhang, et al. Rate-adaptive coding-aware multiple path routing for wireless mesh networks [A]. Proceedings of 2008 IEEE Global Telecommunications Conference [C]. New Orleans: IEEE Computer Society, 2008. 543 – 547.
- [6] M F Jhang, S W Lin, W Liao. C²AR: Coding and capacity aware routing for wireless Ad hoc networks [A]. Proceeding of 2010 IEEE International Conference on Communications [C]. Cape Town: IEEE Computer Society, 2010. 1 – 5.
- [7] 卢文伟, 朱艺华, 陈贵海. 无线传感器网络中基于线性网络编码的节能路由算法 [J]. 电子学报, 2010, 38(10): 2309 – 2314.
W W Lu, Y H Zhu, G H Chen. Energy-efficient routing algorithms based on linear network coding in wireless sensor networks [J]. Acta Electronica Sinica, 2010, 38(10): 2309 – 2314. (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.
J Lu, S Xiao, C K Wu. Opportunistic network coding based delay-sensitive broadcast transmission algorithm [J]. Acta Electronica Sinica, 2011, 39(5): 1214 – 1219. (in Chinese)
- [9] 汪玉, 卢汉成, 洪佩琳等. 网络编码中的侦听管理策略 [J]. 电子学报, 2012, 40(1): 47 – 52.
Y Wang, H C Lu, P L Hong, et al. Overhearing management policies on network coding [J]. Acta Electronica Sinica, 2012, 40(1): 47 – 52. (in Chinese)
- [10] D S J De Couto, D Aguayo, J Bicket, et al. A high-throughput

path metric for multi-hop wireless routing [A]. Proceedings of 9th Annual International Conference on Mobile Computing and Networking [C]. San Diego: Association for Computing Machinery, 2003. 419 – 434.

- [11] R Draves, J Padhye, B Zill. Routing in multiradio, multihop wireless mesh networks [A]. Proceedings of ACM MOBICom [C]. Philadelphia: Association for Computing Machinery, 2004. 114 – 128.
- [12] B Ni, N Santhapuri, Z Zhong, et al. Routing with opportunistically coded exchanges in wireless mesh networks [A]. Proceedings of the 2nd IEEE Workshop on Wireless Mesh Networks [C]. Reston: IEEE Computer Society, 2006. 157 – 159.
- [13] Y Wu, S M Das, R Chandra. Routing with a Markovian metric to promote local mixing [A]. Proceedings of 26th IEEE International Conference on Computer Communications [C]. Anchorage: IEEE Computer Society, 2007. 2381 – 2385.
- [14] S Keshav. A Control-theoretic approach to flow control [A]. Proceedings of the Conference on Communications Architecture & Protocols [C]. Zürich: Association for Computing Machinery, 1991. 3 – 15.
- [15] Matthew S. Gast. 802.11 Wireless Networks-The Definitive Guide [M]. O'Reilly Media Inc, 2005.
- [16] C E Perkins, E M Royer. Ad-hoc on-demand distance vector routing [A]. Proceedings of the Second IEEE Workshop on Mobile Computing Systems and Applications [C]. New Orleans: IEEE Computer Society, 1999. 90 – 100.

作者简介



田贤忠 男, 1968 年出生于浙江杭州, 浙江工业大学副教授, 博士生, 研究方向为网络编码、移动计算、无线网络的协议、算法、性能分析与优化等.

E-mail: txz@zjut.edu.cn



朱艺华(通信作者) 男, 1961 年生于浙江玉环, 博士, 教授, 博士生导师, IEEE 高级会员, 中国计算机学会传感器网络专业委员会委员, 主要研究方向为移动计算、无线网络的协议、算法、性能分析与优化, 在《IEEE Transactions on Wireless Communications》、《IEEE Transactions on Vehicular Technology》、《IEEE Transactions on Network and Service Management》等各类期刊及 IEEE 国际会议论文集上发表学术论文 120 余篇.

缪得志 男, 1985 年出生于浙江湖州, 浙江工业大学在读硕士研究生, 研究方向为无线网络、网络编码.