

基于随机线性网络编码的双源交替调度算法

汪 玉 卢汉成 洪佩琳* 薛开平

(中国科学技术大学电子工程与信息科学系信息网络实验室 合肥 230027)

摘要:为减少单源组播中链路质量最差的组播成员对组播速率的影响,该文提出了适用于双源可靠组播的静态和动态交替调度算法。算法通过交替传输编码数据包,可充分挖掘组播成员到不同源节点链路质量的差异性。与单源组播相比,理论分析和仿真表明:两种调度算法可以有效提高组播速率,从而减少期望传输时间。动态交替调度算法通过动态更新参数,可以获得更低的期望传输时间。

关键词:无线通信;随机线性网络编码;可靠组播;删除信道;调度算法

中图分类号: TP393

文献标识码: A

文章编号: 1009-5896(2011)12-3008-07

DOI: 10.3724/SP.J.1146.2011.00297

Random Linear Network Coding Based Alternative Scheduling Algorithms with Two Sources

Wang Yu Lu Han-cheng Hong Pei-lin Xue Kai-ping

(The Information Network Lab of EEIS Department, University of Science and Technology of China, Hefei 230027, China)

Abstract: To lessen the impact of multicast member with the worst link quality on multicast rate in single source multicast, static and dynamic alternative scheduling algorithms are presented for reliable multicast with two sources. Through transmitting encode packets alternately, the presented algorithms exploit the link heterogeneity between multicast members and different sources. Compared with single source multicast, theoretical analysis and simulations show that both scheduling algorithms can effectively improve the multicast rate, thereby decrease expected time for completing transmission. Dynamic alternative scheduling algorithm achieves less expected transmission time by dynamically updating parameter.

Key words: Wireless communication; Random linear network coding; Reliable multicast; Erasure channel; Scheduling algorithm

1 引言

在无线环境中,组播通信可以节省宝贵的带宽资源,引起了越来越多的关注,其中,可靠组播保证 100%的数据投递率,在可靠文件传输、多媒体会议等方面存在广泛应用。传统的可靠组播通过自动重传请求(Automatic Repeat reQuest, ARQ)、前向纠错(Forward Error Correction, FEC)和混合 ARQ-FEC 等技术实现。在 ARQ 技术中,接收节点通过反馈使源节点重传丢失的数据包, FEC 技术通过传输冗余信息减少反馈次数,混合 ARQ-FEC 将二者结合以获取更高的性能。

在文献[1]提出网络编码后,文献[2,3]首先把它应用到无线环境下的可靠组播中,将发往不同接收节点的丢失数据包进行异或编码,保证丢失数据包

所属的接收节点都可以正确解码,从而减少重传次数。肖潇等人^[4]针对多个接收节点,提出一种简单有效的重传策略。对于基于异或操作的网络编码,Chi 等人^[5]证明了寻找最大可编码的丢失数据包集合是一个 NP 完全问题,为此,文献[5,6]提出多种启发式算法以减少重传次数。基于异或操作的网络编码仅将部分丢失数据包进行编码,因而浪费了一定的编码机会,更重要的是在每次传输后需要收集接收节点丢失数据包信息,因而需要较多的反馈信息。

为进一步降低传输时间,随机线性网络编码^[7]被引入到可靠组播中来,源节点传输多个原始数据包的随机线性组合直到所有接收节点恢复原始数据包,此时,源节点不需要获取接收节点的丢失数据包信息。当随机系数所属的有限域足够大时,未完成传输的接收节点接收到的任意编码包都是有益的,因此,基于随机线性网络编码的性能要优于基于异或操作的可靠组播性能。Eryilmaz 等人^[8]给出了采用随机线性网络编码的期望传输时间的闭合表

2011-03-30 收到, 2011-09-02 改回

国家自然科学基金(60832005, 61170231)和国家重大专项(2011ZX03005-006)资助课题

*通信作者: 洪佩琳 plhong@ustc.edu.cn

达式, Ghaderi 等人^[9]讨论在基于树的可靠组播中, 随机线性网络编码的性能要远优于 ARQ 和端到端 FEC 技术。与传统的 ARQ 技术相比, 文献[10]进一步证明了随机线性网络编码的性能增益为 $\Theta(\log N)$, 其中 N 为接收节点数目。

对于单源组播, 无论是何种调度算法, 文献[3,11]指出组播速率均受限于链路质量最差的接收节点。本文引入额外的源节点并分析双源组播系统的组播速率上界; 提出静态和动态的交替调度算法, 可充分挖掘接收节点与不同源节点之间的链路质量的差异性; 通过随机线性网络编码的使用, 可近似消除重复数据包的接收并降低所需要的反馈次数。理论和实验结果表明, 与单源组播相比, 静态和动态交替调度算法可以有效提高组播速率并接近组播速率上界, 从而减少期望传输时间, 动态交替调度算法通过动态更新参数, 可以获得更低的期望传输时间。

本文第 2 节介绍随机线性网络编码的基本原理, 第 3 节讨论系统模型并分析其组播速率上界, 第 4 节为静态和动态交替调度算法及其性能分析, 第 5 节为仿真分析, 第 6 节为结束语。

2 随机线性网络编码基本原理

在随机线性网络编码中, 以代为单位对数据包进行编码, 每个代包含 K 个原始数据包, K 称为代大小。如式(1)所示, 源节点将同一个代内的原始数据包 P_1, P_2, \dots, P_K 进行线性组合得到编码包, 其中线性组合的系数 $g_i (1 \leq i \leq K)$ 在有限域 $\text{GF}(q)$ 中随机选取, q 为有限域大小, 一般取 $q = 2^d$, 其中 d 为正整数。

$$P_{\text{encoded}} = \sum_{i=1}^K g_i P_i \quad (1)$$

当节点接收到 K 个独立的编码包后, 可通过高斯消元法得到 K 个原始数据包。定义事件 E_K 为节点接收到 K 个编码包后即可恢复原始数据包, 由文献[11]可知 E_K 发生的概率如式(2)所示。

$$P(E_K) = \prod_{k=1}^K (1 - 1/q^k) \quad (2)$$

当编码系数在有限域 $\text{GF}(2^8)$ 中选取时, 节点接收到 K 个编码包后即可解码的概率为 99.61%。因此在之后的分析中可作如下假设: 节点接收到 K 个编码包后即可成功解码。在随机线性网络编码中, 系数作为编码数据包的一部分参与传输, 令原始数据包大小为 b bit, 编码包中系数部分占总数据量的比重为 $Kd/(Kd+b)$, 一般来说, $Kd \ll b$, 因此系数大小的影响可忽略不计。

3 系统模型及组播速率上界

考虑图 1 所示的系统模型, 数据源生成数据包, 经两个转发节点以组播方式发送到 N 个接收节点, 相比于转发节点和接收节点之间的无线链路, 数据源与转发节点之间通过可靠的有线链路连接, 可忽略数据在有线链路传输所需要的时间。本文分析 K 个原始数据包从转发节点组播到所有接收节点所需要的期望传输时间, 称转发节点 S_1, S_2 为源节点。在现有无线场景中, 接收节点可接收到多个基站信号, 且接收节点到不同基站的链路质量差异较大, 因此符合图 1 所示的系统模型。

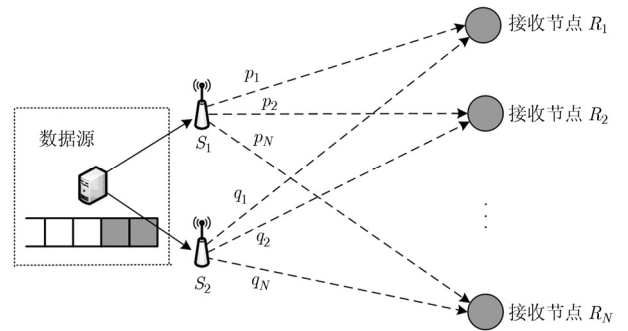


图 1 系统模型

假设源节点以时隙为单位传输数据包, 在一个时隙内仅传输一个数据包, 为防止干扰, 在任意时隙内, 仅有一个源节点传输数据包。当 $S_1 (S_2)$ 传输数据包时, 接收节点 i 成功接收的概率为 $p_i (q_i)$, 并假设源节点到不同接收节点之间的信道是独立的。源节点对 K 个原始数据包进行随机线性网络编码并组播形成的编码包, 接收节点接收到 K 个独立的编码包后经独立信道发送反馈信息至源节点, 当源节点接收到所有接收节点发来的反馈信息后终止传输。相比于普通数据包, 在相同的比特错误率下, 由于反馈信息包比特数很少, 因而可假设反馈信息的传输是可靠的。

考虑图 1 中存在两个接收节点的场景, 假设接收节点到两个源节点之间的成功传输概率分别为 $[p_1, p_2] = [0.3, 0.8]$, $[q_1, q_2] = [0.8, 0.3]$ 。如果仅有 S_1 传输编码数据包, 组播速率受限于 S_1 与 R_1 之间的链路 ($p_1 = 0.3$); 如果仅有 S_2 传输编码数据包, 组播速率受限于 S_2 与 R_2 之间的链路 ($q_2 = 0.3$)。如果 S_1 和 S_2 均参与传输编码数据包, 则 S_1 与 R_1 之间的链路对组播速率的限制由 S_2 减轻, 因为 S_2 与 R_1 之间的链路质量较好 ($q_1 = 0.8$); 同理, S_2 与 R_2 之间的链路对组播速率的限制由 S_1 减轻。因此, 利用不同源节点到接收节点的链路质量差异, 通过源节点之间

的协作传输,可以有效减少组播所需要的平均传输时间。

在图1所示的系统模型中,令源节点总传输次数为 t ,源节点 S_1 传输 t_1 次,则源节点 S_2 传输 $t-t_1$ 次。接收节点 i ($1 \leq i \leq N$)接收到来自 S_1 的数据包数目为 $u_{1,i}(t)$,接收到来自 S_2 的数据包数目为 $u_{2,i}(t)$,一般定义接收节点 i 的接收速率如下:

$$M_{i,t}(t_1) = E(u_{1,i}(t) + u_{2,i}(t)) / t = q_i + (p_i - q_i)t_1 / t \quad (3)$$

接收速率的物理意义为单位时隙内接收到的数据包数目的平均值。组播速率由所有接收节点的最小接收速率决定。

$$\begin{aligned} M_t(t_1) &= \min\{M_{1,t}, M_{2,t}, M_{3,t}, \dots, M_{N,t}\} \\ &= \min_i \{q_i + (p_i - q_i)t_1 / t\} \end{aligned} \quad (4)$$

所有接收节点的最大组播速率如下:

$$M_{t,\max} = \max_{t_1} \left\{ \min_i \{q_i + (p_i - q_i)t_1 / t\} \right\} \quad (5)$$

定理1 对于图1所示的双源组播系统,组播速率上界如下:

$$M_{\text{Bound}} = \max_{\alpha} \left\{ \min_i \{q_i + \alpha(p_i - q_i)\} \right\}, \quad 0 \leq \alpha \leq 1 \quad (6)$$

证明 式(5)给出了总传输次数为 t 时的最大组播速率,此时 $t_1 \in \{0, 1, 2, \dots, t\}$,最大组播速率仅在离散的 $t+1$ 个值中取得。而当 $t \rightarrow \infty$ 时,可令 $\alpha = t_1 / t$,此时 $0 \leq \alpha \leq 1$,可得组播速率上界为 $\max_{\alpha} \left\{ \min_i \{q_i + \alpha(p_i - q_i)\} \right\}$ 。证毕

由文献[3,11]可知,单源组播的组播速率受限于链路质量最差的组播成员,其上界为 $\min_i \{p_i\}$ (或 $\min_i \{q_i\}$),易知 $M_{\text{Bound}} \geq \min_i \{p_i\}$ (或 $\min_i \{q_i\}$)。因而,双源组播可以提高组播速率上界且提高的幅度由接收节点到源节点的成功传输概率分布(p_i 和 $q_i, 1 \leq i \leq N$)决定。

4 交替调度算法

4.1 静态交替调度算法

本文提出一种静态交替调度算法以趋近组播速率上界,算法如下:对于给定的带宽分配因子

α ($0 \leq \alpha \leq 1$),令源节点总传输次数为 t ,对于任意的整数 t ,分配给源节点 S_1 和 S_2 的传输次数分别为 $[\alpha t]$ 和 $t - [\alpha t]$,其中“ $[\bullet]$ ”为四舍五入取整运算。当 t 趋向于无穷大时,带宽分配因子 α 表示源节点 S_1 传输次数占总次数的比重。如果 α 已经确定,源节点的传输顺序也可确定。例如,当 $\alpha=0.7$ 时,传输顺序如下: $S_1 \rightarrow S_2 \rightarrow S_1 \rightarrow S_1 \rightarrow S_1 \rightarrow S_2 \rightarrow S_1 \rightarrow S_1 \rightarrow S_2 \rightarrow \dots$ 。在该调度算法中,源节点可以确定任意一个时隙中自己是否需要传输数据包,因此不存在源节点之间的信道竞争。源节点以固定的带宽分配因子按上述规则交替传输编码数据包直到接收到所有接收节点的反馈信息,因此该调度算法称作静态交替调度算法。具体实现中使用 α 的近似解 α_e ,其算法描述如表1所示。

表1 静态交替调度算法

输入: p_i, q_i ($1 \leq i \leq N$)	
(1)	根据式(14)计算 α_e
(2)	for $t=1$ to ∞ then
(3)	if 源节点接收到所有接收节点的反馈 then
(4)	传输过程终止; break
(5)	end if
(6)	if $[\alpha_e t] - [\alpha_e(t-1)] = 1$ then
(7)	源节点 S_1 传输一个编码包
(8)	else
(9)	源节点 S_2 传输一个编码包
(10)	end if
(11)	end for

为简化分析,在理论推导中假设反馈是即时的。对于任意的带宽分配因子 α ($0 \leq \alpha \leq 1$),源节点 S_1 和 S_2 的传输次数 t_1, t_2 如下:

$$t_1 = [\alpha t] \quad (7)$$

$$t_2 = t - [\alpha t] \quad (8)$$

定义随机变量 T_i 为接收节点 i 恢复原始数据包所需要的传输时间,其概率分布如下:

$$P[T_i = t] = \begin{cases} \sum_{m \in M_1} \left\{ \binom{t_1-1}{m} p_i^{m+1} (1-p_i)^{t_1-m-1} \binom{t_2}{K-m-1} q_i^{K-m-1} (1-q_i)^{t_2-K+m+1} \right\}, & [\alpha t] - [\alpha(t-1)] = 1 \\ \sum_{m \in M_2} \left\{ \binom{t_2-1}{m} q_i^{m+1} (1-q_i)^{t_2-m-1} \binom{t_1}{K-m-1} p_i^{K-m-1} (1-p_i)^{t_1-K+m+1} \right\}, & [\alpha t] - [\alpha(t-1)] = 0 \end{cases} \quad (9)$$

$$M_1 = \{m \mid 0 \leq m \leq t_1 - 1, K - t_2 - 1 \leq m \leq K - 1\}$$

$$M_2 = \{m \mid 0 \leq m \leq t_2 - 1, K - t_1 - 1 \leq m \leq K - 1\}$$

式中 $[\alpha t] - [\alpha(t-1)] = 1$ 表示最后成功传输的数据包来自 S_1 ，此时式(9)为源节点 S_1 传输成功 $m+1$ 次，失败 $t_1 - m - 1$ 次且源节点 S_2 传输成功 $K - m - 1$ 次，失败 $t_2 - K + m + 1$ 次的概率对 m 的累加，其中整数 $m \in M_1$ ；同理可得 $[\alpha t] - [\alpha(t-1)] = 0$ 即最后传输的数据包来自 S_2 时的概率。定义随机变量 T 为所有接收节点恢复原始数据包所需要的时间。则

$$T = \max\{T_1, T_2, T_3, \dots, T_N\} \quad (10)$$

由于接收节点接收数据包是相互独立的，因此可得 T 的分布函数如下：

$$P[T \leq t] = P[T_1 \leq t, T_2 \leq t, T_3 \leq t, \dots, T_N \leq t] \\ = \prod_{i=1}^N P[T_i \leq t] = \prod_{i=1}^N \sum_{r=K}^t P[T_i = r] \quad (11)$$

所有接收节点恢复原始数据包所需要的期望传输时间如下：

$$E[T] = \sum_{t=K}^{\infty} tP[T = t] = K + \sum_{t=K}^{\infty} (t-K)P[T = t] \\ = K + \sum_{t=K}^{\infty} P[T > t] = K + \sum_{t=K}^{\infty} \left(1 - \prod_{i=1}^N \sum_{r=K}^t P[T_i = r] \right) \quad (12)$$

仿真场景如图 2 所示，接收节点随机分布在图中矩形区域内，仅考虑自由空间传播模型，部分参数如下：基站间距为 500 m，传输速率为 2×10^6 bps，数据包大小为 1000 bit，采用 QPSK 调制，带宽为 2×10^6 Hz，载波频率为 2×10^9 Hz，可根据接收节点到基站的距离计算丢包率，例如距离基站最近和最远的两个接收节点的成功传输概率分别为 0.9528 和 0.0902。此时，处于矩形区域内的接收节点均可接收到来自 S_1 和 S_2 的信号，因而符合图 1 所示的系统模型。

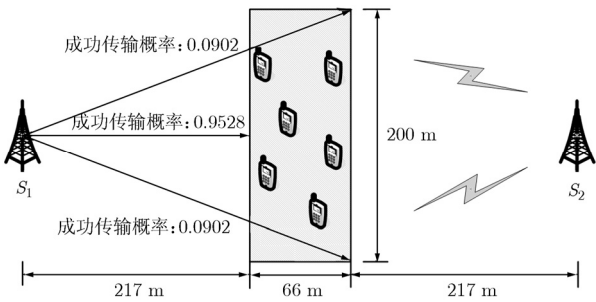


图 2 仿真场景

在图 2 所示的仿真场景中随机取 5 个接收节点并计算接收节点与源节点之间的成功传输概率，进而可计算期望传输时间的理论值如图 3 所示。

图 3 中 $\alpha=0$ 表示仅有 S_2 传输编码数据包，此时 $E[T]=30.60$ ； $\alpha=1$ 表示仅有 S_1 传输编码数据包，

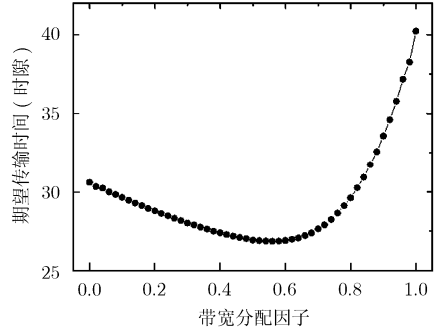


图 3 期望传输时间($K=10, N=5$)

此时 $E[T]=40.22$ 。当 $\alpha=0.56$ 时，源节点之间通过协作传输使得期望传输时间达到最小值，此时 $E[T]=26.87$ 。定义使得 $E[T]$ 最小的带宽分配因子为最优解，可表述如下：

$$\alpha_o = \arg \min_{\alpha} \{E[T]\} \quad (13)$$

由式(9)和式(12)可知难以得到 α_o 的闭合表达式，在静态交替调度方案中，我们采用近似解作为其带宽分配因子。定义近似解如下：

$$\alpha_e = \arg \max_{\alpha} \left\{ \min_i \{q_i + (p_i - q_i)\alpha\} \right\} \quad (14)$$

近似解的求取的目标是使得组播速率可近似达到组播速率上界，由式(4)可知

$$M_t(t_1) = \min_i \{q_i + (p_i - q_i)t_1 / t\} \\ = \min_i \{q_i + (p_i - q_i)[\alpha_e t] / t\} \\ \approx \min_i \{q_i + (p_i - q_i)\alpha_e\} \\ = \max_{\alpha} \left\{ \min_i \{q_i + (p_i - q_i)\alpha\} \right\} = M_{\text{Bound}} \quad (15)$$

此时，式(12)中组播所需要的期望传输时间亦可近似达到期望传输时间下界如式(16)：

$$E[T_{\text{Bound}}] = K / M_{\text{Bound}} \quad (16)$$

近似解的求解可转化为线性规划问题，此处不再赘述。由于对某些接收节点 $p_i = q_i$ ，近似解可能在一个区间内取值，此时，区间的 midpoint 可作为静态交替调度算法的带宽分配因子。图 4 为带宽分配因子分别取近似解和最优解时的期望传输时间对比，二者的单位数据包期望传输时间的差别不超过 3.3%，且随着代大小的增加，二者均趋向于单位数据包期望传输时间下界 $1/M_{\text{Bound}}$ 。但是，更高的代大小会带来更大的延时和解码复杂度，例如使用高斯消元法的解码复杂度为 $O(K^3)$ 。

4.2 动态交替调度算法

在静态交替调度算法中，源节点以固定的带宽分配因子交替传输编码数据包直到所有接收节点恢复所有原始数据包。在实际调度过程中，源节点依

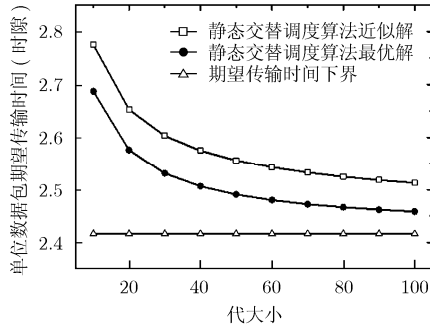


图4 近似解与最优解性能对比(N=5)

次接收到接收节点发送的反馈信息, 因此, 可以根据该反馈信息更新带宽分配因子, 从而进一步提高交替调度算法性能, 降低期望传输时间。我们称之为动态交替调度算法, 其算法描述如表2所示。

表2 动态交替调度算法

```

输入:  $p_i, q_i (1 \leq i \leq N)$ 
(1) 初始化:  $t_i = 0$ 
(2) for  $t=1$  to  $\infty$  then
(3)   if 源节点接收到所有接收节点的反馈 then
(4)     传输过程终止; break
(5)   end if
(6)   根据式(17)计算  $\alpha_e^*$ 
(7)   if  $[\alpha_e^* t] > t_i$  then
(8)     源节点  $S_1$  传输一个编码包;  $t_i \leftarrow t_i + 1$ 
(9)   else
(10)    源节点  $S_2$  传输一个编码包
(11)  end if
(12) end for

```

为简化分析, 在理论分析中同样假设反馈是即时的。定义状态 s 为 $(t_1, t_2, u_1, u_2, \dots, u_N)$, 表示经过 t_1 次 S_1 传输, t_2 次 S_2 传输, 接收节点 i 接收到 u_i 个独立编码包, 其中 $t_1, t_2 \geq 0; 0 \leq u_i \leq K, 1 \leq i \leq N$ 。可忽略已恢复所有原始数据包的接收节点的链路质量对带宽分配因子计算的影响, 因此, 可得更新后的带宽分配因子如下:

$$\alpha_e^* = \arg \max_{\alpha} \left\{ \min_{\{i|u_i < K\}} \{q_i + (p_i - q_i)\alpha\} \right\} \quad (17)$$

当 $u_i (1 \leq i \leq N)$ 不全为 K 时(传输过程未结束), 定义状态 s^* 为 $(t_1^*, t_2^*, u_1^*, u_2^*, \dots, u_N^*)$, 表示经过一次传输后由状态 s 转移到状态 s^* , 其中 $u_i^* = u_i$ 或者 $u_i^* = u_i + 1, 0 \leq u_i^* \leq K, 1 \leq i \leq N, t_1^*, t_2^*$ 由式(18)决定。

$$\begin{cases} t_1^* = t_1 + 1, t_2^* = t_2, [(t_1 + t_2 + 1)\alpha_e^*] > t_1 \\ t_1^* = t_1, t_2^* = t_2 + 1, [(t_1 + t_2 + 1)\alpha_e^*] \leq t_1 \end{cases} \quad (18)$$

$t_1^* = t_1 + 1$ 表示到达状态 s 后下一次由 S_1 传输数据, $t_1^* = t_1$ 表示到达状态 s 后下一次由 S_2 传输数

据。定义 $P(s)$ 为经过 $t_1 + t_2$ 次传输后转移到状态 s 的概率, $P(s^*)$ 为经过 $t_1 + t_2 + 1$ 次传输后转移到 s^* 的概率, 则

$$P(s^*) = \sum_{s \in \Omega} P(s)P(s^*|s) \quad (19)$$

其中 Ω 为所有可经过一次传输转移到 s^* 的状态集合, 转移概率如下:

$$P(s^*|s) = \prod_{\{i|u_i < K\}} \{(p_i(t_1^* - t_1) + q_i(t_2^* - t_2))^{u_i^* - u_i} \cdot (1 - p_i(t_1^* - t_1) - q_i(t_2^* - t_2))^{1 - (u_i^* - u_i)}\} \quad (20)$$

动态交替调度算法的期望传输时间为

$$E[T] = \sum_{u_i = K, 1 \leq i \leq N} P(s)(t_1 + t_2) \quad (21)$$

由理论分析可知, 随着接收节点数目和代大小的增加, 动态交替调度算法的状态数呈指数增加, 难以计算 K 较大时的期望传输时间的准确理论值, 此处针对图3所讨论的场景, 仅计算当 K 较小时动态和静态交替调度算法性能对比如图5, 在第5节中通过仿真观察当 K 较大时的性能对比情况。

由图5可知, 动态交替调度算法略优于静态交替调度算法, 这是由于动态交替调度算法根据接收节点的反馈信息动态调整带宽分配因子; 单位数据包期望传输时间随着代大小的增加而减小, 这是由于代大小越大, 交替调度算法的近似解越接近最优解; 两种调度算法与组播速率上界存在一定差距, 这是由于 K 较小时, 难以发挥交替调度算法的优势。需要注意的是, 对于不同的成功传输概率分布, 静态和动态交替调度算法能够得到不同的性能增益。例如, 对于 $p_i = q_i (1 \leq i \leq N)$, 双源交替调度算法与单源组播所需的期望传输时间相同。

5 仿真结果与分析

本节使用 C++ 实现的离散时间调度器仿真不同参数下静态和动态交替调度算法的性能, 主要包括有限域大小 (q), 代大小 (K), 接收节点数目 (N) 和反馈信息延时对期望传输时间的影响。仿真场景如图2所示, 仿真结果为1000次试验结果的平均值。

在理论分析中, 我们假设随机线性网络编码系数所在有限域足够大, 但是较大的有限域会带来更高的计算复杂度, 表现为有限域上的四则运算更为复杂。为此, 图6通过仿真观察有限域大小对静态交替调度算法性能的影响。由图6可知, 当 GF 域大小不小于16时, 仿真值几乎与理论值相同; 当 GF 域大小为2(XOR操作)或4时, 编解码操作较容易, 但是编码数据包之间的较高的相关性导致需要更多的传输时间。在之后的仿真中, 为保证足够高的数据包独立性, 均采用 GF(256)。

图 7 为 K 较大时动态和静态交替调度算法仿真对比图，其中期望传输时间下界为 $1/M_{\text{Bound}}$ ，与图 5 中 K 较小时的理论结论相类似，动态交替调度算法性能略好，且随着代大小的增加，单位数据包期望传输时间减小并接近期望传输时间下界。图 8 仿真了不同接收节点数目所需要的期望传输时间，其中的曲线波动是由于新增了链路质量更差的接收节点。相比于单源节点的可靠组播，静态和动态交替调度算法可以大大减少期望传输时间；在动态交替调度算法中，源节点根据接收节点的反馈信息动态调整带宽分配因子，因而可以获得更低的期望传输时间；静态和动态交替调度算法均接近期望传输时间下界。

在理论分析中，我们假设接收节点的反馈信息是即时的，图 9 给出了考虑反馈信息传输时间的静态和动态交替调度算法性能。由于反馈信息通过独立信道传输且仅需较少比特，仿真表明反馈信息可在一个数据包传输时隙内传输完毕。为提高调度效率，两种调度算法在调度时不需等待反馈信息。由图 9 可知，反馈信息的延迟仅增加约 1 时隙的传输

时间，这是由于最后一个接收节点的反馈信息到达之前源节点多传输了一次编码包；反馈信息的延迟还会影响动态交替调度算法带宽分配因子的准确度，但仿真结果表明此类影响可忽略不计。因而，动态和静态交替调度算法受延迟的反馈信息影响较小。

6 结束语

本文讨论双源组播系统的速率界并提出静态和动态交替调度算法，通过随机线性网络编码的使用，可充分挖掘接收节点到不同源节点的链路质量的差异性。理论和仿真结果表明：与单源组播相比，静态和动态交替调度算法可以提高组播速率并近似达到组播速率上界，从而有效降低期望传输时间；动态交替调度算法通过动态更新参数，可以获得更低的期望传输时间。进一步工作可将双源组播扩展为基站和中继之间的协作组播，此时基站与中继可看做源节点，而中继与基站之间通过质量较好的无线链路连接，同样可利用接收节点到基站和中继链路质量的差异性，减少组播所需要的期望传输时间。

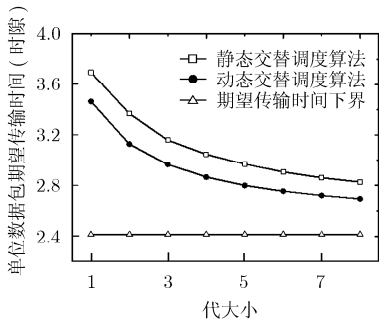


图 5 动态和静态交替调度算法对比($N=5$)

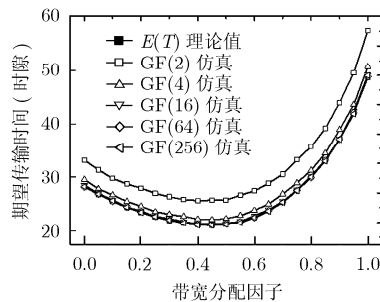


图 6 有限域大小影响($K=10, N=10$)

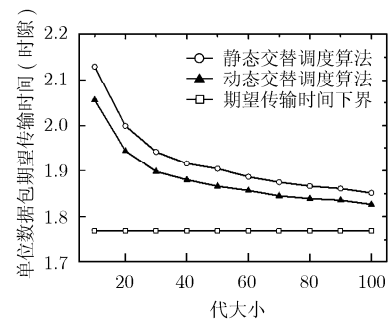


图 7 代大小影响($N=10$)

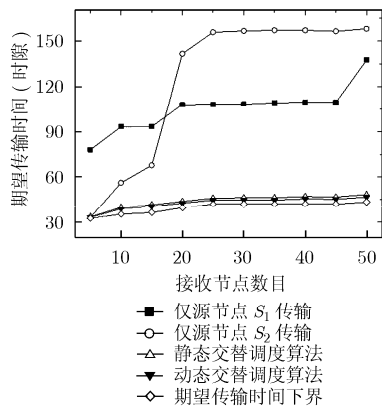


图 8 不同接收节点数目的性能对比($K=20$)

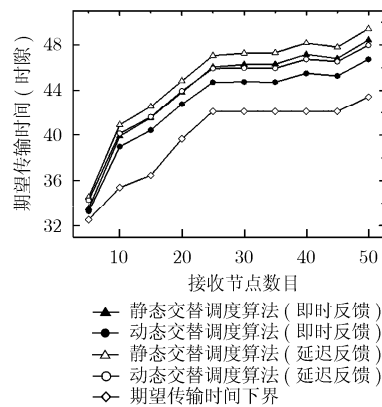


图 9 反馈信息延迟的影响($K=20$)

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] Nguyen D, Nguyen T, and Bose B. Wireless broadcasting using network coding[R]. Technical Report: OSU-TR-2006-06, Oregon State University, 2006.
- [3] Nguyen D, Tran T, Nguyen T, *et al.* Wireless broadcasting using network coding[J]. *IEEE Transactions on Vehicular Technology*, 2009, 58(2): 914–925.
- [4] 肖潇, 王伟平, 杨路明, 等. 基于网络编码的无线网络广播重传方法[J]. *通信学报*, 2009, 30(9): 69–75.
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): 69–75.
- [5] Chi K K, Jiang X H, Ye B L, *et al.* Efficient network coding-based loss recovery for reliable multicast in wireless networks[J]. *IEICE Transactions on Communications*, 2010, E93-B(4): 971–981.
- [6] Zhan C, Xu Y L, Wang J P, *et al.* Reliable multicast in wireless networks using network coding[C]. IEEE 6th International Conference on Mobile Ad hoc and Sensor System, Macou, China, 2009: 506–515.
- [7] Ho T, Koetter R, Medard M, *et al.* The benefits of coding over routing in a randomized setting[C]. IEEE International Symposium on Information Theory, Pacifico Yokohama, Kanagawa, Japan, 2003: 442.
- [8] Eryilmaz A, Ozdaglar A, Medard M, *et al.* On the delay and throughput gains of coding in unreliable networks[J]. *IEEE Transactions on Information Theory*, 2008, 54(12): 5511–5524.
- [9] Ghaderi M, Towsley D, and Kurose J. Network coding performance for reliable multicast[C]. Military Communications Conference(MILCOM), Orlando, FL, USA, 2007: 746–752.
- [10] Ghaderi M, Towsley D, and Kurose J. Reliability gain of network coding in lossy wireless networks[C]. The 27th Conference on Computer Communications (INFOCOM), Phoenix, AZ, USA, 2008: 2171–2175.
- [11] Sagduyu Y E and Ephremides A. On network coding for stable multicast communication[C]. Military Communications Conference(MILCOM), Orlando, FL, USA, 2007: 1425–1431.
- 汪玉: 男, 1987年生, 博士生, 研究方向为无线环境下的网络编码性能分析.
- 卢汉成: 男, 1977年生, 副教授, 研究方向为下一代移动互联网与极端网络环境下的路由机制.
- 洪佩琳: 女, 1961年生, 教授, 博士生导师, 主要研究方向为下一代网络体系结构与网络安全.