

一种基于网络编码的应用层多播算法

史玉琢¹, 郝 琨²

(1.天津财贸管理干部学院, 天津 300170;

2.天津城市建设学院 电子与信息工程系, 天津 300384)

摘要: 研究了对多播网络进行网络编码的方法, 提出了一种基于网络编码的应用层多播算法。该算法在计算网络拓扑时考虑了链路的花费。源端和中间节点使用随机线性编码方法进行编码, 在目的端进行解码操作使得目的端能从乱序的信息和部分丢失的信息中恢复出原始数据, 提高了网络的可靠性。通过对 ns-2 的扩展并进行仿真实验, 结果证明了基于网络编码的应用层多播算法是可以提高网络的吞吐量, 并且和网络中最大的吞吐量比较接近。在信息块不是很大的情况下, 编码延迟率的增长是在一定的范围内的。

关键词: 多播; 网络编码; 吞吐量

中图分类号: TN911.22

文献标识码: A

An algorithm of application layer multicast based on network coding

SHI Yu Zhuo¹, HAO Kun²

(1.Tianjin Institute of Financial and Commercial Management, Tianjin 300170, China;

2.Electronic & Information Engineering Department, Tianjin Institute of Urban Construction, Tianjin 300384, China)

Abstract: This paper proposes an algorithm of application layer multicast based on network coding. A minimum cost algorithm is used when the network topology is established. Source and intermediate nodes code based on random linear mechanism, sink node decodes from the disordered and losing message to restore original data, which raises the credibility of the network. From the simulation results analysis, the throughput algorithm on random network coding is very close to the theoretically most throughput. In the case of code-delay ratio increase is not very significant, the message is not very large.

Key words: multicast; network coding; throughput

随着信息技术的不断发展, 各种通信网络与人们生活各个方面结合得越来越紧密。网络服务的多样化以及针对网络传输质量要求的不断提高, 如何提高现有网络资源的利用率, 优化网络, 已成为当今网络通信研究的重要课题之一。传统的网络多播中, 网络节点只对数据分组进行路由或复制, 很难达到网络多播的最大传播容量。而且多播组上的所有接收者以相同的吞吐量接收数据。这样如果从源到特定的接收者之间的通道上有一个瓶颈链路, 则吞吐量将会被这个瓶颈链路的容量所限制。

2000年, AHLWEDE R^[1]等人提出了网络编码, 其核心思想是在网络中的节点采用不加冗余的编码。利用

网络编码, 网络节点可以把收到的多个数据包通过一定的编码手段重新组包再发送到下一节点, 接收端收到一定数量的包后通过解码获得数据。要对一个任意给定的多播网络进行网络编码, 目前已有的方法有2种: 一种是由 Ralf Koeter 等人提出的基于代数结构的网络编码方法^[2], 这种方法是一种指数时间算法。另外一种重要的网络编码方法是由 Peter Sanders 等人提出的一种多项式时间算法的网络编码方法^[3], 这种方法相对第一种方法而言不仅算法复杂度简化了, 而且有一个很大的优点, 就是在进行从源节点到各个终端节点进行传输信息之前, 先选好从源节点到各个终端节点的传输路径。因此, 在同样的信息传输速率下减小了对网络资源的占

用,同时使网络编码变得更简单。

1 网络编码概念

网络中的节点对信息比特流进行一定的操作,如模2加、“与”、“或”等,而不仅仅是对其进行复制转发,称之为网络编码。在参考文献[1]中,作者给出了一个多播网络的例子,采用网络编码时,传输速率比仅使用路由、转发时的要快,并且达到了多播网络的速率上限 $C = \min\{c_i\}$ (c_i 从信源 s 到接收节点 r_i 的最小割最大流值)。LI SYR^[4]等人随后表明采用线性的网络编码在有限大的域中能够达到速率上限 C 。图1(a)为一个单源二接收多播传输网络图,网络中边上标的是链路的容量,即1 bit/单位时间。实际网络中容量为 K bit/单位时间的链路可拆分成 K 条容量为1 bit/单位时间的并行链路,因此为简单起见,图1中的所有链路的容量都为1 bit/单位时间。

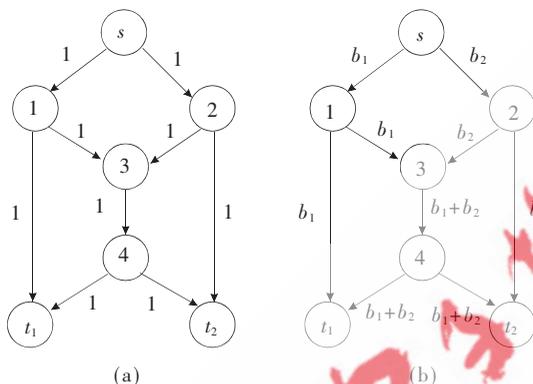


图1 单源二接收网络图

假如网络中的节点只对其收到的信息进行复制转发,则此网络多播速率无法达到2 bit/单位时间。因为接收节点3在1个单位时间内只能转发从节点1和2过来的2个bit中的1个,如果3转发从1节点过来的信息,则接收节点 t_1 ,可以收到2 bit/单位时间,但是接收节点 t_2 ,只能收到1 bit/单位时间。假设从源发向节点1的信息bit为 b_1 ,从源发向节点2的信息bit为 b_2 。图1(b)中的节点3将分别从节点1和2过来的信息 b_1 和 b_2 进行模2加,然后发向节点4,于是接收节点 t_1 在1个单位时间内收到了 b_1 和 b_1+b_2 ,于是可以通过 $b_1+(b_1+b_2)=b_2$ 运算来得到 b_2 ,也就是说,接收节点 t_1 ,在1个单位时间内相当于收到了 b_1 和 b_2 。同理可以知道接收节点 t_2 在1个单位时间内也相当于收到了 b_1 和 b_2 于是图1(b)的传送方法达到了多播速率(2 bit/单位时间)。

2 一种基于网络编码的应用层多播算法

从当前主要的基于网络编码的应用层多播算法的比较分析当中可以看到,网络编码应用到应用层多播需要知道网络的拓扑结构。现有的几种机制都是基于已知的网络拓扑结构,基于代数的构造方式提供了网络编码与应用层多播融合的最基本的模型,它引入了转移矩阵来描述输入变量与输出变量之间的关系,使得对于网络

编码是怎样应用到应用层多播这种机制有了更加清晰的概念,但是这种模型对于具体的网络拓扑没有展开。而基于多项式时间算法,它运用最小割最大流算法构造网络拓扑,使得这种拓扑结构能更加有效地传输编码信息。然而,它要求拓扑结构是一个无延时的理想状态,因此,这种算法求得的网络最大吞吐量可能是理想状态的。正是基于以上分析,本文提出了一种基于随机方式的网络编码的应用层多播算法,在考虑链路花费的情况下,提高网络的吞吐量。

2.1 编码理论

源端将原始的 m 个信息流编为一组并编码成 n ($n \geq m$) 个大小相等的新信息流;中继节点对隶属同组的源端编码信息流进行重编码并转发;目的端收到足够多的编码信息流后利用解码算法恢复出原始信息流。

2.1.1 源端编码

采用随机线性码^[5-6]作为网络编码方案,按照产生的先后顺序,源端将每 m 个信息编为一组,记为 x_1, x_2, \dots, x_m ,并赋予相同的组标识,组标识从0开始递增,直到增加到某个上限后重新归零。当源端要发送该组信息时,从有限域 $F_2^{8 [7]}$ 中选取 m 个随机数作为编码系数 g_1, g_2, \dots, g_m ,并按照式(1)进行线性编码,同时将编码系数和组标识添加到信息头部。若要产生 n 个编码信息 Y 共需进行 n 次相同的编码操作, n 的大小根据网络状态决定。

$$Y_i = \sum_{j=1}^m g_{ij} X_j, i=1, 2, \dots, n \quad (1)$$

2.1.2 中间节点重编码

中间节点对一定时间间隔内接收的编码报文进行存储,并对具有相同组标识的编码报文进行重编码,这样可以进一步降低编码信息间的线性相关性,可提高解码成功概率。假设中间节点R收到 k 个来自编码信息 Y_1, Y_2, \dots , 每个信息 Y_i 对应的编码系数为 $g_{i1}, g_{i2}, \dots, g_{im}$, 其中 $i=1, 2, \dots, k$ 。则中间节点R按照式(2)和式(3)重新产生 k 个新的编码信息及编码系数 $h_{11}, h_{12}, \dots, h_{1k}$ 并继续转发,其中 h 是针对从有限域中产生的新的编码向量 k_{il} 与原始向量 g_{ij} 的内积。

$$h_{ji} = \sum_{l=1}^k k_{jl} g_{li} \quad (2)$$

$$Y_l' = \sum_{j=1}^m h_{jl} x_j, l=1, 2, \dots, k \quad (3)$$

2.1.3 目的端解码

当目的端接收到 m (或大于 m) 组编码数据,就可以采用矩阵转换的方式恢复出原始的 m 个信息。假设接收节点接收到的 m 组数据分别是 Y_1, Y_2, \dots, Y_m , 则接收节点进一步判断这 m 组数据的编码系数 $g_{i1}, g_{i2}, \dots, g_{im}, i=1, 2, \dots, m$ 的线性相关性,若这 m 组编码系数组成的 $m \times$

m 维矩阵满秩, 则可通过公式(4)恢复出原始的 m 个信息。当目的端接收的编码数据小于 m 时, 可通过消息反馈机制通知上游节点对缓存的同组编码数据进行重编码操作并转发, 直至目的端能恢复出 m 个原始信息为止。

$$\begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_m \end{bmatrix} = \begin{bmatrix} g_{11} & \cdots & g_{1m} \\ \vdots & & \vdots \\ g_{m1} & \cdots & g_{mm} \end{bmatrix}^{-1} \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_m \end{bmatrix} \quad (4)$$

2.2 网络拓扑的构造

本文考虑每个结点的最小花费, 采用最小费用最大流方法来构造传输拓扑网。最小费用最大流的基本思想: 对于通信网络 $G(V, E)$, 节点 $s, t \in V$, 从 s 到 t 的最大流 $\max flow(s, t)$ 。设有链路 $(i, j) \in E$, 链路容量为 b_{ij} , c_{ij} 是通过链路 (i, j) 传输 1 个单位信息流的费用, 求从 i 到 j 的最大流 f , 使得流量的总费用 $C(f)$ 为最小, 即:

$$\min C(f) = \sum_{(i,j) \in E} c_{ij} f_{ij} \quad (5)$$

最小费用最大流的求解原理综合了求最大流的原理和求最短路径的原理, 主要依据为: 若 f 是流值为 W 的所有可行流中费用最小者, 而 P 是关于 f 的所有可扩充链中费用最小的可扩充链, 则沿 P 以 ε 调整 f 得到的可行流 f' 是流值为 $W + \varepsilon$ 的所有可行流中费用最小的可行流。如果已知 f 是流值为 W 的最小费用流, 关键是要求出关于 f 的最小费用的可扩充链。所以本算法在原网络图 G 的基础上构造一个新的赋权有向图 $G'(V, E')$, 使其顶点与 G 的相同, 并将 G 中的每条弧变成两个相反方向的弧。在网络 G' 中寻求可行流 f 的最小费用可扩充链, 即找到节点 i 到 j 的最短路。算法描述:

(1) 链路 $(i, j) \in E' f_{ij}^k = 0, k=0$

(2) f_{ij}^k 为经过 k 次调整得到的最小费用流, 构造有向图 $G'(V, E')$, G' 中各条弧的权值与 f 的关系

$$w_{ij} = \begin{cases} c_{ij} & f_{ij} < b_{ij} \\ \infty & f_{ij} = b_{ij} \\ -c_{ij} & f_{ij} > 0 \\ \infty & f_{ij} = 0 \end{cases}$$

(3) 在 $G'(V, E')$ 中应用 Floyd 算法查找最短路径, 若不存在, f_{ij}^k 就是最小费用最大流, 算法终止; 若存在, 则此最短路即为原网络中相应可扩充链。

(4) 在 P 上对 f_{ij}^k 根据式(6)进行调整, 构造新的可行流 f_{ij}^{k+1} , 返回(2)。

$$\begin{cases} f_{ij}^{k+1} = f_{ij}^k + \varepsilon & p \text{ 上的前向弧} \\ f_{ij}^{k+1} = f_{ij}^k - \varepsilon & p \text{ 上的后向弧} \\ f_{ij}^{k+1} = f_{ij}^k & \text{不在 } p \text{ 上的弧} \end{cases} \quad (6)$$

3 性能评价

在本实验中采用最小费用最大流方法来构造传输拓扑网, 生成由 80 个节点组成的拓扑结构。每个链路的带宽范围 (1 MB~2 MB)。交换应用层数据信息的大小 (1 KB~35 KB)。网络编码采用有限域 F_2^8 范围内的随机线性码, 源端和中间节点进行编码和重编码操作, 目的端进行解码操作。所有的仿真实验在扩展的 ns-2 平台上进行, 仿真时间为 500 s。

主要从吞吐量和编码延迟率两方面考虑该算法的性能。编码延迟率是所有节点的编码时间总和与端到端的延迟的比率。不同的信息块的大小对编码延迟率的影响也不同, 信息块大小分别是 1 KB、10 KB、15 KB。如图 2 所示, 信息块越大, 每个节点的编码时间越高, 编码延迟率也相应地偏高, 但增加的幅度是有限的, 平稳的。网络编码的目的是为了达到网络的最大吞吐量。如图 3 所示, 与基于多项式时间的网络编码的算法比较, 二者都比较接近最大吞吐量, 但最小费用最大流的吞吐量要小于基于多项式算法达到的吞吐量, 因为本方案计算了链路的花费, 增大了延迟。但通过实验证明了网络编码确实能够提高网络的吞吐量。

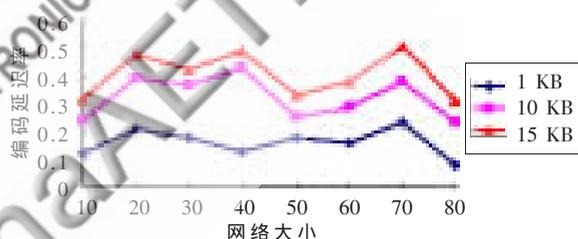


图2 不同信息块编码的延迟率

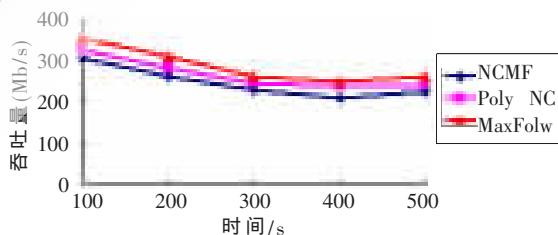


图3 最大吞吐量的比较

网络编码的提出从本质上打破了通信网络中传统的信息处理方式, 目前已是通信网络研究领域中的一个新的研究热点。本文提出一种新的基于随机方式的网络编码的应用层多播算法, 在计算网络拓扑考虑了链路的花费, 源端和中间节点使用随机线性编码方法, 在目的端进行解码操作使得目的端能从乱序的信息和部分丢失的信息中恢复出原始数据, 提高了网络的可靠性。通过对 ns-2 的扩展并进行仿真实验, 结果证明了本文提出的算法可以提高网络的吞吐量, 与网络中最大的吞吐量比较接近。在信息块不是很大的情况下, 编码延迟率

的增长是在一定的范围内的。

参考文献

- [1] AHLWEDE R, CAI N, LI S Y R, et al. Network information flow[J]. IEEE Transactions on Information Theory, 2000, 46:1204-1216.
- [2] KOETER R, MEDARD M. Beyond Routing: An algebraic approach to network coding[C]. 2002 IEEE INFOCOM, 2002.
- [3] SANDERS P, EGNER S, TOLHUIZEN L. Polynomial time algorithms for network information flow [C]. In 15th ACM Symposium on Parallel Algorithms and Architectures, 2003: 286-294.
- [4] LI S Y R, YEUNG R W, CAL N. Linear network coding [J]. IEEE Transactions on Information Theory, 2003, 49: 371-381.
- [5] HO T, KARGER D R, MEDARD M, et al. The benefits of coding over routing in a randomized setting[C]. IEEE International Symposium on Information Theory-Proceedings, 2003:442.
- [6] 杨林.一种集成网络编码的低轨卫星网络多径路由方法[J].中南大学学报(自然科学版),2007,38(5):950-955.
- [7] WANG Dan, ZHANG Qian, LIU Jiang Chuan. Partial network coding: theory and application for continuous sensor data collection [C]. 2006 14th International Workshop on Quality of Service(IEEE Cat. No. 06EX1425), 2006:93-101.

(收稿日期:2009-04-08)

电子技术应用
APPLICATION OF ELECTRONIC TECHNIQUE
www.chinaAET.com