Network Coding and Multimedia Content Delivery

Dah Ming Chiu (IEEE Fellow), the Chinese University of Hong Kong
dmchiu@ie.cuhk.edu.hk

1. CDN and P2P

Multimedia content delivery is projected to be by far the biggest bandwidth consumer of the Internet. For many years, the mechanism for content delivery envisioned by the networking community is network multicast. It is arguably the most efficient. By having the network routers provide the duplication of content and distribute it along a tree spanning all the nodes wanting to receive the content, the least amount of bandwidth can be expended. But multicast has some major limitations. (1) Although it economizes on bandwidth usage, it may not provide the necessary bandwidth to all users. To accommodate users' needs better, multiple distribution trees may be needed, and the determination of the most suitable trees depends on topology and other policy parameters – it is a combinatorial optimization problem too challenging for routers to solve. (2) The multicast service, by requiring a group of users to receive the content simultaneously, is too restrictive. Due to these limitations, it plays only a limited role as a kind of TV services in carefully provisioned edge networks, referred to as IPTV [1].

A more flexible service for users is Video-on-Demand (VoD). The most primitive VoD is based on a client-server solution, which does not scale as the number of users grows. To deal with the scalability problem, Content Delivery Network (CDN) is created. Users are redirected from the main server to servers less busy, or closer to them. CDN plays an extremely important role in providing content delivery in today's Internet, as a large percentage of web-based content are delivered through CDNs [2]. The drawback is that it does not dynamically adapt to user needs, so careful planning and management is needed.

In the last few years, Peer-to-Peer (P2P) systems for content delivery have been developed and been proven to work quite well. BitTorrent (BT) [3] is the most well-known (and perhaps the first) P2P system for file downloading. Coolstreaming [4], perhaps the first to demonstrate the feasibility of P2P streaming, is the work of an MPhil student in my department. By new, many P2P systems using slightly different algorithms have been deployed. It is reported that P2P traffic has become the dominant bandwidth consumer in many networks.

The most distinct feature of BT-like P2P systems is that they divide the content into multiple sub-streams, or if the content is a file then it is divided into multiple chunks. These different sub-streams or chunks will then follow different distribution trees to reach all the receivers, simultaneously. BT and many systems form these distribution trees adaptively and on-demand, depending on who has what, and who is available and willing to serve. In fact, [3] likens the BT mechanism of matching up the providers and receivers (of chunks) with the mechanisms of bringing together sellers and buyers in an economic system. The problem of setting up these multiple distribution trees and allocating bandwidths to different peers is of the same nature, and more complicated than the combinatorial problem network multicast has to solve; the difference is that the peers, not situated at the network cross-roads like routers, are in a better position to handle it.

Although P2P systems have often been used to distribute content violating copy-right protection, this technology is indeed a new paradigm for network content delivery. I like to compare it to the advent of *packet switching*. In the early days of contemporary data networks, the prevailing idea was based on circuit switching – emulating the telephone network service that had been in service for many years. It was the seemingly chaotic packet switching that helped the development of a whole generation of new applications with more bursty traffic for relatively limited bandwidth, including emails and the web. For content distribution networks, multicast is like virtual circuits, emulating the broadcast TV service that have existed for many years, and efficient. But it is the adaptive, multi-path P2P technology that is more likely to satisfy the on-demand, high definition/demand nature of future content distribution needs.

2. The Promise of Network Coding

Network coding is another idea that has clearly caught people's fancy. In less than ten years, more than 1800 papers have been published on the subject. Network coding was invented in my department by Raymond Yeung (who works on Information Theory) and Bob Li (who works on switching theory), among others helping them [5]. Claud Shannon's Information Theory elegantly characterized the capacity of a communication channel. Network Coding is the counterpart of Information Theory applied to a network. It addresses a very important question for network content distribution, albeit in a highly abstract and theoretical manner; that is, what the capacity (or maximum throughput) of a network for distributing content to a set of users is, and how this capacity can be achieved.

It is well-established that the maximum throughput cannot exceed the min-cut between the source of the content and all the receivers. However, it is easy to construct an example to show that no matter how well-informed and smart you are, there are networks for which you cannot arrange the routing and scheduling to achieve this theoretical throughput. The theory of network coding proves that you can always achieve this theoretical throughput by making the intermediate nodes apply some coding (e.g. construct linear combinations of different pieces of the original content). The most often used example is the butterfly network, explained in many papers on network coding including a paper I co-authored several years ago [6].

3. The Application of Network Coding

For evaluating P2P content distribution systems, as explained in [6], the most useful model is the *uplink sharing model*, the term originally coined in [7]. This model assumes the network is not the bottleneck in content distribution. Rather, the bottleneck is the total uplink available at the edge – the content server and the peer nodes (including cache servers). For today's Internet, this model more or less reflects the reality. Based on this model, it is possible to determine the capacity (maximum throughput) for P2P content distribution, assuming we have perfect routing and scheduling, as explained in [6]. The result can be expressed as a simple closed form equation. This capacity, it is shown, happens to correspond to the min-cut from the content server to the set of given receivers. This result is a negative result, as it implies network coding does not help to increase the capacity for networks satisfying the uploading sharing model conditions.

Does this mean network coding will remain a theoretical curiosity only in its application to network content distribution? Not necessarily so. Although network coding may not increase the maximum throughput in common network scenarios, it may be still be quite helpful. It turns out that achieving the maximum throughput in a P2P network is not an easy feat. Peers must exchange lots of information about what they each have, and carefully schedule how they serve others. Random Network Coding, a special algorithm to randomly mix different chunks of a file, can greatly ease the maneuvers needed to maximize throughput. This is because by spreading randomly mixed chunks around, it helps to increase the likelihood that peers have the content others need, hence making the scheduling job easier. This may prove to help achieving higher throughput by minimizing routing overheads and scheduling complexity. A prototype of such a system (called Avalanche) was built by Microsoft Research a few years ago, and the reported results were interesting [8]. The Random Network Coding scheme can also be helpful for networks not satisfying the uplink sharing model. When there are bottlenecks inside the network (not the uplinks), it is not obvious how to program the network to apply network coding targeting the bottlenecks. It is suggested that Random Network Coding can automatically target network bottlenecks to help maximize throughput [9]

4. Conclusions

Network content distribution, both in streaming or non-streaming forms, is envisioned to be the most important types of network activities. CDN and P2P types of platforms will enable many new applications to allow different ways of generating/publishing contents. Network coding is an intriguing new idea for helping to realize these systems. There are many challenges, and many research opportunities.

A new Institute of Network Coding (INC) will be formed at CUHK, located in my department, with the original inventors as directors. I use this occasion to wish this effort to be a great success.

References

[1] IPTV, http://en.wikipedia.org/wiki/IPTV.

[2] CDN, http://www.akamai.com/

[3] Bram Cohen, "Incentives Build Robustness in BitTorrent", 2003

[4] X Zhang, J Liu, B Li and TS Peter Yum, "Coolstreaming/DONet: A Data-driven Overlay Network for Efficient Live Media Streaming", IEEE Infocom, March 2005.

[5] S. Li, R. Yeung and N. Cai, "Linear Network Coding", in IEEE Transactions on Information Theory, Vol 29, No 2, pp 371-381, 2003.

[6] Dah Ming Chiu, Raymond W Yeung, Jiaqing Huang, and Bin Fan, "Can Network Coding Help in P2P Networks" Invited paper in Second Workshop of Network Coding, Boston, MA, USA, April 2006.

[7] Jochen Mundinger, Richard Weber and G Weiss, "Optimal Scheduling of Peer-to-Peer File Dissemination", Journal of Scheduling, Volume 11, Issue 2, 2008.

[8] Christos Gkantsidis and Pablo Rodriguez, "Network Coding for Large Scale Content Distribution", in IEEE Infocom, March 2005

[9] S. Acedariski, S. Deb, M Medard, and R. Koetter, "How Good is Random Linear Network Coding Based Distributed Network Storage?", First Workshop of Network Coding, 2005.

Dah Ming Chiu is the chairman and a professor in the Department of Information Engineering, of the Chinese University of Hong Kong. Prior to joining CUHK, he has worked for Sun Labs, DEC, and AT&T Bell Labs. He received his first degree from Imperial College London, and his PhD degree from Harvard University. He is an IEEE Fellow, for his contribution to the resource allocation algorithms for the Internet. He is currently serving as an associate editor for the IEEE/ACM Transactions on Networking