A FC/AL-Based P2P Network for Personal Archive and Sharing

Eunsam Kim(*), Jonathan C.L. Liu(**), and SungWook Chung(**)

(*)Department of Computer Engineering, Hongik University, Seoul, South Korea
(**) Department of Computer, Information Science and Engineering, University of Florida, Gainesville, Florida, USA

Abstract

Peer-to-Peer (P2P) networking technology can share a variety of contents, including the broadcast TV/Film programs. With the popularity of personal video recorders, people stores their favorite TV/film titles in the hard drive. It is clear that these recorders can easily archive and share TV/film contents over P2P networks once they are equipped with broadband connections. This is important because this emerging P2P network can provide an alternative content archive and distribution channel to the conventional unidirectional broadcasting. In other words, people can have another way to obtain TV contents that they were not able to record at the time of broadcast or purchase via the conventional DVD sale. As part of on-going research, we demonstrate that Fibre Channel -Arbitration Loop (FC-AL) technology can serve well as the storage network for personal archive and real-time content distribution. In our proposed architecture for TV content archive and sharing, the FC-AL directly connects many digital video recorders, a pool of network disks, and a content managing server.

Via our integrated storage/network architecture, users can share their archives either from their own digital video recorders or the pool of networked disks. The results are the significant time-shifting for the users to watch the programs. The proposed network architecture also relieves the storage limitation of the digital video recorders, and users can help each other by archiving the "hard-to-find" contents. In this paper, we present the technical schemes to fully utilize the performance of this emerging network. These schemes include (1) Buffering schemes to guarantee the concurrent video streams; (2) Matching schemes for communication pairs; (3) Load- balancing schemes; and (4) multiple-loop staging schemes to extend the time-shift. With the effective buffering schemes, we have demonstrated over 300 high-quality video streams can be supported by the system with less than 1display jitters.

To maximally utilize the FC-AL bandwidth and full-duplex property in both directions, we have developed effective schemes to match the communication pairs between users and the pool of network disks. It is also a common situation that most playback requests are concentrated on a few popular programs. From the perspective of overall system performance, such a skewed access pattern may cause serious load-unbalancing problems because some overloaded PVRs can become performance bottlenecks even though there are still many disks having sufficient available bandwidth. To solve this problem, we switch playback requests as well as replicate popular program (as minimally as possible). Finally, efficient staging methods have been proposed between the networked disks in different FC-AL loops to successfully extend the time-shift from 3-day to 70 days.

Introduction

Digital video compression technology has become matured in the last two decades (roughly from late 1980s [6] to today). Due to the invention of MPEG (Motion Picture Expert Group) standards such like MPEG-2/4/7/21, users can fully utilize the consumer electronic devices to receive and store the high-quality video titles. Typical examples are HDTV TV, DVD player/recorder and DVR (Digital Video Recorder) on the hard drive like TiVo. Thus, like recording the broadcast program on VHS tapes, today's users can simply "archive" the digital video (including broadcasted movies and TV programs) on the hard drive of their DVRs. As long as the users choose not to delete or over-write the content, some digital video titles can be permanently stored with the hard drive life period. As this behavior proceeds for years, there will be an implicit collection of many digital video titles stored in the hard drives of users' DVRs. Some popular video titles will be perhaps stored with multiple copies in different locations.

Just like VHS tapes with recorded video content can be shared with friends, the "implicitly-archived" digital video on the hard drive also can be shared among the peer friends. Instead of physically sending the tapes or DVDs, digital video content can be shared simply by the on-the-fly transmission. At this moment, the DVR on the market only has limited transmission capability (mainly for downloading the program details). It will be the proved trend that eventually the DVR will be equipped with high-speed networking interfaces. We thus toss the next-generation DVRs as "Personal Video Recorders" (PVR) [3] for the rest of the paper. With the enabled transmission capability, a truly peer-to-peer (P2P) archive-and-share system can be built in the near future. The reasons are because (1) The hard drive space in the PVR will be limited (usually not exceeding 200 GB) and (2) Users simply can not archive ALL the video files throughout the users' life durations. Therefore, if the archive-and-share systems can assure that there will be always copies of the same content available in the system, the local copy within a user's PVR can be simply replaced by new content.

The direct benefit to have the peer-to-peer archive-and-share system is the convenience of sharing. Users do not need to bring or send the DVD or VHS copies to their friends. The delivery is replaced by network transmission. Because of the availability of the multiple copies, the system also provides the time-shift for users to (re-)view the content in the future. Ideally, when the time-shift reaches to the infinity, then the goal of permanent archive is accomplished. Another implicit benefit is the availability of the titles that are hard to find. In addition to the popular titles, it is expected that some "hard-to-find" titles will be copied into some-one's PVR due to the personal interest. We believe this P2P system should be built from inside-out. It should be attempted within the community first. The idea community will be the high-rising apartment building, which has been popular in urban cities. Since the users are clustered in the same building, fiber-optic connections can be activated to support the communication between the PVRs. Then the system can be extended to city-wide and beyond.

This paper presents the novels system design within this emerging P2P archive-and-share system, which has been investigated by us since 2003 [4]. The architecture is integrated with storage devices and PVRs with the Fibre Channel - Arbitration Loop (FC-AL) technology [1]. Specifically, we present the effective buffering scheme which integrates prefetching and caching for supporting concurrent video streams. We call the schemes IPC (Integrated Prefetching and Caching). It is a new line of work after we successfully bring in the timing consideration into the prefetching schemes for supporting concurrent video streams [5]. The IPC dynamically determines threshold intervals so that the system can decide to either prefetch or cache each new stream. By measuring the caching effect in terms of the cache size, we show that the cache size and disk bandwidth can be traded for each other. To maximize utilization of both cache space and disk bandwidth by complementing one another when either of them is exhausted, IPC reduces either the required cache size or disk bandwidth by adjusting the number of immediate blocks and prefetched blocks.

We then introduce the matching schemes to take advantage of the spatial re-use feature provided by FC-AL technology. FC-AL ideally can double the network bandwidth if perfect matchings of communication pairs are all formed (i.e., all the PVRs and network disks send and receive data between each other at the same time). However, since the request pattern of program transmission to another PVR or network disks is very dynamic in a P2P networking environment, it is necessary to develop a sophisticated scheme to match a communication pair to further utilize the network bandwidth in both directions. The idea of matching scheme roots from the heuristic that the requests on the popular contents can be matched by the existing streams. We have derived the schemes into direct-matching and indirect-matching by allowing the system to generate more matching candidates. The results indicate that matching communication pairs through indirect searching is much more effective than simple direct matching since indirect searching can significantly extend the number of matching candidates.

Some video titles are more popular than the others. Therefore, it can be a common situation that most playback requests are concentrated on a few popular programs. From the perspective of overall system performance, such a skewed access pattern may cause serious load-unbalancing problems because some overloaded PVRs can become performance bottlenecks even though there are still many disks having sufficient available bandwidth. Our unique approach is to combine with switching and replication. To solve this problem, we switch playback requests as well as replicate popular program. To reduce replications to as few as necessary since replications require additional disk and FC-AL bandwidth, we make disk bandwidth available for new requests issued to overloaded system components by switching the transmission responsibilities for existing requests to less loaded system components. The goal is to make room for the new request by finding another system component to take over one of the existing requests. Note that for the switching the least loaded component is always chosen so that the load of each component can be maintained at a similar degree. The replication is not performed until switching becomes impossible due to requests that exceed the total processing capability of the entire system. Our experimental results show that the number of switchings was on average about 12 times greater than that of replications. The reason is that switching and replication work together to reduce as many replications as possible by achieving load-balancing among PVRs.

Although the single-loop architecture can serve and share HD-quality programs between attached PVRs, the single-loop system has the limitation in that it can only support a maximum of 127 devices in the loop. Therefore, in order to build a larger system, it needs to be extended to scalable multiple-loop architecture. Extending the single-loop system to the multiple-loop systems is not a simple task. To communicate between the devices in the different loops, a bridging platform (we call it ”shared disk”) needs to be designed to relay the traffic. Via our intensive study on the feasible topology designs, our system with multiple loops accomplishes the goal for providing extensive time-shift with close-to-optimal inter-loop traffic amount. The experimental results reveal that it is possible that system can hold the time-shift as long as nearly 70 days. We believe, with the gradual deployment of fiber optics to reach more users, our proposed P2P archive-and-share will eventually extend beyond the small/medium communities (as the high-rising apartment buildings). It is expected that our proposed schemes can be widely adopted to further improve the system capability.

Buffering Schemes to Guarantee the Concurrent Video Streams

To provide a huge number of concurrent users within the system, it is essential for the system to support sufficient I/O bandwidth for delivering data to clients on time. Disk access time required for reading one disk block involves seek time, rotational latency, and data transfer time. With prefetching, data blocks can be read ahead into the cache before they are actually requested. Thus, prefetching can reduce seek time and rotational latency by reading together several blocks contiguously stored on a disk at a time. Moreover, caching can eliminate the whole disk access time, including data transfer time, by reusing blocks already kept in memory instead of reading a block from a disk.

We have observed that the performance of either prefetching or caching alone cannot continue to increase proportionally to the dedicated cache size. The performance of prefetching becomes saturated at some point when increasing the cache memory size, even though prefetching can increase the number of concurrent streams significantly before that point. The reason is that, as the number of blocks to be prefetched per stream increases with a larger cache memory, the cache size required to hold the blocks until they are transmitted to networks increases even more rapidly. The performance of caching also depends largely on the interval lengths between successive streams for the same video title. As a larger cache memory is employed, the average length of intervals should also increase because larger intervals are additionally cached. Thus, to utilize the cache space more efficiently, it is better to prefetch the blocks that are too long to be cached. Accordingly, prefetching too many blocks or caching too long intervals might quickly exhaust the available cache space that would be otherwise used to support more concurrent streams with the other technique. Therefore, if careful consideration is not given to prefetch or to cache incoming streams, either of the two techniques can decrease the effect of the other, which cannot maximize the number of concurrent streams.

We have proposed an effective scheme, called Integrated Prefetching/Caching (IPC), to simultaneously take advantage of both prefetching and caching when dealing with streaming video in the archive-and-share P2P system. The IPC dynamically determines threshold intervals so that the system can decide to either prefetch or cache each new stream. To derive the threshold intervals, we first analyze how many cache blocks are required to prefetch one additional stream based on the current numbers of prefetched blocks and immediate blocks (i.e., non-prefetched blocks that require immediate transmission to the network within the same cycle when they are read from disks). We also consider another important factor for determining the threshold values, i.e., the caching effect of additionally saving disk data transfer time compared to prefetching. By measuring the caching effect in terms of the cache size, we show that the cache size and disk bandwidth can be traded for each other. The dynamic threshold values enable IPC to efficiently utilize the cache space depending on the lengths of the incoming intervals. If the intervals shorten, the number of cached blocks increases. Otherwise, the number of prefetched blocks increases.

To maximize utilization of both cache space and disk bandwidth by complementing one another when either of them is exhausted, IPC reduces either the required cache size or disk bandwidth by adjusting the number of immediate blocks and prefetched blocks. As a result, IPC can continue to increase the number of concurrent streams until both system resources are used up. By intensive experiments, we show that IPC on average supports 14.1, 35.2, and 80.4% more concurrent users compared to three other cases: prefetching only, interval caching only [2], and no buffering technique used, respectively. As cache size increases, IPC neither saturates the overall performance nor rapidly increases the average cached interval lengths. In other words, IPC continues to increase the number of concurrent streams proportionally to the dedicated system resources. This is possible because it can offer all the benefits of both prefetching and caching.

Matching Schemes for Communication Pairs

It is clear that PVRs can easily share TV contents over P2P networks once they are equipped with broadband connections. This is important because the P2P networks can provide an alternative content distribution channel to unidirectional broadcasting. In other words, people can have another way to obtain TV contents that they were not able to record at the time of broadcast. One of important advantages of the FC-AL is that data can be transmitted in full-duplex mode. To maximally utilize the FC-AL bandwidth in both directions, we have developed a scheme to match a communication pair for spatial reuse. This implies that the FC-AL ideally can double the network bandwidth if perfect matchings of communication pairs are all formed (i.e., all the PVRs and network disks send and receive data between each other at the same time). However, since the request pattern of program transmission to another PVR or network disks is very dynamic in a P2P networking environment, it is necessary to develop a sophisticated scheme to match a communication pair to further utilize the network bandwidth in both directions.

The idea of matching scheme roots from the heuristic that the requests on the popular contents can be matched by the existing streams. Therefore, when the system receives a program transmission request from a system component, it tries to find another component to best match with it as a communication pair by investigating what programs each component is storing and transmitting. Each system component can also issue the matching request when the existing full-duplex transmission should be changed into half-duplex transmission due to the termination of one program having being played back by the other component of the communication pair. The first step is to seek out the direct matching that there exists at least one component to communicate. However, if the first step fails, then the system needs to find the matching pairs indirectly. After choosing one component, the original PVR takes over the transmission responsibility of the program to the chosen component in place of the other components, If both of the above steps fail, then the system is not able to make any match with pending request. Thus, the system needs to choose one component from the network disks. The decision will be made to have the streams from a disk with the most available disk bandwidth. Even though in this case we are not able to further utilize network bandwidth, we can achieve the effect of disk load balancing among components.

We performed extensive simulation experiments with different system parameters. Unlike the IPC schemes proposed in the previous section, matching scheme needs to be jointly evaluated with user's behavior. Since there is no large-scale trials actually reflect the system we propose (i.e., the P2P sharing network with FC-AL), it is very difficult to perform the actual experiments. Therefore, we need assume some reasonable assumptions for the system. For instance, The arrival rate of program playback requests follows a Poisson distribution reflecting varying average values. Each request can be accepted as long as the corresponding resources are available; otherwise it will be rejected. The obtained simulation results are based on the assumption that the tolerable rejection ratio is 5%. It is also assumed that 100 PVRs are connected to an internal FC-AL network and half of them on average are available at a specific time. The number of channels provided is 50.

We also assume the FC-AL has a data transfer rate of 400 MB/s, which at this time is available. The per-node delay of the interface to forward a frame is 6 words (i.e., 240 nanoseconds). The propagation delay and maximum distance between two devices are 4 nanoseconds/meter and 200 meters, respectively. The disk model is based on the IBM DeskStar 120GXP model whose capacity is 120 Gbytes. Its average data transfer rate, seek time, and rotational latency are 35.2 MB/s, 8.8 ms, and 4.17 ms, respectively. We then conduct the simulations for 24 hours (i.e., assuming the daily programs remain consistently similar) while varying several parameters, such as the number of disks, the arrival rate of requests, and the disk block size, to see the impact of each parameter on system performance. The completed simulation results indicate that the matching scheme can support many more requests when the FC-AL utilization is equal. For example, in a case where FC-AL utilization is 86%, 233 requests are supported when matching is not used while the matching scheme extends the number of supported requests to 323, which results in about a 39% performance improvement. This is because the bandwidth of both channels of the FC-AL can be further utilized by matching as many communication pairs as possible. It can also be seen that the data portion transmitted in full-duplex mode continues to increase as the number of transmission requests increases. When the number of requests is 20, the data portion is 14.5%, but it goes up to 62.1% for 275 requests. The reason for this is that with increased transmission requests, the probability of finding communication pairs increases accordingly.

By looking inside the system internal, the results also indicate that matching communication pairs through indirect searching is much more effective than simple direct matching since indirect searching can significantly extend the number of matching candidates. The results seem to be consistent with users' behavior in the modern society. For instance, when you ask a specific VHS/DVD copy of a given title, it is less likely that your direct friends have that particular copy. However, via friends' asking their friends, the probability to locate the specific title is greatly increased. However, the major difference is the networking system is performing the similar task automatically via the proposed P2P system.

Load-Balancing Schemes

It is the human nature that prefer some video titles more than others. We usually refer it as "popularity", and can be modeled as skewed access pattern (i.e., Zipf distribution). Therefor, it can be a common situation that most playback requests are concentrated on a few popular programs. From the perspective of overall system performance, such a skewed access pattern may cause serious load-unbalancing problems because some overloaded PVRs can become performance bottlenecks even though there are still many disks having sufficient available bandwidth. To avoid such a load-unbalancing problem among the system components, many efforts have been made in the past either to distribute segmented blocks on the disks with striping or to replicate popular programs. However, those load-distribution techniques cannot be applied directly to our architecture where each PVR includes only a very small number of disks and it is under each user's control to decide which programs to store. Thus, it is not possible for the system to efficiently place data over all the PVRs without permission. This implies that we need a dynamic load-distribution scheme that is suitable for our architecture.

Our unique approach is to combine with switching and replication. To solve this problem, we switch playback requests as well as replicate popular program. To reduce replications to as few as necessary since replications require additional disk and FC-AL bandwidth, we make disk bandwidth available for new requests issued to overloaded system components by switching the transmission responsibilities for existing requests to less loaded system components. The goal is to make room for the new request by finding another system component to take over one of the existing requests. Note that for the switching the least loaded component is always chosen so that the load of each component can be maintained at a similar degree. The replication is not performed until switching becomes impossible due to requests that exceed the total processing capability of the entire system. Furthermore, when the replication works together with the switching, it also enhances the load distribution of the system. Once a program is replicated, there are more chances that network disks can handle all future requests for the replicated program since they usually have more disk bandwidth than PVRs. For the same reason, network disks also tend to take over the existing requests for the replicated program through the switching scheme so that PVRs can accept the requests for other programs. We employ a dynamic replication policy similarly as [7] to reflect such P2P sharing characteristics. However, an important issue is to determine the trigger condition that when the replication should take place. We have explored the trigger condition, but choose not to present it in this paper due to the space limitation. Readers who are interested in the details of this trigger condition should contact the authors directly.

The experimental results reveal that each PVR can issue about 18% more disk access requests with both switching and replication schemes than with the replication scheme only. For example, when six network disks are used, the supported disk access request rate per PVR is increased to 10.1 from 8.56. Since there are 50 PVRs in the network, the total number of requests supported increases by 77. This demonstrates that the effect of switching playback requests is considerable. Another results also indicate that average number of replications per hour was less than 0.1 for both switching and replication schemes, while it was 9.2 for replication scheme only. This means that for this request rate we can achieve load balancing of playback requests among PVRs with only a switching scheme since the replications are rarely performed. This is because the switching scheme by itself is able to distribute the loads among PVRs quite evenly by choosing the least loaded PVR as a target for switching. Accordingly, the FC-AL bandwidth that can be saved is proportional to the total playback rate for the reduced number of replications, and the number of network disks required to store the replicated programs also decreases to a similar degree.

Our experimental results also show that the number of switchings was on average about 12 times greater than that of replications. The reason is that switching and replication work together in the following way: to reduce as many replications as possible by achieving load-balancing among PVRs, the switchings are performed whenever the total number of requests for each program exceeds its threshold. On the other hand, most of replications are not started until the total disk access requests approach the capacity of the entire system since the switchings have kept them distributed quite evenly among PVRs. Once a program is replicated, network disks handle future requests for the replicated program since their available disk bandwidth tends to be greatest compared to the individual PVRs. As a result, the PVRs storing the replicated program can accept other programs with the available bandwidth that should otherwise be used for the requests for the replicated program. Therefore, we can see that the replication performs much better when combined with the switching than when it works alone.

Mutilple-loop Staging Schemes

Another benefit of adopting the "archive-and-share" P2P system is the extensive time-shift that it can provide. Since the program access frequency distribution is usually skewed to the popular programs, it is clear that we can obtain better performance in terms of overall program hit ratio simply by storing more popular programs. Based on the skewed access frequency distribution, we have determined how many hours of time-shifting the pool of network disks can provide depending on the desired hit ratio while varying the number of disks from 1 to 30. As the preliminary experiments based on a single-loop FC-AL system, we have first found out that 55, 65, 75, 85, and 95% hit ratio can be obtained by storing about 23, 35, 49, 66, 87%, respectively. In addition, as the number of disks increased in each hit ratio case, the time-shifting hours were extended as expected. For example, the pool of network disks could hold the most popular 23% of the programs for about 59 hours with 30 disks (i.e., in the 55% hit ratio case), while they could hold them for only about 2 hours with one disk. It can also be seen that the time-shifting hours supported in the 55% hit ratio case are four times longer than those in the 95% hit ratio case even though the difference of hit ratio is only 40%. The time-shifting hours increase much more rapidly (by approximately 10 times) compared to the difference of hit ratio.

Although the single-loop architecture can serve and share HD-quality programs between attached PVRs, the single-loop system has the limitation in that it can only support a maximum of 127 devices in the loop. Therefore, in order to build a larger system, it needs to be extended to scalable multiple-loop architecture. Extending the single-loop system to the multiple-loop systems is not a simple task. To communicate between the devices in the different loops, a bridging platform (we call it ”shared disk”) needs to be designed to relay the traffic. In addition, another key design factor is how to reduce the traffic among different loops. With much traffic reduced, the bridging stage can require fewer shared disks, thus providing better a degree of overall system scalability. In other words, by having the same number of loops, we should determine the criterion of which design is preferable and desirable. In our paper, we introduce the concept of scalability as the criterion, where we define the scalability as the total number of all attached devices, such as PVRs, network disks, and shared disks.

With the shared disks, we can organize the whole architectures along with several feasible topology designs. That is, we can configure our architecture with the Linear topology, the Ring topology, the Complete Graph topology, and the Edge- Added topology. In addition, to further reduce the actual needed shared disks, we also investigate the effects of different number of interfaces in a single shared disk, so that we can further decrease the total number of needed shared disk, which enables to provide more scalability. Finally, based on our proposed architecture, we demonstrate the time-shifting hours can be increased by increasing the number of loops. We observe that we can hold nearly 70 days when we store 28 programs among 50 incoming programs, which is much better in a single loop architecture case. The significant improvement is due to the obtained scalability by the scalable architecture.

Conclusion

We have presented an overview on various schemes for our proposed P2P “archive-and-share” system. The system takes advantage of high-speed networking technology such like FC-AL and the latest development of PVRs and network-attached storage devices. We specifically focused on the effective buffering schemes, matching schemes, load-balancing schemes and multiple-loop staging schemes to maximally utilize the system performance. The experimental results are quite promising for holding nearly 70 days of time-shift. We continue working on further extension for system to reach beyond the small-size/middle-size communities to the large-size communities. The collaboration schemes between the PVR/disk across different communities are critical to further expand the proposed system. The integration of FC-AL with MANs (e.g., WiMax and ATM) and WANs (e.g., ATM) will be our research focus also.

References

[1] ANSI, Fiber Channel Arbitrated Loop (FC-AL-2), 2001.

[2] A. Dan and D. Sitaram. A generalized interval caching policy for mixed interactive and long video workloads. Proc. Multimedia Networking Conf., pages 344-351, San Jose, USA, 1996.

[3] S. Jung, E. Kim, and D. Lee, Design and Implementation of an Enhanced Personal Video Recorder for DTV, IEEE Transactions on Consumer Electronics, 47(4):915-920, 2001.

[4] E. Kim and J.C.L. Liu, Design of HD-quality streaming networks for real-time content distribution, IEEE Transactions on Consumer Electronics, 52(2):392-401, 2006.

[5] E. Kim and J.C.L. Liu, Time-Aware Prefetching for On-Demand Video Services in a Residential Service Gateway, Multimedia Tools and Applications, 29(3):233-255, 2006.

[6] D. Legall, MPEG - A Video Compression Standard for Multimedia Applications, Communication of ACM, 34, 4, pp. 48-58, 1991.

[7] P.W.K. Lie, J.C.S. Lui, and L. Golubchik. Threshold-based dynamic replication in large-scale video-on-demand systems, Multimedia Tools and Applications, 11(1):35-62, 2000.

Author Biography

Eunsam Kim was born in Busan, South Korea. He received his B.S. and M.S. degrees in computer engineering from Seoul National University, Korea, in 1994 and 1996, respectively. He worked for the Digital TV Research Lab of LG Electronics in Korea from 1996 to 2002. He received the Ph.D. degree from the Department of Computer and Information Science and Engineering, University of Florida in 2005. He is currently an assistant professor with Department of Computer Engineering at Hongik University in Seoul, South Korea. His research interests include distributed multimedia systems, networked storage systems, and multimedia consumer networking and platforms. He published four journal papers and five conference papers in the above research areas.

Jonathan C.L. Liu was born in Taiwan, and formally educated with computer science/engineering BS degree from Chung-Yuan University. He then received the M.S. degree in Computer Science from University of Alabama and the Ph.D. degree in Computer Science from University of Minnesota. In 1993-1996, Dr. Liu had associated with the Distributed Multimedia Research Center(DMRC) at University of Minnesota, where he had R&D cooperations with Honeywell Technology Center (HTC) and IVI Publishing. In 1996-1999, Dr. Liu was an Assistant Professor of Computer Science with the School of Electrical Engineering and Computer Science at Washington State University. Since August 1999, Dr. Liu has been with the CISE Department at University of Florida, where he is now a tenured Associate Professor. His current research interests include high-speed wired and wireless networks, multimedia communications, parallel processing and artificial intelligence. Dr. Liu has published over 70 technical papers including 26 journal papers in the above areas. Dr. Liu is the recipient of the National Science Foundation CAREER Award.

Sung-wook Chung received his BS degree in computer science from Sogang University in Seoul, South Korea, and his MS degree in computer science from University of Florida. He is currently working to complete his PhD degree with Department of Computer, Information Science and Engineering at University of Florida.