

Network-Assisted Wireless Computing

Carri W. Chan and Nicholas Bambos
Electrical Engineering
Stanford University
Stanford, CA, USA
Email: {cwchan,bambos}@stanford.edu

Jatinder Pal Singh
Deutsche Telekom Laboratories
400 S. El Camino Real, Ste. 500
San Mateo, CA, USA
Email: jatinder.singh@telekom.de

Abstract—Multimedia applications for mobile devices are increasing and growing more sophisticated. Many of these applications require computationally intensive processing, such as image processing, source coding, feature extraction and feature matching. If all of this processing were performed on the mobile device, its limited battery supply would quickly deplete. However, if the request must be transmitted through a network and processed at a remote Application Server large delays may be incurred due to communication latency—especially if the original size of the request message is very large. In this paper, we propose the use of mid-network processing on intermediary nodes in a multi-stage tandem network. We refer to this as “Wireless Network-Assisted Computing”. Allowing for mid-network processing can alleviate some of the processing burden on the mobile device, thereby extending its lifetime. It can also reduce communication latency by reducing the amount of information transmitted along each link. Certainly “leasing” processing power from these nodes comes at a price—in some cases it is beneficial to lease, in others it is not. We examine the core tradeoff between battery usage, latency, and usage of processing power at mid-network nodes. We identify some interesting properties of the optimal processing schedule. Through numerical analysis we study these properties and tradeoffs.

I. INTRODUCTION

Mobile devices are rapidly becoming more sophisticated and ubiquitous. As a result, there has been a steady increase in mobile applications to enhance the end-user’s experience. Many of these applications can require extensive computation power to run. Typically, the computation is done either entirely at the Mobile Station and an application request can be satisfied locally; otherwise, the initial request message is transmitted uplink over (possibly multiple) network links to the Application Server which can then satisfy the request remotely. Performing these operations on a mobile device can quickly drain the limited battery resources quickly degrading the user experience. Conversely, transmitting the request and executing on a remote server can lead to large latency—especially in the case of congested networks.

In this paper, we propose the use of “Wireless Network-Assisted Computing” to help alleviate the processing burden of the requests on any one node. Instead of requiring that all computation be done either at the Mobile Station where the request originates or at the Application Server upstream, we allow for computation to be performed mid-network. Of course, a “leasing cost” is incurred for borrowing processing power from the mid-network nodes. This leasing cost can capture a fee paid to the network administrator for using the

processor. In this paper, we examine the dilemma of how to optimally utilize the processing power of the entire network to distribute the processing burden instead of concentrating it at either the source or destination. We will also study the optimal tradeoff between leasing costs, latency, and battery life.

One application for which Wireless Network-Assisted Computing could be beneficial is Mobile Augmented Reality [1], [2], [3], and [4]. In these applications, a user can interact with his surroundings via virtual information. For instance, a mobile device, such as a cell phone may be equipped with a camera. Taking a picture of an object, such as a statue in a museum, can correspond to a request for a stream of audio describing the object’s significance. In order for this request to be satisfied, computationally intensive operations such as pattern recognition, source coding, and feature matching are required. Again, if all of this computation is done at the Mobile Station, the battery will drain quickly limiting the lifetime of the device. Conversely, transmitting the large image file upstream to the Application Server may lead to high latency due to communication constraints. Utilizing Wireless Network-Assisted Computing can help extend the battery life of the mobile device and minimize the service latency.

Previous research has studied the use of Network-Assisted Computing to improve quality of service. The majority of these works has focused on cache management [5], [6], and [7]. Due to the communication limits in wireless networks, a system may idle awaiting the necessary data to complete a request. The question studied in these works is how to pre-fetch information in order to avoid idling and to reduce processing time. This work largely focuses on downlink scheduling of data over a wireless link to expediate processing times. In contrast, the data required for some applications may be so large that such cache management may be impractical. In this paper, we look at the uplink problem of transmitting a request to an Application Server so that the request can be satisfied. We concentrate on the processing that is required to satisfy the request. At each node in the network, some or all of the processing can be done. The decision a system administrator faces is how much processing to do at each node in order to minimize latency and leasing costs.

The rest of the paper proceeds as follows. In Section II we formally introduce the idea of Wireless Network-Assisted Computation. In Section III, we cast the optimization problem as a shortest path problem and use the Dynamic Programming

framework [8] to find the optimal processing policy. In Section IV, we study the core tradeoffs surrounding Wireless Network-Assisted Computing. Finally, we conclude in Section V.

II. PROBLEM FORMULATION

In many media applications, such as the museum application described in the previous section and in [3], a large database of information is required in order to serve a request. It is infeasible to store all of this information on the mobile device. Therefore, a request must be transmitted uplink to the Application Server. Once the request has been fully processed, the desired content can be streamed downlink to the requesting handheld device. There has been an extensive body of work focusing on the problem of downlink streaming of media content (see [9] and references therein). In this paper, we focus on the uplink transmission and processing of the original request.

The uplink pathway from Mobile Station (MS) to Application Server (AS) is shown in Fig. 1. A request originates at the Mobile Station. In order to locate and stream the desired content, a request message must traverse multiple mid-network nodes before arriving at the Application Server. Due to the large file sizes (video/audio streams) which the requests correspond to, it is infeasible to store them all on a memory limited mobile device. As such, they are stored in a large database at the remote Application Server and the request must be transmitted upstream in order to be satisfied. The request message must be processed (image processing, feature extraction, feature matching, etc.) before the media stream can be transmitted downstream. In current systems, all of this processing is either done at the Mobile Station or at the Application Server. The original request message can be a very large image file and transmitting it over multiple congested links to the Application Server will result in large delays. If the request were processed prior to transmission, the information needed to be transmitted may be smaller, significantly reducing the communication delay. However, limited computation power and battery resources makes it undesirable to process the entire request at the Mobile Station.

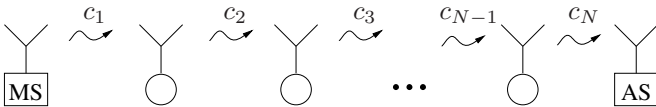


Fig. 1. System Diagram: A request originates at the Mobile Station (MS) and it transmitted over a multihop network to the Application Server (AS). Once the request has reached the Application Server and has been fully processed, it can be satisfied.

We propose to mitigate the power drain at the Mobile Station and the large communication delays by allowing some processing of the request to be performed at mid-network nodes. This removes some of the processing burden off the Mobile Station while reducing the request message, and in turn, the communication delays. Certainly, “leasing” the processing power at the mid-network nodes does not come for free, and we examine how to balance the battery life, latency, and leasing costs.

A. Request Size and Processing Model

A request consists of M stages of processing to be done. For instance, M can represent the amount of time required to fully process the request at the Mobile Station. However, because the processing power at the MS may differ from that at the AS, M is *not* the amount of time required to fully process the request at the Application Server. Therefore, M is a normalized quantity which represents the total amount of processing required to satisfy the request.

If z stages of processing have been performed, $M - z$ stages remain. At each node, n , in the network, some processing $0 \leq \delta z \leq M - z$ can be executed. The processing time required to do this is given by:

$$\tau_p(\delta z, n)$$

We make no restrictions on the functional form of τ_p other than the fact that for fixed n it is increasing in δz , such that the processing time is increasing in the amount of processing done.

As processing is completed, the request message can be reduced in size. For instance, the original image may be reduced to an image with the background extracted after some processing is done. Given that z stages of processing have been completed, the size of the request message is given by $V(z)$, which is decreasing in z and is strictly positive. The positivity is required because, even if all processing is completed ($z = M$), a small message must be transmitted to the Application Server so that it knows what content to begin streaming downlink.

B. Network Processing Model

We assume an upstream path of $N + 1$ network processing nodes in tandem. The request originates at the wireless Mobile Station and must traverse N hops to reach the Application Server. The first few hops may be wireless prior to reaching the Internet gateway. The subsequent hops may be overlay connections in the Internet spanning processing servers available for leasing by the application service provider. At minimum, there is one wireless link (e.g. between the Mobile Station and Base Station or Access Point), but there may be others over a wireless mesh, sensors, etc. Also, at minimum, there is no wireline link; for instance, the Application Server may be co-located with the Base Station or Access Point. However, in general the wireline path to the Application Server could be multihop.

Each link, n (connecting the n^{th} and $(n + 1)^{\text{st}}$ nodes), is characterized by the capacity of this link, c_n , in bits per second. Therefore, if a message with volume V bits needs to be transmitted along the n^{th} link, it requires V/c_n seconds. Hence, the latency incurred on the n^{th} link after z stages of processing has been performed is:

$$\tau_c(z, n) = \frac{V(z)}{c_n}$$

It is easy to see that τ_c is decreasing in c_n as the link becomes less congested. It is also decreasing in z since $V(z)$

is decreasing in z .

C. Leasing Model

Leasing processing power from mid-network nodes can be extremely beneficial to reduce latency. However, it comes with a cost. These costs can capture the fee required to lease CPU power, the processing burden on mid-network nodes, as well as potential security risks by giving access of client data to these nodes. We represent these leasing costs by the following function which is dependent on the amount of processing done, δz , and the node at which it is performed, n :

$$\phi(\delta z, n)$$

On a given node, n , ϕ is increasing in δz , as it should be more costly to process more stages. If $n = 1$, ϕ represents the cost of processing on the Mobile Station. So rather than encompassing leasing costs, which there are none, it represents the cost of draining battery power as well as tying up the processor MS and preventing the use of other applications.

The control dilemma we examine is how much processing should be done at each node given the processing latency, τ_p , communication latency, τ_c , and leasing costs, ϕ . Rather than concentrating all computation at one node or requiring lengthy communication times, we aim to mitigate the processing and communication burdens by spreading it throughout the network. The goal is to determine a computing and transmission control to minimize delay and costs.

III. OPTIMAL COMPUTING/TRANSMISSION CONTROL

In order to determine the optimal computing and transmission control, we cast this as a shortest path problem and use Dynamic Programming to find the optimal control.

The state of the system is given by:

$$(z, n)$$

where $0 \leq z \leq M$ is the amount of processing that has already been completed and n is the node at which the request message is currently located.

At each state, the control to be determined is how much processing must be done prior to transmitting the message uplink along the n^{th} link: $\delta z \in [0, M - z]$. This decision results in processing latency, τ_p , processing costs, ϕ , and communication latency, τ_c . We can group these into latency ($\tau_p + \tau_c$) and processing costs ϕ . In order to study the core tradeoffs we introduce a scale factor, α_n , to weight the processing costs at each node. For instance, we can have $\alpha_1 = \beta$, $\alpha_{N+1} = 1$, and $\alpha_n = \alpha$ for $n \neq 1, N+1$. For $\beta = 0$, there is no cost for draining battery at the MS and for $\beta \rightarrow \infty$ battery costs are extremely expensive and subsequently little, if any, processing should be done at the MS. If $\alpha = 0$, leasing comes for free and we are mostly concerned with latency. Conversely, if $\alpha \rightarrow \infty$, then we are not concerned with latency and processing should be done at the node with the lowest leasing costs.

Define the total cost-to-go under policy π starting in state (z, n) by:

$$J^\pi(z, n) = \sum_{l=n}^N \left\{ \tau_p(\pi(z_l, l), l) + \alpha_n \phi(\pi(z_l, l), l) + \tau_c(z_l + \pi(z_l, l), l) \right\} + \tau_p(M - z_N, N + 1) + \alpha_{N+1} \phi(M - z_N, N + 1) \quad (1)$$

Then we can define $J^*(z, n)$ as the minimum cost-to-go given that z stages of processing have already been completed and the request resides at node n .

$$J^*(z, n) = \min_{0 \leq \delta z \leq M - z} \left\{ \tau_p(\delta z, n) + \tau_c(z + \delta z, n) + \alpha_n \phi(\delta z, n) + J^*(z + \delta z, n + 1) \right\} \quad (2)$$

Once the request reaches the Application Server, the remaining processing stages must be completed. Therefore,

$$J(z, N + 1) = \tau_p(M - z, N + 1) + \alpha_{N+1} \phi(M - z, N + 1) \quad (3)$$

The optimal policy can be calculated via backward recursion and using Eqn. 2 and 3.

A. Properties of Optimal Control

Finding a closed form solution to the preceding formulation is difficult in general. However, we can identify a few important properties of the optimal control.

Suppose that each intermediate node is identical. That is, the processing times and costs are identical. Furthermore, assume that they are linear in the number of stages processed so that $\tau_p(\delta z, n) = k\delta z$ and $\phi(\delta z, n) = g\delta z$ for all $n \neq 1, N + 1$. Then if it is optimal to lease any processing power, it is optimal to lease it all from the first intermediary node. Let δz_n^* denote the number of stages processed at node n under the optimal policy.

Proposition 1: If $\forall n \neq 1, N + 1$, $\tau_p(\delta z, n) = k\delta z$ and $\phi(\delta z, n) = g\delta z$, then $\delta z_n^* = 0$ for all $n \neq 1, N + 1$ and 2.

Sketch of Proof: This is shown via a proof by contradiction. Assume there exists some intermediary node $n \neq 2$ such that $\delta z_n^* > 0$. Now instead, process these stages at node 2. Because $\tau_p(\cdot, \cdot)$ and $\phi(\cdot, \cdot)$ are identical and linear for all intermediary nodes, the processing cost and time is unchanged. However, because more processing is done at node 2, the size of the request message is smaller and communication latency is reduced, leading to lower costs. This contradicts the optimality of $\delta z_n^* > 0$. So $\delta z_n^* = 0$ must hold for $n \neq 1, 2, N + 1$. ■

It's interesting to note that the preceding proposition holds even in the case of identical, concave functions for $\tau_p(\cdot, n)$ and $\phi(\cdot, n)$. To give some intuition why this would be the case recognize that with concave functions, the cost-per-stage in terms of leasing and processing latency is decreasing as more stages are processed together. Hence, it is more beneficial to process many stages at once at one node, rather than process

1 stage at many nodes. We omit the details here for the sake of space.

As the communication link between the Mobile Station and the first network node degrades, communication latency will increase. By processing more stages at the Mobile Station, the request size will decrease, subsequently decreasing the communication latency. Define δz_{MS}^* as the number of stages completed at the Mobile Station. The following proposition formalizes this intuition:

Proposition 2: For fixed costs, δz_{MS}^* is decreasing as the capacity of the first link, c_1 , increases.

Sketch of Proof: This is shown via a proof by contradiction. Consider two systems with identical parameters and cost structures, except for $c_1 < c'_1$. Now assume that $\delta z_1^* < \delta z_1'^*$. It is possible to show that $J^*(z + \delta z_1^*, 2) > J^*(z + \delta z_1'^*, 2)$ since more stages have been processed in the second system and all other costs are equal. Also, $\tau_c(z + \delta z_1^*, 1) > \tau_c^*(z + \delta z_1'^*, 1)$ because the request message size is smaller in the second system. So for δz_1^* to be optimal for the c_1 system, the processing costs ($\tau_p + \phi$) must decrease more than the increase in communication and future costs by processing δz_1^* instead. However, if this were true, then δz_1^* could also be decreased to $\delta z_1'^*$ reducing the overall cost and contradicting its optimality. ■

When the communication bandwidth is very limited, utilizing Network-Assisted Computing is likely to reduce the network traffic by processing the request at earlier nodes and reducing the message size.

The total cost for servicing a request is given by $J(0, 1)$ as a request originates at the Mobile Station, node 1, and no processing has been performed on it yet. This can be broken into different costs:

$$\begin{aligned} J(0, 1) &= C_{\text{Latency}}(p) + C_{\text{Latency}}(e) + \alpha C_{\text{Leasing}} + \beta C_{\text{Battery}} \\ &= C_{\text{Latency}} + \alpha C_{\text{Leasing}} + \beta C_{\text{Battery}} \end{aligned} \quad (4)$$

Where latency can be split into processing and communications latency. In the next section, we will examine the relationship between these costs via numerical studies.

IV. NUMERICAL ANALYSIS

We examine the tradeoff between latency, battery usage, and leasing costs through numerical studies. These costs are tightly intertwined: increasing battery usage will decrease latency and leasing costs; increasing leasing costs will decrease latency and battery usage. We focus on these relationships in this section.

We assume a request requires 10 stages of processing. The size of the original request is 500 kilobytes (roughly the size of a JPEG image) and after completing all stages of processing, it is 1000 bytes. The decrease in request size is quadratic in the number of stages that have been completed, z . The processing time is linear in the number of stages completed and is dependent on the node it is being processed on.

We consider a network with 10 nodes, including the Mobile Station and Application Server. Therefore there are 8 intermediary nodes where processing power can be leased. Each mid-

network is identical in that the processing time and leasing costs are identical. We also assume they are linear in the number of stages processed so that Proposition 1 applies.

We examine the case where the leasing costs $\phi = 1$ for all n . The processing time for one stage at the Mobile Station is 100 milliseconds, while it is a constant ratio, $\frac{100}{r} < 100\text{ms}$, at the intermediary nodes, and $\frac{100}{r^2}$ ms at the Application Server. The bandwidth of the wireless links is uniformly distributed between 5 – 10 Mbits/second.

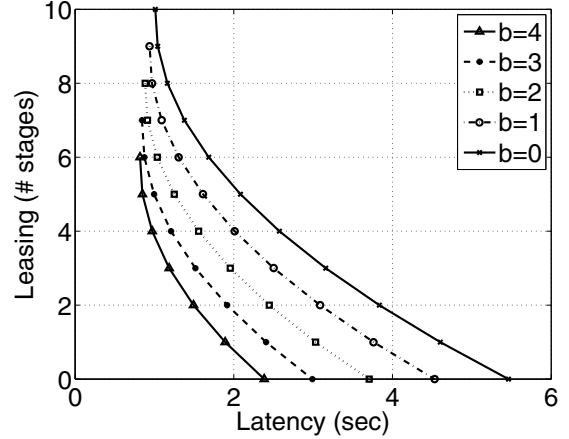


Fig. 2. Leasing vs. Latency for different number of stages (b) processed on the battery limited Mobile Station, i.e. $b = 0$ means no stages are processed at the MS.

In Fig. 2, we see the tradeoff between leasing costs, in terms of the number of processing stages performed on mid-network nodes, and latency, in terms of processing and communication time in seconds, for different amounts of battery usage. As expected, as the battery usage increases, leasing and latency both reduce. It's interesting to note that for extremely delay sensitive applications where response times must be of the order of seconds, leasing should be done very aggressively. In fact, all remaining processing should be leased from the intermediary nodes in order to avoid high delays due to communication over the wireless links. Because all mid-network nodes are identical and costs are linear, Proposition 1 takes effect, so all processing is done on the first mid-network node after the Mobile Station.

In some instances, the first link may be highly congested and processing at the Mobile Station becomes imperative. In Fig. 3, we see how the amount of processing done on the MS varies with the throughput of the first hop between MS and intermediary nodes. As given by Proposition 2, the number of stages processed on the Mobile Station, and subsequently the amount of battery energy that is drained, decreases as the quality of the first communication link improves. As the channel improves, the communication latency decreases and so less processing must be done at the MS to reduce latency. Each line corresponds to different α values to weight the importance between leasing costs and latency. For larger α , leasing becomes more expensive and less desirable. Therefore, to avoid lengthy delays due to the transmission of such a large file, more processing must be done at the MS to reduce the

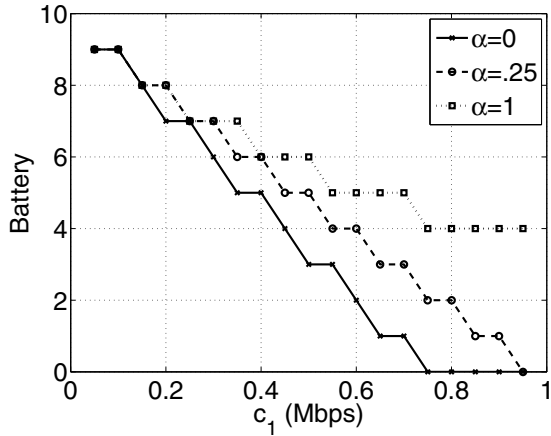


Fig. 3. Battery Usage vs. c_1 , throughput of first network hop. For various tradeoff levels between Leasing costs and Latency.

size of the request message.

The processing times on the various nodes vary due to the different types of processors they may have. For instance, the processor in the Mobile Station may be very limited compared to that of the remote Application Server. r captures the variance between these processing times. The larger the value of r , the more disparate the processing times. Because the processing times per stage improve from the MS to the intermediary nodes to the AS, one suspects that as r increases, latency will decrease significantly. Fig. 4 shows this trend when no processing is done at the MS. It is interesting to note that when jumping from $r = 1$ to $r = 2$ the decrease in latency is much more significant than the jump from $r = 4$ to $r = 20$. Despite the fact that the increase in r corresponds to a decrease in delay, for very large r , the delay is mostly due to communication of the request message rather than processing, so the change is less pronounced.

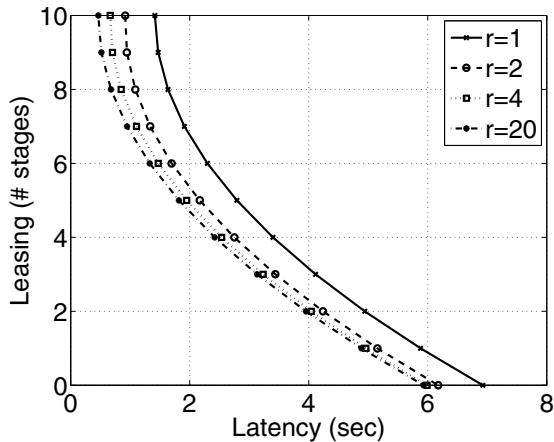


Fig. 4. Leasing vs. Latency for various values of the ratio between processing times on each node, $\frac{1}{r}$.

We have seen that battery usage, latency (both due to processing and communication), and leasing costs are highly intertwined. These costs are also highly dependent on system parameters such as communication bandwidth; processor speeds at the MS, AS, and intermediary nodes; as well as

request message size as a function of the number of stages processed. These parameters and costs functions are under the control of the system designer depending on the specific application. For instance, computation time will depend on the processor manufacturer and type application. By studying these tradeoffs, we can gain a better understanding of the relationships between each cost. This knowledge will help future system design. From a user's perspective, one must determine how much processing power to lease from mid-network nodes in order to satisfy delay constraints and extend battery life. From a network administrator's perspective, one must determine how much to charge for leasing processing power in order to encourage users to use the new feature while generating revenue.

V. CONCLUSION

Mobile multimedia applications are becoming more plentiful and complex. As the demands for processing power increase, physical limitations such as the bandwidth of communication links and the battery life on mobile devices can restrict the number of applications that can be utilized. Using "Wireless Network-Assisted Computing" to minimize the effect of these limits allows for the use of more applications and an enriched user experience. Leasing processing power from mid-network nodes will reduce battery usage as well as the request message size, thereby reducing communication latency. Using Dynamic Programming, we have determined the optimal tradeoff between leasing costs, latency, and battery drain. We have identified a couple of interesting properties of the optimal processing policy. Through numerical analysis we have shown the intricate relationship between these different sources of stress on the system.

REFERENCES

- [1] T. Yeh, K. Tollmar, and T. Darrell, "Searching the Web with Mobile Images for Location Recognition," in Proc. of Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2. IEEE Computer Society, pp. 76-81, 2004.
- [2] G. Fritz, C. Seifert, and L. Paletta, "A Mobile Vision System for Urban Detection with Informative Local Descriptors," in Proc. of IEEE International Conference on Computer Vision Systems (ICVS), pp. 30, 2006.
- [3] H. Bay, B. Fasel, and L. V. Gool, "Interactive Museum Guide: Fast and Robust Recognition of Museum Objects," in Proc. of the First International Workshop on Mobile Vision, May 2006.
- [4] G. Takacs, V. Chandrasekhar, N. Gelfand, Y. Xiong, W.-C. Chen, T. Bismpigianis, R. Grzeszczuk, K. Pulli, and B. Girod, "Outdoors Augmented Reality on Mobile Phone using Loxel-Based Visual Feature Organization," IEEE Trans. Pattern Analysis and Machine Intelligence, submitted Sept. 2007.
- [5] S. Gitzenis and N. Bambos, "Power-Controlled Data Prefetching/Caching in Wireless Packet Networks," IEEE Infocom 2002, pp. 1405-1414, 2002.
- [6] S. Gitzenis and N. Bambos, "Efficient data prefetching for power-controlled wireless packet networks," IEEE Mobile and Ubiquitous Systems: Networking and Services, (MobiQuitous), pp. 64 - 73, 2004.
- [7] S. Drew and B. Liang, "Mobility-aware Web prefetching over heterogeneous wireless networks," in Proc. IEEE Personal, Indoor and Mobile Radio Communications (PIMRC), pp. 687- 691, 2004.
- [8] D. Bertsekas, *Dynamic Programming and Optimal Control*, vol. 1 & 2, 2nd. Ed., Athena Scientific, 2000.
- [9] P. A. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," IEEE Transactions on Multimedia, Vol. 8, No. 2, April 2006.