

Throughput Estimation for Singleton Video Streaming Application over Wireless Sensor Network

S. Guo and T.D.C. Little
Department of Electrical and Computer Engineering
Boston University, Boston, MA 02215
{guosong, tdcl}@bu.edu

MCL Technical Report No. 05-20-2010

Abstract

Recent advances in wireless communications technology and low-power, low-cost CMOS imaging sensors enable a new sensing modality employing ubiquitous distributed video sensing. In many video sensing applications multimedia streams are often required by the base station to provide detailed information of the target spot in the sensor field. Such applications demand a strict throughput requirement for each data stream and expect the underline routing scheme to yield throughput-aware data paths for video data delivery. As a result, data path throughput estimation becomes increasingly important for successful video delivery algorithm designs. In this paper, we propose a data path throughput estimation model for singleton streaming applications, where only one data path is instantiated between source and destination. Coupled with distributed TDMA packet scheduling scheme, this path throughput estimation model can be employed by path formation algorithms to construct data paths from source to destination to yield throughput-aware video delivery. We prove that the throughput of such data path is independent of the path length. OPNET simulation results indicate that our model is accurate and our proposed TDMA packet scheduling scheme is favorable for video streaming applications.

Keywords – Wireless Sensor Network (WSN), Wireless Video Sensor Network (WVSN), Throughput Estimation, Video Streaming.

This material is based upon work support by the National Science Foundation under Grant No. CNS-0435353.

1. Introduction

With the development of low power, low cost CMOS imaging sensors, scientists envision a great potential of multimedia streaming application for wireless sensor networks (WSN) in the area of homeland security, habitat monitoring and image-based monitoring and control. For these applications a sensor node can capture images, audio and/or video information, and send them in a compressed form to a consumer elsewhere on the network. A user need not wait for the download of the entire video sequence but instead can playback the content immediately once data begin to arrive at the receiver.

Today, many of the existing wireless video sensor network (WVSN) applications [1-5] are an integration of Internet video streaming solutions to the domain of WSNs. Some applications rely on conventional wired video cameras. Others assume wireless communications but do not address scale-up to large numbers of video cameras nor the support of multiple streams.

One of the main obstacles for the deployment of WVSN application is the lack of mechanisms to form efficient, reliable, and dynamic video data path without information of the video data delivery performance. A good path formation algorithm can optimize the data delivery performance based on throughput estimation. However, the data path throughput estimation in the wireless ad hoc domain is challenging. Many factors for example, radio interferences and unpredictable link failures, have a significant impact on the accuracy of the throughput estimation. In this paper, we focus on data path throughput estimation modeling of simple WVSN applications, singleton streaming applications, where only one video data path is employed in the network. Our throughput estimation model enables a routing algorithm to accurately predict the throughput of a potential data path. Coupled with distributed TDMA packet scheduling scheme, this path throughput estimation model can be employed to construct throughput-aware video delivery path whose performance is independent of the path length. The rest of our paper is organized as follows: Section 2 reviews the related work of data path throughput estimation. Section 3 derives our throughput estimation model. Section 4 describes our simulation results. Section 5 concludes the paper.

2. Related Work

There are intensive papers which focus on the data path throughput estimation [7-10]. However, most of these works are mainly targeting wired networks or networks with wired infrastructure. Some other efforts in the wireless domain [11-12] lack generalization. These research proposals are only appropriate for particular MAC protocols.

Sun et al, [13] propose a comprehensive solution for the end to end data throughput estimation for WSN named SenProbe. SenProbe relies on the time dispersion between estimation packets to provide data path capacity estimation for WSN. Instead of using back-to-back packet pairs, SenProbe relies on the back-to-back packet train to overcome the effect of hidden terminals in CSMA-CA. The length of this back-to-back packet train depends on the interference range and the transmission range of the specific radio technology.

3. Throughput Estimation Model and Packet Scheduling Scheme

In this section, we introduce the mathematical throughput estimation model for data transmission on a single path and our TDMA-based packet scheduling scheme. We assume that (1) Radio range of each sensor node is identical; (2) A single radio channel is shared by all sensor nodes; (3) Simultaneous radio transmission within one node's radio range will cause packet collisions at the node; (4) Radio transmission outside the node's radio range will not interfere with the packet receiving process of the node.

A. Throughput Estimation Model:

We cluster sensor nodes along a data path into small concatenated groups. The size of each group is determined by the interference range measured in hop count of the sensor node and can be calculated by (1). For simplicity we constrain the interference range of a sensor node along a data path to be one hop and therefore each group in our model contains three nodes with length of two hops as shown in Fig. 1. We propose a distributed TDMA-based packet scheduling scheme within each group to control the packet forwarding process. This scheme eliminates the contention inside each group and allows nodes that are 3 hops away from each other to transmit data simultaneously such as node A and D in Fig 1. Under our network radio model, the TDMA-based packet scheduling scheme achieves the pipelined data forwarding along the path for the streaming application.

$$\text{Group size} = \text{Interference range} + 2 \quad (1)$$

We model each group along the path as a sub-queuing system and notice that if there is one packet being forwarded inside the sub-queuing system, due to the shared channel assumption the arriving packet to the system should be dropped since the packet is likely to be corrupted/interfered by the radio transmission inside the subsystem.

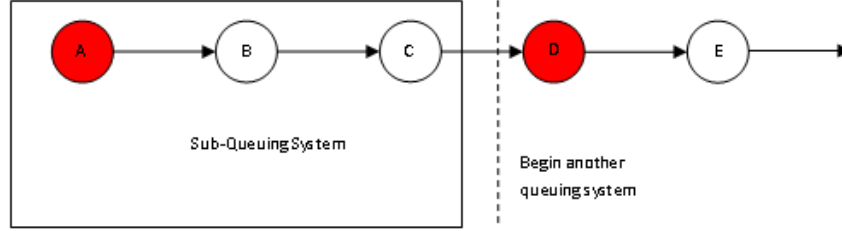


Figure 1. Illustration of Sub-queuing System

If we assume the arrival process is a renewal process, then we can use G/G/1/1[14] queuing model to characterize our first sub-queuing system. Since the arrival process of the cascaded system is the departure process of the prior system, in order to understand the transmission process of the entire path, we derive two important theorems.

Theorem One: If the Input process of our subsystem is a renewal process the output process of our subsystem is still a renewal process.

Proof:

We denote the following variables:

- S_k : Service time of the k th accepted packet in the subsystem
- A_i : Inter arrival time between the i^{th} and $(i+1)^{\text{th}}$ packet arrival
- D_k : Inter departure time between the k^{th} and $(k+1)^{\text{th}}$ accepted packet by subsystem

Without lose of any generality, we assume that A_n is the inter arrival time between the k^{th} accepted packet by system and the next packet to the system. We can derive D_k as following:

$$D_k = \begin{cases} A_n - S_k + S_{k+1} & (S_k < A_n) \\ A_n + A_{n+1} - S_k + S_{k+1} & (A_n < S_k < A_n + A_{n+1}) \\ \dots & \dots \\ \sum_{i=n}^{n+l} A_i - S_k + S_{k+1} & (\sum_{i=n}^{n+l-1} A_i < S_k < \sum_{i=n}^{n+l} A_i) \\ \dots & \dots \end{cases} \quad (2)$$

While the service time of each system is a constant, D_k can be simplified as

$$D_k = \begin{cases} A_n & (S < A_n) \\ A_n + A_{n+1} & (A_n < S < A_n + A_{n+1}) \\ \dots & \dots \\ \sum_{i=n}^{n+l} A_i & (\sum_{i=n}^{n+l-1} A_i < S < \sum_{i=n}^{n+l} A_i) \\ \dots & \dots \end{cases} \quad (3)$$

Since A_i are iid random variables and S is constant for our TDMA based packet scheduling scheme. D_k can be derived as a series of iid random variables expressed in (2). Therefore the departure process is also a renewal process. Under this case we can model the rest of cascaded sub systems to be G/G/1/1 queuing models, and our data path is a cascade of G/G/1/1 queuing systems as in Fig. 2.

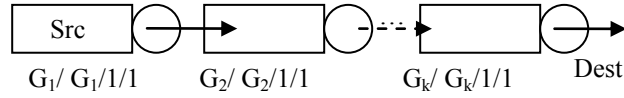


Figure 2. G/G/1/1 Tandem Queue Model of Data Path

Theorem Two: The data output process of the data path is the same as the data output process of the first subsystem along the path and it is independent of the path length.

Proof:

We denote the following variables:

- S : service time of each subsystem
- A_k^i : Inter arrival time between the k^{th} and $(k+1)^{\text{th}}$ packet arrival of the i^{th} subsequent system along the path
- D_k^i : Inter departure time between the k^{th} and $(k+1)^{\text{th}}$ accepted packet of the i^{th} sub system along the path

Given that the departure process of the i^{th} subsystem is exactly the arrival process of the $(i+1)^{\text{th}}$ system, we have

$$A_k^{i+1} = D_k^i$$

As each subsystem will take exactly S time to forward the packet then

$$\begin{aligned} D_k^i > S &\Rightarrow A_k^{i+1} > S \\ \Rightarrow A_k^{i+1} &= D_k^{i+1} \text{ (According to (2))} \\ \Rightarrow D_k^1 &= D_k^2 = \dots = D_k^i = D_k^{i+1} \dots \end{aligned}$$

Hence the inter-departure time of the first subsystem along the path is exactly the inter-departure time of the entire path. As a result, the data output process of the entire data path is independent of the path length and identical to the data output process of the first subsystem along the path.

According to the above two theorems, to characterize the throughput of the entire data path is equivalent to characterize the throughput of the first subsystem along the data path.

We denote the CDF of D_k as $F(t)$. $F(t)$ is expressed in (4).

$$\begin{aligned}
F(t) = & \Pr(D_k < t) = \Pr(S < A_n) * \Pr(A_n < t) + \Pr(A_n < S < A_n + A_{n+1}) \\
& * \Pr(A_n + A_{n+1} < t) + \dots + \Pr(\sum_{i=n}^{n+l-1} A_i < S < \sum_{i=n}^{n+l} A_i) * \Pr(\sum_{i=n}^{n+l} A_i < t) + \\
& \dots
\end{aligned} \tag{4}$$

We denote the Laplace transform of $F(t)$ as $L(F, s)$. If $L(F, s)$ is second order differentiable, we can find the expected inter departure time and its variance in (4) and (5):

$$E(D_k) = - \left. \frac{d[s * L(F, s)]}{ds} \right|_{s=0} \tag{5}$$

$$Var(D_k) = \left. \frac{d^2[s * L(F, s)]}{ds^2} \right|_{s=0} \tag{6}$$

According to the two theorems, we conclude that both the expectation and variance of the inter-departure time of the first subsystem along the path are exactly the same as that of the entire data path. The throughput performance can be accurately estimated using (5) and (6) and it is independent of the path length. This nice property enables routing algorithm to find detour of data routing so as to isolate concurrent streaming paths to avoid path coupling interferences and meanwhile provides guaranteed data path throughput performances.

B. TDMA Based Packet Scheduling Scheme:

We propose a distributed TDMA-based packet scheduling scheme to manage the data forwarding process inside each subsystem. The following elaborates the details of the scheme.

1) Media Access Control

We assign each node inside a subsystem a data sending time slot. The slot length for transmitting one packet can be calculated according to the link bandwidth or manually assigned by the source node. Since our subsystem is only composed of three sensor nodes (2 hops), by assigning each node a transmitting slot, three time slots will be enough for packet transmission in the subsystem.

2) Congestion Control

The first node of each subsystem along the path acts as a cluster head of the subsystem. This node can decide whether the received packet from the upstream subsystem should be dropped or forwarded. The decision is made according to the state of the subsystem. If the subsystem is currently busy forwarding one packet, all the packets received by the first node needs to be dropped otherwise the packet can be forwarded into the subsystem. In order to achieve this function without interchanging state information among nodes inside the subsystem, we set up a count-down timer on the first node when it begins forwarding a packet. The initial value of the timer is the product of the slot length and subsystem size (number of nodes in the subsystem). The first node of each subsystem is not allowed to accept packets during the activation period of the timer. This property is complied with our G/G/1/1 model of the subsystem. The reason that we drop the received packets instead of caching them is due to the fact that these received packets are likely to be corrupted by the packet transmission inside the subsystem given our shared media assumption. According to our simulation results, such packet dropping scheme does not have any performance degradation. Moreover if the source sends packets at a relatively higher rate, such simple proactive packet dropping scheme will yield guaranteed throughput performance.

3) Time Synchronization

Once a packet is forwarded by the first node of the subsystem, a timestamp of the start of current slot is logged into the packet. When the subsequent node (the second node in the subsystem) receives the packet, it adds the agreed slot length to the time stamp to get the time of the first node and therefore synchronize its time with the first node. The third node of the sub-system adds twice the slot length to synchronize with the first node of the subsystem and so on. Each node in the sub system is only synchronized to the first node of the system. Such distributed synchronization scheme simplifies the implementation complexity of the network-wide synchronization. The usage of data packet for synchronization minimizes the synchronization overhead. The slot length can be either predefined by the source node and then propagates to all the sensor nodes along the path or it can be computed dynamically according to the quality of the links along the data path and updated periodically if the wireless environment changes dramatically.

4. Simulation Results

The success of our path formation algorithm is largely depended on the accuracy of our throughput estimation model and the performance of our proposed TDMA scheduling scheme. In this section we examine these two critical components via OPNET simulations.

A. TDMA Scheme vs CTS/RTS

We compare the performance of TDMA and CTS/RTS scheme in two different scenarios. In Scenario 1, the link utilization ratio is very low. One sample simulation result of this scenario is provided in Fig. 3. In this example, the data source generates a packet with a size of 1024 bits every 5 seconds. The link transmission delay is 1 second. The TDMA time slot is set to be slightly larger than the link transmission delay. The length of the path is 8 hops and there is no other radio transmission beyond the data path. The simulation results demonstrate that when the data rate is relatively low, both TDMA and CTS/RTS scheme can eventually achieve same throughput performance but TDMA scheme adapts to the source data rate much faster than the CTS/RTS. The rate adaption delay for CTS/RTS scheme is attributed to the network communication overhead introduced by CTS/RTS hand shake.

In Scenario 2, we push the data rate higher approaching the optimal throughput of the data path. According to our analysis, the optimal data path throughput in our simulation case is roughly one third of the link capacity. The result is displayed in Fig. 4. Once again, the simulation results show that our TDMA scheme quickly adapts to the optimal data path throughput roughly 0.33 in this case. The CTS/RTS scheme performs poorly in this scenario. It only achieves 3/4 of the optimal rate and also generates substantial delay jitter. The delay jitter and low throughput is a consequence of CTS/RTS hand shake and media access contention.

The simulation demonstrates that our TDMA scheme outperforms CTS/RTS scheme in both low data rate and high data rate scenarios. The delay jitter introduced by CTS/RTS and media access contention will largely degrade the playback quality of the received video. Large buffer at the destination is on demand to overcome the unpleasant delay jitter but this solution extends the period of data caching which is not desired for real time video streaming applications.

Yet another advantage of our TDMA scheme is that the implementation of this scheme is simple and the usage of data packet to synchronize sensor node within a small system save the extra energy of sending negotiation packets like CTS/RTS packets in the other scheme.

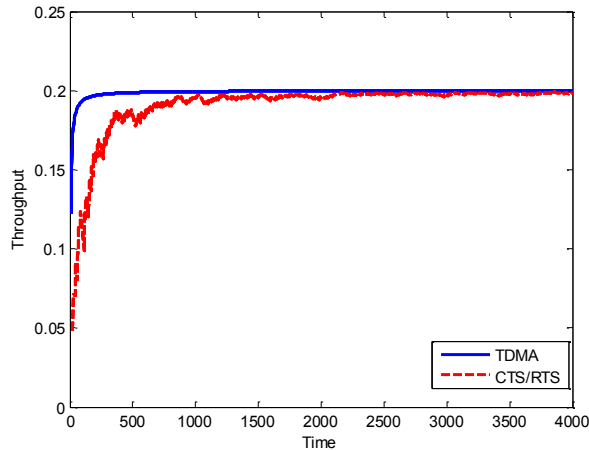


Figure 3. TDMA vs. CTS/RTS in Low Data Rate (Scenario 1)

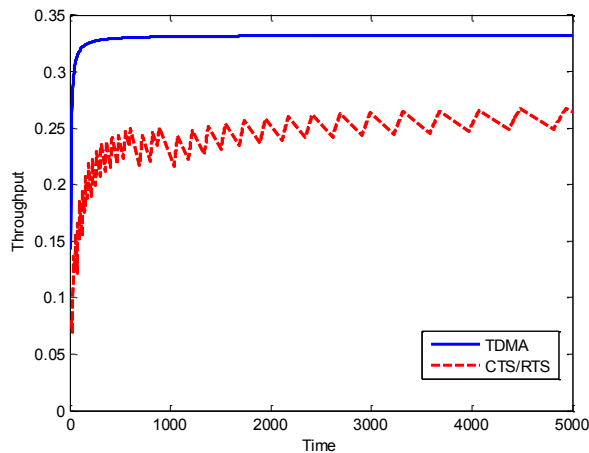


Figure 4. TDMA vs. CTS/RTS in High Data Rate (Scenario 2)

B. Subsystem Throughput Estimation

In order to show the accuracy of our estimation model, we conduct experiments to compare the subsystem throughput simulation results with our theoretical results. For simplicity, we only compare the expected throughput of one subsystem. We assume the initial packet arrival interval of the subsystem is exponential distributed with mean 10 seconds and the link transmission delay of a packet is one second. Consequently, the subsystem delay by applying our TDMA scheme is 3 seconds. By taking expectation on both sides of (2), we derive the theoretical expected throughput of the subsystem to be approximately 0.079 packets per second which is very close to the simulation result roughly 0.0784.

5. Conclusion and Future Work

We have established a data path throughput estimation model for singleton video streaming applications over WSN where only one data streaming path exists in the network. By leveraging this model we can accurately estimate the throughput performance of the potential data path. We prove that the throughput performance of the data path is independent of the path length by applying our distributed TDMA packet scheduling scheme. This nice property can be further exploited by a path formation algorithm to isolate concurrent streaming paths to avoid path coupling interferences and meanwhile provide guaranteed data path throughput performances. In the future, a complimentary data

transmission interference model will be investigated to deal with more complex video streaming applications. This data path interference model is expected to associate throughput estimation to more sophisticated parameters, such as path positions, network size and number of concurrent data sending paths. We believe such basic video data throughput estimation models will inspire advanced path formation algorithms design for video streaming applications to fully explore the capacity of our wireless sensor networks.

6. References

- [1] R. Holman, J. Stanley, and T. Ozkan-Haller. "Applying Video Sensor Networks to Nearshore Environment Monitoring," *IEEE Persave Computing*, 2003, pp. 14-21.
- [2] T. Teixeira, D. Lymberopoulos, E. Culurciello, Y. Aloimonos, and A. Savvides, "A Lightweight Camera Sensor Network Operating on Symbolic Information," in *Proc. First Workshop on Distributed Smart Cameras*, Boulder, CO, USA, 2006.
- [3] M. Chu, J. Reich, and F. Zhao, "Distributed Attention in Large Scale Video Sensor Networks," in *Proc. IEEE Intelligent Distributed Surveillance Systems*, 2004, pp. 61-65.
- [4] T. Teixeira, A. Savvides, "Lightweight People Counting and Localizing in Indoor Spaces Using Camera Sensor Nodes," in *Proc. First ACM/IEEE International Conference*, 2007, pp.36-43.
- [5] P. Kulkarni, D. Ganesan, P. Shenoy, and Q. Lu, "SensEye: A Multi-tier Camera Sensor Network," in *Proc. 13th Annual ACM International Conference on Multimedia*, 2005, pp. 229-238.
- [6] A. Rowe, D. Goel, and R. Rajkumar, "FireFly Mosaic: A Vision-Enabled Wireless Sensor Networking System," in *Proc. 28th IEEE International Real-Time Systems Symposium*, 2007, pp. 459-468.
- [7] C. Dovrolis, P. Ramanathan, and D. Moore, "What do packet dispersion techniques measure," in *Proc. IEEE Infocom*, 2001.
- [8] V. Jacobson, "Pathchar: A tool to infer characteristics of Internet paths," <ftp://ftp.ee.lbl.gov/pathchar>.
- [9] R. Kapoor, L. Chen, L. Lao, M. Gerla, M. Sanadidi, "CapProbe: A Simple and Accurate Capacity Estimation Technique," in *Proc. ACM SIGCOMM 2004*.
- [10] K. Lai, M. Baker, "Measuring Bandwidth," in *Proc. IEEE INFOCOM*, 1999, p. 235-245.
- [11] C. Buratti, F. Fabbri, R. Verdone, "Area Throughput of an IEEE 802.15.4 based Wireless SensorNetwork," in *Proc. European Conference on Wireless Sensor Networks*, EWSN, Cork,Ireland, 2009.
- [12] C. Chan, S. Liew, A. Chan, "Many-to-One ThroughputCapacity of IEEE 802.11 Multihop Wireless Networks," *IEEE Trans. on Mobile Computing*, vol. 8, no. 4, pp.514-527, April 2009.
- [13] T. Sun, L. Chen, G. Yang, M. Sanadidi, and M. Gerla, "Senprobe: Path capacity estimation in wireless sensor networks," in *Proc. Third International Workshop on Measurement, Modeling, and Performance Analysis of Wireless Sensor Networks*, San Diego, USA, 2005.
- [14] L. Kleinrock, *Queueing Systems*, vol. 1. New York: Wiley, 1975.
- [15] D. Bertsekas and R. Gallager, *Data Networks*, Englewood Cliffs, NJ, Prentice-Hall, 1987.