Packet Loss Modeling for Perceptually Optimized 3D Transmission[1]

Irene Cheng*, Lihang Ying and Anup Basu

Dept. of Computing Science, Univ. of Alberta, Edmonton, Alberta, Canada, Email: {lin, lihang, anup}@cs.ualberta.ca

Abstract

Transmissions over unreliable networks, *e.g.* wireless, can lead to packet loss. An area that has received limited research attention is how to tailor multimedia information taking into account the way packets are lost. We provide a brief overview of our research on designing a 3D perceptual quality metric integrating two important factors, resolution of texture and resolution of mesh, which control transmission bandwidth, followed by a suggestion on alternative strategies for packet 3D transmission of both texture and mesh. These strategies are then compared with respect to preserving 3D perceptual quality under packet loss in ad hoc wireless networks. Experiments are conducted to study how buffer size, sending rate, sending intervals and packet size can affect loss in unreliable channels. A model for estimating the optimal packet size is then proposed. We derive the optimal number of packets based on this model, and relate the theoretical derivations to actual network data.

I. INTRODUCTION

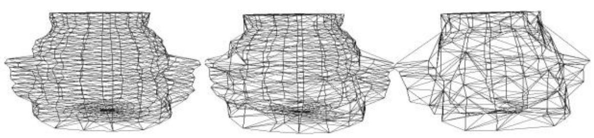

Figure 1: Nutcracker toy model at various mesh resolution levels.

An important consideration in designing effective interactive online 3D systems is to adaptively adjust the model representation, while preserving satisfactory quality as perceived by a viewer. While most research in the literature focus on geometric compression and use only synthetic texture or color, we address both *geometry resolution* and *realistic texture resolution*, and analyze how these factors affect the overall perceptual quality. Our analysis is based on experiments conducted on human observers. The perceptual quality metric derived from experiments allows the appropriate level of detail (LOD) to be selected given the computation and bandwidth constraints. Detailed surveys on simplification algorithms can be found in [1, 2]. In order to easily control the details on a 3D object we will follow a simple model approximation strategy based on multi-resolution representation of texture and mesh. An example of geometric simplification is shown in Figure 1, in which a Nutcracker toy model is simplified to various resolution levels (number of triangles is 1,260 left, 950 middle and 538 right).

One of the major drawbacks with most 3D transmission algorithms is that they do not consider the possibility of packet loss over wireless or unreliable networks. Some wireless protocols proposed in the last decade include Transmission Control Protocol (TCP), User Datagram Protocol (UDP), Indirect-TCP (I-TCP) [3], and so on. For wireless networks, where packet loss occurs as a result of unreliable links and route changes, the TCP strategy leads to further delays and degradation in transmission quality. Even though issues of multimedia transmission over wireless networks have received attention [4], relatively little work has been done addressing wireless 3D transmission. In recent research, approaches for robust transmission of mesh over wireless networks [5,6] have been outlined. However, these methods do not take joint texture and mesh transmission into account. Also, in [5,6] it is assumed that some parts of the mesh can be transmitted without loss over a wireless network, allowing progressive mesh transmission to give good results. However, this assumption implies implementing a special standard with a combination of UDP and TCP protocols, which in general cannot be guaranteed in an arbitrary wireless environment. Special models for packet loss probability have been developed by other researchers [7]. However, these models are usually associated with requirements such as retransmission. To keep our study applicable in an unrestricted ad hoc wireless environment, we simply assume packet-based transmission where a certain percentage of the packets may be lost. In this scenario, we compare how various types of 3D transmission strategies fare, and how to take perceptual quality into account in designing a better strategy.

We consider an approach based on a perceptual quality metric following our earlier work in 2003 [8]. Other approaches to joint texture-mesh transmission have been discussed in [9,10] in 2004. The approach in

[1] The support of Alberta Science and Research Authority (ASRA) and NSERC are gratefully acknowledged. Part of this work was presented in IEEE Int. Conf. on Multimedia & Expo, 2006, in Toronto, Canada. *Irene Cheng is currently an NSERC Fellow at the Computer and Information Science Department, University of Pennsylvania.

[9] is based on view-dependent rate-distortion optimization, whereas our approach is view-independent. Also, both [9,10] are progressive, which necessitates greater protection of base layers in case of packet loss; our approach on the other hand does not need to guarantee delivery of certain packets in order to make other packets useful. Joint texture-mesh transmission of terrains was addressed in [11]; however, the author did not consider perceptual quality optimization.

The remainder of this paper is organized as follows: Section 2 reviews past work on perceptual quality evaluation and discusses how to relate bandwidth with texture and mesh reduction considering perceptual quality. Section 3 examines possible strategies for 3D image transmission and analyzes which one is most suitable for optimizing perceptual quality under packet loss. Experimental results are presented. Different scenarios of packet loss attributed to different factors over a lossy network are presented in Section 4. A strategy for packet size optimization is proposed in Section 5, before the work is concluded in Section 6.

II. 3D Perceptual Quality Optimization

In the area of image compression, Mean Square Error (MSE) is commonly used as a quality predictor. However, past research has shown that MSE does not correlate well to perceived quality based on human evaluation [12]. Since this study, a number of new quality metrics based on the human visual system have been developed [13].

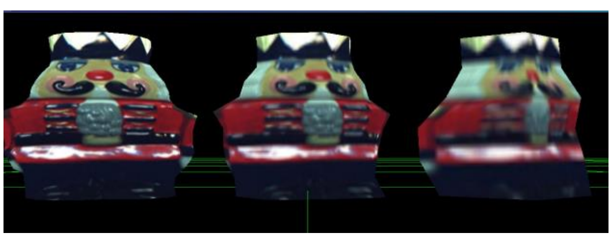

Figure 2: Evaluation Example

Several 3D objects were used as stimuli in our experiments. These objects were captured with the *Zoomage* 3D scanner. The participants (judges) were asked to compare the target stimulus with the two referential stimuli and assign it one of the following ratings: ***very poor (1), poor (2), fair (3), good (4), very good (5).***

Figure 2 illustrates two referential stimuli (left and right) and one target stimulus (center) in the experiment. Considering perceptual evaluations, we observed that:

(i) Perceived quality varies linearly with texture resolution (Fig. 3, top);

(ii) Perceived quality varies following an exponential curve for geometry (Fig. 3, bottom). Scaling the texture (t) and geometry (g) between 0 and 1, it can be shown that:

$$Q(g,t) = \frac{1}{\frac{1}{m+(M-m)t} + \left(\frac{1}{m} - \frac{1}{m+(M-m)t}\right)(1-g)^{c}} \quad (1)$$

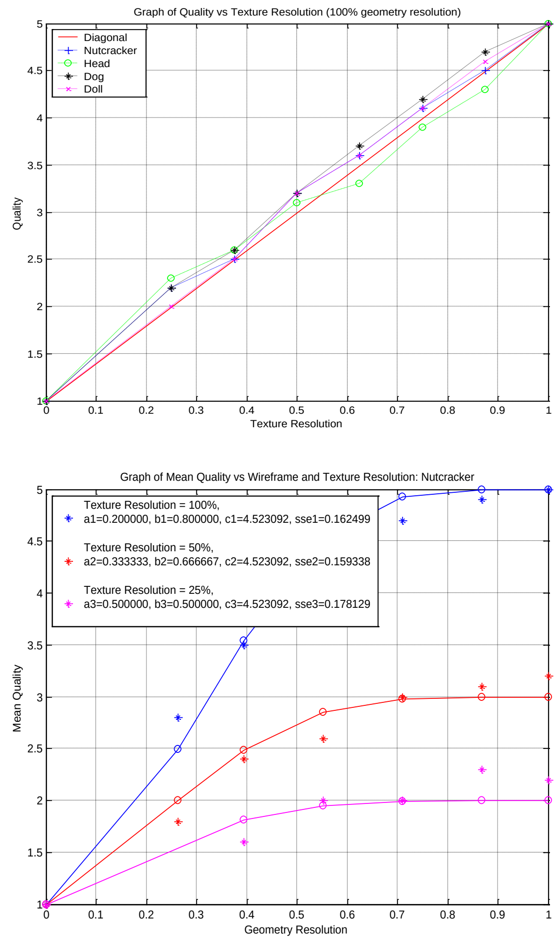

Figure 3: Top: Quality vs. Texture Resolution (100% Geometry Resolution); Bottom: Quality vs. Geometry for various texture resolutions.

Details of the perceptual evaluations and metric derivation can be found in our prior work [8]. Note that the quality value varies in the range of 1 (m) to 5 (M), because of the range of values allowed in the perceptual ratings.

Consider now that b is the estimated total bandwidth for the transmission time interval, T is the texture and G is the geometry file sizes, possibly compressed, at maximum resolution. We assume that as the texture (or geometry) is scaled by a factor t (or g) in both dimensions the corresponding file sizes get reduced to t^2T (or g^2G). To utilize the bandwidth completely we must have:

$$b = t^2 T + g^2 G \quad (2)$$

Given b we can choose the relative proportion of texture and mesh to create a 3D model in many different ways, as long as Equation (2) is satisfied. The question is "What is the optimal choice maximizing perceptual quality?" Considering m = 1, M = 5, and c = 2.7 (approximately) for many objects based on perceptual tests, Equation (1) can be further simplified to:

$$Q(g,t) = \frac{1}{\frac{1}{1+4t}+(1-\frac{1}{1+4t})(1-g)^{2.7}} \quad (3)$$

Maximizing Equation (3) is equivalent to minimizing the inverse of this equation; considering this and Equation (2), optimizing quality reduces to minimizing:

$$Q_{b,G,T}(t) = \frac{1}{1+4t} + (1-\frac{1}{1+4t})(1-\sqrt{\frac{b-t^2T}{G}})^{2.7} \quad (4)$$

where b, G and T are parameters.

Example 1:

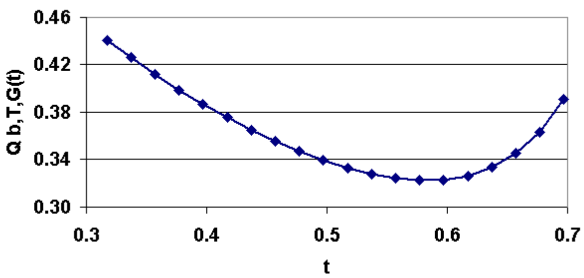

Figure 4: Inverse perceptual quality curve for Example 1.

Let b = 12 Mbits , T = 20 Mbits, and G = 10 Mbits. In this case t can only vary in the range [$\sqrt{2/20}$, $\sqrt{10/20}$] = [0.316, .707] so that Equation (2) can be satisfied. The graph of Equation (4) for varying t for this case is shown in Figure 4. The optimal value of t is close to 0.6 for this example. In general, given T and G for a 3D object, optimum t can be pre-computed for a discrete number of b values in the range [0,T+G].

III. Perceptually Optimized Transmission

To simplify the model of wireless transmission, we assume that data is sent in packets of equal size and there is a possibility that a certain proportion of these packets may be lost. Various protocols [14] suggest re-transmission approaches in case of packet loss; however, re-transmission is not conducive to time bound real-time applications, such as 3D visualization for online games. We considered several possible strategies for packet construction in wireless 3D transmission, and then analyzed the pros and cons of each. We found that breaking up a 3D image into fragments can cause unacceptable voids; progressive transmission [15] necessitates receiving packets at lower levels before packets at higher levels can become useful; and sending duplicate copies of base layer packets in progressive transmission increases bandwidth requirements. We thus focus on the two following strategies, concentrating on regular mesh transmission.

Strategy A:

3d Partial Information Transmission (3PIT): In this approach we break up the texture and mesh into packets by sub-sampling into overlapping but non-identical components. At the client site the overall texture and mesh are reconstructed based on interpolation from the received packets. One implementation of this approach is given by the following algorithm:

SERVER SITE:

T: original texture;
M: original mesh, in a regular form allowing easy sub-sampling;
Construct T_1, T_2, …, T_n by regular, non-identical sub-sampling of T;
(**Comment**: *For example, given a 100 x 100 pixel texture T, we can construct T_1, T_2, …, T_{16} by defining T_1 as T(0+4i, 0+4j), i,j=0,…24; T_2 as T(0+4i, 1+4j), i,j=0,…24; …, T_{16} as T(3+4i, 3+4j), i,j=0,…24.*)
Construct M_1, M_2,…, M_n by regular, non-identical sub-sampling of M;
Form packets P_1, P_2, …, P_n where $P_i = T_i + M_i$; i=1,…,n, with header and sub-sampling information added to each packet;
Transmit n packets to a client on request, possibly in a randomized order;

CLIENT SITE:

Request server to transmit a 3D object;
Receive packets from server;
Uncompress mesh and texture data stored in this packet;
Set up initial display based on first packet received and interpolation information stored in header;
Update display based on next packet received;

Limitations of Strategy A:
One of the shortcomings of this approach is that the texture and mesh data receives equal importance; *i.e.*, the same fraction of each is transmitted in a packet. The perceptual quality analysis in the last section shows that for optimizing perceptual quality the relative importance of texture and mesh can vary depending on the available bandwidth; this issue is not taken into account in Strategy A.

Strategy B:
3d Perceptually Optimized Partial Information Transmission (3POPIT): This approach extends 3PIT by taking perceptual quality into account. The algorithm modifies Strategy A by a bandwidth estimation step followed by perceptually optimized packet creation. Details are described below:

SERVER SITE:

T, M: as for Strategy A;
Receive bandwidth estimate (B_e) and estimated loss proportion (L) from requesting client;
Compute server transmitting bandwidth: $B_s \leftarrow B_e /(1 – L)$;
Compute optimum texture and geometry scaling factors t_e & g_e following procedure for minimizing Equation (4) in the last section, considering bandwidth to be B_e;
Compute scaled texture (T_s) and mesh (G_s), assuming transmitting bandwidth B_s, based on factors t_e & g_e;

(**Comment***: Specifically:* $Ts = \frac{t_e{}^2}{(1-L)} T$ *and* $Gs = \frac{g_e{}^2}{(1-L)} G$ *; with texture and mesh possibly being interpolated to higher than the current maximum size in case the scaling factors are greater than 1.)*
Construct $T_{s1}, T_{s2}, \ldots, T_{sn}$ by regular, non-identical sub-sampling of T_s;
Construct $M_{s1}, M_{s2}, \ldots, M_{sn}$ by regular, non-identical sub-sampling of M_s;
Form packets $P_1, P_2, \ldots, P_n$ where $P_i = T_{si} + M_{si}$; i=1,…,n, with header and sub-sampling information added to each packet;
(**Comment***: Number of packets n is chosen based on prior decision on packet size.)*
Transmit n packets to a client, possibly in a randomized order;

CLIENT SITE:

Request server to transmit a 3D object;
Receive packets from server for bandwidth estimation;
Estimate bandwidth (B_e) based on number of packets received in a certain time interval and estimate loss proportion (L);
Receive packets from server containing partial data on the 3D object;
Uncompress mesh and texture data stored in this packet;
Set up initial display based on first packet received and interpolation information stored in header;
Update display based on next packet received;

Comments on Strategy B:

On first observation it may appear that this strategy does not take packet loss proportion (L) into account in the transmission strategy. However, in reality this is not the case. Without any packet loss, the transmission bandwidth (B_s) would be used to compute the optimum texture and mesh scaling factors. When packets are lost the remaining packets may not be perceptually optimal for the effective bandwidth after packet loss. We thus form packets that are optimal at a lower bandwidth (B_e).

One of the drawbacks of Strategy B is the need to estimate bandwidth and packet loss ratio. This estimation based transmission may not be practical where feedback from client to a server is not reliable, or for multicasting over heterogeneous networks with varying packet loss and bandwidths. This issue needs to be addressed in future research.

Experimental results of strategy B

We show some preliminary implementations towards deploying 3POPIT over a lossy wireless network. Figure 5 shows the effect of receiving and combining 2, 4 and 8 of 16 sub-samples of the nutcracker mesh. Note that results may vary from one execution to another for a random percentage of packet loss.

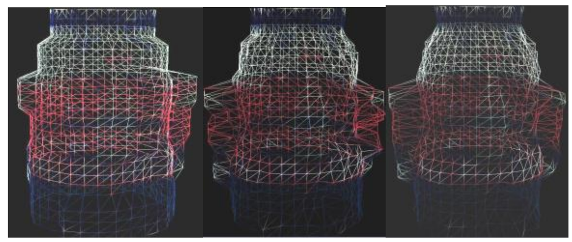

Figure 5: Interpolating and reconstructing mesh of nutcracker model when 2 (left), 4 (middle) and 8 of 16 packets are received.

Figure 6: Two representations of the Nutcracker texture + mesh models: Left has lower quality mesh, requiring 125 Kb total bandwidth, but higher perceptual quality; Right has higher quality mesh, resulting in lower quality texture to keep total bandwidth at 134 Kb, but has lower perceptual quality.

Figure 6 shows the effect of optimized *vs.* non-optimized transmission on perceptual quality. Two versions of the same model are shown, with the mesh on the left and the texture mapped on the right. Although the texture and mesh together for the left and right models use nearly the same bandwidth, 125 and 134 Kb respectively, the left one is favored by most viewers based on perceptual experiments.

Although a regular or semi-regular mesh is used for illustration in this paper, our strategy can be extended to irregular meshes where connectivity information needs to be transmitted; triangular faces are arranged in continuous long strips following the valence driven algorithm, neighboring vertices and connectivity information are distributed evenly into different packets to minimize the risk of losing data affecting a large neighborhood [23]. Figure 7 shows how our strategy, combined with the valence driven encoding and decoding algorithm [24], can be applied to an irregular mesh.

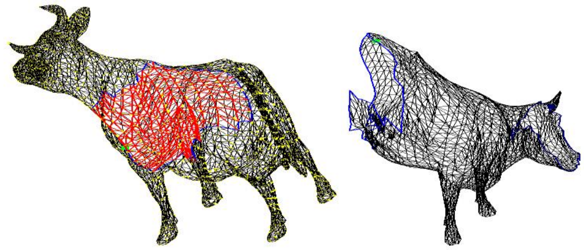

Figure 7: Combining our strategy with the valence driven algorithm on an irregular cow mesh, (left) the colored patch shows the neighborhood after 400 vertices are distributed to different packets. (Right) shows the partly reconstructed mesh after 3,000 vertices are retrieved from the packets.

Multiple-resolution strategy is often used to refine image or mesh data in a progressive manner. However, progressive methods [25, 26] necessitate greater protection of base layers in case of packet loss; our approach on the other hand does not need to guarantee delivery of certain packets in order to make other packets useful.

IV. NETWORK EXPERIMENTS ON PACKET LOSS

The long-term objective of our research is to identify the appropriate parameters (packet size, sending rate, sending interval and buffer size) for different applications to maximize throughput and minimize packet loss of UDP transmission in different Internet environments with Wireless LAN access. In the experiments, the server side was a desktop computer in the Department of Computing Science, University of Alberta, Edmonton, Canada. The client was a laptop, which linked to a router following 802.11b. The router accessed the Internet using a cable network (with a maximum capacity of 640KBps). The client was located in the same city as the server. Both client and server ran Red Hat Linux Release 9 Shrink (2.4.30 Kernel). The experiments were conducted during the day (8:00-19:00) from November 25 to November 27, 2005.

A. Buffer Size

Table 1: Receiving Rate and Packet Loss for Different Buffer Sizes

Buffer Size (bytes)	**Receiving Rate(Bps)**	**Packet Loss (%)**
(a) 32,768	126611	0.98
(b) 16,384	127450	0.39
(c) 4,096	123138	3.91
(d) 2,048	0	100.00

We first discuss the effect of socket buffer size on packet loss. With fixed packet size (4096 bytes) and sending rate (128KBps), Table 1 (a) and (b) show that different buffer sizes larger than the packet size make no significant difference on packet loss. However, if buffer size is less than packet size, all packets are lost as shown in (d). The interesting point is in (c), when buffer size is equal to packet size, there is a significant packet loss as well. This can be attributed to the fluctuating bandwidth; reducing bandwidth capacity can cause an overflow on a congested buffer.

In the experiments reported in sub-sections *B* to *D* below, we wanted to study packet loss independent of buffer size and therefore a large enough buffer size of 65,536 was used.

B. Sending Rate

Next we look into how sending rate affects packet loss. We consider a large enough sending interval and fixed packet size (4096 Bytes), and let sending rate increase from 128 to 1024 KBps. It can be seen from Figure 8, that as sending rate increases, receiving rate increases and packet loss remains around zero until around 500 KBps. However, after sending rate overflows the connection (larger than 500 KBps), packet loss dramatically increases and receiving rate drops owing to packet loss.

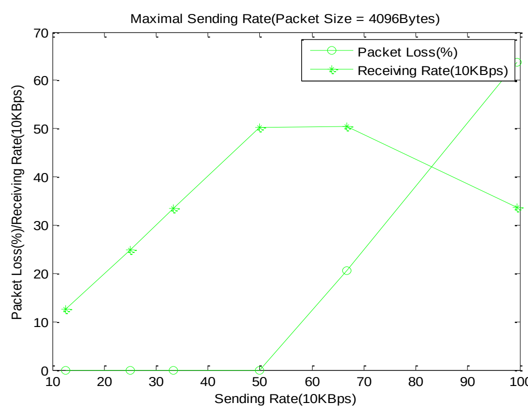

Figure 8: Effect of sending rate on packet loss.

C. Sending Interval

Fixing packet size at 32Bytes and without overflowing the connection, sending intervals varying from 10000, 4000, 3000, 2000, 1000, 800, 600, 500 nanoseconds were used to test the packet loss rate. Figure 9 shows the packet loss plotted against the time interval before transmitting the next packet. Clearly, the sending interval should not be too small (< 2 ms) otherwise loss rate can be high.

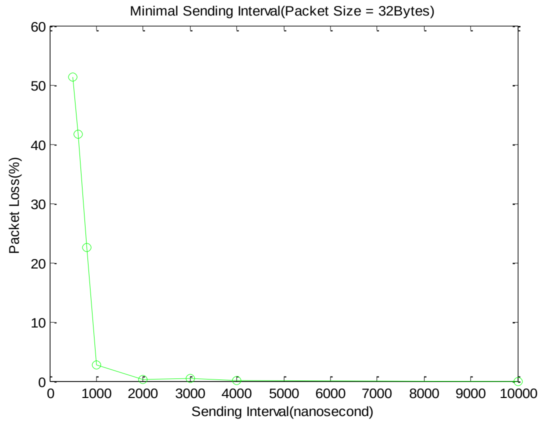

Figure 9: Packet loss vs. sending interval.

D. Packet Size

Now we want to see how different packet sizes affect the receiving rate, as well as the packet loss. First, we performed experiments in an environment without other competing connections. With sending rate around 256KBps, without overflowing the maximum connection capacity of 640KBps, packet size was selected from the set of [65536, 32768, 16384, 8192, 4096, 2048, 1024, 512, 256, 128, 64, 32] bytes. Table 2 shows that when sending interval is large enough (equal to or larger than 2 ms), different packet sizes have not much effect on sending rate, receiving rate, or packet loss. Sending rate remains stable independent of packet size. However, as mentioned above, when the sending interval is too small, packet loss sharply increases and receiving rate drops accordingly.

We then performed experiments in an environment with competing connections. In order to setup a competing environment, we configured the bandwidth of 802.11b to 1Mb. Four additional FTP concurrent connections were opened between the client and the server. Using a sending rate at 64KBps, packet sizes were selected from [128, 256, 512, 1024, 2048, 4096, 8192, 16384] bytes. Figure 10 shows how packet loss varies with packet size under such condition. When packet size is very small or very large, packet loss can be large. It was at the minimum when packet size was around 2,000.

In related work [21], adjusting the packet sending interval based on feedback from the network was proposed, but the work did not study the effect of different packet sizes. In [16] it was proposed to determine the sending rate T as a function of the packet size s, round-trip time $\boldsymbol{R}$, steady-state loss event rate $\boldsymbol{p}$, and the TCP retransmit timeout value t_{RTO}. However, they did not study how the theoretical model works in a Wireless LAN (WLAN) environment. UDP throughput and CPU utility for bulk data transfer with different sender and receiver buffer sizes and packet sizes was studied in [17] and [22] but packet loss was not taken into consideration. The delay of UDP, TCP and TCP with the option NODELAY for voice applications sending 160 bytes per 20 milliseconds was discussed in [18]. Our work differs from others by carrying out a comprehensive study on packet size, sending rate, sending interval, and buffer size through real world experiments in a competing WLAN environment, where there are packet losses besides congestion.

Table 2: Effect of different packet sizes on sending/receiving rate and packet loss (Without Competing Connections)

Packet Size (Byte)	**Sending Interval (nano-second)**	**Sending rate (Bps)**	**Receiving rate (Bps)**	**Packet Loss (%)**
65500	256000	255510	253852	0.00
32768	128000	255694	255694	0.00
16384	64000	255682	252899	0.78
8192	32000	255876	255637	0.00
4096	16000	255690	252476	0.98
2048	8000	255686	256480	0.00
1024	4000	255688	255374	0.00
512	2000	255724	255668	0.00
256	1000	255691	210535	16.58
128	500	255691	118407	53.69
64	250	255682	485	99.81
32	125	255401	499	99.80

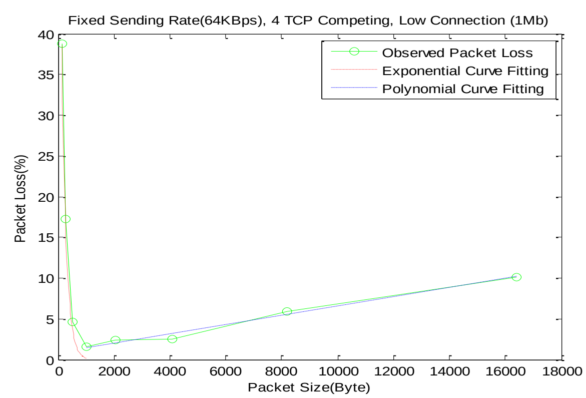

Figure 10: Packet loss vs. packet size in a congested network.

V. PACKET SIZE OPTIMIZATION MODEL

In Figure 10, it can be observed that in the competing environment when packet size is small the loss can be quite high because of congestion resulting from the need to route many packets. As the packet size gradually increases beyond an optimum point with low loss, the loss rate increases again. We propose a strategy to model this characteristic and determine the optimal packet size following some simplifying assumptions. We assume exponential and linear models for packet loss depending on packet size; however, the approach can be extended to other models as well. Even though various models for packet loss over wireless networks have been proposed [19, 4], determining the optimal packet size has not received much attention. For simplicity, we use the following assumptions and notation:

H: total network header size required for each packet;
S: amount of payload (application) data transmitted, if there is only 1 packet used;
s_n: amount of payload data transmitted in each packet if n packets used ;
Probability packet of size D is not lost is

$$e^{-\lambda D} \qquad (5)$$

i.e., an exponential packet loss model is used with parameter λ and packet size as variable.
Let the amount of data transmitted *be* $B = S+H$ when only 1 packet is transmitted, and $s_1 = S$. When 2 packets are transmitted, we have $B = 2s_2+2H$. Since:
$B = S+H = 2s_2+2H$, $s_2 = (S-H)/2$.
In general,

$$s_n = (S-(n-1)H)/n \qquad (6)$$

For large n, *i.e.* when $(n-1)/n \approx 1$,

$$s_n \approx S/n - H \qquad (7)$$

Lemma 1: Following the model and assumptions above and independent transmission of packets, for large n the total expected payload received with n packets given $B=S+H$ is:

$$f(n) \approx e^{\lambda H} [e^{-(\lambda S)/n} (S - nH)]. \qquad (8)$$

Proof: Follows from the definitions, and noting that the total expected payload received equals the sum of individual packet sizes times the probability that it is received:

$$f(n) = e^{-\lambda D}[n\ (S/n - (n-1)H/n)\ ] = e^{-\lambda(S/n-H)} (S - nH).$$

Since $(n-1)/n \approx 1$ for large n:

$$f(n) \approx e^{\lambda H} [e^{-(\lambda S)/n} (S - nH)].$$

Theorem 1: The number of packets n optimizing the expected amount of payload transmitted for the exponential model given $B=S+H$ is an integer equal to either:

$$\left\lfloor \frac{S\sqrt{\lambda^2H^2+4\lambda H}}{2H} - \frac{\lambda S}{2} \right\rfloor \text{ or } \left\lceil \frac{S\sqrt{\lambda^2H^2+4\lambda H}}{2H} - \frac{\lambda S}{2} \right\rceil \qquad (9)$$

Proof: Follows from optimizing the function in Lemma 1 and the fact that the number of packets is an integer.

Now, suppose that the probability a packet of size D is not lost is defined by: $a - \lambda D$; *i.e.*, a linear packet loss model is used with parameter λ and packet size as variable. This model is more meaningful in case the network characteristics follow the data in Figure 10. For this linear model it can be shown that:

Lemma 2: Following the linear model and assumptions above and independent transmission of packets, for large n the total expected payload received with n packets given $B=S+H$ is:

$$f(n) \approx [a - \lambda(S/n - H)]\ (S - nH) \qquad (10)$$

Proof: Follows from the definitions, and noting that the total expected payload received equals the sum of individual packet sizes times the probability that it is received:

$$f(n) = (a - \lambda D)[n\ (S/n - (n-1)H/n)].$$

Since $(n-1)/n \approx 1$ for large n:

$$f(n) \approx [a - \lambda\ (S/n - H)](S - nH).$$

Theorem 2: The number of packets optimizing the expected amount of payload transmitted for the linear model given $B=S+H$ is an integer equal to either:

$$\left\lfloor S\sqrt{\lambda/(aH+\lambda H^2)} \right\rfloor \text{ or } \left\lceil S\sqrt{\lambda/(aH+\lambda H^2)} \right\rceil \qquad (11)$$

Proof: Follows from optimizing the function in Lemma 2 and the fact that the number of packets is an integer.

For the data in Figure 10, we can observe that for the first part the curve fits a decreasing exponential function. If we only consider this part of the curve optimum point is the rightmost point because with increasing packet size (more right on the bottom axis) the overhead from total header sizes of all packets is lower and the packet loss is also lower.
For the second part, after the minimum point of the exponential part, we can fit a linear function $y = 0.8842 + 0.0006\,x$. Thus the probability of a packet not lost equals $(1-y/100) = 0.9912 - 0.000005683$ D; i.e., $a = 0.9912$ and $\lambda = 0.000005683$ in Theorem 2. Given S and H we can determine the optimum number of packets following Theorem 2 for the linear section of the graph in Figure 10.

The network packet size in our optimization strategy is independent of the processing performed at the application level; no matter how the application data, *e.g.* texture image and mesh information, are redistributed and segmented, the processed data, likely compress, are passed to the network as a byte stream (compressed or uncompress), which is then packed into the network packets. In our simulation, an IP header (8 bytes) and a UDP (20 bytes) are added to each network packet. 2 Mbytes application data was used in our packet loss experiment, the result of which is plotted in Figure 10. Substituting S = 2M bytes and H = 28 bytes in Equation (11) we obtain the optimal number of packets by Theorem 2 to be either 904 or 905 corresponding to a packet size of about 2,212 bytes. This shows that the optimal packet size for maximizing payload (actual multimedia data without packet headers) may not necessarily correspond to the packet size with lowest loss rate. For this experiment we assumed that the header for the multimedia data was not included in the packets. If we consider duplicating multimedia header information in packets for increased reliability under packet loss, H in the formula will increase giving a lower optimal number of packets or higher optimal packet size for this example.

VI. Conclusions and Future Work

In this paper, we discussed factors controlling 3D image degradation and outlined an approach for estimating perceptual quality considering variations in mesh and texture resolutions. A theoretical framework for determining the relative importance of texture *vs*. mesh was presented. An approach to optimizing perceptual quality under packet loss was then outlined. Experimental results were described to validate our approach. Finally, an approach for estimating the optimal packet size was proposed, following experimental results to collect real data on packet loss in congested wireless networks. In future work, we will extend and verify our packet size estimation method with more realistic models derived from tests over wireless networks, such as taking channel fading and burst error into consideration, to refine our assumptions. Implementations and user evaluations with handheld devices will also be conducted. Issues relating to MPEG4-3DMC compatibility [20], will be considered in our future work.

References

[1] P.S. Heckbert and M. Garland, "Survey of Polygonal Surface Simplification Algorithms," CMU, 1997.

[2] D. Luebke et al., "Level of Detail for 3D Graphics," Morgan Kaufmann, 2002.

[3] A. Bakre and B.R. Badrinath, "I-TCP: Indirect TCP for mobile hosts," *Int. Conf. Distributed Computing Systems (ICDCS)*, 136–143, 1995.

[4] D. Wu and R. Negi, "Effective Capacity: A Wireless Channel Model for Support of QoS," IEEE Trans. on Wireless Communications, 2002.

[5] G. Alregib, Y. Altunbasak and J. Rossignac, "Error-resilient transmission of 3D Models," ACM Trans. on Graphics, April 05. (Early version in ICASSP 02.)

[6] Z. Chen, B. Bodenheimer and J.F. Barnes, "Robust transmission of 3D geometry over wireless networks," Proceeding of the eighth international conference on 3D Web technology, 2003.

[7] K.V. Lee and S.T. Chanson, "Packet loss probability for real-time wireless communications," IEEE Trans. on Vehicular Technology, Nov. 2002.

[8] Y. Pan, I. Cheng and A. Basu, "Quality metric for approximating subjective evaluation of 3D objects," IEEE Trans. on Multimedia, April 2005. (Short version in IEEE Int. Conference on Image Processing 2003.)

[9] S. Yang et al. "Optimized mesh and texture multiplexing for progressive textured model transmission," ACM Multimedia, 2004

[10] D. Tian and G. AlRegib, "FQM: A fast quality measure for efficient transmission of textured 3D models," ACM Multimedia 2004.

[11] L. Balmelli, "Rate-distortion optimal mesh simplification and communication," Ph.D. dissertation, Ecole Polytechnique Federale de Lausanne, 2001.

[12] J. L. Mannos and D. J. Sakrison, "The Effects of a Visual Fidelity Criterion on Encoding of Images," IEEE Trans. on Information Theory, 1974.

[13] J. O. Limb, "Distortion Criteria of the Human Viewer," IEEE Transactions on SMC, Vol. 9, No.12, pp. 778-793, December 1979.

[14] R. Caceres and L. Iftode, "Improving the performance of reliable transport protocols in mobile computing environments," *IEEE J. Select. Areas Comm.*, vol. 13, pp. 850–857, June 1995.

[15] H. Hoppe, "Progressive meshes," SIGGRPAH 1996, LA, USA.

[16] A. C. Feng, A. C. Kapadia, W. Frng, and G. Belford, Packet Spacing - An Enabling Mechanism for Delivering Multimedia Content in Computational Grids, The Journal of Supercomputing, 23, 51 – 66, 2002

[17] Y. Gu and R. L. Grossman, Optimizing UDP-based Protocol Implementations, Workshop on Protocols for Fast Long-Distance Networks (PFLDNet) , Feb. 2005.

[18] X. Zhang and H. Schulzrinne, Voice over TCP and UDP, Technical Report, Department of Computer Science,Columbia University, 2004

[19] R. El-Azouzi and E. Altman, "A queuing analysis of packet dropping over a wireless link with retransmissions," Proc. PWC, Italy, 2003.

[20] ISO/IEC 14496-2:2000, Amendment 1, "Coding of audio-visual objects - Part 2: Visual version 2," 2000.

[21] S.Floyd, M.Handley, J.Padhye, and J.Widmer, "Equation-Based Congestion Control for Unicast Applications," ACM SIGCOMM'00, 2000.

[22] Y.Gu and R.Grossman, "Optimizing UDP-based Protocol Implementations," Third International Workshop on Protocols for Fast Long-Distance Networks (PFLDNet) , Lyon, France, Feb. 2005.

[23] I. Cheng, L. Ying and A. Basu, "A Perceptually Driven Model for Transmission of Arbitrary 3D Models over Unreliable Networks," 3DPVT 2006, Chapel Hill.

[24] P. Alliez and M. Desbrun. "Valence-Driven Connectivity Encoding of 3D Meshes," *EUROGRAPHICS*, 2001, pages 480-489.

[25] D. Tian and G. AlRegib, "FQM: A fast quality measure for efficient transmission of textured 3D models," *12th ACM Int'l Conf. on Multimedia* 2004.

[26] S. Yang, C. Lee and C. Kuo "Optimized mesh and texture multiplexing for progressive textured model transmission," *12th ACM Int'l Conf. on Multimedia*, 2004.