ADAPTIVE PLAYOUT SCHEDULING FOR VOIP USING THE K-ERLANG DISTRIBUTION

Haopeng Li, Guoqiang Zhang, and W. Bastiaan Kleijn

KTH-Royal Institute of Technology
10044 Stockholm, Sweden
haopeng@kth.se, {guoqiang.zhang, bastiaan.kleijn}@ee.kth.se

ABSTRACT

We propose a new adaptive playout scheme for VoIP. The k-Erlang distribution is introduced to model the packet inter-arrival time distribution. A cost function is proposed for the next played out packet in the buffer based on modelling packet-arrival times with the k-Erlang distribution. The cost function essentially balances the average buffering delay and the packet-loss rate. The optimal playout length of the packet is determined by minimizing the cost function and realized by either inserting or dropping pitch cycles from the packet. Our real-world data experiments show that our scheme outperforms two reference methods for both low-jitter and high-jitter cases.

1. INTRODUCTION

In recent years, the usage of internet telephony, commonly known as VoIP, has expanded rapidly. This service allows real-time voice communications between computer, mobile and regular phone. The service requires high transmission reliability and network stability for good quality. Packet loss and undesirable varying network delays, known as jitter, are the two main network impairments. It is known that the packet loss during transmission directly degrades the speech quality, and delay variation can introduce either underflow or overflow [1-4].

To reduce the packet loss ratio, a jitter buffer is commonly introduced [1-4] at the receiver side. The jitter buffer stores incoming packets before playout. A long buffer length evens out jitter significantly, but it introduces a negative effect on the conversational interactivity. However, a short buffer length may be not sufficient to eliminate the jitter. The problem of designing a proper buffer length is commonly referred to the playout scheduling problem. The objective of adaptive playout scheduling is to pursue an optimal trade-off between packet-loss rate and average buffering delay.

Existing methods for playout scheduling can be classified into two classes. A segment of speech can be separated into "talkspurts" and "silence periods" [1]. The first class only adjusts the lengths of silence packets, referred as *silence-oriented* adjustment. The scheme has the advantage that the adjustment does not influence the speech quality. However, when the talkspurts in a speech signal are quite long, the scheme is not effective for adjusting the buffer length. The second class of methods adjusts the playout lengths of both active packets and silence packets [2]. This scheme is referred to as *voice-oriented* adjustment. Compared to the first class, it provides better control of the buffer length. Motivated by this advantage we choose the voice-oriented adjustment for our approach.

In this paper we propose a new playout scheduling method by using the k-Erlang distribution modelling. The k-Erlang distribution has been widely used in queuing theory (e.g., Internet traffic modelling) [5] and in mathematical models for hydrology [9], as well as other fields. It also has been successfully applied in modelling the packet arrival process in video transmission [6]. However, in many applications the buffering delay is less important than a discontinuity in video transmission. Thus, the authors in [6] did not take delay into consideration explicitly. In contrast, in this paper we directly focus on balancing the cost of delay and packet loss rate.

We consider using a k-Erlang distribution to model the statistics of the inter-arrival time between successive packets. To our best knowledge, this distribution has not been used before for playout scheduling of VoIP. The distribution has the advantage that the jitter level of the packet network delay can be described accurately. To determine a proper playout length for the next played out packet in the buffer, a cost function is proposed. Then the optimal playout length of the consid-ered packet is determined for this cost function.

The remainder of this paper is organized as follows. Section 2 motivates the model of packet inter-arrival time by using a k-Erlang distribution. Section 3 derives the cost function to determine the optimal playout length of the considered packet. Implementation issues are discussed in Section 4. The verification of model and the comparison between our algorithm and the other two algorithms are in Section 5, followed by a short conclusion.

2. SYSTEM MODELLING

Speech packets are transmitted periodically from a sender with a production rate λ_{pac} (number of packets per ms). We denote the production interval time between two successive packets as T, which is also equal to the length of a packet. Because of the undesirable time-variation of network delays, the inter-arrival time at the jitter buffer fluctuates around the production time T. We denote the inter-arrival time between packets i and $i-1$ as X_i. Obviously, the expected length of inter-arrival time is $E[X_i] = T = 1/\lambda_{\text{pac}}$. We assume that X_i is independent and identically distributed (i.i.d), thus enabling a k-Erlang distribution model of X_i.

In this work, we use a k-Erlang distribution to model the packet inter-arrival process X_i. It was show in [6] that a k-Erlang distribution captures the jitter effect better than an exponential distribution.

The probability density function of a k-Erlang distribution is

$$f(t;k,\lambda) = \frac{\lambda^k t^{k-1} e^{-\lambda t}}{(k-1)!} \quad t,\lambda \geq 0. \tag{1}$$

© EURASIP, 2010 ISSN 2076-1465

We use $F^k(t)$ to denote the corresponding cumulative distribution function. The parameter k is called the *shape parameter* which is used to represent the jitter level. The parameter λ is called the *rate parameter*, which essentially determines the average packet arrival rate ($\lambda/k = 1/T$).

We choose moment estimation to estimate the parameters of an Erlang distribution. It is known that the moment estimation method is unbiased in this case [7]. Given a sequence of inter-arrival time ($x_1, x_2 \ldots x_n$), the moment estimation of the parameters $\hat{k}$ and $\hat{\lambda}$is given by

$$\begin{aligned} \hat{k} &= \frac{n \cdot \bar{x}^2}{\sum_{i=1}^{n} (x_i - \bar{x})^2}, \qquad (2) \\ \hat{\lambda} &= \frac{n \cdot \bar{x}}{\sum_{i=1}^{n} (x_i - \bar{x})^2}, \end{aligned}$$

where $\bar{x}$ is the average value of x_i, which is approximately $20ms$ when n is large.

Since a k-Erlang distribution can be obtained as a k-fold convolution of an exponential distribution, we can interpret the inter-arrival time X_i as a summation of k i.i.d. exponentially distributed random variables with mean value T/k. Denote these k variables as $Y_{ij}, j = 1,2\ldots k$. Then we have $X_i = \sum_{j=1}^{k} Y_{ij}$. It is immediate that each sequence $Y_{ij}, j = 1,2\ldots k$ is a Poisson process. Thus, k independent Poisson processes are coupled to render a k-Erlang distribution.

The level of network jitter is reflected by the variance of X_i, i.e.,$Var[X_i] = k \cdot Var[Y_{ij}] = T^2/k$. When k is large, the packet-arrival process has a narrow distribution. On the other hand, a small k indicates that the distribution is broad. This means the delay trace exhibits high jitter level. To summarize, the parameter k can be used to describe the jitter level.

3. OPTIMAL PLAYOUT STRATEGY

In this section, a cost function will be constructed. Then, the optimal playout length of the considered packet will be derived by minimizing the cost function. Finally, we will discuss the proper choices of the weighting coefficients in the cost function.

3.1 Cost function

In this subsection we propose a cost function that reflects the conversational quality. To build the cost function, we consider three measures that are known to be relevant to the conversational quality. The measures are functions of the playout length of the next played out packet in the buffer.

We first give motivations for the three measures. Generally speaking, the underflow risk becomes relatively high when there is only one packet in the jitter buffer. The human perceptual system is sensitive to the playout cut-off caused by underflow. In such situations, it is desirable to extend the playout length of the packet (to reduce the underflow risk). On the other hand, the buffering length could become large at some time due to "delay spikes". In this situation, it is desirable to speed up the playout rate to reduce the buffering delay. However, delay adjustments introduce distortion to the speech. Thus, a penalty on the adjustment should also be introduced into the cost function.

We are now in a position to introduce the three measures. The first measure, *"expected buffering length"*, is a measurement of the expected buffer length (in *ms*) after the current packet is played out. It penalizes long delays that degrade conversational quality. The second measure is called *"adjustment distortion"*, which measures the distortion introduced to the current packet by extending or compressing the packet length. The third measure is *"average waiting time"*, which measures the expected time it takes for a new packet to arrive after the current packet is played out. It is taken into account only when the current packet is the last one in the buffer. In this situation, "average waiting time" measures the duration of probable underflow.

We now describe these three measures in detail. We derive the analytic expression for each of them by using the k-Erlang distribution. We denote the number of packets currently stored in the buffer as $n_{\mathrm{pac}} + 1$, and the playout length of the next played out packet in the buffer as D_{p}.

We first consider the expected buffering length. We know from the k-Erlang distribution that the packet arrival rate is λ/k. Thus, the expected number of arriving packets is $\lambda D_{\mathrm{p}}/k$. There are n_{pac} packets left in the buffer in addition to the current one. The cost should increase non-linearly with the increasing of n_{pac}. Therefore, we define the cost introduced by the expected buffering length as

$$J_1(D_{\mathrm{p}}) = \left(n_{\mathrm{pac}} \cdot T + \frac{\lambda D_{\mathrm{p}}}{k} \cdot T \right)^2. \qquad (3)$$

Since $\lambda/k = 1/T$, the cost J_1 can be simplified to

$$J_1(D_{\mathrm{p}}) = \left(n_{\mathrm{pac}} \cdot T + D_{\mathrm{p}} \right)^2. \qquad (4)$$

The second measure "adjustment" is relatively easy to deal with. We define the cost of buffer-length adjustment as

$$J_2(D_{\mathrm{p}}) = \left(D_{\mathrm{p}} - T \right)^2. \qquad (5)$$

The third measure "average waiting time" is considered only when one packet is stored in the buffer. It measures the expected time it takes to wait for the next packet arriving after current packet is played. We define:

$$d = \begin{cases} 0, & t < D_{\mathrm{p}} \\ t - D_{\mathrm{p}}, & t \geq D_{\mathrm{p}} \end{cases} . \qquad (6)$$

The expectation of the waiting time is thus given as:

$$\begin{aligned} E[d] &= \int_{D_{\mathrm{p}}}^{\infty} (t - D_{\mathrm{p}}) \cdot f(t; k, \lambda) dt \qquad (7) \\ &= \frac{k}{\lambda} \cdot \left[1 - F^{k+1}(D_{\mathrm{p}}) \right] - D_{\mathrm{p}} \cdot \left[1 - F^k(D_{\mathrm{p}}) \right]. \end{aligned}$$

Similar to the definition of J_1 and J_2, we use square of the average waiting time $E[d]$, to define the third cost

$$J_3(D_{\mathrm{p}}) = \left[\frac{k}{\lambda} \cdot \left[1 - F^{k+1}(D_{\mathrm{p}}) \right] - D_{\mathrm{p}} \cdot \left[1 - F^k(D_{\mathrm{p}}) \right] \right]^2 . \qquad (8)$$

It can be viewed as a measurement of the expected underflow risk.

Based on the above three introduced measures, a global cost function can then be proposed. We define the global cost function as a weighted summation of the above three sub-cost functions:

$$J(D_\mathrm{p}) = \begin{cases} w_1 \cdot J_1 + w_2 \cdot J_2 + w_3 \cdot J_3, & \text{one packet} \\ w_1 \cdot J_1 + w_2 \cdot J_2, & \text{otherwise} \end{cases} \qquad (9)$$

where w_1, w_2, w_3 are the nonnegative weighting coefficients in the cost function. We will explain how to choose their values in subsection 3.3. The optimal playout length D_opt is defined as the one that minimizes the cost function in (9), expressed as

$$D_\mathrm{opt} = \arg\min_{D_\mathrm{p}} J(D_\mathrm{p}). \qquad (10)$$

3.2 Minimizing the cost function

In this subsection, we consider the derivation of the optimal playout length D_opt in (10). We first study the convexity of the global cost function (9). It is obvious that the two costs J_1 and J_2 are convex functions since they have quadratic expressions.

We now study this convexity of J_3. First, we compute the first and second derivatives of $E[d]$.

$$\frac{\partial E[d]}{\partial D_\mathrm{p}} = -\left[1 - F^k(D_\mathrm{p})\right] < 0, \qquad (11)$$

$$\frac{\partial^2 E[d]}{\partial D_\mathrm{p}^2} = f(t;k,\lambda) > 0. \qquad (12)$$

By using the chain rule, we arrive at

$$\frac{\partial J_3}{\partial D_\mathrm{p}} = 2 \cdot E[d] \cdot \frac{\partial E[d]}{\partial D_\mathrm{p}} < 0, \qquad (13)$$

$$\frac{\partial^2 J_3}{\partial D_\mathrm{p}^2} = 2 \cdot \left(\frac{\partial^2 E[d]}{\partial D_\mathrm{p}^2}\right) + 2 \cdot E[d] \cdot \frac{\partial^2 E[d]}{\partial D_\mathrm{p}^2} > 0. \qquad (14)$$

From (13) and (14), it is immediate that the term J_3 is a convex function over D_p Consequently, J, a nonnegative weighted sum of convex functions in (9) is convex [8].

We now discuss how to find the optimal value D_opt in (10). For the case with only one packet in the buffer, the cost function appears complicated. We use a greedy search method to find the optimal solution. For the other case, the global cost function is quite simple. The expression of the optimal playout length takes the form:

$$D_\mathrm{opt} = \frac{T \cdot \left(1 - \frac{w_1}{w_2} \cdot n_\mathrm{pac}\right)}{1 + \frac{w_1}{w_2}}. \qquad (15)$$

3.3 Boundaries of the weighting coefficients

In subsection 3.2, the optimal playout length D_opt was derived. We now explain how to choose proper weighting coefficients in the global cost function J. Two constraints will be considered to derive the boundaries of the weighting coefficients.

The first constraint is related with reducing the underflow risk. Specifically, when there is only one packet in the buffer, we let $D_\mathrm{opt} > T$. Note that J is a convex function of D_p. This property guarantees that when $D_\mathrm{p} > D_\mathrm{opt}$ the first derivative of J is positive, and when $D_\mathrm{p} \le D_\mathrm{opt}$, the first derivative of J is non-positive. Thus, the first derivative of J at $D_\mathrm{p} = T$ is non-positive. In addition, the first derivative of J_2 at $D_\mathrm{p} = T$ is zero. Hence, we arrive at:

$$\frac{w_3}{w_1} \ge \frac{T}{[1 - F^k(T)] \cdot \left[\frac{k}{\lambda} \cdot [1 - F^{k+1}(T)] - D_\mathrm{p} \cdot [1 - F^k(T)]\right]}. \qquad (16)$$

We note that $F^k(T)$ and $F^{k+1}(T)$ are the probabilities that a packet arrives in a packet production period for two different k values. It is obvious that they are close to 1. In addition, the ratio k/λ is fully determined by the packet production rate. Thus, the lower bound of w_3/w_1 in (16) can be properly approximated. This is appropriate for any trace.

Next we impose a constraint to find the relationship between w_1 and w_2. In the case that more than one packet is stored in the buffer, the optimal playout length takes the form (15). We let $D_\mathrm{p} > D_\mathrm{cont}$, for a range of n_pac within $[1, n_\mathrm{max}]$. The parameter D_cont is a threshold on the playout length. Considering the extreme case of n_max in (15), we obtain a lower bound of the ratio between w_1 and w_2 as:

$$\frac{w_2}{w_1} > \frac{T \cdot n_\mathrm{pac} + D_\mathrm{cont}}{T - D_\mathrm{cont}}. \qquad (17)$$

By empirical study, we find that $n_\mathrm{max} = 50$ provides good performance in most cases. Due to the network dynamics, it might happen that the number of packets in the buffer is more than n_max. In this situation, the optimal playout length of the considered packet is then set to be D_cont.

4. IMPLEMENTATION ISSUES

4.1 Implementation of algorithm

In practice, a speech segment usually exhibits periodicity. Thus, the playout length of the considered packet should be carefully chosen such that it is close to optimal playout length D_opt, and the processed speech still preserves periodicity as before.

In the implementation, we consider two scenarios. The first scenario refers to the situation that one or more packets stored in the buffer. We check if the next played out packet contains silence or active speech. For the active speech, if it is a voiced packet, we estimate the pitch period from the speech segment in the packet using pitch analysis algorithm [10], if it is an unvoiced packet, we assume a typical pitch length for it. Then we either subtract from or add integer multiple of the pitch cycles to the segment to produce a playout length closest to D_opt. On the other hand, if it is a silence packet, we simply set the playout length to D_opt.

In the second scenario no packet remains in the buffer. Then we apply a special procedure to fill the playout gap. If the last played packet contains active speech, we insert an integer number of pitch cycles from the last packet into the buffer until a new packet comes. If the last played packet contains silence speech, we only insert a silence segment into the buffer until a new packet arrives. In addition, it may happen that a packet is lost during transmission, which means that another packet with higher sequence index arrives before the packet. We then simply insert an integer number of pitch cycles or a silence segment to reconstruct the lost packet.

4.2 Performance measurements

The packet-buffering delay and packet adjustment ratio are two important measurements in evaluating the performance of a playout scheduling method. The low values of both lead to a higher conversational quality. The packet-buffering delay is measured by averaging the differences between packet arriving time and packet playout time, defined as:

$$De = \frac{\sum_{i\subseteq N}\left(t_{i,\text{playout}} - t_{i,\text{arrive}}\right)}{N}, \qquad (18)$$

where we denote the number of packets in the delay trace as N.

Next, we introduce the packet adjustment ratio. We denote the round-off version of optimal playout length as $D^{'}_{\text{opt}}$, the length of pitch cycles of packet i as l_i, the number of pitch cycles that are used to conceal the gap as n_i. The first term (case A) of the packet adjustment ratio is the adjustment of playout length. The second term (case B) is the number of pitch cycles that are used to conceal the underflow. The third term (case C) is the number of pitch cycles inserted to compensate for packets lost by the network. Note that we select the larger value between compensated pitch cycles and T to calculate the adjustment for packets lost by the network.

$$Ar = \frac{\sum_{i\subseteq \text{A}}\left|D^{'}_{\text{opt}} - T\right| + \sum_{i\subseteq \text{B}} n_\text{i}\cdot \text{l}_\text{i} + \sum_{i\subseteq \text{C}} \max(n_\text{i}\cdot \text{l}_\text{i}, T)}{\sum_{i\subseteq \text{active}} T}. \qquad (19)$$

5. EXPERIMENTS

In this section, we first verify the accuracy of using a k-Erlang distribution model. Then we compare the performance between our algorithm and two other reference algorithms.

5.1 Model verification

We tested 26 different traces which were collected from different remote locations, and found that inter-arrival histograms extracted from these traces can be matched accurately by appropriate k-Erlang distributions, especially in low jitter case. We used the Kullback-Leibler divergence to measure the difference between a k-Erlang distribution and a real data distribution. We discretized the k-Erlang distribution into a probability mass function to make it comparable to the real data distribution.

There are two typical cases in the modelling (Figure 1). One is for the low jitter case (corresponding to order 47, the K-L divergence is 0.0110), and the other is for high jitter case (corresponding to order 7, the K-L divergence is 0.0483).

The K-L divergence between the k-Erlang distribution and a real data distribution decreases with increasing order of Erlang distribution (Figure 2). This verifies that the k-Erlang distribution modelling is accurate for both low and high jitter cases.

5.2 Performance comparison

We compared the performance of our proposed playout scheme with that of two existing methods. Algorithm 1 [1] is based on the silence-oriented adjustment. If the current packet is checked to be a silence packet, the absolute playout delay (the end-to-end network delay plus the buffering delay) of this packet is set to be $\hat{d}_i + \beta\cdot\hat{v}_i$, where $\hat{d}_i$ and $\hat{v}_i$ are the estimated network delay and variation. We adjust β to control the buffering delay and the packet loss ratio. Note that in this silence-oriented adjustment, the packet loss ratio is only related to underflow and and packets lost by the network.

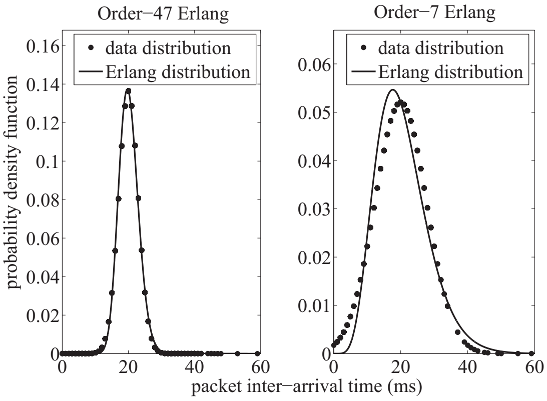

Figure 1: Illustration of the k-Erlang distribution modelling of the inter-arrival time.

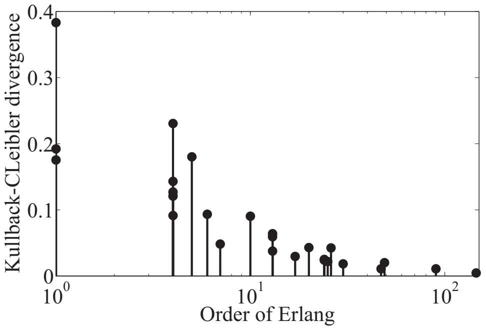

Figure 2: The order of the Erlang distribution and the Kullback-Leibler divergence.

Algorithm 2 is based on voice-oriented adjustment [2]. The algorithm uses the statistics of the past end-to-end network delay trace to determine the optimal playout length of the considered packet. When it is a active packet, the segment length will be scaled to meet the optimal playout length.

For our experiments, delay traces were collected by selecting different remote locations (in Table 1). Each trace consists of 10000 packets, and each packet consists of 20 ms speech content. A real speech segment from a male speaker was imposed on each delay trace in the test. The weighting coefficients were selected based on (16) and (17) in subsection 3.3: $w_3/w_1 = 80$ and w_2/w_1 is from 50 to 160.

Table 1: Collected network delay traces

Trace	Hosts' Location	Remote Host IP
Trace01	Paris → Stockholm	80.14.55.252
Trace02	Massachusetts → Stockholm	128.31.1.14
Trace03	Stockholm → Berkeley	169.229.50.12
Trace04	Stockholm → Hongkong	137.189.98.30

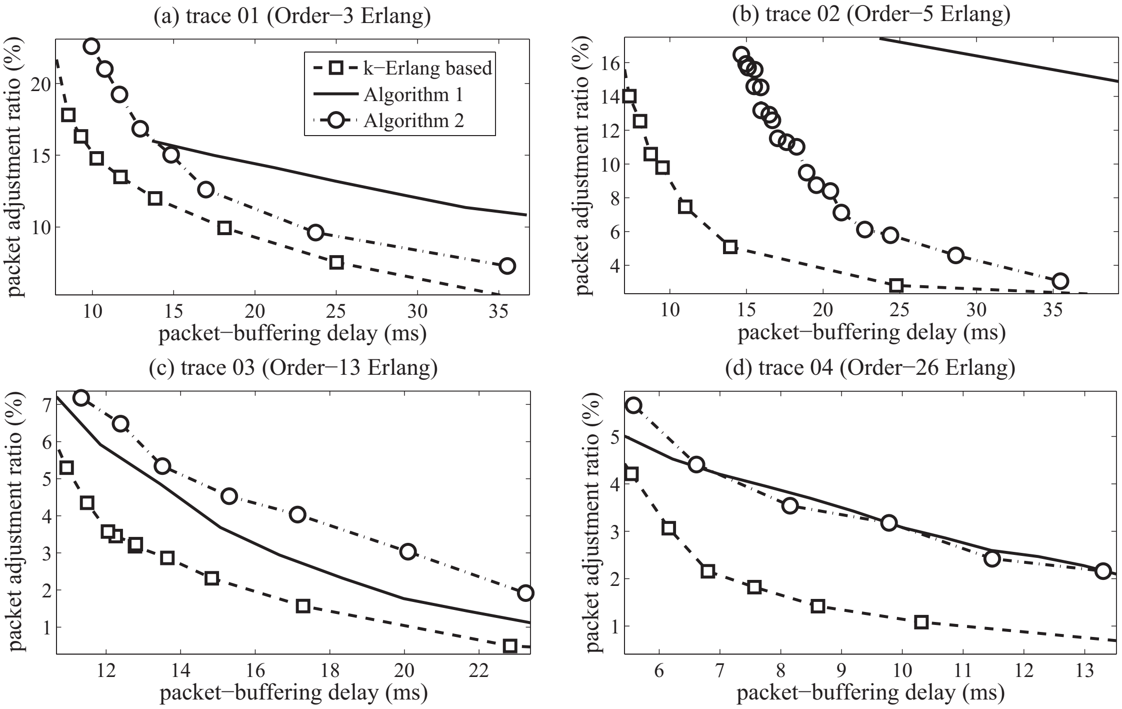

Figure 3: Performance comparison of the algorithms in high jitter and low jitter conditions. The subplot (a) and (b) are high jitter cases, the subplot (c) and (d) are low jitter cases. The labels of the curves in subplots (b), (c) and (d) are the same as those of (a).

From Figure 3, one observes that our proposed algorithm outperforms the reference algorithms in both high and low jitter cases. Especially when the jitter level is high, the performance gain is significant. It is likely that in a high jitter situation, the network delays of adjacent packets are less correlated. Thus the prediction accuracy of the other two algorithms is low. Our algorithm is more suitable in this situation.

6. CONCLUSIONS

We conclude that a k-Erlang distribution can describe the jitter level well. We construct a cost function with the playout length of the next played out packet in the buffer as its variable. It takes into account the future expected buffering length, the artificial adjustment length, and the expected underflow risk. The optimal playout length of the packet is determined by minimizing the cost function. The experimental results show that our method outperforms two classical approaches in both low jitter and high jitter cases.

REFERENCES

[1] R. Ramjee, J. Kurose and D. Towsley, “Adaptive playout mechanisms for packetized audio applica-tions in wide-area networks,” *Proc. of the IEEE Infocom*, vol. 2, pp. 680–688, Jun. 1994.

[2] Y. J. Liang, N. Farber and B. Girod, “Adaptive playout scheduling and loss concealment for voice communications over IP networks,” *IEEE Transactions on Multimedia*, vol. 5(4), pp. 532–543, Dec. 2003.

[3] G. Zhang, H. Lundin and W. B. Kleijn, “Band control policy of playout of scheduling for voice over,” *Proc. EUSIPCO 2008*, Aug. 2008.

[4] L. Sun, E. Ifeachor, “New models for perceived voice quality prediction and their applications in playout buffer optimization for VoIP networks,” *Proc. IEEE Communication*, vol. 3, pp. 1478–1483 , June. 2004.

[5] H. Chen and D. D. Yao, *Fundamentals of Queueing Networks: Performance, Asymptotics, and Optimization.* New York: Springer-Verlag, 2001.

[6] N. Laoutaris, B. V. Houdt, and I. Stavrakakis, “Optimization of a packet video receiver under different levels of delay jitter: An analytical approach,” *Performance Evaluation*, vol. 55(3-4), pp. 251–275, 2004.

[7] J. Bai, A. J. Jakeman and M. McAleer, “A new approach to maximum likelihood estimation of the three-parameter Gamma and Weibull distributions,” *Australian J. Statistics*, vol. 33, pp. 397–410, 1991.

[8] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge University Press, 2004.

[9] S. Yue, T. Ouarda and B. Bobee, “A review of bivariate gamma distribution for hydrological application,” *Journal of Hydrology*, vol. 246, pp. 1–18, Jan. 2001.

[10] A. D. Cheveigne and H. K. Yin, “ A fundamental frequency estimator for speech and music,” *The Journal of the Acoustical Society of America*, vol. 111, pp. 1917–1930, Apr. 2002.