SHORT COMMUNICATIONS

Differential AR algorithm for packet delay prediction*

JIAO Liangbao[1,2**], ZHANG De[1,2] and BI Houjie[2]

(1. Institute of Acoustics, State Key Laboratory of Modern Acoustics, Nanjing University, Nanjing 210093, China; 2. Institute of Communication Technique, Nanjing University, Nanjing 210093, China)

Received July 21, 2005; revised November 1, 2005

Abstract Different delay prediction algorithms have been applied in multimedia communication, among which linear prediction is attractive because of its low complexity. AR (auto regressive) algorithm is a traditional one with low computation cost, while NLMS (normalize least mean square) algorithm is more precise. In this paper, referring to ARIMA (auto regression integrated with moving averages) model, a differential AR algorithm (DIAR) is proposed based on the analyses of both AR and NLMS algorithms. The prediction precision of the new algorithm is about 5—10 db higher than that of the AR algorithm without increasing the computation complexity. Compared with NLMS algorithm, its precision slightly improves by 0.1 db on average, but the algorithm complexity reduces more than 90%. Our simulation and tests also demonstrate that this method improves the performance of the average end-to-end delay and packet loss ratio significantly.

Keywords: **AR algorithm, NLMS algorithm, differential AR algorithm, ARIMA model.**

Studies on packet delay prediction algorithms, which come from QoS (quality of service) guarantee of VOIP (voice over IP), now have been extended to all the network multimedia applications. The main problem existing in the multimedia communication is the network delay and jitter. The purpose of algorithm research is to reduce the packet loss ratio as far as possible under the condition of ensuring the average end-to-end delay small enough[1]. Meanwhile, the computation complexity is also an important factor in algorithm evaluation because of the real time demands.

The delay prediction algorithms can be classified into two categories: one is based on linear prediction[2—7] and the other is based on statistics probability analysis[8—10]. Linear prediction algorithms which have lower complexity have been widely applied to industrialization. Among them, AR (auto regressive)[2] algorithm, based on AR model which is a special case of ARMA (auto regressive move average) model[11], is a traditional one with low computation cost; while NLMS (normalize least mean square)[5] algorithm, based on LMS (least mean square)[11] model, is a comparatively precise algorithm. Recently, studies on AR algorithm are concentrated on improving its prediction precision while keeping its low computation cost. In this apper, based on ARIMA (auto regression integrated with moving averages) model, DIAR algorithm is proposed to compensate the weakness of the above two algorithms. In the proposed algorithm, by pre-processing difference of the delay series, the prediction precision is significantly improved while keeping the advantage of simple computation, which is better than NLMS algorithm.

1 Analysis of the linear prediction model and observations

The ARMA(p, q) model and AR(p) model can be described respectively as

$$\Phi_p(\boldsymbol{B})z_t = \theta_0 a_t + \theta_1 a_{t-1} + \cdots + \theta_q a_{t-q}, \quad (1)$$

$$\Phi_p(\boldsymbol{B})z_t = \theta_0 a_t, \quad (2)$$

where $\Phi_p(\boldsymbol{B})$ is a p-order autoregressive operator, and $\boldsymbol{B}$ is the coefficient vector; the stationary random series z_t is a random variable needing to be predicted; a_t is a white noise series; $\theta_0 \cdots \theta_q$ are the moving average coefficients. From Eqs. (1) (2), we can see that ARMA model will become AR model when $\theta_1 = \theta_2 = \cdots = \theta_q = 0$, that is to say, AR model is a special case of ARMA model.

* Supported by National Natural Science Foundation of China (Grant No. 10234060)

** To whom correspondence should be addressed. E-mail: ljiaotom@tom.com, deedez@nju.edu.cn

In AR algorithm[2], the estimation of packet delay is calculated by

$$d_t = \alpha d_{i-1} + (1-\alpha) n_i, \quad (3)$$

and the packet delay variation and total end-to-end delay are

$$\begin{cases} v_i = \alpha v_{i-1} + (1-\alpha) \mid d_i - n_i \mid \\ D_i = d_i + \beta v_i \end{cases}, \quad (4)$$

where d_i is the autoregressive estimation of packet delay, v_i is the delay variation, n_i is the actual delay of packet i, α is a weighting factor which controls the adaptation rate, and D_i is the total end-to-end delay. To avoid the fact that the estimation of packet delay is smaller than the actual value, the compensation factor βv_i should be added to the estimation, where β is a safety factor used to the trade-off between the total end-to-end delay and packet loss ratio. The suggested values of α and β are 0.998002 and 4, respectively. Initially d_0 is set to be n_0, and v_0 is set to be $n_0/2$[2].

From Eqs. (2) (3), it can be seen that AR algorithm is actually one-order autoregressive estimation, i.e. $p=1$, where $a_t = (1-\alpha) n_i$ and $\Phi(\boldsymbol{B}) = 1 - \alpha \boldsymbol{B}$. To ensure the prediction precision of the AR algorithm, n_t should be a white noise series with normal distribution, which could generally use a homogeneous stationary time series instead.

However, in an actual network multimedia system, the packets are lined in queue when they pass through routers. According to the queue theory, the various delays of consequent packets are correlated in time and trend when network congestion occurs. The packet delay series is therefore a homogeneous non-stationary time series, which does not satisfy the requirement of ARMA model. According to ARIMA model[11] (Box-Jenkins) used in economics estimation, the correlation in trend and time of homogeneous non-stationary time series can be reduced by differencing operation, and then series can be converted to homogeneous stationary time series.

Differencing can reduce the obvious trend and periodicity correlation, which can be demonstrated by comparing the original packet delay series and the series after differencing (Figs. 1, 2). The curve in Fig.1 is the one unprocessed, and the time series is obtained by pinging from Nanjing University to Harvard University. The time correlation and trend can be obviously found. After differencing, the new series are shown in Fig.2, whose curve is closer to the homogeneous stationary series. It is obvious that the series after differencing is more satisfying to the requirement of ARMA model and AR model.

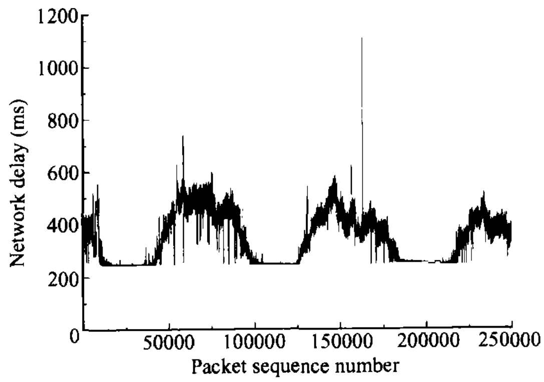

Fig. 1. Initial series of Harvard trace.

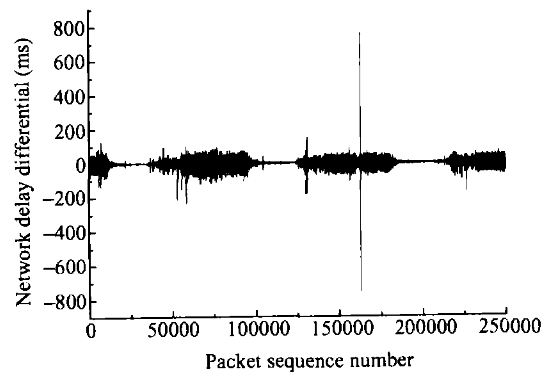

Fig. 2. Harvard trace after differencing.

2 Differential AR algorithm

As described above, network delay series is generally a non-stationary random process, so a significant error exists in predicting delay of network congestion when traditional AR is used. From Eq. (3), we have

$$d_i - d_{i-1} = \alpha(d_{i-1} - d_{i-2}) + (1-\alpha)(n_i - n_{i-1}). \quad (5)$$

Let $\Delta d_i = d_i - d_{i-1}$, $\Delta n_i = n_i - n_{i-1}$, we can obtained

$$\Delta d_i = \alpha \Delta d_{i-1} + (1-\alpha) \Delta n_i. \quad (6)$$

Now, by performing difference operation on Eq. (4) and replacing D_{i-1} with the actual delay n_{i-1} of packet $i-1$, the delay difference jitter Δv_i and end-to-end delay D_i are computed by

$$\begin{cases} \Delta v_i = \alpha \Delta v_{i-1} + (1-\alpha) \mid \Delta d_i - \Delta n_i \mid \\ D_i = D_{i-1} + \Delta D_i = D_{i-1} + \Delta d_i + \beta \Delta v_i \\ \quad = n_{i-1} + \Delta d_i + \beta \Delta v_i. \end{cases} \quad (7)$$

Eqs. (6) and (7) constitute the differential AR algorithm, shortly named DIAR.

In addition, the actual packet delay in Eq. (3) was obtained by means of send-receiver syncs or NTP (network time protocol) protocol, which may cause syncs error. However, Eq. (8) shows that the error of input series only depends on the machine clock precision and does not relate to sync error when the difference series is used as input, where $T_{R,i}$ represents receive time of packet i and $T_{S,i}$ represents its send time. Therefore, DIAR algorithm can prevent generation of syncs error:

$$\begin{aligned}\Delta n_i &= n_i - n_{i-1} \\ &= (T_{R,i} - T_{S,i}) - (T_{R,i-1} - T_{S,i-1}) \\ &= (T_{R,i} - T_{R,i-1}) - (T_{S,i} - T_{S,i-1}). \quad (8)\end{aligned}$$

3 Simulation and performance comparison

To compare the performance of the DIAR, NLMS and AR algorithms, we used 32 groups of data obtained by pinging from Nanjing University to eight different universities in USA and UK at different moments. All these ping operations lasted over 24 hours. The total amount of data exceeds 4 millions (Table 1).

To compare the prediction precision of AR, NLMS and DIAR algorithms, the prediction signal-to-error ratio (SRR) is adopted as a reference. SRR is defined as

$$SRR = 10\log_{10}\frac{E[y^2(n+1)]}{E[e^2(n+1)]}(\text{db}), \quad (9)$$

where $y(n+1)$ is the actual network delay of the $(n+1)$th packet, $\hat{y}(n+1)$ is its delay estimation, and $e(n+1) = y(n+1) - \hat{y}(n+1)$ is the prediction error.

Table 1. Network delay traces for simulations and tests

Destination	Time span 1			Time span 2			Time span 3			Time span 4		
	Packet number	$E(d)$ 1)	$D(d)$ 2)	Packet number	$E(d)$ 1)	$D(d)$ 2)	Packet number	$E(d)$ 1)	$D(d)$ 2)	Packet number	$E(d)$ 1)	$D(d)$ 2)
CAM_AC	90758	349.12	9.42	43430	347.40	0.68	85190	351.21	37.50	251151	349.38	15.60
CMU	91678	377.69	99.03	43442	326.22	51.41	85148	408.71	130.21	250259	421.07	130.41
Columbia	90686	366.42	90.13	43234	326.21	51.50	84955	402.21	129.09	248527	463.20	109.76
Harvard	90748	323.65	59.61	43341	291.21	44.77	84961	346.61	95.04	251033	343.99	90.57
MIT	91173	310.54	69.21	43585	253.29	13.97	84104	263.84	29.50	246578	253.60	17.22
Stanford	91182	262.80	75.78	43532	251.47	55.84	85041	308.78	100.05	249860	293.45	88.72
USC	91729	348.73	96.65	43331	235.77	45.72	85073	298.36	101.79	249934	283.29	85.64
Washington	91227	253.52	69.61	43322	196.00	13.98	84064	209.81	32.02	247327	195.14	17.31

1) $E(d)$, the average actual delay of each series, the unit is ms
2) $D(d)$, the mean square error(MSE) of each series, the unit is ms

Table 2 compares the prediction precisions when the AR, NLMS and DIAR algorithms are adopted to deal with the series in Table 1. The SRR value is the average of four time spans.

Table 2. SRR comparison of DIAR & NLMS Unit: db

Destination	CAM_AC	CMU	Columbia	Harvard	MIT	Stanford	USC	Washington
AR	31.930	20.237	17.286	20.137	22.964	8.027	18.396	20.431
NLMS	35.698	30.664	26.890	27.331	27.869	15.847	28.085	25.080
DIAR	35.713	30.703	27.035	27.403	27.937	16.143	28.157	25.183

Table 3 is the number of the variables and operations used in the DIAR and the NLMS algorithms.

Table 3. Number of variables and operations of DIAR & NLMS

Algorithm	Variables number	Operations number	
		Divide	Multiple
DIAR/AR	3	0	3
NLMS	38	18	93

From Tables 2 and 3, we can see that $E(d)$ and $D(d)$ vary critically. However, the performance of DIAR is improved significantly. Compared with the AR algorithm, SRR obtained by DIAR algorithm is 5—10 db higher. In light of the NLMS algorithm, although SRR only slightly increases about 0.1 db, the computation cost decreases a lot (Table 3). Details of the NLMS algorithm may be referred to Eqs. (4) and (5) in Ref. [5]. Particularly for the data groups of CMU, Harvard and USC, the performance of DIAR improves more significantly because the series varies more seriously and the time correlation is stronger.

The curves of end-to-end delay versus the packet loss ratio are plotted according to the AR/NLMS/DIAR algorithms, respectively. The curves all demonstrate the significant improvement of the DIAR algorithm compared with the AR algorithm. The performance of the DIAR algorithm is close to the NLMS algorithm with only 10% computation cost. Two examples are shown in Fig. 3 and Fig. 4.

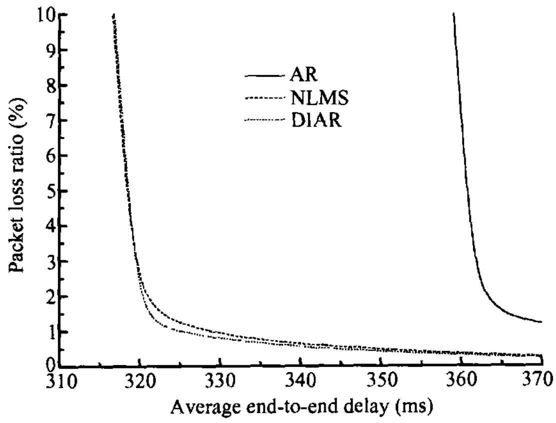

Fig. 3. Average end-to-end delay versus the packet loss ratio Group 4 of Stanford trace.

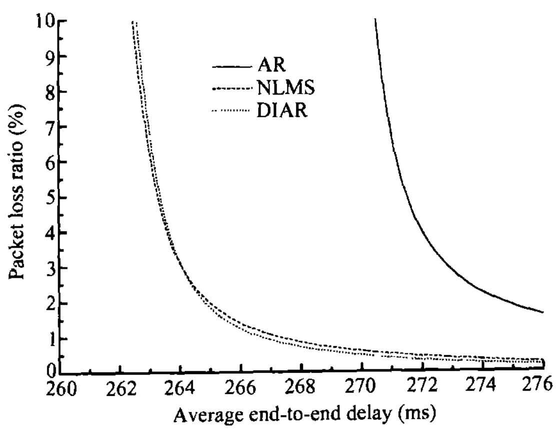

Fig. 4. Average end-to-end delay versus the packet loss ratio Group 4 of MIT trace.

4 Conclusion

This paper mainly focuses on AR and NLMS algorithms based on linear prediction. The DIAR algorithm has been proposed to compensate the weakness of both AR and NLMS algorithms based on the ARIMA model. In the proposed method, difference of the delay series is pre-processed to satisfy the requirement of ARMA model, which significantly improves the prediction precision. Compared with the traditional AR, the proposed method improves the prediction precision SRR by 5—10 db without increasing the computation complexity. Compared with NLMS, the proposed method possesses the advantage of simple computation, while the prediction precision has also increased about 0.1 db on average. Simulation and tests on actual data groups have demonstrated that the DIAR algorithm can optimize the performances of both end-to-end delay and packet loss ratio.

Acknowledgement The authors would like to thank Prof. Zheng Baoyu of Nanjing University of Posts & Telecommunications for his helpful suggestions.

References

1 Sreenan C. J., Chen J. C., Agrawal P. et al. Delay reduction techniques for playout buffering. IEEE Transactions on Multimedia, 2000, 2(2): 88—100.

2 Ramjee R., Kurose J., Towsley D. et al. Adaptive playout mechanisms for packetized audio applications in wide-area networks. In: Proceedings of IEEE INFOCOM. Toronto, Canada, Jun., 1994, (2): 680—688.

3 Liang Y. J., Farber N. and Girod B. Adaptive playout scheduling and loss concealment for voice communications over IP networks. IEEE Transactions on Multimedia, 2003, 5(4): 532—543.

4 Kansal A. and Karandikar A. Adaptive delay estimation for low jitter audio over internet. In: Proceedings IEEE Global Telecommunication Conference. San Antonio, USA, Nov., 2001, (4): 2591—2595.

5 DeLeon P. and Sreenan C. An adaptive predictor for media playout buffering. In: Proceedings IEEE International Conference Acoustics, Speech, Signal Processing. Phoenix, USA, Mar., 1999, 3097—3100.

6 Shallwani A. and Kabal P. An adaptive playout algorithm with delay spike detection for real-time VOIP. In: Proceedings IEEE Canadian Conf. Electrical, Computer Engineering. Montreal, Canada, May, 2003, 997—1000.

7 Jeon W. J., Nahrstedt K., Basar T. et al. Robust delay estimation for internet multimedia applications. In: Proceedings of the Joint International Workshop on Interactive Distributed Multimedia Systems Protocols for Multimedia Systems, Nov., 2002, 255—262.

8 Moon S. B., Kurose J. and Towsley D. Packet audio playout delay adjustment: Performance bounds and algorithms. Multimedia Systems, 1998, 6: 17—28.

9 Pinto J. and Christensen K. J. An algorithm for playout of packet voice based on adaptive adjustment of talkspurt silence periods. In: Proceedings 24th Conference on Local Computer Networks. Lowell, USA, Oct., 1999, 224—231.

10 Marjamaki H. and Kantola R. Performance evaluation of an IP voice terminal. In: Proceedings Fifth IFIP Conference on Intelligence in Networks, Asian Institute of Technology. Thailand, Nov., 1999, 349—362.

11 Box G. and Jenkins G. Time Series Analysis, Forecasting and Control (3rd ed.). Oakland: Holden-Day, 1994, 278—290.