A Poker-Game-Based Feedback Suppression Algorithm for Satellite Reliable Multicast

Sungrae Cho and Ian F. Akyildiz

Broadband and Wireless Networking Laboratory
School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA 30332
Tel: (404) 894-6616; Fax: (404) 894-7883
Email: {srcho, ian}@ece.gatech.edu

Abstract—**In this paper, a poker-game-based feedback suppression (PFS) algorithm is proposed for scalable satellite reliable multicast protocols. An analytical model is provided for the feedback suppression performance of the PFS scheme. This model is validated by simulation results. Numerical examples show that the feedbacks can be effectively suppressed by introducing PFS algorithm.**

I. Introduction

SATELLITE networks will have a crucial role for global multicast services due to their large coverage area, abundant bandwidth particularly at higher frequency, and rapid network setup. As in terrestrial reliable multicasting cases, however, *feedback implosion* [4], [6], [9] is identified as a major problem in satellite multicast. The feedback implosion occurs when a large number of receivers sends their feedback to the satellite. The amount of potential feedback increases linearly with the number of receivers and may lead to a high traffic concentration at the satellite.

In this paper, we propose a *poker-game-based feedback suppression* (PFS) algorithm to reduce the number of feedback messages. The remainder of this paper is organized as follows: the PFS algorithm is described in Section II. Section III provides parameter choice and estimation techniques for the PFS scheme. Performance results are given in Section IV. Finally, conclusion is drawn in Section V.

II. Poker-Game-Based Feedback Suppression

We consider a point-to-multipoint network between a satellite and R direct receivers where a reliable multicast protocol performs. Suppose the satellite multicasts a packet to all receivers. Based on its channel condition, receiver i for $1 \leq \forall i \leq R$ is able to estimate its optimum number of redundancy blocks (l_i) to successfully decode the received packet [1]. Then, the satellite will be interested in transmitting $l_{\max}$ redundancy blocks to salvage the receiver in the worst channel condition, i.e., the receiver requiring $l_{\max}$ where $l_{\max} = \max_{\forall i}\{l_i\}$. In order for the satellite to obtain $l_{\max}$, all receivers have to report their number of optimum redundancy blocks via feedback messages. However, these feedback messages cause the feedback implosion problem [4], [6], [9] as R increases.

This work is supported by NASA-GRC under grant NAG3-2585.

Since each receiver would experience different channel condition, each receiver requires different number of redundancy blocks. Suppose m represents the maximum allowable number of redundancy blocks ($l_{\max} \leq m$). Then, a set of possible feedbacks $\mathcal{F}$ is given by $\mathcal{F} = \{1, 2, \dots, m\}$ where each element represents the number of redundancy blocks receivers require.

Since the satellite is interested in $l_{\max}$, the feedback of the larger number of blocks has a precedence (priority) over that of the smaller number of blocks. Based on the priority, our feedback suppression policy is as follows:

- **Suppression Policy:** The lower-priority feedback (with the smaller number of blocks) is suppressed by the higher-priority feedback (with the larger number of blocks).

Hence, we also refer the PFS scheme as *priority-based feedback suppression* algorithm. In the satellite multicasting, the following situations are observed:

- Each receiver is not able to perceive other receivers' feedback priority since we assume there is no direct connection among receivers.
- The satellite considers only one of the highest-priority feedbacks received.

Similar behaviors can be observed in a poker game [8] where cards are dealt to each player face-down by a dealer. Each player sees his own cards but not his opponents' cards. After a betting period, there is a showdown in which players show their cards and the highest hand wins the round. The nine categories of poker hands are from *high card* to *straight flush* (from the lowest to the highest). After a round, the winner will be the next dealer. In the betting period, the players who want to pass the round show their hands and become inactive in that period. If two players' categories are the same in the showdown, a tie-breaking rule is applied.

Our proposed feedback suppression algorithm performs with three levels of suppressions: (1) *feedback suppression by a dealer receiver*, (2) *inter-group suppression*, and (3)

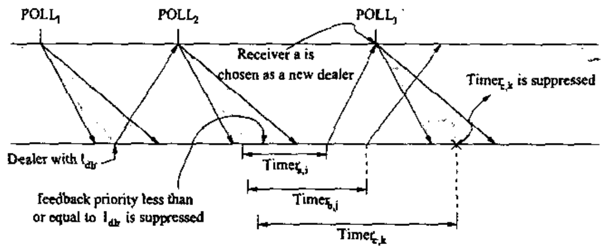


Fig. 1. The example of PFS scheme.

intra-group suppression.

A. Feedback Suppression by a Dealer Receiver

In order to provide scalable feedbacks, a receiver is elected as a *dealer receiver* which is the receiver that formerly had the highest-priority feedback. The dealer receiver may also have the highest-priority feedback in later time if the channel condition is slowly varying such as satellite channel.

The satellite broadcasts POLL_1 message (to explore dealer's feedback priority) to all receivers as shown in Fig. 1. Once the POLL_1 is received, the dealer receiver responds immediately to the satellite with its own feedback priority l_{dlr}. Then, the satellite broadcasts $POLL_2$ message with the dealer's priority value l_{dlr} (saying "is there any other priority higher than l_{dlr}?") to all receivers. Receivers with the feedback priority not exceeding l_{dlr} suppress their feedback according to the suppression policy mentioned earlier.

B. Inter-Group Feedback Suppression by T

The receivers with priority greater than l_{dlr} want to transmit their feedback. To further suppress their feedback, the PFS algorithm exploits feedback scheduling (timer-based approach), i.e., receivers with different priorities send their feedback in different time frames. If the time frames are sufficiently separated, we can effectively reduce the number of feedback messages since receivers in later time frame will be more likely to suppress their feedback due to other receivers' feedbacks in the earlier time frame.

We define a *group* by the set of receivers with the same redundancy blocks. A group with j redundancy blocks is called as group j.

Receivers in group j must wait at least for $(m-j)T$ and need to send their feedback within a time frame $[(m-j)T,(m-j+1)T)$ where m is the highest priority value defined in set $\mathcal{F}$; and T is the time frame size which is a design parameter and will be discussed in Section III. Receivers know T and m since the satellite broadcasts them via POLL_2 message.

Remark 1: In the poker game, the player with hand closer to ***straight flush*** has more chance to win the round. They are willing to show their cards earlier in the showdown. An analogy applies to our scheme. Receivers with priority value closer to m have more chance to send their feedback and to be the next dealer receiver. Therefore, they are willing to send their feedback earlier than other lower-priority receivers. Consequently, the higher priority a receiver has, the shorter timer value it schedules.

C. Intra-Group Feedback Suppression by μ

Inside a group (receivers in a group have the same priority feedback.), we still need to suppress their feedback. This suppression corresponds to a tie-break in the poker game. The random residual (or tie-break) timer $T_{i,j}$ for receiver i in group j is chosen from its density function $f_{T_{i,j}}(t_{i,j},\mu)$ where μ is a slope parameter for the residual timer. μ is a design parameter and will be discussed in Section III. Receivers also know μ since it is broadcasted with POLL_2 message.

Remark 2: From the above three levels of suppressions, a timer for receiver i in group j is scheduled as

$$Timer_{i,j} = \begin{cases} (m-j)T + T_{i,j}, & j > l_{dlr} \\ \infty, & j \le l_{dlr} \end{cases} \tag{1}$$

when the receiver receives POLL_2 as shown in Fig. 1. If the satellite receives feedback(s) from any receiver(s) after transmitting POLL_2, the receiver with the earliest feedback will be elected as a new dealer receiver (receiver a in group i, in Fig. 1). Then the satellite broadcasts POLL_3 (saying "do not send me the corresponding feedback") with the next dealer's priority value to all receivers to avoid further feedbacks. Once received POLL_3, those receivers with an active timer suppress (or cancel) their timer.

III. Parameter Choice for the PFS Algorithm

So far, we have deferred to explain how to choose parameters μ and T introduced in Section II which will be described in this section.

An important metric for the PFS algorithm is the number of feedback messages. Since receivers with feedback priority not exceeding l_{dlr} suppress their feedback, the total expected number of feedback messages $E[F]$ per poll round is given by

$$E[F] = \left\{ \sum_{j=l_{dlr}+1}^{m} E[F_j] \right\} + 1 \tag{2}$$

where l_{dlr} is the feedback priority at the dealer receiver; m is the maximum feedback priority; and 1 in the above equation represents the dealer's feedback.

Let $RX_{i,j}$ denote a receiver i in group j, and R_j the number of receivers in group j. The expected number of feedbacks from group j per poll round, $E[F_j]$, is obtained by

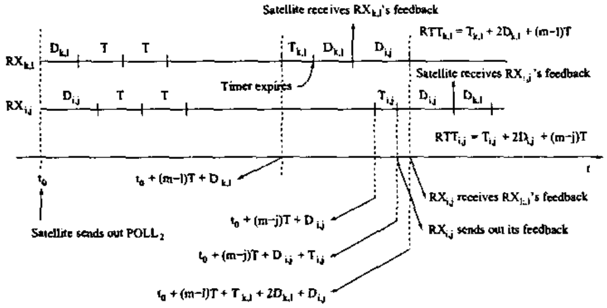


Fig. 2. The condition that $RX_{k,l}$ does not suppress the feedback of $RX_{i,j}$.

$$E[F_j] = \sum_{i=1}^{R_j} P(RX_{i,j} \text{ sends a FB}) \tag{3}$$

where FB represents a feedback.

Consider two receivers $RX_{i,j}$ and $RX_{k,l}$. Fig. 2 illustrates the condition that $RX_{k,l}$ does not suppress the feedback of $RX_{i,j}$. Let $D_{i,j}$ and $D_{k,l}$ denote the one-way transmission delays from satellite to $RX_{i,j}$ and $RX_{k,l}$, respectively. We assume here that the transmission delay from a satellite to a receiver is identical to the transmission delay in opposite direction. Also, let $T_{i,j}$ and $T_{k,l}$ denote the residual (tie-break) timers for $RX_{i,j}$ and $RX_{k,l}$, respectively, where $0 \le T_{i,j} \le T$ and $0 \le T_{k,l} \le T$.

Suppose that a satellite sends out $POLL_2$ at $t = t_0$ as shown in Fig. 2. At $t = t_0 + D_{k,l}$, $RX_{k,l}$ receives the poll, and at $t = t_0 + D_{i,j}$, $RX_{i,j}$ receives the poll. After receiving the poll, $RX_{k,l}$ schedules its timer value as $Timer_{k,l} = (m-l)T + T_{k,l}$ while $RX_{i,j}$ sets its timer as $Timer_{i,j} = (m-j)T + T_{i,j}$. Suppose that $RX_{k,l}$'s timer expires at $t = t_0 + D_{k,l} + Timer_{k,l}$, and the satellite receives $RX_{k,l}$'s feedback at $t = t_0 + Timer_{k,l} + 2D_{k,l}$. Since the satellite broadcasts $RX_{k,l}$'s feedback to all receivers, $RX_{i,j}$ will receive $RX_{k,l}$'s feedback at $t = t_0 + Timer_{k,l} + 2D_{k,l} + D_{i,j}$. $RX_{i,j}$ will suppress its feedback if $RX_{k,l}$'s feedback from the satellite arrives at $RX_{i,j}$ before $RX_{i,j}$'s timer expiration ($t_0 + D_{i,j} + Timer_{i,j}$). If $RX_{i,j}$ sends out its feedback before receiving $RX_{k,l}$'s feedback, then $RX_{k,l}$ cannot suppress the $RX_{i,j}$'s feedback. Therefore, the condition that $RX_{k,l}$ does not suppress the feedback of $RX_{i,j}$ is

$$T_{k,l} + 2D_{k,l} + (j-l)T > T_{i,j}. \tag{4}$$

If we choose the exponentially-distributed residual (tie-break) timer of which the probability density is given by

$$f_{T_{i,j}}(t_{i,j}) = \frac{\mu e^{\mu t_{i,j}}}{e^{\mu T} - 1} \{u(t_{i,j}) - u(t_{i,j} - T)\} \tag{5}$$

where $u(\cdot)$ is a unit step function, we have [1]

$$\begin{aligned} &P(RX_{i,j} \text{ sends a FB}) \\ &= \frac{\mu}{e^{\mu T} - 1} \int_0^T e^{\mu t_{i,j}} \prod_{\substack{l=j \\ (i,j)\neq(k,l)}}^{m} \prod_{k=1}^{R_l} P(T_{k,l} + 2D_{k,l} > t_{i,j} + \{l-j\}T) dt_{i,j} \end{aligned} \tag{6}$$

and

$$\begin{aligned} &P(T_{k,l} + 2D_{k,l} > t_{i,j} + \{l-j\}T) \\ &= \frac{\mu}{e^{\mu T} - 1} \int_0^T e^{\mu t_{k,l}} \left[1 - F_{D_{k,l}}\left(\frac{t_{i,j} + \{l-j\}T - t_{k,l}}{2}\right)\right] dt_{k,l}. \end{aligned} \tag{7}$$

where $F_{D_{k,l}}(d_{k,l})$ is the cumulative distribution function of one-way transmission delay $D_{k,l}$.

A. Estimation of Delay Distribution $F_{D_{k,l}}(d_{k,l})$

Since the delay distribution $F_{D_{k,l}}(d_{k,l})$ is not available at the satellite in general, a scheme to estimate the delay distribution is required. The satellite measures $\mathrm{RTT}_{k,l}$ which is equal to $T_{k,l} + 2D_{k,l} + (m-l)T$ as shown in Fig. 2. Since the satellite knows parameters m and T, and $RX_{k,l}$ informs $T_{k,l}$ and l to the satellite, the satellite is able to compute $D_{k,l}$ as

$$D_{k,l} = \frac{\mathrm{RTT}_{k,l} - (m-l)T - T_{k,l}}{2}. \tag{8}$$

The satellite collects every sample for $D_{k,l}$ and estimates the distribution $F_{D_{k,l}}(d_{k,l})$. Let $n_{d_{k,l}}$ denote the number of $D_{k,l}$'s that do not exceed $d_{k,l}$, and n denote the number of samples. Then, we obtain the empirical estimate of the distribution as

$$\hat{F}_{D_{k,l}}(d_{k,l}) = \frac{n_{d_{k,l}}}{n}. \tag{9}$$

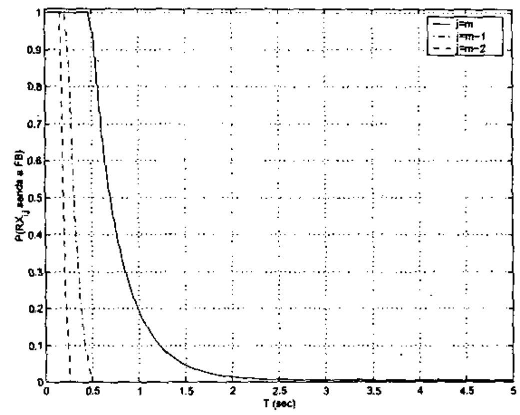
The unknown $F_{D_{k,l}}(d_{k,l})$ is bounded by [10]

$$\left|F_{D_{k,l}}(d_{k,l}) - \hat{F}_{D_{k,l}}(d_{k,l})\right| < \frac{z_{(\gamma+1)/2}}{\sqrt{n}} \sqrt{\hat{F}_{D_{k,l}}(d_{k,l})\left(1 - \hat{F}_{D_{k,l}}(d_{k,l})\right)} \tag{10}$$

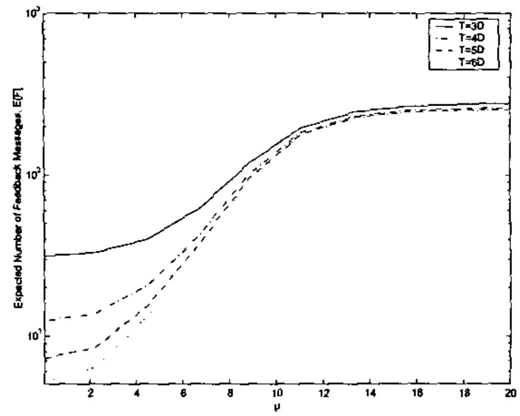
with confidence coefficient γ where $z_{(\gamma+1)/2}$ is the standard normal percentile.

B. Estimation of R_j's

At the satellite, the number of receivers R_j in group j is not available for $1 \le j \le m$. Therefore, estimation technique for R_j's is needed. From (2), (3), (6), and (7), we



(a)



(b)

Fig. 3. (a) $P(RX_{i,j}$ sends a FB) ($\mu = 0.3$ and $R_m = 10^3$) and (b) $E[F]$ vs. μ ($R = 10^3$ and $R_{m-n} = 250$ where $n = 0, 1, 2, 3$).

conclude that the expected number of feedback messages is a function of $\underline{R}$, T, μ, and $F_{D_{k,l}}(d_{k,l})$, where $\underline{R}$ is a vector of the number of receivers in each group given by $\underline{R} = [R_m, R_{m-1}, \ldots, R_{l_{dlr}+1}]^t$ where the number of receivers in groups less than or equal to l_{dlr} is not considered since those receivers suppress their feedback.

If $\underline{F}$ denote a vector of the number of received feedback messages from each group, we can express $\underline{F}$ as

$$\underline{F} = f(\underline{R}, T, \mu, F_{D_{k,l}}(d_{k,l})) \tag{11}$$

where $\underline{F} = [F_m, F_{m-1}, \ldots, F_{l_{dlr}+1}]^t$, and $f(\cdot)$ denotes the function of the expected number of feedbacks with given parameters $\underline{R}$, T, μ, and $F_{D_{k,l}}(d_{k,l})$.

Then, estimate $\hat{\underline{R}}$ of $\underline{R}$ for given parameters $T = T_0$, $\mu = \mu_0$, $\underline{F} = \underline{F_0}$, and $F_{D_{k,l}}(d_{k,l}) = \hat{F}_{D_{k,l}}(d_{k,l})$ is obtained as

$$\underline{F}|_{\underline{F_0}} = f(\hat{\underline{R}}, T, \mu, F_{D_{k,l}}(d_{k,l}))\Big|_{\substack{T=T_0, \mu=\mu_0, \\ F_{D_{k,l}}(d_{k,l}) = \hat{F}_{D_{k,l}}(d_{k,l})}} \tag{12}$$

where T_0, μ_0, and $\underline{F_0}$ are all known values at the satellite.

C. Choice of Parameters μ and T

Now, from the above scheme, we can estimate the vector of the number of receivers in each group as $\hat{\underline{R}}$. The satellite then selects μ and T which satisfy the following condition:

$$f(\hat{\underline{R}}, T, \mu, F_{D_{k,l}}(d_{k,l}))\Big|_{\hat{\underline{R}} = \hat{\underline{R}_0}, F_{D_{k,l}}(d_{k,l}) = \hat{F}_{D_{k,l}}(d_{k,l})} \leq \tilde{F} \tag{13}$$

where $\tilde{F}$ is the target (or desired) number of feedback messages.

IV. Numerical Examples

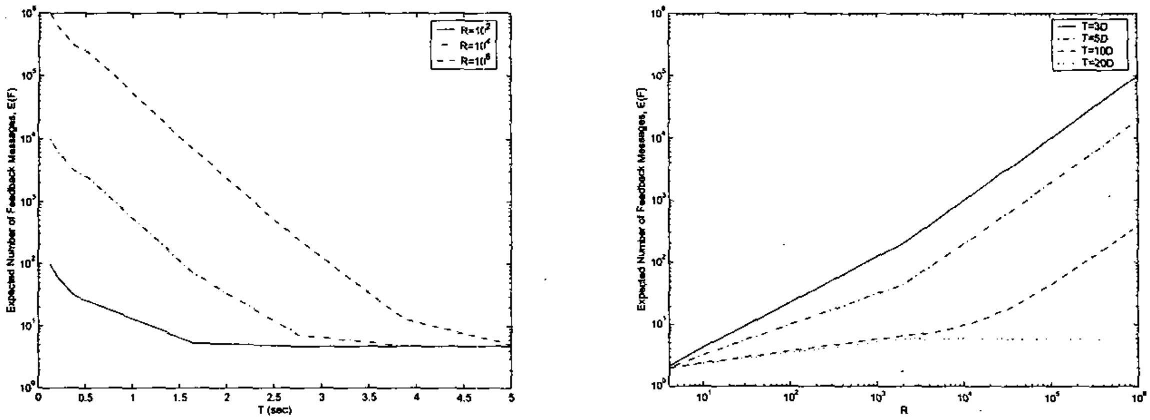
In the numerical examples, we assume beta-distributed one-way transmission delay [11] with mean D and minimum $D_{\min}$.

In a typical GEO satellite, the one-way transmission delay varies from 239.6 to 279.0 msec [2]. Therefore, we assume $D_{\min}$ and D as 239.6 msec and 259.3 msec, respectively; consequently, $D_{\max} = 2D - D_{\min} = 279$ msec.

According to numerical analysis based on the analytical model in (2), (3), (6), and (7), we derive suppression performance. In Fig. 3 (a), we show the $P(RX_{i,j}$ sends a FB) versus T with different parameters j where there are 10^3 receivers in group m. Overall, we observe that $P(PX_{i,j}$ sends a FB) decreases with increasing parameter T. When $T \geq 2D$, the receivers in group $j < m$ do not send their feedback since those receivers will receive feedbacks from group m before their timer expiration. Only receivers in group m have a possibility to send their feedback at $T \geq 2D$. In Fig. 3 (a), for fixed T, $P(PX_{i,j}$ sends a FB) decreases as j decreases since we have a suppression policy that the lower-priority feedback is suppressed by the higher-priority feedbacks.

Fig. 3 (b) depicts the expected number of feedback messages versus μ with various parameters T where $R = 10^3$. We observe that $E[F]$ increases with increasing μ. If $\mu \leq 1$, the slope is rather flat. Apparently, we see the expected number of feedbacks decreases with increasing parameter T which can be also observed in Fig. 4. Fig. 4 (a) illustrates the number of feedback messages versus T with various number of receivers R where $\mu = 0.3$ and the identical number of receivers is assumed for each group, i.e., $R_{m-n} = R/4, \forall n = 0, 1, 2, 3$. We observe that at $T = 5$ sec, we achieve almost the same suppression performance independent of R ($R = 10^2$, 10^4, and 10^6).

In Fig. 4 (b), we show the expected number of feedback messages versus the number of receivers with parameter



(a) (b)

Fig. 4. Expected number of FB, $E[F]$ ($\mu = 0.3$ and $R_{m-n} = R/4$, $\forall n = 0, 1, 2, 3$): (a) $E[F]$ vs. T; (b) $E[F]$ vs. R.

$\mu = 0.3$ and different parameters T. First, we observe that approximately constant suppression is achieved for a wide range of the number of receivers ($R > 100$) with $T = 20D$. Also, very small number of feedbacks can be obtained with different parameters T, e.g., with $T = 10D$, $E[F] < 10$ for $R = 10^4$, and even with $T = 5D$, $E[F] < 25$ for $R = 10^3$.

In order to verify our analytical suppression model in Section III, we compare the analytical model with simulation results. In Fig. 5, we show the expected number of feedback messages versus the number of receivers. We assume here that $\mu = 0.3$, $m = 10$, and the identical number of receivers, i.e., $R_{m-n} = R/4, \forall n = 0, 1, 2, 3$. From Fig. 5, we observe that our analytical model closely matches to the simulation results.

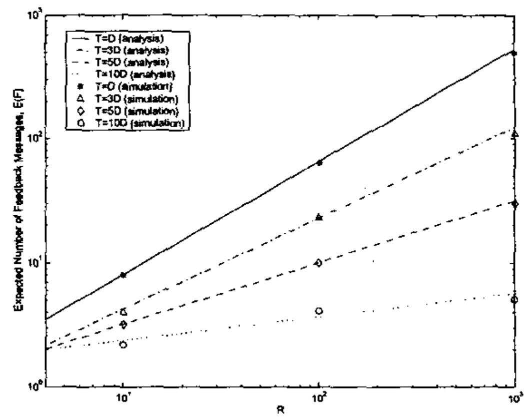


Fig. 5. $E[F]$ vs. R ($\mu = 0.3$ and $R_{m-n} = R/4, \forall n = 0, 1, 2, 3$)

V. Conclusions

To avoid feedback implosion problem, a novel scheme called *poker-game-based feedback suppression* (PFS) algorithm is proposed. An analytical model is provided for the expected number of feedback messages in the PFS scheme. The PFS algorithm has an advantage to support any types of satellite and receivers.

References

[1] I. F. Akyildiz, "A Multicast Protocol Suite for Satellite IP Networks," *Technical Report*, Georgia Institute of Technology, January 2002.

[2] M. Allman, D. Glover, and L. Sanchez, "Enhancing TCP over Satellite Channels using Standard Mechanisms," *RFC 2488*, January 1999.

[3] S. Cho, "Rate-Adaptive Error Control for Multimedia Multicast Services in Satellite-Terrestrial Hybrid Networks," *in Proc. of IEEE ICC 2000*, New Orleans, USA, pp. 446–450, June 2000.

[4] C. Diot, W. Dabbous, and J. Crowcroft, "Multipoint Communication: A Survey of Protocols, Functions, and Mechanism," *IEEE Journal on Selected Areas in Communications*, pp. 277–290, April 1997.

[5] S. Floyd, V. Jacobson, C. Liu, S. McCanne, and L. Zhang, "A Reliable Multicast Framework for Light-Weight Sessions and Application Level Framing," *IEEE/ACM Transactions on Networking*, vol. 5, no. 6, pp. 784–803, December 1997.

[6] S. K. Kasera, G. Hjalmtysson, D. F. Towsley, and J. F. Kurose, "Scalable Reliable Multicast Using Multiple Multicast Channels," *IEEE/ACM Transactions on Networking*, vol. 8, no. 3, pp. 294–310, June 2000.

[7] R. Kermode, "Scoped Hybrid Automatic Repeat Request with Forward Error Correction (SHARQFEC)," *in Proc. ACM SIGCOMM '98*, Vancouver, Canada, pp. 278–289, October 1998.

[8] P. Morehead, G. Mott-Smith, A. H. Morehead, *Hoyle's Rules of Games*, 1991.

[9] J. Nonnenmacher, E. W. Biersack, and D. Towsley, "Parity-Based Loss Recovery for Reliable Multicast Transmission," *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 4, pp. 349–361, August 1998.

[10] A. Papoulis, *Probability, Random Variables, and Stochastic Process*, McGraw-Hill, 1991.

[11] L. Wei and D. Estrin, "The Trade-offs of Multicast Trees and Algorithms," *in Proc. of ICCCN '94*, San Francisco, USA, 1994.