A Chain-Binomial Model for Pull and Push-Based Information Diffusion[°]

Mine Çağlar
Department of Mathematics
Koç University
Istanbul, Turkey
{mcaglar@ku.edu.tr}

Öznur Özkasap
Department of Computer Engineering
Koç University
Istanbul, Turkey
{oozkasap@ku.edu.tr}

***Abstract*—We compare pull and push-based epidemic paradigms for information diffusion in large scale networks. Key benefits of these approaches are that they are fully distributed, utilize local information only via pair-wise interactions, and provide eventual consistency, scalability and communication topology-independence, which make them suitable for peer-to-peer distributed systems. We develop a chain-Binomial epidemic probability model for these algorithms. Our main contribution is the exact computation of message delivery latency observed by each peer, which corresponds to a first passage time of the underlying Markov chain. Such an analytical tool facilitates the comparison of pull and push-based spread for different group sizes, initial number of infectious peers and fan-out values which are also accomplished in this study. Via our analytical stochastic model, we show that push-based approach is expected to facilitate faster information spread both for the whole group and as experienced by each member.**

Keywords—chain-binomial; epidemic algorithms; anti-entropy; peer-to-peer

I. Introduction

A low-overhead and scalable method for information spread is to use epidemic algorithms that involve pair-wise propagation of updates. Epidemic algorithms are based on the theory of epidemics which studies the spreading of infectious diseases through a population. Such protocols are simple, scale well, are robust against common failures, and provide eventual consistency as well. They combine benefits of efficiency in hierarchical data dissemination with robustness in flooding protocols. Epidemic communication allows temporary inconsistencies in shared data among participants, in exchange for low-overhead implementation. Information changes are spread throughout the participants without incurring the latency and bursty communication that are typical for systems achieving a strong form of consistency. In fact, this is especially important for large systems, where failure is common, communication latency is high and applications may contain a large number of participants.

Epidemic communication mechanisms were first proposed for spreading updates in a replicated database [1]. The aim in this case is to infect all replicas with new updates as fast as possible. Later on, epidemic or gossip style of communication has been used in several contexts such as large-scale direct mail systems [2], group membership tracking [3], support for replicated services [4], deciding when a message can be garbage collected [5], failure detection [6], loss recovery in reliable multicast [7], and distributed information management [8]. Reference [9] gives an overview of epidemic information dissemination in which the focus is on four design constraints namely, membership, network awareness, buffer management, and message filtering. Another study offers reliability via epidemic algorithms in content-based publish-subscribe [10]. There are some recent adaptive mechanisms suggested in the context of epidemic communication as well. For example, [11] proposes an adaptive epidemic communication mechanism based on adjusting the fan-out parameter. The aim is to enhance resiliency of epidemic algorithms adaptively in case of perturbations such as node failures.

Rather than using pull and push-based epidemic anti-entropy as a background mechanism to recover from failures, we investigate its usage for spreading information via periodic state exchanges. Key benefits of these approaches are that they are fully distributed, utilize local information only via pair-wise interactions, and provide eventual consistency, scalability and communication topology-independence. These properties make them suitable for peer-to-peer distributed systems. We develop a chain-Binomial epidemic probability model for push-based approach, similar to that used earlier for pull-based approach [12,13]. For pull-based approach, we adjust the model for different values of fan-out, that is, the number of other peers that a member is allowed to send a digest message at each round. On the basis of these models, our main contribution is the exact computation of message delivery latency observed by each peer, which corresponds to a first passage time of the underlying Markov chain. Such an analytical tool facilitates the comparison of pull and push-based spread for different group sizes, initial number of infectious peers and fan-out values which are also

[°] This work is supported in part by TUBITAK (The Scientific and Technical Research Council of Turkey) under CAREER Award Grant 104E064.

accomplished in this study. Any earlier work on delay calculations based on chain-Binomial model has been about delivery to the whole group, mainly for pull policy and is only approximate.

The paper is organized as follows. In Section II, pull and push-based epidemic approaches that we utilize for scalable information spread are described. Our chain-Binomial model developed for the pull and push-based approaches is given in Section III, followed by the numerical results presented in Section IV. Concluding remarks and future work are given in Section V.

II. Pull and Push-Based Models

Anti-entropy is an epidemic communication strategy introduced for achieving and maintaining consistency among the sites of a widely replicated database. Compared to deterministic algorithms for replicated database consistency, this strategy also reduces network traffic [1]. Anti-entropy has been proposed as a mechanism that runs in background for recovering errors of direct mail in large network [2] and for loss recovery in Bimodal Multicast protocol [12] that utilizes this mechanism for probabilistically reliable multicast communication.

In the anti-entropy process, non-faulty peers are always either *susceptible* or *infectious*. A site or peer holding information or an update it is willing to share is called infectious. A peer is called susceptible if it has not yet received an update. Periodically, each peer picks another site at random, and exchanges its state information with the selected one. For spreading information, we investigate pull and push-based approaches which are described next.

A. Pull-based approach

When an infectious peer holding information to be shared, picks randomly a susceptible peer lacking the specific information, this triggers information dissemination from infectious peer to the susceptible. If we consider a fixed population of size n, among which k peers are already infected, and infection occurs in rounds; the probability of infection can be formulated as follows. For the pull-based approach; assume that $P_{\text{pull}}(k,n)$ is the probability that a particular susceptible (uninfected) peer is infected in a round if k peers are already infected. The probability of infection for fan-out 1 is

$$P_{\text{pull}}(k,n) = 1 - P(\textit{nobody infects the susceptible peer})$$
$$= 1 - (1-1/n)^k$$

We will use this probability to track the total number of infected peers at each stage in the next section. In general, if the fan-out is f

$$P_{\text{pull}}(k,n) = 1 - (1-f/n)^k$$

Steps involved in the dissemination between two such peers is depicted in Fig.1(a) where infectious peer (on the left) has the data to be disseminated. In this scenario, (1) the infectious one picks a susceptible peer lacking data, and sends a digest (also referred to as gossip) message including its state. (2) On receiving digest and comparing it with its local information, the susceptible peer finds out it lacks data and sends a request for the data back to the infectious. (3) Upon getting request, infectious peer sends a retransmission of data which causes the other peer to be infectious for that data. In fact, each peer in the system performs state information exchange periodically and concurrently with the others. Moreover, each peer may have a set of information in its local buffer. Therefore, a digest message generated by a peer would consist of state information on the current contents of its message buffer. In that respect, the figure simplifies the scenario and illustrates the communication between two sample peers among the population of peers for one piece of information. Spreading updates is triggered by susceptible peers when they are picked as gossip destinations by infectious peers.

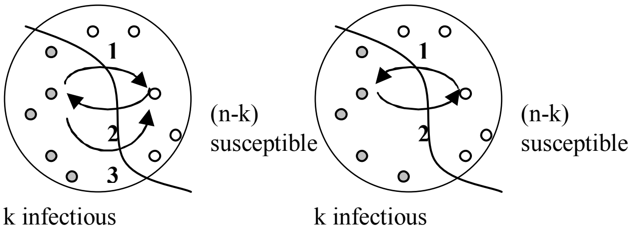

Figure 1. Illustration of (a) pull-based (b) push-based approaches.

B. Push-based approach

In this model, if a susceptible peer picks an infectious peer randomly, and sends its state information, this triggers information dissemination from infectious peer to the susceptible. For the push-based approach; assume that $P_{\text{push}}(k,n)$ is the probability that a particular susceptible (uninfected) peer is infected in a round if k peers are already infected out of a peer population with size n. In this case, a susceptible member's selecting an infectious peer will be sufficient for infection. Therefore, the probability of infection for fan-out 1 is

$$P_{\text{push}}(k,n) = 1 - P(\textit{the susceptible peer is not infected})$$
$$= 1 - (n-k)/n = k/n$$

When the fan-out is f in general, note that the susceptible peer must choose all f peers to gossip from the susceptible group in order not to get infected. Hence, the probability of infection becomes

$$P_{\text{push}}(k,n) = 1 - C(n-k,f)/C(n,f)$$

where C refers to combination.

Steps involved in the dissemination between two such peers is depicted in Fig.1(b) where infectious peer (on the left) has information labeled A. In this scenario, (1) on receiving digest and comparing it with its local information, the infectious peer finds out that the digest owner lacks the data and (2) directly retransmits, or pushes the data which causes the other peer to become infectious. As illustrated in the figure, in the push-based approach, no request messages are used. Spreading updates is triggered by infectious peers when they are selected as gossip targets by susceptible peers. Push-based approach may be beneficial due to its low overhead since it does not require request messages to initiate dissemination.

III. Delay through Chain-Binomial model

We use the chain-Binomial model introduced earlier for epidemics [12,14]. Let I_t denote the number of infectious peers at time *t*, which is convenient to track for our purposes rather than susceptible ones as in [15].

The probability that there are *j* infectious peers or holders of the message at the next stage when there are *k* infectious peers at present can be computed by first considering the number of susceptible peers. The probability *p* of success, namely getting infected, for a particular susceptible process depends on *k*. The value of *p* depends on the type of approach and the fan-out *f* as given in Section II. In the chain-Binomial model, the susceptible peers are assumed to get infected independently from each other. Therefore, the number of susceptible peers which get infected in the next round is distributed Binomially with parameters *n-k* and *p*. Having *j* infectious peers in the next round, when there are *k* at present is equivalent to getting (*j-k*) susceptible peers infected, which can occur in $C(n-k, j-k)$ different combinations. As a result, the transition probability is

$$P_{kj} = P\{I_{t+1} = j \big| I_t = k\} = c\, C(n-k, j-k) p^{j-k} (1-p)^{n-j} \quad (1)$$

where *c* is a normalizing constant in the case of pull-based approach for ensuring $\sum_j P_{kj} = 1$ as $j = k, k+1, \ldots, \min\{2k, n\}$. This is a slight variation of the chain-Binomial model for representing the true process in a more realistic way. If there are *i* infectious peers, then they can infect at most *i* susceptible peers with $f = 1$ resulting in at most 2*i* infectious ones. When the fan-out *f* is arbitrary, *i* infectious peers can infect at most $f\,i$ susceptible peers resulting in at most $(f+1)\,i$ infectious peers in the next stage. On the other hand, for the push model $j = k, k+1, \ldots, n$ as one infectious peer can infect several susceptible peers at a time with no upper limit. In this case, the constant *c* is equal to 1, as P_{kj} form a full set of Binomial probabilities. In view of Eq.(1), it is clear that the process $\{I(t) : t = 0,1,2,\ldots\}$ is a Markov chain.

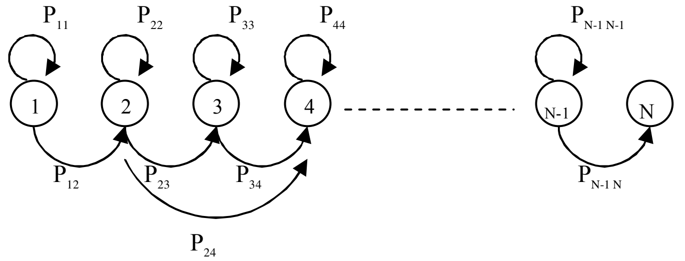

Figure 2. Transition probability graph for Markov chain.

Using this model, we analyze the performance of pull and push policies in epidemic dissemination. An important performance measure is the mean delay per user from user perspective. On the other hand, the total latency for dissemination to all group members gives an overall measure for the system. We exactly find both quantities.

The states of the Markov chain are illustrated in Fig.2 where arcs are present only if there is a positive probability to go from one state to another. Let $s_{i\bar{j}}$ denote the first passage time from state *i* to the set of states $\bar{j} = \{j, j+1, \ldots, N\}$, for $i = 1, \ldots, j-1$. If the Markov chain enters the set $\bar{j}$ by taking a value *l* which is different from *j*, there will be at least *j* infectious peers in the system and the *j*th infection will occur only at the time of transition to *l*, since $l > j$. Therefore, we can interpret $s_{i\bar{j}}$ as the expected time for the *j*th infection to occur, which is the same as the mean delay that the *j*th member to receive the message experiences. There is a positive probability that this delay may be the same as the *l*th member experiences for some $l > j$, in view of the argument above and due to the discreteness of time in our model. However, the mean delays will be different.

For each *j*, we form a set of equations to solve for $s_{i\bar{j}}$ using one step analysis of the Markov chain. Recall that $P_{ik} \neq 0$ only if $k \geq i$ as *P* is upper triangular. For $j = 2$,

$$s_{1\bar{2}} = 1 + P_{11} s_{1\bar{2}}$$

as the chain has to make at least one transition, equivalent to one gossip round to enter the states $\{2,3,\ldots,n\}$. If it remains in state 1 which occurs with P_{11} probability, then the process restarts itself and has to wait $s_{1\bar{2}}$ amount of time again on average. As a result, $s_{1\bar{2}} = 1/(1-P_{11})$, the mean of a geometric random variable as expected. Similarly,

$$s_{1\bar{3}} = 1 + P_{11} s_{1\bar{3}} + P_{12} s_{2\bar{3}}$$

$$s_{2\bar{3}} = 1 + P_{22} s_{2\bar{3}}$$

Solving these equations, we get

$$s_{1\bar{3}} = \frac{1}{1-P_{11}} \left(1 + \frac{P_{12}}{1-P_{22}} \right).$$

For finding $s_{1\bar{4}}$, three equations have to be set up as

$$s_{1\bar{4}} = 1 + P_{11} s_{1\bar{4}} + P_{12} s_{2\bar{4}} + P_{13} s_{3\bar{4}}$$

$$s_{2\bar{4}} = 1 + P_{22} s_{2\bar{4}} + P_{23} s_{3\bar{4}}$$

$$s_{3\bar{4}} = 1 + P_{33} s_{3\bar{4}}$$

where P_{13} is 0 in pull-based approach when *f*=1. These equations are equivalent to the system

$$(I - P_{\bar{4}})S_{\bar{4}} = \underline{1}$$

where *I* is a 3x3 identity matrix, $\underline{1}$ is a vector of 1's of length 3, $S_{\bar{4}} = [s_{1\bar{4}}, s_{2\bar{4}}, s_{3\bar{4}}]$ and $P_{\bar{4}}$ is the upper left 3x3 part of the matrix *P* given by $\begin{bmatrix} P_{11} & P_{12} & P_{13} \\ 0 & P_{22} & P_{23} \\ 0 & 0 & P_{33} \end{bmatrix}$.

In general, for $3 : j : n$,

$$s_{i\bar{j}} = 1 + \sum_{k=i}^{n} P_{ik} s_{k\bar{j}} \qquad i = 1,2,\ldots, j - 1 .$$

which is equivalent to the system

$$(I - P_{\bar{j}})S_{\bar{j}} = \underline{1}$$

where $P_{\bar{j}}$ is the upper left $(j-1)\times(j-1)$ part of the matrix *P* and $S_{\bar{j}} = [s_{1\bar{j}}, s_{2\bar{j}}, \ldots, s_{j-1,\bar{j}}]$. The system being an upper triangular system of linear equations can be solved very efficiently. The k^{th} row of the solution matrix *S* provides information on expected number of rounds for the j^{th} infection to occur starting with *k* infectious peers, for $j \geq k$. As the model is a Markov chain, we can obtain the performance of a group having *k* infectious members initially from this information.

IV. Numerical Results

In this section, the analytical models of pull and push-based approaches are evaluated numerically for different performance and system parameters. Each approach is compared within itself and also with respect to the other.

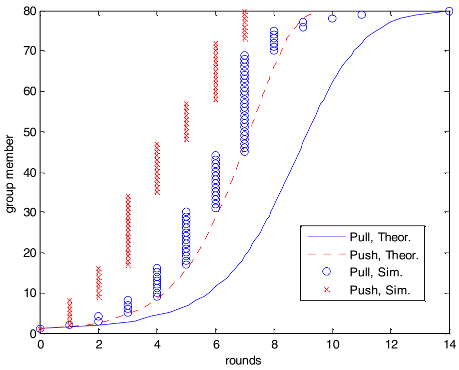

Figure 3. Group member versus message delivery time from theory and simulation for *n*=80.

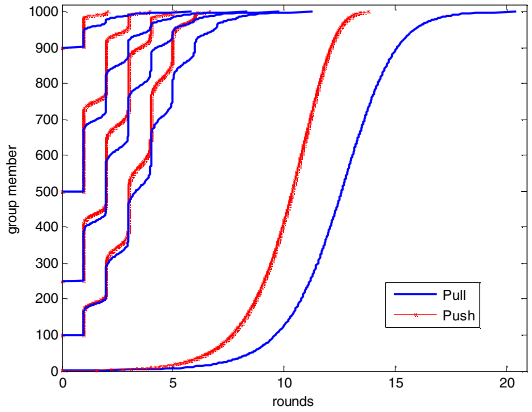

Figure 4. Group member versus message delivery time starting with 1, 100, 250, 500 and 900 infectives at time 0, for *n*=1000.

In Fig.3, a set of simulated data for a group of size 80 are plotted together with results of our theoretical model for pull and push epidemic approach evaluated at *n*=80. The plot shows the group members (in the order of receiving time) versus time (in units of rounds) that they receive the message. The simulations are performed on TinyOS TOSSIM simulator for an ad hoc information spread scenario using algorithms of our pull and push-based approaches described. Results are only for one run and for dissemination of a single message. Hence, several members may become infected at one round. The theoretical curves include expected time that is why they can take continuous values for rounds. The qualitative behavior is quite similar in both simulation and the theory for the push model. However, simulated curve for the pull model is sharper than the theory predicts. That might be due to the fact that the chain-Binomial model may not be as adequate for the pull approach. The push approach delivers the message faster than the pull approach. Although this study does not aim to fit parameters of the chain-Binomial model over simulated values, scaling the parameter *p* can serve this purpose.

An interesting further study would be investigating the effect of network conditions such as node/link failures, congestion that may occur in bottleneck links when spreading large amount of data and the associated overhead of both models. In fact, as reported in the empirical results of [15], push model becomes inefficient in terms of average delay observed by peers when spreading large amount of data even in the lack of network failures. The chain-Binomial model for multiple data could be extended to reflect any network congestion in this case. A single infectious peer may attempt to infect several members at once in push model and cause congestion especially during continuous data dissemination.

The delay time experienced by each peer to receive the message is an important performance measure, from user perspective. In Fig.4, the group members in the order they receive the message are plotted against expected number of rounds for different starting number *k* of infectious processes, for *k*=1,10,25,90 and *n*=100. The message is received much faster for larger *k* and also with clear bursts even for the mean delivery time which takes continuous values. For larger *k*, several members have very close expected delivery time which is the reason of the bursts. An analogous plot is given in Fig.5 for the case of a group of size *n*=1000. Similar conclusions can be drawn as in *n*=100. The mean time to receive a message as experienced by a peer is given in tables I and II, respectively for *n*=100 and *n*=1000. These are summary measures for the complete information given in Figs.4 and 5. It is interesting that for group sizes 100 and 1000 the mean number of rounds to receive the message per user is almost the same for comparable ratios of initial number of infectious members. The numbers are different only for *k*=1, which clearly has different ratio in different *n*. Push policy delivers the message faster per member as also evident from the previous figures.

TABLE I. EXPECTED NUMBER OF ROUNDS PER PEER FOR *n*=100.

	***k*=1**	***k*=25**	***k*=50**	***k*=90**
Pull	8.8	2.78	2.12	1.65
Push	6.7	2.33	1.64	1.10

TABLE II. EXPECTED NUMBER OF ROUNDS PER PEER FOR *n*=1000.

	***k*=1**	***k*=250**	***k*=500**	***k*=900**
Pull	12.4	2.78	2.12	1.65
Push	10.1	2.32	1.63	1.10

For comparison purposes, we depict the cases of *n*=100 and *n*=1000 together in Fig.6 for *k*=1. Although the curves are similar, the time it takes to reach 100 is shorter for *n*=1000 due to the efficiency gained from larger group size.

In Figs. 1 to 5, we have taken fan-out *f* to be 1, that is, a member gossips to one member at each round. To see the effect of *f* for both pull and push approaches, we change *f* as 1,2 and 4. The results are given in Figs. 7 and 8, respectively for *n*=100 and *n*=1000 and initial number of infectious peers *k*=1. The qualitative behavior is similar for both group sizes. Certainly, increasing *f* decreases the average delay experienced by each member. However, doubling *f* from 1 to 2 is more effective than doubling it from 2 to 4. Moreover, the difference between pull and push based approaches become almost indistinguishable for *f*=4. The mean time to receive a message as experienced by a peer summarized in table III for different *f* starting with *k*=1 confirm these conclusions.

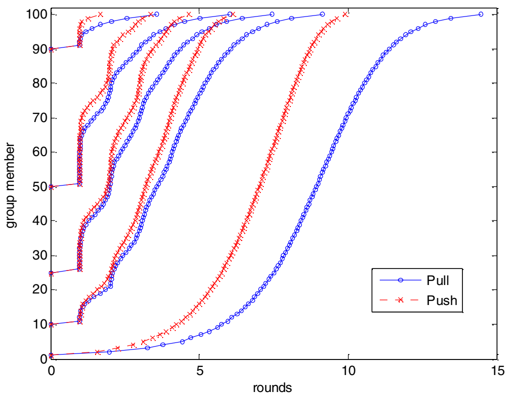

Figure 5. Group member versus message delivery time starting with 1, 10, 25, 50 and 90 infectives at time 0, for *n*=100.

TABLE III. EXPECTED NUMBER OF ROUNDS PER PEER FOR VARIOUS *f*

	***f*=1**		***f*=2**		***f*=4**	
	Pull	**Push**	**Pull**	**Push**	**Pull**	**Push**
***n*=100**	8.8	6.7	5.1	4.3	3.3	3.0
***n*=1000**	12.4	10.1	7.3	6.4	4.9	4.5

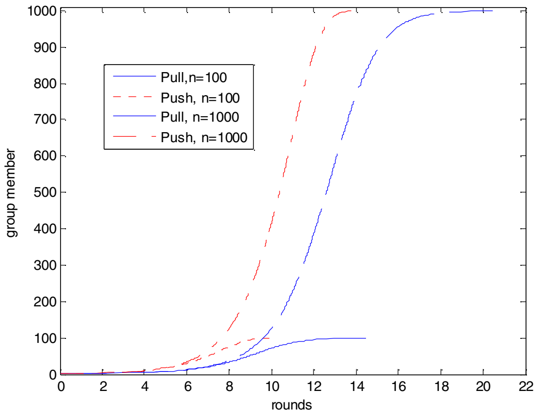

Figure 6. Group member versus message delivery time for n=100 and n=1000.

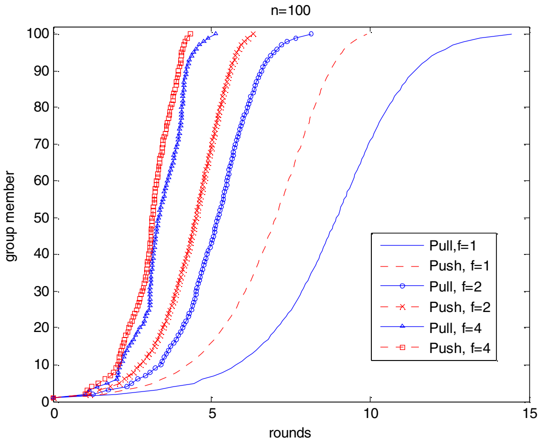

Figure 7. Group member versus message delivery time for *n*=100 and various values of fanout *f*.

V. CONCLUSION

We have developed analytical models for pull and push-based approaches in anti-entropy/epidemic protocols to compare their performance. These approaches are fully distributed, utilize local information only via pair-wise interactions, and provide eventual consistency, scalability and communication topology-independence which make them suitable for peer-to-peer distributed systems. In the analytical model, our use of the underlying Markov chain for exact delay computations is novel. This approach replaces any approximate approaches of earlier studies and has made it possible to evaluate the delay performance observed by each peer.

Our numerical investigations predict that push-based approach inherently facilitates faster information spread both for the whole group and as experienced by each member. This theoretical result is confirmed by network simulations for one message, whereas it is in contradiction when multiple messages are released. Incorporation of the latter case into the models developed in this paper will be future work.

We will further improve the pull model by modifying the probability of infection. Based on these probabilities, the independence property used in Binomial distribution is just an approximation for the pull case whereas it is exact for the push case.

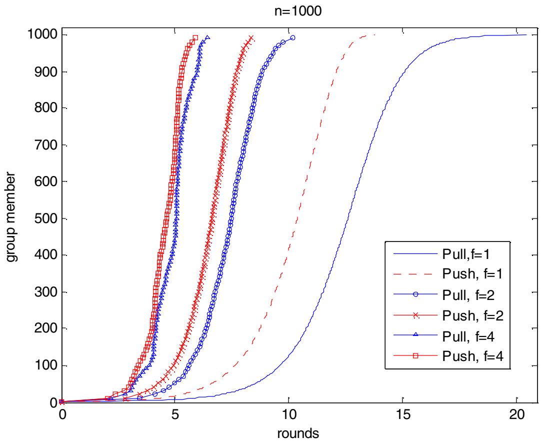

Figure 8. Group member versus message delivery time for n=1000 and various values of fanout f.

REFERENCES

[1] Demers, A., Greene, D., Hauser, C., Irish, W., Larson, J., Shenker, S., Sturgis, H., Swinehart, D. and Terry, D., Epidemic Algorithms for Replicated Database Maintenance, Proc. of the Sixth ACM Symp. on Principles of Distributed Computing, 1-12p, 1987.

[2] Birrell, A.D., Levin, R., Needham, R.M. and Schroeder, M.D., Grapevine, An Exercise in Distributed Computing, Communications of the ACM, 25(4), 260-274p, 1982.

[3] Golding, R.A. and Taylor K., Group Membership in the Epidemic Style, Technical Report, UCSC-CRL-92-13, University of California at Santa Cruz, 1992.

[4] Ladin, R., Lishov, B., Shrira, L. and Ghemawat, S., Providing Availability using Lazy Replication, ACM Transactions on Computer Systems, 10(4), 360-391p, 1992.

[5] Guo, K., Scalable Message Stability Detection Protocols, Ph.D. dissertation, Cornell University Dept. of Computer Science, 1998.

[6] van Renesse, R., Minsky, Y. and Hayden, M., A Gossip-style Failure Detection Service, Proceedings of Middleware'98, 55-70p, 1998.

[7] Xiao, Z. and Birman, KP., 2001, A Randomized Error Recovery Algorithm for Reliable Multicast, Proceedings, IEEE Infocom 2001.

[8] van Renesse, R., Birman, K.P., Vogels, W., Astrolabe: A Robust and Scalable Technology for Distributed System Monitoring, Management, and Data Mining, ACM Transactions on Computer Systems, Vol. 21, No. 2, pp. 164–206, May 2003.

[9] P.T. Eugster, R. Guerraoui, A-M. Kermarrec, L. Massoulie, Epidemic Information Dissemination in Distributed Systems, IEEE Computer, May 2004, pp. 60-67.

[10] Costa, P., Migliavacca, M., Picco, G.P., Cugola, G., Introducing Reliability in Content-Based Publish-Subscribe through Epidemic Algorithms, 2nd International Workshop on Distributed Event-Based Systems (DEBS'03), 2003.

[11] Tsuchiya, T., and Kikuno, T., An Adaptive Mechanism for Epidemic Communication', in Proc. Bio-ADIT 2004.

[12] Birman, K.P., Hayden, M., Ozkasap, O., Xiao, Z., Budiu, M. and Minsky, Y., Bimodal Multicast, ACM Transactions on Computer Systems, 17(2), 41-88p., May 1999.

[13] Ozkasap, O., Caglar, M. Traffic Characterization of Transport Level Reliable Multicasting: Comparison of Epidemic and Feedback Controlled Loss Recovery. To appear in *Computer Networks*.

[14] Bailey, N.T.J., The Mathematical Theory of Infectious Diseases and its Applications, Charles Griffin and Company, London, 1975.

[15] Ozkasap, O., An Empirical Investigation of Peer-to-Peer Epidemic Anti-Entropy Algorithms, unpublished, 2005.