Gossip-based Delay-Sensitive N-to-N Information Dissemination Protocol

Vincent Wing-Hei Luk, Albert Kai-Sun Wong, Robin Wentao Ouyang, Chin-Tau Lea

***Abstract*—This paper presents GConf, a peer-to-peer Gossip-based delay sensitive N-to-N information dissemination protocol, intended for applications such as real-time multiparty conferencing. Through 3-way gossip-based exchanges with randomly selected peers, information frames can be spread throughout the whole community within a delay of 1.5 round-trip-times. The non-delivery probability can be improved exponentially as a function of the peer fanout. Performance evaluations based on a mathematical model and simulations are presented.**

***Index Terms*— Communication Protocols, Distributed algorithms, Epidemiology, Reachability Analysis**

I. Introduction

In this paper we propose a new gossip-based delay sensitive peer-to-peer model for N-to-N information dissemination. In traditional N-to-N information dissemination via the full mesh topology, as shown in Figure 1a, each participant has to send and receive (N-1) streams. Obviously, the full mesh communication model has a scalability problem if the number of peers increases.

The protocol proposed, which we call GConf, is designed to solve the scalability problem. This paper describes this protocol and presents a mathematical model for analyzing GConf's performance together with the results from a packet-level discrete event model simulator implemented to verify the mathematical model. We show that with a target delay which is set to less than 1.5 RTT (Round trip time), a sufficiently small probability that a node cannot receive the disseminated information can be archieved if a minimum fanout is enforced,

The rest of the paper is organized as follows. Section 2 presents an overview of existing gossip algorithms. Section 3 describes the GConf protocol. Section 4 presents the performance evaluation results from a mathematical model and from the simulator. Section 5 concludes the paper.

Vincent Wing-Hei Luk, Albert Kai-Sun Wong, Chin-Tau Lea, Wentao Ouyang are with the Department of Electronic & Computer Engineering, Hong Kong University of Science & Technology, Hong Kong (email: vincentl@ece.ust.hk; eealbert@ece.ust.hk; eelea@ece.ust.hk; oywtece@ust.hk).

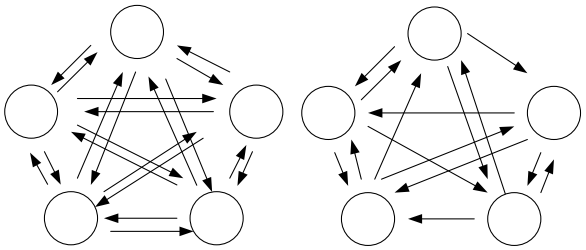
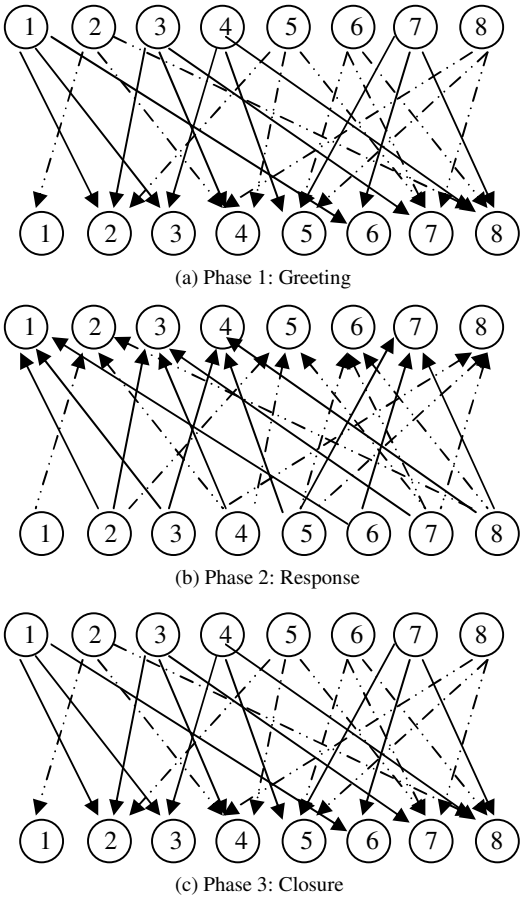


Fig. 1. (a) Information Dissemination in Full Mesh Topology
(b) Information Dissemination in Gossiping with fanout=3

II. Related Work

The gossip protocol was first proposed as an epidemic algorithm for replicated database maintenance in [1]. In that work, three methods for maintaining loose database synchronization are examined: *direct mail*, *anti-entropy*, and *rumor mongering*, the latter being gossip. In direct mailing, each update entry is immediately mailed from its entry site to all other sites. Evidently, direct mailing is not completely reliable, as the mailing site may not know about all other sites and as sometimes mails can be lost. Hence, an anti-entropy protocol is proposed to run in the background so that sites will exchange database content and resolve differences in *push*, *pull*, and *push-pull* modes. The eventual synchronization of all sites in an expected time proportional to the log of the population size is quoted as a basic result from epidemic theory. In rumor mongering, a site with a new update becomes a hot rumor and periodically selects another site at random and ensures that the update is received by the other site. There is a die-down mechanism such that if a site sees that too many of its gossip targets have already received the update, the site will stop treating the update as "hot" and cease to gossip it further.

Since then, gossip protocols have been further studied for consistency management in replicated database [2], failure detection [3], bimodal multicasting [4], garbage collection [5], leader election [6], and more recently, in one-to-many message dissemination [2, 7, 8] and distributed averaging[9]. However, none of these protocols is optimized for N-to-N multi-party communication.



(a) Phase 1: Greeting

(b) Phase 2: Response

(c) Phase 3: Closure

Fig. 2. Active frames are diffused via gossips that are 3-way message exchanges in one cycle. Fanout=3.

Worth noting is the mathematical analysis in [8], where an upper bound was derived on the probability of full connectivity of the directed random graph resulting from the gossip algorithm:

$$\lim_{n \to \infty} \pi(p_n, n) = e^{-e^{-c}}$$

where the gossip fanout is $\log n + c + o(1)$, with $o(1)$, the little-o, meaning a term that becomes insignificant compared to 1 as n becomes large. Nevertheless, this bound is valid only with the delivery latency ignored.

Our Contributions

Gossip protocols mentioned above are not designed for N-to-N multi-party communication with delay considerations – applications such as multi-party P2P conferencing. In addition to delay considerations, another difference for P2P conferencing is that 100% packet delivery is not required, especially if delivery is achieved only with an excessive delay. This delay and delivery tradeoff will be considered in our protocol. An analytical model is developed to evaluate the performance of the protocol. A discrete-event simulation system, one that has incorporated the protocol prototype and random network delays, is developed gain insight into the protocol and to validate the analytical results.

III. GOSSIPING PROTOCOL

The GConf model consists of n nodes that operate in cycles which are initiated at fixed intervals. In each cycle, each node can generate at most one information frame (a voice frame, for example) to be distributed to the remaining n-1 nodes. A node that has a frame to distribute is called "active" in that cycle. Gossiping in GConf means that each node, regardless of whether it is active or not, will randomly selects a small number of peers to gossip with at the start of each cycle. The number of peers selected is called the *fanout*. Each gossip cycle in GConf consists of exchanges of messages in three phases as shown in Figure 2: *Greeting*, *Response* and *Closure*, completed in 1.5 round-trip-times (RTTs). Information frames generated by active nodes are diffused across the network through the three message types, as depicted in Figure 2. For each node, a timer is set for each cycle so that upon the timer expiry, received frames will be aggregated and sent to upper layer for further processing. Note that before the completion of each cycle, new cycles can be launched and messages associated with different cycles are processed in parallel. Messages associated with different cycles are identified by a corresponding time-stamp t, or cycle id. Nodes need to be loosely synchronized so that each cycle can start at around the same time (synchronized in cycle). In our mathematical analysis, we assume the nodes are totally synchronized, even in phase (meaning that all nodes complete one phase together before moving to the next phase), in order to make the analysis tractable. We later also show simulation results for a practical system which is synchronized in cycle but not in phase because of varied and random delays. We have performed simulations with n ranging from 20 to 500 and measured the non-delivery probability (the probability that a node fails to receive a given information frame) and the network traffic in relation to the fanout.

A. The Protocol Format

The prototype message format in GConf is shown in Figure 3. The format applies to all three message types: Greeting, Response, and Closure, and the message type is indicated by the first two bits of the message header. Following the message type field is the time-stamp (or cycle id). For each cycle, each node keeps track of a Status-Map (SM) that indicates the information frames that it has already received for that cycle. For example, bit M_i of the SM is set to '1' if an information frame from peer i has been received, "0" otherwise. The SM is included in every message that a node transmits. The length of the SM can be variable as there the number of nodes may vary. Based on the SMs received, a node decides how to distribute a frame to its peers. Following the status map, each message contains a variable number of information frames to be disseminated. Each information frame is preceded by a frame id that identifies the node that has generated the frame.

B. The 3-Phase Exchange

The main feature of the GConf protocol is the way that information frames are pushed to other peers. The protocol has

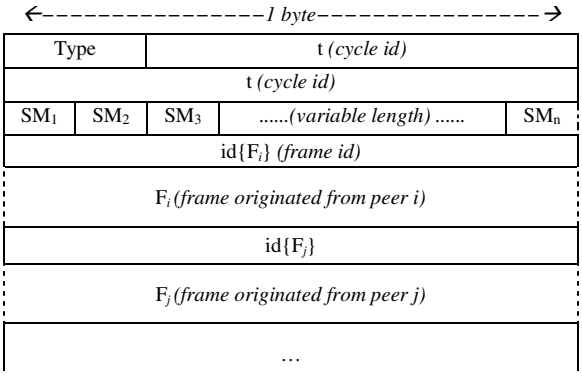


Fig. 3. GConf Message format

three phases. In phase 1 (the Greeting phase), every node, regardless of whether it is active or not, will randomly select *b* peers to gossip with for the current cycle. The *b* selected peers are called the "*children*" of the selecting node, which is now the "*parent*". Two nodes that share a common parent are called "*siblings*". Two nodes that share a common child are called "*co-parents*". All children of a node's co-parents are called "*step children*" of that node. An example is shown in Figure 2. Each of all eight nodes randomly selects 3 (i.e. b=3) children during the Greeting phase. Nodes 1, 3, 4, and 7 are active and each creates an information frame for this cycle. Node 1 has two co-parents: nodes 3 and 4 as each has a common child in node 1. Similarly, node 3 also has two *co-parents*. In Figure 2, node 4 is not a child of node 1, but is a step child of node 1.

In the second phase (the Response phase), a child will send back its SM and all the received frames during the first phase to its parents (unless a frame is already received by a parent as indicated by the SM in the parent's greeting message). The status map in each node will be updated whenever a new message containing information frames arrives. During the third phase (called the Closure phase), a station will send all frames that it has accumulated during the first two phases to a child if the child does not have these frames (as determined by the status map). Note that the parent-child relationship is determined during the greeting phase and is not changed by the subsequent response and closure phases. At the end of the third phase, all the frames will be passed up to the application layer.

IV. Performance Analysis

In this section, we develop an analytical model to study the non-delivery rate of information frame in the proposed GConf protocol. The analysis is based on an idealized 3-way gossip exchange model in which all nodes act in perfect cycle and phase synchronization. The idealized model allows us to provide a first-cut mathematical analysis. We will present simulation results for the non-ideal case in the next section.

The 3-way gossip exchanges create a family tree for each node. Even in the idealized situation with phase synchronization and without transmission losses and faulty nodes, only peers in the family tree of a source node *s* can receive the information frame from *s* in the given cycle. Therefore the probability that any given node is left out of the family tree of a source node provides a lower bound to the frame non-delivery probability. Note that a physical node may assume multiple capacities in relation to *s*. When no node assumes more than one capacity the number of nodes in the family tree is maximized with:

b	number of children;
m_c	number of co-parents
m_c (b-1)	number of step-children
m_p	number of parents
m_p (b-1)	number of siblings

We observe that in the network, each node has a fixed number of *b* children. However, the number of parents that a given node has is a random variable. Consider p_p, the probability that a given node is a parent of *s*. Obviously, $p_p=b/n-1$, and m_p, the number of parents of *s*, follows the binomial distribution

$$B_{n-1,p_p}(k)=\binom{n-1}{k}p_p{}^k(1-p_p)^{n-1-k}$$

with an expected value of

$$E[m_p]=(n-1)p_p=b\,.$$

This implies that each node has an average number of *b* parents.

Our analysis establishes also that m_c, the number of co-parents of *s*, follows a binomial distribution $B(n-1, p_c)$ where p_c is the probability any node is a co-parent to *s*. p_c is given by:

$$p_c=1-\frac{\binom{n-1-b}{b}}{\binom{n-1}{b}}\,.$$

$$=1-(1-\frac{b}{n-1})(1-\frac{b}{n-2})\cdots(1-\frac{b}{n-b})$$

An upper and lower bound can be written for p_c:

$$1-(1-\frac{b}{n-1})^b<p_c<1-(1-\frac{b}{n-b})^b$$

This bound establishes that the expected number of co-parents, $E[m_c]=(n-1)\,p_c$, is smaller than but very close to b^2 when *n* becomes large.

To estimate the number of step-children, consider p_s, the probability that a given node is a step-child of *s*. We have:

$$p_s=\sum_0^{n-1}P[\begin{matrix}\text{a given node is}\\ \text{a step child of } s\end{matrix}\ |\ \begin{matrix}s \text{ has } k\\ \text{co-parents}\end{matrix}]P[k \text{ co-parents}]$$

$$=\sum_0^{n-1}[1-\left(\frac{n-1-b}{n-1}\right)^k]\binom{n-1}{k}p_c{}^k(1-p_c)^{n-1-k}$$

The number of step children is the binomial distribution $B(n-1, p_s)$ and can be shown to be of order b^3. Since the number of step children equals the number of receiving parties, we expect that the fanout *b* must be of the order $n^{1/3}$ or otherwise many nodes will be left out of the family tree. We assume that the fanout is a constant *c* times the 3rd root of *n*; that is, we assume $b = c\,n^{1/3}$ and focus on the constant *c* in our study.

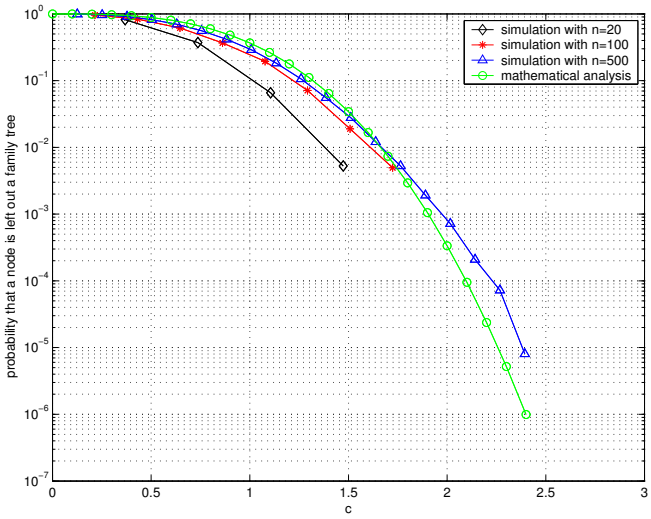


Fig. 4. p_l from simulation and from analytic model with phase synchronization

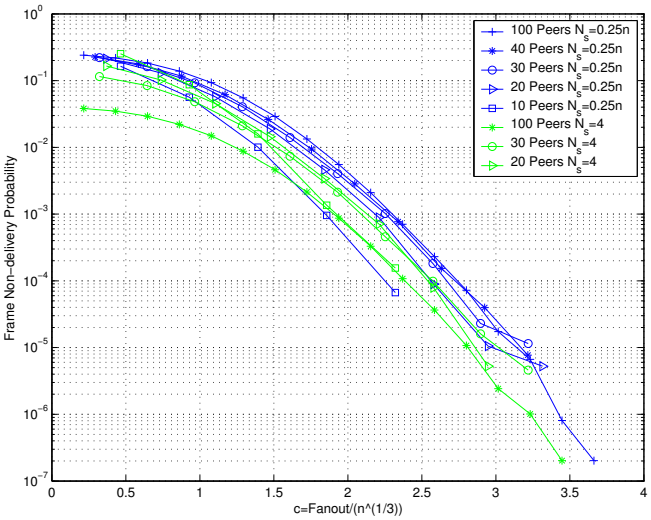


Fig. 5. Frame non-delivery probability from simulation without phase synchronization

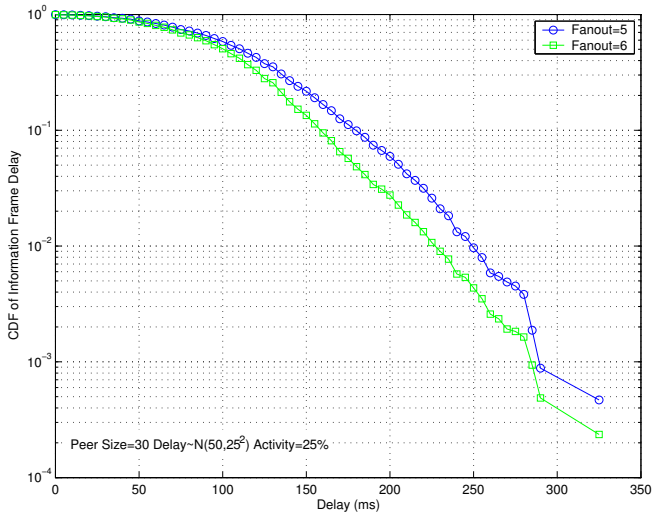


Fig. 6. CDF of first information frame delay without phase synchronization

We can now propose an approximation on p_l, the probability that a node is left out of the family tree, as follows: ignoring duplicated identity, and given that each node chooses b children independently to start each cycle, we observe that there are, on average, $b+E[m_p]+E[m_c] \approx b^2+2b$ nodes whose children will all be part of the source node's family tree. The term is dominated by the number of co-parents. Also, for the co-parent random variable, the ratio of the standard deviation to mean, $(np_c(1-p_c))^{0.5}/(n-1)\ p_c$, is very small, and hence the probability mass will center on the mean. Thus p_l can be approximated by $(1-b/(n-1))^{b^2+2b}$, where $(1-b/(n-1))$ is the probability that a given node is not selected by a particular parent picking b children out of n-1 nodes.

Simplifying and using the fact that $\lim_{\varepsilon \to 0}(1-\varepsilon)^{1/\varepsilon} = e^{-1}$, we obtain:

$$p_l \approx \left(1-\frac{b}{n-1}\right)^{b^2+2b} = \left(1-\frac{b}{n-1}\right)^{\frac{c^3 n}{b}} \left(\left(1-\frac{b}{n-1}\right)^{\frac{2c^3 n}{b}}\right)^{\frac{1}{b}}$$

$$\approx e^{-c^3} e^{-2c^3/b} \approx e^{-c^3}$$

as n/b becomes large.

Intuitively, the analytical result can be interpreted as follow: in each cycle, approximately $c^3 n$ duplicated copies of any given information frame are created in the network and an average of c^3 of them will be received by a given node. The probably that a given node fails to receive any copy of an information frame is the Poisson zero-arrival probability e^{-c^3}.

V. Simulation Result

We have verified through simulation of the children selection process that e^{-c^3} does indeed provide an excellent approximation to the probability that a node is left out of the family tree. As shown in Figure 4, when n is equal to 20, 100, and 500 respectively, the p_l as determined from simulation is very closely approximated by e^{-c^3}, particularly for the larger values of n.

We have performed simulations for synchronized and unsynchronized systems. With random node-to-node delays and without phase synchronization among the nodes, information frames may not be able to propagate to all members of the family tree during a cycle. Results in Figure 5 verify this fact. Figure 5 shows the frame non-delivery probability of a practical system in which cycles start at about the same time, but phases are not synchronized. Without phase synchronization, a larger value of c is needed to achieve the same non-delivery probability.

Also, in Figure 5 we vary the number of active nodes for each round, from a fixed number ($N_s = 4$) to being a fixed percentage

of n ($N_s = 0.25n$). The number of active sources has a relatively small impact on the performance.

Figure 6 represents a preliminary attempt to evaluate the delay performance of the GConf multiparty communication model. In a gossip-based protocol such as GConf, a node may receive multiple copies of the same information frame, and the information dissemination is determined by the arrival time of the first copy. For Figure 6, the delay between any two nodes is set to be normally distributed with mean of 50ms and standard deviation of 25ms. Then simulation is run to measure the complementary distribution function of the first-copy delay of all information frames, under n = 30 and b = 5 and 6. We observe that the 99-percentile of the first-copy delay distribution lies between 200ms to 250ms, which is somewhat greater than the 1.5 RTT of 150 ms. Such a level of delay may only be marginally acceptable for real-time multiparty conferencing applications.

VI. Conclusion

In this paper, we proposed a new multi-party information dissemination model called GConf which leverages peer-to-peer collaboration to solve with minimal delay the scalability problem in the full mesh communication model.

Compared with the full mesh architecture, the excess fanout in random gossip, manifested in the constant c being greater than 1, leads to excess traffic in the network. In future works, we will formulate different schemes for reducing this excess traffic while keeping the delay and frame loss at an acceptable level. We will also study ways to mitigate the effect of random delays and degradation due to lack of synchronization among nodes.

References

[1] A. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehart and D. Terry, "Epidemic algorithms for replicated database maintenance," *Proceedings of the Sixth Annual ACM Symposium on Principles of Distributed Computing,* pp. 1-12, 1987.

[2] R. Golding, K. Taylor, Santa Cruz University of California and Computer Research Laboratory, *Group Membership in the Epidemic Style.* University of California, Santa Cruz, Computer Research Laboratory, 1992,

[3] R. van Renesse, Y. Minsky and M. Hayden, "A gossip-style failure detection service," *Middleware,* vol. 98, pp. 55–70, 1998.

[4] K. P. Birman, M. Hayden, O. Ozkasap, Z. Xiao, M. Budiu and Y. Minsky, "Bimodal multicast," *ACM Transactions on Computer Systems (TOCS),* vol. 17, pp. 41-88, 1999.

[5] K. Guo, M. Hayden, R. van Renesse, W. Vogels and K. Birman, "GSGC: An Efficient Gossip-Based Garbage Collection Scheme for Scalable Reliable Multicast," *Technical Report,* 1997.

[6] I. Gupta, R. van Renesse and K. P. Birman, "A Probabilistically Correct Leader Election Protocol for Large Groups," 2000.

[7] K. Jenkins, K. Hopkinson and K. Birman, "A Gossip Protocol for Subgroup Multicast," *International Workshop on Applied Reliable Group Communication (WARGC),* 2001.

[8] A. M. Kermarrec, L. Massoulie and A. J. Ganesh, "Probabilistic reliable dissemination in large-scale systems," *IEEE Trans. Parallel Distrib. Syst.,* vol. 14, pp. 248-258, 2003.

[9] S. Boyd, A. Ghosh, B. Prabhakar and D. Shah, "Mixing times for random walks on geometric random graphs," *SIAM ANALCO,* 2005.