# Probabilistic Analysis of Message Forwarding

Louise E. Moser and P. M. Melliar-Smith
Department of Electrical and Computer Engineering
University of California, Santa Barbara
Santa Barbara, CA 93106 USA
moser@ece.ucsb.edu, pmms@ece.ucsb.edu

***Abstract*—In this paper, we present a novel algorithm that finds the probability density function for the number of distinct nodes reached, within a specific number of levels of message forwarding, for a fixed size network. The algorithm also finds the expected number of distinct nodes to which a message is forwarded, within a specific number of levels of message forwarding, as the sum of the number of nodes, weighted by the probability of reaching that number of nodes. In addition, the algorithm finds the probability density function for the number of distinct nodes at a given level of message forwarding, and the expected number of distinct nodes at that level. Using the algorithm, we calculate these probability density functions and expected values for various size networks, degrees of message forwarding, probabilities of message forwarding, and levels of message forwarding. The algorithm has application to the Internet, peer-to-peer networks, ad-hoc networks, and social networks, and to multicasting, gossiping, and rumor and epidemic protocols.**

***Keywords*—message forwarding; probabilistic analysis; multicasting and gossiping; rumor and epidemic protocols; peer-to-peer networks; ad-hoc and sensor networks; social networks***

## I. Introduction

Message forwarding is used in the Internet [8], peer-to-peer networks [15], ad-hoc networks [19], [25], [28], sensor networks [2], [3], [17], and social networks [1]. Forwarding forms the basis of multicasting [4], [20], gossiping [24], [29], and rumor and epidemic protocols [10], [18]. Forwarding has been applied to replication and data consistency [9], failure detection [30], and buffer management and message filtering [12]. Forwarding has also been applied to data aggregation [22] and decentralized search [26]. In each of these cases, messages are communicated to multiple nodes in the network.

One way of communicating messages to multiple nodes in a network is for the source node to multicast the messages directly to all of those nodes. Doing so might not be feasible in a large network. Another way of communicating messages to multiple nodes in a network is for the source node to transmit the message to a small number of nodes and to have each of those nodes forward the messages to a small number of nodes, and so on. Such message forwarding has the advantage that it shares the processing, buffering, and communication costs among multiple nodes, instead of placing the entire burden on the source node.

To avoid unconstrained flooding when messages are forwarded, and to conserve network bandwidth and reduce processing and buffering costs, several mechanisms have been adopted [8]. One mechanism requires a node that receives a message it previously received (a duplicate) to discard the duplicate message. A second mechanism includes a time-to-live or hop count in the message header. When the time-to-live or hop count bound is reached, the node no longer forwards the message. The time-to-live or hop count corresponds to the level of message forwarding that we employ in this paper. A third mechanism forwards a message according to a predetermined probability. When a node receives a non-duplicate message, the node forwards the message with that probability. The forwarding probability that we employ in this paper is such a mechanism. Other mechanisms to constrain message forwarding exist, and real systems might be more complex; however, in this paper, we focus on these three mechanisms.

The analysis of message forwarding is straightforward if the effects of duplicate messages and probabilistic forwarding are ignored, and other papers employ a simplistic analysis similar to (2) in Section IV. However, if these effects are taken into account, the problem is more challenging, as will become evident in Section V. When the number of nodes in the network is small, the probability of receiving a duplicate message is relatively high, which can result in substantial variability in the number of nodes that receive the message. In addition, small values of the forwarding probability can result in substantial variability in the number of nodes that receive a message.

This paper makes the following contributions:

- A novel algorithm that performs a probabilistic analysis for message forwarding and that calculates:
  - The probability density function for the number of distinct nodes reached, within a specific number of levels of message forwarding
  - The expected number of distinct nodes to which a message is forwarded in a fixed size network, within a specific number of levels of message forwarding
  - The probability density function for the number of new nodes at a given level of message forwarding
  - The expected number of new nodes at a given level of message forwarding.
- A set of results obtained from the algorithm for various values of the parameters:
  - Number $n$ of nodes in the network
  - Degree $c$ of message forwarding
  - Probability $f$ of message forwarding
  - Level $l$ of message forwarding.

The advantage of a probability density function is that it can predict the full range of behavior, rather than hiding that range behind a mean value. The results given in Section VI show that the range of behavior can be quite wide. In this paper, we use the acronym pdf for probability density function.

The probabilistic analysis of message forwarding, presented in this paper, is important for designers and users of protocols and applications that employ message forwarding.

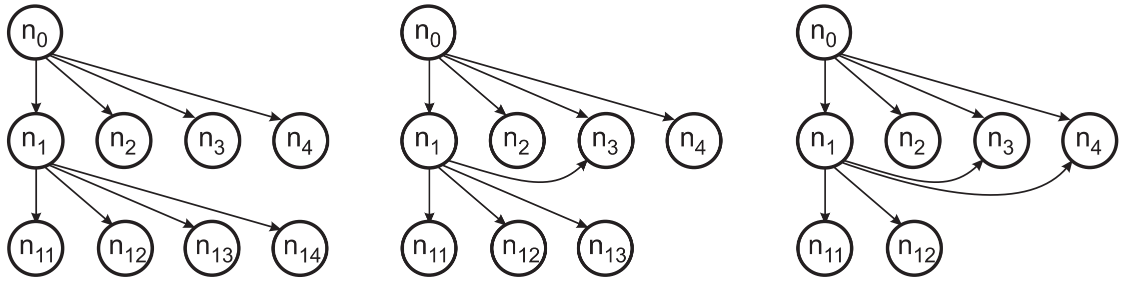


Fig. 1. Three examples of nodes at levels $l = 0, 1, 2$ with $c = 4$.

The rest of this paper is organized as follows. In Section II, we discuss related work. In Section III, we state the problem to be solved and, in Section IV, we present the foundations of our work. In Section V, we present the algorithm for analyzing the forwarding probabilities and, in Section VI, we present results obtained from the algorithm. Finally, in Section VII, we present conclusions and future work.

## II. Related Work

Much research has been undertaken on communicating information to multiple destinations in a network or distributed system by multicasting and gossiping. Message forwarding has been used as an alternative to multicasting by a single source node, and in gossiping protocols.

Deering and Cheriton [8] provide a survey of multicast routing, using spanning trees along which messages are forwarded in internetworks and extended local-area networks. Lin *et al.* [24] exploit directional information for gossiping in wide-area networks. Castro *et al.* [4] describe a scalable, decentralized multicast infrastructure based on gossiping, which is extended to a hierarchical infrastructure in [18].

Gossiping is particularly important for routing in wireless mobile ad-hoc networks. Hass *et al.* [19] use gossiping with probabilistic forwarding to reduce the message overhead of flooding in ad-hoc networks. Luo *et al.* [25] investigate route-driven gossiping for multicasting in ad-hoc networks. Sasson *et al.* [28] investigate probabilistic broadcast for flooding in wireless ad-hoc networks using percolation theory.

Forwarding is also important for sensor networks. Busse *et al.* [2] describe two forwarding schemes that trade off delivery rate and energy costs by maximizing energy efficiency. Gu and He [17] investigate data forwarding in low duty-cycle sensor networks in the presence of unreliable communication links. Cao *et al.* [3] present cluster-based forwarding for wireless sensor networks, where any node in a node's cluster may forward the message.

Other researchers use forwarding in delay-tolerant networks. Chen *et al.* [5] show that probability delegation forwarding has a lower routing cost (in terms of the number of duplicates of a message), and a similar delivery ratio, compared to delegation forwarding [11]. Hou and Shen [21] adaptively adjust the probability of replicating a message, to achieve a balance between the delivery delay and the number of copies of the message.

Theoretical research has addressed both deterministic and probabilistic algorithms for gossiping and multicasting. Hedetniemi *et al.* [20] present a survey of the theory of gossiping and broadcasting in communication networks. Shah [29] provides a comprehensive discussion of gossip algorithms. Farley [13] presents algorithms that construct broadcast networks with approximately the minimum number of links. He also determines upper and lower bounds to broadcast $m$ messages throughout a network of $n$ nodes [14]. However, he does not consider probability density functions.

The research on the spread of rumors [6], [10], [27] and epidemic processes [7], [16], [23] is interesting, and includes detailed theoretical analysis. However, both rumors and epidemics involve specific criteria for communicating and discarding messages, that are different from the general criteria used by our algorithm for message forwarding.

## III. The Problem

We consider a fully connected network $N$ with $n$ nodes, and we assume that communication between the nodes is reliable. A node $n_0$ sends a message to $c$, $c > 1$, randomly chosen nodes $\{n_1, n_2, \ldots, n_c\}$ at level 1, excluding itself, with forwarding probability $f$, $0 \leq f \leq 1$. Each node $n_i$, $1 \leq i \leq c$, forwards $n_0$'s message to $c$ randomly chosen nodes $\{n_{i1}, n_{i2}, \ldots, n_{ic}\}$ at level 2, excluding itself, with forwarding probability $f$, and so on. Note that $n_{ij}$ might be $n_0$ or $n_{i'}$, $1 \leq i' \leq c$, $i' \neq i$, and that $n_{ij}$ might be $n_{i'j'}$, $1 \leq i' \leq c$, $1 \leq j' \leq c$, $i' \neq i$. We refer to such nodes as duplicate nodes.

Figure 1 shows three examples of nodes at levels $l = 0, 1, 2$ with $c = 4$. In the middle example, $n_4 = n_{14}$ is a duplicate node. In the example on the right, $n_3 = n_{13}$ is a duplicate node and $n_4 = n_{14}$ is a duplicate node. Our algorithm eliminates such duplicate nodes.

## IV. Foundations

An upper bound $UB_s$ on the number of nodes at level $l$ in the network that have, or receive, $n_0$'s message is given by:

$$UB_s = c^l \tag{1}$$

The expression in (1) assumes that all of the nodes at level $l$ and previous levels are distinct. Note that this expression is independent of $n$, and the least upper bound might be $n$.

The expected number $E_s$ of distinct nodes at level $l$, to which $n_0$'s message is forwarded, is calculated by our algorithm after eliminating duplicates at the same or different levels, and is dependent on $n$. Thus, $E_s \leq UB_s$.

Similarly, an upper bound $UB_d$ on the number of nodes in the network that have, or receive, $n_0$'s message at levels 0 through $l$ is given by:

$$\begin{aligned} UB_d &= 1 + c + c^2 + \ldots + c^l \\ &= \begin{cases} \frac{c^{l+1}-1}{c-1} & \text{if } c > 1 \\ l+1 & \text{if } c = 1 \end{cases} \end{aligned} \tag{2}$$

```
evalp(n, a, b, k)
if (a > 0 and b > 0)
    topj = min(n − b, a − k − 1)
    if (k > 0)
        prod1 = 1.0
        for j = 0 to k − 1 do
            prod1 = prod1 ×((b − j)/(n − j))
        prod2 = 1.0
        for j = 0 to topj do
            prod2 = prod2 ×((n − b − j)/(n − k − j))
        prod = choose(a, k) × prod1 × prod2
    else // k = 0
        for j = 0 to topj do
            prod2 = prod2 ×((n − b − j)/(n − j))
        prod = prod2
else // a = 0 or b = 0
    if (k > 0)
        prod = 0.0
    else // k = 0
        prod = 1.0
return prod
```

Fig. 2. The method for evaluating the probability given in (3), where $0 \leq k \leq a \leq b \leq n$.

The expression in (2) overstates the number of nodes at levels 0 through $l$, if some of these nodes are duplicates of nodes at the same or different levels. Note that this expression is independent of $n$, and the least upper bound might be $n$.

The expected number $E_d$ of distinct nodes at levels 0 through $l$, to which $n_0$'s message is forwarded, is calculated by our algorithm after eliminating duplicates at the same or different levels, and is dependent on $n$. Thus, $E_d \leq UB_d$.

To find the expected number $E_s$ of distinct nodes at level $l$, to which $n_0$'s message is forwarded, and the expected number $E_d$ of distinct nodes at levels 0 through $l$, to which $n_0$'s message is forwarded, we employ the following theorem.

**Theorem.** Let $N$ be a set having cardinality $n$, $A$ be a subset of $N$ having cardinality $a$, and $B$ be a subset of $N$ having cardinality $b$, where $a \leq b$. The pdf $p(k)$, $0 \leq k \leq a$, that $A \cap B$ has cardinality $k$ is given by:

$$p(k) = \binom{a}{k} \prod_{j=0}^{\min(b,k-1)} \frac{b-j}{n-j} \prod_{j=0}^{\min(n-b,a-k-1)} \frac{n-b-j}{n-k-j} \qquad (3)$$

Note that, in (3), if the upper index is less than 0, then we have an empty product, which is equal to 1. The proof of the theorem is omitted due to space constraints.

The pseudocode to evaluate the probability in (3) is given in Figure 2. In the pseudocode, we use the standard *choose* and *min* functions.

## V. The Algorithm

The algorithm for the forwarding analysis is illustrated in Figures 3 and 4, and the pseudocode for the algorithm is given in Figure 5.

The algorithm calculates the pdf for the number of nodes at level $l$, $0 \leq l \leq L$, to which $n_0$'s message is forwarded, and the pdf for the number of nodes to which $n_0$'s message is forwarded at levels 0 through $L$. It also calculates the expected number of nodes at level $l$, and the expected number of nodes at levels 0 through $L$. The calculation involves a combination of probabilities and probability density functions, and depends on independence of the considered sets.

We let $S_0 = \{n_0\}$ be the set consisting of the node $n_0$ at level 0, $S_1 = \{n_1, n_2, \ldots, n_c\}$ be the set of distinct nodes at level 1 to which $n_0$ forwarded its message and, in general, $S_l$ be the set of distinct nodes at level $l$ to which the nodes at level $l-1$ forwarded $n_0$'s message. We consider the upper bound $smax_l$ on the cardinality of $S_l$, defined by $smax_0 = 1$, $smax_1 = c$ and, in general, $smax_l = \min\{c^l, n\}$ for $l \geq 0$.

We consider the pdf $s[l,k]$ for the cardinality of the set $S_l$, $0 < k < smax_l$. Note that $s[l,0]$ is the probability that there exists no new node in the set $S_l$ (*i.e.*, all of the nodes at level $l$ already occurred at levels 0 through $l-1$), $s[l,1]$ is the probability that there exists one new node in the set $S_l$ (*i.e.*, there exists only one node at level $l$ that has not yet occurred at levels 0 through $l-1$), and so on.

We let $D_l = \cup_{i=0}^{l} S_i$ be the set of distinct nodes at levels 0 through $l$, where $l \geq 0$. We consider the upper bound $dmax_l$ on the cardinality of the set $D_l$, defined by $dmax_0 = 1$, $dmax_1 = 1 + c$ and, in general, $dmax_l = \min\{1 + c + c^2 + \cdots + c^l, n\}$ for $l \geq 0$. Note that, for $c > 1$, $dmax_{l-1} \leq smax_l \leq dmax_l$. We consider the pdf $d[l,k]$ for the cardinality of the set $D_l$, $0 \leq k \leq dmax_l$.

For $l \geq 2$, $D_l = D_{l-1} \cup S_l$. Attempting to derive $s[l,j]$ as the difference of $d[l-1,i]$ and $d[l,i]$ leads to numerical instabilities. Attempting to derive $d[l,i]$ directly from $d[l-1,i]$ and $s[l,j]$ leads to a combination of sets that are not independent, resulting in erroneous results. Thus, we are led to the derivation described below, in which $s[l,j]$ and $d[l,i]$ are derived independently from $sprev[j]$.

As illustrated in Figure 3, each node in $S_{l-1}$, as represented by the pdf $s[l-1,j]$, $0 \leq j \leq smaxlminus1$, forwards $n_0$'s message to $c$ nodes in the set $C_{lk}$. For each number of nodes in the set $D_{l-1}$, as represented by the pdf $d[l-1,i]$, $0 \leq i \leq dmax_{l-1}$ and, for each set $C_{lk}$, the algorithm calculates the probabilities that the nodes in $C_{lk}$ are duplicates, either of a node already in $D_{l-1}$ or of a node already added through a previous set $C_{lk'}$ at level $l$, and then accumulates a pdf $scurr[j]$, $0 \leq j \leq smaxl$, that represents the probabilities of the numbers of new nodes added at level $l$. These pdfs are weighted by the probability of $i$ nodes in the set $D_{l-1}$ and are accumulated in the pdf $d[l,i]$, $0 \leq i \leq dmax_l$, and the pdf $s[l,j]$, $0 \leq j \leq smaxl$.

More specifically, we calculate the pdf $s[l,h]$, $0 \leq h \leq smax_l$, for the cardinality of the set $S_l$. For $l = 0$, $smax_0 = 1$ and the pdf is $s[0,0] = 0.0$ and $s[0,1] = 1.0$. For $l = 1$, $smax_1 = c$ and the pdf is $s[1,c] = 1.0$ and $s[0,h] = 0.0$ for $0 \leq h \leq c$, $h \neq c$.

We consider the set $C_{lk}$ of $c$ nodes at level $l$ to which node $k$ in the set $S_{l-1}$ at level $l-1$ forwarded $n_0$'s message. We consider the pdf $c_{lk}[j]$, $0 \leq j \leq c$, that there exists such a node $k$ in the set $S_{l-1}$, whose forwarding leads to the set $C_{lk}$. We calculate the probability $q[l-1,k]$, $0 \leq k \leq smax_{l-1}$, that there exists such a node $k$ in the set $S_{l-1}$, derived from the pdf $s[l-1,j]$, $0 \leq j \leq smaxlminus1$. We let $f$ be the probability that such a node $k$ in the set $S_{l-1}$ forwards $n_0$'s message to $c$ nodes at level $l$. Thus, we obtain the pdf $c_{lk}[j]$, $0 \leq j \leq c$, given by $c_{lk}[0] = 1 - (f \times q[l-1,k])$, $c_{lk}(c) = f \times q[l-1,k]$, and $c_{lk}[j] = 0$ for $0 < j < c$.

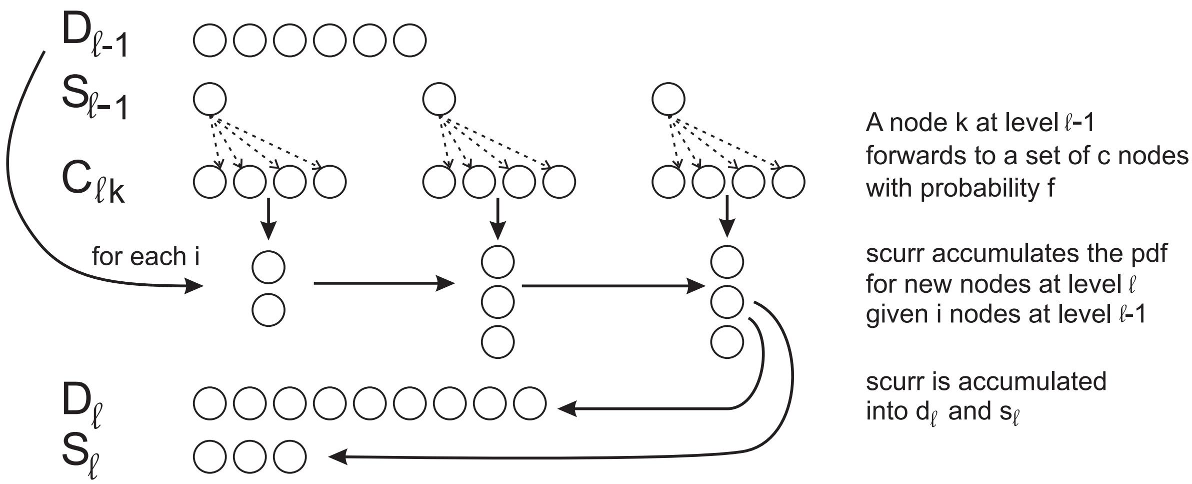


Fig. 3. The relationship between the sets whose pdfs the algorithm calculates.

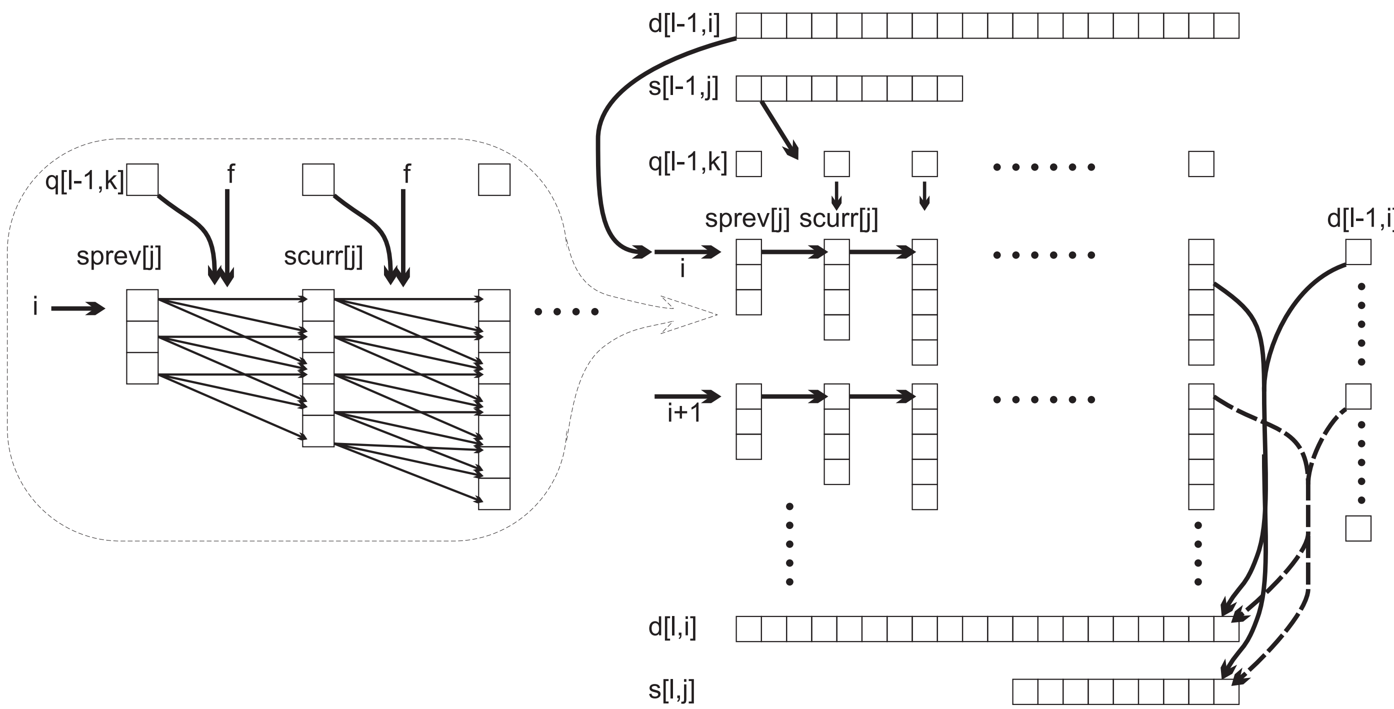


Fig. 4. The computation of $scurr[j]$ is performed for each index $i$ of $d[l-1,i]$ and each index $k$ of $q[l-1,k]$. As shown on the left of the figure, the previous value of $scurr[j]$ is held in $sprev[j]$ and is converted into the next value of $scurr[j]$. When $scurr[j]$ has been calculated for all of the values of $k$, it is weighted by $d[l-1,i]$ and accumulated into $d[l,i]$ and $s[l,j]$. Another set of values for $scurr[k]$ is then calculated for the next value of $i$.

The illustration in Figure 4 is an elaboration of that in Figure 3. Each value of the pdf $scurr[j]$ is derived from the previous value of $scurr[j]$, which is represented by the pdf $sprev[j]$, for $0 \leq j \leq smaxl$. Each index $j$ in $sprev[j]$ represents $j$ nodes already added at level $l$, whereas each index $i$ in $d[l-1,i]$ represents $i$ nodes already present at level $l-1$ and all previous levels. Given $i+j$ existing nodes, $scurr[j]$ is calculated using the $evalp()$ method and the probability $q[l-1,k]$ that there exists a node $k$ in $S_{l-1}$ and the probability $f$ that the node forwards the message to $c$ nodes.

When the values of $scurr[j]$, $0 \leq j \leq smaxl$, have been calculated for all values of $k$ for which nodes are present in $S_{l-1}$, the pdf $scurr[j]$, $0 \leq j \leq smaxl$, is weighted by $d[l-1,i]$, $0 \leq i \leq dmaxlminus1$, and accumulated into $d[l,i]$, $0 \leq i \leq dmaxl$, and $s[l,j]$, $0 \leq j \leq smaxl$. The accumulation into $d[l,i]$, $0 \leq i \leq dmaxl$, is offset by $i$ to represent the $i$ nodes already present at level $l-1$ and all previous levels.

From the pdf $s[l,j]$, $0 \leq j \leq smax_l$, we find the expected number of new nodes at level $l$ as a sum of $j$ weighted by $s[l,j]$. Similarly, from the pdf $d[l,i]$, $0 \leq i \leq dmax_l$, we find the expected number of nodes reached at level $l$ or previous levels as a sum of $i$ weighted by $d[l,i]$.

The pseudocode for the forwarding analysis is given in Figure 5. As is evident from the pseudocode, the algorithm has polynomial complexity. Substantial reductions in the computation time are achieved by ignoring small terms, terminating loops when terms become small, and recording an appropriate starting value for the next iteration of the loops.

```
Algorithm
for i = 0 to n do
    for j = 0 to n do
        s[i, j] = 0.0
        d[i, j] = 0.0
        q[i, j] = 0.0
        if j <= c then
            q[1, j] = 1.0
d[1, 1 + c] = 1.0
smaxlminus1 = min(c, n)
dmaxlminus1 = min(1 + c, n)
for l = 2 to L do
    smaxl = min(smaxlminus1 × c, n)
    dmaxl = min(dmaxlminus1 + smaxl, n)
    for i = c + 1 to dmaxlminus1 do
        if d[l − 1, i] > 0 then
               for j = 0 to smaxl do
                  sprev[j] = 0.0
                  scurr[j] = 0.0
            sprev[0] = 1.0
            for k = 1 to smaxlminus1 do
                if q[l − 1, k] > 0.0000001 then
                    for j = 0 to smaxl do
                        toph = min(c, smaxl − j)
                        for h = 0 to toph do
                            scurr[j + h] = scurr[j + h] + sprev[j] × f × q[l − 1, k] × evalp(n, c, i + j, c − h)
                        scurr[j] = scurr[j] + sprev[j] × (1 − f × q[l − 1, k])
                    for j = 0 to smaxl do
                        sprev[j] = scurr[j]
                        scurr[j] = 0.0
            for j = 0 to smaxl do
                s[l, j] = s[l, j] + sprev[j] × d[l − 1, i]
                d[l, i + j] = d[l, i + j] + sprev[j] × d[l − 1, i]
    sum = 0.0
    for j = 0 to smaxl do
        sum = sum + s[l, j] × j
    Es = sum
    sum = 0.0
    for k = 0 to dmaxl do
        sum = sum + d[l, k] × k
    Ed = sum
    for k = 1 to smaxl do
        sum = 0.0
        for j = 0 to k − 1 do
            sum = sum + s[l, j]
        q[l, k] = 1 − sum
    smaxlminus1 = smaxl
    dmaxlminus1 = dmaxll
```

Fig. 5. The algorithm for calculating the pdf $s[l, j]$ and the expected number $Es$ of new nodes at each level $l$, where $0 \leq l \leq L$, that have, or receive $n_0$'s message, and the pdf $d[l, k]$ and the expected number $Ed$ of distinct nodes at levels 0 through $L$ that have, or receive, $n_0$'s message.

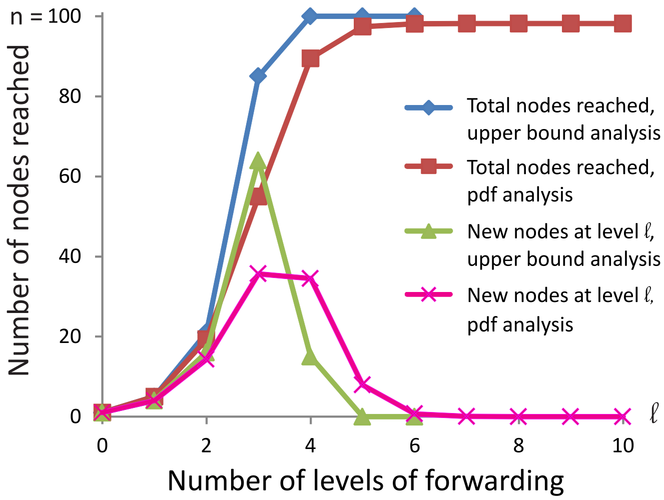


Fig. 6. The expected number of nodes reached at level $l$ and all previous levels, and the number of new nodes at level $l$, from the upper bound analysis and the algorithm for different values of $l$, when $n = 100$, $c = 4$, and $f = 1.0$.

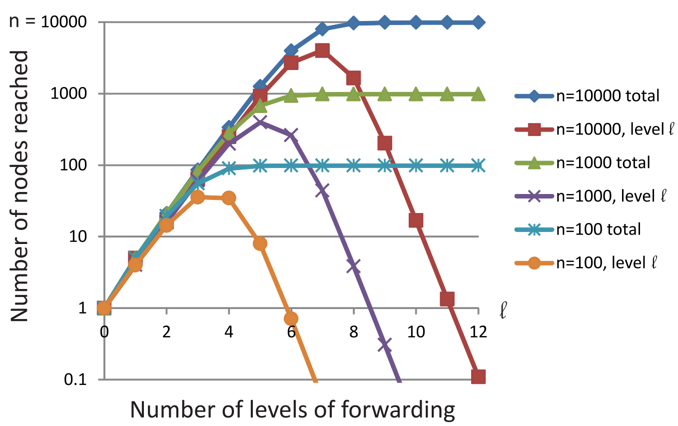


Fig. 7. The expected number of nodes reached at level $l$ and all previous levels, and the expected number of new nodes at level $l$, for different values of $n$ and different values of $l$, when $c = 4$ and $f = 1.0$, shown on a logarithmic scale.

## VI. Results

We ran the algorithm for several representative values of the number $n$ of nodes in the network, the degree $c$ of message forwarding, the probability $f$ of message forwarding, and the number $l$ of levels of message forwarding. We checked that the probabilities sum to 1.0. The results are shown in Figures 6-13, and are discussed below.

Figure 6 shows the expected number of nodes reached at level $l$ and all previous levels, and the number of new nodes at level $l$, from the upper bound analysis and the algorithm, for $n = 100$, $c = 4$ and $f = 1.0$ for different values of $l$. In the figure, we see that the effect of duplicates is to reduce the number of nodes reached. For example, the expected number of nodes reached at level $l = 3$ and all previous levels is 55, rather than the 85 nodes indicated by the upper bound analysis. Likewise, the expected number of new nodes reached at level $l = 3$ is 36, rather than the 64 new nodes indicated by the upper bound analysis.

Figure 7 uses a logarithmic scale to compare the expected number of nodes reached at level $l$ and all previous levels, and also the expected number of new nodes at level $l$, for $n = 100, 1000, 10000$ and for different values of $l$, when $c = 4$ and $f = 1.0$. Not surprisingly, for larger values of $n$, more levels of message forwarding are required to reach almost all of the nodes: 4 levels for $n = 100$, 6 levels for $n = 1000$, and 8 levels for $n = 10000$.

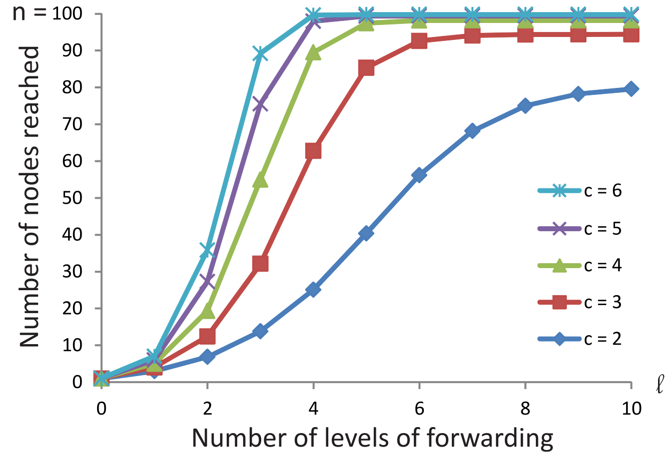


Fig. 8. The expected number of nodes reached at level $l$ and all previous levels for different values of $c$ and different values of $l$, when $n = 100$ and $f = 1.0$.

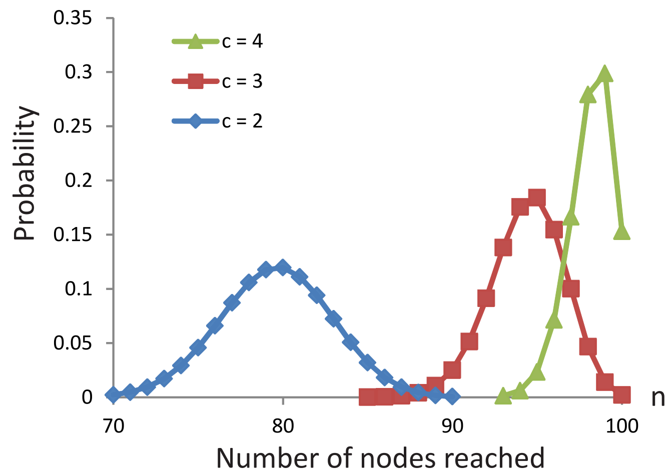


Fig. 9. The pdfs for the number of nodes reached at level $l = 10$ and all previous levels for different values of $c$, when $n = 100$ and $f = 1.0$.

Figure 8 shows a comparison of the expected number of nodes reached at level $l$ and all previous levels for $c = 2, 3, 4, 5, 6$ and for different values of $l$, when $n = 100$ and $f = 1.0$. In the figure, we see that, for smaller values of $c$, the horizontal asymptote for the expected number of nodes reached is substantially less than 100: only 80 for $c = 2$, 94 for $c = 3$, and 98 for $c = 4$. Not only is the expected number of nodes reached much lower, but also the range of that number of nodes is much wider, than for the upper bound analysis.

Figure 9 shows a comparison of the pdfs for the number of nodes reached at level $l = 10$ and all previous levels for $c = 2, 3, 4$, when $n = 100$ and $f = 1.0$. Particularly striking is the wide range of values of the pdfs for the number of nodes reached. For $c = 2$, the range is from 70 to 90 and, for $c = 3$, the range is from 85 to 100. The pdf for $c = 2$ is wider than the pdf for $c = 3$ or $c = 4$ because, for $c = 2$, fewer nodes are rejected as duplicates.

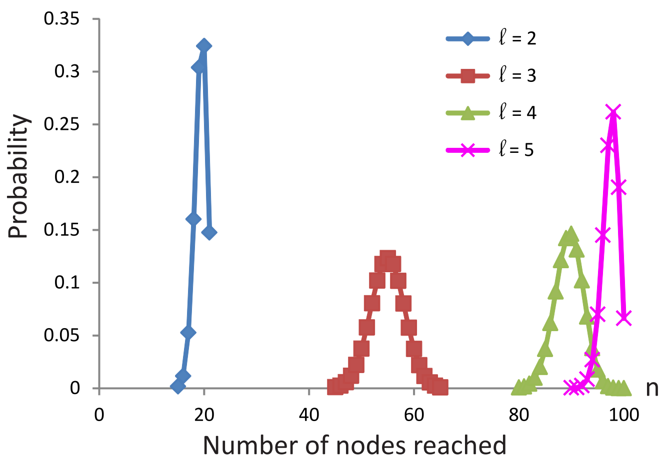


Fig. 10. The pdfs for the number of nodes reached at level $l$ and all previous levels for different values of $l$, when $n = 100$, $c = 4$, and $f = 1.0$.

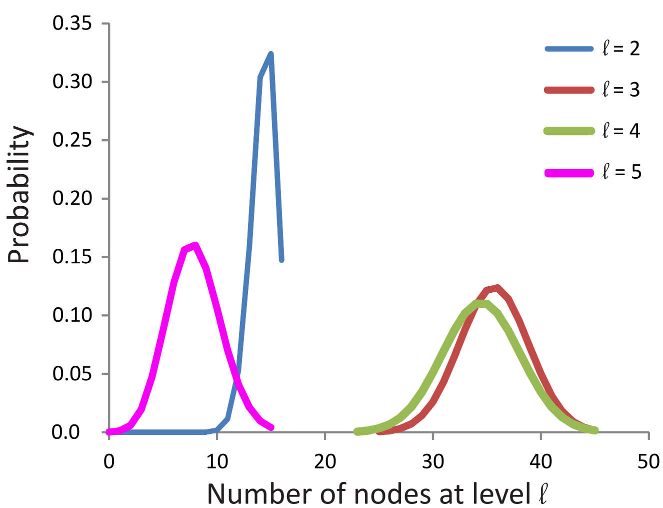


Fig. 11. The pdfs for the number of new nodes at level $l$ for different values of $l$, when $n = 100$, $c = 4$, and $f = 1.0$.

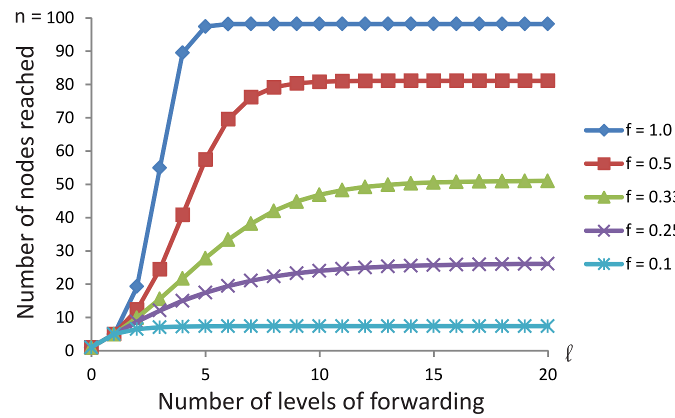


Fig. 12. The expected number of nodes reached at level $l$ and all previous levels for different values of $f$ and different levels of message forwarding, when $n = 100$ and $c = 4$.

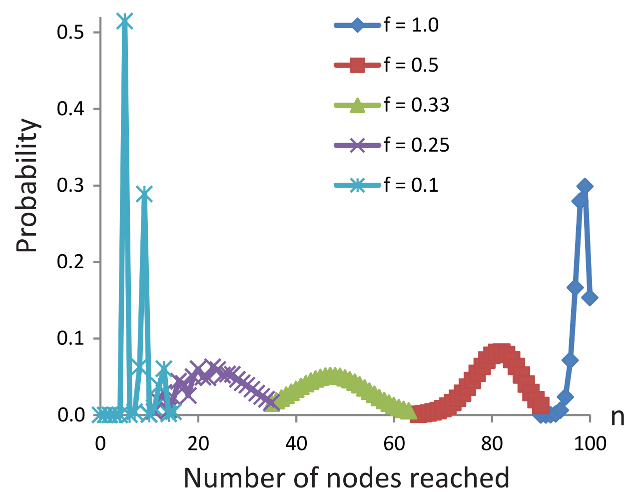


Fig. 13. The pdfs for the number of nodes reached at level $l = 10$ and all previous levels for different values of $f$, when $n = 100$ and $c = 4$.

Figure 10 shows a comparison of the pdfs for the number of nodes reached at level $l$ and all previous levels for $l = 2, 3, 4, 5$, when $n = 100$, $c = 4$, and $f = 1.0$. Again striking is the wide range of the values of the pdfs for the number of nodes reached, particularly for $l = 3$ and $l = 4$ levels of message forwarding. For $l = 3$, the number of nodes reached ranges from 45 to 60 whereas, for $l = 4$, the number of nodes reached ranges from 83 to 92. Not only are there wide ranges for the number of nodes at level $l$ and all previous levels, but also the ranges for the number of new nodes at level $l$ are wide.

Figure 11 shows a comparison of the pdfs for the number of new nodes at level $l$, for $l = 2, 3, 4, 5$, when $n = 100$, $c = 4$, and $f = 1.0$. The number of new nodes at a level also exhibits a wide range. At level $l = 3$, the number of new nodes ranges from 29 to 43 whereas, at level $l = 4$, the number of new nodes ranges from 27 to 42. These wide ranges make the performance of message forwarding protocols and algorithms somewhat unpredictable.

Figure 12 shows a comparison of the expected numbet of nodes reached at level $l$ and all previous levels for $f = 1.0, 0.5, 0.33, 0.25, 0.1$ and different levels of message forwarding, when $n = 100$ and $c = 4$. As the probability $f$ of message forwarding decreases, the expected number of nodes reached decreases dramatically. In particular, the algorithm yields horizontal asymptotes for the expected number of nodes reached of 81 for $f = 0.5$, 51 for $f = 0.33$, and only 26 for $f = 0.25$. Not only do small values of $f$ result in lower values of the expected number of nodes reached at level $l$ and all previous levels, but also the ranges for the number of nodes reached at level $l$ and all previous levels are wide.

Figure 13 shows a comparison of the pdfs for the number of nodes reached at level $l = 10$ and all previous levels for $f = 1.0, 0.5, 0.33, 0.25, 0.1$, when $n = 100$ and $c = 4$. Again striking is the wide range of the values of the pdfs for the number of nodes reached. For $f = 0.5$, the number of nodes reached ranges from 74 to 93. For $f = 0.33$, the number of nodes reached ranges from 36 to 64. For $f = 0.25$, the number of nodes reached ranges from 12 to 40. Also interesting, for small values of $f$ such as $f = 0.1$, is the periodicity in the number of nodes reached, with the period 4 corresponding to the value $c = 4$.

The results presented here are important for designers and users of protocols and applications that use message forwarding, but they might need to run the algorithm for their own parameter values.

## VII. Conclusions and Future Work

We have presented a novel algorithm that finds the probability density function for the number of distinct nodes, and the expected number of distinct nodes, to which a message is forwarded, at a given level and all previous levels of message forwarding. The algorithm also finds the probability density function for the number of distinct nodes, and the expected number of distinct nodes, to which a message is forwarded, at a particular level of message forwarding.

Using the algorithm, we have obtained results for various values of the number of nodes in the network, the degree of message forwarding, the probability of message forwarding, and the number of levels of message forwarding. The expected number of distinct nodes reached at a given level and all previous levels, and the expected number of new nodes reached at a particular level, obtained from the algorithm are substantially less than the corresponding numbers obtained from the upper bound analysis. Furthermore, as the probability of message forwarding decreases, the expected number of nodes reached decreases quite rapidly. Moreover, the number of nodes reached at a given level and all previous levels, and the number of new nodes reached at a particular level, exhibit a wide range, particularly for smaller values of the degree of message forwarding and smaller values of the probability of message forwarding.

In future work, we plan to investigate forwarding within and between neighborhoods of nodes, where the neighborhoods are completely connected but the network might not be. We also plan to investigate forwarding within a network that is subject to node faults and link faults, including Byzantine faults. In addition, we plan to investigate the time to reach a certain number of nodes, given a particular probability density function for the time that a node takes to forward messages. We also plan to apply the results of this probabilistic analysis of message forwarding to our existing iTrust information publication, search and retrieval network [26].

## Acknowledgement

This research was supported in part by U.S. National Science Foundation grant number NSF CNS 10-16193.

## References

[1] D. J. Bartholomew, *Stochastic Models for Social Processes*, 3rd ed., New York, NY: John Wiley & Sons, Jan. 1982.

[2] M. Busse, T. Haenselmann and W. Effelsberg, "Energy-efficient forwarding schemes for wireless sensor networks," in *Proc. International Symposium on a World of Wireless, Mobile and Multimedia Networks*, Buffalo, NY, June 2006, pp. 125–133.

[3] Q. Cao, T. Abdelsaher, T. He and R. Kravets, "Cluster-based forwarding for reliable end-to-end delivery in wireless sensor networks," in *Proc. IEEE International Conference on Computer Communications*, May 2007, Anchorage, AK, pp. 1928–1936.

[4] M. Castro, P. Druschel, A. M. Kermarrec and A. I. Rowstron, "Scribe: A large-scale and decentralized application-level multicast infrastructure," *IEEE Jour. Selected Areas in Communications*, vol. 20, no. 8, pp. 1489–1499, 2002.

[5] X. Chen, "Probability delegation forwarding in delay-tolerant networks," in *Proc. 18th International Conference on Computer Communications and Networks*, San Francisco, CA, Aug. 2009, pp. 1–6.

[6] D. J. Daley and D. G. Kendall, "Stochastic rumours," *IMA Jour. Applied Mathematics*, vol. 1, no. 1, pp. 42–55, 1965.

[7] D. J. Daley and J. Gani, *Epidemic Modeling: An Introduction*, Cambridge, UK: Cambridge University Press, 1999.

[8] S. E. Deering and D. R. Cheriton, "Multicast routing in datagram internetworks and extended LANs," *ACM Trans. Computer Systems*, vol. 8, no. 2, pp. 85–110, 1990.

[9] A. J. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehart and D. Terry, "Epidemic algorithms for replicated database maintenance," in *Proc. 6th Annual ACM Symposium on Principles of Distributed Computing*, 1987, pp. 1–12.

[10] K. Dietz, "Epidemics and rumours: A survey," *Jour. Royal Statistical Society*, series A, vol. 130, no. 4, pp. 505-528, 1967.

[11] V. Erramilli, A. Crovella, A. Chaintreau and C. Diot, "Delegation forwarding," in *Proc. Tenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*, New Orleans, LA, May 2009, pp. 251–259.

[12] P. T. Eugster, R. Guerraoui, A. M. Kermarrec and L. Massoulie, "Epidemic information distribution in distributed systems," *IEEE Computer*, vol. 37, iss. 5, pp. 60–67, May 2004.

[13] A. M. Farley, "Minimal broadcast networks," *Networks*, vol. 9, iss. 4, pp. 313–322, 1979.

[14] A. M. Farley, "Broadcast time in communication networks," *SIAM Jour. Applied Mathematics*, vol. 39, no. 2, pp. 385–390, Oct. 1980.

[15] Gnutella, http://en.wikipedia.org/wiki/Gnutella

[16] W. Goffman and V. A. Newill, "Communication and epidemic processes," in *Proc. Royal Society of London*, series A, vol. 298, no. 1454, pp. 316–334, May 1967.

[17] Y. Gu and T. He, "Data forwarding in extremely low duty-cycle sensor networks with unreliable communication links," in *Proc. 5th International Conference on Embedded Networked Sensor Systems*, Sydney, Australia, Nov. 2007, pp. 321–334.

[18] I. Gupta, A. M. Kermarrec and A. J. Ganesh, "Efficient and adaptive epidemic-style protocols for reliable and scalable multicast," *IEEE Trans. Parallel and Distributed Systems*, vol. 17, no. 7, pp. 593–605, 2006.

[19] Z. J. Hass, J. Y. Halpern and L. Li, "Gossip-based ad hoc routing," *IEEE/ACM Trans. Networking*, vol. 14, no. 3, pp. 479–491, June 2006.

[20] S. M. Hedetniemi, S. T. Hedetniemi and A. L. Liestman, "A survey of gossiping and broadcasting in communications networks," *Networks*, vol. 18, pp. 319–349, 1988.

[21] F. Hou and X. Shen, "An adaptive forwarding scheme for message delivery over delay tolerant networks," in *Proc. IEEE GLOBECOM*, Honolulu, HI, Nov.–Dec. 2009, pp. 1–5.

[22] M. Jelasity, A. Montresor and O. Babaoglu, "Gossip-based aggregation in large dynamic networks," *ACM Trans. Computer Systems*, vol. 23, no. 3, pp. 219–252, Aug. 2005.

[23] D. G. Kendall, "Deterministic and stochastic epidemics in closed populations," in *Proc. Third Berkeley Symposium on Mathematical Statistics and Probability*, vol. 4, 1956, pp. 149–165.

[24] M. J. Lin and K. Marzullo, "Directional gossip: Gossip in a wide area network," in *Proc. Third European Dependable Computing Conference*, LNCS 1667, Springer, 1999, pp. 364–379.

[25] J. Luo, P. T. Eugster and J. P. Hubaux, "Route driven gossip: Probabilistic reliable multicast in ad hoc networks," in *Proc. IEEE INFOCOM*, vol. 3, Apr. 2003, pp. 2229–2239.

[26] P. M. Melliar-Smith, L. E. Moser, I. Michel Lombera and Y. T. Chuang, "iTrust: Trustworthy information publication, search and retrieval," in *Proc. 13th International Conference on Distributed Computing and Networking*, Hong Kong, China, Jan. 2012, LNCS 7129, Springer, pp. 351–366.

[27] A. Rapoport and L. I. Rebhun, "On the mathematical theory of rumor spread," *Bull. Mathematical Biophysics*, vol. 14, pp. 375–383, 1952.

[28] Y. Sasson, D. Cavin and A. Schiper, "Probabilistic broadcast for flooding in wireless ad hoc networks," in *Proc. IEEE Wireless Communications and Networking Conference*, vol. 2, Mar. 2003, pp. 1124–1130.

[29] D. Shah, "Gossip algorithms," *Foundations and Trends in Networking*, vol. 3, no. 1, pp. 1–125, 2008.

[30] R. van Renesse, Y. Minsky and M. Hayden, "A gossip-style failure detection service," in *Proc. IFIP Middleware Conference*, Lake District, UK, Sep. 1998, pp. 55–70.