Byzantine Modification Detection in Multicast Networks using Randomized Network Coding

Tracey Ho, Ben Leong, Ralf Koetter, Muriel Médard, Michelle Effros, and David R. Karger

***Abstract*— Distributed randomized network coding is a flexible and robust approach to transmitting and compressing information in multi-source multicast networks. In this paper, we show how the path diversity and distributed randomness of this approach can be exploited to provide for Byzantine modification detection. This is achieved by incorporating a simple polynomial hash value into each packet, which adds minimal computational and communication overhead. The effectiveness of our scheme relies only on a Byzantine attacker being unable to design and supply modified packets with complete knowledge of other packets received by other nodes. The detection probability can be traded off against the overhead (i.e., the ratio of hash bits to data bits) – the detection probability increases with the overhead, as well as with the number of unmodified packets obtained at the receiver whose contents are unknown to the attacker.**

I. Introduction

Distributed randomized network coding was introduced in [1] as a simple, robust means of transmitting and compressing information in multi-source multicast networks. Network coding offers many advantages over conventional tree-based multicast, like good path diversity, robustness and minimal routing state [1], [2], [3].

In an overlay multicast or ad hoc multicast setting, end hosts help to forward packets to other end hosts. Such networks are thus more susceptible to Byzantine (i.e., arbitrary) attacks from compromised end hosts, which have access to the same source information as other end hosts, and can forward to them arbitrarily modified information. In this paper, we show that Byzantine modification detection capability can be added to a multicast scheme based on randomized network coding, with minimal additional computational and communication overhead, by incorporating a simple polynomial hash value in each packet. The key insight in our approach is that path diversity, coupled with the randomized and distributed choice of codes, makes it hard for an attacker to observe or predict the exact combinations of source information in all other packets received at the receivers. With our approach, a receiver can detect Byzantine modifications with high probability, as long as these modifications have not been designed using knowledge of all the other packets it receives.

The detection probability can be traded off against the overhead (i.e., the ratio of hash bits to data bits) – the detection probability increases with the overhead, as well as with the number of unmodified packets obtained at the receiver whose contents are unknown to the attacker. Depending on the application, various responses may be employed upon detection of a Byzantine fault, such as collecting more packets from different nodes to obtain a consistent decoding set, or employing a more complex Byzantine agreement algorithm to identify the Byzantine node(s).

We are able to use a simple polynomial function instead of a complex cryptographic hash function (such as MD5) because our scheme's effectiveness depends only on the fact that there are independent sources of randomness, not all of which are known to the attacker because of path diversity. The use of a simple polynomial function is desirable because it incurs less computational overhead than existing cryptographic hashes.

A. Background and related work

Distributed randomized network coding is a recently proposed approach for multi-source multicast in a distributed setting. In this technique, nodes independently select random linear mappings from inputs onto outputs over some finite field, which achieves all feasible connections with probability tending to 1 as the field size grows [1]. The receivers need only know the overall linear combination of source processes in each of their incoming signals. This information can be sent through the network as a vector, for each signal, of coefficients corresponding to each of the source processes, updated at each coding node by applying the same linear mappings to the coefficient vectors as to the information signals. This approach efficiently exploits multiple available (possibly shared) paths to the receiver nodes, achieving greater ro-

Tracey Ho and Muriel Médard are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139, e-mail: {trace, medard}@mit.edu

Ben Leong and David R. Karger are with the Computer Science and Artificial Intelligence Laboratory, MIT, e-mail {benleong, karger}@csail.mit.edu

Ralf Koetter is with the Coordinated Science Laboratory, University of Illinois, Urbana, IL 61801, e-mail: koetter@csl.uiuc.edu

Michelle Effros is with the Data Compression Laboratory, California Institute of Technology, Pasadena, CA 91125, e-mail: effros@caltech.edu

bustness to link failures and errors in the random selection of codes as excess capacity in the network increases [2]. Chou et al. proposed a practical packet-based implementation in which source packets are divided into generations, and only packets in the same generation are linearly combined [3].

The Byzantine problem was first formalized in [4], and has been studied extensively in a variety of contexts such as reliable distributed networks of processors [5], [6] and secure network communications [7], [8], [9], [10], [11]. These and other existing works generally either use cryptographic functions, multiple rounds of message passing or some combination of the two to detect and recover from Byzantine faults. References [6] and [8] optimize for normal performance by using less complex message authentication codes and signed digests respectively during normal operation, resorting to more complex recovery mechanisms only upon detection of a fault. Our technique allows for detection without the use of any cryptographic functions (thereby incurring little computation overhead), and can similarly be used in conjunction with more complex recovery techniques which are activated upon detection of a Byzantine fault.

The reliance on random values unknown to the attacker is reminiscent of one-time pads [12], but our scheme is different because the one-time pad provides secrecy and not authenticity[1], while our scheme attempts to provide the latter. Also, unlike one-time pads, the burden of generating the random values is distributed over the network rather than falling solely on the source. Cai and Yeung [13] have also studied the problem of providing secrecy, in a network coding setting.

II. Model

We consider multi-source multicast mesh networks with multiple paths between each source and receiver. This set-up encompasses a rich family of problems, such as traditional multicast for content delivery and the reach-back problem for sensor networks, in which several, possibly correlated, sources transmit to a single receiver.

Consider a set of r source packets which are coded together and multicast, using distributed randomized network coding in the finite field $\mathbb{F}_q$. Let the data content of each packet be represented by d elements from $\mathbb{F}_q$, and the hash value by c elements from the same field, and let row vector $\underline{m_i} \in \mathbb{F}_q^{(c+d)}$ represent the concatenation of the data and corresponding hash value for packet i. We denote by M the matrix whose i^{th} row is $\underline{m_i}$.

A genuine, or unmodified, packet contains a random linear combination of one or more of these vectors, along with the coefficients of the combination. This information, for a set $\mathcal{U}$ of unmodified packets, can be represented as the matrix product $C(\mathcal{U})\,[M|I]$, where the coefficient matrix $C(\mathcal{U})$ for the set $\mathcal{U}$ is defined as the $|\mathcal{U}| \times r$ matrix whose i^{th} row is the vector of code coefficients of the i^{th} packet. Decoding for a set $\mathcal{U}$ of r linearly independent packets corresponds to pre-multiplying the associated matrix $C(\mathcal{U})\,[M|I]$ with $C(\mathcal{U})^{-1}$.

Modified packets may contain arbitrary data and hash values. A set of modified packets can be represented in general by $[C_bM + V|C_b]$, where V is an arbitrary $(r - s) \times (c + d)$ matrix. Inconsistent data and hash values, i.e. $V \neq 0$, will cause the decoded packets to differ from the original packets.

Suppose the receiver tries to decode using s unmodified packets and $r - s$ modified packets, where $1 \leq s \leq r - 1$. Let C_a and C_b be the coefficient matrices of the set of unmodified packets and the set of modified packets respectively, and let $C = \left[\frac{C_a}{C_b}\right]$. The receiver's decoding process is equivalent to pre-multiplying the matrix

$$\left[\begin{array}{c|c} C_aM & C_a \\ \hline C_bM + V & C_b \end{array}\right] = \left[\begin{array}{c|c} CM + \left[\frac{0}{V}\right] & C \end{array}\right]$$

with C^{-1}. This gives

$$\left[\begin{array}{c|c} M + C^{-1}\left[\frac{0}{V}\right] & I \end{array}\right]$$

i.e., the receiver decodes to $M + \Delta M$, where

$$\Delta M = C^{-1}\left[\frac{0}{V}\right] \tag{1}$$

gives the disparity between the decoded packets and the original packets.

III. Main results

Consider a Byzantine attacker that supplies modified packets, without knowing the contents of $s \geq 1$ genuine unmodified packets that will be part of a set of r packets used for decoding at a receiver. This is a reasonable assumption given the distributed randomness and path diversity of network coding. The only essential condition is that the attacker does not create its packets knowing the contents of all other packets used for decoding, which makes our results very general: they apply regardless of whether the attacker knows which or how many of its own packets will be used for decoding, and whether there are some unmodified packets whose contents are known to the attacker.

Let ν be the rank of the matrix V, defined in the previous section, that represents the modifications.

[1] Secrecy and authenticity are known to be independent attributes of a cryptographic system [12].

The following result characterizes the family of potential outcomes of decoding from the set of packets– the attacker cannot narrow down the set of possible outcomes beyond this regardless of how it designs its modified packets.

Theorem 1: The attacker cannot determine which of a set of $q^{s\nu}$ potential decoding outcomes the receiver will obtain. In particular, there will be at least s packets such that, for each of these, the attacker knows only that the vector representation of its decoded value will be one of q^{ν} possibilities $\{\underline{m}_i + \sum_{j=1}^{\nu} \gamma_{i,j}\underline{v}_j | \gamma_{i,j} \in \mathbb{F}_q\}$, where $\underline{m}_i$ is the vector representation of the data and hash value of some original packet, and $\underline{v}_j$ is determined by the attacker's modifications. □

The next result provides, for a simple polynomial hash function, an upper bound on the proportion of potential decoding outcomes that can have consistent data and hash values, in terms of $k = \lceil \frac{d}{c} \rceil$, the ceiling of the ratio of the number of data symbols to hash symbols. Larger values for k correspond to lower overheads but higher probability of a successful attack. This tradeoff is a design parameter for the network.

Theorem 2: Suppose each packet contains d data symbols $x_1, \ldots, x_d$ and $c \leq d$ hash symbols $y_1, \ldots, y_c$. Consider the function $h : \mathbb{F}_q^k \rightarrow \mathbb{F}_q$ mapping $(x_1, \ldots, x_k)$, $x_i \in \mathbb{F}_q$, to $h(x_1, \ldots, x_k) = x_1^2 + \ldots + x_k^{k+1}$. If y_i is set to $h(x_{(i-1)k+1}, \ldots, x_{ik})$ for $i = 1, \ldots, c-1$ and y_c to $h(x_{(c-1)k+1}, \ldots, x_d)$, then the decoded packets can have consistent data and hash values under at most a fraction $\left(\frac{k+1}{q}\right)^s$ of potential values of the unmodified packets, or, from an alternate viewpoint, at most a fraction $\left(\frac{k+1}{q}\right)^s$ of potential outcomes can have consistent data and hash values. □

Corollary 1: If the receiver obtains more than r packets, it can use all the packets by decoding from more than one set. If $s' \geq 1$ of the packets are unmodified and have contents that are unknown to the attacker, then the decoded packets can have consistent data and hash values under at most a fraction $\left(\frac{k+1}{q}\right)^{s'}$ of potential values of the unmodified packets (at most a fraction $\left(\frac{k+1}{q}\right)^{s'}$ of potential outcomes can have consistent data and hash values).

IV. Detailed development, proofs and ancillary results

A. Vulnerable scenario

Before proving the results stated in the previous section, we first point out that this approach does not apply in the case where the attacker knows, or has information allowing it to predict with reasonable probability, that it is the only node supplying information to a receiver on a particular subset of the original packets. In such a case, this kind of non-cryptographic scheme cannot prevent the attacker from supplying spurious packets with consistent data and hash values. However, such a scenario is unlikely to persist if sources are reasonably well connected, and nodes periodically and randomly switch their connections among neighboring nodes.

Mathematically, this case corresponds to the attacker knowing that a particular set of columns of any potential matrix C_a for the receiver will be zero. Without loss of generality, assume that the last $t \leq r - s$ columns of C_a are zero. The attacker can then make C a block diagonal matrix by choosing C_b to be of the form $\left[\begin{array}{c|c} C_b' & 0 \\ \hline 0 & C_b'' \end{array}\right]$, where C_b'' is a $t \times t$ matrix and the rows of $\left[\begin{array}{c|c} C_b' & 0 \end{array}\right]$ are independent of the rows of C_a. Then C^{-1} is also block diagonal, of the form

$$\left[\begin{array}{c|c} \left[\frac{C_a'}{C_b'}\right]^{-1} & 0 \\ \hline 0 & C_b''^{-1} \end{array}\right]$$

Since C_b'' is determined by the attacker, it can choose ΔM by setting $V = \left[\frac{0}{V'}\right]$, where V' is an appropriately chosen $t \times (c+d)$ matrix.

B. Protected scenario

We next consider the case where the attacker does not know the contents of other packets the receiver will use for decoding. In this case, it designs its packets, i.e. fixes C_b and V, knowing only that $\left[\frac{C_a}{C_b}\right]$ is nonsingular.

Proof of Theorem 1: Consider any fixed C_b and V. A receiver decodes only when it has a set of packets such that corresponding coefficients of the matrix C is non-singular. Therefore, we consider the set $\mathcal{A}$ consisting of the values of C_a that satisfy the condition that $C = \left[\frac{C_a}{C_b}\right]$ is nonsingular.

We show that we can partition the set $\mathcal{A}$ into cosets

$$\mathcal{A}_i = \{C_i + RC_b | R \in \mathbb{F}_q^{s \times (r-s)}\}, i = 1, 2, \ldots, \chi$$

where

$$\begin{aligned} \chi &= \frac{|\mathcal{A}|}{q^{s(r-s)}} \\ &= \frac{\prod_{k=0}^{s-1}(q^r - q^{r-s+k})}{q^{s(r-s)}} \\ &= q^{s(s-1)/2}\prod_{k=1}^{s}(q^k - 1) \end{aligned}$$

Next, we show each coset can be further partitioned into equal-sized sets that each generate, via (1), the full set of possible modifications ΔM. Hence, it suffices to focus on just one of these subsets of $\mathcal{A}$ in proving Theorem 1.

To see that we can partition $\mathcal{A}$ into cosets, consider the following procedure for constructing such cosets: Any element of $\mathcal{A}$ can be chosen as C_1. Next, $C_2, C_3, \ldots, C_\chi$ are chosen sequentially to be any element of $\mathcal{A}$ not in the cosets $\mathcal{A}_j$ of previously chosen elements. Note that this forms a partition of $\mathcal{A}$, since the presence of some element c in two sets $\mathcal{A}_i$ and $\mathcal{A}_j$ implies that C_j is also in $\mathcal{A}_i$, which is a contradiction. It is also clear that each coset has size

$$\left|\{R|R \in \mathbb{F}_q^{s\times(r-s)}\}\right| = q^{s(r-s)}$$

since C_b has full row rank.

For each such coset $\mathcal{A}_{\hat{i}}$, the corresponding values of ΔM satisfy, from (1),

$$\begin{aligned}\left(\left[\frac{C_{\hat{i}}}{C_b}\right]+\left[\frac{RC_b}{0}\right]\right)\Delta M &= \left[\frac{0}{V}\right]\\ \left[\frac{C_{\hat{i}}}{C_b}\right]\Delta M &= \left[\frac{-R}{I}\right]V\\ \Delta M &= \left[\frac{C_{\hat{i}}}{C_b}\right]^{-1}\left[\frac{-R}{I}\right]V\end{aligned}$$

where each entry $r_{i,j} = R(i,j)$ of $R \in \mathbb{F}_q^{s\times(r-s)}$ is, to the attacker, an unknown variable that can take potentially any value in $\mathbb{F}_q$.

We note that even within one of these cosets $\mathcal{A}_{\hat{i}}$, multiple values of R will map to the same value of ΔM if V has dependent rows. If so, we further partition each coset into subsets such that the elements in each subset are in one-to-one correspondence with the full set of possible values for ΔM.

Consider a set of ν independent rows of V (where ν is the rank of V, defined previously). Denote by $\mathcal{I}$ the corresponding set of row indexes, and denote by $V_\mathcal{I}$ the submatrix consisting of those rows. Each row $\underline{v}_j$ of V can be represented as a linear combination $\underline{v}_j = \sum_{k=1}^{\nu} l_{j,k}\underline{v}_k$ of rows of $V_\mathcal{I}$. The coefficients $l_{j,k}$ can be collected into an $(r-s)\times\nu$ matrix L whose $(j,k)^{th}$ entry is $l_{j,k}$, which satisfies

$$V = LV_\mathcal{I}$$

We define $R_\mathcal{I} = RL$, noting that

$$R_\mathcal{I}V_\mathcal{I} = RLV_\mathcal{I} = RV$$

Note also that the submatrix of L consisting of its ν rows corresponding to set $\mathcal{I}$ is an identity matrix. Thus, each variable $r_{i,j}, j \in \mathcal{I}$, appears in exactly one entry of $R_\mathcal{I}$ as part of a linear combination with one or more variables $r_{i,j}, j \notin \mathcal{I}$. It follows that $R_\mathcal{I}$ can take potentially any value in $\mathbb{F}_q^{s\times\nu}$, and every row of $R_\mathcal{I}V_\mathcal{I}$ can take on any value in the row space of V. Furthermore, the possible values of R can be partitioned into equal-sized sets, each of which contains all values $\tilde{R} \in \mathbb{F}_q^{s\times(r-s)}$ such that $\tilde{R}L$ equals some particular value $\tilde{R_\mathcal{I}}$. The $q^{s\nu}$ possible values for $R_\mathcal{I}$ give rise to $q^{s\nu}$ distinct values for $R_\mathcal{I}V$, which give in turn $q^{s\nu}$ distinct values for ΔM.

We note that the set of values

$$\begin{aligned}&\left\{\left[\frac{C_{\hat{i}}}{C_b}\right]^{-1}\left[\frac{-R}{I}\right]V \;\middle|\; R \in \mathbb{F}_q^{s\times(r-s)}\right\}\\ &=\left\{\left[\frac{C_{\hat{i}}}{C_b}\right]^{-1}\left[\frac{-R_\mathcal{I}}{L}\right]V_\mathcal{I} \;\middle|\; R_\mathcal{I} \in \mathbb{F}_q^{s\times\nu}\right\}\end{aligned}$$

corresponding to any single coset $\mathcal{A}_{\hat{i}}$ is in one-to-one correspondence with that of any other coset. To see this, observe that for any fixed $R_\mathcal{I} \in \mathbb{F}_q^{s\times\nu}$ and fixed distinct C_a, C'_a, we obtain the same values for $\Delta M = \left[\frac{C_a}{C_b}\right]^{-1}\left[\frac{-R_\mathcal{I}}{L}\right]V_\mathcal{I}$ and $\Delta M' = \left[\frac{C'_a}{C_b}\right]^{-1}\left[\frac{-R'_\mathcal{I}}{L}\right]V_\mathcal{I}$ by setting

$$R'_\mathcal{I} = -C'_a\left[\frac{C_a}{C_b}\right]^{-1}\left[\frac{-R_\mathcal{I}}{L}\right]$$

which gives:

$$\begin{aligned}\Delta M' &= \left[\frac{C'_a}{C_b}\right]^{-1}\left[\frac{C'_a\left[\frac{C_a}{C_b}\right]^{-1}\left[\frac{-R_\mathcal{I}}{L}\right]V_\mathcal{I}}{LV_\mathcal{I}}\right]\\ &= \left[\frac{C'_a}{C_b}\right]^{-1}\left[\frac{C'_a\Delta M}{V}\right]\\ &= \left[\frac{C'_a}{C_b}\right]^{-1}\left[\frac{C'_a}{C_b}\right]\Delta M\\ &= \Delta M\end{aligned}$$

These observations allow us to focus on a single set

$$\left\{\left[\frac{C_{\hat{i}}}{C_b}\right]^{-1}\left[\frac{-R_\mathcal{I}}{L}\right]V_\mathcal{I} \;\middle|\; R_\mathcal{I} \in \mathbb{F}_q^{s\times\nu}\right\}$$

corresponding to any coset $\mathcal{A}_{\hat{i}}$.

Let $\left[\frac{C_{\hat{i}}}{C_b}\right]^{-1}\left[\frac{-R_\mathcal{I}}{L}\right]$ be denoted by S. Each row of S is an linear function of one or more rows of $R_\mathcal{I}$, either constant, or else dependent on $R_\mathcal{I}$ and taking potentially any value in $\mathbb{F}_q^\nu$. Since $\left[\frac{C_{\hat{i}}}{C_b}\right]^{-1}$ is nonsingular, at least s

rows of S are dependent on $R_{\mathcal{I}}$. The corresponding rows of $SV_{\mathcal{I}}$ are also dependent on $R_{\mathcal{I}}$; for the i^{th} of these rows, the potential values form a set $\{\sum_{j=1}^{\nu}\gamma_{i,j}\underline{v}_j|\gamma_{i,j}\in\mathbb{F}_q\}$, where vector $\underline{v}_j$ corresponds to the j^{th} row of $V_{\mathcal{I}}$. The potential values of the corresponding decoded packets then form a set $\{\underline{m}_i+\sum_{j=1}^{\nu}\gamma_{i,j}\underline{v}_j|\gamma_{i,j}\in\mathbb{F}_q\}$, where $\underline{m}_i$ is the vector representation of the data and hash value of the i^{th} packet. ■

The following lemma is useful in the proof of Theorem 2.

Lemma 1: Consider the following hash function $h:\mathbb{F}_q^k\rightarrow\mathbb{F}_q$ mapping $(x_1,\ldots,x_k)$, $x_i\in\mathbb{F}_q$, to $h(x_1,\ldots,x_k)=x_1^2+\ldots+x_k^{k+1}$, and denote by $\mathcal{Q}(\underline{u},\underline{v})$ the set of vectors $\{\underline{u}+\gamma\underline{v}|\gamma\in\mathbb{F}_q\}$, where $\underline{u}$ and $\underline{v}$ are fixed vectors. At most $k+1$ out of the q vectors in a set $\mathcal{Q}(\underline{u},\underline{v})$, where $\underline{u}=(u_1,\ldots,u_{k+1})$ is a fixed length-$(k+1)$ vector and $\underline{v}=(v_1,\ldots,v_{k+1})$ a fixed nonzero length-$(k+1)$ vector, can satisfy the property that the last element of the vector equals the hash of the first k elements.

Proof: Suppose some vector $\underline{u}+\gamma\underline{v}$ satisfies this property, i.e.

$$u_{k+1}+\gamma v_{k+1}=(u_1+\gamma v_1)^2+\ldots+(u_k+\gamma v_k)^{k+1} \quad (2)$$

Note that for any fixed value of $\underline{u}$ and any fixed nonzero value of $\underline{v}$, (2) is a polynomial equation in γ of degree equal to $1+\tilde{k}$, where $\tilde{k}\in[1,k]$ is the highest index for which the corresponding $v_{k'}$ is nonzero, i.e. $v_{\tilde{k}}\neq 0, v_{k'}=0\ \forall\ k'>\tilde{k}$. By the fundamental theorem of algebra, this equation can have at most $1+\tilde{k}\leq 1+k$ roots. Thus, the property can be satisfied for at most $1+k$ values of γ. ■

Proof of Theorem 2: Each hash symbol is used to protect k data symbols. We consider the set of possible outcomes when a modification is made to at least one symbol of a set consisting of k data symbols and their corresponding hash symbol.

Continuing from the proof of the Theorem 1, we note that S contains s rows that are independent linear combinations of rows of $R_{\mathcal{I}}$. For any particular values of a subset of these rows, each of the remaining rows can take potentially any value in $\mathbb{F}_q^{\nu}$. We consider each of the corresponding rows of $SV_{\mathcal{I}}$ in turn, noting that the set of potential values for the i^{th} of these rows, for any particular values of previously considered rows, is of the form $\{\sum_{j=1}^{\nu}\gamma'_{i,j}\underline{v}_j|\gamma'_{i,j}\in\mathbb{F}_q\}$, and that the set of potential values of the corresponding decoded packets is of the form $\{\underline{m}_i+\sum_{j=1}^{\nu}\gamma'_{i,j}\underline{v}_j|\gamma'_{i,j}\in\mathbb{F}_q\}$.

If $\nu>1$, the q^{ν}-element set $\{\underline{m}_i+\sum_{j=1}^{\nu}\gamma_{i,j}\underline{v}_{i,j}|\gamma_{i,j}\in\mathbb{F}_q\}$ can be partitioned into $q^{\nu-1}$ size-q sets $\{\underline{m}_i+\sum_{j=1}^{\nu-1}\gamma_{i,j}\underline{v}_{i,j}+\gamma_{i,\nu}\underline{v}_{i,\nu}|\gamma_{i,\nu}\in\mathbb{F}_q\}$, where each set corresponds to a different set of values for $\gamma_{i,1},\ldots,\gamma_{i,\nu-1}$.

Applying Lemma 1 to each set $\mathcal{Q}(\underline{m}_i+\sum_{j=1}^{\nu-1}\gamma_{i,j}\underline{v}_{i,j},\underline{v}_{i,\nu})$ gives the desired result. Note that the case where $\underline{v}=\underline{0}$ corresponds to the trivial case where no Byzantine modifications are introduced. ■

Proof of Corollary 1: Suppose more than one different sets of packets are used for decoding. Consider the sets in turn, denoting by s_i the number of unmodified packets in the i^{th} set that are not in any set $j<i$. For any particular values of packets in sets $j<i$, we have from Theorem 2 that at most a fraction $\left(\frac{k+1}{q}\right)^{s_i}$ of decoding outcomes for set i have consistent data and hash values. Thus, the overall fraction of consistent decoding outcomes is at most $\left(\frac{k+1}{q}\right)^{\sum_i s_i}=\left(\frac{k+1}{q}\right)^{s'}$.

V. Conclusion

This paper has described a new, low overhead approach for detecting Byzantine modifications in multi-source multicast networks using distributed randomized network coding. Byzantine modification detection capability is added very inexpensively by augmenting each packet with a number of hash bits that form a small fraction of the total number of bits; this overhead can be traded off against the detection probability. We use a simple polynomial function of the data bits, so very little computational overhead is added. The effectiveness of our approach depends only on the inability of a Byzantine attacker to insert modified packets designed using knowledge of all other packets received by other nodes, and not on the complexity of the hash. This makes our approach quite different from most existing multicast Byzantine detection schemes. This approach can be used in conjunction with a variety of more expensive schemes that are activated only upon detection of a Byzantine node.

References

[1] T. Ho, R. Koetter, M. Médard, D. R. Karger, and M. Effros, "The benefi ts of coding over routing in a randomized setting," in *Proceedings of 2003 IEEE International Symposium on Information Theory*, June 2003.

[2] T. Ho, M. Médard, J. Shi, M. Effros, and D. R. Karger, "On randomized network coding," in *Proceedings of 41st Annual Allerton Conference on Communication, Control, and Computing*, October 2003.

[3] P. A. Chou, Y. Wu, and K. Jain, "Practical network coding," in *Proceedings of 41st Annual Allerton Conference on Communication, Control, and Computing*, October 2003.

[4] L. Lamport, R. Shostak, and M. Pease, "The byzantine generals problem," *ACM Transactions on Programming Languages and Systems*, vol. 4 (3), pp. 382–401, July 1982.

[5] J. Garay and Y. Moses, "Fully polynomial byzantine agreement for n 3t processors in t+1 rounds," *SIAM Journal of Computing*, vol. 27(1), pp. 247–290, February 1998.

[6] M. Castro and B. Liskov, "Practical byzantine fault tolerance," in *OSDI: Symposium on Operating Systems Design and Implementation*. 1999, USENIX Association, Co-sponsored by IEEE TCOS and ACM SIGOPS.

[7] D. Malkhi and M. Reiter, "A high-throughput secure reliable multicast protocol," *Journal of Computer Security*, vol. 5, pp. 113–127, 1997.

[8] K. P. Kihlstrom, L. E. Moser, and P. M. Melliar-Smith, "The SecureRing protocols for securing group communication," in *Proceedings of the 31st Annual Hawaii International Conference on System Sciences (HICSS)*. 1998, vol. 3, pp. 317–326, IEEE Computer Society Press.

[9] R. Perlman, *Network Layer Protocols with Byzantine Robustness*, Ph.D. thesis, Massachusetts Institute of Technology, October 1988.

[10] P. Papadimitratos and Z.J. Haas, "Secure routing for mobile ad hoc networks," January 2002.

[11] Y. Desmedt and Y. Wang, "Perfectly secure message transmission revisited," in *Theory and Application of Cryptographic Techniques*, 2002, pp. 502–517.

[12] J. L. Massey, "Contemporary cryptography: An introduction," *Contemporary Cryptology: The Science of Information Integrity*, pp. 1–39, 1991.

[13] N. Cai and R. W. Yeung, "Secure network coding," in *Proceedings. 2002 IEEE International Symposium on Information Theory*, 2002, p. 323.