Error Estimating Codes with Constant Overhead: A Random Walk Approach

Hongyi Yao and Tracey Ho
California Institute of Technology
{hyao,tho}@Caltech.edu

Abstract—**The paper studies the construction of error-estimating codes (EEC), which estimate the bit-error-rate (BER) of packet transmissions. The concept of EEC was first proposed by Chen et.al [1], who provided a construction based on group sampling, which we term group-sampling error-estimating codes (GSEEC). In this paper, *random walk* based error-estimating codes (RAKEE) are proposed, which achieves *constant* communication overhead and *linear* coding complexity respect to packet length. Compared with GSEEC, better error decaying performance is proved for ALEEC. Numerical experiments show that RAKEE improves GSEEC on both estimating bias and estimating mean square error.**

I. Introduction

Error-correcting codes (ECCs) e.g. [2]–[4] allow packets to be transmitted without error over a noisy channel, and have traditionally played a major role in communications and information theory. Motivated by recent advances in wireless networking that leverage partially correct packets, e.g. in scenarios such as rate-adaption [5]–[7] and real-time video streaming [8], Chen et.al [1] proposed the concept of error-estimating codes (EEC). Instead of correcting bit errors as in ECC, EEC allows the receiver to estimate the bit-error-rate (BER) with lower overhead compared to ECC. Using this BER information, the authors in [1] showed that the performance of upper-layer applications can be significantly improved. Furthermore, an error estimation code based on group sampling, which we term group-sampling error-estimating codes (GSEEC), was proposed in [1]. In the same work, the authors showed that GSEEC achieves significant less communication overheads and computational complexities compared with existing ECCs.

Based on the idea of random walk, in this paper we propose a novel error estimation code, RAKEE. Theoretical analysis proves that RAKEE achieves the same asymptotic (respect to packet length) performance as GSEEC, that is *constant* communication overhead and *linear* coding complexity. Moreover, compared with GSEEC better error decaying performance is proved for ALEEC. To be precise, under GSEEC the probability of unreliable estimation is proved to decay polynomially when the communication overheads increase, while under ALEEC such probability decays super-polynomially. Numerical experiments show that RAKEE improves GSEEC on both estimating bias and estimating mean square error.

A. Organization of the paper

The rest of this paper is organized as follows. Section II formulates the problem. RAKEE is constructed in Section III. In Section III-A, the main theorem of the paper is shown. Experimental results are shown in Section IV to support the theoretical analysis of RAKEE. In the end, we summarize the paper in Section V.

II. Error Estimation Framework

Our model is based on the framework for EEC introduced in [1]. The number of data bits in a packet is denoted by n. The EEC encoding process adds k bits which constitute the overhead needed for error estimation. The length-$(n+k)$ packet consisting of the data and overhead bits is transmitted over a channel that can cause some of the bits to be erroneous (i.e. flipped). The goal of EEC is to estimate the fraction of erroneous bits, i.e. the BER of the packet, denoted ρ. There can be arbitrary and unknown correlations in the error positions; this is dealt with using randomization in the EEC construction.

To be more precise, let X be the length-n binary data vector in the packet transmitted by the sender, called Alice, and let X' be the length-n vector representing the corresponding bits received by the sink, called Bob. For any binary string, say X, let $X(i)$ be the i'th bit of X. Let $H(X, X')$ be the hamming distance between X and X', *i.e.*, $H(X, X') = |\{i : X(i) \neq X'(i), i = 1, 2..., n\}|$. The BER of X is $\rho = H(X, X')/n$.

For any constants $0 < \epsilon < 1$ and $0 < \delta < 1$, the estimate $\hat{\rho}$ output by the EEC is said to be (ϵ, δ) reliable if and only if the probability that $|\hat{\rho} - \rho| < \epsilon\rho$ is at least $1 - \delta$, where the probability is over the randomness in the EEC algorithm.

III. Random Walk based Error-Estimating Codes

A. Main Result

We propose random walk based error-estimating codes (RAKEE), which achieve (ϵ, δ) estimation reliability with overhead constant in the packet length. Our main result is as follows:

Theorem 1: For any constants $0 < \epsilon < 1$, $0 < \delta < 1$ and $0 < a < b < 1/2$,[1] we can construct a RAKEE code with

[0] adas

[1] Note that we consider the communication environment where BER is independent of the packet length. Thus, parameters a and b are chosen as constants respect to n. In [1], the BER can depend on n and therefore a can be as small as $1/n$. When a is a constant independent of n, the asymptotic communication overhead of GSEEC is also a constant respect to n. We note that RAKEE also works when $a = 1/n$, which induces asymptotic communication overhead $O(\log(n))$.

communication overhead $O(1)$ and coding complexity $O(n)$ such that with a probability at least $1-\delta$,

- If $a < \rho < b$, $|\hat{\rho} - \rho| \leq \epsilon\rho$.
- If $\rho \leq a$, $\hat{\rho} \leq (1+\epsilon)a$.

Section III-E gives a detailed proof of this result. In the following we describe the mathematical principles behind the construction as well as the encoding and decoding procedures. For the desired (ϵ, δ, a, b), Table I gives the exact values of the parameters in RAKEE.

Note that the constants of the asymptotic communication overhead and computational complexity depend on (ϵ, δ, a, b). To be precise, the communication overhead of RAKEE is $m\beta/(1-H(b))$ and the computational complexity is dominated by

$$mn + C(m\beta)),$$

where m and β are defined in Table I, $H(b) = -b\log_2(b) - (1-b)\log_2(1-b)$ is the achievable rate of binary symmetric channel with bit flipping probability less than b, and $C(m\beta)$ is the coding complexity of transmitting $m\beta$ bits by using error correction code on such binary symmetric channel. In particular, when ϵ and δ approach zero, the communication overhead increases with order term $\Theta(\log\log(1/(\epsilon\delta))\log(1/\delta)/\epsilon^2)$. Thus, the probability of unreliable estimation δ decays super-polynomially when communication overhead increases. Under GSEEC [1], such probability is proved to decay polynomially.

TABLE I
Parameters for RAKEE

Parameters	Value
m	$48\ln(8/\delta)/\epsilon^2$
Δ	$\epsilon a/8$
β	$\log_2(2+\ln(1/(\epsilon\delta))) + \log_2(2+(8/(\epsilon a))) + 5$

B. Overview of the Main Techniques

Let R be a length-n random vector whose components are i.i.d. random variables over $\{-1, 1\}$ with uniform distribution. Let R_1, R_2..., and R_m be independent samples of R known to both Alice and Bob, where the parameter m only depends on the reliability parameters ϵ, δ, a and not on n, and its value is given in the Table I. Note that in practice, the shared randomness $\{R_1, R_2, ..., R_m\}$ can be constructed by using a pseudo-random number generator with a common random seed known *a priori* to both Alice and Bob.

The main mathematical elements of RAKEE are as follows:

- Step 1: As shown in Figure 1, Alice first computes m hash values of X; specifically, for each $i = 1, 2, ..., m$, Alice computes the inner product $< X, R_i >$ between X and R_i over the *real field*.
- Step 2: Besides the message vector X, Alice also sends a *compressed* version of the above hash values protected by a standard error-correcting code, using constant overhead.
- Step 3: Using $\{R_1, R_2, ..., R_m\}$, Bob similarly computes the inner products $< X', R_i >$ for the received bit vector X'. As we show in the proof of Theorem 1, the expectation $E(< X, R_i > - < X', R_i >)^2$ equals the number of positions at which X differs from X', and the average of $(< X, R_i > - < X', R_i >)^2$ over $i = 1, 2, ..., m$ is concentrated around its expectation. Thus, the computed inner products and the hash information received from Alice can be used to estimate BER ρ.

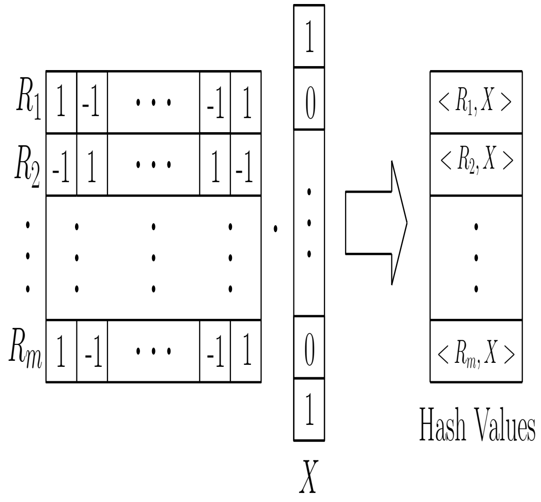

Fig. 1. Hash computations for X.

Note that the inner product $< X, R_i >$ may be as large as n, which would require $\log(n)$ bits to encode. To achieve constant overhead that does not increase with n, the hash values are compressed in Step 2 based on the following two ideas:

1) The inner product $< X, R_i >$ is in fact a symmetric random walk of $|X|$ steps, where $|X| \leq n$ is the number of 1s in X. Thus, with high probability $< X, R_i > / \sqrt{n}$ is bounded by a constant.
2) Alice quantizes $< X, R_i > / \sqrt{n}$ such that it can be represented by a constant number of bits. As we show in the proof of Theorem 1, it turns out that a constant number of bits suffices for constant ϵ, δ, a. Since m is also constant with respect to n, the overall overhead is constant.

C. Alice's Encoder

For each $i = \{1, 2, ..., m\}$, Alice computes the compressed hashes as follows:

1) Alice computes $I_i =< X, R_i > / \sqrt{n}$ over the real field.
2) Alice *quantizes* I_i and represents it by a length-β binary vector S_i. The quantization resolution Δ and the vector length β are constants that depend only on the reliability parameters ϵ, δ, a, and their values are given in the Table I. Specifically, as illustrated in Figure 2, Alice performs
   - **Quantization**. Let H_i denote the (signed) index of the point in the set $\{... - 2\Delta, -\Delta, 0, \Delta, 2\Delta, ..., \}$ that is closest to I_i.
   - **Binary Encoding**. The first $\beta - 1$ bits of S_i are the binary expression of $min(|H_i|, 2^{\beta-1} - 1)$, while the last bit of S_i indicates the sign of H_i.

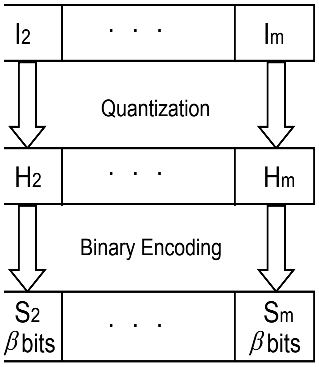

Fig. 2. Compressing the hash values $\{I_i : i = 1, 2, ..., m\}$.

Note that the hash information $\mathcal{S} = \{S_i : i = 1, 2, ..., m\}$ consists of $m\beta$ bits, which is a constant with respect to n. Thus, $\mathcal{S}$ can be transmitted with constant overhead and complexity using a standard error-correcting code e.g. [2], [3] whose rate is chosen based on a given upper bound $b < 1/2$ on the BER. As in GSEEC [1], in order to handle location-dependent error distributions, the hash information is randomly located within the packet using randomness common to Alice and Bob.

D. Bob's Estimator

After receiving the packet, Bob estimates ρ as follows:

1) Bob extracts the bits that are used to send $\mathcal{S}$ and decodes the error correcting code to recover $\mathcal{S} = \{S_i : i = 1, 2, ..., m\}$.
2) For each $i \in \{1, 2, ..., m\}$, Bob computes $\hat{H}_i$ such that $|\hat{H}_i|$ corresponds to the binary-expression of string $S_i(1)S_i(2)...S_i(\beta-1)$, and $\hat{H}_i$ is negative if $S_i(\beta) = 1$ and positive otherwise.
3) For each i, Bob estimates I_i by $\hat{I}_i = \hat{H}_i\Delta$.
4) Bob computes $I'_i =< X', R_i > / \sqrt{n}$ over the real field.
5) Bob estimates the bit-error-rate ρ as $\hat{\rho} = \sum_{i=1}^{m}(\hat{I}_i - I'_i)^2/m$.

E. Proof of Theorem 1

For any constants $0 < \epsilon < 1$, $0 < \delta < 1$ and $0 < a < b < 1/2$ in Theorem 1, we choose the parameters following Table I.

For each $i \in \{1, 2, ..., m\}$, let

$$\Delta_i = \hat{I}_i - I_i$$

be the estimation error of I_i, which is due to quantization or transmission noise. Thus,

$$\begin{aligned}\hat{\rho} &= \sum_{i=1}^{m}(\hat{I}_i - I'_i)^2/m \\ &= \sum_{i=1}^{m}(I_i - I'_i + \Delta_i)^2/m \\ &= \sum_{i=1}^{m}(I_i - I'_i)^2/m + \sum_{i=1}^{m}[2(I_i - I'_i)\Delta_i + \Delta_i^2]/m \\ &= \sum_{i=1}^{m}(I_i - I'_i)^2/m + Err,\end{aligned}$$

where $Err = \sum_{i=1}^{m}[2(I_i - I'_i)\Delta_i + \Delta_i^2]/m$ denotes the noise term.

Using $m = 48\ln(8/\delta)/\epsilon^2$, we have:

Lemma 2: With a probability at least $1 - \delta/4$,

$$|\sum_{i=1}^{m}(I_i - I'_i)^2/m - \rho| \le \rho\epsilon/2.$$

Proof of Lemma 2: Since $(I_i - I'_i) =< X - X', R_i > / \sqrt{n}$, we have

$$\begin{aligned}E((I_i - I'_i)^2) &= E((\sum_{j=1}^{n}(X(j) - X'(j))R_i(j))^2)/n \\ &= \sum_{j=1}^{n}(X(j) - X'(j))^2 E(R_i(j)^2)/n \\ &+ \sum_{j\neq k}(X(j) - X'(j))(X(k) - X'(k))E(R_i(j))E(R_i(k))/n \\ &= \sum_{j=1}^{n}(X(j) - X'(j))^2 E(R_i(j)^2)/n \\ &= \rho\end{aligned}$$

and

$$\begin{aligned}E((I_i - I'_i)^4 n^2) &= E((\sum_{j=1}^{n}(X(j) - X'(j))R_i(j))^4) \\ &= E(\sum_{j=1}^{n}(X(j) - X'(j))^2 R_i(j)^2 \\ &+ 2\sum_{j\neq k}(X(j) - X'(j))(X(k) - X'(k))R_i(j)R_i(k))^2.\end{aligned}$$

Note $R_i(j)$ is independently chosen for each $i = 1, 2..., n$. Thus, after expanding above equation, any monomial term with an odd power (for $\{R_i(j) : j = 1, 2, ...n\}$) has expected value zero, and any term with only even powers has expected value 1. There are $n\rho$ terms of the form $R_i(j)^4$ and $6\binom{n\rho}{2}$ terms of the form $R_i(j)^2 R_i(k)^2$. Thus, we have

$$\begin{aligned}E((I_i - I'_i)^4 n^2) &= n\rho + 6\binom{n\rho}{2} \\ &\le 4(n\rho)^2.\end{aligned}$$

Thus, we have $E((I_i - I'_i)^4 \le 4\rho^2$ and therefore $Var((I_i - I'_i)^2) \le 3\rho^2$.

Note that since R_i is independently chosen for each i, $(I_i - I'_i)^2$ is independent for each i. Let

$$\begin{aligned}D_i &= ((I_i - I'_i)^2 - \rho)/n \\ &= (< X - X', R_i >^2 /n - \rho)/n.\end{aligned}$$

Note that D_i is independent for different i, $E(D_i) = 0$, $Var(D_i) \le 3\rho^2/n^2$ and $|D_i| < 1$. Using Chernoff Bound for discrete random variables [9] we have

$$Pr(|\sum_{i=1}^{m} D_i|/m \le \rho\epsilon/(2n)) \ge 1 - 2e^{-1/48(\epsilon^2 m)}.$$

Since $|\sum_{i=1}^{m}(I_i - I'_i)^2/m - \rho| \le \rho\epsilon/2$ is equivalent to $|\sum_{i=1}^{m} D_i|/m \le \rho\epsilon/(2n)$, we conclude

$$Pr(|\sum_{i=1}^{m}(I_i - I_i')^2/m - \rho| \le \rho\epsilon/2) \ge 1 - 2e^{-1/48(\epsilon^2 m)}.$$

Assuming $m = 48\ln(8/\delta)/\epsilon^2$, we prove the lemma. □

We then analyze the error term Δ_i. Using $\Delta = \epsilon a/8$ and $\beta = \log_2 \ln(1/(\epsilon\delta)) + \log_2(8/(\epsilon a)) + 5$, we have $\beta = \log_2 \ln(1/(\epsilon\delta)) + \log_2(1/\Delta) + 5$ and then

Lemma 3: When $\rho < b$, with a probability at least $1 - \delta/2$ we have

$$|\Delta_i| = |\hat{I}_i - I_i| \le \Delta/2$$

for each for each $i \in \{1, 2, ..., m\}$.

Proof of Lemma 3: For each $i = 1, 2, ..., m$, $|H_i| > 2^{\beta-1} - 1$ only if

$$| < X, R_i > | > \sqrt{n}(2^{\beta-1} - 2)\Delta. \tag{1}$$

Note that $< X, R_i >$ is in fact the summation of $|X|$ (*i.e.*, the number of 1s in X) random Bernoulli variables, each of which has expected value 0 and variance 1.

Using the Chernoff Bound [9], Equation (1) holds with a probability no more than

$$2e^{-(2^{\beta-1}-2)^2\Delta^2/4}$$

. Using Union Bound [10], with a probability at least $1 - 2me^{-(2^{\beta-1}-2)^2\Delta^2/4}$, we have

$$| < X, R_i > | \le \sqrt{n}(2^{\beta-1} - 2)\Delta$$

and therefore

$$|H_i| \le 2^{\beta-1} - 1$$

for each $i = 1, 2, ..., m$. Since $\beta = \log_2(2 + \ln(1/(\epsilon\delta))) + \log_2(2 + 1/\Delta) + 5$, we have

$$2me^{-(2^{\beta-1}-2)^2\Delta^2/4} \le \delta/4.$$

Thus, with a probability at least $1-\delta/4$, we have $|H_i| \le 2^{\beta-1}-1$ for each $i = 1, 2, ..., m$, and therefore $\mathcal{S} = \{S_i : i = 1, 2, ..., m\}$ correctly encode $\{H_i : i = 1, 2, ..., m\}$.

An error-correction-code e.g. [2], [3] is used for transmitting the $m\beta$ bits of $\mathcal{S}$. We can choose the parameters of the error-correction-code such that when the error rate is no more than $b < 1/2$, $\mathcal{S}$ can be correctly transmitted with a probability $1 - \delta/4$. Using Union Bound [10], with a probability at least $1 - \delta/2$, we have $\hat{H}_i = H_i$ for each $i \in \{1, 2, ..., m\}$.

Using $|H_i\Delta - I_i| \le \Delta/2$, with a probability at least $1 - \delta/2$ we have

$$\begin{aligned} |\Delta_i| &= |\hat{I}_i - I_i| = |\hat{H}_i\Delta - I_i| \\ &= |H_i\Delta - I_i| \\ &\le \Delta/2 \end{aligned}$$

for each $i \in \{1, 2, ..., m\}$. □

Using $\Delta = a\epsilon/8$, the noise term Err satisfies:

$$|Err| < 2(\epsilon a/16)\sum_{i=1}^{m}|(I_i - I_i')|/m + \epsilon a/16.$$

Note that $E(|I_i - I_i'|) \le \sqrt{E(|I_i - I_i'|^2)} = \sqrt{\rho} \le 1$ and $Var(|I_i - I_i'|) \le E((|I_i - I_i'|)^2) = \rho$ for each $i = 1, 2, ..., m$. As the proof of Lemma 2, similarly using Chernoff Bound [9] we have $|\sum_{i=1}^{m}(I_i - I_i')/m| \le 3$ and therefore $|Err| < \epsilon a/2$ with a probability at least $1 - 2e^{-m/(4\sqrt{\rho})}$. Using $m = 48\ln(8/\delta)/\epsilon^2$, we have $|Err| < \epsilon a/2$ with a probability at least $1 - \delta/4$.

In the end, using Union Bound [10], with a probability at least $1 - \delta$ all above random events happens. Therefore with a probability at least $1 - \delta$ we have

$$\begin{aligned} |\hat{\rho} - \rho| &\le |\sum_{i=1}^{m}(I_i - I_i')^2/m - \rho| + |Err| \\ &\le \epsilon(\rho + a)/2. \end{aligned}$$

Thus, with a probability at least $1 - \delta$, when $\rho \ge a$ we have $|\hat{\rho} - \rho| \le \epsilon\rho$; Otherwise when $\rho < a$ we have $\rho < a(1 + \epsilon)$.

The communication overhead of RAKEE equals the length of error-correction-codeword for $\mathcal{S}$. Since $\mathcal{S}$ contains $m\beta$ bits, the overhead is independent of n and therefore $O(1)$. The coding complexity of RAKEE is dominated by the complexity of computing the m hash values, which is $mn = O(n)$. This completes the proof.

IV. Experimental simulations

In the section we provide numerical simulations to complement the theoretical analysis of RAKEE. We adopt the experimental setting in [1]. To be concrete, we assume that $n = 2^{14}$ and BER ρ ranges from 0.001 to 0.15.

In the simulations, the packet error is generated in the following manner: For each packet generation, let ρ be i.i.d. chosen from $[0.001, 0.15]$ with uniform distribution. Then , each bit in the packet independently suffers flipping with a probability ρ. Note that for both RAKEE and GSEEC, when the hashes are randomly located in the packet, it is not necessary to assume "independent" bit-flipping errors.

In the following, for different communication overhead ℓ, we examine the performances and complexities of both RAKEE and GSEEC. Let $x = \ell/n$ to be the relative overhead.

A. Estimation Performance

The experimental results in this part shows that for a fixed overhead, RAKEE achieves *smaller* estimation-error and *stabler* performance than those of GSEEC.

In the simulations of RAKEE, the system parameters Δ and β are fixed to be $1/16$ and 6, respectively. Thus, the hashes $\mathcal{S} = \{S_i : i = 1, 2, ..., m\}$ corresponds to $6m$ bits. We assume Reed Solomon (RS) codes [3] are used for error-correction-encoding $\mathcal{S}$. In particular, we assume each RS symbol is over GF(2^5) and therefore encodes 2 bits information. Using maximum likelihood decoding [11], for bit-error-rate 0.15, symbol-error-rate 0.14 is achievable, which corresponds to symbol rate $1 - 2 * 0.14 = 0.72$ for a RS codewords. Thus, for a given overhead ℓ bits, the RS codeword has $\ell/5$ symbols, which is able to encode $0.72 * 2 * \ell/5 = 0.288\ell$ bits. To encode $6m$ bits, m is set to be $m = \ell/21$.

For each packet, say g, let $\epsilon(g) = (\hat{\rho}(g) - \rho(g))$ denote the relative estimation error of BER. In Figure 3, corresponding to different relative overhead $x = \ell/n$, the solid line shows the estimating mean (over 200 packet generations) square error $\epsilon_A = \sum_{g=1}^{200} \epsilon(g)^2/200$ of RAKEE. In Figure 4, the solid line shows the estimating bias $Bias_A = \sum_{g=1}^{200} \epsilon(g)/200$ of the relative errors of RAKEE.

In the simulations of GSEEC, we adopt the parameters shown in Section 4 of [1]. Specially, the improved estimation algorithm (Section A.7 in [1]) is performed, where we assume $c_1 = 0.25$, $c_2 = 0.4$ and 9 levels are used. For each overhead ℓ, the system parameter s for GSEEC is therefore set to be $s = \ell/9$. For GSEEC, the dashed line in Figure 3 shows the estimating mean square error, while the dashed line in Figure 4 shows the estimating bias.

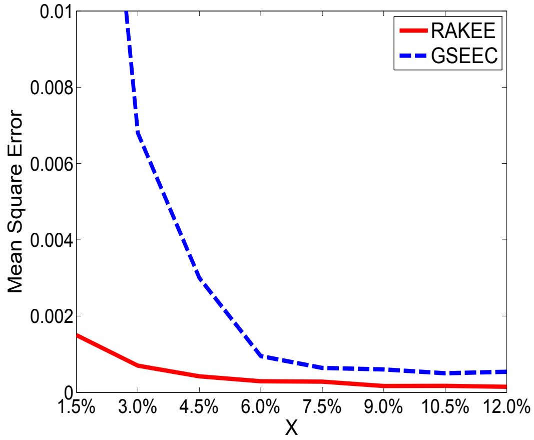

Fig. 3. Estimating mean square error of RAKEE and GSEEC.

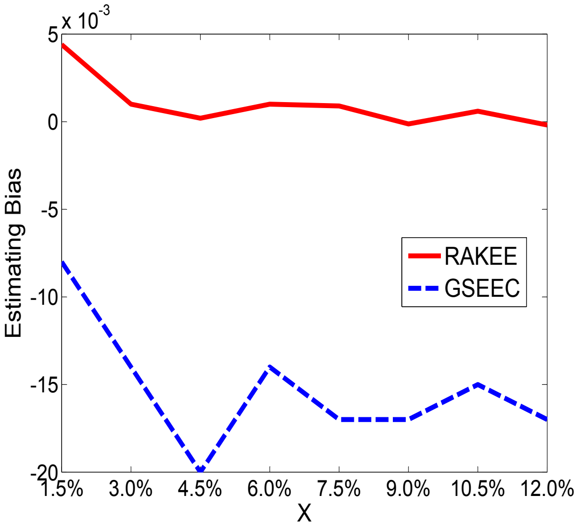

Fig. 4. Estimating bias of RAKEE and GSEEC.

V. Conclusions

Based on the idea of random walk, in this paper we construct error estimation code, RAKEE, to estimate the bit-error-rate (BER) of a transmitted packet. RAKEE is proved to achieve asymptotic (respect to packet length) constant communication overhead and linear computational complexity. Moreover, under ALEEC the probability of unreliable estimation decays super-polynomially when the communication overheads increase, while under GSEEC such probability is proved to decay polynomially. Simulation experiments show that RAKEE improves previous error estimation codes on both estimating mean square error and estimating bias.

VI. Acknowledgment

This work was supported by the Air Force Office of Scientific Research under grant FA9550-10-1-0166 and Caltech's Lee Center for Advanced Networking, National Natural Science Foundation of China Grant 61033001,61073174 and 61061130540, the National Basic Research Program of China Grant 2007CB807900 and 2007CB807901, Hi-Tech research and Development Program of China Grant 2006AA10Z216. Part of Hongyi Yao's work was done when he was in Tsinghua University.

References

[1] B. Chen, Z. Zhou, Y. Zhao, and H. Yu, "Efficient error estimating coding: Feasibility and applications," Conference version was presented in SIGCOMM, technical report is avaible at: http://www.comp.nus.edu.sg/ ~yuhf/eec-tr.pdf, Tech. Rep., 2010.

[2] S. Dolinar, D. Divsalar, and F. Pollara, "Code performance as a function of block size," TMO Progress Report, Tech. Rep., 1998.

[3] U. K. Sorger, "A new reed-solomon code decoding algorithm based on newtons interpolation," *IEEE Trans on Information Theory*, 1993.

[4] E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Trans. on Information Theory*, 2009.

[5] J. C. Bicket, "Bit-rate selection in wireless networks," Master's thesis, MIT, 2005.

[6] G. Holland, N. Vaidya, and P. Bahl, "A rate-adaptive mac protocol for multi-hop wireless networks," in *In Proc. of MOBICOM*, 2001.

[7] S. Wong, H. Yang, S. Lu, and V. Bharghavan, "Robust rate adaptation for 802.11 wireless networks," in *In Proc. of MOBICOM*, 2006.

[8] M. Elaoud and P. Ramanathan, "Adaptive use of error-correcting codes for real-time communication in wireless networks," in *In Proc. of INFOCOM*, 1998.

[9] V. Vu, "Chernoff bound," University of California, San Diego. Avaible at: http://cseweb.ucsd.edu/ klevchen/techniques/chernoff.pdf, Tech. Rep.

[10] M. Mitzenmacher and E. Upfal, *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005.

[11] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Code*. New York: North-Holland Publishing Company, 1977.