Machine Learning Methods for Connection RTT and Loss Rate Estimation Using MPI Measurements Under Random Losses

Nageswara S. V. Rao[1(✉)], Neena Imam[1], Zhengchun Liu[2], Rajkumar Kettimuthu[2], and Ian Foster[2]

[1] Oak Ridge National Laboratory, Oak Ridge, TN, USA
{raons,nimam}@ornl.gov
[2] Argonne National Laboratory, Argonne, IL, USA
{zhengchun.liu,kettimut,foster}@anl.gov

Abstract. Scientific computations are expected to be increasingly distributed across wide-area networks, and Message Passing Interface (MPI) has been shown to scale to support their communications over long distances. Application-level measurements of MPI operations reflect the connection Round-Trip Time (RTT) and loss rate, and machine learning methods have been previously developed to estimate them under deterministic periodic losses. In this paper, we consider more complex, random losses with uniform, Poisson and Gaussian distributions. We study five disparate machine leaning methods, with linear and non-linear, and smooth and non-smooth properties, to estimate RTT and loss rate over 10 Gbps connections with 0–366 ms RTT. The diversity and complexity of these estimators combined with the randomness of losses and TCP's non-linear response together rule out the selection of a single best among them; instead, we fuse them to retain their design diversity. Overall, the results show that accurate estimates can be generated at low loss rates but become inaccurate at loss rates 10% and higher, thereby illustrating both their strengths and limitations.

Keywords: Round Trip Time · Loss rate · Message Passing Interface · Machine Learning · Generalization bounds · Regression · Information fusion

1 Introduction

Computations distributed across geographically dispersed facilities, such as multiple supercomputer sites connected over a wide-area network, are of increasing

This work is funded by RAMSES project and Applied Mathematics program, Office of Advanced Computing Research, U.S. Department of Energy, and by Extreme Scale Systems Center, sponsored by U.S. Department of Defense, and performed at Oak Ridge National Laboratory managed by UT-Battelle, LLC for U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

© IFIP International Federation for Information Processing 2020
Published by Springer Nature Switzerland AG 2020
S. Boumerdassi et al. (Eds.): MLN 2019, LNCS 12081, pp. 154–174, 2020.
https://doi.org/10.1007/978-3-030-45778-5_11

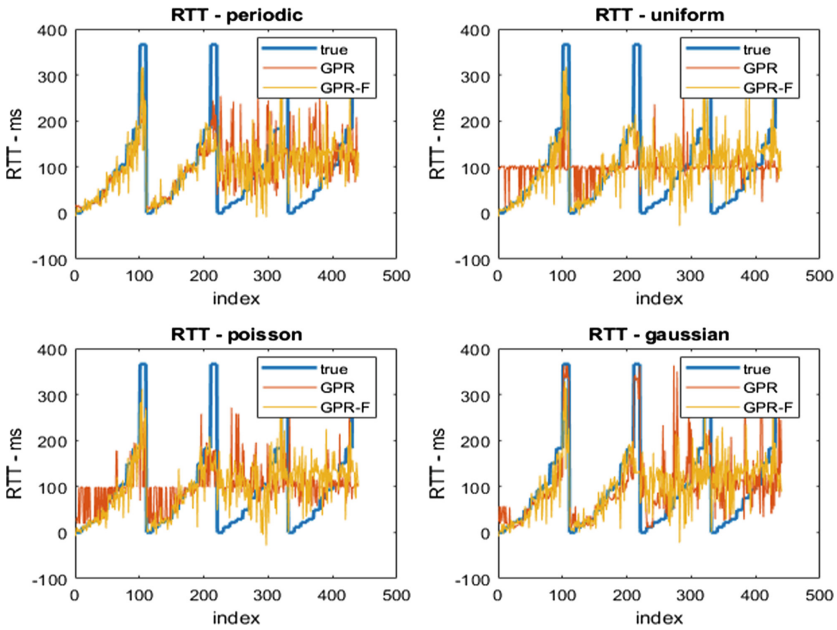

Fig. 1. RTT estimates with lowest RMS error of individual and fuser methods.

interest in science applications. Their execution times are effected by network latencies and loss processes, often in a complex way, due to the close coupling between computations and communications in these applications. Recently, Message Passing Interface (MPI) has been shown to be effective in supporting communications over wide-area connections, including ones long enough to span the globe, under external packet loss rates up to 20% [14]. In contrast with executions at a single facility, these distributed computations need to account for the longer and more varied executions times of MPI operations to avoid inefficiencies due to unbalanced computing and networking operations; for example, MPI join operation over connections with wide ranging latencies will be delayed by the longest. Motivated by such considerations, Round Trip Times (RTT) and loss rates of wide-area connections are estimated using execution time measurements of MPI primitives in distributed computations [15]. A main contributor to these execution times is the Transmission Control Protocol (TCP) which is a dominant underlying transport mechanism of MPI for wide-area connections. In particular, at increased loss rates and randomness, the execution time variations are dominated by TCP's highly non-linear response dynamics [7,9].

Machine Learning (ML) methods have been developed for a number of networking tasks for science data flows, for example, detecting flow anomalies [6] and classifying elephant and mice flows [4]. In particular, ML methods are developed to estimate the connection RTT and loss rate under deterministic periodic losses in [15] for 10 Gbps emulated connections with 0–366 ms RTT; these connections

Table 1. RMS errors of RTT estimation by individual and fuser methods.

Loss type	EOT	GPR	LR	RT	SVM	LR-F	GPR-F	$\tilde{\Delta}_{\text{LR-F}}$	$\tilde{\Delta}_{\text{GPR-F}}$
Periodic	109.20	**87.78**	102.27	95.31	103.49	92.94	**85.50**	−1.63	2.28
Poisson	101.89	**91.90**	104.32	104.31	120.35	89.85	**85.69**	2.05	6.21
Gaussian	84.33	**73.13**	2020.55	73.28	137.29	88.22	**87.06**	−15.06	−13.92
Uniform	109.11	**99.59**	106.47	107.62	121.81	90.12	**84.90**	9.46	14.69

represent local, cross country, continental and round the earth distances. In this paper, we consider more realistic, complex scenarios with random losses, in particular under unform, Poisson and Gaussian distributions up to 20% loss rates, to study the strengths and limitations of ML methods. We study five disparate ML methods, with linear and non-linear, and smooth and non-smooth properties, to estimate connection RTT and loss rate. They include four non-linear estimators, namely, smooth Support Vector Machine (SVM) and Gaussian Process Regression (GPR), and non-smooth Ensemble of Trees (EOT) and Regression Trees (RT), in addition to the baseline Linear Regression (LR) method. The diversity and complexity of these estimators combined with the randomness of losses and TCP's non-linear response rule out the identification of a single best among the estimators. Analytical results establish the finite-sample limits in asserting the performance superiority of any such method based on samples [5]. In particular, the training error is an insufficient indicator of estimator's performance due to potential over-fitting that leads to poor generalization performance on future datasets.

Over-fitting is often specific to an estimator method and is less likely to occur across estimators of radically different designs. In several cases, by fusing diverse estimators both the performance and diversity of design are preserved [10]. However, the fused estimators are also subject to finite sample limits since they are also estimators. We study linear regression fusion (LR-F) and GPR fusion (GPR-F) methods, and our results show that the latter achieves lowest Root Mean Square Error (RMSE) among all estimators for RTT in three out of four scenarios. We develop analytical characterization of the performance improvements of fused estimates over individual RTT estimates under finite sample, distribution-free framework [17].

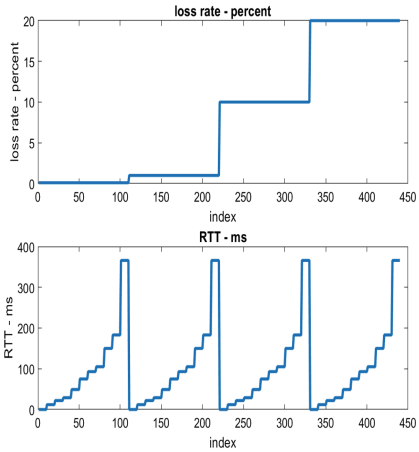

Fig. 2. Index representing increase of loss rate and RTT from left to right.

By using MPI execution times as the independent variable, we formulate the problem of estimating RTT and loss rate as a regression estimation problem. The overall results are illustrated using RTT estimates with smallest RMSE among individual methods and fusers in Fig. 1 for four loss rates. In each plot, datasets are concatenated at four loss rates in increasing order and at each loss rate RTT is increased left to right, and measurements are repeated 10 times at each RTT value as shown in Fig. 2. Among individual RTT estimates, GPR has the lowest RMSE in all four scenarios, and GPR-F fuser achieved even lower RMSE in three out of four loss scenarios, as shown in Table 1 while encompassing the design diversity of individual methods. Overall, our results show that accurate estimates can be generated at low loss rates but become inaccurate at loss rates 10% and higher, wherein the datasets appear much too complex for these methods (as in the case of deterministic periodic losses [15]). In addition, they reveal some subtle performance effects including over-smoothing by some estimators in achieving lower RMSE, and bleeding effects of RTT in loss rate estimates.

The organization of this paper is as follows. The testbed used in collecting MPI execution time measurements is described in Sect. 2. An analytical formulation of the underlying regression problem is presented in Sect. 3. Various datasets of execution time measurements are qualitatively described in Sect. 4. RTT estimators are described in Sect. 5, wherein five different ML methods are described in Sect. 5.1 and two fusers are described in Sect. 5.2. Generalization equations of the fusers for RTT estimation are described in Sect. 6. Loss rate estimators are described in Sect. 7. The performance of the estimators is qualitatively interpreted in the context of datasets at lower and higher loss rates in Sect. 8. A summary of results and directions for future work are described in Sect. 9.

2 Test Configuration

A computing cluster with InifiniBand (IB) interconnect is expanded to constitute a testbed to run MPI codes across the wide-area Ethernet connections. Additional Ethernet Network Interface Cards (NIC) are installed in two cluster computing nodes (tait1 and tait2), which are connected to Ethernet switches and a hardware-based Ethernet emulator in the configuration shown in Fig. 3. The IB connections of the cluster are subject to 2.5 ms latency limit, and hence MPI measurements over IB are not indicative of the performance over long distance connections. Specifically, the shorter distances combined with credit-based IB protocol flow control do not adequately reflect the complex variations of TCP over wide-area connections, particularly under packet losses. Furthermore, due to their latencies, wide-area networks are more prone to more losses compared to IB networks.

Typical wide-area connections consist of a number of switches and routers whereas IB connections have fewer IB switches. This testbed connection consists of two Ethernet switches between the source computing node and a port of the emulator, which reflects a site connection. Similarly at the other end, the connection consists of two Ethernet switches between the second port of the emulator and destination computing node. Thus, this symmetric end-to-end connection consists of four Ethernet internal cross-connections, six short Ethernet segments and one emulated long distance Ethernet connection with variable latency, loss rate and type of loss distribution.

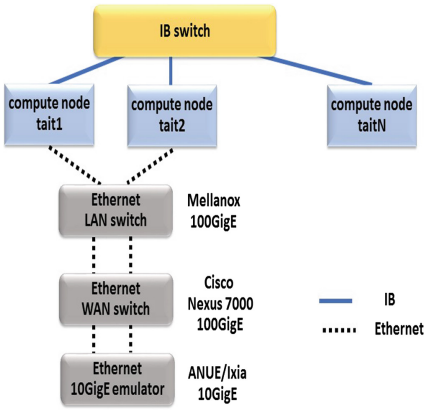

Fig. 3. Configuration for long Ethernet connection between compute nodes of IB cluster.

ANUE/Ixia hardware-based emulator is used to collect MPI measurements over Ethernet connections with 11 Round Trip Times (RTT) in 0–366 ms range. These RTT values are strategically chosen to represent three ranges: (a) smaller values represent cross-country connections, for example, computing facilities distributed across the US, (b) 93–183 ms represent inter-continental connections, and (c) 366 ms represents a connection spanning the globe, which is mainly used as a limiting case. External periodic and random packets losses are introduced using ANUE/Ixia devices at four different loss rates. These emulators delay the packets as per the specified RTT value, and thus closely emulate the physical long distance path. Equally importantly, these emulations closely match TCP dynamics of physical connections with corresponding RTTs, which is a critical factor in assessing MPI performance over long distance connections. In particular, these emulations lead to different TCP dynamics and responses under deterministic (periodic) and random losses of Ethernet segments [14], which result in a wider spread of the execution times at high loss rates under random losses (Sect. 8).

3 Analytical Formulation

We now provide a formal description of the underlying estimation problems to support subsequent analytical treatment of RTT and loss rate estimation methods. Let E be a random variable representing the execution time of MPI Send_Receive primitive; it is distributed according to the joint probability distribution $\mathbb{P}_{E,R,L}$, where R and L are the random variables representing RTT and loss rate, respectively. In general, the distribution $\mathbb{P}_{E,R,L}$ is quite complex since it depends on the properties of the network connection and host systems, and also the software stack consisting of the operating system, networking and

MPI modules. Given an execution time measurement $E = e$, the conditional distribution $\mathbb{P}_{R,L|e} = \mathbb{P}_{R,L|E=e}$, characterizes the distribution of RTT and loss rate at this value e. Then, *RTT-regression* function f^R is defined as the expected value of RTT at $E = e$ given by

$$f^R(e) = \int R d\mathbb{P}_{R,L|e},$$

which is averaged over both R and L at each e. The *loss-regression* function f^L is the expected value of the loss rate at $E = e$, which is similarly given by

$$f^L(e) = \int L d\mathbb{P}_{R,L|e}.$$

In general, these regressions cannot be obtained even in theory since the underlying distribution $\mathbb{P}_{E,R,L}$ is unknown. In stead, ML methods are employed to estimate their approximations using a training sample $(E_i, R_i, L_i), i = 1, 2, \ldots, l$, wherein E_i is the execution time measured over a connection with RTT R_i and loss rate L_i. The distributions of the connection parameters R and L are determined by the design of connection configurations, and are fixed while the measurements of E are repeated. Thus, the distribution of E encompasses factors due to the properties of physical connection parameters as well as operating system, TCP and MPI modules.

Then, RTT and loss rate estimation problems can be cast as estimating the regression functions f^R and f^L, respectively, using measurements. We consider that RTT-regression estimate $\hat{f}_A^R$ is obtained by method $A \in \mathscr{A} = \{\text{EOT,GPR,LR,RT,SVM,LR}-\text{F,GPR}-\text{F}\}$ using the measurement pairs $(E_i, R_i), i = 1, 2, \ldots, l$. Similarly, the loss-regression estimate $\hat{f}_A^L$ is obtained by method $A \in \mathscr{A}$ using the measurement pairs $(E_i, L_i), i = 1, 2, \ldots, l$. At a given execution time $E = e$, $\hat{f}_A^R(e)$ and $\hat{f}_A^L(e)$ are the estimates of RTT and loss rate, respectively, provided by method A.

4 Execution Time Measurements

The execution times of MPI Send_Receive operations collected at the application-level are shown as a function of RTT in Fig. 4 for loss rates, 0.1, 1, 10 and 20%, of externally induced losses under four loss scenarios, one deterministic periodic and three random, namely uniform, Poisson and Gaussian. Their increasing trend as a function of RTT is evident at lower loss rates, 0.1% and 1%, but it becomes less prominent at 10% loss rate, and essentially disappears at 20% loss rate as outliers dominate. Overall, the execution times as well as their variations increase as loss rate is increased, which is an indication of the increased complexity of their estimation at higher loss rates.

In terms of losses, the execution times are shown as function of loss rates 0.1, 1, 10 and 20% in Fig. 5 under the four loss scenarios. The measurements at any loss rate encompass all 11 RTT values, and 10 repeated measurements

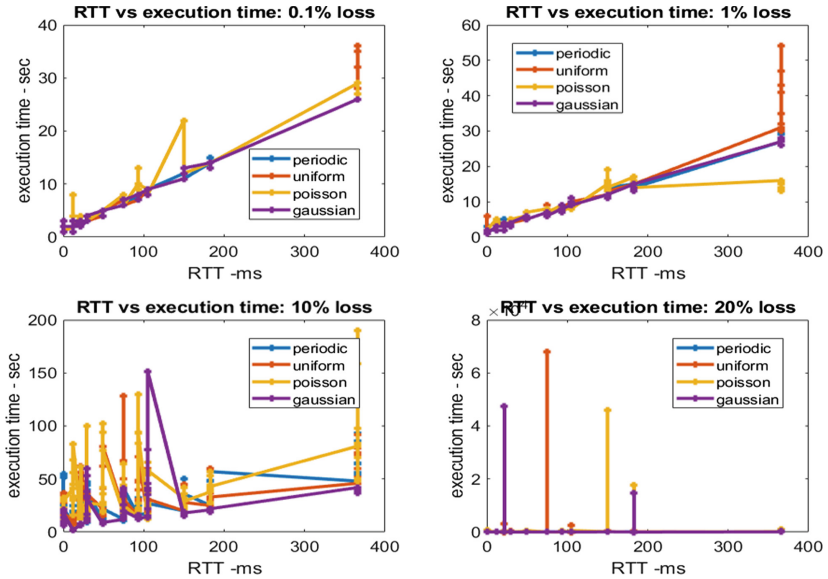

Fig. 4. Execution times of MPI_Sendrecv operations as function of RTT under four external loss scenarios, namely, periodic, uniform, Poisson and Gaussian.

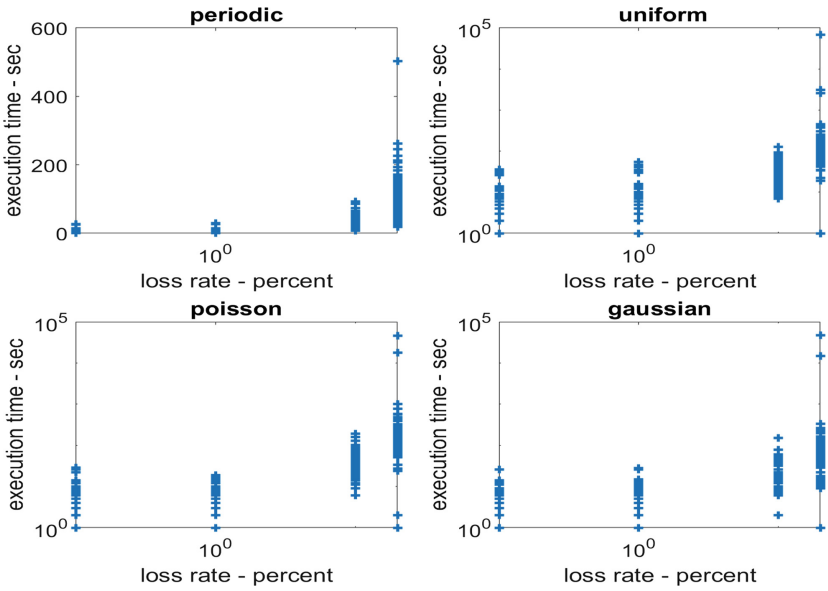

Fig. 5. Execution times of MPI_Sendrecv operations as function of loss rate under four external loss scenarios, namely, periodic, uniform, Poisson and Gaussian.

at each RTT value. Their increasing trend as a function of loss rate is evident overall but is sharper for periodic losses and is more diffused with overlaps across loss rates in all three random loss scenarios. The ranges of execution times are much wider for random losses compared to periodic losses. Also, the execution times as well as their variations increase overall as loss rate is increased for deterministic periodic loss scenario. But, in random loss scenarios the variations are more subtle: their spread is similar at all loss rates except at 20%, wherein a few measurements are very large, which indicate the complexity of loss rate estimation in these scenarios.

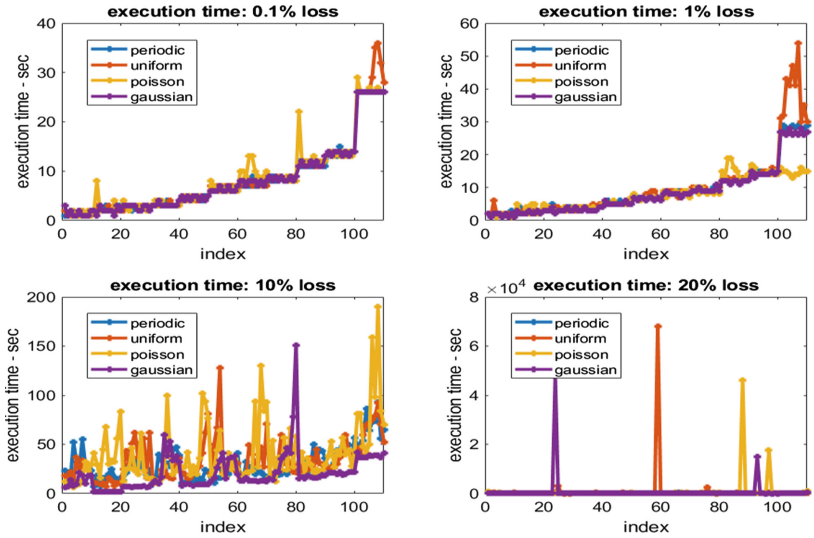

Fig. 6. Traces of execution times in seconds of MPI_Sendrecv operations under four external loss cases.

5 RTT Estimators

We present RTT estimates in the form of traces that are indexed by groups of 440 measurements that correspond to increasing loss rates, and within each group we have 11 sub-groups that correspond to increasing RTT values as shown in Fig. 2; each sub-group corresponds to 10 repeated measurements at a fixed pair of loss rate and RTT. The 440 measurements for each loss scenario shown in Fig. 6 will be used to compare qualitatively with the corresponding RTT estimate traces. We utilize the regression estimation codes from matlab statistics toolbox.

5.1 Five Estimators

The five estimation methods are chosen to reflect different characteristics of the underlying regressions, namely, smooth and non-smooth functions, respectively.

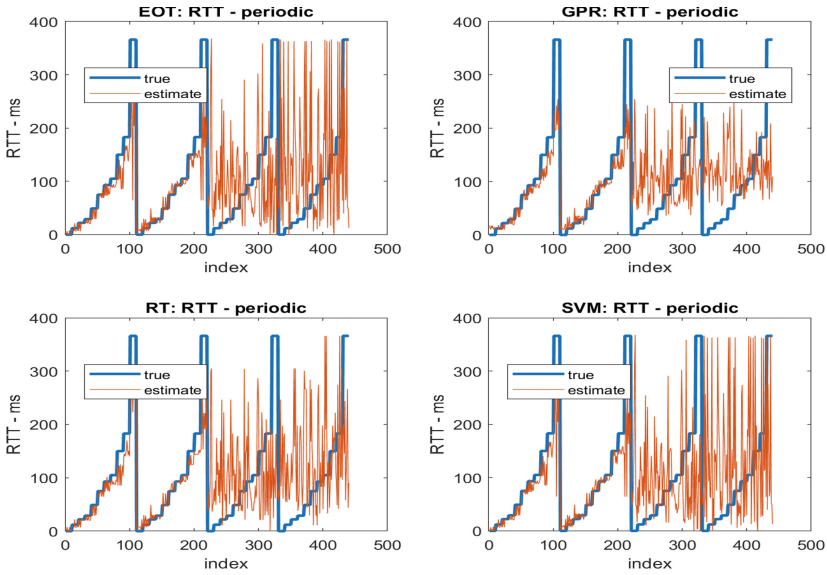

Fig. 7. Periodic losses.

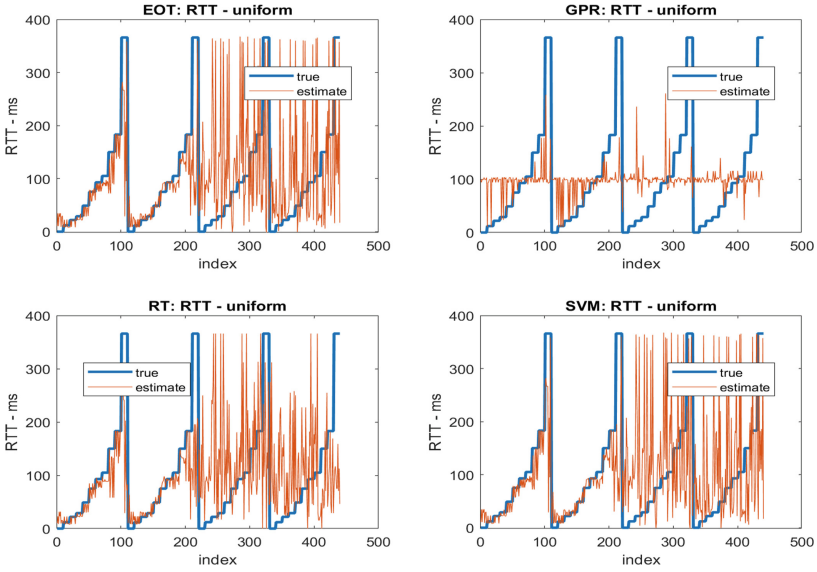

Fig. 8. Uniform losses.

GPR and SVM with Gaussian kernels [16] are based on non-linearly transforming the feature space X into regression space of Y. They both provide smooth regression functions f_{GPR} and f_{SVM}, and their respective function classes $\mathscr{F}_{\mathrm{GPR}}$ and $\mathscr{F}_{\mathrm{SVM}}$ consist of smooth functions as a result of Gaussian kernels. EOT [2,8] method is based on boosting of a collection of classification trees that are customized to fit the training data using the AdaBoost method. RT [3] methods is also based on trees that are customized to fit the training data. They both lead to a highly non-smooth regression functions f_{EOT} and f_{RT}, and their function classes $\mathscr{F}_{\mathrm{EOT}}$ and $\mathscr{F}_{\mathrm{RT}}$ consists of a collection of decision tree. LR is a smooth and linear method and leads to f_{LR} from the function classes $\mathscr{F}_{\mathrm{LR}}$, which is effective in RTT estimation under no losses [15] but is quite limited under losses as indicated by its RMSE in Tables 1 and 2.

The estimators under periodic, uniform, Poisson and Gaussian loss scenarios are shown in Figs. 7, 8, 9 and 10, respectively. Under periodic and Gaussian losses, all estimates are more accurate at 0.1% and 1% loss rate but are inaccurate at higher loss rates; in particular, they capture the increasing trends in RTT at low loss rate but exhibit high variation at 10% and 20% loss rate. Under unform and Poisson losses, GPR method does not capture the increasing RTT trend at any loss rate, while other non-linear estimates captured it. Interestingly, GPR achieved lowest RMSE among individual estimators which is due to the inclusion of measurements at higher loss rate that resulted in "averaging" across all loss rates. This undesirable artifact of low RMSE but less accurate estimate at low loss rates is illustrated in Figs. 8 and 9. LR and SVM methods have highest and second highest RMSE among the twenty cases in Table 1.

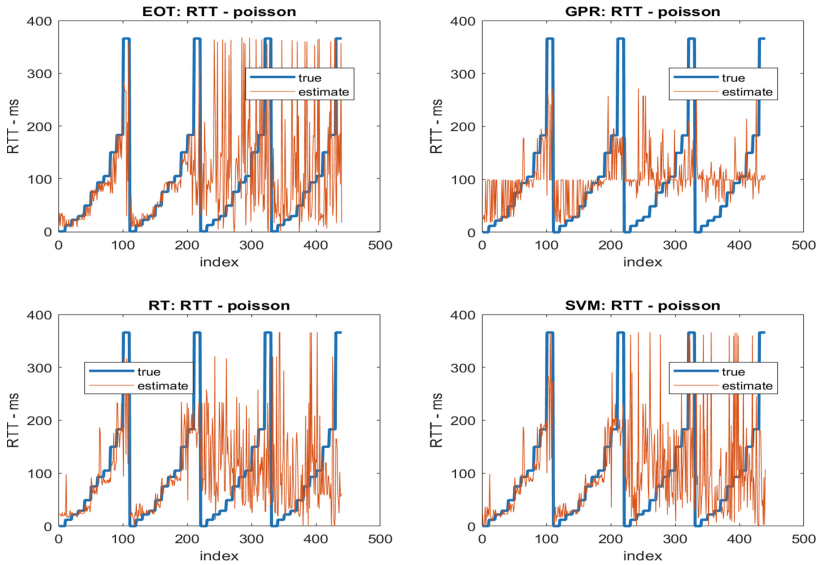

Fig. 9. Poisson losses.

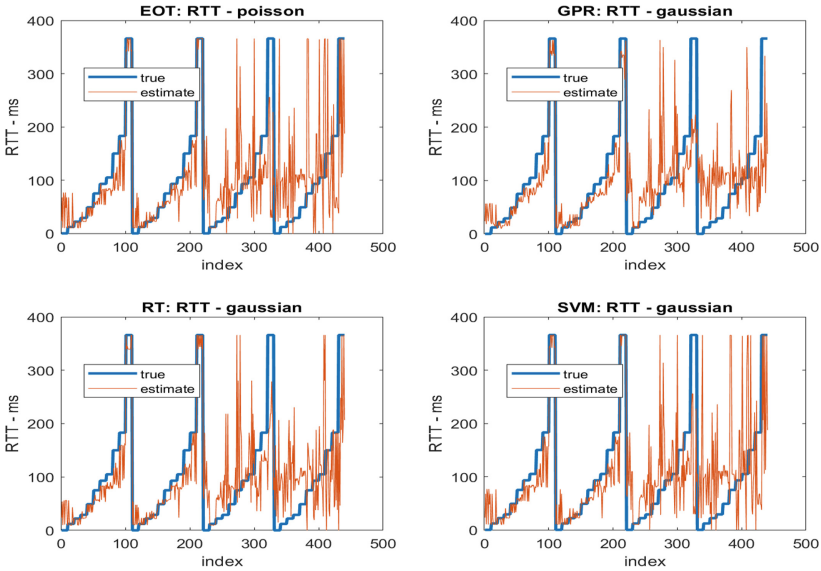

Fig. 10. Gaussian losses.

5.2 Estimator Fusers

The estimators from five individual methods are used as 5-dimensional input to a fuser which produces RTT as its output. The linear regression fuser (LR-F) is a linear combination of the individual estimators, and the GPR fuser (GPR-F) is obtained using GPR method based on the outputs of five estimators corresponding to the training sample. GPR-F achieved lower RMSE than best individual estimator GPR in all except under Gaussian losses, whereas LR-F has lower RMSE for Poisson and uniform losses. As shown in Fig. 1, the fused estimates were able to capture the increasing RTT trend at lower loss rates while achieving lower RMSE error than GPR under uniform and Poisson losses, unlike GPR estimator with lowest RMSE among individual estimators.

6 Generalization Equations for Fused Estimates

We consider five individual estimates, indexed by $A \in \mathscr{A}_I = \{\text{EOT,GPR,LR,RT,SVM}\}$, such that the fuser input vector X consists of five real-valued components, $X^A, A \in \mathscr{A}_I$, and output Y is a real-valued estimate of RTT. RMSE values in Table 1 have been used to compare the performance of fusers and individual estimators in previous sections, which are subject to statistical variations since they are computed based on a sample. We now derive confidence bounds for these RMSE values which provide analytical justification for their use. For simplicity of presentation of analytical results, we use MSE in place of RMSE following the common practice in finite sample theory [17].

6.1 Regression Problem: Finite Sample Generalization

In a generic regression estimation problem the feature vector $X \in \Re^d$ and the output vector $Y \in \Re$ are distributed jointly accordingly to an unknown distribution $\mathbb{P}_{X,Y}$. The *expected error* of a regression function f is

$$I(f) = \int (f(X) - Y)^2 \, d\mathbb{P}_{X,Y}.$$

The *expected best* regression estimator f^* minimizes $I(.)$ over $\mathscr{F}$, i.e., $I(f^*) = \min_{f \in \mathscr{F}} I(f)$. The *empirical error* $\hat{I}(f)$ based on training data $(X_i, Y_i), i = 1, 2, \ldots, l$, is defined as

$$\hat{I}(f) = \frac{1}{l} \sum_{i=1}^{l} (f(X_i) - Y_i)^2$$

It is an approximation of $I(f)$ computed based on the training data. The *empirical best* regression estimator $\tilde{f}$ minimizes $\hat{I}(.)$ over $\mathscr{F}$, i.e., $\hat{I}(\tilde{f}) = \min_{f \in \mathscr{F}} \hat{I}(f)$.

The joint distribution $\mathbb{P}_{X,Y}$ of data is complex, domain specific, and is only partially known. In our context, it depends on the finer details of the underlying software and hardware components, which may manifest as additional random variables. For an individual estimator $A \in \mathscr{A}_I$, X and Y correspond to execution time E and RTT R, respectively, and $\mathbb{P}_{X,Y}$ corresponds to $\mathbb{P}_{E,R,L}$ which involves additional random variable of the loss rate L. For fusers, X and Y correspond to 5-dimensional vector consisting of outputs of estimators and RTT R, respectively. In general, an optimal f^* cannot be computed precisely with probability one even in principle, since $\mathbb{P}_{X,Y}$ is either unknown or not computationally conducive. Under certain conditions, Vapnik's generalization theory [17] establishes that there exists a *confidence function* $\delta(.)$ such that for a "suitable" estimator $\hat{f}$ obtained from training data we have

$$\mathbb{P}^l_{X,Y} \left[I(\hat{f}) - I(f^*) > \varepsilon \right] < \delta(\varepsilon, \hat{\varepsilon}, l) \tag{1}$$

where $\varepsilon, \hat{\varepsilon} > 0$, $0 < \delta < 1$, and $\hat{I}(\hat{f}) = \min_{f \in \mathscr{F}} \hat{I}(f) + \hat{\varepsilon}$. This condition ensures that "error" of $\hat{f}$ is within ε of optimal error (of f^*) with probability $1-\delta$, *irrespective* of the underlying measured or computed data distribution $\mathbb{P}^l_{X,Y}$. Furthermore, under these conditions, the confidence parameter $\delta(\varepsilon, \hat{\varepsilon}, l)$ approaches 1 as the sample size l approaches infinity.

Consider the fuser class $\mathscr{F}_F$ used in fusing the estimators $f_A \in \mathscr{F}_A, A \in \mathscr{A}_I$. Let f_F denote the regression function obtained by composing f_A's with the fuser function from $\mathscr{F}_F$. The *error reduction* Δ_F of the fused estimate over the best individual classifier is defined as

$$\Delta_F = \min_{A \in \mathscr{A}_I} I(f_A) - I(f_F).$$

Then, if $\mathscr{F}_F$ has the isolation property [11], then $\Delta_F \geq 0$. The best error reduction is given by

$$\Delta^*_F = \min_{A \in \mathscr{A}_I} I(f^*_A) - I(f^*_F).$$

and its estimate based on a sample is given by

$$\tilde{\Delta}_F = \min_{A \in \mathscr{A}_I} \hat{I}(f_A) - \hat{I}(f_F).$$

The error reduction values Δ_F based on measurements are shown in Table 1 for the two fusers LR-F and GPR-F. GPR-F has positive $\tilde{\Delta}_F$ values in three scenarios indicating that the fused estimate has lower RMSE than the lowest of its constituent estimators (namely, GPR). LR-F has positive $\tilde{\Delta}_F$ values in two scenarios, which might be attributed to the lack of the required statistical independence in estimator outputs. We show in the next section that the estimate $\tilde{\Delta}_F$ reflects the optimal improvement Δ_F^* achievable by the fuser within a formal framework.

6.2 Estimator Fusers: Generalization Equations

The generalization bound $\delta(\varepsilon, \hat{\varepsilon}, l)$ applicable to five individual estimators can be derived using various properties of the corresponding estimator classes [12]. In particular, these bounds for GPR and SVM with Gaussian kernels could be based on fat-shattering index [16], and for EOT and RT they may be based on bounded total variation [1]. In Theorem 1, we assume that these generalization bounds are available from existing works, and their detailed derivations are beyond the scope of this paper.

We now show that the estimate $\tilde{\Delta}_F$ is within ε of the optimal Δ_F^* with a probability that improves with the training data size l independent of the underlying distribution $\mathbb{P}_{Y,X}$.

Theorem 1. *Consider that there exists $\delta_B(\varepsilon, \hat{\varepsilon}_B, l)$ such that based on i.i.d. l-sample, we have*

$$\mathbb{P}^l_{X,Y}\left[I(\hat{f}_B) - I(f_B^*) > \varepsilon\right] < \delta_B\left(\varepsilon, \hat{\varepsilon}_B, l\right). \tag{2}$$

for all individual estimators $B \in \mathscr{A}_I$, $N_{\mathscr{A}_I} = |\mathscr{A}_I|$, and both fusers $B = \text{LR-F}, \text{GPR-F}$ such that $\delta_B(\varepsilon, \hat{\varepsilon}_B, l) \to 0$ as $l \to \infty$. Then, the probability that the closeness between $\tilde{\Delta}_F$ and Δ_F^* is within ε is bounded as

$$\begin{aligned}&\mathbb{P}^l_{X,Y}\left[|\tilde{\Delta}_F - \Delta_F^*| < \varepsilon\right]\\ &\quad > 1 - \delta_D\left(\varepsilon/2, \hat{\varepsilon}_D, l\right) - \sum_{A \in \mathscr{A}_I} \delta_A\left(\varepsilon/(2N_{\mathscr{A}_I}), \hat{\varepsilon}_A, l\right),\end{aligned}$$

for both fusers $D = \text{LR-F}, \text{GPR-F}$.

Proof. We first note that for $D = \text{LR-F}, \text{GPR-F}$

$$|\tilde{\Delta}_F - \Delta_F^*| \le |\hat{I}(\hat{f}_D) - I(f_D^*)| + \left|\min_{A \in \mathscr{A}_I} \hat{I}(\hat{f}_A) - \min_{A \in \mathscr{A}_I} I(f_A^*)\right|,$$

which establishes that the condition $|\tilde{\Delta}_F - \Delta_F^*| > \varepsilon$ implies that at least one term on the right hand side is greater than $\varepsilon/2$. We now have

$$|\hat{I}(\hat{f}_D) - I(f_D^*)| \leq |\hat{I}(\hat{f}_D) - I(\hat{f}_D)| + |I(\hat{f}_D) - I(f_D^*)|,$$

which in turn establishes that the condition $|\hat{I}(\hat{f}_D) - I(f_D^*)| > \varepsilon/2$ implies that at least one term on the right hand side is greater than $\varepsilon/4$. Then, by hypothesis in Eq. (2), both conditions are simultaneously satisfied with probability at most $\delta_D\left(\varepsilon/4, \hat{\varepsilon}_d, l\right)$. Similarly, we have

$$\left|\min_{A\in\mathscr{A}_I} \hat{I}(\hat{f}_A) - \min_{A\in\mathscr{A}_I} I(f_A^*)\right| \leq \left|\min_{A\in\mathscr{A}_I} \hat{I}(\hat{f}_A) - \min_{A\in\mathscr{A}_I} I(\hat{f}_A)\right| + \left|\min_{A\in\mathscr{A}_I} I(\hat{f}_A) - \min_{A\in\mathscr{A}} I(f_A^*)\right|,$$

which in turn establishes that the condition $\left|\min_{A\in\mathscr{A}_I} \hat{I}(\hat{f}_A) - \min_{A\in\mathscr{A}_I} I(f_A^*)\right| > \varepsilon/2$ implies that at least one term on the right hand side is greater than $\varepsilon/4$. Then, we consider the two upper bounds

$$\left|\min_{A\in\mathscr{A}_I} \hat{I}(\hat{f}_A) - \min_{A\in\mathscr{A}} I(\hat{f}_A)\right| \leq \sum_{A\in\mathscr{A}_I} \left|\hat{I}(\hat{f}_A) - I(\hat{f}_A)\right|$$

$$\left|\min_{A\in\mathscr{A}_I} I(\hat{f}_A) - \min_{A\in\mathscr{A}_I} I(f_A^*)\right| \leq \sum_{A\in\mathscr{A}_I} \left|I(\hat{f}_A) - I(f_A^*)\right|.$$

In each case, the condition that left hand side is larger than $\varepsilon/2$ implies at least one of the terms under the summation is greater $\varepsilon/(2N_{\mathscr{A}_I})$. Under the hypothesis of this theorem in Eq. (2), both conditions are satisfied with probability at most

$$\sum_{A\in\mathscr{A}_I} \delta_A(\varepsilon/(2N_{\mathscr{A}_I}), \hat{\varepsilon}_a, l).$$

By combining the above terms together, we have

$$\mathbb{P}^l_{X,Y}\left[|\tilde{\Delta}_F - \Delta_F^*| > \varepsilon\right] < \delta_D\left(\varepsilon/2, \hat{\varepsilon}_D, l\right) + \sum_{A\in\mathscr{A}_I} \delta_A\left(\varepsilon/(2N_{\mathscr{A}_I}), \hat{\varepsilon}_A, l\right),$$

which proves the theorem. □

The confidence bound in this theorem is distribution-free in that it does not depend on $\mathbb{P}_{X,Y}$. It is expressed in terms of the *precision* parameter ε and the *confidence* parameter $\left[1 - \delta_D\left(\varepsilon/2, \hat{\varepsilon}_D, l\right) - \sum_{A\in\mathscr{A}_I} \delta_A\left(\varepsilon/(2N_{\mathscr{A}_I}), \hat{\varepsilon}_A, l\right)\right]$, which approaches 1 with increasing number of measurements l.

Table 2. Loss rate estimates RMSE of five estimators and linear fuser.

Loss type	EOT	GPR	LR	RT	SVM	LR-F
Gaussian	**5.19**	6.39	5.34	6.82	7.17	6.42
Periodic	7.26	**6.62**	6.82	7.00	6.91	6.81
Poisson	6.99	**6.61**	7.42	6.93	6.72	6.74
Uniform	7.26	**6.62**	6.82	7.00	6.91	6.81

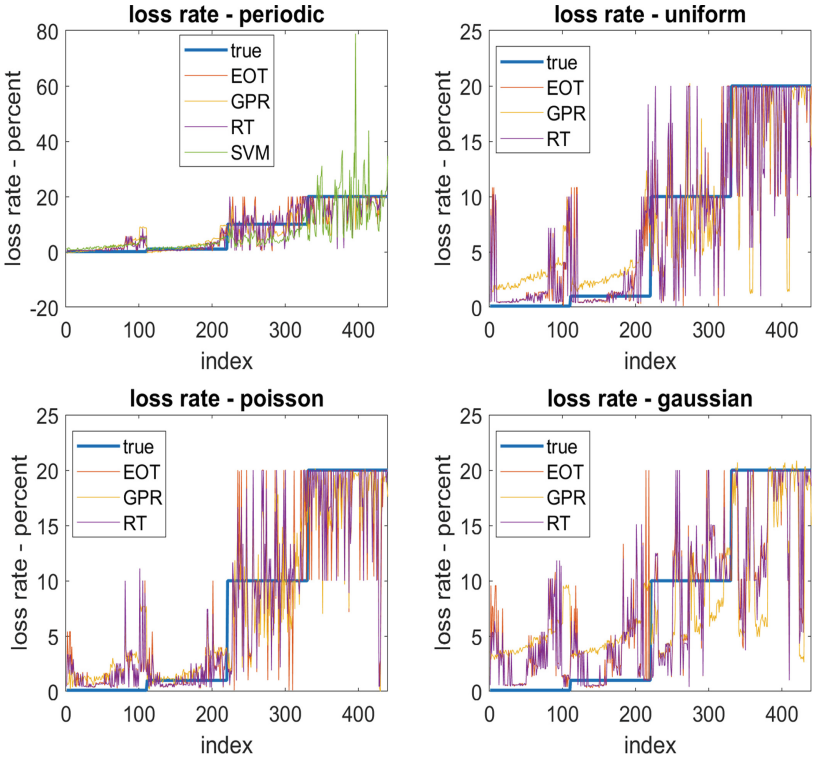

Fig. 11. Loss rate estimates with lowest RMSE of individual methods.

7 Loss Rate Estimators

The loss estimates of four non-linear estimators are shown in top left plot of Fig. 11 for periodic losses. For random losses, SVM estimates have extreme variations and hence are omitted in the plots. Also, LR estimator is omitted in all plots due to its extremely large variation under all loss scenarios. Qualitatively, smooth SVM and non-smooth RT methods both exhibit large variations, which indicate the underlying properties of the data rather than these methods; indeed GPR is the only method that did not produce large variations. RMSE of five estimators and linear fuser LR-F are shown in Table 2. Methods with lowest

RMSE for loss rate estimation are shown in Fig. 12, which are EOT for periodic losses and GPR for all others. Both of them exhibited lower variations at low loss rate and an increasing trend as RTT is increased at a fixed loss rate. As in the case of RTT estimation, the "averaging" by GPR resulted in lower RMSE but less accurate estimates at low loss rates; but, interestingly, this effect is more dominant for Gaussian errors unlike for RTT estimation. In almost all cases, at fixed loss rate, the estimators showed an increasing trend as RTT is increased.

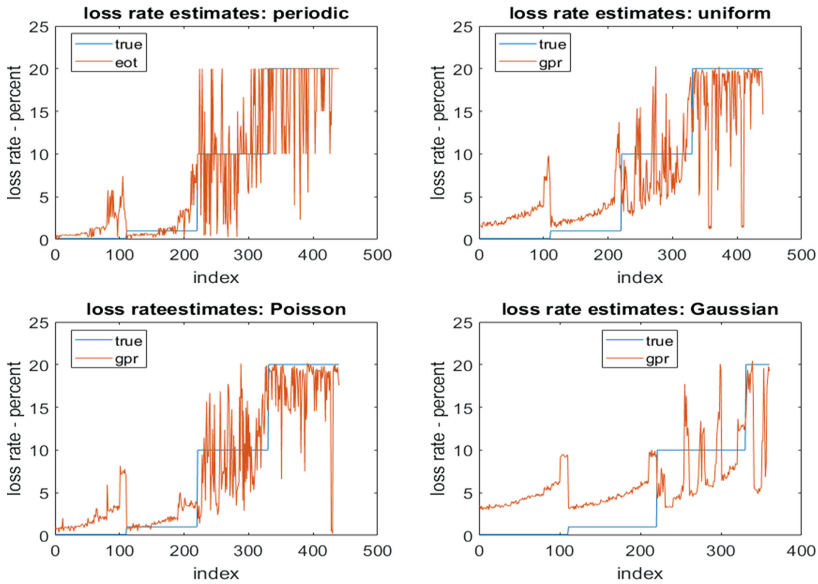

Fig. 12. Loss rate estimates with lowest RMSE of individual methods.

The large variations shown in the scatter plots in Fig. 5 indicate high RMSE by any estimate since its output is a function and data dispersed around it contributes to RMSE. In particular, a smooth estimate will not be able to capture these variations as applicable to GPR and SVM methods. While tree-based methods in principles can capture such variations, they require a large number of leaf nodes, and those with smaller number will result in large RMSE. In summary, the results indicate the challenging nature of the underlying datasets, which in some sense expose the limitations of the conventional ML approaches for loss rate estimation.

8 Execution Time Measurements: Data Regressions

Qualitative insights into the performance of regression estimators can be gained by examining the scatter plots of measurements separately at low and high loss rates. Overall increasing trend of RTT when plotted as a function of execution

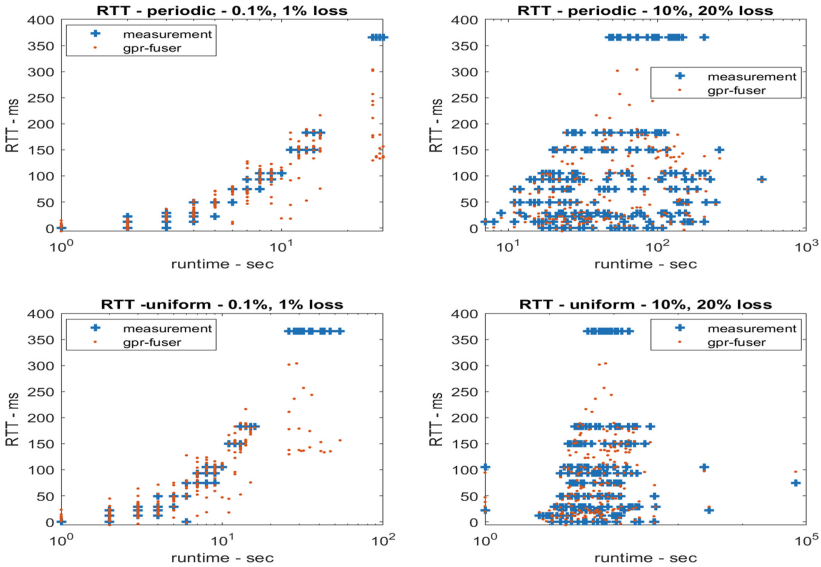

Fig. 13. Data regressions under periodic and uniform losses.

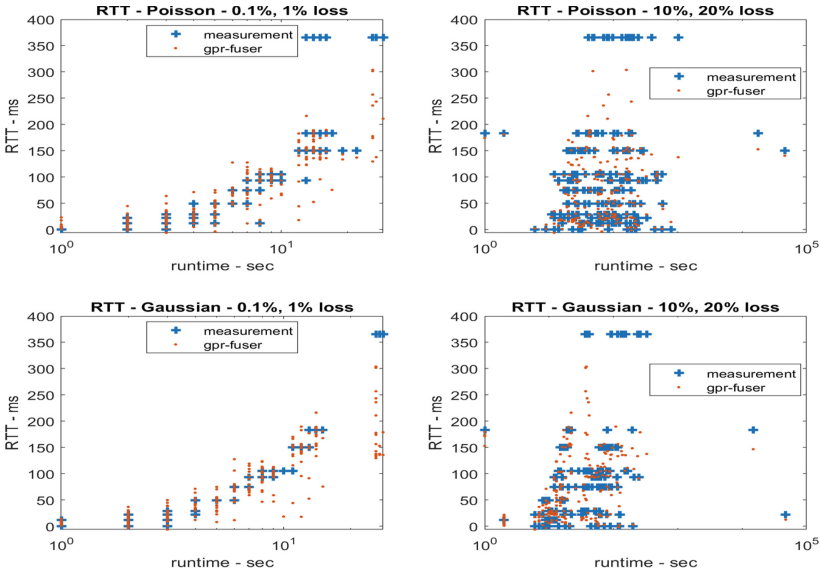

Fig. 14. Data regressions under Poisson and Gaussian losses.

time is evident under 0.1% and 1% loss rates, but not under and 10% and 20% loss rates in all four scenarios, as shown in Fig. 13 for periodic and unform losses, and in Fig. 14 for Poisson and Gaussian losses. Even GPR-F estimate with the lowest overall RMSE captures the increasing trend only in the former case but not in the latter case. Qualitatively, the wide spread of measurements at high loss rates indicates the lack of information needed to estimate RTT by any method that uses regression function, smooth or non-smooth.

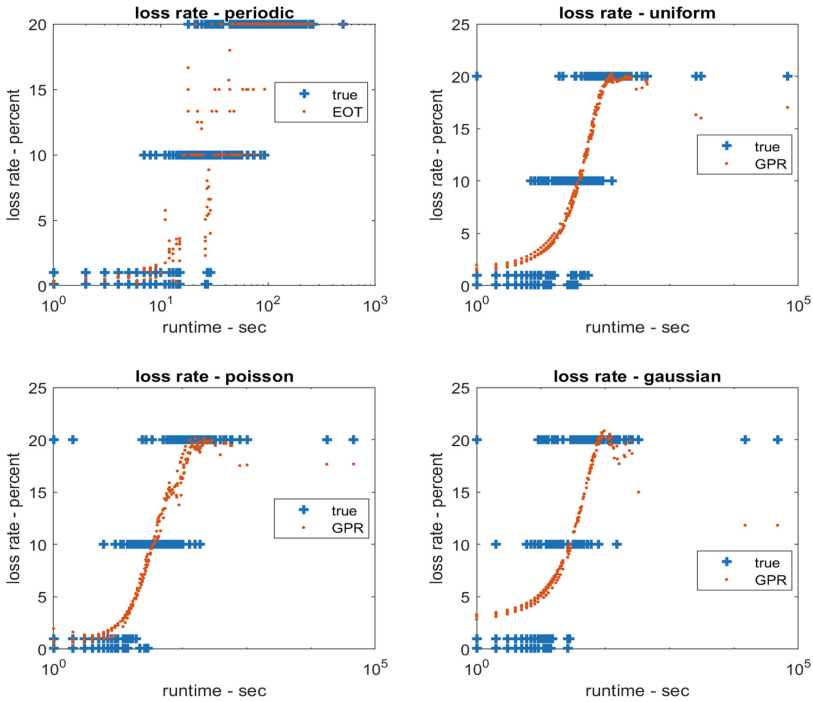

Fig. 15. Data regressions of loss estimators with lowest RMSE.

For loss rates, the scatter plots are shown in Fig. 15, which have significant variations in execution times at fixed values of loss rate. The non-smooth EOT method with lowest RMSE under periodic losses captures several loss rates at 10 and 20% loss rates as shown in top left plot. The smooth GPR estimator is plotted along with data in all three random loss scenarios in which it has lowest RMSE; several estimated points are in between the loss rate values, and the estimator shows a continuous trend in the mapping from measurements to loss rate. Similar to RTT estimates, an overall increasing trend of loss rate when plotted as a function of execution time is evident under 0.1% and 1% loss rate; but, the loss rate is fixed at two values 0.1 and 1% as shown in Fig. 16 for periodic and unform losses, and in Fig. 17 for Poisson and Gaussian losses.

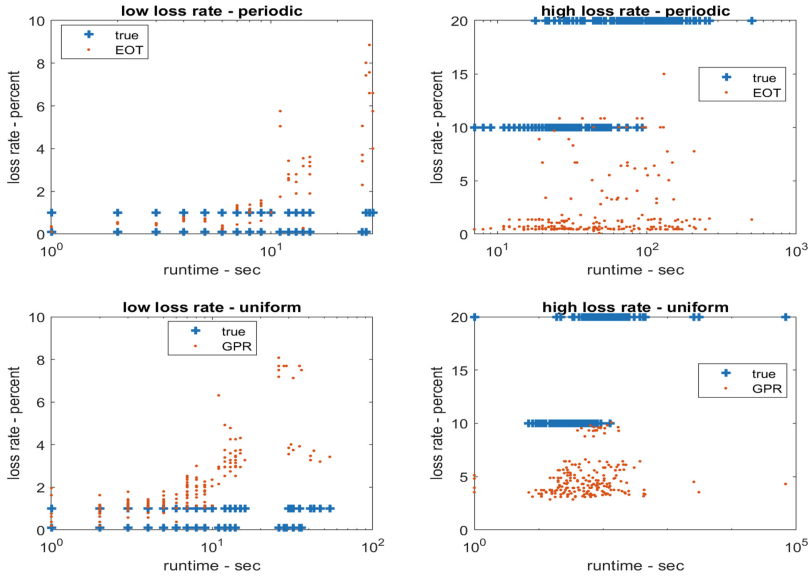

Fig. 16. Loss regression at low and high loss rates for periodic and uniform losses.

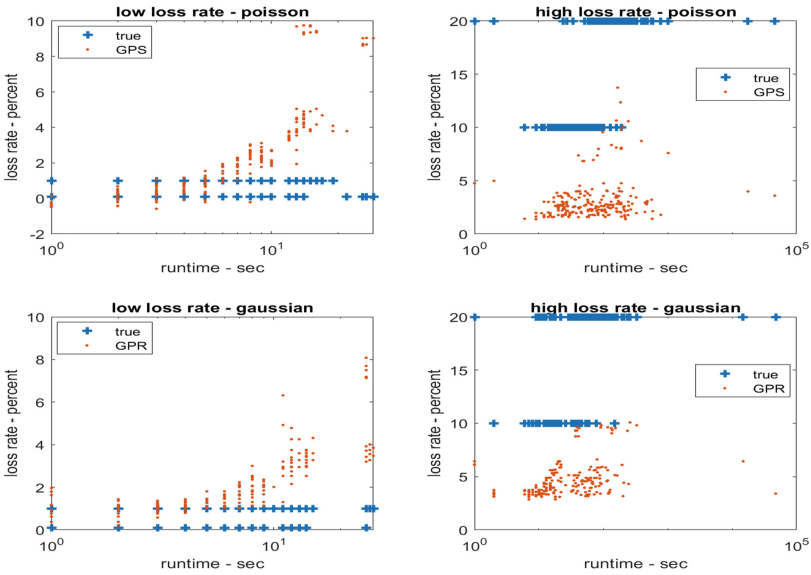

Fig. 17. Loss regression at low and high loss rates for Poisson and Gaussian losses.

This is a "bleed over" artifact due to increasing RTT at both 0.1 and 1%loss rates. Under 10% and 20% loss rates in all four scenarios there is a significant scatter in loss estimates, indicating the underlying complexity of regression estimation.

9 Conclusions

Rich datasets of MPI measurements are becoming increasingly available as more and more computations are distributed over wide-area networks. These measurements exhibit certain characteristics, such as longer executions times and large variations, that are atypical of conventional MPI applications executed on single computing systems with Inifiniband or custom interconnects. Losses are integral to wide-area networks as TCP that supports MPI utilizes self-induced losses to pace its flows. Consequently, these distributed computations need to mitigate the inefficiencies due to network delays and their variations. These computations may be distributed across geographically dispersed nodes that are dynamically identified; consequently, the RTTs and loss rates of the underlying connections may not be a priori known. The MPI measurements collected at the application-level reflect the connection length and losses, and have been shown to be useful in estimating RTT and loss rate using ML methods, albeit accurately only at low loss rates.

Complementing previous works under deterministic periodic loss scenarios, we studied five ML methods to estimate the connection RTT and loss rates under random losses, which are more reflective of practical scenarios. As in previous works [15], the results show that accurate estimates can be generated at low loss rates but they become inaccurate at loss rates 10% and higher. However, this randomness manifests in subtle ways, resulting in different performances of non-linear estimators; in particular, GPR that achieves low RMSE does not provide accurate RTT estimates at low loss levels, unlike others with higher RMSE. These effects are mainly due to the highly non-linear response of the underlying TCP dynamics that "amplify" the randomness of losses. Furthermore, it is equally complex to assess the performance of ML methods due to their non-linear nature, and their fusers are only effective in some scenarios for RTT estimation. In another direction, these results highlight the strengths and limitations of ML methods for network-level estimation problems using application-level measurements.

This work constitutes only initial steps in understanding the complexity of estimating network-level parameters using application-level measurements, and the performance of various ML solutions, including individual and fused estimates. Future work may involve studying the random losses due to external traffic in production networks, which may not follow known random processes. Since there is no universal way to choose among various ML methods from sample performance only, it would be of future interest to investigate into domain specific customizations, hyper-parameter tuning, fusers and other approaches to RTT and loss rate estimation [12,13].

References

1. Anthony, M., Bartlett, P.L.: Neural Network Learning: Theoretical Foundations. Cambridge University Press, Cambridge (1999)
2. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
3. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: Classification and Regression Trees. Wadsworth and Brooks, Grove (1984)
4. Chhabra, A., Kiran, M.: Classifying elephant and mice flows in high-speed scientific networks. In: IEEE/ACM Workshop on Innovating the Network for Data-Intensive Science (INDIS) (2017)
5. Devroye, L., Gyorfi, L., Lugosi, G.: A Probabilistic Theory of Pattern Recognition. Springer, New York (1996). https://doi.org/10.1007/978-1-4612-0711-5
6. Giannakou, A., Gunter, D., Peisert, S.: Flowzilla: a methodology for detecting data transfer anomalies in research networks. In: IEEE/ACM Workshop on Innovating the Network for Data-Intensive Science (INDIS) (2018)
7. Hassan, M., Jain, R.: High Performance TCP/IP Networking: Concepts, Issues, and Solutions. Prentice Hall, Upper Saddle River (2004)
8. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer, New York (2001). https://doi.org/10.1007/978-0-387-84858-7
9. Lakshman, T.V., Madhow, U., Suter, B.: TCP/IP performance with random loss and bidirectional congestion. IEEE/ACM Trans. Netw. **8**(5), 541–555 (2000)
10. Rao, N.S.V.: On fusers that perform better than best sensor. IEEE Trans. Pattern Anal. Mach. Intell. **23**(8), 904–909 (2001)
11. Rao, N.S.V.: Measurement-based statistical fusion methods for distributed sensor networks. In: Iyengar, S.S., Brooks, R.R. (eds.) Distributed Sensor Networks, 2nd edn. Chapman and Hall/CRC Publishers, Boca Raton (2011)
12. Rao, N.S.V.: Finite-sample generalization theory for machine learning practice for science. In: DOE ASCR Scientific Machine Learning Workshop (2018)
13. Rao, N.S.V., et al.: Classifiers for dissolution events in processing facility using effluents measurements. In: Institute of Nuclear Materials Management Annual Meeting (2019)
14. Rao, N.S.V., Imam, N., Boehm, S.: A case study of MPI over long distance connections. In: 13th Annual IEEE International Systems Conference (2019)
15. Rao, N.S.V., Imam, N., Liu, Z., Kettimuthu, R., Foster, I.: Estimation of RTT and loss rate of wide-area connections using MPI measurements. In: IEEE/ACM Workshop Innovating the Network for Data-Intensive Science (INDIS) (2019)
16. Scholkopf, B., Burges, C.J.C., Smola, A.J. (eds.): Advances in Kernel Methods. MIT Press, Cambridge (1999)
17. Vapnik, V.N.: Statistical Learning Theory. Wiley, New York (1998)