## Title

Decentralizing Network Inference Problems with Multiple-Description Fusion Estimation (MDFE)



## Authors

Malboubi, Mehdi
Vu, Cuong
Chuah, Chen-Nee
et al.

## Publication Date

2015-09-01

Peer reviewed

# Decentralizing Network Inference Problems with Multiple-Description Fusion Estimation (MDFE)

Mehdi Malboubi, *Student Member, IEEE*, Cuong Vu, Chen-Nee Chuah, *Fellow, IEEE*, and Puneet Sharma, *Fellow, IEEE*

***Abstract*—Network inference (or tomography) problems, such as traffic matrix estimation or completion and link loss inference, have been studied rigorously in different networking applications. These problems are often posed as under-determined linear inverse (UDLI) problems and solved in a *centralized* manner, where all the measurements are collected at a central node, which then applies a variety of inference techniques to estimate the attributes of interest.**

**This paper proposes a novel framework for *decentralizing* these large-scale under-determined network inference problems by intelligently partitioning it into smaller sub-problems and solving them independently and in parallel. The resulting estimates, referred to as *multiple descriptions*, can then be fused together to compute the global estimate. We apply this Multiple Description and Fusion Estimation (MDFE) framework to three classical problems: traffic matrix estimation, traffic matrix completion, and loss inference. Using real topologies and traces, we demonstrate how MDFE can speed up computation while maintaining (even improving) the estimation accuracy and how it enhances robustness against noise and failures. We also show that our MDFE framework is compatible with a variety of existing inference techniques used to solve the UDLI problems.**

***Index Terms*—Network inference, distributed estimation, multiple description fusion, traffic matrix estimation, under-determined inverse problem.**

## I. Introduction

Due to the complexity of current Internet and the exploding volume of data traffic, there are often aspects of the networks that are challenging or infeasible to measure directly. This has drawn researchers to the field of network inference (e.g. network tomography [1]), which involves applying a variety of inference strategies to estimate network's internal characteristics based on a partial set of measurements. Many network inference problems are formulated as Under-Determined Linear Inverse (UDLI) problems. These problems are naturally ill-posed in the sense that the number of measurements are not sufficient to uniquely determine the solution. Two forms of network inference problems [2] have been studied rigorously: (a) origin-destination (path-level) traffic volume estimation based on link-level traffic measurements, such as traffic matrix (TM) estimation [3] or TM completion [4], and (b) link-level parameter's (such as loss, delay, or bottleneck bandwidth) estimation based on end-to-end measurements [5], [6], [7].

Mehdi Malboubi, and Chen-Nee Chuah are with the Dept. of Electrical and Computer Engineering, University of California at Davis, Davis, CA 95616 USA (e-mail: {mmalboubi,chuah}@ucdavis.edu).

Cuong Vu graduated from the Dept. of Computer Science, University of California at Davis, Davis, CA 95616 USA (e-mail: cnvu@ucdavis.edu).

Puneet Sharma is with HP-Labs, Palo Alto, CA 94304 USA (e-mail: puneet.sharma@hp.com)

Prior work has mostly focused on designing better measurement methodology and inference techniques to improve the accuracy of the solution. For this purpose, side information is incorporated to change an ill-posed problem to a well-posed problem. Side information, based on the application, is provided from different sources , e.g., auxiliary measurements such as NetFlow data [8], and from diverse perspectives, e.g., using underlying deterministic or statistical models [3], [4], [7], [9].

Although the uniqueness and accuracy of the solution are important, many network inference problems need to be solved in a timely manner for practical deployment. Nevertheless, most existing studies attempt to solve the network inference problem in a one-shot, centralized manner, where all measurements are collected at a central node, which then applies domain-specific inference techniques to estimate the attributes of interest. However, the computational complexity of these centralized inference techniques hinders their deployment in large-scale production networks.

This paper tackles these network inference problems from a new angle and asks the question: *can we design an efficient and robust framework to solve these large-scale UDLI problems in a decentralized manner?* Our goal is to speed up the computation process to produce timely estimates (especially in a dynamic network environment), without compromising the accuracy of the solution. Towards this end, we propose Multiple Description and Fusion Estimation (MDFE) framework that decentralizes a large-scale network inference problem by intelligently partitioning it into smaller sub-problems and solving them independently and in parallel. The results, solved in respective sub-spaces and referred to as multiple descriptions, are then fused together to reconstruct the global estimate. Each sub-space could potentially produce a more precise description of a sub-set of the solution; in fact, these descriptions are considered as side/supplementary information for each other, provided from different perspectives.

MDFE is a flexible framework that can be applied to different UDLI problems, and is complementary to the inference techniques proposed previously for solving specific network inference problems. In this paper and in [10], we demonstrate how MDFE can be applied to network inference problems such as TM Estimation (TME), TM Completion (TMC) and Loss Inference (LI), and we show, MDFE is compatible with different previously proposed inference techniques, including least square error estimation, expectation maximization, and regularized matrix factorization methods [8], [11], [4], [7].

By reducing the problem complexity, MDFE can signif-

icantly speed up the computation and reduce required processing power. Through evaluation using real topologies and data, we demonstrate the possibility of achieving this computational efficiency without compromising the accuracy of the global estimate. This, specifically, has important implications in distributed and dynamic environments (e.g., distributed data centers or clouds), where inference process must be performed at much faster time scales. This framework is suitable for today's computing paradigm where a large-scale problem can be divided into smaller sub-problems and distributed among multi processors. Also, by exploiting redundancy between different sub-spaces, MDFE can enhance the robustness against noise and failures in the monitoring infrastructures. It also can reduce the overhead involved in sending all measurements to a central node for global estimation.

The improvement in the estimation accuracy using MDFE depends on the structure of the problem, sub-space estimation method, partitioning technique and the fusion process which are discussed in this paper. Our main contributions are:

- To the best of our knowledge, we are first to develop the concept and theory of MDFE for solving large-scale UDLI problems. We demonstrate how to effectively design the MDFE framework and realize it in practice.
- We develop and evaluate three algorithms to partition the original large-scale problem into smaller sub-problems under MDFE; we also introduce different fusion methods to combine the multiple descriptions to produce the global solution.
- We demonstrate the efficacy of MDFE in practice by applying it to three important problems in network monitoring and management: TM estimation , TM completion and loss inference.
- Using realistic network topologies and traffic data, we show how MDFE can speed up computation by maintaining (and even improving) the accuracy of the global estimates.

The rest of this paper is organized as follows. Section II discusses the most relevant work in the context of the three example network inference problems. Section III develops the theory of MDFE and addresses main steps in the implementation of this framework in practice. In Section IV we define the metrics that we have used in evaluating the performance of our estimation framework and introduce the networks and data sets under our study; furthermore, we explain the details of the MDFE process in solving UDLI problems using illustrative examples. Then, in Section V the performance and efficiency of this framework are evaluated for different applications in networking. Finally, Section VI summarizes the main results of the paper. Due to space limitation, we refer to [10] for more detailed discussions and additional results.

## II. Background and Related Work

There is a rich literature on network tomography and it would be impossible to enumerate all the related work. We would like to emphasize that the *main goal* of this paper is not to design new, improved algorithm for solving specific network inference problem. Instead, we are proposing a framework for efficiently solving a class of UDLI problems by adopting a divide-and-conquer approach and leveraging existing inference techniques to solve the intelligently partitioned sub-problems under MDFE. In this paper, three network inference problems (TM estimation, TM completion, and loss inference) are used to showcase MDFE framework (see Section V). Here, we will briefly discuss the most relevant work in the context of these network inference problems, where side information from different sources/perspectives is provided to uniquely identify the solution.

The traffic matrix (TM) is a measure of origin-destination traffic intensity that can be defined at different levels: between routers, IP-prefixes, or even AS domains. It provides essential information for network design, traffic engineering, and anomaly detection. TM estimation [12] is often formulated as a linear constrained optimization problem where link and flow conservation constraints are added to reduce the feasible solution space. Side information can also be provided as the underlined statistical models of Origin-Destination Flows (ODFs). In [1] and [13], it is assumed that ODFs are generated from a collection of independent Poisson distributions and Bayesian estimator is used to estimate the parameters of the Poisson distributions. In [11] independent Gaussian distributions are considered for ODFs and Maximum Likelihood Estimation (MLE) is used to estimate the parameters of the Gaussian distributions. To make the TM estimation more accurate and robust, [8] has combined data from multiple sources including SNMP link loads and *Sampled* NetFlow records.

On the other hand, for TM completion, the low-rank property of TMs are used for interpolating missed TMs. The low-rank property means that TM entries are related and a small amount of information can be used to construct the original TM [14]. A sparsity regularized matrix factorization method is developed in [4] to find local low-rank approximations of TMs that account for spatial and temporal properties of real TMs. These low-rank approximations are augmented with local interpolation to estimate missing TM values. The required side information are provided by capturing local spatial-temporal structures and redundancies.

Another network inference problem is network performance tomography, which is defined as the inference of internal link properties from end-to-end measurements [6]. In [7], links loss rate inference problem is modeled as an UDLI problem where two key properties of network losses are used as side information to uniquely estimate link loss rates: first, losses due to congestion occur in bursts (thus loss rates of congested links have high variances), and second, loss rates of most un-congested links in the Internet have virtually zero first- and second-order moments. Ghita et al [15], [16] applied network tomography to identify frequency with which peer links are congested in practical scenario that considers correlation between links. Also, in [17], first and second-order moments of end-to-end measurements are combined to estimate loss rates.

## III. Estimation with Multiple Description Fusion

Consider the under-determined linear system of equations Eq.(1) where $Y$ is an $(m \times 1)$ observation vector, $H$ is an

$(m \times n)$ observation matrix $(m < n)$ and $X$ is an $(n \times 1)$ vector of unknowns. The general solution to this problem is of the form $\hat{X} = X + \mathcal{N}(H)$ where $\mathcal{N}(H)$ represents a solution from the span of the null space of $H$; therefore, there are many solutions for this problem. A linear inverse problem is defined as the process of uniquely inferring $X$ as a linear function of observation $Y$ which can be formulated as an (un)constrained optimization problem where the main goal is to minimize error $e = Y - HX$ in an appropriate sense.

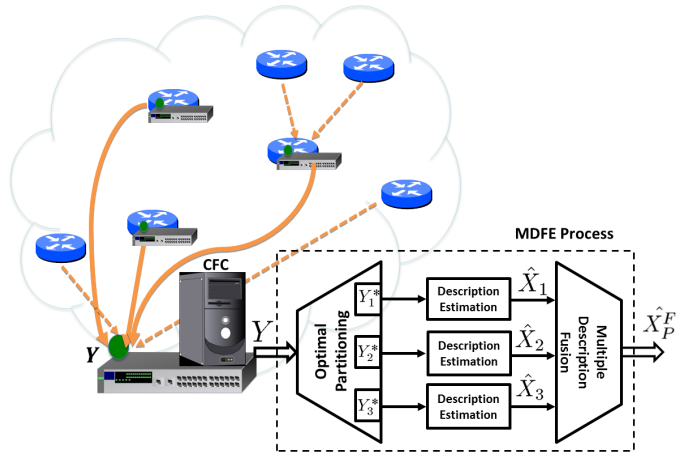
$$Y = HX \tag{1}$$

In MDFE framework, the original (global) UDLI problem described by Eq.(1) is partitioned into $L$ local sub-problems shown in Eq.(2), which are independently solved and sub-space estimates/descriptions $\{\hat{X}_i\}_{i=1}^{L}$ are then fused together to improve the efficiency of computations without compromising the accuracy of the solution (Figure 1(a)). At the Central Fusion Center (CFC) where all local descriptions are available, the fusion process is accomplished by applying appropriate weights to each local estimate during the fusion phase. Eq.(3) describes this process where operator $\oplus$ denotes the fusion process of the partitioned problem, that is, combining the subset of unknowns observed and estimated by different sub-spaces. Figure 1(b) gives an intuitive perspective of this Multiple Description Fusion (MDF) process where the original problem is partitioned into 3 sub-problems. Each sub-problem is solved to estimate three local views of the TM, namely $\{\hat{X}_i\}_{i=1}^{3}$. After applying appropriate weights to each local estimate, OD flows are fused and added together to construct estimate $\hat{X}_P^F$.

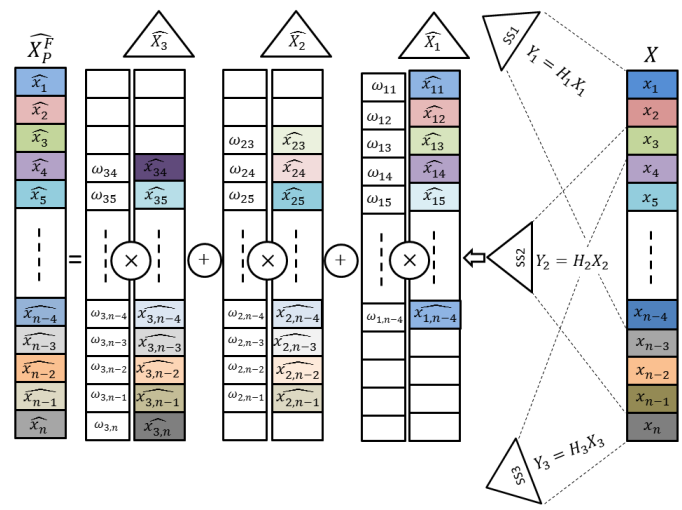
$$Y = HX \Leftrightarrow \begin{bmatrix} Y_1 \\ \hline \vdots \\ \hline Y_L \end{bmatrix} = \begin{bmatrix} H_1 X_1 \\ \hline \vdots \\ \hline H_L X_L \end{bmatrix} \tag{2}$$

$$\hat{X}_P^F = \oplus_{i=1}^{L} \omega_i^F \hat{X}_i \tag{3}$$

The overall performance of MDFE framework for computing $\hat{X}_P^F$ is a *joint* function of sub-space estimation technique, measurement set partition $P$ and fusion process $F$. Hence, to successfully apply the MDFE framework in practice, three steps must be accomplished correctly: a) effectively partition the problem into sub-problems, b) construct multiple descriptions by adopting proper sub-space estimation techniques to solve the sub-problems, and c) fuse the sub-space estimates to provide more precise and robust description. The essence of this joint optimization problem lies in an NP-hard set partitioning problem that is extremely difficult to solve. Hence, we *decouple* and address steps a-c, independently. Since the estimation techniques to solve specific network inference problems are well studied, we first discuss how these existing techniques can be leveraged to provide sub-space estimates (step b). Then, taking practical constraints into account, we discuss the design of the most effective partitioning and fusion methods. It should be noted that, to have a fair comparison between the global and MDFE cases, the estimation techniques among both cases are the same.




(a) General block diagram of MDFE.




(b) TM Estimation with MDF from sub-spaces.

*Fig. 1:* MDFE process: a system perspective.

### A. MDFE in Practice: Multiple Description Construction

To construct multiple descriptions, sub-inference problems must be properly defined and the best sub-space estimation technique is selected, depending on the characteristics of the input $X$, matrix $H$ and problem's side information(/constraints).

Let $I$ denotes the set of all indicies of observations ($I = \{1, 2, ..., m\}$) and $I_i$ denotes the $i^{th}$ set of disjoint indices of measurements where $I = \bigcup_{i=1}^{L} I_i$ and $I_i \bigcap I_j = \emptyset$ for $i \neq j$. Then, set $P = \bigcup_{i=1}^{L} I_i$ forms a *Partition* of $I$. Let $J$ denotes the set of all indices of unknowns ($J = \{1, 2, ..., n\}$) and $J_i$ denotes the $i^{th}$ set of indices of unknowns where $J = \bigcup_{i=1}^{L} J_i$; however, the intersection of $J_i$ and $J_j$ is not necessarily empty. Now, lets $Y_i := \{y_k\}_{k \in I_i}$, $H_i := H(I_i, J_i)$ and $X_i := \{x_k\}_{k \in J_i}$. Accordingly, the original problem Eq.(1) is divided into $L$ sub-problems as $Y_i = H_i X_i$ (see Eq.(2)) and the $i^{th}$ local estimate is computed by solving this sub-problem.

Since many UDLI problems in networking, communication and signal processing are formulated as Least Norm Estimation (LNE) problems and to develop the basic theory

of MDFE, here, we consider the unconstrained least norm minimization as our sub-space estimation technique. Hence, it is assumed that input vector $X$ does not include unusual inputs that differ in size by large order of magnitudes [18]. Accordingly, in the global case, the LNE is computed using the pseudoinverse of $H$ (denoted by $H^{\dagger}$) which can accurately obtained using the Singular Value Decomposition (SVD) with computation complexity $O(mn^2)$ flops. Also, the $i^{th}$ local LNE is computed using the pseudoinverse of $H_i$ (an $m_i \times n_i$ matrix) with complexity $O(m_i n_i^2)$ flops (see Eq.(5)). Note that, the solution of global and local problems (i.e. Eq.(4) and Eq.(5)) could be different because the null space of $H$ and $H_i$ are not necessarily equal.

$$Y = HX \Rightarrow \hat{X} = H^{\dagger}Y = (H^T(HH^T)^{-1})Y \tag{4}$$

$$Y_i = H_i X_i \Rightarrow \hat{X}_i = H_i^{\dagger} Y_i = H_i^T (H_i H_i^T)^{-1} Y_i \tag{5}$$

### *B. MDFE in Practice: Partition Design*

The accuracy of redundant estimates from sub-spaces depends on the design of partition $P$ that can be formulated as an integer optimization problem to achieve the best possible performance. Assuming there are $m$ measurements and $L$ sub-spaces, then there are $S_{m,L} = \frac{1}{L!}\sum_{j=0}^{L}(-1)^{L-j}\binom{L}{j}j^m$ partitions, where $S_{m,L}$ denotes Stirling number of the second kind. The number of partitions with $K$ elements in each subset (where $K \propto \frac{m}{L}$) is a fraction of Stirling number $S_{m,L}$ that is still a large number in large-scale networks, where $m >> L$. To simplify this NP-hard problem and maximize the MDFE performance, here, pseudo-optimal or heuristics partitioning algorithms are developed. Note that in these algorithms, $L$ is a design parameter and it is assumed to be known a-prior. Among these, Alg.1 and Alg.2 are more suitable in the centralized implementation of MDFE where all network measurements are available at the CFC, and Alg.3 is suitable for the distributed implementation of MDFE where measurements are collected by distributed nodes spread among the network, which compute local descriptions and transmit them to the CFC for the final reconstruction through fusion process. For further details of portioning algorithms, please refer to [10].

In the Greedy CN based Partitioning algorithm (Alg.1), the effectiveness of sub-spaces are sequentially measured and maximized to form partition $P$. Here, the criterion used to evaluate the partition choice is the Condition Number (CN) of the observation matrix $H_i$, denoted by $\kappa(H_i)$. The condition number is defined as the ratio of the maximum and minimum singular values of the matrix $H_i$ and it is an indication of the quality of a matrix which determines a bound ($CN \geq 1$) on the rate at which the solution will change with respect to a change in measurements. The lower the CN is, the more well-conditioned problem and the more accurate solution are. This fact has been proved in Proposition 2 and, also, justified in Section IV-D.

In Alg.1, the best sub-spaces are sequentially chosen to get the best possible partition $P = \bigcup_{i=1}^{L} I_i$ with the lowest CN, which can provide a well behaved partition $P$, and a more accurate and stable solution in each sub-space. This algorithm starts from the first row of $H$ and sequentially chooses the row that minimize the CN of the sub-matrix. This continues to complete the first sub-space $I_1$ with $K$ rows. After removing these $K$ rows from $H$, the algorithm repeats from the beginning. Consequently, after constructing the sub-spaces using Alg.1, the sub-problems can be solved in parallel or sequentially. Note that the CN of each individual row of $H$ is one; however, this is not an interesting case because: 1) in practice, the number of processors/sub-spaces ($L$) are limited (in parallel case), and 2) large $L$'s reduces processing gain $\Delta_s$ (in sequential case). Also, as we have shown in [10], the computational complexity of this algorithm is low. Thus, for a large-scale NI problem which is inefficient or impossible to be solved in a centralized manner, the MDFE approach with partitioning Alg.1 proposes an efficient framework, with manageable computational complexity and without compromising the accuracy of the solution.

In the QRP based Partitioning algorithm (Alg.2), partition $P$ is designed using the structure of the observation matrix $H$, captured by QR decomposition of $H$, where $H$ is:

$$H = QR = Q_{m\times m}\begin{bmatrix} R^{11}_{r\times r} & R^{12}_{r\times(n-r)} \end{bmatrix} \tag{6}$$

with orthonormal matrix $Q$, upper-triangular matrix $R^{11}$, and $rank(H) = r(= m)$. For rank deficient matricies, QR decomposition with pivoting, known as QRP, is used to solve linear system of equations and recognize singularities or rank deficiency. Here, the pivoting strategy attempts to produce $R^{11}$ as well-conditioned as possible. Accordingly, the diagonal elements of $|R|$ occur in decreasing order, revealing the linear in/dependence among the rows of H [19]. In Alg.2 diagonal elements of matrix $|R|$ are grouped to construct initial partition $P_0$ where each batch consists of a set of indicies of successive diagonal entries of matrix $|R|$. Initial Partition $P_0$ is then modified, by extending or shrinking the boundaries of sets $\{I_i\}_{i=1}^{L}$, to improve the performance of MDFE and achieve a pseudo-optimum partition $P$. Using this algorithm, the best possible partition can be achieved through a trial-and-error process and using a learning-set of inputs $X$ to evaluate the performance in each step. In this algorithm, observation matrix $H$ is assumed to be full row-rank. Thus, rows corresponding to small values of $diag(|R|)$ are removed.

The performance of the MDFE process using the partitions designed by these two algorithms are *close* to the optimal obtained by exhaustive search among all possible partitions. To show this, lets consider a small part of the network shown in Figure 2 with the first 18 ($= m$) link load measurements. The optimal partition can be found through exhaustive search among all $M = 17153136$ partitions with $L = 3$ subsets and $K = 6$ elements in each subset. Table I shows this comparison where the performance of pseudo-optimal partitioning algorithms is close to optimal by 0.05% and 3%, respectively. In this table, $Gain_{L2}$ (defined in Table II) quantifies the performance improvement using MDFE framework comparing with the global estimation case.

The third algorithm, known as Graph based Partitioning

| Partitioning Technique | Optimal | Alg.1 | Alg.2 |
|---|---|---|---|
| $Gain_{L2}(\%)$ | 9.2530 | 8.7072 | 6.1654 |

*TABLE I:* The comparison between optimal and pseudo-optimal partitioning algorithms where $X \sim U(100, 500)$.

**Algorithm 1** : Greedy CN based Partitioning

Initialization: $I = \{1, ..., m\}$ and $i = 1$
**while** $i \leq L$ **do**
- Construct $I_i$ by sequentially choosing $K$ rows of $H$ with lowest CN
- Set $I = I \backslash I_i$ and $i = i + 1$
**end while**

algorithm (Alg.3), is a heuristic partitioning algorithm that uses the topology of the network where $L$ nodes with highest degrees are selected as clustering nodes. Observations measured at clustering nodes along with measurements that can be transferred to these nodes with minimum cost (e.g. communication cost & delay) form a partition of the set of measurements $I$. This heuristic partitioning algorithm is important where the *nature* of the estimation problem is distributed and communication costs and/or delays must be considered in the implementation of MDFE framework.

### C. MDFE in Practice: Fusion Algorithm

MDFE process is completed by applying fusion process $F$ to the local estimates at the CFC where all local descriptions are available. Here in Eq.(7), three different weighting functions are defined for the fusion operator in Eq.(3). Let ICN and RoD denote the Inverse of Condition Number and Rank over Dimension (i.e. # of unknowns in each sub-space), respectively. By applying fusion operator $\oplus$, unknowns observed in different sup-spaces are combined to produce the final estimate $\hat{X}_P^F$. The first two fusion functions choose $x_i$ from the sub-space with highest $ICN$ or $RoD$, while the third one computes the average of the observed $x_i$'s produced by different sub-spaces. This averaging process, by itself, can improve the accuracy of the estimation by increasing the Signal-to-Noise Ratio (SNR). These fusion techniques are also efficient because the computation overhead of using these fusion methods are negligible compared with the computation time of sub-space estimation techniques, especially for large-scale problems.

$$\omega_{ij}^{ICN} = \begin{cases} 1, & \text{if } i \in \text{sub-space with highest ICN and } j \in J_i \\ 0, & ow \end{cases}$$
$$\omega_{ij}^{RoD} = \begin{cases} 1, & \text{if } i \in \text{sub-space with highest RoD and } j \in J_i \\ 0, & ow \end{cases} \tag{7}$$
$$\omega_{ij}^{Avg} = \left( \frac{1}{\text{\# of repetition of } X_j \text{ among all sub-spaces}} \right)$$
$$\text{for } i = 1, ..., L \text{ and } j = 1, ..., n$$

**Algorithm 2** : QRP based Partitioning

Initialization:
- Compute the QRP factorization of routing matrix $H$
- Divide $|R|$ into $L$ batches with almost similar successive values
- Construct $P_0$ as rows of $H$ corresponding to indicies of $|R|$ in each batch
- Set $i = 1$
**while** $i \leq L$ **do**
- Modify (i.e. extend or shrink) the boundaries of set $I_i$
- Check the performance on the training-set, and repeat this process until the highest possible gain is achieved
- $i = i + 1$
**end while**

**Algorithm 3** : Graph based Partitioning

Initialization:
- $I = \{1, ..., m\}$, $i = j = 1$, and $I_S = \emptyset$
- Choose $L$ nodes with highest degree as clustering nodes
**while** $i \leq L$ **do**
- $I_i = \{$the index of observations measured at $i^{th}$ cluster node$\}$
- $I_S = I_S \bigcup I_i$ and $i = i + 1$
**end while**
- Set $I_{\bar{S}} = I \backslash I_S$ and compute the cost of transferring all measurements in $I_{\bar{S}}$ to all $L$ clustering nodes
**while** $j \leq |I_{\bar{S}}|$ **do**
- For each measurement $y_j$ (in $I_{\bar{S}}$) find the clustering node with minimum transferring cost (indicated by $i$)
- $I_i = I_i \bigcup \{j\}$ and $j = j + 1$
**end while**

### D. The Efficiency of MDFE

MDFE is an efficient framework that can improve the performance of system from different perspectives. In fact, MDFE not only reduces the required processing time/power, but also provides better estimates in most observed cases, and can improve the robustness of the system against noise and failures.

MDFE is able to provide more accurate estimates due to two factors. First, partitioning increases the redundancy between descriptions, produced by observing each unknown $x_j$ from different sub-spaces (Figures 1(b)). This redundancy is used by the fusion process to improve the accuracy of estimation. The amount of redundancy depends on the number of subsets ($L$) in partition $P$ and the structure of observation matrix $H$. This redundancy is evaluated by two measures: a) the sum of the Number of Unknowns (NoU) observed by different subspaces ($Rdn1$) and b) the sum of the ratio $RoD_i := \frac{rank(H_i)}{n_i}$, representing the contribution of each independent measurement into the estimation of each unknown $x_j$, as shown below:

$$Rdn1 = \sum_{i=1}^{L} NoU_i \quad \& \quad Rdn2 = \sum_{i=1}^{L} RoD_i \tag{8}$$

Second, partitioning does not change the input-output relationship. However, the row partitioning of observation matrix $H$ improves the CN of $H_i$'s denoted by $\kappa(H_i)$(Proposition 1). Therefore, the computational accuracy and stability of estimation procedures in sub-spaces are enhanced. In fact, in [20] it has been shown that the forward and backward errors of least

norm estimates $\hat{X} = H^{\dagger}Y$ are proportionally bounded by the CN of the observation matrix (i.e. $\kappa(H)$) via coefficients $\alpha^F_{mn}$ and $\alpha^B_{mn}$, respectively. Eq.(9) represents these bounds where $u$ controls the row-wise backward error; also, $B$ is the perturbed $X$, satisfying $(H + \Delta H)B = Y + \Delta Y$, and $\epsilon$ appropriately controls the amount of perturbations in $\Delta H$ and $\Delta Y$. In this equation $\alpha^F_{mn}$ and $\alpha^B_{mn}$ are increasing functions [1] of $m$ (number of observations) and $n$ (number of unknowns); please see [20] for further details. Therefore, in comparing the global and MDFE cases, the MDFE framework can significantly reduces the first term on the RHSs of Eq.(9) because the number of observations/unknowns and the condition number of sub-spaces are lower than the global case, that is, $\{m_i\}^L_{i=} \leq m$, $\{n_i\}^L_{i=} \leq n$ and $\{\kappa(H_i)\}^L_{i=} \leq \kappa(H)$ (e.g. see Table V). Therefore, for small $u$ and $\epsilon$, the dominant factors in Eq.(9) are the first terms which remarkably provide lower upper bounds for forward and backward errors. Hence, the MDFE framework can potentially enhance the forward and backward stabilities of the computation of minimum-norm estimates in sub-spaces, resulting in more robust local descriptions.

$$\begin{aligned} &\frac{\left\|X - \hat{X}\right\|_2}{\|X\|_2} \leq \alpha^F_{mn}\kappa(H) + O(u^2) \quad \text{Forward Error Bound} \\ &\frac{\left\|X - \hat{B}\right\|_2}{\|X\|_2} \leq \alpha^B_{mn}\epsilon\kappa(H) + O(\epsilon^2) \quad \text{Backward Error Bound} \end{aligned} \tag{9}$$

*Proposition 1:* Let $H$ be a matrix in ($R^{m\times n}$ with rank $m$) and $H_i$ denotes a matrix constructed from a set of rows of $H$ where $H_i \in R^{m_i\times n_i}$ and $1 \leq m_i < m, 1 \leq n_i \leq n$ . Then: $\kappa(H_i) \leq \kappa(H)$ (For proof, please refer to supplementary Appendix A or [10]).

In addition, in the presence of noisy observations, sub-spaces with lower CN's reduce the variance of error. This has been shown in part (a) of Proposition 2 where , as in [8], it is assumed that link load measurements are contaminated with Gaussian noise due to disalignment of polling intervals. In part (b) of Proposition 2, using a simple model for the covariance matrix of the traffic matrix $X$ (defined as $C_X = E[XX^T]$), we also show that Total Error Variance (TEV) can be reduced in sub-spaces since in UDLI problems the number of unknowns are larger than the number of measurements. In this proposition we also proved that MDFE framework is potentially able to attain lower Mean-Square Error (MSE) because the lower bound on MSE is reduced by partitioning. This fact was also verified through our direct investigation where we observed a positive correlation between the performance of MDFE and the CN of sub-spaces (see Section IV-D). Such an additive Gaussian noise model is important in many other applications where MDFE can be applied to improve the performance.

Moreover, since MDFE can provide more redundant and accurate estimates; it can also improve the robustness of the system against noise, failure and information-loss, in the computing and monitoring infrastructures.

[1]In [20], and mainly in chapter 21, coefficients $\alpha^F_{mn}$ and $\alpha^B_{mn}$ are computed as $\alpha^F_{mn} = min\{3, n-m+2\}\sqrt{mn}$ and $\alpha^B_{mn} = m\frac{cnu}{1-cnu}$ where $c$ is a small integer constant. These coefficients are increasing functions of $m$ and $n$ as we have shown in [10].

*Proposition 2:* Let $Y = HX + \epsilon$ where $\epsilon \sim \mathcal{N}(0, \sigma_0^2 I_m)$ denotes measurement noise. Then: 1) $Var(\mathcal{E}) \propto CN(H)$ where $\mathcal{E} := X - LNE(X) = X - H^{\dagger}Y$; 2) assuming $C_X = \sigma_X^2 I$ , if $(n_i - m_i) \leq (n - m)$ then total error variance is reduced by partitioning and 3) the lower-bound for the MSE of LNE is reduced by partitioning (For proof, please refer to supplementary Appendix A or [10]).

Besides improving accuracy, MDFE can also reduce required processing time, significantly. This is achieved by reducing the dimension of the problem in each sub-space. Considering the complexity $O(m_i n_i^2)$ flops for LNE in each sub-space, since $m_i < m$ and $n_i < n$, local inference problems can be solved more efficiently. Using parallel computing infrastructures, the processing time can be bounded by maximum local processing time. However, using sequential computing infrastructures, reduction in processing time can be achieved if the sum of local processing times is less than the global processing time. In this case, the number of sub-spaces must be carefully chosen. This remarkable gain can be easily achieved in today's distributed multiprocessor or multicore computing infrastructures where communication delays are negligible in comparison with processing times. In these computing systems, MDFE can also enhance memory usage efficiency by distributing a large-scale problem among multiple processors with local memories and reducing memory access times. Note that, in large-scale problems even storing an observation matrix is difficult or sometimes infeasible. Considering the fact that the required processing power is also proportional with the computational complexity of the problem, the same argument can be used to show that, based on the number of sub-spaces, MDFE can also be a power efficient framework where the sum of local required processing powers is less than global required processing power.

In the MDFE framework, the number of sub-spaces can be chosen with a reasonable balance between improvement in desirable and feasible computation time and estimation accuracy. To clarify this, lets consider the Total Processing Times (TPT) of the global and MDFE cases as defined in Eq.(10) where $GPT$ (global processing time) and $LPT_i$ (the $i^{th}$ local processing time) along with processing gains $\Delta_p$ and $\Delta_s$ are defined in Table II in Section IV. Furthermore, in equation Eq.(10), $\mathcal{T}_0$ is running time of the system or equivalently traffic/input duration (see Table III), $T_t^Y$ is measurement collection/transmission time to the CFC in the global case, $T_{Par}$ is the partitioning time, $T_t^{Y_i}$ is measurement collection/distribution time at the $i^{th}$ local processor, $T_t^{X_i}$ is the transmission time of the $i^{th}$ local description to CFC, and $T_F$ is fusion processing time. Among these, in large-scale networks $\{LPT_i\}^L_{i=1} << GPT$ (e.g. for our moderate size illustrative example in Section IV-C, $GPT$ is in the order of multiple milliseconds and $LPT$s are small fractions of a millisecond). Also, $T_{Par}$ is the time which is considered only once, and it is different for different partitioning algorithms as $\left(T^{Alg.2}_{Par} \text{ and } T^{Alg.3}_{Par}\right) < T^{Alg.1}_{Par}$. Moreover, $T_F$ is negligible compared with $LPT$s, for instance in our illustrative example in Section IV-C, $T_F$ is 13 times smaller than the minimum LPT. Note that $T_F \approx 0$ for both $ICN$ and $RoD$ based fusion

techniques since the estimated unknown can be selected from a pre-determined sub-space. Moreover, in applications such as TM estimation where measurement intervals are in the order of multiple minutes (e.g. 5 min or 15 min according to Table III), then partitioning and fusion times are insignificant.

Therefore, if the global NI problem can not be efficiently solved in a global manner and using all measurements, then MDFE framework proposes an alternate approach that is practical. For this purpose, MDFE breaks down the computational complexity into a sequence of processes which can be *eventually* solved. In the centralized case, where all measurements are available at the CFC, then Alg. 1 and Alg. 2 can be used to partition the problem into multiple sub-problems. In this case, $T_t^Y = 0$, and $\left\{T_t^{Y_i}\right\}_{i=1}^{L}$ and $\left\{T_t^{X_i}\right\}_{i=1}^{L}$ are negligible compared with $LPT$s. Thus, when $GPT$ and input duration $\mathcal{T}_0$ are large, and also, $T_{Par}$ is negligible compared with $\mathcal{T}_0$, then $TPT^G$ and $TPT^{MDFE}$ in Eq.(10) can be respectively approximated by only considering $GPT$ and $LPT$s, as the main factors. On the other hand, in the distributed case Alg. 3 provides a very efficient way (with small $T_{Par}$) to partition the problem and distribute the measurements among local clusters that solve the sub-problems. In this case, measurements are locally collected among the network and it is reasonably expected that $\left\{T_t^{Y_i}\right\}_{i=1}^{L} \leq T_t^Y$. In addition, when $\mathcal{T}_0$ and the $GPT$ of solving a large-scale NI problem are large, the contribution of $T_t^{Y_i}$s and $T_t^{X_i}$s in $TPT^{MDFE}$ can be reasonably ignored and $TPT^G$ and $TPT^{MDFE}$ in Eq.(10) can be respectively approximated by only considering $GPT$ and $LPT$s. Note that, part of $T_t^{X_i}$s have been compensated with the reduction in $T_t^{Y_i}$s and $T_F$ is also very small. Therefore, in both centralized and distributed cases $\Delta_p$ and $\Delta_s$ can correctly represent processing gains achieved using MDFE (please refer to [10] for more discussion).

$$\begin{aligned} TPT^G &= \sum_{t=1}^{\mathcal{T}_0}\left(GPT + T_t^Y\right) \\ TPT^{MDFE} &= T_{Par} + \sum_{t=1}^{\mathcal{T}_0}\left(\sum_{i=1}^{L}\left(T_t^{Y_i} + LPT_i + T_t^{X_i}\right) + T_F\right) \end{aligned} \tag{10}$$

Accordingly, since the global and local processing times are proportional with the computational complexity of the underlying NI method, one possible approach for estimating the number of sub-spaces can be achieved by replacing $GPT$ and $LPT$s with corresponding computational complexities. Thus, given a feasible and desirable processing gain $\Delta_p$ and assuming the complexity of SVD as the main factor in computing LNE, then the number of sub-spaces can be estimated using Eq.(11) where $\hat{n}_s^2$ can be approximated as the number of unknowns in $\lceil \frac{m}{L} \rceil$ rows of $H$ with the highest number of unknowns. This is the worst case design scenario as the maximum computational complexity with minimum nominal $RoD$ have been considered. For example, targeting $\Delta_p = 60\%$, $70\%$, $80\%$ for the network shown in Figure 2, $\hat{L}$ is respectively estimated as $\hat{L} = 3$, 5, 8 which is close to our results in Figure 8. Having an estimate for the number of sub-spaces, the real value of $L$ can be selected by trading-off between estimation accuracy and the processing gain using a trial and error process and by considering all practical constraints and possible additional costs in distributed systems (e.g. communication and deployment costs). Note that, in MDFE since the set of measurements is *partitioned* (i.e. there is no redundancy between measurements in sub-spaces), then the large increasing of $L$, first, leads to reduction in the number of measurements for each sub-space and this may affect both sub-space and ultimate estimation accuracies. And second, by increasing $L$ the marginal gain in local processing time diminishes and the improvement in $\Delta_p$ eventually tapers off as it is shown in Figure 8.

$$\hat{L} = \underset{L}{min}\left|\Delta_p - 100 \times \frac{mn^2 - \lceil \frac{m}{L} \rceil \hat{n}_s^2}{mn^2}\right| \tag{11}$$

## IV. Performance Evaluation of MDFE

In this section we define the metrics that we have used to evaluate the performance of MDFE framework; furthermore, we introduce the networks and data sets that we have considered for our study. In addition, we show the details of the efficiency of the MDFE process in solving UDLI problems by considering different illustrative examples.

### A. MDFE: Performance Evaluation Metrics

The performance of the MDFE is evaluated using various criteria which are introduced in Table II. In this table, $\hat{X_t^G}$ denotes the $t^{th}$ global estimate, and $\hat{X_{w_t}^F}$ denotes the $t^{th}$ MDFE estimate where $w$ denotes the type of fusion function in Eq.(7). Furthermore, $LE_i$ denotes the estimation error for the sub-set of unknowns observed and resolved at $i^{th}$ sub-space, and $GLE_i$ measures the error of global estimates for the sub-set of unknowns observed at $i^{th}$ sub-space. Also, $Gain_{L2}$ quantifies the performance improvement using MDFE framework comparing with global estimation case. For the sake of simplicity, throughout this paper, the subscript $t$ has been dropped from unknown vector $X_t$, and its global and MDFE estimates.

Parallel and sequential processing gains ($\Delta_p$ and $\Delta_s$) measure the reduction in computation time using MDFE structure where the $GPT$ and $LPT$ directly measure the average internal time of the execution of each inference method in a Monte-Carlo simulation and on an Intel Core-i7 platform. The, sequential processing gain can also be an indication of the reduction in required processing power using MDFE.

### B. Networks and Data Sets Under Study

To evaluate the performance of our framework, here, three different networks are considered, including: 14-Node Tier-1 PoP Topology (Figure 2), Abilene [21] and GEANT [22] networks. The routing matrix $H$ of the first network is a $(50\times182)$ matrix with density $D = 0.0415$ ($D = \frac{\#ofnon-zeroentries}{m\times n}$). $H_{Abilene}$ is a $(30 \times 144)$ matrix with density $D = 0.0353$ and $H_{Geant}$ is a $(74 \times 529)$ matrix with density $D = 0.036$ ($H_{Geant}$ a minimum hop routing matrix). All routing matrices are binary and full row-rank.

In addition, synthetic inputs are generated using three different distributions [3]: 1) Uniform distribution where $x_i \sim$

| |
|---|
| $GE = \frac{1}{\mathcal{T}_0}\sum_{t=1}^{\mathcal{T}_0}\frac{\left\|X_t-\hat{X}_t^G\right\|_2}{\left\|X_t\right\|_2}$ |
| $FE_w = \frac{1}{\mathcal{T}_0}\sum_{t=1}^{\mathcal{T}_0}\frac{\left\|X_t-\hat{X}_{w_t}^F\right\|_2}{\left\|X_t\right\|_2}$ |
| $Gain_{L2} = 100 \times \frac{GE-FE_{Avg}}{GE}$ |
| $\Delta_p = 100 \times \frac{GPT-max(\{LPT_i\}_{i=1}^L)}{GPT}$ |
| $\Delta_s = 100 \times \frac{GPT-\sum_{i=1}^L LPT_i}{GPT}$ |
| $GLE_i = \frac{1}{\mathcal{T}_0}\sum_{t=1}^{\mathcal{T}_0}\frac{\left\|X_t\{k\}_{k\in J_i}-\hat{X}_t^G\{k\}_{k\in J_i}\right\|_2}{\left\|X_t\{k\}_{k\in J_i}\right\|_2}$ |
| $LE_i = \frac{1}{\mathcal{T}_0}\sum_{t=1}^{\mathcal{T}_0}\frac{\left\|X_t\{k\}_{k\in J_i}-\hat{X}_{it}^L\right\|_2}{\left\|X_t\{k\}_{k\in J_i}\right\|_2}$ |

*TABLE II:* Performance evaluation criteria where $t$ denotes the $t^{th}$ time interval, $X_t$ denotes the $t^{th}$ traffic matrix, $w$ denotes the type of fusion function in Eq.(7) (that is, $ICN$, $RoD$ or $Avg$) , and $GPT$ and $LPT_i$ denote the Global Processing Time and the $i^{th}$ Local Processing Time, respectively.

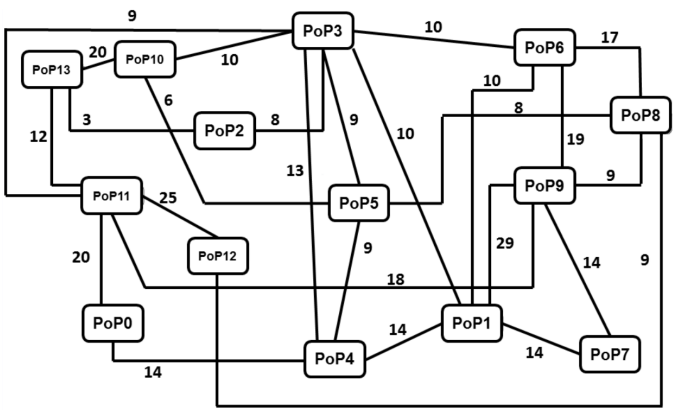



*Fig. 2:* 14-Node Tier-1 POP Topology [3].

$U(100, 500)$; 2) Gaussian distribution where $x_i \sim \mathcal{N}(\mu_i, 40)$ (and $\mu_i \sim U[100, 500]$ ), and 3) Poisson distribution where $x_i \sim Pois(\lambda_i)$, $\lambda_i \sim U[100, 500]$. These synthetic data are used to evaluate the performance of TME on the first network (Figure 2), and likewise, real network traffic traces (Table III) are used on Abilene and GEANT networks. Note that, although we use synthetic data for TME in networking, inputs with Uniform, Gaussian and Poisson distributions are appeared in many applications in communication, signal processing and control, where MDFE can be applied to improve the performance.

### C. Illustrative Case Studies

In the first example, consider UDLI problem $Y = HX$ where $Y = [y_1, y_2]^T$, $H_{2\times 3} = [1, 1, 0; 0, 1, 1]$ and true $X$ is $X = [x_1, x_2, x_3]^T = [1, 1, 1]^T$ and $Y = [2, 2]^T$. Assume we partition this problem into two sub-problems defined as $y_1 = H_1X_1 = [1, 1][x_1, x_2]^T$ and $y_2 = H_2X_2 = [1, 1][x_2, x_3]^T$ where $CN(H) = 1.73$, $CN(H_1) = CN(H_2) = 1$ and $RoD_1 = RoD_2 = \frac{1}{2}$. In this example, the global LNE is computed using Eq.(4) as $\hat{X}^G = [\frac{2}{3}, \frac{4}{3}, \frac{2}{3}]^T$ and local LNEs are computed using Eq.(5) as $\hat{X}_1 = [1, 1, 0]^T$ and $\hat{X}_2 = [0, 1, 1]^T$. Note that, the global solution $\hat{X}^G$ of original problem, and local estimates $\hat{X}_1$ and $\hat{X}_2$ may be different as $\mathcal{N}(H)$, $\mathcal{N}(H_1)$, and $\mathcal{N}(H_2)$ are different. By fusing local descriptions using Eq.(3), the MDFE estimates are computed as $\hat{X}_{ICN}^F = \hat{X}_{RoD}^F = [\hat{x}_1(1), \hat{x}_i(2), \hat{x}_2(3)]^T = [1, 1, 1]^T$ (where $i$ denotes the subspace with maximum $ICN$ and/or $RoD$) or $\hat{X}_{Avg}^F = [\hat{x}_1(1), \frac{\hat{x}_1(2)+\hat{x}_2(2)}{2}, \hat{x}_2(3)]^T = [1, \frac{1+1}{2}, 1]^T = [1, 1, 1]^T$, and accordingly, $GE = 0.3333$ and $FE_{ICN} = FE_{RoD} = FE_{Avg} = 0$.

| Network | Date | Duration | Resolution | TM Size ($n \times \mathcal{T}_0$) |
|---|---|---|---|---|
| Abilene | 2004-05-01 | 1 week | 5 min. | 144 × 2016 |
| GEANT | 2005-01-08 | 1 week | 15 min. | 529 × 672 |

*TABLE III:* Real Datasets under study.

| SS1 | SS2 | SS3 | SS4 | SS5 |
|---|---|---|---|---|
| (i,O,D) | (i,O,D) | (i,O,D) | (i,O,D) | (i,O,D) |
| (1, 6, 3) | (8, 3, 6) | (12, 13, 2) | (20, 9, 8),(29, 3, 5) | (38, 0, 11),(45, 3, 1) |
| (2, 1, 3) | (9, 3, 2) | (13, 2, 13) | (21, 9, 7),(30, 3, 4) | (39, 3, 11),(46, 4, 1) |
| (3, 5, 3) | (10, 9, 6) | (14, 13, 10) | (22, 11, 12),(31, 3, 10) | (40, 13, 11),(47, 1, 6) |
| (4, 2, 3) | (11, 11, 13) | (15, 10, 13) | (23, 6, 8),(32, 5, 8) | (41, 11, 0),(48, 8, 6) |
| (5, 4, 3) | | (16, 6, 9) | (24, 8, 12),(33, 5, 10) | (42, 4, 0),(49, 0, 4) |
| (6, 10, 3) | | (17, 8, 9) | (25, 12, 8),(34, 5, 4) | (43, 7, 1),(50, 1, 7) |
| (7, 11, 3) | | (18, 7, 9) | (26, 8, 5),(35, 1, 4) | (44, 6, 1) |
| | | (19, 11, 9) | (27, 4, 5),(36, 12, 11) | |
| | | | (28, 10, 5),(37, 9, 11) | |

*TABLE IV:* The Origin-Destination relation of link counts in different sub-spaces (*(i,O,D)* denotes (index,Origin,Destination).

In the second example, we consider the network shown in Figure 2 where link count vector $Y$ is produced by Eq.(1) where $X$ is the input vector with different distributions. This problem is partitioned into $L = 5$ sub-problems using Alg. 2 where disjoint observation sets $\{I_i\}_{i=1}^5$ are defined as $I_1 = \{1, ..., 7\}$, $I_2 = \{8, ..., 11\}$, $I_3 = \{12, ..., 19\}$, $I_4 = \{20, ..., 37\}$, $I_5 = \{38, ..., 50\}$. Table IV indicates the mapping of the various origin-destination and link counts in different Sub-Spaces (SS), as compared to the original Global Space (GS). Having this partition and the routing matrix $H$, the corresponding sets $\{J_i\}_{i=1}^L$ are easily found. Similar to [3], the routing matrix $H$ can be computed using the link weights in Fig.2 and by running the Dijkstra's algorithm.

Table V provides the main characteristics of the original problem and the sub-spaces. The $ICN$ of each sub-space is higher than the $ICN$ of the original global problem. The total redundancy using this partition is improved, that is, $Rdn1 = 313$ which is $\frac{313}{182} = 1.72$ times of the original problem. Also, $Rdn2$ is greater than the $RoD$ of the original global space as $Rdn2 = 0.7812$ and $RoD = 0.2747$.

Figure 3 shows the $L_2$ norm errors for this example, where errors are computed through Monte-Carlo simulations (with $\mathcal{T}_0 = 10000$ runs) using uniformly distributed TM entries. It is clear that MDFE improves the accuracy of the global TM estimation. This Figure plots the average fusion error $FE_{Avg}$ (across all runs), which is lower than average $GE$ error. Some of the local estimates (before applying the fusion process) also achieve lower error than the $GE$. Figure 3 also confirms the fact that the higher improvement in local estimates $LE_i$ is

| Sub-space Features | GS | SS1 | SS2 | SS3 | SS4 | SS5 |
|---|---|---|---|---|---|---|
| # of Link Counts | 50 | 7 | 4 | 8 | 18 | 13 |
| # of OD Flows | 182 | 70 | 32 | 43 | 99 | 69 |
| Rank | 50 | 7 | 4 | 8 | 18 | 13 |
| RoD | 0.2747 | 0.1000 | 0.1250 | 0.1860 | 0.1818 | 0.1884 |
| ICN | 0.1669 | 0.6124 | 0.2425 | 0.2993 | 0.2664 | 0.3399 |
| $(n-m)$ | 132 | 63 | 28 | 35 | 81 | 56 |
| Complexity | O(1656200) | O(34300) | O(4096) | O(14792) | O(176418) | O(61893) |
| $Rdn1$ = 313, $Rdn2$=0.7812 | | | | | | |

*TABLE V:* The features of original Global Space (GS) & Sub-Spaces (SS) where computational complexity is proportional to $O(mn^2)$.

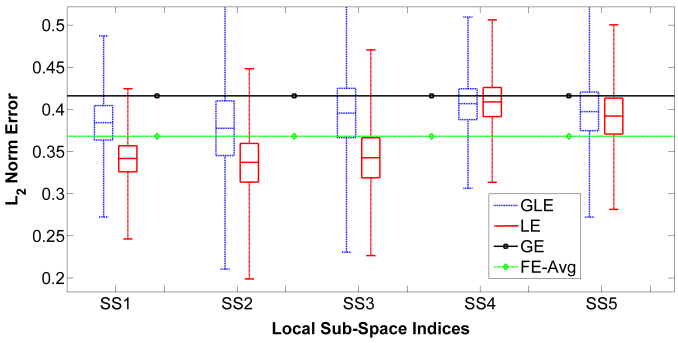



*Fig. 3:* $L_2$ norm error: global vs. local.

achieved in sub-spaces with the lowest $(n_i - m_i)$ (see Table V), confirming our results in Proposition 2.

Table VI summarizes the performance of MDFE for different traffic distributions introduced in Section IV-B. It shows that: 1) the precision of TM estimation is improved for different distributions of the TMs; 2) using MDFE, significant processing gains can be achieved in both parallel and sequential processing methods, respectively indicated by $\Delta_p$ and $\Delta_s$ and 3) MDFE improves the performance of TM estimation not only in average but also in a majority of iterations where Improvement Ratio (IR) indicates the percentage of iterations in which $FE_{Avg}$ is less than $GE$. The remarkable improvement in sequential processing gain matches with what we have shown in Table V where $\sum_{i=1}^{L} O(m_i n_i^2) << O(mn)$. Therefore, the partition in Table IV is able to decrease the required processing time and power, simultaneously.

### D. Correlation of CN and Estimation Accuracy

To investigate the proportionality of the performance of MDFE and ICN, we set $L = 3$ for the network shown in Figure 2 and consider all possible $\binom{N}{L} = \binom{14}{3} = 364$ configurations where Alg. 3 has been used for partitioning. Figure 4 illustrates the positive correlation of ICN and the estimation performance of MDFE in sub-spaces where, in general, a higher ICN results in a better estimation accuracy. In fact, the $i^{th}$ correlation coefficient $\rho_i = \frac{Cov(ICN, RGain^i_{L2})}{\sqrt{Var(ICN)Var(RGain^i_{L2})}}$ for three sub-spaces are $\rho_1 = 0.8979$, $\rho_2 = 0.6954$ and $\rho_3 = 0.7229$ which shows the high correlation between ICN and estimation accuracy. In this figure local relative gains are defined as $RGain^i_{L2} = \frac{GLE_i - LE_i}{GLE_i}$ for $i = 1, ..., L$. It is clear that local relative gains are higher for sub-spaces with higher ICNs.

| Gain | Uniform | Gaussian | Poisson |
|---|---|---|---|
| $Gain_{L2}(\%)$ | 11.5128 | 12.3687 | 12.1626 |
| $\Delta_p(\%)$ | 80.2401 | 80.1866 | 80.3435 |
| $\Delta_s(\%)$ | 53.3655 | 53.7257 | 53.5877 |
| Improvement Ratio(%) | 73 | 82 | 82 |

*TABLE VI:* The performance of MDFE for different distributions.

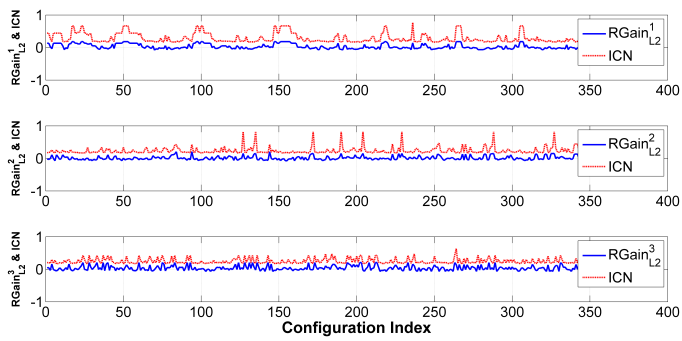



*Fig. 4:* $RGain_{L2}$ for different configurations of the network shown in Figure 2.

## V. Network Inference using MDFE

The main goal in this section is to show the effectiveness of MDFE framework in different applications, including TM estimation, TM completion and loss inference. In fact, we illustrate that MDFE framework is compatible with a variety of existing inference techniques used to solve the UDLI problems. We also show that MDFE is effective for inputs with different distributions and on networks with different topologies. Among these, different partitioning algorithms are used to show the effectiveness of MDFE framework.

### A. Traffic Matrix Estimation

Considering $Y = HX$, TME is an under-determined inference problem where $X$ is the TM (each entry of $X$ represents an ODF in the network) and it is estimated by knowing routing matrix $H$ and observing link load measurement vector $Y$. In the first evaluation, the network shown in Figure 2, is partitioned into $L = 1, ..., 14$ sub-spaces using Alg.3 and synthetic TM inputs are applied to generate $Y$; then MDFE with LNE is used to infer $\hat{X}^F_w$ at each time interval $t$. For each $L$, this process is repeated using a Monte-Carlo simulation.

Figures 5 and 6 show the improvement achieved by applying MDFE for TM estimation on TM inputs with different distribution (as in [3], [11]) where the number of sub-spaces varies from 1-to-14. In these figures, configuration index denotes the number of clustering nodes or equivalently the number of sub-spaces that has been used for partitioning using Alg.3. Table VII shows the clustering nodes for each configuration. In addition, in [10], we have shown the communication cost and delay under each configuration.

Figure 7 shows that the redundancy of observed unknowns is increased as the number of local sub-spaces varies from $L$ =1 to the maximum possible $L$ =14. Also, the ICN of local sub-spaces have been improved as it was proved in Proposition 1; in this figure, different colors represent the

| Configuration Index | Clustering Node(s) in Alg.3 |
|---|---|
| 1 | [3] |
| 2 | [3,9] |
| 3 | [3,9,11] |
| 4 | [3,9,1,11] |
| 5 | [3,9,5,11,1] |
| 6 | [3,9,5,11,1,4] |
| 7 | [3,9,5,11,1,4,8] |
| 8 | [3,9,5,11,1,4,8,6] |
| 9 | [3,9,5,11,1,4,8,6,13] |
| 10 | [3,9,5,11,1,4,8,6,13,10] |
| 11 | [3,9,5,11,1,4,8,6,13,10,2] |
| 12 | [3,9,5,11,1,4,8,6,13,10,2,12] |
| 13 | [3,9,5,11,1,4,8,6,13,10,2,12,0] |
| 14 | [3,9,5,11,1,4,8,6,13,10,2,12,0,7] |

*TABLE VII:* Clustering node(s) for each configuration.

ICN of each sub-space for each configuration. These figures prove the proportionality of MDFE performance with the enhancement of ICN and redundancy of unknowns observed in different sub-spaces. In addition, Figure 5 indicates the performance of different weighting functions (Eq.(7)) in the multiple description fusion process. Among these, computing the mean value of observed ODFs (using $\omega^{Avg}$) have the best performance. However, $RoD$ and $ICN$ based fusion techniques can also achieve good performance while reducing the communication cost in the distributed/decentralized implementation of MDFE. On average, these improvements are almost achieved over 80% of the iterations.

Figure 8 shows that processing gain is significantly improved when the TM estimation problem is distributed among local sub-spaces. Note that, although $\Delta_p$ enhances as $L$ increases, the improvement in $\Delta_p$ eventually tapers off because the marginal gain in local processing time diminishes. Also, this figure indicates that there is an optimum number of sub-spaces (5 in this case) for sequential TM estimation in terms of sequential processing gain $\Delta_s$. This is due to the fact that in real computer systems, in addition to the summation of local processing times, inter-communication times among different processes in computer architectures plays a major role in determining the sequential processing time. This figure along with Figure 5 experimentally show the trade-off between efficiency and accuracy that can be achieved in practice. Accordingly, the number of sub-spaces can be practically chosen using these two figures and by considering all practical constraints and possible additional costs (e.g. communication and deployment costs).

Note that, since the required processing power is a function of the complexity of the algorithm, $\Delta_s$ in Figure 8 also indicates that MDFE can reduce the required processing power. When the number of parallel processors are limited; the performance of MDFE can be increased by the partitioning of each sub-problem into multiple sub-problems where multiple description fusion can be performed in multi-stages at each local node. MDFE framework is also match with the architecture of today's multi-processor computing systems where a large-scale system can be divided into smaller sub-problems solved by each processor. This facilitates the problem of storing a large scale system, and further reduction in processing time is

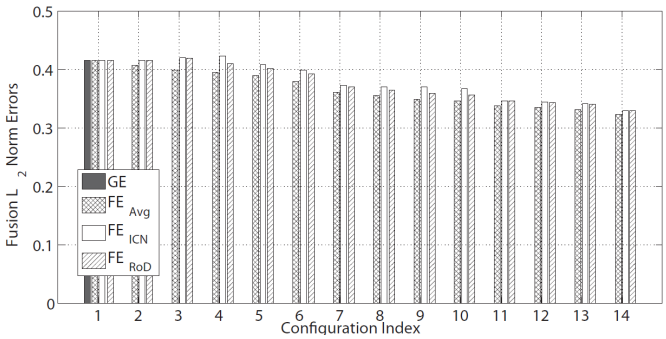



*Fig. 5:* Global&MDFE errors v.s. # of subspaces ($X \sim U(100, 500)$).

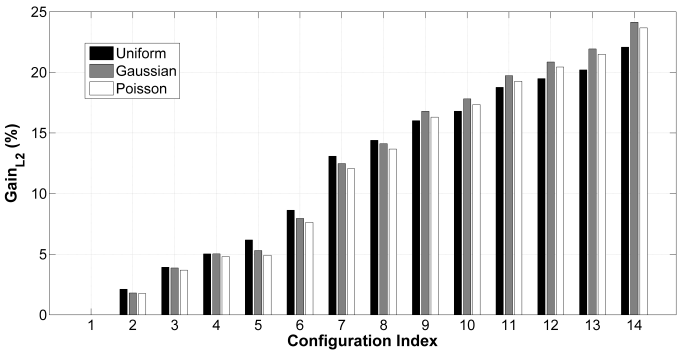



*Fig. 6:* $Gain_{L2}$ for different distributions.

achieved by using local-fast memories.

*1) Robustness of MDFE:* MDFE improves the robustness of the system against noise in link load measurements and lossy informations (due to failures in communication networks and computing infrastructure). According to [8], noise in link load measurements (due to disalignment of polling intervals) can be modeled as a White Gaussian Noise (WGN); therefore, we added WGN to link measurement vector $Y$ with different SNR values to evaluate the performance. Table VIII shows that MDFE is able to achieve better improvement in the presence of noisy link load measurements, even at very low SNR regimes. Our results also indicate that MDFE is robust against sub-space erasure in the system. To increase the robustness, sub-spaces with higher number of observed unknowns (e.g. sub-space 4) must be effectively protected and/or the number of sub-spaces must be increased.

*2) EM Compatibility:* To show the compatibility of MDFE with EM algorithm (as an ML estimator) we implemented TME method in [11] using MDFE framework. Table IX summarizes the estimation gain of MDFE when two different TM estimation methods (LNE and EM) are used. It shows that: 1) MDFE reduces the estimation error in both cases, which implies that MDFE framework is compatible with both TME methods, and 2) using the prior knowledge about the distribution of the TMs improves the accuracy of the MDFE.

*3) Ridge-Regression Compatibility:* To show the compatibility of MDFE with standard L2-regularized estimation or ridge-regression techniques, we set the sub-space estimation technique for both global and local sub-spaces as the optimization framework in Eq.(12). Table X shows the performance of MDFE process in this case, indicating its better estimation

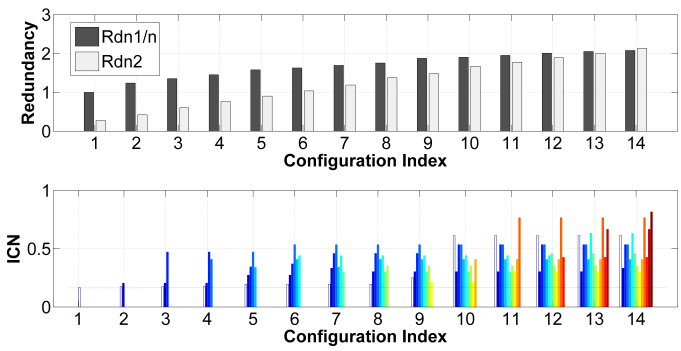



*Fig. 7:* The characteristics of local sub-spaces.

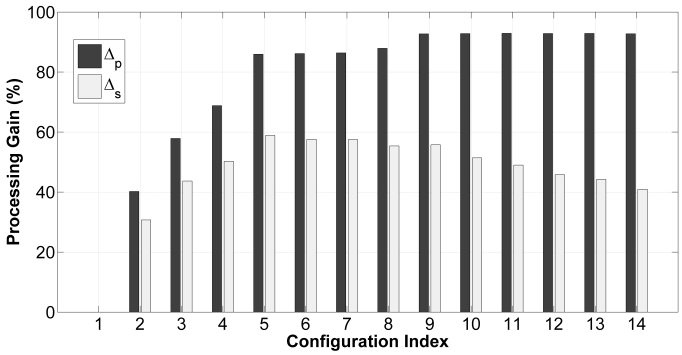



*Fig. 8:* Processing gain v.s. # of sub-spaces.

accuracy. Here, the regularization parameter $\lambda$ is chosen using L-curve criterion for the *global* TM estimation in a supervised setting with a small fraction of the traffic trace.

$$\hat{X} = \underset{X}{\text{minimize}} \, \|Y - HX\|_2^2 + \lambda^2 \|X\|_2^2 \quad \text{s.t. } X \geq 0 \tag{12}$$

*4) Compatibility with Different Sources of Data:* Nowadays, NetFlow records are widely supported by vendors and deployed in most of the operational IP networks. Such partial TM measurements can be used as side information to improve the accuracy of TM estimates. However, real TM measurements and SNMP data are noisy due to sampling and polling processes, respectively [8]. To address these challenges we adopt the TME method in [8] which is formulated as:

$$\hat{X} = X + \epsilon^X \;\; \& \;\; \hat{Y} = HX + \epsilon^Y \;\; \Rightarrow \;\; V = CX + \epsilon \tag{13}$$

where $\hat{X}$ denotes the TM measurement from NetFlow, $\hat{Y}$ denotes SNMP link load measurements, $\epsilon^X$ and $\epsilon^Y$ are respectively Gaussian noises in NetFlow and SNMP records, $V = [\hat{X}; \hat{Y}]$, $C = [I_n; H]$ and $\epsilon = [\epsilon^X; \epsilon^Y]$. Then, having covariance matrix $\mathcal{K} = E[\epsilon\epsilon^T]$; $X$ is estimated by:

$$\hat{X} = \left(C^T \mathcal{K}^{-1} C\right)^{-1} C^T \mathcal{K}^{-1} V \tag{14}$$

To apply our MDFE framework on this setup, Alg.1 is used to partition the network into $L = 8$ sub-spaces. Then, Eq.(14) is properly adapted to solve the problem in each sub-space (where corresponding parameters $C_i$, $\mathcal{K}_i$ and $V_i$ are used based on routing matrix $H_i(I_i, J_i)$ and observations $\hat{Y}_i$ and $\hat{X}_i$ in each sub-space). Accordingly, processing gains $\Delta_p = 88\%$ and $\Delta_p = 58\%$ are achieved, indicating that MDFE speeds up the process, significantly. In addition, Figure 9 shows $Gain_{L2}$ at different SNR values, indicating that MDFE can improve the performance. This gain is remarkable at low NetFlow SNR values (i.e. low sampling rates) where sampling and storing overheads are challenging limitations for direct measurement of TMs. Therefore, MDFE can be utilized to *propose* a new hybrid TM measurement method where important TMs can be measured with higher sampling rates and MDFE is applied on the other TMs to improve the accuracy of TM estimation. This is of particular importance in today's network monitoring systems where sampling and storing a sheer volume of today's traffic and fast TM estimation are challenging problems, particularly for large scale and dynamic environments. Therefore, MDFE is not only compatible with the idea of using multiple sources of data [8], but it can also enhance its performance.

| SNR(dB) | -6 | -3 | 0 | 3 | 6 |
|---|---|---|---|---|---|
| $GE$ | 0.6797 | 0.6668 | 0.6488 | 0.6242 | 0.5928 |
| $FE_{Avg}$ | 0.5823 | 0.5673 | 0.5469 | 0.5212 | 0.4903 |
| Erased Sub-Space | 1 | 2 | 3 | 4 | 5 |
| $GE$ | 0.4295 | 0.4276 | 0.4287 | 0.4861 | 0.4647 |
| $FE_{Avg}$ | 0.4166 | 0.3882 | 0.4078 | 0.5503 | 0.4800 |

*TABLE VIII:* Performance of MDFE in the presence of noise and sub-space erasure (using Alg.2 with $L = 5$ on network in Figure 2).

| LNE ($GE$) | EM ($GE$) | LNE ($FE_{Avg}$) | EM ($FE_{Avg}$) |
|---|---|---|---|
| 0.4126 | 0.3215 | 0.3626 | 0.2956 |

*TABLE IX:* LNE and EM based TM estimation: A comparison (using Alg.2 with $L = 5$ (see Table IV) on network in Figure 2).

To evaluate the performance of this method in the presence of noisy observations on real network traffic, Gaussian noise is added to link load measurements at different *practical* SNR values according to [8]. The first part of Table XI shows that MDFE provides more reliable performance in the presence of noisy observations; note that higher gain is achieved at lower SNRs which shows the ability of MDFE framework in noisy environments. The second part of this table, also, indicates that MDFE is robust against failures in the system. However, this robustness is a function of the sub-space erased from the fusion process in MDFE. Our results showed that MDFE is more sensitive to the erasure of sub-spaces with larger number of ODFs. Note that, comparing to the global scenario, MDFE is more robust against failures because in the centralized global case the presence of failure results in a total loss of information.

### *B. Traffic Matrix Completion*

In [4], a Sparsity Regularized SVD (SRSVD) method is introduced for TM Completion (TMC) where the columns of traffic matrix $Z$ is formed by the unknown vector $X$ in our TME setup at different times ($t = 1, ..., \mathcal{T} \leq \mathcal{T}_0$). Assuming $Z$ can be factored as $Z_{n\times\mathcal{T}} = LR^T$; then, TMC is formulated as the following optimization problem to estimate missed entries of $Z$.

$$\hat{Z} = \underset{L,R}{min} \left\|\mathcal{B} - \mathcal{A}(LR^T)\right\|_F^2 + \lambda \left(\|L\|_F^2 + \|R\|_F^2\right) \tag{15}$$

| Traffic Dist. | $GE$ | $FE_{Avg}$ | $Gain_{L2}$ (%) |
|---|---|---|---|
| Uniform | 0.4174 | 0.3548 | 14.9881 |
| Gaussian | 0.4167 | 0.3544 | 14.9395 |
| Poisson | 0.4202 | 0.3704 | 11.8392 |

*TABLE X:* MDFE performance with L2-regularized estimation technique (using Alg.2 with L=5 on network in Figure 2).

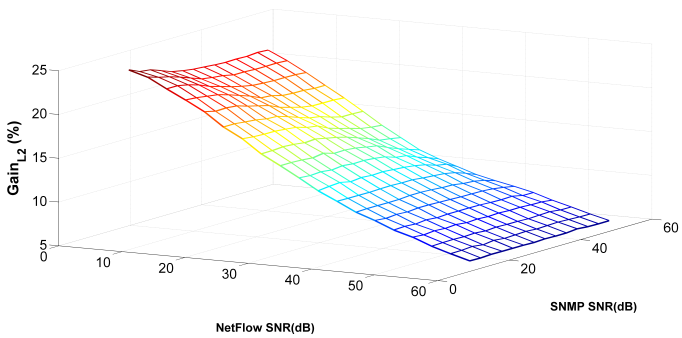



*Fig. 9:* $Gain_{L2}$ vs. SNR in NetFlow and SNMP.

Here, $\mathcal{B}$ and $\mathcal{A}$ respectively denote the set of measurements and a linear operator satisfying $\mathcal{A}(Z) = \mathcal{B}$. To apply our MDFE framework, we adopt this method and modified the formulation in Eq.(15). In our Modified SRSVD (MSRSVD) method, $A_t = [diag(M_t); H]$ and $b_t = [X_t. * M_t; Y_t]$ where $M_t$ is a binary column vector (where zeros represent missing entries), $H$ is the routing matrix, $Y_t = HX_t$ denotes $t^{th}$ link load measurement vector and $.*$ denotes an element-wise product; accordingly, $\mathcal{A} = \{blockdiag(\mathcal{A}, A_t)\}_{t=1}^{\mathcal{T}}$, $\mathcal{B} = [b_1; ...; b_{\mathcal{T}}]$ and $\mathcal{M} = [M_1, ..., M_{\mathcal{T}}]$. Figure 10 shows that our new MSRSVD TM completion method significantly improves the performance where TMC is applied onto normalized TMs where $X_{base}$ [4] is assumed to be known. It also compares the TM completion performance between Global-TMC and MDFE-TMC on real Abilene and GEANT networks and data. Here, Alg.1 is used for partitioning where $L^{Abilene} = 10$ and $L^{GEANT} = 6$ and we set MSRSVD as sub-space TMC technique in MDFE framework. Also, $\lambda_{Abilene} = 0.01$ and $\lambda_{GEANT} = 0.1$ and for both networks we fixed $r = 2$ (i.e. rank-2 approximation). The Normalized Mean Absolute Error (NMAE) is computed over interpolated values as $NMAE = \frac{\sum_{i,j:\mathcal{M}(i,j)=0} |Z(i,j) - \hat{Z}(i,j)|}{\sum_{i,j:\mathcal{M}(i,j)=0} |Z(i,j)|}$. MDFE can improve the estimation performance in the presence of high loss by reducing the number of unknowns in each sub-space and fusing local descriptions. For low loss rates, the performance of both methods are close together. However, the MDFE can speed up the TMC process by reducing the dimension of sub-problems, for example, for Abilene network $(max\left(\{m_i\}_{i=1}^{10}\right) = 3 < 30 = m, max\left(\{n_i\}_{i=1}^{10}\right) = 64 < 144 = n)$, and for GEANT network $(max\left(\{m_i\}_{i=1}^{6}\right) = 14 < 74 = m, max\left(\{n_i\}_{i=1}^{6}\right) = 290 < 529 = n)$.

### C. Link Loss Inference

Considering $Y = HX$, loss inference is also an UDLI problem where $H$ is a routing matrix, and $X$ and $Y$ are

| Network | Abilene | | | GEANT | | |
|---|---|---|---|---|---|---|
| SNR(dB) | 10 | 20 | 40 | 10 | 20 | 40 |
| $FE_{Avg}$ | 0.3771 | 0.3256 | 0.3173 | 0.3234 | 0.3019 | 0.2960 |
| $Gain_{L2}$ (%) | 19 | 10 | 8 | 25 | 21 | 21 |

| Network | Abilene | | GEANT | | | | |
|---|---|---|---|---|---|---|---|
| Erased SS | 1 | 3 | 2 | 4 | 5 | 11 | 12 |
| $GE$ | 0.3534 | 0.3507 | 0.3753 | 0.3747 | 0.3790 | 0.3752 | 0.3747 |
| $FE_{Avg}$ | 0.3343 | 0.3244 | 0.3080 | 0.3011 | 0.3321 | 0.3024 | 0.2957 |

*TABLE XI:* The performance of MDFE on Abilene and GEANT networks in the presence of noise and sub-space erasure where Alg. 2 used to partition both networks into $L = 5$ and $L = 14$ sub-spaces, respectively (see [10] for details). Here, Erased SS denotes the index of the sub-space erased from the MDFE process.

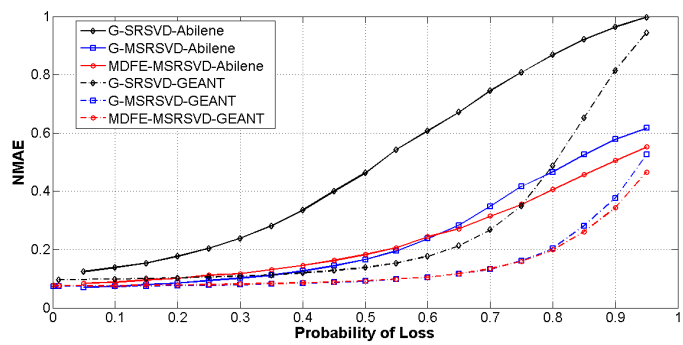



*Fig. 10:* NMAE v.s probability of loss in TM completion (G: global).

defined as $X = \{x_j\}_{j=1}^{n} = \{log\hat{\phi}_{e_j}\}_{j=1}^{n}$ and $Y = \{y_i\}_{i=1}^{m} = \{log\hat{\phi}_i\}_{i=1}^{m}$. Here, $\hat{\phi}_i$ represents the fraction of $S$ probes that arrive correctly at the destination and $\hat{\phi_{e_j}}$ is the fraction of probes from all paths passing through link $e_j$ that have not been dropped by that link [7]. Here, a Loss Inference Algorithm (LIA) is adopted from [7] as the sub-space estimation technique to apply the MDFE framework. Three real network topologies are considered and the proportion of the links that are congested is fixed and is varied to evaluate the performance of MDFE framework in terms of $GE$ and $FE_{Avg}$. Here, congested and non-congested links have loss rates uniformly distributed in [0.05, 0.2] and [0,0.002], respectively. Figure 11 shows the improvement achieved by applying MDFE for loss inference where Alg.2 is used to construct sub-problems; Table IV and [10] provide the details of sub-spaces designed, here. In fact, Figure 11 shows that MDFE is more effective for higher loss-rates.

### D. MDFE with Set-Covering

LNE is not effective in the presence of unusual inputs. Therefore, in many cases, $L_1$ and $L_\infty$ constrained minimization techniques can be effectively applied to UDLI problems with heavy-tailed distributed inputs. These problems are generally solved using numerical optimization techniques. To show the possibility of applying MDFE framework for this set of problems, redundant set $\mathscr{C} = \bigcup_{i=1}^{L} I_i$ is defined to cover set $I$ where $I_i \bigcap I_j$ $(i \neq j)$ is not necessarily empty. In fact, in this case, set-partitioning problem is changed to set-covering problem which is still an NP-hard problem for large-

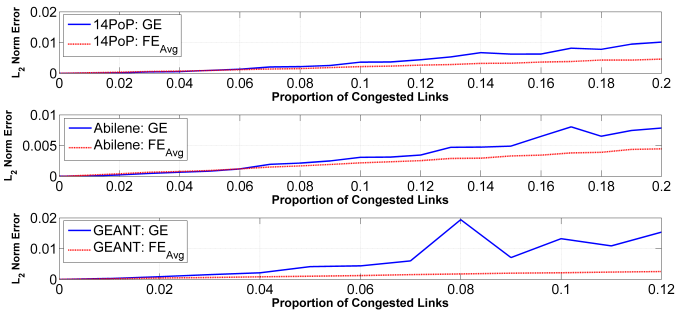



*Fig. 11:* $GE$ and $FE_{Avg}$ vs. the proportion of congested links where $L^{14PoP} = 5$, $L^{Abilene} = 5$, $L^{GEANT} = 14$, # of beacons $S$=1000 and # of iterations = 100.

| Opt. Method | $GE$ | $FE_{Avg}$ | $Gain_{L2}(\%)$ |
|---|---|---|---|
| Eq.(16) | 0.5116 | 0.4800 | 6.1751 |
| Eq.(17) | 0.6093 | 0.5736 | 5.8616 |

*TABLE XII:* MDFE performance using $L_1$ and $L_\infty$ optimization techniques ($L = 5$).

scale systems. Here, we consider GEANT network and its real data set which contains large amount of unusual inputs. To find cover $\mathscr{C}$, compatible with MDFE framework, we randomly choose subsets $\{I_i\}_{i=1}^{L}$ so that set $I$ is covered, that is, $\mathscr{C} = I$. Then, using a small subset of inputs, $GE$ and $FE_{Avg}$ are computed and compared. This process is repeated to achieve desirable performance and the best cover is used to test the algorithm on the whole data set. A more structured set-covering algorithm for MDFE framework has been introduced in [23]. Table XII indicates the improvement achieved by applying MDFE framework where constrained optimization techniques represented by Eq.(16) and Eq.(17) (adopted from [8]) are used and solved using CVX for TM estimation in sub-spaces.

$$\hat{X} = \min_{X} \|Y - HX\|_\infty \text{ s.t. } X \geq 0 \tag{16}$$

$$\hat{X} = \min_{X} \|Y - HX\|_1 + \lambda \|x\|_1 \text{ s.t. } X \geq 0 \tag{17}$$

### *E. MDFE on Random Observation Matrices*

To show that MDFE can be applied on a wider range of problems, we did an extensive Monte-Carlo simulation over random-binary observation matrices. Figure 12 shows that MDFE can significantly improve the performance for fat (large $\frac{n}{m}$) and low-density matrices where partitioning helps to construct sub-spaces with smaller number of coherent unknowns, observed in different sub-spaces, with the capability of producing more precise estimates (since CN of sub-spaces are improved). Note that, all three networks used in our study are fat and low-density matrices.

## VI. Conclusion

In this paper, a novel approach for solving UDLI problems was introduced where a large-scale sparse problem is partitioned and solved in sub-spaces. By fusion the solution

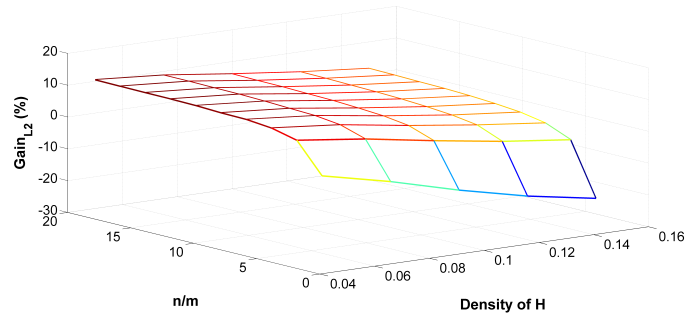



*Fig. 12:* $Gain_{L2}$ v.s. ratio $\frac{n}{m}$ and D (using Alg.1 and LNE where $X \sim U(0,1), m = 50, L = 5$).

from sub-spaces, we showed the possibility of improving the efficiency and robustness of computation process without compromising the accuracy of the solution. These are important factors in distributed and dynamic environments where accurate, quick and efficient inference are highly demanding. We examined the performance of MDFE in different applications, and we showed that MDFE is flexible and compatible with estimation techniques and input characteristics.

## References


[1] Y. Vardi, "Network tomography: Estimating source-destination traffic intensities from link data," *Journal of American Statistical Association*, vol. 91, pp. 365–377, 1996.

[2] R. Castro, M. Coates, G. Liang, R. Nowak, and B. Yu, "Internet tomography: Recent developments," *Statistical Science*, vol. 19, 2003.

[3] A. Medina, N. Taft, K. Salamatian, S. Bhattacharyya, and C. Diot, "Traffic matrix estimation:existing techniques and new directions," 2002.

[4] M. Roughan, Y. Zhang, W. Willinger, and L. Qiu, "Spatio-temporal compressive sensing and internet traffic matrices," *IEEE/ACM Transactions on Networking*, vol. 20, pp. 662–676, 2012.

[5] N. Duffield, J. Horowitz, and D. Towsley, "Multicast-based inference of network-internal loss characteristics," *IEEE Transactions on Information Theory*, vol. 45, pp. 2462–2480, 2006.

[6] N. Dufeld, "Network tomography of binary network performance characteristics," *IEEE Transactions on Information Theory*, vol. 52, p. 53735388, 2006.

[7] H. Nguyen and P. Thiran, "Network loss inference with second order statistics of end-to-end flows," *ACM-IMC*, 2007.

[8] Q. Zhao, Z. Ge, J. Wang, and J. Xu, "Robust traffic matrix estimation with imperfect information: Making use of multiple data sources," *ACM-SIGMETRICS*, 2006.

[9] A. Nucci and K. Papagiannaki in *Design, Measurement and Management of Large-Scale IP Networks*, Cambridge Univ. Press, 2009.

[10] M. Malboubi, C. Vu, C. Chuah, and P. Sharma, "Decentralizing network inference problems with multiple-description fusion estimation (mdfe)," tech. rep., 2012. At: http://web.ece.ucdavis.edu/cerl/techreports/2012-2/.

[11] J. Cao, D. Davis, S. Wiel, and B. Yu, "Time varying network tomography: Router link data," *Journal of American Statistical Association*, 2000.

[12] O. Goldschmidt, "Isp backbone traffic inference methods to support traffic engineering," *Internet Stat. and Metrics Analysis (ISMA)*, 2000.

[13] C. Tebaldi and M. West, "Bayesian inference of network traffic using link count data," *Journal of American Statistical Association*, vol. 93, pp. 557–573, 1998.

[14] G. Gursun and M. Crovella, "On traffic matrix completion in the internet," *ACM-IMC*, 2012.

[15] D. Ghita, C. Karakus, K. J. Argyraki, and P. Thiran, "Shifting network tomography toward a practical goal," p. 24, 2011.

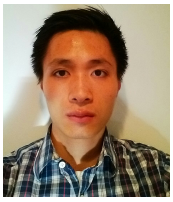
[16] D. Ghita, K. Argyraki, and P. Thiran, "Network tomography on correlated links," *ACM-IMC*, 2010.

[17] D. Ghita, H. Nguyen, M. Kurant, K. Argyraki, and P. Thiran, "Netsope: Practical network loss tomography," *IEEE-INFOCOM*, 2010.

[18] T. Hastier, R. Tibshirani, and J. Friedman, *The elements of statistical learning: data mining, inference, and prediction: with 200 full-color illustrations*. New York: Springer-Verlag, 2009.

[19] J. Demmel in *Applied Numerical Linear Algebra*, 1997.
[20] N. Higham in *Accuracy and Stability of Numerical Algorithms*, SIAM, 2002.
[21] "Abilene traffic." www.cs.utexas.edu/ yzhang/research/AbileneTM.
[22] "Geant network:." http://totem.info.ucl.ac.be/dataset.html.
[23] M. Malboubi, C. Vu, C.-N. Chuah, and P. Sharma, "Compressive sensing network inferrence with multiple description fusion estimation," *IEEE GLOBECOM*, 2013.

**Mehdi Malboubi** is a PhD candidate in the Department of Electrical and Computer Engineering in University of California at Davis. He completed his M.Sc. in the Department of Electrical Engineering and Computer Science in University of California at Berkeley. His main research interests include networking, communications, signal/image processing, machine learning/intelligence and data analysis/mining.



**Coung Vu** received his B.Sc. in Computer Science from the University of California, Davis. He joined UC Davis in 2010 and has participated in various undergraduate research activities. He created an earthquake flux visualization that was later on presented at a research conference at Davis. He also wrote network simulations to measure the accuracy of different network inference algorithms.

**Chen-Nee Chuah** is a Professor in Electrical and Computer Engineering at the University of California, Davis. She received her Ph.D. in Electrical Engineering and Computer Sciences from the University of California, Berkeley. Her research interests include Internet measurements, network management, anomaly detection, online social networks, and vehicular ad hoc networks. Chuah is a Fellow of the IEEE and an ACM Distinguished Scientist. She received the NSF CAREER Award in 2003, and was named a Chancellors Fellow of UC Davis in 2008. She has served on the executive/technical program committee of several ACM and IEEE conferences. She was an Associate Editor for IEEE/ACM Transactions on Networking from 2008 to 2012.

**Puneet Sharma** is a Principal Research Scientist at HP Labs where he conducts research on Data Center Networks, Cloud Architectures, Network Monitoring and Mobility. Prior to joining the HP Labs, he received his PhD in Computer Science from the University of Southern California. He also holds a B.Tech. in Computer Science and Engineering from IIT Delhi. His work on Mobile Collaborative Communities was featured in the New Scientist Magazine. He has participated in various standardization efforts. He contributed to the UPnP's QoS Working Group efforts as co-author for QoSv3 standards. Earlier, he had co-authored the IETF standards' RFCs on the multicast routing protocol PIM. Puneet is a Distinguished Scientist of the ACM and a Fellow of the IEEE.