Controlled Synthesis of Traffic Matrices

Paul Tune and Matthew Roughan, *Senior Member, IEEE*

***Abstract*—The traffic matrix (TM) is a chief input in many network design and planning applications. In this paper, we propose a model, called the Spherically Additive Noise Model (SANM). In conjunction with Iterative Proportional Fitting (IPF), it enables fast generation of synthetic TMs around a predicted TM. We analyse SANM and IPF's action on the model to show theoretical guarantees on asymptotic convergence, in particular, convergence to the well-known gravity model.**

***Index Terms*—Internet traffic matrix, Iterative Proportional Fitting, sensitivity analysis, synthetic generation**

I. Introduction

THE traffic matrix (TM) is an important input in a variety of tasks required in operating a backbone network. Chief amongst these tasks include network planning [20], network design [36], and traffic engineering [10], [23]. All these tasks are important to ensure that the network runs efficiently under various traffic demands.

In practice, there exists prior information about networks. The information may come from historical operational data for existing networks, or from user demand and demographic data for new networks. These data are then utilised to form *predicted TMs* that are used as input into the design of a network. However, the data contain noise, so the predictions are likely to contain errors.

One approach to deal with errors is to design a network that is oblivious to the TM. For instance, the Valiant network design [36] is designed to serve TMs with specific total ingress and egress traffic. Oblivious design avoids the need for predictions. However, the cost of TM agnosticism is that prior information is ignored, resulting in inefficient designs.

Instead, we would like a technique that incorporates information, but allow for errors. A network design based on this technique has an advantage over oblivious design, and also accounts for possible contingencies due to errors in prediction.

One idea to incorporate prediction errors is to generate an ensemble of TMs around the predicted TM to simulate prediction errors. Generated matrices should be

- *admissible*: they satisfy some set of constraints (*e.g.,* non-negativity);
- *centred*: they are perturbations around the predicted matrix; and
- *controlled*: their variance around the predicted matrix can be controlled, ideally through a simple, linear parameter.

The centred property could be thought of as the dual of unbiasedness of an estimate in estimation theory; here, we want ensembles to revolve around a prediction. If we didn't want matrices centred around a prediction, then we must have additional information about the noise model that should be incorporated clearly, instead of by accident. So, 'centredness' is the sensible default position to start any such model.

Synthesis of TMs has been done before. Nucci *et al.* [16] proposed a heuristic-based method of generating TMs, with entries distributed according to the lognormal distribution. Roughan [19] instead generated TMs using the gravity model. These schemes, however, were proposed to alleviate the paucity of available public data of TMs, not simulate prediction errors. They do not fit the needs described above.

Moreover, the canonical example of TM ensembles in many practical situations is to add IID (Independent Identically Distributed) Gaussian *noise* to the prediction. However, in our problem, simple additive noise fails the first two criteria *i.e.,* admissibility and centredness, necessitating an alternative.

In this paper, we propose a simple scheme to generate an ensemble of random TMs. The main contribution of this paper is the *Spherically Additive Noise Model (SANM)* for the prediction errors of TMs around the predicted TM. To ensure that the model conforms to the constraints, we use *Iterative Proportional Fitting (IPF)*. Our proposed scheme satisfies all three criteria outlined above.

Furthermore, the method is fast and has a sound theoretical basis. In particular, the generated TMs have an information theoretic interpretation and are connected to the gravity model [33]. Additionally, they possess a property called *Proportional Error Variance Aggregation (PEVA)*, which means that the magnitude of the prediction errors of the entries in the TMs are proportional to the size of the traffic demand, inline with prediction errors in practice. These useful properties make our scheme flexible and powerful enough to be used for a variety of simulation tasks encountered by network operators.

Our work is useful because it is the first contribution towards generating ensembles of TMs taking prediction errors into account. The results can be used in a variety of tasks that require the TM as input, allowing testing of the robustness of algorithms to solve these tasks.

The present paper is a companion to our previous work published in [29], where we showed how our scheme can be applied to perform sensitivity analysis to network design algorithms, illustrating that some designs perform poorly for TMs they are not designed for. We also showed that we could create efficient, robust optimisation procedures. Rather than optimise the design over the space of all admissible TMs, we generate ensembles of TMs around the predicted TM (to account for prediction errors) as input to the network optimisation problem. In contrast, our emphasis here in is on theoretical properties of SANM and IPF, details of which were omitted from [29].

Paul Tune is at Image Intelligence, Email: anamolous.behaviour@gmail.com. Matthew Roughan is at the ARC Centre of Excellence Mathematical and Statistical Frontiers (ACEMS), at the School of Mathematical Sciences, The University of Adelaide, Australia. Email: matthew.roughan@adelaide.edu.au
This paper was presented in part at ACM SIGMETRICS, Austin, TX, USA, June, 2014.

II. BACKGROUND

A traffic matrix (TM) describes the volume of traffic (typically in bytes) from one point in a network to another in a time interval. Its natural representation is a three dimensional array $\mathbf{T}(\tau)$ with elements $t_{i,j}(\tau)$ that represent the traffic volume from source i to destination j during the time interval $[\tau, \tau + \Delta\tau)$. Time intervals depend on the time resolution of measurements. Typical intervals are of the order of minutes to hours (5, 15, and 60 minutes are very common). In our applications, the locations i and j will either be routers or, more typically Points-of-Presence (PoPs), and we only consider Ingress/Egress (IE) TMs here (not underlying Origin/Destination (OD) matrices as these are usually unobservable). We will denote the number of ingress and egress points by N. Throughout the paper, we concentrate on a single snapshot of the TM at time τ, so we will omit the time index for convenience. For a more complete description of TMs see [28], and for spatio-temporal models see [30].

The best current practice for measuring a network's TM is to use sampled flow-level measurements (*e.g.,* [8]). These measurements provide TMs on a time granularity on the order of minutes, but these matrices will contain errors. Such errors arise from sampling and from lack of synchronisation in time intervals used in measurements. Control and calibration of the size of these errors, as far as we are aware, is not currently implemented. However, one might hope they were fairly small, say at the level of a few percent, in a well-engineered measurement system.

Alternative approaches to direct (sampled) collection include a body of research [13], [32–34] devoted to developing TM inference methods from more easily collected link-load measurements. These methods, however, are limited by the underconstrained nature of the problem, so average errors on the order of 10–20% are possible. Despite these errors, [23] showed that traffic engineering tasks, such as routing, can certainly benefit from an inferred TM as compared to having no knowledge whatsoever.

More importantly, one of the underlying themes of that research is that it is much easier to measure the traffic volumes on links than to determine where that traffic is going (and hence the TM). It is quite easy to measure values such as the row sums of the TMs $r_i = \sum_j t_{i,j}$ (which tell us the total traffic coming into a PoP) and the column sums $c_j = \sum_i t_{i,j}$ (which tell us the total traffic leaving through a PoP).

Network planning requires forecasting future TMs to the date relevant for the plans. The time in advance is often called the *planning horizon*. The prediction process may vary, but it will inevitably introduce prediction errors. Here, we concentrate on purely spatial TMs, as in the applications we target such as network capacity planning, we have shown that the worst case (spatial) TM is most important [31].

Common sources of prediction error are

- *Statistical inference errors:* due to stochastic variability within the model, *i.e.,* the model for the data is a good approximation, but the particular realisation we observed varies from that predicted.
- *Large, short-term fluctuations from the prediction: e.g.,* routing changes (either caused by internal link failures, or external routing policy changes), can alter the egress points of traffic, altering the IE TM (even if the underlying OD matrix remains unchanged) [27]; or flash-crowds can cause large, short-term changes to traffic. Some of these changes might be considered simple stochastic variation (as above), but some are larger, and more sudden than can be accounted for by normal variation, and their causes can often be identified as singular events.
- *Modelling errors:* resulting from use of a prediction technique whose underlying model is inaccurate over the prediction period. For instance, growth in traffic may appear to be exponential over some period, but actually be logistic (appearing exponential early on, but slowing as demand becomes saturated).

Our primary goal is to *develop a method of generating TMs that takes these prediction errors into account*. We do this by generating controlled perturbations around the predicted TM. As modelling errors should be preventable by careful analysis of sufficient periods of historical data, we focus on the other two fluctuations which are intrinsic to prediction.

The level of aggregation of traffic is often a large factor in the size of the statistical prediction errors. Large aggregates of traffic are more predictable because they statistically aggregate the behaviour of many more sources. It is natural then to expect errors to have some dependence on the size of TM element. TM elements have a somewhat skewed distribution (with many small values, and few large) though not formally heavy-tailed [16], [19], [26], implying that we have a large group of smaller, more inaccurate predictions, and a smaller group of large, but comparatively accurate predictions. The Norros model [15], empirically tested in [21] and based on similar assumptions to the independent flow model [7] proposed for TMs, proposes that variance should be proportional to the mean of the traffic. Moreover, the model has consistent variances for aggregated traffic. We need an error model satisfying this property, which we call *proportional error variance aggregation*, or PEVA.

We might also reasonably assume that prediction errors for aggregates such as the $2N$ row and column sums of a TM will be smaller than for the N^2 elements. Moreover, prediction of PoP (in/out) volumes can take advantage of demographic, marketing, business and financial predictions much more easily than can the $t_{i,j}$.

The point we are making is that TMs contain considerable value, but it must be assumed that it contains errors. However, these errors are not completely arbitrary. Larger matrix elements have smaller relative errors, and aggregates (such as row and column sums) often have much smaller errors than those in the TM elements themselves.

III. SYNTHESIS ALGORITHM

Our objective is to generate an ensemble of TMs with controlled perturbations around a predicted TM. We focus on generating Ingress/Egress (IE) TMs, which are important and practical at the Points-of-Presence (PoP) level.

There are prescribed constraints on these matrices, so not every matrix is a valid TM. Although there are many choices for the constraints, in this paper we will focus on 4 simple, intuitive, and yet reasonably powerful constraints. For a matrix $\mathbf{T} = [t_{i,j}]$, we have

(i) *Non-negativity*: $t_{i,j} \geq 0, \forall i, j = 1, 2, \cdots, N$,

(ii) *Row sums*: $\sum_j t_{i,j} = r_i$, $\forall i$,

(iii) *Column sums*: $\sum_i t_{i,j} = c_j$, $\forall j$, and

(iv) *Traffic conservation*: $\sum_{i,j} t_{i,j} = \sum_i r_i = \sum_j c_j = T$,

where row and column sums are summarised in vectors $\mathbf{r}$ and $\mathbf{c}$ respectively, which for the moment we consider to be constant.

The constraints above are motivated by the fact that the row and column sums represent the total ingress and egress traffic of a network of PoPs, which are relatively easy to obtain from SNMP information [28]. Moreover, these constraints are a natural fit in inference problems, where one might use them to construct the gravity model [33] as a first order estimate. The constraints also provide a way to fairly compare the performance of valid network designs, as in [29].

Generally, we might replace the row, column and total traffic constraints with any set of linear (or more generally, convex) constraints on the matrix, but we use the constraints above for illustration because of their clear meanings in context.

A. Spherically Additive Noise Model

The goal of our approach is to guarantee the admissibility constraints are (at least in part) automatically satisfied by any matrix we generate. To do so we divide the constraints into *particular* constraints (the summation constraints, which are dependent on the particular network), and *universal* constraints (non-negativity and traffic conservation), which all traffic matrices should satisfy.

A standard error model, the *additive noise model*, adds white noise to the entries of $\mathbf{T}$, *i.e.,*

$$y_{i,j} = t_{i,j} + \sigma z_{i,j}, \ \forall i, j, \tag{1}$$

where $z_{i,j} \sim \mathcal{N}(0, 1)$, *i.e.,* the standard normal distribution, and $\sigma > 0$ controls the noise strength. While this model is controllable by a single parameter σ, there is a major problem. The entries $y_{i,j}$ may not be non-negative, so a preprocessing step is required: projecting the entries onto the non-negative plane. Optimisation theory tells us that this projection is equivalent to setting any negative entry to zero [4]. As a result, with large noise, *i.e.,* large σ, many entries are zeroed out, or *truncated*. The model after truncation is no longer centred and generates unrealistic TMs. Another standard error model is the *multiplicative model*, defined by

$$y_{i,j} = (1 + \sigma z_{i,j}) t_{i,j}, \ \forall i, j, \tag{2}$$

but it suffers from the same problems as the additive model.

We aim to develop a model that avoids truncation. Given the non-negativity, it is possible to write any TM $\mathbf{T} = [t_{i,j}]$ in the form

$$\mathbf{T} = \left[a_{i,j}^2\right],$$

where $a_{i,j} = \sqrt{t_{i,j}}$. The modification seems trivial, but given real values $a_{i,j}$, the matrix is now inherently non-negative. Moreover, the total TM constraint becomes

$$\sum_{i,j} a_{i,j}^2 = T,$$

so we can see the $a_{i,j}$s as lying on the N^2-dimensional hypersphere with radius $\sqrt{T}$ (see Figure 1). This transformation has been used before in optimisation literature for similar reasons.

Our approach, therefore, is to perturb the matrix by finding a new point on this hypersphere via an approximately additive model. This allows us to add noise in a controlled and centred manner, while preserving the universal constraints.

Our *Spherically Additive Noise Model* (SANM) is

$$y_{i,j} = (a_{i,j} + \beta z_{i,j})^2, \ \forall i, j, \tag{3}$$

where $z_{i,j} \sim \mathcal{N}(0, 1)$, and $\beta \in [0, \infty)$ is a parameter we can use to tune the strength of the noise. We chose a simple IID noise process, as we have no *a priori* reason to assume correlations in the noise.

Typically, the size of the noise β would take small values, but we show in Section III-D that the model behaves correctly as $\beta \to \infty$. For large β, it approaches the gravity model, which is the natural model when we have little information except for the row and column sums.

The beauty of the simplicity of this model is that it guarantees non-negativity without truncation, and so not only does it avoid artificial zeros in the matrix, it is also simple to analyse. Since $z_{i,j} \sim \mathcal{N}(0, 1)$, independent of $a_{i,j}$, we get

$$\begin{aligned} \mathsf{E}[y_{i,j}] &= \mathsf{E}\left[(a_{i,j} + \beta z_{i,j})^2\right] \\ &= \mathsf{E}\left[a_{i,j}^2 + 2\beta a_{i,j} z_{i,j} + \beta^2 z_{i,j}^2\right] \\ &= t_{i,j} + 2\beta \mathsf{E}[a_{i,j} z_{i,j}] + \beta^2 \mathsf{E}\left[z_{i,j}^2\right] \\ &= t_{i,j} + 2\beta a_{i,j} \mathsf{E}[z_{i,j}] + \beta^2 \\ &= t_{i,j} + \beta^2. \end{aligned}$$

This is not ideal (yet), because the resulting matrix doesn't satisfy the centring condition, so a procedure is needed to force the matrix to satisfy this, and the additional constraints.

B. Iterative Proportional Fitting (IPF)

The matrix $\mathbf{Y}$ does not sit on the hypersphere defined by the total traffic constraints (or the manifolds defined by the other constraints). Intuitively, what we need is a procedure to *project* the perturbed solution onto the space defined by the set of constraints. We use Iterative Proportional Fitting (IPF). Figure 1 presents geometric representation of how the perturbed matrices $\mathbf{Y}_1$ and $\mathbf{Y}_2$ from SANM are projected to the set of matrices satisfying the constraints.

Algorithms 1 and 2 outline our method. The algorithm is very simple indeed, with the only complication occurring in the implementation of IPF, which we detail below.

IPF was originally developed to adjust contingency tables in statistics such that their marginals, given by the row and column sums, satisfy known constraints [5]. This is almost exactly what we aim to do, the only difference being terminology, where we apply IPF to TMs instead.

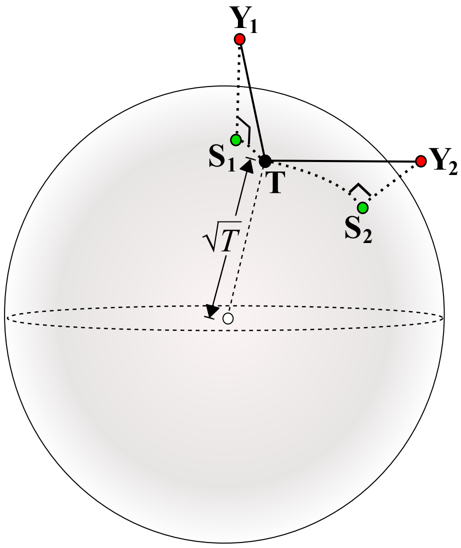


Fig. 1. Illustrating the action of SANM and IPF on the hypersphere of non-negative matrices with a total sum of T. After perturbation from $\mathbf{T}$, the resulting matrices $\mathbf{Y}_1$, and $\mathbf{Y}_2$ no longer sit on the hypersphere, especially if β is large. This necessitates the use of IPF to enforce constraints and the resulting projections are $\mathbf{S}_1$ and $\mathbf{S}_2$ respectively. Note that $\mathbf{S}_1$ and $\mathbf{S}_2$ are members of the restricted set of matrices satisfying the row and column sum constraints.

Input: $\mathbf{T}$, the predicted traffic matrix
Input: β, noise variance
Input: $\mathbf{r}$, $\mathbf{c}$, row and column sum constraints
Input: ϵ, tolerance for IPF
Output: $\mathbf{S}$, the synthetic traffic matrix
Generate $\mathbf{Z}$, with $z_{i,j} \sim \mathcal{N}(0,1)$
Generate $\mathbf{Y}$, where $y_{i,j} = (t_{i,j}^{1/2} + \beta z_{i,j})^2$
$\mathbf{S} = \text{IPF}(\mathbf{Y}, \mathbf{r}, \mathbf{c}, \epsilon)$ `/* See Algorithm 2 */`

Algorithm 1: SANM

IPF has been used extensively in the transportation literature. Activity-based travel models require the use of synthetic populations to generate microsimulations of traffic. There is, therefore, a direct analogy between the population and TMs. The earliest use of IPF in transportation studies is to combine different sources of data (for instance, population census data and metropolitan transportation surveys) to estimate socioeconomic variables of interest [6].

Beckman *et al.* [1] used IPF to generate synthetic populations via the following procedure: sample a subpopulation from a census dataset with desired demographic properties, then use IPF to fit this subpopulation to a set of marginal constraints (like the row and column sums corresponding to total population of demographic groups). In this sense, their technique has similarities with our work, but here we first developed a model (SANM) as an input to IPF, whereas they used empirical data as their input. Beckman *et al.* 's techniques were incorporated into TRANSIMS [25], a microsimulation software based on cellular automata.

More recently, IPF has been adapted to work with an arbitrary set of measurement constraints (provided the constraints

Input: $\mathbf{Y}$, the SANM output
Input: $\mathbf{r}$, $\mathbf{c}$, row and column sum constraints
Input: ϵ, tolerance for IPF
Output: $\mathbf{S}$, the synthetic traffic matrix
$\mathbf{S}^{(0)} \leftarrow \mathbf{Y}$
$k \leftarrow 1$
while (not converged) **do**
  `/* Scale the rows */`
  for $i := 1$ **to** N **do**
    $\mathbf{S}_{i,*}^{(k-1/2)} = \mathbf{S}_{i,*}^{(k-1)} r_i / \sum_j \mathbf{S}_{*,j}^{(k-1)}$
  end
  `/* Scale the columns */`
  for $j := 1$ **to** N **do**
    $\mathbf{S}_{*,j}^{(k)} = \mathbf{S}_{*,j}^{(k-1/2)} c_j / \sum_i \mathbf{S}_{i,*}^{(k-1/2)}$
  end
  $k \leftarrow k + 1$
end

Algorithm 2: Iterative Proportional Fitting

lie in a convex set) [12]. This implies that many types of constraints in network design can be incorporated into IPF. Here, we restrict our attention to only row and column sum constraints which naturally map to our current problem.

IPF consists of iteratively scaling the rows and columns of the matrix until the scaled row and column sums match the objective row and column sums (see Algorithm 2). Here, $*$ denotes a wildcard to specify the rows or columns of the matrix at once. The typical test for convergence is to compare the row and column sums of the iterate $\mathbf{S}^{(k)}$ to $\mathbf{r}$ and $\mathbf{c}$ at the end of each iteration k. Once the total difference falls below some required tolerance ϵ, the algorithm has converged. We use the convergence criterion

$$\sum_i \left| \sum_j s_{i,j} - r_i \right| + \sum_j \left| \sum_i s_{i,j} - c_j \right| < \epsilon,$$

which measures the deviation of the matrix's row and column sums from $\mathbf{r}$ and $\mathbf{c}$ [18].

IPF has a strong connection to the Kullback-Leibler (KL) divergence [2], defined as

$$D(\mathbf{P} \,||\, \mathbf{Q}) = \sum_{i,j} p_{i,j} \log \frac{p_{i,j}}{q_{i,j}}, \tag{4}$$

between the probability distributions $\mathbf{P}$ and $\mathbf{Q}$, where $p_{i,j},\ q_{i,j} \in [0,1]$ for all i,j and $\sum_{i,j} p_{i,j} = \sum_{i,j} q_{i,j} = 1$ (we use the standard convention $\log(0/0) = 0$). Let $T = \sum_i r_i = \sum_j c_j$. IPF is an iterative, fixed-point solution to the problem of minimising the general KL divergence, defined by

$$D_{GKL}(\mathbf{X} \,||\, \mathbf{Y}) = T \times D\left(\frac{1}{T}\mathbf{X} \,\middle|\middle|\, \frac{1}{T}\mathbf{Y}\right) - \sum_{i,j} x_{i,j} + \sum_{i,j} y_{i,j} \tag{5}$$

between the input to the final solution, subject to constraints [3], [11]. The proof of IPF's convergence can be found in [3]. More complex algorithms can be used to solve this problem, but IPF's simplicity and fast convergence are ideal here.

An additional property of IPF worth mentioning is that it is guaranteed to converge to a solution satisfying the row and column constraints only if the initial input matrix is non-negative [5]. Our error model is guaranteed to be non-negative, so convergence is assured.

Moreover, IPF preserves zeros [5], [24]. Thus problematic zeros introduced by truncation (if a simplistic error model was used) would be retained by this step. This is undesirable, as real TMs do not have many zero elements. SANM is unlikely to have zero elements for $\beta > 0$. Zeros occur only if $t_{i,j}^{1/2} = -\beta z_{i,j}$, but since $z_{i,j}$ is a continuous random variable, this happens with probability 0. So our noise model creates elements that are almost-surely positive (simulations confirm this, the results of which we omit here in the interest of space). Therefore, IPF's outputs are also positive.

IPF isn't novel, even in the context of TMs, where it has been used to enforce row and column sum constraints before [33]. However, besides [1] in transportation literature, in the cases we are aware of, IPF was used as part of an inference technique, whereas here it is being used for synthesis.

C. Discussion

In this synthesis problem, we are essentially finding a set of random matrices $\mathbf{S} = \mathbf{T} + \mathbf{W}$, where $\mathbf{T}$ is the predicted TM and $\mathbf{W}$ are the prediction errors, such that the $\mathbf{S}$ satisfy a set of constraints

$$\mathcal{A}(\mathbf{S}) = \mathbf{b}.$$

The operator $\mathcal{A}$ might represent a set of measurement operations, where $\mathbf{b}$ are observed measurements, or in the preceding section we considered $\mathcal{A}$ to take row and column sums and $\mathbf{b}$ to be the specified values of those sums. In general, we only require linear operators $\mathcal{A}$, and linearity implies that $\mathcal{A}(\mathbf{T}) = \mathcal{A}(\mathbf{S}) = \mathcal{A}(\mathbf{T} + \mathbf{W}) = \mathcal{A}(\mathbf{T}) + \mathcal{A}(\mathbf{W}) = \mathbf{b}$, or

$$\mathcal{A}(\mathbf{W}) = \mathbf{0},$$

i.e., the noise $\mathbf{W}$ must lie in the null space of $\mathcal{A}$, with the additional complication of requiring non-negativity.

Note that, although the noise matrix $\mathbf{Z}$ is IID, the subsequent matrix $\mathbf{W}$ will not be, because of this requirement.

One method of finding such matrices is by defining a manifold on which all valid TMs reside and forming the above additive model on this manifold. However, that requires

(i) defining invertible maps between the manifold and the generated TMs, and
(ii) search algorithms on manifolds (which are much slower than IPF).

For instance, see [9] for an example of a manifold and its relationship to IPF. So the approach we propose is solving a more general problem, but in a fast, easy to implement manner.

Although some measurements are more or less accurate, few are perfect. For example, the row and column sums may be subject to variations over time. Regularisation methods may be used to handle variations of these measurements, but a simpler way we adopt in the paper is to add a small amount of noise to the constraint values $\mathbf{b}$ prior to TM generation. For a valid TM, the constraints must be self-consistent, for *e.g.,* the total of the row and column sums must be equal to the total traffic traversing the network, but this is easy to ensure. So the SANM allows controllable errors in the constraints as well as the generated TMs themselves. We test SANM on variations of the row and column sums and demonstrate their usefulness in network sensitivity design in [29].

It is worth noting that our method does not constrain the individual elements of the TM directly. One motivation for direct constraints occurs between the different customers of an ISP, as each may have different SLAs, and are therefore each sensitive to different variations in traffic. To capture these variations, ideally, one would have to define a distribution to specify the traffic variations. It is possible to do so in our model, however, this would

(i) require additional work in specifying the type of noise used in SANM, as the noise may no longer be IID, and
(ii) define new constraints in IPF.

Although feasible, more care has to be taken when specifying the noise and new constraints in the context of the problem.

D. Properties and Asymptotic Behaviour

In this section, we discuss the theoretical properties of the centredness, convergence, variation around the predicted TM and asymptotic behaviour of our algorithm.

Centredness. The SANM inherently generates admissible matrices. Two other criteria for this model were that it be centred and controllable. The SANM is nonlinear, and hence somewhat hard to analyse, but intuitively, IPF is finding the *closest* point on the appropriate manifold to project the perturbed solution to, so we might expect these properties.

Analytically, we consider the simplest case where the row and column constraints are ignored, and only the total traffic constraint is included (for more complex row and column problems we quantify the effect in the following section). In this simple case, IPF will just scale the values so that the total is correct, *i.e.,* on average,

$$\mathsf{E}[s_{i,j}] = c\mathsf{E}[y_{i,j}] = c\left(t_{i,j} + \beta^2\right).$$

The scaling constant c will be chosen so that $\sum_{i,j} s_{i,j} = T$, so we know that

$$\mathsf{E}\left[\sum_{i,j} s_{i,j}\right] = c\mathsf{E}\left[\sum_{i,j} t_{i,j} + \beta^2\right] = c\left(T + N^2\beta^2\right) = T,$$

and so

$$c = \frac{1}{1 + N^2\beta^2/T}.$$

Take the average matrix element to be $\bar{t} = T/N^2$ and we get

$$\mathsf{E}[s_{i,j}] = \frac{t_{i,j} + \beta^2}{1 + \beta^2/\bar{t}} = t_{i,j} + \beta^2\left(1 - \frac{t_{i,j}}{\bar{t}}\right) + O(\beta^4).$$

So, in expectation, larger elements of the matrix become smaller, and smaller elements become larger, but the effect is $O(\beta^2)$, and so for small values of β the non-linearity is negligible. We therefore have approximate centering of the synthetic matrices.

In addition, we can consider the variance of the synthetic matrix about the predicted matrix by looking at the variance of $s_{i,j}$ and noting that at least approximately this is the same as the variation around $t_{i,j}$:

$$\text{Var}[s_{i,j}] = \mathsf{E}\left[s_{i,j}^2\right] - \mathsf{E}[s_{i,j}]^2 \simeq \mathsf{E}\left[s_{i,j}^2\right] - t_{i,j}^2 = \mathsf{E}\left[(s_{i,j} - t_{i,j})^2\right],$$

and

$$\begin{aligned} \mathsf{E}\left[s_{i,j}^2\right] &= c^2 \mathsf{E}\left[y_{i,j}^2\right] = \frac{\mathsf{E}\left[\left(a_{i,j} + \beta z_{i,j}\right)^4\right]}{\left(1 + \beta^2/\bar{t}\right)^2} \\ &= t_{i,j}^2 + \beta^2 t_{i,j}\left(6 - 2\frac{t_{i,j}}{\bar{t}}\right) + O(\beta^4), \end{aligned} \quad (6)$$

so the standard deviation of the variability around the predicted matrix is approximately linear in β, for small β. Averaging over the whole matrix to obtain a relative measure of standard deviation around matrix elements we get

$$\sqrt{\frac{1}{N^2}\sum_{i,j}\frac{\mathsf{E}\left[s_{i,j}^2\right] - t_{i,j}^2}{t_{i,j}^2}} \simeq \beta\sqrt{\frac{1}{N^2}\sum_{i,j}\frac{6}{t_{i,j}} - \frac{2}{\bar{t}}}. \quad (7)$$

We will examine the range of linearity of this approximation empirically for the more general case including row and column constraints in the following section. We find the approximation to be very good in the range $0 \le \beta \le 0.2$, and reasonable for β up to 0.4.

Convergence. A known result, Theorem 3 of [18], states that IPF converges to a unique solution if and only if there exists a final matrix generated by IPF matching the zeros of any predicted TM generated by SANM, its row and column sums matches $\mathbf{r}$ and $\mathbf{c}$, and that the traffic conservation constraint is satisfied. Given that the row and column sums are equal to the total traffic and since every matrix generated by SANM has positive entries, there exists such a matrix: $\mathbf{r}\mathbf{c}^{\mathrm{T}}/T$ (the gravity model), which matches $\mathbf{r}$ and $\mathbf{c}$. Since a solution exists, IPF is guaranteed to converge.

Variation around the predicted TM. The synthetic TMs generated by our algorithm are guaranteed to be full rank almost surely. Rank is an important consideration here as IPF doesn't necessarily return a unique solution if the input matrix has a particular structure, because while the KL divergence (4) is convex in $\mathbf{X}$, it is not strictly convex. For instance, if the input matrix has entries $y_{i,j} = u_i v_j$, with $u_i, v_j > 0\ \forall i, j$, *i.e.,* the matrix is positive and single rank, then by running IPF, the solution is exactly the gravity model. Following Algorithm 2, in step 1/2,

$$s_{i,j}^{(1/2)} = \frac{r_i u_i v_j}{u_i \sum_k v_k} = \frac{r_i v_j}{\sum_k v_k}.$$

By step 1, we have

$$s_{i,j}^{(1)} = \frac{r_i v_j c_j}{\sum_k v_k} \frac{\sum_k v_k}{v_j \sum_\ell r_\ell} = \frac{r_i c_j}{T}.$$

Thus, with single rank matrices, IPF converges within a step to the gravity model.

To prove full rankedness, we first require the following result:

Theorem 1: IPF preserves the rank of the input matrix.

Proof: There exists two diagonal matrices $\mathbf{D}_1$ and $\mathbf{D}_2$ with positive entries such that the solution has the form $\mathbf{X} = \mathbf{D}_1\mathbf{Y}\mathbf{D}_2$ [3, Corollary 3.3]. IPF preserves the rank of the input matrix: $\text{rank}(\mathbf{X}) = \text{rank}(\mathbf{D}_1\mathbf{Y}\mathbf{D}_2) = \text{rank}(\mathbf{Y})$, since both $\mathbf{D}_1$ and $\mathbf{D}_2$ are full rank. ■

The joint distribution of the entries of output of SANM, matrix $\mathbf{Y}$, has an absolutely continuous density, because its cumulative distribution function is continuous. The set of singular matrices, *i.e.,* $\mathcal{Y} := \{\mathbf{Y} \mid \det(\mathbf{Y}) = 0\}$ has positive codimension (since the matrices contain a dependent row or column and would not be full rank) so the set has measure zero in the Lebesgue measure. Since $\mathbf{Y}$'s entries have an absolutely continuous density in the Lebesgue measure, $\mathbf{Y}$ is almost surely full rank. Finally, by Theorem 1, since IPF preserves the rank, then the synthetic TMs generated by our model are full rank almost surely.

Our result shows that with the various inputs from SANM, the output from IPF will not be the gravity model almost surely. This proves that our algorithm generates variations around the predicted TM. We want this property because testing network designs for robustness require an ensemble of TMs with variations.

Asymptotic behaviour. Although IPF's output is not the gravity model almost surely, the average of the synthetic TM ensemble, as $\beta \to \infty$ is well-approximated by the gravity model. In the limit $\beta \to \infty$, $y_{i,j} \simeq \beta^2 z_{i,j}^2$, where $z_{i,j}^2$ is IID according to the chi-squared distribution with a degree of freedom of one, so $\mathsf{E}[\mathbf{Y}] = \beta^2 \mathbf{1}_N \mathbf{1}_N^{\mathrm{T}}$. As a result, the average of $\mathbf{S}$ is approximated by the gravity model, which is the solution of IPF when $\mathsf{E}[\mathbf{Y}]$ is IPF's input (since it is a single rank matrix). We defer a more detailed proof of this to Appendix A.

IV. Evaluation

Our earlier work [29] was focussed on demonstrating the usefulness of SANM through two case studies. The goal of these was to show how network design benefitted from the ability to generate an ensemble of matrices. The results showed that some networks that were designed to serve TMs with a specific $\mathbf{r}$ and $\mathbf{c}$, such as the Valiant [36] and star network designs, were not as robust to TM variations.

The main lesson from our previous work was that if the assumptions on the type of TMs the oblivious network designs were meant to serve were violated, then these designs require far more capacity than they should. In this sense, these networks are *sensitive* to their design assumptions. Surprisingly, the Abilene design [14] did very well under variety of TMs, proving that Abilene's designers did a thorough job.

Our previous work also proposed a *risk-based* framework for designing networks. Here, there are two competing objectives in network design: risk and utility. Risk can be thought of as the cost of overexploiting resources of the network (for *e.g.,* bandwidth). Utility is an quantity the network operator wants to maximise, such as revenue. Thus, the goal is deliver a design that meets the trade-off between risk and utility.

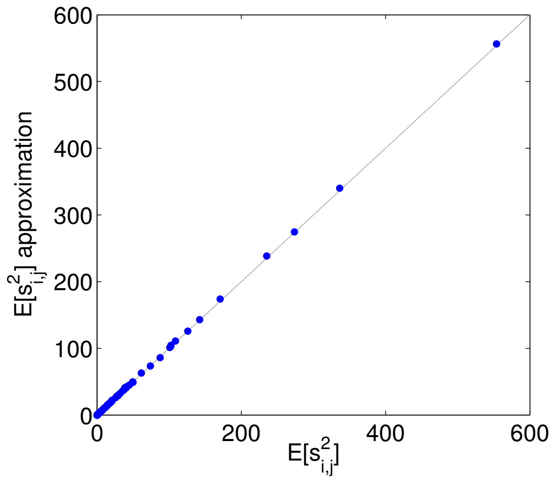


Fig. 2. Testing the approximation (6) against IPF's output $\mathsf{E}[s_{i,j}^2]$ on a single snapshot from the Abilene [14] data, the average traffic over time 02:00 to 02:05 hours, 1st March 2004, with $\beta = 0.2$.

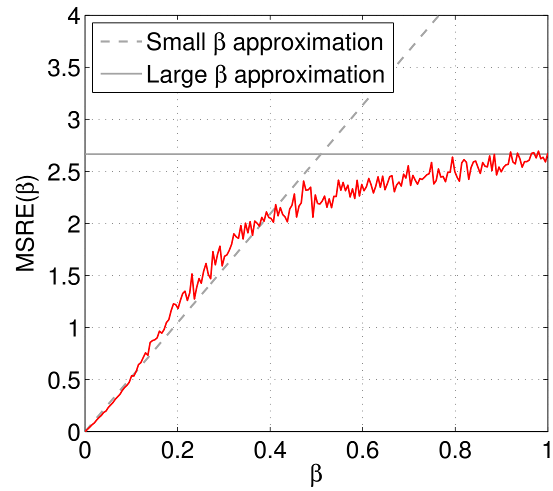


Fig. 3. MRSE as a function of β. The solid curve is the average response of IPF's output over 1000 trials per data point. The predicted TM $\mathbf{T}$ is from Abilene data, the average traffic over time 02:00 to 02:05 hours, 1st March 2004. The dashed and flat lines show the approximation for small β (see approximation (7)) and large β (see details in the Appendix A).

Trying to find a network design that possesses the best trade-off between the utility and risk requires an exhaustive search over all TMs, which is hard to do. Instead, we used our algorithm to generate ensembles of TMs to search for an acceptable network design over a range of variations on a predicted TM.

Instead of replicating the case studies here, we instead investigate the properties of SANM in-depth. Readers interested in the applications of SANM are referred to [29].

A. Tests

In the following results, the predicted TM $\mathbf{T}$ is a single snapshot from the Abilene [14] data, the average traffic over time 02:00 to 02:05 hours, 1st March 2004.

We first test the approximation (6). In Figure 2, we set $\beta = 0.2$ and plot approximation (6) against empirical measurements of $\mathsf{E}\left[s_{i,j}^2\right]$ (which also includes the row and column sum constraints), averaged over 1000 trials. The middle solid line denotes the equality between elements, *i.e.,* if the approximation equals $\mathsf{E}\left[s_{i,j}^2\right]$. We find that the fit is very good indeed, as the data points lie close to the solid line, despite the fact the approximation did not try to account for the row and column sum constraints. We have tested this with various predicted TMs with similar results. The approximation is fair up to $\beta = 0.4$, beyond which the approximation quality degrades.

Our model also coincides with the intuition about the modelling of aggregated flows per the Norros model [22]. In real traffic, as flows are combined, the aggregated flow has a higher variance, and this variation is modelled to be proportional to the size of the aggregated traffic flow in the Norros model. These larger flows would naturally have larger measurement error. From approximation (6), we find the errors scale in proportion to $t_{i,j}$, aligning with the intuition of the Norros model. In the SANM, the error would therefore scale proportional to the size of the predicted TM's elements. We call this the *proportional error variance aggregation* (PEVA) property.

Another consideration is how far the generated TMs are from the original input. Since the SANM is nonlinear, will there be significant distortions of the matrices?

We measure the distortion using the Mean Relative Square Error (MRSE) of the generated traffic matrices to the input $\mathbf{T}$, defined by

$$\text{MRSE}(\beta) = \sqrt{\frac{1}{N^2}\sum_{i,j}\frac{(s_{i,j}(\beta) - t_{i,j})^2}{t_{i,j}^2}}.$$

The metric quantifies the effect of the perturbation of IPF's solution to the predicted TM. Ideally, we would like a linear relationship between β and the MRSE, implying that $s_{i,j}$ is proportional to $t_{i,j}$, and this is predicted for small β by (7), but without row and column sum constraints. Here, we include those constraints and test the MRSE response empirically.

We examine the MRSE response of IPF in Figure 3. The solid curve is the average response of the generated TMs. The curve was generated from 1000 trials. The sloped dashed line is the response of the approximation (7) for the small β regime, while the flat line is the response of the gravity model for the large β regime.

In Figure 3, the response of the MRSE to β is almost linear up to about $\beta = 0.4$ before starting to saturate to the response of the gravity model (see Appendix A).

In Figure 4, we fixed $\beta = 0.2$ to illustrate the variation around the same predicted TM as before by plotting the histogram of the MRSE, computed from 1000 trials of synthetic matrices. We see that although most synthetic TMs are close to the predicted TM, the MRSE variance around the average response is large, so our error model generates variation which is useful in applications. The point that we make here is that our scheme does not produce TMs that have little to no variation from the predicted TM.

Table I compares the SANM against the (truncated) additive (1) and multiplicative (2) models across several important

Model	Parameter	Mean % zeros	Admissible	Centred	Controlled	PEVA
Additive	$\sigma = \bar{t}$	≈ 50.0	✓	✗	✓	✗
Multiplicative	$\sigma = 1$	13.6	✓	Almost	✓	✗
SANM	$\beta = 0.2$	0.0	✓	✓	✓	✓

TABLE I
COMPARING THE (TRUNCATED) ADDITIVE, MULTIPLICATIVE ERROR MODELS AND SANM OVER SEVERAL CRITERIA. PEVA IMPLIES THAT THE ERROR OF THE GENERATED MATRIX ELEMENTS SCALE IN PROPORTION TO THE SIZE OF THE PREDICTED TM ELEMENT AS NOISE INCREASES. MODEL PARAMETERS WERE CHOSEN SO THAT THEIR MRSES ARE ROUGHLY 1.

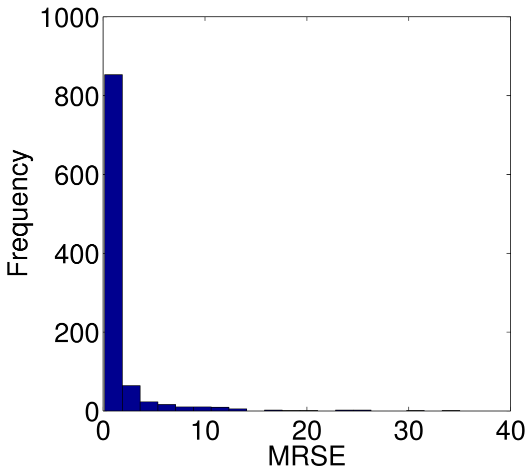


Fig. 4. MRSE variation around the predicted TM with $\beta = 0.2$. The predicted TM $\mathbf{T}$ is from Abilene data, the average traffic over time 02:00 to 02:05 hours, 1st March 2004. The histogram is based on 1000 trials. Although many synthetic TMs are close to the predicted TM, there is a noticeable variability from the predicted TM.

criteria: admissibility, centredness, controllability and whether they satisfy PEVA. For a fair comparison, their parameters were chosen so that their MRSEs are roughly 1, by setting parameters to the values in the second column of the table. We conducted 10,000 trials of each model for testing.

Only SANM satisfies all the required criteria. Also, as seen in the third column, because the other models produce a high number of zeros, convergence of IPF based on their inputs is not guaranteed. For instance, in our trials, the additive model produces 2 inputs that do not converge with IPF. We didn't observe this problem for the multiplicative model, but SANM has a theoretical convergence guarantee that it lacks.

By computing the average of the 10,000 trials, we found that the multiplicative model is almost centred. PEVA is not satisfied for the additive and multiplicative models. Both can be explained by truncation effects: the restriction of TMs to the non-negative plane results in generated entries that are asymmetric around $\mathbf{T}$'s entries, so there are generally more values that are greater than the predicted TM's entries. In contrast, SANM's generated entries are symmetric around $\mathbf{T}$'s entries, as it produces non-negative matrices for IPF.

In practice, the saturation to the gravity model in the high β regime is not a handicap in any way. Since we are mostly interested in variations around $\mathbf{T}$ in the MRSE range $(0, 1]$, so β does not have to exceed 0.2 in most cases.

If there are very large errors present in traffic measurements, then the measurements essentially convey no information. Our model handles large errors gracefully, since it produces a positive matrix almost surely, so IPF is guaranteed to converge to a solution, and that solution is the gravity model which is a reasonable choice in the absence of other information.

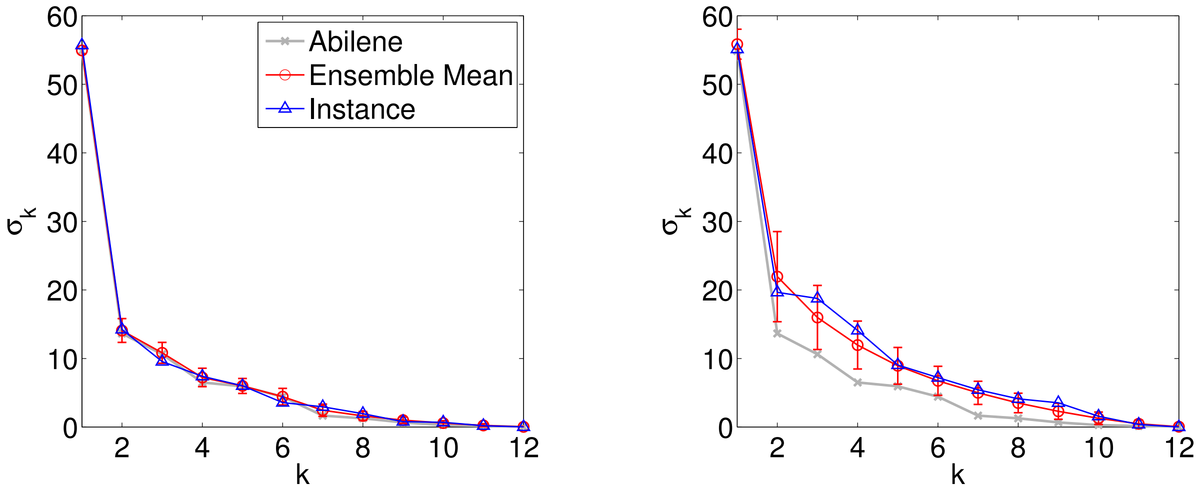
Under more general convex constraints, IPF converges to the maximum entropy model. When very large errors are present, the prior distribution becomes uninformative and is no better than the uniform distribution, so the KL divergence reduces to the Shannon entropy [2]. Minimising the KL divergence to the uniform distribution is equivalent to maximising Shannon entropy.

Real TMs are known to be approximately low rank [35]. We therefore test if our synthetic TMs are approximately low rank. The distribution of the TMs' singular values σ_k are shown in Figure 5 for $\beta = 0.2$ and 1, representing low- and high-noise regimes. The predicted TM $\mathbf{T}$ from the Abilene data has approximately low rank. We see that in both cases, the generated TMs have similar rank structure. From the bars *i.e.,* the 95% confidence level, we also see that there is more variation in the singular values when $\beta = 1$. Also shown, in both cases, are the singular values of an instance of a synthetic TM from SANM, which lie within the 95% confidence level.

We see here that if $\mathbf{T}$ is approximately low rank, and if the noise does not change the approximate low-rank nature, the generated TMs are approximately low rank. Moreover, even with $\beta = 1$, the TMs remain approximately low rank, though the singular values no longer track those of $\mathbf{T}$'s as closely. Unless β is set to extremely high values, the approximate low-rank property should remain as long as $\mathbf{T}$ itself is approximately low rank. Though not shown here due to space constraints, the property is preserved for various other TMs taken from Abilene data.

V. CONCLUSION

Predictions of a TM, although useful, include errors which may impact the applications they are used for. In this paper, we proposed a method to generate synthetic TMs around a predicted TM to account for the variations due to prediction errors. The method is admissible, centred and more importantly, controllable by a single parameter. Our method is also fast, satisfies the PEVA property, and preserves the rank structure of real TMs on which we validated our method. The method has been applied to the design of networks and their sensitivity analysis to TM variations, as well as the design of networks based on risk measures in [29]. Future work includes generating ensembles of TMs satisfying prescribed moment information and studying IPF's action on random non-negative matrices.



(a) $\beta = 0.2$.

(b) $\beta = 1$.

Fig. 5. Comparing the ensemble mean singular values σ_k of TMs generated by SANM to the predicted TM $\mathbf{T}$ from Abilene and an instance of a single synthetic TM from SANM. The bars represent 95% confidence intervals on the means. The predicted TM $\mathbf{T}$ is from Abilene data, the average traffic over time 02:00 to 02:05 hours, 1st March 2004. All generated TMs are approximately low rank, and the singular values in the low noise regime (a) are closer to the predicted TM's than in the high noise regime (b). Note that the singular values of a synthetic TM instance is well-within the 95% confidence interval.

ACKNOWLEDGMENT

This work was conducted while the authors were supported by the Australian Research Council Centre of Excellence Mathematical and Statistical Frontiers (ACEMS) and grant DP110103505. The authors thank eResearch South Australia (eRSA) for the use of their computational resources in the authors' experiments, and Abilene for providing data.

APPENDIX A
ASYMPTOTICS OF THE SANM

Here we obtain intuition on the average behaviour of the SANM in the limit of large β. The limit corresponds to the case where we have very inaccurate TM predictions. The matrices must still satisfy admissibility constraints: non-negativity, row and columns sums, and conservation, but otherwise we have essentially no information about these matrices. In this limit, it would make sense to use a maximum entropy model, that is, to use a model that postulates the minimum additional assumptions about the traffic matrix, in the absence of any prior beliefs beyond the explicit constraints. Prior work [17], [34] has shown that the maximum entropy model for a TM under these constraints is the gravity model. We indeed found that this was a reasonable approximation for the limiting distribution of our matrices, and so we seek to explain it intuitively in this appendix.

We are interested in $\mathsf{E}[\mathbf{S}]$ when $\beta \to \infty$. The major complication is the nonlinear action of IPF on $\mathbf{Y}$, making it difficult to analyse the output of IPF. Moreover, since the input are random matrices, $D_{GKL}(\mathbf{X}\,||\,\mathbf{Y})$ is itself a random quantity (in this sense, it is not the same as the canonical KL divergence), so minimising $D_{GKL}(\mathbf{X}\,||\,\mathbf{Y})$ is only deterministic conditioned on knowing $\mathbf{Y}$.

Recall that IPF is minimising (5), so

$$\mathsf{E}[\mathbf{S}] = \mathsf{E}\left[\operatorname*{argmin}_{\mathbf{X}\in\mathbb{R}^{N\times N}} D_{GKL}(\mathbf{X}\,||\,\mathbf{Y})\right], \tag{8}$$

subject to the usual admissibility constraints.

First, as $\beta \to \infty$, the predicted traffic matrix has an insignificant effect on the matrix $\mathbf{Y}$, so we find that the $y_{i,j}$ are distributed as scaled χ^2 random variables, with one degree of freedom, and that $\mathsf{E}[y_{i,j}] \simeq \beta^2$, *i.e.,* (3) leads to

$$y_{i,j} \simeq \beta^2 z_{i,j}^2, \tag{9}$$

where $z_{i,j} \sim \mathcal{N}(0,1)$, and hence $\mathsf{E}[\mathbf{Y}] = \beta^2 \mathbf{1}_N \mathbf{1}_N^{\mathrm{T}}$, where $\mathbf{1}_N$ is the vector of N ones.

We next show that β does not affect the final IPF solution beyond the first half step. Let $\text{Beta}(a,b)$ denote the Beta distribution with scale parameters $a, b > 0$. We express (9) as

$$y_{i,j} \simeq \beta^2 X_{i,j}, \tag{10}$$

where $X_{i,j} := z_{i,j}^2$. In the first half step *i.e.,* $k = 1/2$, we have, $\forall i$,

$$s_{i,j}^{(1/2)} = r_i \frac{\beta X_{i,j}}{\beta \sum_{j=1}^{N} X_{i,j}} = r_i \frac{X_{i,j}}{\sum_{j=1}^{N} X_{i,j}},$$

where the random variable

$$\frac{X_{i,j}}{\sum_{j=1}^{N} X_{i,j}} \sim \text{Beta}\left(\frac{1}{2}, \frac{N-1}{2}\right).$$

The mean of this quantity is $1/N$, so $\mathsf{E}\left[s_{i,j}^{(1/2)}\right] = r_i/N$. Clearly, after this half step onwards, the parameter β has been cancelled out, so we don't need to consider it henceforth.

It is may be possible to compute the distribution of the output of IPF beyond step 1/2, but this would entail deriving the distribution of the ratios of random variables (the distribution of the ratio of Beta distributed random variables is needed), but this distribution does not have a closed form.

Instead, we consider the problem

$$\tilde{\mathbf{S}} = \operatorname*{argmin}_{\mathbf{X}\in\mathbb{R}^{N\times N}} D_{GKL}\left(\mathbf{X}\,\|\,\mathsf{E}[\mathbf{Y}]\right), \tag{11}$$

subject to the admissibility constraints, which is the solution of the optimisation problem for the average matrix $\mathsf{E}[\mathbf{Y}]$. Since $\mathsf{E}[\mathbf{Y}] \simeq \beta^2 \mathbf{1}_N \mathbf{1}_N^{\mathrm{T}}$ and is therefore a single rank matrix, then by the discussion in Section III, $\tilde{\mathbf{S}} = \mathbf{r}\mathbf{c}^{\mathrm{T}}/T$.

The solutions of (8) and (11) are not equal, but we can argue that they are close in the limit. First, note that the $y_{i,j}$ are IID random variables, so $\mathbf{Y}$ is composed of an ensemble of N^2 IID random variables. The Asymptotic Equipartition Theorem [2, Ch. 3], guarantees that asymptotically the *typical* set of instances of $\mathbf{Y}$ will have probability near one, and elements of this set will have approximately constant probability. So we can loosely think of the ensemble $\mathbf{Y}$ as being largely composed of typical instances, each equiprobable. The constraints in the problem are linear, so the space onto which we are projecting is a linear subspace of $\mathbb{R}^{N\times N}$. Moreover, the optimisation is to minimise the "distance" to this subspace, and so we can think of it as projecting the ensemble $\mathbf{Y}$ onto the subspace. Finally expectation is a linear operator itself, so for N^2 large and $\beta \to \infty$, we have $\tilde{\mathbf{S}} \sim \mathsf{E}[\mathbf{S}]$.

Intuitively, once the noise overwhelms any prior beliefs of the network's traffic obtained from the predicted matrix, the IPF defaults to a solution with maximum entropy, which matches the empirical results.

References

[1] R. J. Beckman, K. A. Baggerly, and M. D. McKay. Creating synthetic baseline populations. *Transpn. Res. A*, 30(6):415–429, 1996.

[2] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley and Sons, Inc., 2nd edition, 2006.

[3] I. Csiszár. i-divergence of geometry of probability distributions and minimization problems. *Ann. Prob.*, 3(1):146–158, 1975.

[4] J. Dattorro. *Convex Optimization and Euclidean Distance Geometry*. Meboo, 2010.

[5] W. E. Deming and F. F. Stephan. On a least squares adjustment of a sampled frequency table when the expected marginal totals are known. *Ann. Math. Stat*, 11(4):427–444, 1940.

[6] G. Duguay, W. Jung, and D. McFadden. *SYNSAM: A Methodology for Synthesizing Household Transportation Survey Data*. Urban Travel Demand Forecasting Project, Institute of Transportation Studies, 1976.

[7] V. Erramilli, M. Crovella, and N. Taft. An independent-connection model for traffic matrices. In *ACM IMC 2006*, pages 251–256, October 2006.

[8] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, J. Rexford, and F. True. Deriving demands for operational IP networks: Methodology and experience. *IEEE/ACM Trans. Networking*, 9:265–280, June 2001.

[9] S. E. Fienberg. An iterative procedure for estimation in contingency tables. *Ann. Math. Stat.*, 41(3):907–917, 1970.

[10] O. Goldschmidt. ISP backbone inference methods to support traffic engineering: Methodology and experience. In *Internet Statistics and Metrics Analysis (ISMA) Workshop*, December 2000.

[11] S. Kullback. *Information Theory and Statistics*. John Wiley and Sons, Inc., 1959.

[12] G. Liang, B. Yu, and N. Taft. A fast lightweight approach to origin-destination IP traffic estimation using partial measurements. *IEEE/ACM Trans. Networking*, 14:2634–2648, June 2006.

[13] A. Medina, N. Taft, K. Salmatian, S. Bhattacharyya, and C. Diot. Traffic matrix estimation: Existing techniques and new directions. In *ACM SIGCOMM*, 2002.

[14] NLANR. Abilene Trace Data. http://pma.nlanr.net/Special/ipls3.html.

[15] I. Norros. On the use of fractional Brownian motion in the theory of connectionless networks. *IEEE Journal on Selected Areas in Communications*, 13(6):953–961, 1995.

[16] A. Nucci, A. Sridharan, and N. Taft. The problem of synthetically generating IP traffic matrices: Initial recommendations. *SIGCOMM Comput. Commun. Rev.*, 35:19–32, July 2005.

[17] K. N. Oikonomou. Analytic forms for most likely matrices derived from incomplete information. *Int. J. Systems Science*, 43(3):443–458, September 2010.

[18] F. Pukelsheim. Biproportional scaling of matrices and the iterative proportional fitting procedure. *Ann. Oper. Res.*, 215(1):269–283, April 2014.

[19] M. Roughan. Simplifying the synthesis of Internet traffic matrices. *SIGCOMM Comput. Commun. Rev.*, 35(5):93–96, 2005.

[20] M. Roughan. Robust network planning. In C. R. Kalmanek, S. Misra, and R. Yang, editors, *The Guide to Reliable Internet Services and Applications*, chapter 5, pages 137–177. Springer, 2010.

[21] M. Roughan and J. Gottlieb. Large-scale measurement and modeling of backbone Internet traffic. In *SPIE ITCom*, Boston, 2002.

[22] M. Roughan, A. Greenberg, C. Kalmanek, M. Rumsewicz, J. Yates, and Y. Zhang. Experience in measuring backbone traffic variability: Models, metrics, measurements and meaning. In *ACM IMW*, 2002.

[23] M. Roughan, M. Thorup, and Y. Zhang. Traffic engineering with estimated traffic matrices. In *ACM IMC 2003*, pages 248–258, October 2003.

[24] R. Sinkhorn and P. Knopp. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 21(2), 1967.

[25] L. Smith, R. Beckman, K. Baggerly, D. Anson, and M. Williams. *TRANSIMS: TRansportation ANalysis and SIMulation System: Project Summary and Status*. NASA Open Source Agreement Version 1.3, 1995.

[26] A. Soule, A. Nucci, R. Cruz, E. Leonardi, and N. Taft. How to identify and estimate the largest traffic matrix elements in a dynamic environment. In *ACM SIGMETRICS 2004*, pages 73–84, June 2004.

[27] R. Teixera, N. Duffield, J. Rexford, and M. Roughan. Traffic matrix reloaded: Impact of routing changes. In *Workshop on Passive and Active Measurements*, 2005.

[28] P. Tune and M. Roughan. Internet traffic matrices: A primer. In H. Haddadi and O. Bonaventure, editors, *Recent Advances in Networking, Vol. 1*. ACM SIGCOMM, August 2013.

[29] P. Tune and M. Roughan. Network-design sensitivity analysis. In *ACM SIGMETRICS 2014*, pages 449–461, June 2014.

[30] P. Tune and M. Roughan. Spatiotemporal traffic matrix synthesis. In *ACM Sigcomm*, London, UK, August 2015.

[31] P. Tune and M. Roughan. Spatiotemporal traffic matrix synthesis. In *ACM SIGCOMM 2015*, pages 579–592, August 2015.

[32] Y. Vardi. Network Tomography: Estimating source-destination traffic intensities from link data. *J. Am. Statist. Assoc.*, 91:365–377, 1996.

[33] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg. Fast accurate computation of large-scale IP traffic matrices from link loads. In *ACM SIGMETRICS 2003*, pages 206–217, 2003.

[34] Y. Zhang, M. Roughan, C. Lund, and D. Donoho. Estimating Point-to-Point and Point-to-Multipoint traffic matrices: An information-theoretic approach. *IEEE/ACM Trans. Networking*, 13(5):947–960, October 2005.

[35] Y. Zhang, M. Roughan, W. Willinger, and L. Qiu. Spatio-Temporal compressive sensing and Internet traffic matrices. In *ACM SIGCOMM*, pages 267–278, August 2009.

[36] R. Zhang-Shen and M. McKeown. Designing a predictable Internet backbone with Valiant load-balancing. In *Proceedings of the 13th International Conference on Quality of Service (IWQoS)*, pages 178–192, 2005.

Paul Tune Paul Tune is at Image Intelligence. His research has focused on Internet traffic modelling, notably on the generative models for synthesising large ensembles of traffic matrices. He received his PhD from the University of Melbourne in 2010, under the supervision of Professor Darryl Veitch. His other research interests includes compressive sensing, information theory and signal processing.

Matthew Roughan (M97-SM09) received the Ph.D. degree in applied probability from the University of Adelaide, Adelaide, Australia, in 1994. He joined the School of Mathematical Sciences, University of Adelaide, in 2004. Prior to that, he was with AT&T Labs - Research, Florham Park, NJ. His research interests lie in measurement and modeling of the Internet, and his background is in stochastic modeling.