



ESTIMATION OF TRAFFIC MATRIX FROM LINKS LOAD USING GENETIC ALGORITHM

JOSEPH L PACHUAU, ARNAB ROY, GOPAL KRISHNA AND ANISH KUMAR SAHA *

Abstract. Traffic Matrix (TM) is a representation of all traffic flows in a network. It is helpful for traffic engineering and network management. It contains the traffic measurement for all parts of a network and thus for larger network it is difficult to measure precisely. Link load are easily obtainable but they fail to provide a complete TM representation. Also link load and TM relationship forms an under-determined system with infinite set of solutions. One of the well known traffic models Gravity model provides a rough estimation of the TM. We have proposed a Genetic algorithm (GA) based optimization method to further the solutions of the Gravity model. The Gravity model is applied as an initial solution and then GA model is applied taking the link load-TM relationship as a objective function. Results shows improvement over Gravity model.

Key words: Traffic Matrix, Traffic Engineering, Gravity model, Link loads, Optimization, Genetic Algorithm, IP networks.

AMS subject classifications. 68M14

1. Introduction. Traffic matrix is a representation of traffic volume flowing between node pair in a network. It plays an important role in traffic engineering. Traffic matrix helps network manager in task like load balancing, network optimization, anomaly detection etc. The ingress-egress traffic matrix is collected between ingress and egress routers in the network. Origin destination matrix measures the traffic flow from actual source destination. The point where the packages are created and where they are received. For large IP networks this produces an extremely large and sparse matrix. For such a case an aggregated IP or blocks of IP may be considered as a single point. Direct measurement of traffic matrix requires placement of flow measurement at each ingress/egress points. This is impractical for large IP networks in terms of cost, time and effort [1]. As a result, several approaches have been implemented to estimate or model the traffic matrix from other available measurements. Link measurements provides traffic data for each link in a network. These measurements are easily collected from routers using SNMP (Simple network management protocol). In [2], authors proposed the method of network tomography, where used of traffic matrix is estimated from link measurements and routing information. The link and traffic matrix relationship has greater number of unknown than the number of equations. Thus, it cannot be solved for a unique solution. Approaches like Bayesian and Expectation Maximization model estimates the TM from statistical features of the traffic data [3][4]. These approaches take into account nature of traffic with changes in time. Spatial estimation of traffic matrix ignores the temporal features and works for a single time instance of TM matrix. Spatial model like Gravity model, Discrete choice model, independent connections etc. model have been implemented to produce TM. Different traffic model provides useful estimations based on the implementation. But these models are not actual representations and so they have inaccuracy. Several approaches were introduced where the output from these model are improved for higher accuracy. Traffic matrix estimation by using a combination of network tomography and spatial models was proposed in [5]. Estimation method in [6] implemented neural networks to improve TM based on expectation maximization model. Applying optimization methods to estimate and improve traffic matrix accuracy was proven successful in various studies [7, 8, 9]. In this paper we explore the use of GA as an optimization tool. Genetic Algorithm is a soft optimization technique which uses guided random techniques to search for an optimized solution. GA is known to work well for noisy environments and large parameter problems. For traffic matrix estimation using neural networks GA was applied to optimize the weights [10]. GA also found its implementation in Distributed Denial of Service attack. Parameters of traffic matrix were

*Department of Computer Science and Engineering, National Institute of Technology Silchar, India (anishkumarsaha@gmail.com)

optimized using GA to detect the attacks [11]. We proposed the combination of Gravity model and GA optimization to estimate and improve the accuracy of traffic matrix.

2. Related Works. Traffic Engineering (TE) deals with improvement of network performance by analysing traffic data and patterns. Traffic prediction is used in TE to control time varying traffic. Traffic variation are categorised into short-term and long-term variation. The long term variation helps to define the daily behaviour of traffic, while short term variation are handled to avoid congestion. Prediction is used to manage the traffic for optimal link utilization[12]. Dynamic behaviour of routing depends on proper prediction of traffic. In [13], the authors proposed optimization of router deployment depending on traffic flow to improve network efficiency. In their method, traffic flows are directed to a service node for making egress based optimization routing. The network paths are planned to facilitate an effective TE for improvement of quality of network traffic. In [14], a sequence-to-sequence model is proposed to improve the challenges caused by growth of Internet traffic. The sequence-to-sequence model learns forwarding path based on traffic data. In [15], authors investigated various TE techniques to improve different aspects of a network. They proposed Centralized Optimal Traffic Engineering system, Suboptimal Solution, and Distributed & Greedy solution to maximize data delivery. The performance of the network is measured in aggregated network throughput, average end to end delay and flow fairness. In [16], a heuristic approach is proposed to optimize routing over multiple TMs. The routes obtained through this methods are loop free and optimized bandwidth of links.

GA due to it's flexibility is implemented for different problems in networking. In [17], authors give weightage on quality of service for mixed traffic environment. They proposed a scheduling algorithm using GA for optimal resource allocation in a network. The quality of service is taken as fitness function in their proposed GA. The traffic is possible to be classified into different categories as shown in [18]. They used wavelet kernel function with GA for classification. This type of classification helps in intruder detection for unwanted traffics of various applications. There are other example of GA based traffic classifiers like, distance-based, K-Nearest Neighbors, and neural networks. Data points are separated using Mutual information, Dunn, and SD based biased measurement. It helps in Peer to Peer and non Peer to Peer traffic classification [19]. Classification of packets require a robust scheme that provide scalability, reliability and quality services. Feature selection, a classification process, selects relevant features during prediction. Both GA based classification and feature selection are used to classify packets as shown in [20]. An incomplete collection of traffic data degrade the integrity, information and quality. Fuzzy C-Means is an algorithm used to tackle clustering problem of incomplete traffic data. The missing data is filled up with the process called imputation, where estimated data replace the missed data. Fuzzy C-means with GA is making a good hybrid model for traffic data estimation[21].

Effective TE applications requires accurate estimation of Traffic matrix. A model named, network tomography equation establishes a relationship between link measurements and traffic matrix. This relationship is an under determined system. Due to this ill posed problem, numerous works focus on combining network tomography method with other mathematical models. The Gravity model construct TMs by assuming Origin-Destination flows proportional to the incoming and outgoing traffic of nodes. The generalised gravity further improves this process by classifying the traffic flows. The tomogravity model combines gravity and tomography model to increase accuracy of TM. Advanced tomogravity method introduces a relativity factor to further improve the estimation as in [22]. The co-variance method [23] uses co-variance matrix of link count sample to make up for insufficient information. TMs are estimated from the link count covariance matrix. This method provides a light weight estimation consistent with actual link measurements. The Generalized Autoregressive Conditional Heteroscedasticity model [24] deals with the ill posed nature of TMs. It provides a flexible approach to capture self similarity in traffic behaviour. Comparison of different TM estimation method [25] shows that tomogravity and entropy maximization performs better than linear programming approach. Adding extra constraints in entropy maximization further increase the performance. In [26], authors present the use of Simple Network Management Protocol for complete traffic collection with known estimation techniques to improve accuracy. The adaptive information gain maximization also focuses on traffic collection for improvement in accuracy. The most informative flow from traffic is determined for estimation of TMs. This approach increases the accuracy with a small increase in measurement resources [27]. For a large network, estimating TM takes high computation time for which division of network is one of the solutions. In [25], authors proposed divide and conquer method for estimating TM for large size network. The large network is divided into smaller

Table 3.1: An Instance of Traffic Matrix for duration of 5 min. on 01/03/2004

	D ₁	D ₂	D ₃	D ₄	D ₅	D ₆	D ₇	D ₈	D ₉	D ₁₀	D ₁₁	D ₁₂
S ₁	10000	195828	615502	125898	154887	183703	136904	306701	169523	280277	145619	1178115
S ₂	166931	30997781	6106169	1938388	1474276	10256961	1816513	6836966	4801483	533208	3359056	19405592
S ₃	1422635	4775747	10779162	5151845	3686376	9884916	2451656	10415963	5286877	681353	1685721	3992645
S ₄	86552	878056	14444321	1410746	3153286	2702903	1480837	8015838	2146278	5394549	3978549	4593323
S ₅	89641	1108792	6334313	1384882	2102362	3428964	2673813	36921723	2724474	1105266	839099	3189994
S ₆	1787597	3470525	45766716	8016824	12440169	8568396	3709065	9151891	15231037	1299895	5209570	15426785
S ₇	157860	1666336	10114602	2022864	2053539	3006659	1508587	3252594	4815904	458907	916602	4518164
S ₈	127373	7272971	33646381	3389768	3386575	16020094	5088844	4263424	22936657	866828	9569795	20897395
S ₉	1461615	15332940	20127858	6129395	8995420	31247043	9287731	26633460	51298517	3596757	8225459	41947778
S ₁₀	10000	390452	1892275	3902592	538756	1448104	786402	750079	829298	5351103	1236618	745390
S ₁₁	41632	5955580	8442220	1627941	4238538	2884194	847818	6662390	9317015	1783127	389384	3944017
S ₁₂	4207163	47226648	24952949	13523783	5789742	23543180	12241025	34378360	50123027	742716	11160076	70370056

parts then combined them for the full estimation of TM. This approach deals with smaller routing table and allows parallel processing which improves computational time. Artificial Neural Network approach is possible to improve estimation of TM. In [28], authors proposed Recurrent Neural Network approach for estimation of TM. The Recurrent Neural Network takes real world data for training of the model to extract spatio-temporal features of TM. Convolutional neural network is another finding for spatial relationship between link loads and Origin-Destination flows[29]. These artificial neural networks approach are good in finding hidden features of TM, although it requires large training data. TM estimation often includes aggregation of large traffic flows which is a difficult task at real time. In [30], authors proposed distributed Map Reduce approach for aggregation of large traffic flows. The Map Reduce approach uses topology information to identify origin-destination links, for easier aggregation of traffic flows. The input for the Map reduce is collected using a big data streaming module and the result shows near real time estimation of TM.

3. Proposed model for GA based Traffic Matrix Estimation.

3.1. Basics of Traffic Matrix Model. We have used a well known network for TM estimation named Abilene network collected by Y.Zhang, which contains data set of actual traffic matrix of the Abilene network for 6 months [31]. Each instance of TM collects 5 minutes of traffic between all source and destination. The routing matrix and topology are included in the data set. It consist of 12 nodes and 54 links. There are 32 uni-directional links between nodes and each nodes has an inbound and outbound link, therefore a total of 54 links. The TM of an instance X_i is a 2D matrix of 12×12 in size, an example is shown in Table 3.1. Each value x_{ij} in the matrix denotes the traffic flow from source S_i to destination D_j . With elapsed time t , the traffic matrix extend to a complete 3D traffic matrix as $X_1, X_2, X_3, \dots, X_i$.

For determining TM X , two inputs are required namely, routing matrix A and link loads Y respectively. The relationship among them is as follows,

$$AX = Y \quad (3.1)$$

For this, all links load are measured for a network, from which different models are applied to have more precise TM values. In Abilene network, 54 numbers of link loads are available as seen 1D vector. Link loads are direct count of packets for a link. Abilene network has 144 (12×12) numbers of all possible combination of {Source, Destination} for all 12 nodes and 54 numbers of various links. Thus making the routing matrix A of $\{144 \times 54\}$ in dimension size. The entries of A are as follows,

$$A_{ij} = \begin{cases} 1, & \text{if link } i \text{ is utilized in } \{\text{Source-Destination}\} j \text{ path} \\ 0, & \text{otherwise} \end{cases} \quad (3.2)$$

Eq. 3.1 is a linear system where the number of unknowns are larger than the number of equations. There are multiple solutions for the value of X . Proper use of mathematical models such like, Gravity model, provides a close estimation, yet further improvement is possible in the solution.

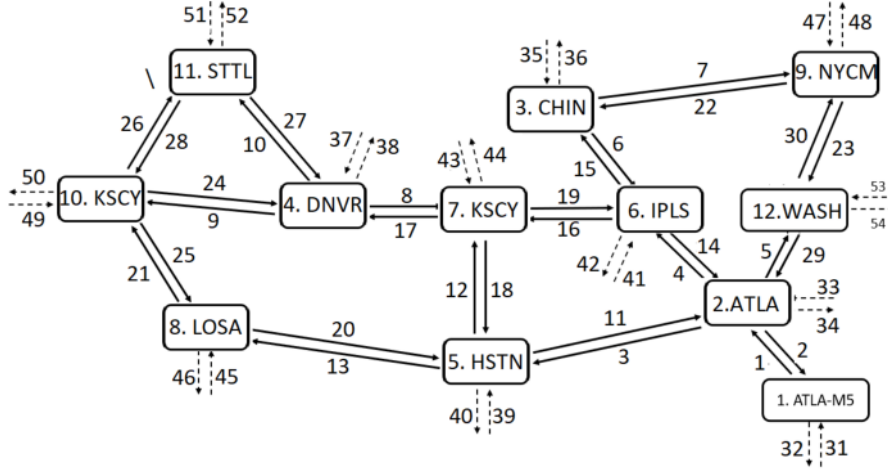


Fig. 3.1: Abilene network topology with various link traffic direction

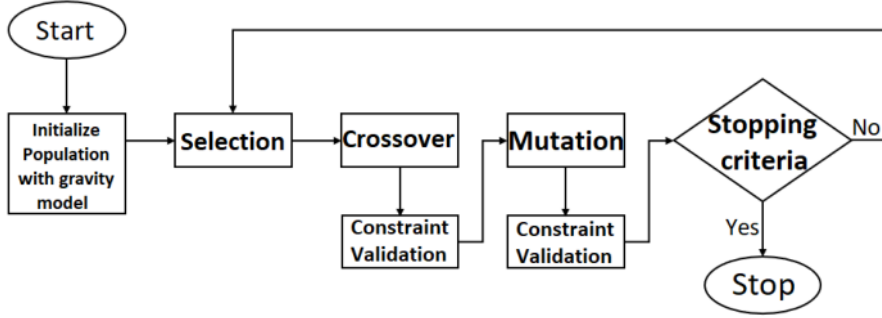


Fig. 3.2: Flowchart for proposed GA

3.2. Basics of Gravity Model. The Gravity model as the name suggest its an adaptation of Newton's Gravity model. This model is applied in different transportation problems, like road traffic, goods etc. In internet TM also it is applied to estimate traffic matrix by taking the ingress and egress flow of a node [32]. Each entry of the TM, X_{ij} represent the traffic from node i to node j . It is considered proportional to all traffic flowing out from i , T_i^{out} and all traffic flowing in to j , T_j^{in} . The total traffic $T^{total} = \sum_{k=1}^n T_k^{in} = \sum_{k=1}^n T_k^{out}$, is used for normalization. The expression for the TM entry is as follows:

$$X_{ij} = \frac{T_i^{out} \times T_j^{in}}{T^{total}} \quad (3.3)$$

where n is number of nodes in the network and the output is a traffic matrix of $X\{n \times n\}$ dimension, where $1 \leq i \leq n$ and $1 \leq j \leq n$.

3.3. Proposed TM model. The Concept of GA, was introduced as an optimization technique that models after the evolution of biological being in the natural world. GA is a heuristic random search that finds sub-optimal solution. The first step of GA is initialization, where a set of random feasible solution for the optimization is generated. GA basic operators, like crossover and mutation are applied to form new population. The best solutions are selected from the population. The operations are repeated until the result converged towards a sub-optimal solution [33]. The gravity model solution is possible to further optimize using GA to

closer match the link measurements. The optimization with the help of Eq. 3.1 is defined as follows:

$$\begin{aligned} \text{Min.} \quad & \|AX - Y\| + w \|X - X_{GM}\| \\ \text{Subject to} \quad & \sum_{j=1}^n x_{ij} = I_i \\ & \sum_{i=1}^n x_{ij} = J_j \\ & x_{ij} \geq 0 \quad i = 1 \dots n \ \& \ j = 1 \dots n. \end{aligned} \quad (3.4)$$

where, I_i and J_j are the total incoming and outgoing traffic for node i and node j respectively. The first part of the objective function gives the distance of calculated link load from the link measurement, while the second part calculates the TM solution distance from the Gravity model. The w is a weight applied to the distance with gravity model in order to adjust the significance of Gravity model in the optimization. First and second constraint maintains the conservation of traffic, i.e. total traffic is constant. Third constraint makes sure that the traffic between any two nodes is always positive. The details of the proposed GA operators, shown in Fig. 3.2, for optimization of Eq. 3.4 are discussed below.

3.3.1. Initialization. We need a set of random solution known as initial population. For making initial population, a first individual is generated from Gravity model. Then after the remaining population are generated by making random changes to the first individual while maintaining all constraints. For this we introduce a matrix Δ of dimension $\{n \times n\}$ which is defined as:

$$\Delta = \begin{bmatrix} [B] & -[B] \\ -[B] & [B] \end{bmatrix}. \quad (3.5)$$

B is a matrix of size $(\frac{n}{2} \times \frac{n}{2})$, where each element b_{ij} takes a random value from the set $\{-1, 1\}$. Therefore Δ contains random elements of 1 or -1, and due to its arrangement given in (1), the sum of each row and column is equal to 0. A random multiplier, r is generated to calculate random solution. If X_{GM} represents the TM obtained by gravity model, then the rest of individual can be calculated as,

$$X = X_{GM} + r\Delta \quad (3.6)$$

Eq. 3.6 is possible to evaluate with multiple repetition for different values of r for generating multiple individuals. Since the sum of rows and columns of Δ is 0, the new individuals satisfy the first and second constraint.

3.3.2. Selection. Individuals for crossover are selected using the selection operator. For this purpose a roulette wheel is implemented. Roulette wheel as the name suggest takes inspiration from the game. The individuals are placed on the wheel as a pie chart structure. The fitter individuals occupy larger space on the wheel. A fixed pointer is placed on the wheel. When the wheel is spun the individual that the pointer lands on is selected. The roulette wheel provides a random selection while giving preferences to the better solution.

3.3.3. Crossover. The crossover takes two individuals and exchange information between them to form new individuals or offspring. Two individuals P and Q are selected as parents where P is the fitter parent and the generated offspring are C and D . A random gene location with index i and j is selected to apply crossover as below,

$$\begin{aligned} c_{ij} &= p_{ij} + \beta \times (p_{ij} - q_{ij}) \\ d_{ij} &= q_{ij} + \beta \times (p_{ij} - q_{ij}) \end{aligned} \quad (3.7)$$

where β is a random number in the range of $[0, b]$. The remaining gene locations in C and D are directly copied from P and Q respectively. This crossover is known as the heuristic crossover [34]. Other crossovers like, average, arithmetic, blend etc., are also applicable in this process. The heuristic crossover is suitable here for finding a new solution closer to the fitter parent and the range of values for child genes are adjustable with b . The same is shown in Fig. 3.3.

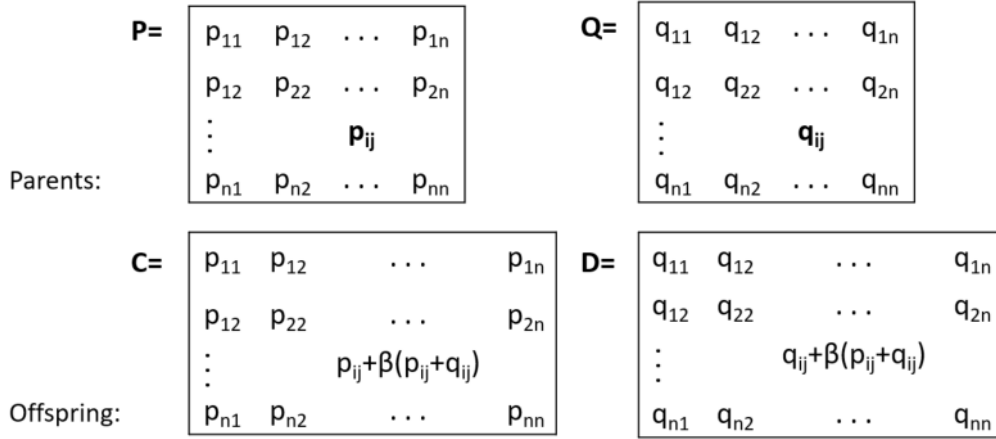


Fig. 3.3: Crossover operation

3.3.4. Mutation. The Mutation makes random change in gene values. This process is generally used to prevent the solutions converging to a local optimum point or premature convergence. For an individual P mutation is performed to produce one offspring C as follows,

$$c_{ij} = \alpha \times p_{ij} \quad (3.8)$$

where α is a random number in the range of $[0, a]$. The gene location with index i and j are randomly selected. The rest of the gene remain unchanged. Here, the rate of mutation is kept to 0.2.

3.3.5. Constraint Validation. The crossover and mutation when applied to a single gene value leads to violation in first and second constraints. These violation of constraints are handled using direct method as explained in [35]. The offspring of crossover and mutation are processed for constraint validation. If x'_{ij} and x_{ij} denotes the old and updated values for offspring X, the change in gene value is calculated as $\delta = x_{ij} - x'_{ij}$. Changes are made in X as follows:

$$x_{kl} = \begin{cases} x_{kl}, & \text{if } i = k \text{ and } j = l \\ x_{kl} + \frac{\delta}{n+1}, & \text{if either } i = k \text{ or } j = l \\ x_{kl} + \frac{\delta}{(n+1)^2}, & \text{otherwise} \end{cases} \quad (3.9)$$

where $k = 1 \dots n$ and $l = 1 \dots n$. This maintains the traffic conservation of first two constraints. Constraint 3 is handled by assigning a penalty to the fitness function when constraint is violated.

4. Results and Analysis. The results obtained from Gravity model and the proposed GA for the same traffic are compared. The traffic is collected from Abilene network available in [31]. The error is calculated using Root Mean Square Error(RMSE). The RMSE is calculated as follows,

$$RMSE = \sqrt{\frac{1}{n} (X_{estimated} - X_{raw})^2} \quad (4.1)$$

Population size plays an important role in the proposed model. The algorithm is executed for different population size for the same time instance $t = 1$ as shown in Figure 4.1. It is observed that above 300, almost the RMSE is reaching to lowest value. A population size of 300 is taken as the optimal size. Figure 4.2 shows the comparison of TM values for the proposed GA and Tomogravity proposed in [5]. We observe that the GA estimation are closer to the real value, which is represented as a diagonal line. The RMSE and average error comparison is shown in Table 4.2. It is observed that the RMSE value shows good improvement while

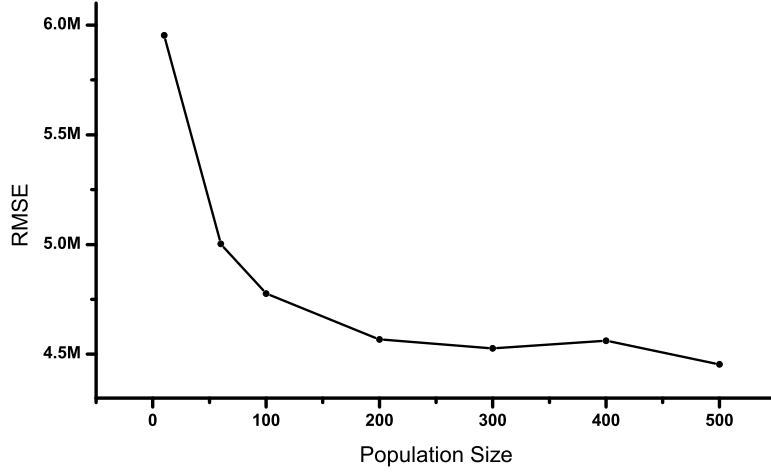


Fig. 4.1: Output of proposed GA with Population size for time instance, t=1.

Table 4.1: Effects of parameters in GA operators.

(a) Effect of α range in mutation.

Value of a	Generation number
0.25	417
0.5	344
1	247
2	253
4	280
8	329
16	365

(b) Effect of β range in crossover.

Value of b	Generation number
1	678
2	337
4	291
8	240
16	237
32	251
64	287

the average error improvement is lesser. This shows that the proposed GA estimation give higher accuracy for large traffic flows which contribute more to the entire traffic. The TM estimation for time instances $t = 1$ to 50 shows that the proposed GA provides improved results over the initial Gravity model and the Tomogravity results as seen in Figure 4.3.

The effectiveness of mutation and crossover decides the performance of GA. Varying the range of the random multiplier α and β from the range $[0, a]$ and $[0, b]$ affects the performance of the proposed GA in mutation and crossover respectively. The optimal value of the proposed GA for different values of a and b are shown in Table 4.1. The proposed algorithm is executed with different values of a and b , and the outcomes are measured in the number of generation to converge to an optimized solution. For mutation Table 4.1a, $a = 1$ shows the best result, obtaining the solution at 247 generations. For crossover Table 4.1b, $b = 16$ shows the best result, obtaining the solution at 237 generations.

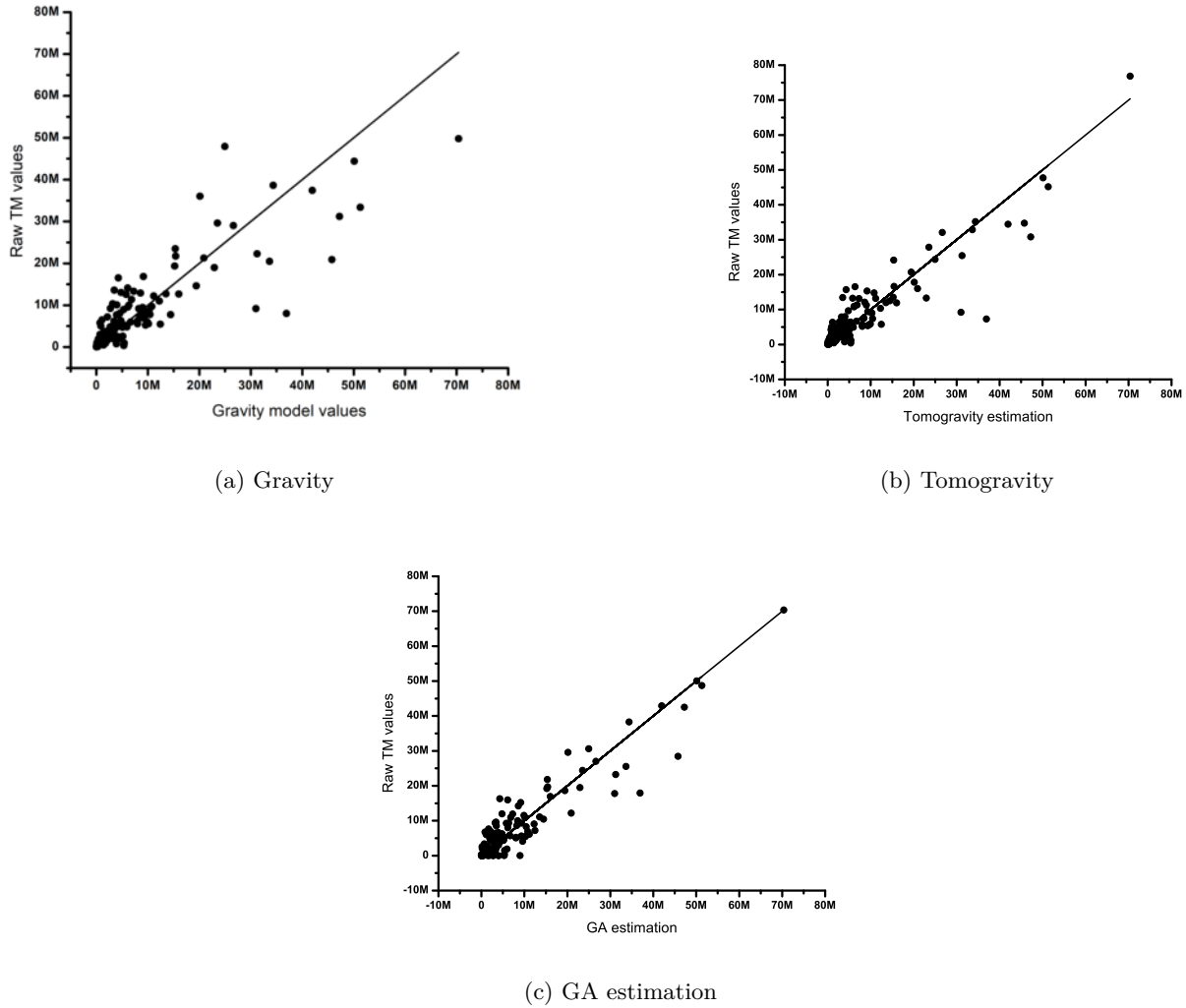


Fig. 4.2: Comparison of various TM to raw value for instance t=1.

Table 4.2: Comparison of RMSE and Average error TM instance t=1.

	Gravity Model	Tomogravity	GA estimation
RMSE	6.1510×10^6	4.7038×10^6	4.2787×10^6
Average error	3.4572×10^6	2.6660×10^6	2.7638×10^6

5. Conclusion. Traffic matrix calculation for large IP network is a difficult task due to insufficient information. In this paper we have proposed a way to improve estimation for traffic matrix. The gravity model provides a reasonable estimation but can be improved with GA optimization. Higher population size provides a more diverse population thus allowing for better exploration of the search space. As the TM consist of 144 elements, such a high dimensional search space requires a larger population size. The GA estimation gives reasonable result at population size beyond 100, and as it further increases beyond 300 no significant improvement

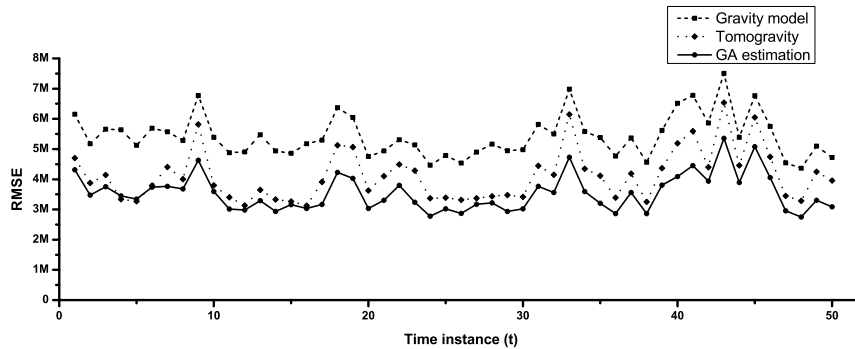


Fig. 4.3: RMSE of Gravity model and proposed GA for different time instances $t=1$ to 50 & fixed population size= 300.

is observed. Our proposed model shows improvement over the gravity model in both RMSE and average error, while comparison with tomogravity results shows improvement in RMSE value shows better improvement than the average error. This shows that the propose GA estimation is more sensitive to the larger traffic values which are more significant to the overall matrix. Overall the proposed model provides improvement for traffic matrix model and theoretically it can be used with any TM model by replacing the Gravity model for an improved result.

REFERENCES

- [1] P. TUNE, M. ROUGHAN, H. HADDADI, AND O. BONAVENTURE, "Internet traffic matrices: A primer," *Recent Advances in Networking*, vol. 1, pp. 1–56, 2013.
- [2] Y. VARDI, "Network tomography: Estimating source-destination traffic intensities from link data," *Journal of the American statistical association*, vol. 91, no. 433, pp. 365–377, 1996.
- [3] C. TEBALDI AND M. WEST, "Bayesian inference on network traffic using link count data," *Journal of the American Statistical Association*, vol. 93, no. 442, pp. 557–573, 1998.
- [4] J. CAO, D. DAVIS, S. VANDER WIEL, AND B. YU, "Time-varying network tomography: router link data," *Journal of the American statistical association*, vol. 95, no. 452, pp. 1063–1075, 2000.
- [5] Y. ZHANG, M. ROUGHAN, N. DUFFIELD, AND A. GREENBERG, "Fast accurate computation of large-scale ip traffic matrices from link loads," *ACM SIGMETRICS Performance Evaluation Review*, vol. 31, no. 1, pp. 206–217, 2003.
- [6] H. ZHOU, L. TAN, Q. ZENG, AND C. WU, "Traffic matrix estimation: A neural network approach with extended input and expectation maximization iteration," *Journal of Network and Computer Applications*, vol. 60, pp. 220–232, 2016.
- [7] S. EUM, J. MURPHY, AND R. HARRIS, "A fast accurate lp approach for traffic matrix estimation," in *International Teletraffic Congress ITC19*, 2005.
- [8] O. GOLDSCHMIDT, "Isp backbone traffic inference methods to support traffic engineering," in *Internet Statistics and Metrics Analysis (ISMA) Workshop*, pp. 1063–1075, 2000.
- [9] D. JIANG, X. WANG, AND L. GUO, "An optimization method of large-scale ip traffic matrix estimation," *AEU-International Journal of Electronics and Communications*, vol. 64, no. 7, pp. 685–689, 2010.
- [10] A. OMIDVAR AND H. SHAHJOSEINI, "Intelligent ip traffic matrix estimation by neural network and genetic algorithm," in *2011 IEEE 7th International Symposium on Intelligent Signal Processing*, pp. 1–6, IEEE, 2011.
- [11] S. M. LEE, D. S. KIM, J. H. LEE, AND J. S. PARK, "Detection of ddos attacks using optimized traffic matrix," *Computers & Mathematics with Applications*, vol. 63, no. 2, pp. 501–510, 2012.
- [12] T. OTOSHI, Y. OHSITA, M. MURATA, Y. TAKAHASHI, K. ISHIBASHI, AND K. SHIOMOTO, "Traffic prediction for dynamic traffic engineering," *Computer Networks*, vol. 85, pp. 36–50, 2015.
- [13] J. SUN, S. SUN, K. LI, D. LIAO, A. K. SANGAIAH, AND V. CHANG, "Efficient algorithm for traffic engineering in cloud-of-things and edge computing," *Computers & Electrical Engineering*, vol. 69, pp. 610–627, 2018.
- [14] Y. ZUO, Y. WU, G. MIN, AND L. CUI, "Learning-based network path planning for traffic engineering," *Future Generation*

- Computer Systems*, vol. 92, pp. 59–67, 2019.
- [15] M. ISLAM, M. A. RAZZAQUE, M. MAMUN-OR-RASHID, M. M. HASSAN, A. ALELAIWI, AND A. ALAMRI, “Traffic engineering in cognitive mesh networks: Joint link-channel selection and power allocation,” *Computer Communications*, vol. 116, pp. 212–224, 2018.
- [16] Y. GUO, Z. WANG, X. YIN, X. SHI, AND J. WU, “Traffic engineering in hybrid sdn networks with multiple traffic matrices,” *Computer Networks*, vol. 126, pp. 187–199, 2017.
- [17] S. H. DA MATA AND P. R. GUARDIEIRO, “Resource allocation for the lte uplink based on genetic algorithms in mixed traffic environments,” *Computer Communications*, vol. 107, pp. 125–137, 2017.
- [18] F. ERTAM AND E. AVCI, “A new approach for internet traffic classification: Ga-wk-elm,” *Measurement*, vol. 95, pp. 135–142, 2017.
- [19] M. MOHAMMADI, B. RAAHEMI, A. AKBARI, H. MOEINZADEH, AND B. NASERSHARIF, “Genetic-based minimum classification error mapping for accurate identifying peer-to-peer applications in the internet traffic,” *Expert Systems with applications*, vol. 38, no. 6, pp. 6417–6423, 2011.
- [20] J. PARK, H.-R. TYAN, AND C.-C. J. KUO, “Ga-based internet traffic classification technique for qos provisioning,” in *2006 International Conference on Intelligent Information Hiding and Multimedia*, pp. 251–254, IEEE, 2006.
- [21] J. TANG, G. ZHANG, Y. WANG, H. WANG, AND F. LIU, “A hybrid approach to integrate fuzzy c-means based imputation method with genetic algorithm for missing traffic volume data estimation,” *Transportation Research Part C: Emerging Technologies*, vol. 51, pp. 29–40, 2015.
- [22] H. ZHOU, L. TAN, F. GE, AND S. CHAN, “Traffic matrix estimation: Advanced-tomography method based on a precise gravity model,” *International Journal of Communication Systems*, vol. 28, no. 10, pp. 1709–1728, 2015.
- [23] I. JUVA, S. VATON, AND J. VIRTAMO, “Quick traffic matrix estimation based on link count covariances,” in *2006 IEEE International Conference on Communications*, vol. 2, pp. 603–608, IEEE, 2006.
- [24] D. JIANG AND G. HU, “Garch model-based large-scale ip traffic matrix estimation,” *IEEE Communications Letters*, vol. 13, no. 1, pp. 52–54, 2009.
- [25] M. M. RAHMAN, S. SAHA, U. CHENGAN, AND A. S. ALFA, “Ip traffic matrix estimation methods: Comparisons and improvements,” in *2006 IEEE International Conference on Communications*, vol. 1, pp. 90–96, IEEE, 2006.
- [26] A. GUNNAR, M. JOHANSSON, AND T. TELKAMP, “Traffic matrix estimation on a large ip backbone: a comparison on real data,” in *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, pp. 149–160, 2004.
- [27] D. LI, C. XING, N. DAI, F. DAI, AND G. ZHANG, “Estimating sdn traffic matrix based on online adaptive information gain maximization method,” *Peer-to-Peer Networking and Applications*, vol. 12, no. 2, pp. 465–480, 2019.
- [28] J. ZHAO, H. QU, J. ZHAO, AND D. JIANG, “Towards traffic matrix prediction with lstm recurrent neural networks,” *Electronics Letters*, vol. 54, no. 9, pp. 566–568, 2018.
- [29] M. EMAMI, R. AKBARI, R. JAVIDAN, AND A. ZAMANI, “A new approach for traffic matrix estimation in high load computer networks based on graph embedding and convolutional neural network,” *Transactions on Emerging Telecommunications Technologies*, vol. 30, no. 6, p. e3604, 2019.
- [30] W. J. QUEIROZ, M. A. CAPRETZ, AND M. A. DANTAS, “A mapreduce approach for traffic matrix estimation in sdn,” *IEEE Access*, vol. 8, pp. 149065–149076, 2020.
- [31] Y. ZHANG, *Abilene Dataset*, <https://www.cs.utexas.edu/~y Zhang/research/AbileneTM/>, 2004.
- [32] M. ROUGHAN, “Simplifying the synthesis of internet traffic matrices,” *ACM SIGCOMM Computer Communication Review*, vol. 35, no. 5, pp. 93–96, 2005.
- [33] M. MITCHELL, *An introduction to genetic algorithms*. MIT press, 1998.
- [34] A. H. WRIGHT, “Genetic algorithms for real parameter optimization,” in *Foundations of genetic algorithms*, vol. 1, pp. 205–218, Elsevier, 1991.
- [35] B. CRAENEN, A. EIBEN, AND E. MARCHIORI, “How to handle constraints with evolutionary algorithms,” *Practical Handbook Of Genetic Algorithms: Applications*, pp. 341–361, 2001.

Edited by: Dana Petcu

Received: Nov 13, 2020

Accepted: Jan 21, 2021