# T-Sample: A Dual Reservoir-based Sampling Method for Characterizing Large Graph Streams

Lingling Zhang, Hong Jiang*, Fang Wang‡, Dan Feng, and Yanwen Xie
Wuhan National Laboratory for Optoelectronics,
Key Laboratory of Information Storage System,
Engineering Research Center of data storage systems and Technology, Ministry of Education of China,
School of Computer Science and Technology, Huazhong University of Science and Technology,
Shenzhen Huazhong University of Science and Technology Research Institute
* Department of Computer Science and Engineering, University of Texas at Arlington, USA
{llzh, wangfang, dfeng, ywxie}@hust.edu.cn; * hong.jiang@uta.edu; ‡ Corresponding author

***Abstract*—Reservoir sampling is widely employed to characterize connectivity of large graph streams by producing edge samples. However, existing reservoir-based sampling methods mainly characterize large graph streams by a measure of counting triangle but perform poorly in accuracy when used to analyze the topological characteristics reflected by node degrees because they produce disconnected edge samples, making them ineffective in many applications that require both types of connectivity estimation simultaneously in real time. This paper proposes a new method, called triangle-induced reservoir sampling, or T-Sample, to produce connected edge samples. While every edge in a graph stream is still processed only once by T-Sample, a dual sampling mechanism performing both uniform sampling and non-uniform sampling is carefully designed with a base reservoir and an incremental reservoir. Specifically, the uniform sampling can be used to count triangles by employing the existing algorithms while the non-uniform sampling ensures that the edge samples are connected. Experimental results driven by real datasets show that T-Sample can obtain much more accurate estimations on the distributions of node degrees than the existing reservoir-based sampling methods.**

## I. Introduction

The rapid growth in the scale of real-world application scenarios, e.g., bioinformatics, social media and computer network traffic, necessitates the storage, processing and analysis of the data content in the form of large graph streams for which each edge carries the information about interaction between one node (entity) and another node (entity) [1]. Given the sheer size of data, many recent studies focus on *one-pass stream sampling methods* [2], [3], in which each edge is processed for only one time to conduct two analyses. The first is an analysis of the total number of triangles (called triangle count) of a graph stream, which is used to provide an overall description of the connectivity of a large graph stream and has attracted considerable attentions [2]. The second is to show the connectivity of a large graph stream by an analysis of the distributions of the node degrees. This analysis is used to provide a quantitative description of a specific connectivity among the entities in a large graph stream.

In many classes of applications, the analyses of both the triangle count and node degrees of a large graph stream are required simultaneously. For example, in social media, the triangle count reflects the overall connectivity among the users. On the other hand, the specific connectivity and relationship among the users are uncovered through data mining based on node degrees [4]. The combination of these two analyses is helpful in evaluating the influence of the social media while providing precise predictions, e.g., product recommendations and extent of rumor spreads.

However, existing one-pass sampling methods only focus on producing edge samples either for the triangle count or node degrees but not both. Even if two corresponding processes of existing one-pass sampling methods cooperate to uncover the connectivity of a large graph stream with both analyses, there are two main problems. First, when the two methods, one for each of the two analyses, are executed simultaneously, more computation and memory resources are consumed. Second and more importantly, they cannot obtain these two analyses in real-time when the two methods are executed asynchronously. Even without these problems, existing one-pass sampling methods suffer from inaccurate estimations when they are used to obtain information on node degrees by producing edge samples that are rarely connected.

In a one-pass stream sampling method, a reservoir is used to preserve the sampled edges of a graph stream [5], [6], which gives rise to the name of ***a reservoir-based sampling method***. In this paper, we propose a new one-pass stream sampling method, called ***triangle-induced reservoir sampling*** or ***T-Sample***, to better characterize the connectivity of a large graph stream from the perspective of counting the triangles and obtaining the information on node degrees simultaneously. Specifically, T-Sample employs a dual sampling mechanism, namely, combining a uniform sampling with a non-uniform sampling to produce connected edge samples, which overcomes the problems of the single sampling mechanism used by existing one-pass sampling methods.

## II. Motivation

The existing reservoir-based sampling methods can be classified into two categories, i.e., uniform reservoir-based

sampling, which is capable of learning the probability of an edge entering a reservoir in a graph stream prior to sampling, and non-uniform reservoir-based sampling for which the probability of an edge entering a reservoir is not known before the sampling process. However, both the existing uniform and non-uniform reservoir-based sampling methods produce edge samples that no longer contain or convey sufficient actual connectivity of a graph stream.

As shown in Figure 1, the existing reservoir-based sampling methods provide very limited information about the different types of node degrees as the percentage of the nodes with degrees more than 10 is almost equal to zero, where GPS-Post [7], Triest-IMPR [8], GSH [9] are the uniform reservoir-based sampling while NeiSampling [10], StreamSampling [11] and PIES [6] are the non-uniform reservoir-based sampling. Furthermore, Figure 1 also shows that the degrees of most of the nodes ($70\%-92\%$) are equal to one, based on the edge samples obtained by the existing reservoir-based methods implemented in the same platform (described in Section IV). Since the degree of any node is at least equal to one because each edge consists of exactly two nodes, the results in Figure 1 clearly imply that the edge samples produced by the existing reservoir based sampling methods are mostly isolated, unconnected edges.

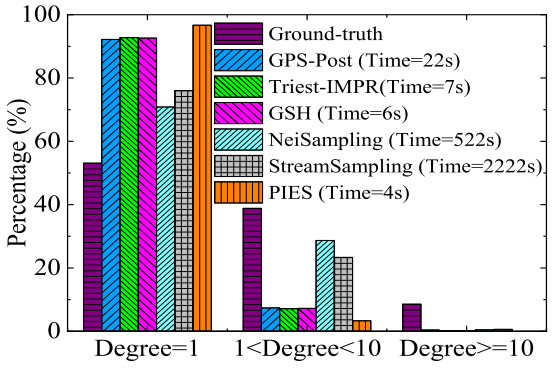



**Fig. 1: The distribution of node-degree counts (and processing times) of edge samples generated by the existing reservoir-based sampling methods over the Youtube graph stream (Section IV) when the capacity of the reservoir is set to 5K.**

Furthermore, except for PIES which does not estimate the triangle count, Figure 1 shows that the non-uniform sampling methods (NeiSampling and StreamSampling) have slightly better results than the uniform sampling ones in estimating the information of node degrees while they spend much more time than the latter. Such experimental results imply that it is more cost-efficient to estimate the triangle count using the uniform reservoir-based sampling while it is more effective in estimating the specific topological characteristics by non-uniform reservoir-based sampling. Therefore, to meet the requirement of the applications to analyze both the triangle count and the node degrees, a new reservoir-based sampling method should inherit the advantages of both categories of uniform and non-uniform sampling while alleviating their disadvantages. Motivated by these insights, we propose in this paper a new reservoir-based sampling method, called T-Sample that employs a dual-sampling mechanism and is capable of producing connected edge samples.

## III. Design and analysis of T-Sample

In this section, we first elaborate on the design of T-Sample's dual sampling mechanism. Then, we analyze the probabilities of an edge entering the two types of reservoirs in T-Sample based on the triangle count obtained and the connected edge samples produced.

### A. Dual sampling

T-Sample, as a one-pass sampling method for which each edge of a graph stream is processed only one time, employs a dual sampling mechanism (uniform and non-uniform). Specifically, such a sampling process relies on two types of reservoirs, base reservoir and incremental reservoir, to estimate the triangle count while simultaneously obtaining the actual connectivity information of a graph stream based on node degrees (i.e., the degree distributions).

***Base reservoir.*** T-Sample's uniform sampling employs a reservoir with a static capacity, namely, a base reservoir $R_{base}$. An important characteristic for the edge samples preserved in the base reservoir is that these edges are updated frequently with the arrival of each new edge in a graph stream while the number of the edges preserved in it is static.

***Incremental reservoir.*** T-Sample's non-uniform sampling employs a reservoir with a dynamic capacity, namely, an incremental reservoir $R_{incre}$, to produce the connected edge samples. The ***prerequisite*** for an edge to enter the incremental reservoir is that the edge can form triangles with the edges currently preserved in the base reservoir. An edge once sampled by the non-uniform sampling cannot be removed from the incremental reservoir. Therefore, the volume of the edges preserved in the incremental reservoir, is always non-decreasing. To limit the memory space used by the incremental reservoir, we design a parameter to control the probability of an edge entering the incremental reservoir by exploiting the density/sparsity of the connectivity of a graph stream. Before we derive the sampling probabilities, we first present the work flow of T-Sample with its dual-sampling mechanism.

As illustrated in Figure 2, at the very beginning of T-Sample's process, the front $c$ edges of a graph stream are directly preserved in the base reservoir of capacity $c$. From this point on, *whether a newly arrived edge is sampled or not by T-Sample depends on if the edge has a chance to be preserved in either the base reservoir or the incremental reservoir.* Notice that any edge of a graph stream can only be preserved in at most one of the two reservoirs. Figure 2 depicts the T-Sample's dual sampling process: the $i^{th}$ $(i > c)$ edge will first try to enter the base reservoir and, when this effort fails, it then tries to enter the incremental reservoir.

Generally speaking, each edge in a graph stream has a chance to be preserved in the base reservoir. Thus, the probabilities by which the triangles are formed based on the whole

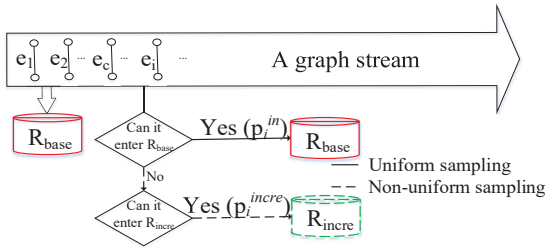



**Fig. 2: T-Sample's dual sampling mechanism, with uniform sampling in solid black lines and non-uniform sampling in dashed black lines while $p_i^*$ denotes the probability of entering a reservoir respectively for the $i^{th}$ edge.**

graph stream can be inferred to estimate the triangle count as described in [8]. On the other hand, the edges that fail to enter the base reservoir have chances of entering the incremental reservoir by leveraging the important structures of triangles that express the basic and cohesive connectivity among the edges in a graph stream. Thus, the edge samples, preserved in both the base and incremental reservoirs, are able to largely preserve the connectivity.

### *B. Probabilities of an edge entering the reservoirs*

Due to the dual sampling mechanism, the probability of an edge entering the base reservoir or the incremental reservoir in T-Sample is analyzed from two cases as follows: that in uniform sampling and that in non-uniform sampling.

***Uniform sampling.*** Let $p_i^{in}$ denote the probability of the $i^{th}$ arriving edge entering the reservoir and $p_{out}$ the probability of an edge already preserved in the reservoir being replaced by the newly sampled edge. $p_i^{in}$ and $p_{out}$ are given as,

$$p_i^{in} = min\{1, \frac{c}{i}\},\ p_{out} = \frac{1}{c}, \tag{1}$$

Notice that T-Sample's uniform sampling does not change the sampling processes of Triest and Triest-IMPR that were proposed in [8]. Therefore, the algorithms for counting the triangles by Triest and Triest-IMPR can be used to count the triangles during T-Sample's sampling process.

***Non-uniform sampling.*** In T-Sample, a newly arrived edge, which has failed to enter the base reservoir, has a chance to enter into the incremental reservoir if the edge satisfies the prerequisite for entering the incremental reservoir. Since the size of the incremental reservoir is non-decreasing with $i$, the probabilities for edges to be preserved in it must be properly controlled to limit its memory usage while preserving the topological structures approximately.

Intuitively, a more densely connected graph stream tends to have a correspondingly higher triangle count, implying that a newly arrived edge is more likely to form at least one triangle with edges preserved in the base reservoir and thus meet the perquisite for entering the incremental reservoir. On the other hand, the opposite is true for a sparsely connected graph stream, i.e., a newly arrived edge is less likely to meet the prerequisite. Based on this intuition, the parameter $\frac{c}{c+num_i}$ helps indicate whether a graph stream being sampled is densely or sparsely connected, where $num_i$ is the total number of edges satisfying the prerequisite for entering the incremental reservoir among the front $i-1$ edges and can be calculated during the process of counting the triangles. That is, the lower the value of this parameter, the more densely connected a graph stream is. Thus, we use this parameter $\frac{c}{c+num_i}$ to control the probability of an edge entering the incremental reservoir and further limit T-Sample's memory usage.

Specifically, in face of a densely connected graph stream, the value of $\frac{c}{c+num_i}$ decreases rapidly as $i$ increases, meaning that the probability of an edge entering the incremental reservoir will diminish rapidly. This helps limit the number of edges added to the incremental reservoir when sampling a densely connected graph stream for which there are indeed many edges already preserved in the incremental reservoir. On the other hand, for a sparely connected graph stream, the value of $\frac{c}{c+num_i}$ decreases very slowly as $i$ increases, meaning that the probability of an edge entering the incremental reservoir will remain relatively steady. This helps obtain as many connected edge samples as possible for uncovering the original connectivity of a sparsely connected graph stream.

Therefore, the probability $p_i^{incre}$ of the $i^{th}$ edge entering the incremental reservoir is given as,

$$p_i^{incre} = p_i^{meetPre} \times (1 - p_i^{in}) \times \frac{c}{c + num_i}, \tag{2}$$

where $p_i^{meetPre}$ signifies whether an edge meets the prerequisite to enter the incremental reservoir. In other words, in an actual sampling process, $p_i^{meetPre} = 1$ means the $i^{th}$ edge meets the prerequisite, or $p_i^{meetPre} = 0$ otherwise.

## IV. Evaluation

***Platform and Workload.*** The simulations are conducted on a computer with Intel Xeon E5620 processors and 64-bit Ubuntu Linux OS. Each experiment, which employs a single core with at most 4GB of RAM, entails 20 runs of the simulation so that the results reported are statistically stable and meaningful. The workload traces, summarized in Table I and downloadable from [12] and [13] include two public real-world graph datasets (graph streams) of which one contains more than a billion edges.

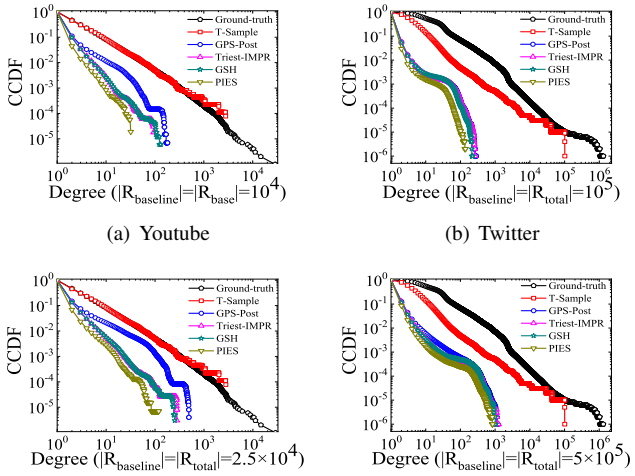
**TABLE I: Summary of Graph Datasets, where $|V|$, $|E|$ and $\triangle_{total}$ denote the total numbers of nodes, edges and triangles in a graph stream $G = (V, E)$, respectively.**

| Graph | $\|V\|$ | $\|E\|$ | $\triangle_{total}$ |
|---|---|---|---|
| Youtube | 1,134,890 | 2,987,624 | 3,056,386 |
| Twitter | 41,652,230 | 1,468,365,182 | 34,824,916,864 |

***Baseline methods.*** The following state-of-the-art reservoir-based sampling methods, GPS Post-Stream (GPS-Post) [7], Triest-IMPR [3], GSH [9] and PIES [6], are considered the evaluation baselines for T-Sample. Although In-Stream (GPS-In), proposed in [7], shows smaller estimation errors and variances than GPS-Post, it consumes much more time than GPS-Post and thus is not selected as a baseline. In GPS-Post, the weight of a newly arrived edge is set as the number of the

triangles formed by it and the edges preserved in the reservoir thus far [7]. All these sampling schemes are implemented in C++. Since T-Sample can use the method proposed in [3] to count the triangles and produce the same experimental results as those reported in [3], we do not further illustrate the results of counting the triangles base on the process of T-Sample due to the page limits.

***Capacity of the reservoir.*** As described in Section III, for T-Sample, the base and incremental reservoirs are used to obtain information on node degrees. However, for all the baseline methods, only one reservoir is used to preserve edge samples. When the baseline schemes are used to estimate the node degrees, there are two cases for comparison with T-Sample in terms of estimation accuracy, time and memory costs. In the first case, the capacity of the reservoir for the baseline sampling schemes is set to be the same as that of T-Sample's base reservoir, which means that T-Sample will use more total memory capacity to obtain information about node degrees. The second case sets the reservoir capacity for the baseline schemes to be the sum of those for the base and incremental reservoirs of T-Sample, $|R_{baseline}| = |R_{total}| = |R_{base}| + |R_{incre}|$.




(a) Youtube (b) Twitter

(c) Youtube (d) Twitter

**Fig. 3: The distributions of node-degree counts over Youtube and Twitter with $|R_{baseline}| = |R_{base}|$ and $|R_{baseline}| = |R_{total}|$ respectively. Note that each data point $(x, y)$ in the figures indicates that $100 \times y\%$ of nodes are of degree equal to or smaller than $x$.**

***Node-degree counts*** refer to the numbers of nodes with different node degrees and are measured by the distributions of node-degree types inferred by edge samples (or the ground truth from the dataset) among the nodes in a graph stream. This measure indicates how closely the node-degree counts inferred by edge samples reflect the ground truth with a very small sample set. Figure 3 shows that, whether $|R_{baseline}| = |R_{base}|$ ($10^4$ in Youtube and $10^5$ in twitter) or $|R_{baseline}| = |R_{total}|$ ($2.5 \times 10^4$ in Youtube and $5 \times 10^5$ in twitter), T-Sample obtains the node-degree counts that are much closer to the ground-truth values than the four baseline methods, as measured in the complementary cumulative distribution function (CCDF), over YouTube and Twitter. Furthermore, the node-degree counts distributions obtained by the baseline methods do not change significantly with the increase of the sample size, as shown in Figure 3, because these methods are not able to produce connected edge samples.

## V. Conclusions

In this paper, we propose a new reservoir-based sampling method, called triangle-induced sampling or T-Sample, which can leverage the existing uniform reservoir-based sampling process to count the triangles over large graph streams efficiently. Furthermore, significantly different from existing reservoir-based sampling methods, T-Sample is a first attempt at producing connected edge samples. Extensive dataset-driven experimental results show that T-Sample characterizes the connectivity of a graph stream much more accurately than the existing reservoir-based sampling methods at the same memory costs.

## VI. Acknowledgements

We would like to thank the anonymous reviewers of this paper. Furthermore, this work is supported by NSFC No.61772216, National Defense Preliminary Research Project (31511010202), Wuhan application basic research project 2017010201010103, Fund from Science, Technology and Innovation Commission of Shenzhen Municipality (JCYJ20170307172248636). In addition, this work is also supported by the US NSF under Grants No.CCF-1704504 and No.CCF-1629625.

## References

[1] A. McGregor, "Graph stream algorithms: a survey," *ACM SIGMOD*, 2014.
[2] P. Wang, Y. Qi, Y. Sun, X. Zhang, J. Tao, and X. Guan, "Approximately counting triangles in large graph streams including edge duplicates with a fixed memory usage," *VLDB*, vol. 11, no. 2, 2017.
[3] L. D. Stefani, A. Epasto, M. Riondato, and E. Upfal, "Triest: Counting local and global triangles in fully dynamic streams with fixed memory size," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 11, no. 4, p. 43, 2017.
[4] A. Ching, S. Edunov, M. Kabiljo, D. Logothetis, and S. Muthukrishnan, "One trillion edges: Graph processing at facebook-scale," *VLDB*, vol. 8, no. 12, 2015.
[5] J. S. Vitter, "Random sampling with a reservoir," *ACM Transactions on Mathematical Software (TOMS)*, vol. 11, no. 1, pp. 37–57, 1985.
[6] N. K. Ahmed, J. Neville, and R. Kompella, "Network sampling: From static to streaming graphs," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 8, no. 2, p. 7, 2014.
[7] N. K. Ahmed, N. Duffield, T. L. Willke, and R. A. Rossi, "On sampling from massive graph streams," *VLDB*, vol. 10, no. 11, 2017.
[8] L. De Stefani, A. Epasto, M. Riondato, and E. Upfal, "Trièst: Counting local and global triangles in fully-dynamic streams with fixed memory size," in *ACM KDD*, 2016, pp. 825–834.
[9] N. K. Ahmed, N. Duffield, J. Neville, and R. Kompella, "Graph sample and hold: A framework for big-graph analytics," in *ACM KDD*, 2014.
[10] A. Pavan, K. Tangwongsan, S. Tirthapura, and K.-L. Wu, "Counting and sampling triangles from a graph stream," *VLDB*, vol. 6, no. 14, 2013.
[11] M. Jha, C. Seshadhri, and A. Pinar, "A space-efficient streaming algorithm for estimating transitivity and triangle counts using the birthday paradox," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 9, no. 3, p. 15, 2015.
[12] "Snap datasets," http://snap.stanford.edu/.
[13] "Konect datasets," http://konect.uni-koblenz.de/.