Cost-based analyses of random neighbor and derived sampling methods

Yitzchak Novick1,2* and Amotz Bar-Noy3

Abstract
Random neighbor sampling, or RN, is a method for sampling vertices with a mean degree greater than that of the graph. Instead of naively sampling a vertex from a graph and retaining it (‘random vertex’ or RV), a neighbor of the vertex is selected instead. While considerable research has analyzed various aspects of RN, the extra cost of sampling a second vertex is typically not addressed. This paper explores RN sampling from the perspective of cost. We break down the cost of sampling into two distinct costs, that of sampling a vertex and that of sampling a neighbor of an already sampled vertex, and we also include the cost of actually selecting a vertex/neighbor and retaining it for use rather than discarding it. With these three costs as our cost-model, we explore RN and compare it to RV in a more fair manner than comparisons that have been made in previous research. As we delve into costs, a number of variants to RN are introduced. These variants improve on the cost-effectiveness of RN in regard to particular costs and priorities. Our full cost-benefit analysis highlights strengths and weaknesses of the methods. We particularly focus on how our methods perform for sampling high-degree and low-degree vertices in separate sampling phases.

Keywords: Fair cost comparison, Random neighbor sampling, High-degree vertex sampling

Introduction
Efficiently locating high-degree vertices in a graph can be important in many contexts. Without total knowledge of the graph, sampling a vertex purely at random, a method we call ‘Random Vertex’ or RV, will return a vertex whose expected degree is the mean degree of the graph. In 2003, Cohen et al. (2003) introduced a new sampling method where a vertex is sampled at random, but then exchanged for one of its neighboring vertices which is sampled at random from among all neighbors and selected in place of the first vertex. We call this method ‘Random Neighbor’ or RN. The method is loosely inspired by Scott Feld’s friendship paradox (Feld 1991) which states that the mean degree
of a graph’s collection of neighbors is higher than the mean degree of the set of the graph’s vertices.

RN is, in fact, a superior sampling method to RV for finding high-degree vertices and has gained popularity in many contexts and areas of research (for example, see Han et al. 2013; Lü et al. 2016; Christakis and Fowler 2010). However, there is a cost for this gain. Specifically, every vertex that is ultimately selected requires two vertices be sampled, the original vertex that is discarded and the selected neighbor.

The concept is perhaps best illustrated with an example. Figure 1 demonstrates an experiment on a set of Barabási-Albert (BA) graphs. We fix some budget, \(b\), that represents the number of vertices we will sample. We then sample with both RV and RN until the budget is exhausted. The x-axis represents increasing values of \(b\). The points in the lowest curve represent the maximum degree vertex in the collection when the entire budget is spent on vertices, RV sampling. The points in the highest curve represent the maximum degree in a collection of \(b\) neighbors. This represents RN sampling, but only if we allow the entire budget to be spent on the selected neighbors. Doing so ignores any costs associated with sampling the original vertices that led to these neighbors. The points in the middle curve demonstrate what we consider a more realistic representation of RN’s results. A budget \(b\) should only yield \(b/2\) neighbors, because half of the budget had to be spent on the initial vertices that were used to acquire the neighbors. So these middle values represent the maximum degree of a collection comprised of only \(b/2\) neighbors instead of \(b\). This demonstrates the strength of RN as compared to RV from a new perspective, one that accounts for cost. It is worth noting that one could reasonably object and suggest the \(b/2\) vertices that were sampled in order to acquire the neighbors should also be included in the final collection for a total size equal to the sampling budget. This alternative sampling method, which we call RVN, is something we analyze later in this paper. For this particular example we will demonstrate that it gives negligible benefit.

In order to analyze RN from a cost-based perspective, we will define costs related to the processes of sampling and selecting vertices. These costs will provide a model that can be used to analyze the true value of RN. In addition, the introduction of costs suggests alternative sampling methods that may maximize performance for particular cost considerations. We will present a number of these alternatives here and provide
an in-depth analysis of their advantages and disadvantages vis-à-vis the different costs. This paper is an extension of our previous work (Novick and Bar-Noy 2021) where we first introduced our cost model. We build on this work here by adding additional results for many of the new sampling methods introduced there. We also provide previously omitted analyses that explore the $R_kN$ sampling method and the two-phase methods we introduced in that paper.

Terminology and notation
When we refer to the set of vertices in a graph, the term ‘vertex’ retains its traditional meaning. However, when we refer to a single entity sampled from a graph as a ‘vertex,’ we specifically mean that it was sampled from the collection of vertices and not from the collection of neighbors of a specific vertex. We will use the term ‘neighbor’ to refer to a sample taken from one vertex's neighbors.

In this paper we use abbreviations to refer to sampling methods. So for example, we will use $RV$ to refer to random vertex sampling, $RN$ to refer to random neighbor sampling, etc. However, we will also use these abbreviations in mathematical expressions such as equations and inequalities. For example, the inequality $RN \geq RV$ would mean that $RN$ is a superior or equivalent sampling method to $RV$ in terms of finding high degree vertices. We rely on context to clarify the meaning of every abbreviation. Also, while we will employ multiple metrics of a sampling method’s success, when the method is not specified the assumed metric should be the expected degree of a single vertex returned by the method. Therefore, unless a different metric is specified, the inequality $RN \geq RV$ means $\mathbb{E}[RN] \geq \mathbb{E}[RV]$, or “The expected degree of a vertex sampled by $RN$ is greater than or equal to the expected degree of a vertex sampled by $RV$”.

Preliminary
There are a few characteristics of $RN$ that bear mentioning as foundational to our research. First, it is worth recognizing that the friendship paradox does not actually prove that $RN$ is superior to $RV$. This is demonstrated by Kumar et al in an earlier draft of Kumar et al. (2021) that can be found on Vineet Kumar’s webpage in the Yale University website. Construct a graph comprised of two separate subgraphs, one of size $i$ and one of size $j$ with $i > j \geq 2$. The friendship paradox applies in this graph because there is a variance of degree so the mean degree of neighbors is strictly greater than the mean degree of the graph. Yet, by symmetry, we know that $RN = RV$ in this graph. In reality, the true sampling method suggested by the friendship paradox would be random edge (see Leskovec and Faloutsos 2006) which we compared to $RN$ in Novick and BarNoy (2020). However $RN$'s superiority to $RV$ has been demonstrated in Cohen et al. (2003) and Momeni and Rabbat (2018) among others. We have also constructed a simple proof that $RN \geq RV$ which we later found in both Kumar et al. (2021) as well as the aforementioned draft where it is further attributed to a comment on an online article in the New York Times’s website. However, to our knowledge this proof has never appeared in a peer-reviewed publication so we will include it as an appendix to this paper.

Two other areas of interest are $RN$’s performance in Erdős Rényi (ER) graphs (Erdos and Rényi 1960) versus Barabási Albert (BA) graphs (Barabási and Albert 1999), and $RN$’s performance for finding high-degree vertices, which we will informally refer to as
'hubs', versus finding low-degree vertices, which we will informally refer to as 'leaves.' We will explore both of these topics here.

**RN in ER and BA graphs**

In Novick and BarNoy (2020) we demonstrated experimentally that $RN$ outperforms $RV$ significantly in BA graphs while it is of minimal benefit in ER graphs. We suggest that there is a connection between the power-law distribution of degree (Faloutsos et al. 1999) that characterizes BA graphs and $RN$'s performance in these graphs. We will informally explain this connection here.

The connection between degree-homophily and $RN$ has been discussed in Novick and BarNoy (2020) and Kumar et al. (2021). In Novick and BarNoy (2020) we used the well known measure of assortativity (Newman 2002) (also Piraveenan et al. 2010; Thedchanamoorthy et al. 2014; Jackson 2019; Pal et al. 2019), whereas Kumar et al argue for their own measure, inversity. But here we will use the term assortativity to loosely refer to both measures as inversity correlates strongly with assortativity despite the significant differences the authors highlight in their paper. Intuitively, any difference between $RN$ and $RV$ clearly requires at least some amount of disassortativity in order for the neighbor to differ from the vertex, and less assortativity in fact increases this effect. Newman demonstrated that both ER and BA graphs tend towards zero assortativity (Newman 2002), neither positive or negative. However, research has shown that this value in BA graphs is an aggregate measure of two sharply contrasting types of edges (Bertotti and Modanese 2019). A number of leaves are highly assortative, connected to other leaves like themselves. However, the hubs connect to many leaves as well, and these connections are extremely disassortative. This suggests an intuition for $RN$'s strong performance in BA graphs. The power-law distribution implies that a randomly sampled vertex is far more likely to be a leaf than a hub. However, exchanging it for one of its neighbors has a reasonable probability of increasing the resulting degree because of the significant likelihood that the leaf is disassortatively connected to a hub.

In truth though, the power-law distribution directly suggests the existence of the disassortative connections as well. The famous Erdős Gallai (1960) and Havel-Hakimi (1955; 1962) theorems are both in part predicated on a simple premise. If a graphic degree sequence is partitioned into high-degree hubs and low-degree leaves, any edge endpoints of the hubs that cannot be satisfied by connecting to other hubs must be satisfied by connecting to leaves instead. This necessity definitionally translates into some amount of disassortativity. While a comparatively large amount of leaves does not necessarily imply that the hubs cannot be entirely interconnected among themselves, a typical power-law distribution will have the number of hubs being far fewer than their accumulated degrees and this explains why BA graphs in particular show a strong performance for $RN$ over $RV$.

**$RN$'s inefficiency in finding leaves**

This understanding of the mechanics of $RN$'s success in BA graphs explains another important characteristic of $RN$ which is its inferiority as a sampling method for selecting leaves in a graph. While a single sampling of $RV$ will find any given leaf with probability $1/n$, in order for $RN$ to select a given leaf, it would have to find it via one of its neighbors,
a particularly poor strategy when the number of neighbors is small, even worse when it neighbors a hub that neighbors so many other vertices.

**Hubs versus leaves in a star graph**

As an example, consider the star graph of \( n \) vertices. The star graph can be used as an exaggerated illustration of a power-law distribution as it has very few hubs (1) of very high-degree \((n - 1)\), and very many leaves \((n - 1)\) of very low-degree (1). It is therefore often useful to analyze a star graph in order to explore a feature of BA and other power-law graphs.

When sampling from the star graph with \( RV \), all vertices have an equal probability of being sampled, \( P(v_i) = \frac{1}{n} \). The expected number of samples required to find the center is \( E[C] = n \), and the expected number of samples required to find all leaves is \( E[L] = n(H_n - 1) = \Theta(n \log n) \) per the coupon collector’s problem.

Contrast this with sampling using \( RN \). The probability of selecting the center is \( P(C) = \frac{n - 1}{n} \) and the probability of selecting a leaf vertex is \( P(L) = \frac{1}{n} \). The expected number of samples required to find the center approaches 1 at \( E[S_c] = n(n - 1) \), but the expected number of samples required to collect all leaves is

\[
E[S_L] = \sum_{i=1}^{n-1} \left( \frac{1}{n} \right)^{i-1} = n(n - 1) \sum_{i=1}^{n-1} \frac{1}{n - i} = n(n - 1) H_n = \Theta(n^2 \log n)
\]

**Hubs versus leaves in BA and ER graphs**

We further demonstrate this phenomenon with a simple experiment. We repeatedly select vertices with both \( RV \) and \( RN \) and track how many iterations of each sampling method are required to select the top \( x\% \) of the vertices ranked in descending order of degree. The aggregate results for repeated experiments on sets of BA and ER graphs is shown in Fig. 2. Because \( RV \) is naïve sampling, it finds the entire graph with \( \Theta(n \log n) \) samples per the coupon collector’s problem. \( RN \), on the hand, finds hubs very quickly, but struggles greatly to complete the collection and find the lowest-degree vertices in the graph. As \( RN \geq RV \), the phenomenon is still true in ER graphs. But it is comparatively muted for the reason we discussed. BA graphs have a strong element of disassortativity between hubs and some connected leaves, while ER graphs are more homogeneously unassortative.

**Sampling costs—\( C_v \) and \( C_n \)**

The main focus of our research is a thorough analysis of costs that are associated with \( RV \), \( RN \), and the other sampling methods that we will introduce. In our introduction we mentioned the most obvious cost, sampling a vertex. In many contexts this cost would be equivalent for sampling a vertex from the graph and for sampling a vertex from the neighbors of an already sampled vertex, and we will in fact make this assumption in some of our analyses. However, we suggest that this may not always be the case. Sampling a neighbor my be
less expensive as the set from which the neighbor will be sampled is smaller. Or, perhaps there is a privacy concern related to learning connections that would apply only to sampling a neighbor which would make sampling a neighbor more expensive. We therefore generalize the sampling costs to two distinct costs, $C_v$, the cost of sampling a vertex, and $C_n$, the cost of sampling a neighbor.

**Critical $C_n$**

Let us fix $C_v = 1$ so that we are expressing both $C_n$ and total cost in terms of $C_v$. We seek a ‘critical $C_n$’ value ($CC_n$), that is a value for $C_n$ where $RV$ and $RN$ perform equally well in light of their associated costs. Knowledge of a such a value for a specific graph would allow a proper evaluation of whether or not $RN$ should be used instead of $RV$. $CC_n$ is, ultimately, a measure of $RN$’s superiority over $RV$ as the higher the degree of a selected neighbor is compared to that of a sampled vertex, the more one would be willing to pay in order to sample the neighbor. Following the same logic, finding some scenario where $CC_n < 0$ would indicate that (somehow) $RV > RN$.

**$CC_n$ for expected degree**

Obviously, in order to quantify $CC_n$, we first need to define what metric of success we are using to quantify the respective performances of $RV$ and $RN$. We will first focus on the expected degree of a vertex/neighbor selected by each. We have fixed $C_v = 1$, so a vertex selected by $RV$, with its corresponding expected degree, requires one cost unit. A neighbor selected by $RN$ costs $C_v + C_n = 1 + C_n$. Therefore, the $CC_n$ value that equates the two methods in terms of cost for their respective expected degrees can be calculated as follows:

$$RV = \frac{RN}{1 + CC_n}$$
$$RV + RV(CC_n) = RN$$
$$CC_n = \frac{RN}{RV} - 1 \tag{2}$$

There is a strong intuition to this expression. The ratio $\frac{RN}{RV}$ should capture how much more someone would be willing to pay for a neighbor over a vertex. Also notice that, if $\frac{RN}{RV} < 2$, $CC_n < 1$. This means that if $C_v = C_n$, which would arguably be our natural
assumption in most scenarios, sampling with $RN$ would only be preferred to sampling with $RV$ if $RN$ is twice as strong as $RV$ for the desired metric. Otherwise, a more robust cost model would be required in order to justify the intuitive appeal of $RN$ sampling.

**$CC_n$ in canonical graphs**

We will apply Eq. 2 to a few famous graph types.

**$d$-regular Graph** In any perfectly assortative graph $RN$ reduces to $RV$ and $CC_n = 0$. The intuition is obvious. In a graph where $RN$ offers no advantage, any positive cost would be wasted.

**Star Graph** $RV$ in a star graph is equal to $2(n − 1)/n$ and $RN$ is equal to $(2(n − 1)^2 + 1)/n$, so

$$CC_n = \frac{1 + (n − 1)^2}{2(n − 1)} − 1$$

As $n$ increases, $CC_n \to n$. This is of course the same bound as the degree of the hub. The expression’s similarity to the degree of the hub reflects the high $C_n$ price worth paying for taking the leaf vertex one would initially sample with high probability and exchanging it for the hub.

**Complete bipartite graph** Assume we have a complete bipartite graph with sides $L$ and $R$. All vertices in $L$ are of degree $R$, and all vertices in $R$ are of degree $L$. $RV = (LR + RL)/(L + R)$, and $RN = (L^2 + R^2)/(L + R)$. Therefore, in a complete bipartite graph, $CC_n = (L^2 + R^2)/2LR − 1$.

**$CC_n$ for different sampling amounts and results**

Our second exploration of $CC_n$ will define it as a function of either how many samples are taken or a function of some desired result. Importantly, this means that these versions of $CC_n$ will not be fixed for a graph. This is an important use of $CC_n$ because it demonstrates how $RN$’s value can fluctuate even for the same graph depending on how long it is used or what the desired outcome is. For these analyses we will define three metrics to quantify the success of a sampling method:

1. **Total Degrees**—We repeatedly sample vertices from a graph with replacement and track the sum of the degrees of all selected vertices. $CC_n$ for this metric should converge on the $CC_n$ value based on expected degree defined in Eq. 2.

2. **Total Unique Degrees**—We repeatedly sample vertices from a graph with replacement and track the sum of the degrees of any new vertices selected. Here we will present resulting values as a percentage of the sum of all degrees in the graph.

3. **Max Degree**—We repeatedly sample vertices from a graph with replacement and track the maximum degree vertex selected. Here we will present resulting degree values as a percentage of the max-degree vertex in the graph.

The second metric corrects for the inclusion of duplicates in the first metric. If our goal is to immunize a network, for example, we probably can’t take credit for inoculating the same vertex twice. We include the first metric mostly for comparison, but we still suggest it might be useful in some scenarios. For example, in a situation where the goal of
sampling is information dissemination, our goal would be to reach as many high-degree vertices as possible in order to spread the information to their neighbors. But we might still appreciate selecting the same vertex multiple times as each selection reiterates the importance of the information and therefore increases the likelihood of it being shared.

In order to calculate $CC_n$ as a function of sampling iterations, let $RV(i)$ and $RN(i)$ be the resulting values, according to one of the success metrics, of selecting $i$ vertices with $RV$ and with $RN$ respectively. The cost of $i$ vertices selected with $RV$ is $i$ and the cost of $i$ neighbors selected with $RN$ is $i(1 + C_n)$. Therefore, for any $i$, we can calculate $CC_n(i)$ as follows:

$$\frac{RV(i)}{i} = \frac{RN(i)}{i(1 + CC_n(i))}$$

$$CC_n(i) = \frac{RN(i)}{RV(i)} - 1$$

(4)

To calculate $CC_n$ as a function of resulting values, assume some resulting value $V$ requires $i$ sampling iterations of $RV$ and $j$ sampling iterations of $RN$, or $V = RV(i) = RN(j)$. For this value $V$, we can calculate $CC_n(V)$ as

$$\frac{V}{i} = \frac{V}{j(1 + CC_n(V))}$$

$$CC_n(V) = \frac{i}{j} - 1$$

(5)

We experimented with ER and BA graphs with varying parameters as well as the graphs of multiple real world networks taken from the Koblenz online collection (Kunegis 2013). Figure 3 shows results from some of the experiments on BA graphs. These results were fairly typical for ER and real world graphs as well.

The results for total degrees correlated with $RN/RV - 1$ as predicted. For the other two measures, the results are somewhat more interesting. The bottom two charts measure success by max degree. The first chart shows calculated values of $CC_n$ for samples taken. $CC_n$ starts off high, because $RN$ gives a higher maximum degree than $RV$. However, as we continue to take samples, $RV$ will eventually find the max-degree vertex in the graph, and any further sampling for either method accomplishes nothing. This explains the (roughly) monotonically decreasing values of $CC_n$ ultimately converging on 0. The second chart plots $CC_n$ as a function of the percent of the maximum degree vertex being sought. This plot is noisier because sampling for a max degree vertex will not normalize as easily with repeated experiments, but the generally increasing nature of the function shows that $RN$ has more relative value compared to $RV$ as the degree of the maximum degree vertex being sought increases.

The middle charts are measuring the sum of all unique degrees accumulated. The chart on the left sees $CC_n$ steadily decrease as the hubs have already been selected and $RN$'s value is diminishing. Interestingly, $CC_n$ is actually negative for a range of values. This corresponds to the point where $RN$ is continuing to sample hubs that have already been selected and therefore has no value, but $RV$ is still finding new leaves. In this range $RV > RN$ which explains the negative $CC_n$ value. Then $RV$ also finds all of the vertices it will find and $CC_n$ levels out at 0, neither method offers any advantage. The chart on the
right shows a roughly monotonically decreasing $CC_n$. As we seek a higher and higher percentage of the total degrees in the graph, $RN$'s value over $RV$ decreases because of its failure to find leaves. Eventually, when enough of the degrees are being sought, $CC_n$ becomes negative because of the difficulty it has finding leaves while $RV$ is continuing to sample all vertices with uniform probability.

**Selection costs ($C_s$) and $RVN$ sampling**

In our introductory example in Fig. 1 we mentioned that, given a budget $b$ which we would use for sampling, and assuming $C_v = C_n = 1$, we could collect at most $b/2$ neighbors, but we could also retain the $b/2$ vertices we used as a means of collecting the neighbors and our final collection would be of size $b$. We refer to the sampling method where we select both the originally sampled vertex as well as its sampled neighbor as $RVN$. 

---

**Fig. 3** Experimentally calculated $CC_n$ values in BA graphs for varied metrics of success
The obvious explanation for why one would opt for RN over RVN is a cost that would be associated with selecting a vertex, which we will call $C_v$. Even having spent $C_v + C_n$ to sample the vertex and its neighbor, we select only the neighbor in order to capitalize on the $C_s$ we spend to do so.

The inclusion of $C_s$ in our model gives us a mathematical language for explaining priorities in a particular sampling endeavor. For example, one could ask why the initial paper of Cohen et al. (2003) ignores cost in the context of network immunization. By defining $C_v$ and $C_n$, we can offer a formal explanation by saying that perhaps $C_n \ll C_v$, and the extra cost of sampling the neighbor can be ignored. However, the far more likely explanation is that $C_s \gg C_v \approx C_n$. If the immunizations are in very short supply, it is wasteful to administer one to the low-degree vertex instead of paying an additional $C_v + C_n$ to find another high-degree neighbor. Even without generalizing sampling costs to two separate values, the evaluation of the respective costs of sampling versus selecting in a given scenario will provide an indication of which sampling method to choose.

**RVN versus RN**

We will now present a few comparisons between RVN and RN. Like the $CC_n$ value we sought for RV and RN, we will discuss a similar value that equates sampling costs with selection costs. We will also discuss why the two sampling methods are roughly the same for the metric of max-degree, and explore how RVN compares to RN for collecting leaves.

**Sampling costs versus selection costs and critical $\alpha$**

Let us ignore the direct comparison between $C_v$ and $C_n$ and simply define $\alpha = C_v + C_n$ so that $\alpha$ is the collective cost of sampling the vertex and neighbor from which we will select either both or just the neighbor by itself. We obviously assume a preference for spending the lower expense in order to capitalize on the higher expense. Therefore, if $\alpha < C_s$ we are more likely to sample again in order to spend $C_s$ on the higher-degree neighbor, and if $\alpha > C_s$ we are more likely to spend $C_s$ on selecting the vertex in order to capitalize more on the $\alpha$ we have already spent. But of course the direct comparison of RV and RN’s respective performances influences this decision as well.

We will define a ‘critical $\alpha$’ ($C\alpha$) as the $\alpha$ value for which RVN and RN are equal and use it to relate $\alpha$ to $C_s$ and the RN/RV ratio.

**$C\alpha$ for expected degrees**

We will calculate $C\alpha$ using the expected values of RVN and RN. We will compare a single selection of RVN to a single selection of RN. Every odd selection of RVN is the selection of a vertex and every even selection of RVN is a neighbor, so we will express the expected degree of an average selection of RVN as $RVN = (RV + RN)/2$. Therefore

$$\frac{RV + RN}{C\alpha + 2C_s} = \frac{RN}{C\alpha + C_s}$$

$$C\alpha = C_s \left( \frac{RN}{RV} - 1 \right)$$

(6)
This is the same expression as the one we found for $CC_n$ in Eq. 2. We see that a stronger $RN$ as compared to $RV$ leads to a higher $C\alpha$, that is $\alpha$ must be significantly larger than $C_s$ for the selection of the first vertex to be worthwhile. As $RN$ weakens vis-à-vis $RV$, $C\alpha$ decreases and the importance of capitalizing on $\alpha$ increases relative to the importance of capitalizing on $C_s$.

**Max degree for RN and RVN**

It is worth explaining why $RVN$ and $RN$ give comparable results when seeking to maximize the maximum-degree selected. The first chart in Fig. 4 is a direct comparison of $RN$ and $RVN$. It shows that selecting the vertex gives negligible benefit over selecting the neighbor by itself. The second chart reproduces the chart from our introduction in Fig. 1 and the third chart shows the same experiment using $RVN$ instead of $RN$ as the fair-cost method. Again, the gain in maximum degree is negligible.

The obvious explanation for this is that the maximum degree vertex is so likely to be found in the collection of selected neighbors that selecting the vertex fails to raise the maximum degree in a significant way. To corroborate this, we conducted experiments that test this exact hypothesis. We repeatedly sampled with $RVN$ in sets of ER and BA graphs with various parameters and tracked the percent of times the maximum degree vertex in the collection was a neighbor. The results are summarized in Table 1. Clearly even in ER graphs, but especially in BA graphs, there is a very high probability of finding the maximum degree entity among the neighbors. Some more extreme values for $n$ and/or $\mu$ may markedly change the characteristics of the graphs from those that typify the model, but the moderate values give a strong indication of why $RVN$ is not significantly better than $RN$ for the metric of maximum degree.

**RVN versus RN for selecting hubs and leaves**

One area worth exploring in $RVN$ sampling is how it addresses the specific weakness of $RN$ for finding leaves. Presumably, selecting vertices alongside the neighbors should provide additional coverage of the graph, and specifically cover leaves.

Figure 5 repeats our earlier experiment (Fig. 2) for assessing performances for hubs and leaves, but it includes $RVN$ and it breaks down the results by costs. The top row demonstrates results that are fairly obvious. Results for $C_v$ are on the left. Clearly, when $C_n = 0$, $RVN$ adds to the performance of both methods for no cost. Similarly, if $C_n = 0$ and we only focus on $C_n$ as we do in the chart on the right, $RV$ gives unrealistic results.

The third chart is more meaningful. As before we ignore a direct comparison between $C_v$
Required $C_v$, $C_n$, and $C_s$ costs for finding the top x% of high-degree vertices in BA graphs

**Table 1** Frequency of the max-degree vertex being among the selected neighbors. $n$ is the size of the graph, $\mu$ is the average degree

|                  | ER graphs |                  |                  | BA graphs |                  |                  |                  |
|------------------|-----------|------------------|------------------|-----------|------------------|------------------|------------------|
|                  | $\mu = 4$ | $\mu = 10$      | $\mu = 16$      | $\mu = 30$| $\mu = 4$       | $\mu = 10$      | $\mu = 16$      | $\mu = 30$      |
| Sampling 0.025n  |           |                  |                  |           |                  |                  |                  |
| $n = 500$        | .77       | .72              | .66              | .65       | .92              | .87              | .84              | .82              |
| $n = 1000$       | .8        | .72              | .68              | .65       | .95              | .94              | .91              | .85              |
| $n = 2500$       | .81       | .74              | .7              | .64       | .99              | .98              | .96              | .92              |
| $n = 5000$       | .84       | .72              | .72              | .65       | .99              | .98              | .96              | .94              |
| Sampling 0.05n   |           |                  |                  |           |                  |                  |                  |
| $n = 500$        | .78       | .72              | .70              | .67       | .96              | .92              | .86              | .85              |
| $n = 1000$       | .82       | .74              | .72              | .68       | .97              | .95              | .91              | .88              |
| $n = 2500$       | .84       | .8               | .72              | .71       | .98              | .98              | .96              | .93              |
| $n = 5000$       | .83       | .75              | .72              | .68       | .99              | .98              | .97              | .95              |
| Sampling 0.075n  |           |                  |                  |           |                  |                  |                  |
| $n = 500$        | .83       | .73              | .72              | .66       | .98              | .93              | .88              | .88              |
| $n = 1000$       | .83       | .74              | .73              | .67       | .98              | .96              | .95              | .95              |
| $n = 2500$       | .86       | .76              | .75              | .69       | .99              | .98              | .96              | .94              |
| $n = 5000$       | .87       | .78              | .72              | .7        | .99              | .99              | .98              | .97              |
| Sampling 0.1n    |           |                  |                  |           |                  |                  |                  |
| $n = 500$        | .82       | .77              | .72              | .68       | .98              | .95              | .9              | .87              |
| $n = 1000$       | .85       | .75              | .73              | .71       | .99              | .96              | .96              | .93              |
| $n = 2500$       | .85       | .8               | .73              | .71       | .99              | .99              | .98              | .95              |
| $n = 5000$       | .87       | .81              | .76              | .67       | 1                | .99              | .99              | .97              |
and \( C_n \) by assuming they are equal and track the total costs of the two combined. Here we see that \( RVN \) overemphasizes hubs enough to be superior to \( RV \) for hubs but inferior for leaves, but it still collects leaves far more quickly than \( RN \) which always selects neighbors and ignores vertices. The fourth chart shows results in terms of \( C_s \). Ignoring sampling costs makes this chart mostly irrelevant for an actual analysis, but it does allow us to focus on how quickly vertices are sampled as that is when \( C_s \) will be paid in order to select them. A higher \( C_s \) cost for hubs indicates that more leaves have been selected along with the selected hubs. The chart demonstrates that the more a method focuses on hubs, the more leaves are ignored while selecting the hubs, whereas methods that focus on leaves more find them and pay \( C_s \) for them sooner.

The heatmaps in Figs. 6 and 7 give a perspective on how the significance of the costs manifest when hubs or leaves are sought. Figure 6 shows the total cost of selecting the top .1 and top .9 of the graph for \( RV, RN, \) and \( RVN \). The x axes represent \( C_v \) costs in the range of \( 0 \leq C_v \leq 1 \) and the y axes represent \( C_n \) costs in the range of \( 0 \leq C_n \leq 1 \). For \( RV, C_n \) is irrelevant and regardless of the percent the total cost increases with \( C_v \) alone. Because \( C_s \) is not a factor, \( RN \) and \( RVN \) appear the same. Total cost appears to be slightly more influenced by \( C_m \) but generally influenced by both. In Fig. 7, the x axes represent \( C_s \) costs and the y axes represent \( \alpha \) costs, the collective cost of sampling a vertex and neighbor pair and ignoring how this cost is distributed between the two. \( RV \) is a function of both sampling and selection costs regardless of whether it is hubs or leaves being sought. In both \( RN \) and \( RVN \), finding hubs is mostly a function of selection costs, whereas finding leaves is more greatly influenced by sampling costs.

**RkN sampling**

The sampling costs \( C_v \) and \( C_n \) suggest another alternative sampling method. As noted, \( RN \) is predicated on the understanding that exchanging the initially sampled vertex for its neighbor raises the expected degree. If we completely ignore \( C_v \), it is perhaps not illogical to take our second vertex by repeating the process again, sampling a new vertex and then selecting one of its neighbors. But if \( C_v \) is significant, it makes sense to sample multiple neighbors of the same vertex in order to capitalize on the \( C_v \) that was already spent instead of selecting only the one neighbor and then immediately paying \( C_v \) again.

We call this alternative method \( RkN \) sampling. We select \( k \) random neighbors of every randomly sampled vertex. \( RkN \) is a generalization of \( RN \) with \( RN \) being the specific case of \( k = 1 \). We can quantify the cost of each neighbor selected with \( RkN \) as \( C_v/k + C_n \) rather than the \( C_v + C_n \) cost of \( RN \).

We conducted many repeated experiments with BA graphs and found that the average degree of all \( k \)th selected neighbors typically decreases as \( k \) increases. This is corroborated by the cost analysis shown in Fig. 8. The first chart shows the significant decrease in \( C_v \) for accumulating various percentages of the total unique degrees in the graph as \( k \) increases. However, the second plot shows that \( C_n \) decreases far less rapidly, because selecting additional neighbors of the sampled vertex will not have a positive effect on the expected degree. In fact, the third plot shows that \( C_s \) actually increases slightly as more neighbors of lower degree are being selected. Ultimately, if the general cost of sampling is significant, \( RkN \) appears to be a useful sampling method. But theoretically, if \( C_v = 0 \), it is probably worth sampling a new vertex and selecting
its first neighbor instead of selecting additional neighbors of the first sampled vertex. This probably also reflects somewhat on the lopsided distribution of degree in BA graphs. The initially sampled vertex is likely a leaf, but the leaf is not necessarily connected to any hubs, and if it is it is likely connected to a very small number of them. While increasing $k$ for a leaf that is connected to a hub does increase the likelihood of selecting the hub, it only adds additional leaves when the initially sampled vertex is an assortative leaf. And very large values of $k$ will have even less of an effect because they will only be relevant when the initially sampled vertex is a hub and it will likely result only in the collection of additional leaves. Because of the disparity in degree between leaves and hubs, even a modest increase in the probability of selecting a hub can raise

**Fig. 6** Heat maps representing the effects of $C_v$ and $C_n$ (sampling) costs for the top 10% and 90% of high-degree vertices
the average degree significantly. But when the focus is $C_n$ or $C_s$ and not $C_v$, the oversampling of neighbors does not have a strong positive net effect.

**RVkN sampling**
In the same way we tweaked $RN$ with $RVN$, we are able tweak $RkN$ by selecting the initial vertex along with the $k$ neighbors, $RVkN$ sampling. In scenarios where $C_s$ is negligible and sampling costs are the sole consideration, the initial vertex should clearly not be discarded. Also, as $k$ increases and the average neighbor degree decreases, the initial

---

**Fig. 7** Heat maps representing effects of sampling ($C_v$ and $C_n$) and selection ($C_s$) costs for the top 10% and 90% of high-degree vertices.
vertex may be of similar degree to many of the neighbors and may as well be selected. We will include \( RVkN \) in our later analyses.

**Full fair-cost analysis**

We are now ready to perform a full analysis of our sampling methods and costs. We will start with a theoretical analysis on a star graph to accentuate the strengths and weaknesses of our methods in regard to selecting hubs and leaves, then do an experimental analysis on BA graphs.

Below is a summary of the methods discussed so far:

- **RV**—Random Vertex Sampling. The naïve method of sampling and selecting a vertex at random from the set of vertices in the graph. Every selected vertex costs \( C_v + C_s \).
• **RN**—Random Neighbor Sampling, the method of Cohen et al. (2003). We sample a vertex, then sample one of its neighbors and select the neighbor. The cost of every neighbor is $C_v + C_n + C_s$.

• **RVN**—We sample a vertex, sample one of its neighbors, then select both. The average cost of a selection is $(C_v + C_n)/2 + C_s$.

• **RkN**—We sample a vertex, then sample and select $k$ of its neighbors. The cost of every neighbor is $C_v/k + C_n + C_s$.

• **RVkN**—We sample a vertex and $k$ of its neighbors, selecting all samples. The cost of a vertex is $C_v + C_s$, the cost of a neighbor is $C_n + C_s$, and the cost of an average selection is $(C_v + kC_n)/(k + 1) + C_s$.

### Fair-cost analysis in the star graph

As noted above, analyzing a star graph theoretically can highlight important strengths and weaknesses of a sampling method because of its exaggerated distinction between the lone hub and the $n - 1$ leaves. It should be noted though, that the analysis of a star graph is not a thorough examination because many concepts do not apply due to the graph’s simplicity. For example, we will not include $C_s$ in this analysis as this value will always be 1 or $n - 1$ for the two distinct collections of a hub and all leaves. Furthermore, $RkN$ is of no interest as $k$ only matters when we sample the hub as our initial vertex, and when we do the only logical value of $k$ is $n - 1$. With these excluded concepts noted, we present the expected $C_v$ and $C_n$ costs for selecting the hub and all leaves in Table 2.

For $RV$, the expected number of samplings required to obtain the center is $n$, and getting all leaves is $n \log n$. These are all $C_v$ costs, there are no $C_n$ costs. $RN$ finds the center with just over 1 sampling that incurs identical costs for $C_v$ and $C_n$. As we noted, though, it is very weak for finding leaves, requiring $n^2 \log n$ units of both $C_v$ and $C_n$. In the star graph, $RVN$ is guaranteed to acquire the center after one iteration at a cost of $C_v + C_n$. But it pays $C_n$ costs in order to acquire leaves whereas $RV$ does not.

### Experimental cost analysis on BA graphs

Our second analysis involved generating a set of BA graphs and recording the average costs for acquiring some percent of the graph. Here we have to note another oversimplification of the star graph, the clear delineation between hubs and leaves. In a BA graph this will of course not be as well defined. Our examination will consider the top 5% of high-degree vertices to be hubs. For this experiment we used BA graphs with $n = 4000$, $m = 3$. The results are presented in Table 3.
The top section demonstrates the characteristics of $RV$, $RN$, and $RVN$ that we have discussed. $RN$ does far better than $RV$ for hubs, far worse for leaves. And $RVN$ does better than both methods in most categories, but spends more units of $C_s$ on hubs because of the additional leaves it selects as a result of retaining the initially sampled vertex. The fact that $RVN$ has lower sampling costs for hubs implies that the initially sampled vertices contain some hubs that are retained. But $C_s$ is still higher because of the leaves that are selected.

We also include results for $RkN$ and $RVkN$. As discussed, increases in $k$ have a strong impact on $C_v$, but mixed results on the other costs. Again, the initially sampled vertices include some hubs, so $RVkN$ reduces $C_v$ and $C_n$ over $RN$ for lower values of $k$.

### Two-phase sampling methods

The weakness of $RN$ for finding leaves suggests an entirely new category of sampling methods. We could use a method like $RN$ in a first phase, trying to collect hubs, but then switch to $RV$ in a second phase in order to collect the leaves that $RN$ typically struggles to sample. We will call this method $RN$-$RV$. Significantly though, any of the $RN$ variants we introduced here can be used for the first phase for the sake of the advantages discussed. We can therefore add the following new, ‘Two-Phase’ sampling methods:

- **$RN$-$RV$**—A two-phase method that starts with $RN$ to find hubs, then switches to $RV$ to find leaves.
- **$RVN$-$RV$**—A two-phase method that seeks to find leaves sooner by selecting vertices along with neighbors in the first phase.
- **$RkN$-$RV$**—A two-phase method that tries to find hubs faster by sampling and selecting more neighbors per vertex in the first phase.

---

**Table 3** An experimental analysis of $C_v$, $C_n$, and $C_s$ costs in BA Graphs

| Method          | Selecting hubs (top 5%) | Selecting full graph |
|-----------------|--------------------------|-----------------------|
|                 | $C_v$ | $C_n$ | $C_s$ | $C_v$ | $C_n$ | $C_s$ |
| $RV$            | 23,061 | 3979   | 35,251 | 4000  |        |
| $RN$            | 6351  | 6351   | 2464   | 300,857 | 300,857 | 4000  |
| $RVN$           | 4837  | 4837   | 3404   | 24,531 | 24,530  | 4000  |
| $RkN-k = 1(RN)$ | 6351  | 6351   | 2464   | 300,857 | 300,857 | 4000  |
| $RkN-k = 2$     | 3166  | 6331   | 2463   | 139,021 | 278,040 | 4000  |
| $RkN-k = 3$     | 2191  | 6571   | 2493   | 94,175  | 282,519 | 4000  |
| $RkN-k = 6$     | 1497  | 6421   | 2590   | 48,621  | 208,456 | 4000  |
| $RkN-k = 7$     | 1481  | 6663   | 2653   | 40,491  | 182,306 | 4000  |
| $RkN-k = 8$     | 1420  | 6637   | 2661   | 34,483  | 161,077 | 4000  |
| $RkN-k = \infty$ | 1204 | 7205   | 2867   | 10,559  | 63,310  | 4000  |
| $RVkN-k = 1(RVN)$ | 4837 | 4837   | 3404   | 24,531  | 24,530  | 4000  |
| $RVkN-k = 2$   | 2807  | 5613   | 3138   | 20,994  | 41,987  | 4000  |
| $RVkN-k = 3$   | 1880  | 5638   | 2931   | 18,700  | 56,096  | 4000  |
| $RVkN-k = 6$   | 1450  | 6212   | 2965   | 14,268  | 61,185  | 4000  |
| $RVkN-k = 7$   | 1379  | 6203   | 2969   | 13,419  | 60,434  | 4000  |
| $RVkN-k = 8$   | 1365  | 6376   | 2995   | 12,844  | 60,006  | 4000  |
| $RVkN-k = \infty$ | 1118 | 6694   | 3054   | 7951    | 47,650  | 4000  |
• **RVkN-RV**—A combination of the previous two methods that selects the sampled vertex along with the $k$ neighbors in the first phase before switching to $RV$.

**Two-phase methods in the star graph**

Table 4 shows the results for $RN-RV$ and $RVN-RV$ in the star graph (as noted, $RkN$ and $RVkN$ are uninteresting in the star). The expected iterations for $RN-RV$ to find the hub approaches 1 and $RVN-RV$ will always find it in its first two selections. After that, the second phase finds the leaves for $n \log n$ units of $C_v$ as expected, but saves the $C_n$ costs of a one-phase method because it stops sampling neighbors in the second phase.

**Two-phase methods in BA graphs**

We will now explore the two-phase methods experimentally in BA graphs. Here we will once again have to note a few details of this analysis that do not apply to the star graph.

The most obvious question that arises for a two-phase method would be the point at which the method would switch phases. And of course this will largely depend on how we delineate between hubs and leaves. Notice again how both of these issues are trivial in the star graph as discussed above. There is only one hub, and both phase-one methods find it after one iteration with at least high probability, so we can choose a number of phase-one iterations that provides a satisfactory probability of selecting the hub and then switch to $RV$.

Another issue that arises is the desired coverage for each set of vertices. If the intention is to select every hub in the first phase and then every remaining leaf in the second, two phase methods offer negligible benefit over $RV$ by itself. As effective as $RN$ or any variant is, it will almost certainly miss some hubs and then $RV$ will have to find them. And if $RV$ has to find all leaves, it will probably find most of the hubs already selected in the first phase while it is sampling. We therefore define a parameter $\rho$ with $0 < \rho < 1$ where $\rho_h$ would be the percent of hubs that need to be selected in the first phase, and $\rho_l$ the percent of the leaves before terminating.

Our approach to simplify this problem and give meaningful results is as follows. First, we will generalize $\rho = \rho_h = \rho_l$, assume that we desire the same coverage for both hubs and leaves. We will also work with the same delineation between hubs and leaves that we employed previously in BA graphs, a .05/.95 split. Rather than specifying a switching point and testing whether or not we have acquired the requisite $\rho$ percent, we will run each phase until we have selected $\rho$ of the desired vertices and report the costs incurred in each phase.

| Method   | Selecting center | Selecting all leaves |
|----------|------------------|---------------------|
|          | $E[C_v]$         | $E[C_n]$            | $E[C_v]$         | $E[C_n]$ |
| $RN-RV$  | $n/(n-1)$        | $n/(n-1)$           | $n \log n$      | 0         |
| $RVN-RV$ | 1                | 1                   | $n \log n$      | 0         |

**Table 4** An analysis of costs $C_v$ and $C_n$ in a star graph of $n$ vertices for two-phase methods.
Also, because the connections are more complex than in a star graph, $RkN$ sampling is again of interest, so we will include $RkN-RV$ and $RVkN-RV$ in this study. Table 5 summarizes costs for $\rho \in \{.7, .8, .9\}$.

As expected, $RN$ outperforms $RV$ for hubs and $RV$ is better for leaves, while $RVN$ is a strong compromise for sampling costs but not for $C_s$. As before we see that increases in $k$ give diminishing returns in $C_s$ for both $RkN-RV$ and $RVkN-RV$, and do not meaningfully impact $C_i$ or $C_s$. And we see that $RVkN-RV$ reduces sampling costs over $RkN-RV$.

The results also corroborate our explanation that, while $RkN$ does find hubs faster than $RN$, many of its selections would have been found as vertices rather than neighbors. This is why $RVN-RV$ pays a higher $C_s$ price than $RVkN-RV$ in phase 1 and lower in phase 2, because so many of the neighbors that would have been selected along with the first vertex by increasing $k$, are selected as vertices anyway by $RVN$, to the point where it makes more selections.

**Conclusion and future research directions**

In this paper, we have presented an analysis of the famous $RN$ sampling method from the perspective of cost. We have built a useful cost-model that considers both sampling and selection which provides an infrastructure for a true fair-cost comparison of $RN$ to $RV$. We described ‘critical cost’ values that can be used to evaluate a graph in order to contrast two different sampling methods and determine their relative values for a desired goal. We highlighted an interesting weakness of $RN$, its inability to find leaves efficiently. We also offered numerous tweaks to $RN$ that seek to capitalize on certain costs over others which would allow us to pick an appropriate method for the costs of a given scenario.

We consider this groundbreaking work which opens many avenues for future research. Our cost-model can clearly be expanded to account for other costs that might exist in specific scenarios. It is possible that exploring additional costs will lead to even more tweaks to $RN$ that can be more performant for these new costs. In particular, we believe we have only scratched the surface in exploring the two-phase methods. Further analysis and experimentation could help establish stronger ideas of how methods can be combined and what criteria would determine the point for switching phases.

**Appendix**

**Proof that $RN \geq RV$**

Calculating $RN$ requires calculating the average degree of every vertex’s individual collection of neighbors, then averaging all these values across all vertices. We can therefore express $RN$ as:

$$RN = \frac{1}{n} \sum_{v \in V} \left( \sum_{u \in \mathcal{L}_v} d_u \right) \frac{d_v}{d_v} \tag{7}$$

Essentially, the probability of sampling any vertex $v$ as the initial vertex is $1/n$, and the probability of selecting a neighbor $u$ of an initially sampled vertex $v$ is $1/d_v$. Notice that every edge $(u, v)$ contributes $d_u/d_v + d_v/d_u$ to the outer summation, which allows us to express $RN$ as
Table 5  Experimentally calculated $C_v$, $C_n$, and $C_s$ costs of two-phase sampling methods in BA graphs

| Phase 1 Method | Phase 1 | Phase 2 (RV) | Total costs |
|----------------|---------|-------------|-------------|
|                | $C_v$   | $C_n$       | $C_s$       | $C_v$ | $C_n$ | $C_s$ |
| $\rho = .7$    |         |             |             |       |       |       |
| RV             | 4822    | 0           | 2796        | 156   | 47    | 4978  |
| RN             | 1057    | 1057        | 794         | 4060  | 2006  | 5117  |
| RYN            | 854     | 854         | 1307        | 3342  | 1493  | 4197  |
| $RKN-k = 1$ (RN) | 1057    | 1057        | 794         | 4060  | 2006  | 5117  |
| $RKN-k = 2$    | 532     | 1065        | 798         | 4058  | 2002  | 4590  |
| $RKN-k = 3$    | 356     | 1068        | 800         | 4049  | 2000  | 4405  |
| $RKN-k = 6$    | 262     | 1123        | 851         | 3982  | 1949  | 4243  |
| $RKN-k = 7$    | 251     | 1130        | 856         | 3976  | 1944  | 4227  |
| $RKN-k = 8$    | 243     | 1133        | 863         | 3973  | 1937  | 4216  |
| $RKN-k = \infty$ | 202    | 1210        | 930         | 3883  | 1870  | 4085  |
| RV$kN-k = 1$ (RN) | 854     | 854         | 1307        | 3342  | 1493  | 4197  |
| RV$kN-k = 2$    | 472     | 943         | 1087        | 3664  | 1713  | 4135  |
| RV$kN-k = 3$    | 328     | 984         | 1007        | 3776  | 1794  | 4104  |
| RV$kN-k = 6$    | 247     | 1057        | 1001        | 3789  | 1800  | 4036  |
| RV$kN-k = 7$    | 235     | 1060        | 997         | 3791  | 1803  | 4026  |
| RV$kN-k = 8$    | 231     | 1075        | 1005        | 3785  | 1796  | 4016  |
| RV$kN-k = \infty$ | 193    | 1160        | 1044        | 3727  | 1757  | 3920  |
| RV $\rho = .8$ |         |             |             |       |       |       |
| $RKN-k = 1$ (RN) | 6399    | 0           | 3184        | 253   | 53    | 6651  |
| $RKN-k = 2$    | 472     | 943         | 1087        | 3664  | 1713  | 4135  |
| $RKN-k = 3$    | 328     | 984         | 1007        | 3776  | 1794  | 4104  |
| $RKN-k = 6$    | 247     | 1057        | 1001        | 3789  | 1800  | 4036  |
| $RKN-k = 7$    | 235     | 1060        | 997         | 3791  | 1803  | 4026  |
| $RKN-k = 8$    | 231     | 1075        | 1005        | 3785  | 1796  | 4016  |
| $RKN-k = \infty$ | 193    | 1160        | 1044        | 3727  | 1757  | 3920  |
| RV$kN-k = 1$ (RN) | 854     | 854         | 1307        | 3342  | 1493  | 4197  |
| RV$kN-k = 2$    | 472     | 943         | 1087        | 3664  | 1713  | 4135  |
| RV$kN-k = 3$    | 328     | 984         | 1007        | 3776  | 1794  | 4104  |
| RV$kN-k = 6$    | 247     | 1057        | 1001        | 3789  | 1800  | 4036  |
| RV$kN-k = 7$    | 235     | 1060        | 997         | 3791  | 1803  | 4026  |
| RV$kN-k = 8$    | 231     | 1075        | 1005        | 3785  | 1796  | 4016  |
| RV$kN-k = \infty$ | 193    | 1160        | 1044        | 3727  | 1757  | 3920  |
| RV $\rho = .9$ |         |             |             |       |       |       |
| $RKN-k = 1$ (RN) | 120     | 0           | 3583        | 398   | 43    | 9518  |
| $RKN-k = 2$    | 472     | 943         | 1087        | 3664  | 1713  | 4135  |
| $RKN-k = 3$    | 328     | 984         | 1007        | 3776  | 1794  | 4104  |
| $RKN-k = 6$    | 247     | 1057        | 1001        | 3789  | 1800  | 4036  |
| $RKN-k = 7$    | 235     | 1060        | 997         | 3791  | 1803  | 4026  |
| $RKN-k = 8$    | 231     | 1075        | 1005        | 3785  | 1796  | 4016  |
| $RKN-k = \infty$ | 193    | 1160        | 1044        | 3727  | 1757  | 3920  |
Using Eq. 8, we claim

The right side of the inequality is simply the sum of all degrees in the graph, or $2m$

The left side of the inequality contains $m$ terms in the form of $\frac{a}{b} + \frac{b}{a}$ and $\frac{a}{b} + \frac{b}{a} \geq 2$ for all $a, b$, with $a > 0, b > 0$.

We can also derive the following corollary:

**Corollary:** in a graph with at least one edge between vertices of unequal degrees, $RN > RV$

**Proof** $\frac{a}{b} + \frac{b}{a} > 2$ for all $a, b$ with $a > 0, b > 0$, and $a \neq b$.

### Abbreviations

RV: Random vertex sampling; RN: Random neighbor sampling; RVN: Random vertex and neighbor sampling; RkN: Random k neighbor sampling; RVkN: Random vertex and k neighbor sampling; BA: Barabási-Albert; ER: Erdős Rényi.

### Acknowledgements

Not applicable.

### Author contributions

ABN oversaw the research and suggested analytical directions. YN conducted the experiments and analytics and wrote the text of the paper. Both authors read and approved the final manuscript.

### Funding

Not applicable.

### Availability of data and materials

Sample real-world networks for some experiments were taken from the Koblenz network collection, http://konect.uni-koblenz.de/.
Declarations
Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Author details
1Computer Science Department, City University of New York Graduate Center, New York, USA. 2Computer Science Department, Touro University, New York, USA. 3Computer Science Department, Brooklyn College and Graduate Center, City University of New York, Brooklyn, USA.

Received: 27 February 2022   Accepted: 12 May 2022

Published online: 01 June 2022

References
Barabási A-L, Albert R (1999) Emergence of scaling in random networks. Science 286(5439):509–512
Bertotti ML, Modanese G (2019) The bass diffusion model on finite Barabasi-Albert networks. Complexity 2019
Christakis NA, Fowler JH (2010) Social network sensors for early detection of contagious outbreaks. PLoS ONE 5(9):12948
Cohen R, Havlin S, Ben-Avraham D (2003) Efficient immunization strategies for computer networks and populations. Phys Rev Lett 91(24):247901
Erdös P, Gallai T (1960) Gráfok előírt fokú pontokkal. Mat lapok 11:264–274
Erdos P, Rényi A et al (1960) On the evolution of random graphs. Publ Math Inst Hung Acad Sci 5(1):17–60
Faloutsos M, Faloutsos P, Faloutsos C (1999) On power-law relationships of the internet topology. ACM SIGCOMM Comput. Commun. Rev. 29(4):251–262
Feld SL (1991) Why your friends have more friends than you do. Am J Sociol 96(6):1464–1477
Hakimi SL (1962) On realizability of a set of integers as degrees of the vertices of a linear graph. i. J Soc Ind Appl Math 10(3):496–506
Han B, Li J, Srivivasan A (2013) Your friends have more friends than you do: identifying influential mobile users through random-walk sampling. IEEE/ACM Trans Netw 22(5):1389–1400
Havel V (1955) A remark on the existence of finite graphs. Casopis Pest Mat 80:477–480
Jackson MO (2019) The friendship paradox and systematic biases in perceptions and social norms. J Polit Econ 127(2):777–818
Kumar V, Krackhardt D, Feld S (2021) Interventions with inversity in unknown networks can help regulate contagion. arXiv preprint arXiv:2105.08758
Kunegis J (2013) Konect: the koblenz network collection. In: Proceedings of the 22nd international conference on world wide web, pp 1343–1350
Leskovec J, Faloutsos C (2008) Sampling from large graphs. In: Proceedings of the 12th ACM SIGKDD international conference on knowledge discovery and data mining, pp 631–636
Lü L, Chen D, Ren X-L, Zhang Q-M, Zhang Y-C, Zhou T (2016) Vital nodes identification in complex networks. Phys Rep 650:1–63
Momeni N, Rabitat MG (2018) Effectiveness of alter sampling in social networks. arXiv preprint arXiv:1812.03096
Newman ME (2002) Assortative mixing in networks. Phys Rev Lett 89(20):208701
Novick Y, Bar-Noy A (2021) A fair-cost analysis of the random neighbor sampling method. In: International conference on complex networks and their applications. Springer, pp 3–15
Novick Y, Bar-Noy A (2020) Finding high-degree vertices with inclusive random sampling. In: International conference on complex networks and their applications. Springer, pp 319–329
Pal S, Yu F, Novick Y, Swami A, Bar-Noy A (2019) A study on the friendship paradox-quantitative analysis and relationship with assortative mixing. Appl Netw Sci 4(1):1–26
Praveenman M, Prokopenko M, Zomaya AY (2010) Classifying complex networks using unbiased local assortativity. In: ALIFE, pp 329–336
Thedchanamoorthy G, Piraveenan M, Kathiriratna D, Senanayake U (2014) Node assortativity in complex networks: an alternative approach. Procedia Comput Sci 29:2449–2461

Publisher’s note
Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.