De-anonymizing scale-free social networks by percolation graph matching

Abstract—**We address the problem of social network de-anonymization when relationships between people are described by scale-free graphs. In particular, we propose a rigorous, asymptotic mathematical analysis of the network de-anonymization problem while capturing the impact of power-law node degree distribution, which is a fundamental and quite ubiquitous feature of many complex systems such as social networks. By applying bootstrap percolation and a novel graph slicing technique, we prove that large inhomogeneities in the node degree lead to a dramatic reduction of the initial set of nodes that must be known a priori (the seeds) in order to successfully identify all other users. We characterize the size of this set when seeds are selected using different criteria, and we show that their number can be as small as n^ϵ, for any small $\epsilon > 0$. Our results are validated through simulation experiments on real social network graphs.**

I. Introduction

The increasing availability of always-on connectivity on affordable portable devices, coupled with the proliferation of services and online social platforms, has provided unprecedented opportunities to interact and exchange information among people. At the same time, electronic traces of our communications, searches and mobility patterns, specifically their collection and analysis by service providers and unintended third parties, are posing serious treats to user privacy. This fact raises a number of well known and hotly debated issues, which have recently caused quite a stir in the media.

A distinctive feature of this trend is the uncontrolled proliferation of different accounts/identities associated to each individual. Most of us have more than one mobile subscription, more than one email address, and a plethora of accounts (even multiple) on popular platforms such as Facebook, Twitter, LinkedIn and so on. A specific issue that naturally arises in this context is the identification of the different identities/accounts belonging to the same individual. This problem, which has strong implications with user privacy, is known in the scientific literature as social network de-anonymization (or reconciliation). The two most frequently cited reasons why companies/organizations are interested in network de-anonymization are user profiling (for targeted advertising and marketing research) and national security (i.e., the prevention of terrorism and other forms of criminal activity).

It is fundamental to notice that privacy concerns related to de-anonymization are very subjective: some people do not care at all about providing "personally identifiable information" in their service registrations, explicitly linking their accounts "for-free". As we will see, such users play a fundamental role in the network de-anonymization problem, acting as "seeds" to identify other users. On the other extreme, some people are totally obsessed by the idea of Big Brother spying into their life and compiling tons of information on all of us. Such users try to hide themselves behind anonymous identities containing the minimum possible amount of personal data and linkage information with other identities. In the worst case (for the entity trying to solve the de-anonymization problem), an identity consists just of a random identifier (e.g., a code or a string).

One recent, dramatic discovery in the network security field [1] is the following: user privacy (in terms of anonymity) cannot be guaranteed by just resorting to anonymous identifiers. In particular, the identities used by a user across different systems can be matched together by using only the network structure of the communications made by users (i.e., electronic traces of who has come in contact with whom). More formally, considering just the simple case of two systems, the vertices (i.e., the users) of two social network graphs G_1 and G_2, where edges represent the observed contacts among users in the two systems, can be perfectly matched under very mild conditions on the graph structures [2].

As anticipated, the complexity of the network de-anonymization problem can be greatly reduced by having an initial (even small) number of users already correctly matched (the seeds). Such initial side information is often indeed available, thanks to users who have explicitly linked their accounts, to the presence of compromised or fake users, as well as to other forms of external information providing total or partial correlations among identities. Starting from the seeds, one can design clever algorithms to progressively expand the set of matched vertices, incurring only negligible probability to match wrong pairs [3].

In previous work [4], the number of seeds that allows to de-anonymize two networks has been characterized for the case of Erdös–Rényi random graphs, adopting a convenient probabilistic model for $\mathcal{G}_1$ and $\mathcal{G}_2$. By reducing the graph matching problem to a bootstrap percolation problem, authors identify a phase transition in the number of seeds required by their algorithm. In particular, in the case of sparse networks with average vertex degree $\Theta(\log n)$, the number of seeds that are provably sufficient to match the vertices scales as $\Theta(\frac{n}{\log^{4/3} n})$, which is (unfortunately) only a poly-log factor less than n. One obvious limitation of the results in [4] is that they apply only to Erdös–Rényi random graphs, which are a poor representation of real social networks.

Contribution. In our work we extend the results in [4] by considering a family of random graphs that incorporates one of the most fundamental properties of real social networks (and many other complex systems) not yet considered in analytical work, namely, the scale-free vertex degree distribution [5].

We propose a novel algorithm for graph matching, hereinafter referred to as degree-driven graph matching (DDM), and show that DDM successfully matches a large fraction of the nodes. Similarly to [4], we are interested in the scaling law of the number of seeds that are needed to make the nodes' identification process 'percolate', i.e., to propagate almost to the entire set of nodes.

Our results mark a striking difference with those obtained for Erdös–Rényi graphs. In particular, when initial seeds are uniformly distributed among the vertices, order of $n^{\frac{1}{2}+\epsilon}$ seeds (for an an arbitrarily small ϵ) are sufficient to match most of the vertices. Even more amazing results hold when initial seeds can be chosen (e.g., by the attacker) considering their degree: in this case, as few as n^ϵ seeds are sufficient. The implications of our results are clear: scale-free social networks can be surprisingly simple to match (i.e., de-anonymize), especially when initial seeds are properly selected among the population.

Moreover, scale-free networks appear to be so amenable to de-anonymization that, differently from [4], we can establish our results even in the case of finite average node degree (i.e., we do not need any densification assumption, which is necessary in Erdös–Rényi graphs if only to guarantee connectivity). We remark that an algorithm to match scale-free networks has been recently proposed in [3]. However, in [3] authors do not identify any phase transition effect related to bootstrap percolation. Actually, they consider a simple direct identification strategy that requires $\Omega(\frac{n}{\log n})$ seeds and essentially prove that their algorithm is unlikely to match wrong pairs. Also, their analysis is complicated by the adoption of the preferential attachment model by Barabási and Albert [5], whereas here we adopt a different model of scale-free networks that greatly simplifies the analysis.

Finally, we emphasize that our model captures, in isolation, only the impact of power-law degree, without jointly accounting for other salient features of real social networks such as clustering, community structure and so on. For this reason, we have also empirically validated our findings running the DDM algorithm on realistic data sets. Our preliminary experimental results confirm that real social networks are indeed surprisingly simple to de-anonymize starting from very limited side information.

II. Model and matching algorithm

A. Basic assumptions

We study the network de-anonymization problem in the case of two social networks $\mathcal{G}_1(\mathcal{V}_1, \mathcal{E}_1)$ and $\mathcal{G}_2(\mathcal{V}_2, \mathcal{E}_2)$, although our model and analysis can be extended to the case in which more than two networks are available. Both $\mathcal{G}_1$ and $\mathcal{G}_2$ can be fairly considered to be sub-graphs of a larger, inaccessible graph $\mathcal{G}_\mathrm{T}(\mathcal{V}, \mathcal{E})$ representing the groundtruth, i.e., the underlying social relationships between people. We will assume for simplicity that all graphs above have the same set of vertices $\mathcal{V}$ with cardinality $|\mathcal{V}| = n$, i.e., $\mathcal{V}_1 = \mathcal{V}_2 = \mathcal{V}$, although this assumption can be easily removed by seeking to match only the intersection of vertices belonging to $\mathcal{G}_1$ and $\mathcal{G}_2$. We emphasize that $\mathcal{G}_1$ and $\mathcal{G}_2$ do not necessarily represent subsets of social relationships as observed in totally different systems (e.g., Facebook and Twitter). They could also be obtained within the same communication system (i.e., from traces of emails, or from traces of phone calls), due to the fact that users employs two ID's in the same system (i.e., two email addresses, or two SIM cards).

We need a mathematical model describing how edges $\mathcal{E}_1$ and $\mathcal{E}_2$ are selected from the groundtruth set of edges $\mathcal{E}$. Any such model will necessarily be an imperfect representation of reality, since a large variety of different situations can occur. A user might employ either of her ID's to exchange messages with a friend, or use only one of them to communicate with a given subset of friends. General, realistic models trying to capture possibly heterogeneous correlations (positive or negative) in the set of neighbors of a vertex as seen in $\mathcal{G}_1$ and $\mathcal{G}_2$ become inevitably mathematically intractable. We therefore resort to the same assumption adopted in previous mathematical work [2, 3, 4]: each edge in $\mathcal{E}$ is retained in $\mathcal{G}_1$ (or $\mathcal{G}_2$) with a fixed probability s, independently between $\mathcal{G}_1$ and $\mathcal{G}_2$, and independently of all other edges[1]. This model serves as a reasonable, first-step approximation of real systems, and permits obtaining fundamental analytical insights. Moreover, authors in [2] have experimentally found, by looking at temporal snapshots of an email network, that the above independence assumption is largely acceptable in their case.

Another key element is the model for the underlying social graph $\mathcal{G}_\mathrm{T}$. To understand the impact of the power-law distribution of vertex degree, we have chosen a simple model known in the literature as Chung-Lu random graph model [6]. In contrast to the classic model of Erdös–Rényi, Chung-Lu graphs permit considering a fairly general vertex degree distribution while preserving the nice property of independence among edge probabilities, which is of paramount importance to develop the analysis.

Definition 2.1: A Chung-Lu graph is a random graph of n vertices where each vertex i is associated with a positive weight w_i. Let $\bar{w} = \frac{1}{n}\sum_n w_i$ be the average weight. Given two vertices $i, j \in \mathcal{V}$, with $i \neq j$, the undirected edge (i, j) is included in the graph with probability $p_{ij} = \min\left\{\frac{w_i w_j}{n\bar{w}}, 1\right\}$, independently of the inclusion of any other edge in $\mathcal{E}$.

To avoid pathological behavior, it is customary in the Chung-Lu model to assume that the maximum vertex weight is $O(n^{1/2})$. Doing so, weight w_i essentially coincides with the average degree of vertex i, i.e., $p_{ij} = w_i w_j/(n\bar{w})$. In our work, we will assume for simplicity that weights are deterministic [2] (but note that they depend on n, albeit we avoid explicitly indicating this). A suitable way to obtain a power-law degree sequence with exponent β (with $2 < \beta < 3$, as typically observed in real systems) is to set $w_i = \bar{w}\frac{\beta-2}{\beta-1}\left(\frac{n}{i+i_0}\right)^{1/(\beta-1)}$ where i_0 can be chosen such that the maximum degree is $O(n^{1/2})$. In the following, we will assume $\bar{w}$ to be a finite constant, although our analysis can be easily extended to the more general case in which $\bar{w}$ scales with n.

B. Problem definition

The network de-anonymization problem under study can be formulated as follows. We assume the underlying social network graph $\mathcal{G}_\mathrm{T}(\mathcal{V}, \mathcal{E})$ to be an instance of a Chung-Lu graph having power-law degree distribution with exponent β (with

[1] Two different probabilities for $\mathcal{G}_1$ and $\mathcal{G}_2$ (also different from vertex to vertex) could be considered, provided that they do not depend on n.

[2] Our results generalize to the case of weights being r.v. as well.

$2 < \beta < 3$). However, we cannot access its edge set $\mathcal{E}$. Instead, we know the complete structure of two sub-graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ obtained by independently sampling each edge of $\mathcal{E}$ with probability s. Also, each edge in $\mathcal{E}$ is assumed to be (independently) sampled twice, the first time to determine its presence in $\mathcal{E}_1$, the second time to determine its presence in $\mathcal{E}_2$. Note that the sets of vertices $\mathcal{V}_1$ and $\mathcal{V}_2$ must be considered to be assigned after a random permutation of indexes $1, 2, \ldots, n$. The objective is to find the correct match among them, i.e., to identify all pairs of vertices $[i_1, i_2] \in \mathcal{V}_1 \times \mathcal{V}_2$ such that i_1 and i_2 correspond to the same vertex $i \in \mathcal{V}$.

We define the graph of all possible vertex pairs as the pairs graph $\mathcal{P}(\boldsymbol{\mathcal{V}}, \boldsymbol{\mathcal{E}})$, with $\boldsymbol{\mathcal{V}} = \mathcal{V}_1 \times \mathcal{V}_2$ and $\boldsymbol{\mathcal{E}} = \mathcal{E}_1 \times \mathcal{E}_2$, In $\mathcal{P}(\boldsymbol{\mathcal{V}}, \boldsymbol{\mathcal{E}})$ there exists an edge connecting $[i_1, j_2]$ with $[k_1, l_2]$ iff edge $(i_1, k_1) \in \mathcal{E}_1$ and edge $(j_2, l_2) \in \mathcal{E}_2$. We will slightly abuse the notation and denote the pairs graph related to[3] $\mathcal{G}_\mathrm{T}$ simply by $\mathcal{P}(\mathcal{G}_\mathrm{T})$.

Two pairs $[i_1, j_2]$ and $[k_1, l_2]$ in $\mathcal{P}(\mathcal{G}_\mathrm{T})$, are said to be *conflicting* pairs if either $i_1 = k_1$ and $j_2 \neq l_2$, or $j_2 = l_2$ and $i_1 \neq k_1$. We will refer to pairs $[i_1, i_2]$, whose vertices correspond to the same vertex $i \in \mathcal{G}_\mathrm{T}$ as good pairs, and to all others (e.g., $[i_1, j_2]$) as bad pairs. The generic pair will be denoted by $[*_1, *_2]$.

To help identifying good pairs, we assume there exists a subset of a-priori matched vertices, named *seed* set and denoted by $\mathcal{A}_0(n)$, of cardinality a_0. We will consider two variants of the problem, which differ in the way seeds are assumed to be selected among the n nodes. In the first variant, we assume that seeds can be selected at will among the nodes, but using just information on the vertex degree. In the second variant, we assume that seeds are distributed uniformly at random among the nodes.

Algorithm 1 The PGM algorithm

```
$\mathcal{A}_0 = \mathcal{B}_0 = \mathcal{A}_0(n)$, $\mathcal{Z}_0 = \emptyset$
while $\mathcal{A}_t \setminus \mathcal{Z}_t \neq \emptyset$ do
    $t = t + 1$
    Randomly select a pair $[*_1, *_2] \in \mathcal{A}_{t-1} \setminus \mathcal{Z}_{t-1}$ and add
    one mark to all neighboring pairs of $[*_1, *_2]$ in $\mathcal{P}(\mathcal{G}_\mathrm{T})$.
    Let $\Delta\mathcal{B}_t$ be the set of all neighboring pairs of $[*_1, *_2]$
    in $\mathcal{P}(\mathcal{G}_\mathrm{T})$ whose mark counter has reached threshold $r$
    at time $t$.
    Construct set $\Delta\mathcal{A}_t \subseteq \Delta\mathcal{B}_t$ as follows. Order the pairs
    in $\Delta\mathcal{B}_t$ in an arbitrary way, select them sequentially
    and test them for inclusion in $\Delta A_t$:
    if the selected pair in $\Delta\mathcal{B}_t$ has no conflicting pair in
     $\mathcal{A}_{t-1}$ or $\Delta\mathcal{A}_t$ then
        Insert the pair in $\Delta\mathcal{A}_t$
    else
        Discard it
    $\mathcal{Z}_t = \mathcal{Z}_{t-1} \cup [*_1, *_2]$, $\mathcal{B}_t = \mathcal{B}_{t-1} \cup \Delta\mathcal{B}_t$, $\mathcal{A}_t = \mathcal{A}_{t-1} \cup \Delta\mathcal{A}_t$
return $T = t$, $\mathcal{Z}_T = \mathcal{A}_T$
```

C. Overview of the DDM algorithm

Before providing a high-level description of our matching algorithm (DDM), we briefly recall the simple procedure adopted in [4] in the case of Erdös–Rényi graphs. In essence, their algorithm, referred to as PGM (percolation graph matching), maintains a mark counter, initialized to zero, for any candidate pair $[*_1, *_2] \in \mathcal{P}(\mathcal{G}_\mathrm{T})$ that can still potentially be matched. The counter is increased by one whenever the candidate pair becomes *neighbor* of an already matched pair. Two pairs $[*_1, *_2]$ and $[*_1', *_2']$ are said to be neighbors if they are adjacent on $\mathcal{P}(\mathcal{G}_\mathrm{T})$, i.e., edge $(*_1, *_1') \in \mathcal{E}_1$ and edge $(*_2, *_2') \in \mathcal{E}_2$. Among the candidate pairs whose counter is larger than or equal to a fixed threshold r, the algorithm selects one uniformly at random, adding it to the set of matched pairs. After this, counters are updated. Note that some candidate pairs might have to be permanently discarded because they are conflicting with previously matched pairs. The algorithm proceeds until no more pairs can be matched. Of course seeds will be matched irrespective of their mark counter. The PGM algorithm, although potentially suboptimal, is simple enough that its performance can be predicted using known results from bootstrap percolation [7], establishing a lower bound on the number of seeds required to correctly match almost all vertices. A more formal description of the PGM algorithm is given in Alg. 1, where:

- $\mathcal{B}_t(\mathcal{G}_\mathrm{T})$ is the set of pairs in $\mathcal{P}(\mathcal{G}_\mathrm{T})$ that at time step t have already collected a least r marks. It is composed of good pairs $\mathcal{B}_t'(\mathcal{G}_\mathrm{T})$ and bad pairs $\mathcal{B}_t''(\mathcal{G}_\mathrm{T})$.
- $\mathcal{A}_t(\mathcal{G}_\mathrm{T})$ is the set of matchable pairs at time t. Similarly to $\mathcal{B}_t(\mathcal{G}_\mathrm{T})$, it comprises good pairs $\mathcal{A}_t'(\mathcal{G}_\mathrm{T})$ and bad pairs $\mathcal{A}_t''(\mathcal{G}_\mathrm{T})$. In general, $\mathcal{A}_t(\mathcal{G}_\mathrm{T})$ and $\mathcal{B}_t(\mathcal{G}_\mathrm{T})$ do not coincide as $\mathcal{B}_t(\mathcal{G}_\mathrm{T})$ may include conflicting pairs that are not present in $\mathcal{A}_t(\mathcal{G}_\mathrm{T})$.
- $\mathcal{Z}_t(\mathcal{G}_\mathrm{T})$ is the set of pairs that have been matched up to time t. By construction, $|\mathcal{Z}_t| = t$, $\forall t$.

Similarly to [4], in our work we want to establish lower bounds on the number of seeds by means of bootstrap percolation theory. To do so, our algorithm maintains the simplicity of the PGM algorithm, adding some fundamental improvements to exploit the heterogeneity of vertex degrees. Before explaining our approach, we make the following observations on the PGM algorithm described above for Erdös–Rényi graphs. First, pairs are selected irrespective of the degree of their constituting vertices. Intuitively, in Erdös–Rényi graphs this is not so important, since node degree (which is binomial distributed) is highly concentrated around the mean, and all matchable pairs are essentially equivalent. Second, there exists a unique threshold r, common to all pairs, which is a fixed parameter of the algorithm (subject to the constraint $r \geq 4$).

Our DDM algorithm for power-law graphs is based instead on partitioning the vertices on the basis of their degree. It requires a careful expansion of the set of matched pairs through the various partitions, using also different thresholds and seed sets at the various stages of the process.

In particular, we first isolate a specific slice $\mathcal{P}_1$ of the pair-graph (i.e., a sub-graph of $\mathcal{P}(\mathcal{G}_\mathrm{T})$), induced by vertices having large (but not too large) degree. $\mathcal{P}_1$ includes pairs whose vertices have weights between $\alpha_1 = n^\gamma$ and $\alpha_2 = n^\gamma/2$, where γ is a constant (slightly) smaller than $1/2$. This slice is somehow the crucial one: we show that its percolation triggers the entire matching process, as the identification of all other vertices in the network follows easily after we correctly match

[3] In the following, we generalize the concept of pairs graph to a generic graph $\mathcal{G}$, from which $\mathcal{G}_1$ and $\mathcal{G}_2$ are obtained by independent sampling.

all pairs in $\mathcal{P}_1$. Note that degrees of vertices in $\mathcal{P}_1$ are fairly homogeneous (a constant factor of difference), so that the results for Erdös–Rényi graphs can be adapted to this slice.

Vertices having degree smaller than those in $\mathcal{P}_1$ are partitioned in geometric slices $\mathcal{P}_k$ including vertex pairs with weights between α_k and $\alpha_{k+1} = \alpha_k/2$, with $k \geq 2$. Then, a top-down cascading process is unfolded starting from $\mathcal{P}_1$, where matched pairs in a slice are used as seeds to identify the good pairs in the slice below, and so on.

At last, vertices with very large degree are identified at the end, using as seed set a properly defined subset of previously matched pairs with relatively small degree.

Here we have provided just the basic idea of our DDM algorithm: many subtleties must be addressed to show its correctness. Among them, we emphasize the problem that the DDM algorithm has no direct access to vertex weights (i.e., it does not know the original degree of a vertex in $\mathcal{G}_{\mathrm{T}}$), and can only make use of the observable vertex degrees in $\mathcal{G}_1$ and $\mathcal{G}_2$.

We remark that, when $\bar{w}$ is constant, a finite fraction of good pairs cannot be identified by a threshold-based algorithm like ours. This is due to the fact that a good pair $[i_1, i_2]$ can be identified only if both i_1 and i_2 have at least r neighbors in $\mathcal{G}_1$ and $\mathcal{G}_2$. Clearly, a non-vanishing fraction of vertices in $\mathcal{G}_{\mathrm{T}}$, having bounded degree, gives origin to vertices with degree smaller than r in either $\mathcal{G}_1$ or $\mathcal{G}_2$. This explains why we say that our algorithm can match a large fraction of the nodes, but not all of the nodes (even asymptotically).

III. Notation and preliminary results

We first recall the results on Erdös-Rényi graphs obtained in [4]. In particular, one of the main results that we will use in our analysis is stated in the following theorem [4, Th. 1].

Theorem 1: Let the groundtruth graph be an Erdös-Rényi random graph $G(n, p)$. Let $r \geq 4$ and

$$a_c = \left(1 - \frac{1}{r}\right)\left(\frac{(r-1)!}{n(ps^2)^r}\right)^{\frac{1}{r-1}} . \qquad (1)$$

For $n^{-1} \ll ps^2 \leq s^2 n^{-\frac{4}{r}}$, we have: if $a_o/a_c \to a > 1$, the PGM algorithm matches a number of good pairs equal to $|\mathcal{A}'_T| = n - o(n)$ w.h.p. Furthermore, $\mathcal{A}''_T = \emptyset$ w.h.p.

Observe that, under the assumptions of Theorem 1, we have $T = |\mathcal{A}_T| = |\mathcal{A}'_T| = n - o(n)$. The two corollaries below, which can be derived from the arguments presented in [4], strengthen the result in Theorem 1 and will come in handy in the following.

Corollary 1: For any $\epsilon > 0$, define $t_0 = \min\left(T, \frac{n^{-3/r-\epsilon}}{(ps)^2}\right)$. Then, $\mathcal{B}''_{t_0} = \emptyset$ w.h.p.

When $t_0 = T$, the corollary guarantees that $\mathcal{A}''_T \subseteq \mathcal{B}''_T = \emptyset$, i.e., no bad pairs are matched by the PGM algorithm. When $t_0 < T$ (i.e., for $p \gg \sqrt{\frac{n^{-3/r-\epsilon-1}}{s^2}}$), we complement the above statement with the corollary below.

Corollary 2: Under the conditions of Theorem 1, for $p \gg \sqrt{\frac{n^{-3/r-1}}{s^2}}$, let $t_0 = \frac{n^{-3/r-\epsilon}}{(ps)^2}$ for any $0 < \epsilon < \frac{1}{r}$. Then, $|\mathcal{B}'_{t_0}| = n$ w.h.p.

The fact that, for some $t_0 < T$, $|\mathcal{B}'_{t_0}| = n$ and $\mathcal{B}''_{t_0} = \emptyset$ jointly occur w.h.p. implies that the PGM algorithm matches almost all the good pairs (i.e., $|\mathcal{A}'_T| = n$ and $\mathcal{A}''_T = \emptyset$) w.h.p. This is because, by construction, $\mathcal{A}'_{t_0} = \mathcal{B}'_{t_0}$. Indeed, $\mathcal{B}'_{t_0}$ contains no conflicting pairs and none of the pairs in $\mathcal{B}'_{t_0}$ can be blocked by previously matched bad pairs since $\mathcal{B}''_{t_0} = \emptyset$.

We now extend the above results to Chung-Lu graphs. First, we introduce the key concept of *increasing* property.

Let $\mathcal{H}(\mathcal{V}, \mathcal{E}_H)$ and $\mathcal{K}(\mathcal{V}, \mathcal{E}_K)$ be two random graphs insisting on the same set of vertices $\mathcal{V}$, and such that $\mathcal{E}_H \subseteq \mathcal{E}_K$, i.e., $\mathcal{E}_H$ can be obtained by sampling $\mathcal{E}_K$. We introduce the following partial order relationship: $\mathcal{H}(\mathcal{V}, \mathcal{E}_H) \leq_{st} \mathcal{K}(\mathcal{V}, \mathcal{E}_K)$. Then, we consider a vertex property $\mathcal{R}$ satisfied by a subset of vertices, and denote with $\mathcal{R}(\mathcal{H}) \subseteq \mathcal{V}$ the set of vertices of $\mathcal{H}$ that satisfy property $\mathcal{R}$. We say that $\mathcal{R}$ is *monotonically increasing with respect to the graph ordering relation "$\leq_{st}$"* if $\mathcal{R}(\mathcal{H}) \subseteq \mathcal{R}(\mathcal{K})$ whenever $\mathcal{H} \leq_{st} \mathcal{K}$.

In our case, for any $0 \leq t \leq T$, sets $\mathcal{B}_t$, $\mathcal{B}'_t$, $\mathcal{B}''_t$ are all monotonically increasing with respect to relationship "$\leq_{\mathrm{st}}$" defined on the pairs graph $\mathcal{P}(\mathcal{G}_{\mathrm{T}})$. Instead, nothing can be said on $\mathcal{A}_t$, $\mathcal{A}'_t$ and $\mathcal{A}''_t$ due to the effect of mutual conflicts among pairs (i.e., the presence of a pair in $\mathcal{A}_t$ prevents the further addition of all conflicting pairs in $\mathcal{B}_t$). We will leverage such observations to prove Theorem 2 below.

As a preliminary step, we show that a properly defined sub-graph $\mathcal{G}_0$ of a Chung-Lu graph can be lower and upper bounded (w.r.t. "$\leq_{\mathrm{st}}$" relation) by Erdös-Rényi graphs. Then, we observe that a similar relationship holds for the associated pairs graphs. (Proofs are omitted for brevity; they can be found in [8].)

Proposition 1: Given a Chung-Lu random graph $\mathcal{G}_{\mathrm{T}}(\mathcal{V}, \mathcal{E})$, for any given interval of vertex weights $[w_{\min}, w_{\max}]$, we define: $\mathcal{V}_0 \subseteq \mathcal{V}$, $\mathcal{V}_0 = \{i \in \mathcal{V} | w_i \in [w_{\min}, w_{\max}]\}$ with $|\mathcal{V}_0| = n_0$ and $\mathcal{E}_0 = \{(i,j) \in \mathcal{E} | i, j \in \mathcal{V}_0\}$. Now, consider $\mathcal{G}_0 = (\mathcal{V}_0, \mathcal{E}_0)$, i.e., the sub-graph of $\mathcal{G}_T(\mathcal{V}, \mathcal{E})$ induced by only vertices in $\mathcal{V}_0$. The following relationship holds: $G(n_0, p_{\min}) \leq_{st} \mathcal{G}_0 \leq_{st} G(n_0, p_{\max})$, with $G(n_0, p_{\min})$ and $G(n_0, p_{\max})$ being Erdös-Rényi graphs and $p_{\min} = w^2_{\min}/(n\bar{w})$ and $p_{\max} = w^2_{\max}/(n\bar{w})$.

Proposition 2: Given the above Chung-Lu subgraph $\mathcal{G}_0(\mathcal{V}_0, \mathcal{E}_0)$ and the Erdös-Rényi graphs $G(n_0, p_{\min})$ and $G(n_0, p_{\max})$, consider the two graphs obtained from each of them by independent edge sampling with probability s. Let $\mathcal{P}(\mathcal{G}_0)$, $\mathcal{P}(G(n_0, p_{\min}))$ and $\mathcal{P}(G(n_0, p_{\max}))$ be the corresponding pairs graphs. If $G(n_0, p_{\min}) \leq_{st} \mathcal{G}_0 \leq_{st} G(n_0, p_{\max})$, then $\mathcal{P}(G(n_0, p_{\min})) \leq_{st} \mathcal{P}(\mathcal{G}_0) \leq_{st} \mathcal{P}(G(n_0, p_{\max}))$.

Next, we present our first main result, which shows that the PGM algorithm can successfully match all good pairs in the above specified sub-graph $\mathcal{G}_0$ of a Chung-Lu graph.

Theorem 2: Consider $\mathcal{G}_0$ obtained from $\mathcal{G}_{\mathrm{T}}$ as defined in Proposition 1. The application of the PGM algorithm on $\mathcal{P}(\mathcal{G}_0)$ guarantees that $|A_T(\mathcal{G}_0)| = n_0$ and $\mathcal{A}''_*(\mathcal{G}_0) = \emptyset$ w.h.p., provided that:

1) $n_0 \to \infty$ as $n \to \infty$;

2) $p_{\min} = w^2_{\min}/(n\bar{w})$ satisfies: $p_{\min} \gg \sqrt{\frac{n^{-3/r-1}}{s^2}}$;

3) $p_{\max} = w_{\max}^2/(n\bar{w})$ satisfies: $p_{\max} \le n_0^{-\frac{4}{r}}$;
4) $\lim_{n\to\infty} a_o/a_c > 1$ with a_c computed from (1) by setting $p = p_{\min}$.

Proof: First observe that, if we find t_0 with $t_0 = o(n_0)$ such that $\mathcal{B}''_{t_0}(\mathcal{G}_0) = \emptyset$ w.h.p., then we have w.h.p that $\forall t \le t_0$:

$$|\mathcal{A}_t(\mathcal{G}_0)| = |\mathcal{B}'_t(\mathcal{G}_0)| \overset{(a)}{\ge} |\mathcal{B}'_t(G(n_0, p_{\min}))| \overset{(b)}{=} |\mathcal{A}_t(G(n_0, p_{\min}))| \overset{(c)}{>} t. \qquad (2)$$

In (2), inequality (a) holds by monotonicity of sets $\mathcal{B}'_t$ with respect to "$\le_{\text{st}}$", while equality (b) descends from Theorem 1. Inequality (c) descends from the following argument. Denoted by $T_G = \min\{t\,, s.t.\, |\mathcal{A}_t(G(n_0, p_{\min})| = t\}$, by Theorem 1 we have $T_G = n_0 - o(n_0)$. Since $t_0 = o(n_0)$, $t_0 < T_G$, i.e., $|\mathcal{A}_t(G(n_0, p_{\min}))| > t$ for $t \le t_0$. From (2), we immediately get $t_0 < T$, with $T = \min\{t\,, s.t.\, |\mathcal{A}_t(\mathcal{G}_0)| = t\}$.

Now, let us define, for an arbitrarily small $\epsilon > 0$, $t_0 = \frac{n_0^{-3/r-\epsilon}}{(p_{\max} s)^2}$; observe that, by construction, $t_0 = o(n_0)$. We prove that $\mathcal{B}''_{t_0}(\mathcal{G}_0) = \emptyset$ exploiting the monotonicity of $\mathcal{B}''_{t_0}$ with respect to "$\le_{\text{st}}$". Indeed, $|\mathcal{B}''_{t_0}(\mathcal{G}_0)| \le |\mathcal{B}''_{t_0}(\mathcal{G}(n_0, p_{\max}))|$, with $\mathcal{B}''_{t_0}(\mathcal{G}(n_0, p_{\max})) = \emptyset$ w.h.p., as immediate consequence of Corollary 1 (recall that $n_0 \to \infty$ as $n \to \infty$). Furthermore, by Corollary 2, for an arbitrary $0 < \epsilon' < 1/r$, define $t_1 = \frac{n_0^{-3/r-\epsilon'}}{(p_{\min} s)^2} = o(n_0)$. We have: $|\mathcal{B}'_{t_1}(\mathcal{G}(n_0, p_{\min})| = n_0$. Next, by monotonicity, we get $|\mathcal{B}'_{t_1}(\mathcal{G}_0)| \ge |\mathcal{B}'_{t_1}(\mathcal{G}(n_0, p_{\min})| = n_0$, provided that $t_1 \le T$.

At last, since $p_{\max}/p_{\min} = K^2$, we can always choose an $\epsilon < \epsilon'$ such that $T > t_0 = \frac{n_0^{-3/r-\epsilon}}{(p_{\max} s)^2} > \frac{n_0^{-3/r-\epsilon'}}{(p_{\min} s)^2} = t_1$. Thus, since $\mathcal{B}'_{t_0}(\mathcal{G}_0)$ is by construction non-decreasing with t, we have: $|\mathcal{B}'_{t_0}(\mathcal{G}_0)| \ge |\mathcal{B}'_{t_1}(\mathcal{G}_0)| = n_0$. In conclusion, there exists a $t_0 < T$ such that $|\mathcal{B}'_{t_0}(\mathcal{G}_0)| = n_0$ and $\mathcal{B}''_{t_0}(\mathcal{G}_0) = \emptyset$. Hence, $|\mathcal{A}'_T(\mathcal{G}_0)| = |\mathcal{A}'_{t_0}(\mathcal{G}_0)| = |\mathcal{B}'_{t_0}(\mathcal{G}_0)| = n_0$ and $|\mathcal{A}''_T(\mathcal{G}_0)| = |\mathcal{B}''_{t_0}(\mathcal{G}_0)| = 0$. ∎

IV. DDM ALGORITHM AND ANALYSIS

Here we present the details of the DDM algorithm and prove the following main results of its analysis.

(i) For a sufficiently large seed set, the DDM algorithm successfully matches $\Theta(n)$ good pairs and no bad pairs. Moreover, it matches all good pairs (except for a negligible fraction) constituted by vertices with sufficiently high weight, i.e., a weight that tends to infinity as $n \to \infty$.

(ii) The above result holds for a seed set as small as n^{ϵ} (with any arbitrary $\epsilon > 0$) when the seeds can be chosen based on the vertices' degree. When, instead, seeds are uniformly distributed among the nodes, $n^{\frac{1}{2}+\epsilon}$ seeds are sufficient.

(iii) More in general, when seeds are arbitrarily distributed, the key parameter governing the percolation of the graph matching process is not the number of seeds but the cardinality of the set of edges connecting the initially matched pairs (associated to the seeds) to the other pairs of the graph.

We start by generalizing a node partitioning approach originally proposed in [9] for the study of bootstrap percolation in power-law random graphs. We slice the pairs graph $\mathcal{P}(\mathcal{G}_{\text{T}})$ into subgraphs $\mathcal{P}_x$, including pairs of vertices whose weight is comprised between thresholds α_x and α_{x+1} ($x \in \mathbb{N}$). By doing so, we assume the vertices weights to be directly accessible by the DDM algorithm. In practice, this is not possible: the DDM algorithm has direct access only to vertex degrees on $\mathcal{G}_1$ and $\mathcal{G}_2$. In the Appendix, we present a technique to work around this issue and relax the above assumption.

Slices of the pairs graph are constructed as follows:

(i) $\mathcal{P}_0$ includes pairs whose vertices have weights between $\alpha_0 = n^{1/2}$ and $\alpha_1 = n^{\gamma}$, with $0 < \gamma < 1/2$;

(ii) $\mathcal{P}_1$ includes pairs whose vertices have weights between $\alpha_1 = n^{\gamma}$ and $\alpha_2 = n^{\gamma}/2$;

(iii) $\mathcal{P}_k$ includes vertex pairs with weights between α_k and α_{k+1}, with $k \ge 2$, $\alpha_k = \alpha_{k-1}/2$, $\alpha_k > \left(\frac{8\bar{w}\log n}{Cs^2(1-\epsilon)^2}\right)^{\frac{1}{3-\beta}}$ for some $\epsilon > 0$;

(iv) $\mathcal{P}_h$ includes vertex pairs with weights between α_h and α_{h+1}, with $\alpha_h = \alpha_{h-1}/2$ and $\alpha_h \le \left(\frac{8\bar{w}\log n}{Cs^2(1-\epsilon)^2}\right)^{\frac{1}{3-\beta}}$ but $\alpha_h \to \infty$ as $n \to \infty$;

(v) $\mathcal{P}_q$ includes vertices with weights between α_q and α_{q+1}, with $\alpha_q = \alpha_{q-1}/2$ and $\limsup \alpha_q < \infty$.

We initially leave out slice $\mathcal{P}_0$, populated by just few nodes with very large degree (the hubs). The reason is that these nodes share a non-negligible number of neighbors, hence including them at the beginning of the matching process would likely lead to errors. Instead, we consider $\mathcal{P}_0$ at the very end, when most of the nodes have already been identified, and hubs can then be matched more carefully without risk of error.

We therefore start the node identification process from $\mathcal{P}_1$. In essence, we consider the matching process in $\mathcal{P}_1$ in isolation, using just the seeds initially present in it, and we establish sufficient conditions for the correct identification of all good pairs in $\mathcal{P}_1$, applying Theorem 2. We denote by $\mathcal{A}_0^1$ the seed set in $\mathcal{P}_1$.

Proposition 3: All good pairs are successfully matched in $\mathcal{P}_1$, provided that the following conditions are jointly satisfied: $\frac{1}{4} - \frac{3}{2r} < \gamma < \frac{1}{\beta-1}$, $r \ge \frac{4[1+\gamma(1-\beta)]}{1-2\gamma}$ and $|\mathcal{A}_0^1| \gg n^{\frac{(1-2\gamma)r+\gamma(\beta-1)-1}{r-1}}$.

Proof: First, we compute the number of good pairs in $\mathcal{P}_1$, denoted by N_1, and make sure that N_1 grows to infinite when $n \to \infty$ (as requested by condition 1) of Theorem 2. We have:

$$N_1 = \sum_{i\in\mathcal{V}} \mathbb{1}_{\{w_i \in [\alpha_2, \alpha_1]\}} \approx \int_{\alpha_2}^{\alpha_1} n x^{-\beta}\, \mathrm{d}x = C n^{1+\gamma(1-\beta)}$$

where C is a proper constant. Clearly, $N_1 \to \infty$ provided that $1 + \gamma(1-\beta) > 0$, i.e., $\gamma < \frac{1}{(\beta-1)}$. Now, probabilities $p_{\min}$ and $p_{\max}$, defined as in Theorem 2, satisfy the following relationship:

$$p_{\min,\max} = \Theta\left(\frac{n^{2\gamma}}{n\bar{w}}\right) = \Theta(n^{2\gamma-1}).$$

To verify condition 2) in Theorem 2, we must have: $-\frac{3}{2r} - \frac{1}{2} < 2\gamma - 1$, thus $\gamma > \frac{1}{4} - \frac{3}{4r}$, and to verify condition 3) (i.e., $p_{\max} < N_1^{-\frac{4}{r}}$), we need: $n^{2\gamma-1} \le n^{[1+\gamma(1-\beta)]4/r}$ or, equivalently,

$$r \geq \frac{4[1+\gamma(1-\beta)]}{1-2\gamma}. \tag{3}$$

Next, we observe that:

$$a_1^c(N_1) = \left(1-\frac{1}{r}\right)\left(\frac{(r-1)!}{N_1 p_{\min}^r}\right)^{1/(r-1)} = \Theta(n^{\frac{(1-2\gamma)r+\gamma(\beta-1)-1}{r-1}}).$$

Thus, condition 4) of Theorem 2 is surely satisfied if $|\mathcal{A}_0^1| \gg n^{\frac{(1-2\gamma)r+\gamma(\beta-1)-1}{r-1}}$. ■

The above proposition already provides one of our key results. Essentially, it states that we can chose any $\frac{1}{4} \leq \gamma < \frac{1}{2}$ and determine a minimal threshold r and a minimal $|\mathcal{A}_0^1|$ such that all good pairs in $\mathcal{P}_1$ can be successfully identified (after this, the process easily percolates to the rest of the network, as we will see). Note that, if we want to minimize $|\mathcal{A}_0^1|$, γ should be chosen as close as possible to $\frac{1}{2}$ (i.e., $\gamma = \frac{1}{2} - \epsilon$ for some small ϵ). Then, asymptotically, for a sufficiently large r, we can make the seed set arbitrarily small (in order sense) and still correctly match all pairs.

We now consider slice $\mathcal{P}_k$ ($k > 1$) and prove that: (i) the node identification process successfully propagates from one slide to the next and (ii) no errors are made. To this end, we first compute the number of edges from the good pairs in a slice toward those in the slice above and show that the probability that this number is smaller than or equal to a given threshold goes to 0 sufficiently fast. We emphasize that in the following analysis it is important to explicitly find the minimum value of n for which the above result holds. Indeed, later we will have to show that a correct identification is guaranteed to occur uniformly over all the considered slices, for sufficiently large n.

Theorem 3: Consider the good pairs $[i_1, i_2] \in \mathcal{P}_k$, with vertex weight $w_i \in [\alpha_{k+1}, \alpha_k]$. For any such pair $[i_1, i_2] \in \mathcal{P}_k$, and any $\epsilon > 0$, with probability greater than $1 - n^{-2}$, the number of its neighboring good pairs $[l_1, l_2] \in \mathcal{P}_{k-1}$ is greater than $\rho_k = \max(4, \frac{(\alpha_k)^{4-\beta}}{\sqrt{n}})$, as long as $\left(\frac{8\bar{w}\log n}{Cs^2(1-\epsilon)^2}\right)^{\frac{1}{3-\beta}} = \alpha_k^* < \alpha_k < n^\gamma$ (with $1/4 < \gamma < 1/2$), and $n > n_1 = \max\left\{\exp\left[\left(\frac{8\bar{w}}{Cs^2}\right)^{2-\beta}\epsilon^{\beta-3}\right], \left(\frac{2\bar{w}}{Cs^2\epsilon}\right)^{\frac{2}{1-2\gamma}}\right\}$. Furthermore, the above property holds uniformly over the good pairs in $\mathcal{P}_k$ with a probability greater than $1 - n^{-1}$, under the same conditions as before on α_k and n.

Proof: Given a pair $[i_1, i_2] \in \mathcal{P}_k$, for any pair $[l_1, l_2] \in \mathcal{P}_{k-1}$, we denote with $\mathbb{1}_{i,l}$ the indicator function associated to the presence of an edge between $[i_1, i_2]$ and $[l_1, l_2]$ in $\mathcal{P}(\mathcal{G}_\mathrm{T})$. Note that $\mathbb{E}[\mathbb{1}_{i,l}] \geq \frac{\alpha_{k+1}\alpha_k s^2}{n\bar{w}} = \frac{\alpha_k^2 s^2}{2n\bar{w}} = p_{\min}$, and that $\mathbb{1}_{i,l}$'s are independent r.v. Thus, by denoting the number of good pairs in $\mathcal{P}_{k-1}$ with $N_{k-1} = Cn\alpha_k^{(1-\beta)}$, and defining $\mu = N_{k-1}p_{\min} = Cns^2\alpha_k^{1-\beta}\frac{\alpha_k^2}{2n\bar{w}} = \Theta\left(s^2(\alpha_k)^{3-\beta}\right)$, for any $\rho_k < \mu$, we have:

$$\begin{aligned}\mathbb{P}\left(\sum_{l\in\mathcal{P}_{k-1}} \mathbb{1}_{i,l} \leq \rho_k\right) \quad &< \quad \mathbb{P}(\mathrm{Bi}(N_{k-1}, p_{\min}) \leq \rho_k) \\ &\leq \quad \exp(-\delta^2\mu/2) \end{aligned} \tag{4}$$

with $\delta = \frac{\mu-\rho_k}{\mu}$. In the above derivation, the first inequality descends from the fact that $\sum_{[l_1,l_2]\in\mathcal{P}_{k-1}} \mathbb{1}_{i,l}$ can be stochastically lower bounded by a sum of N_{k-1} independent Bernoulli r.v. with average $p_{\min}$, while the second descends from the Chernoff bound. Now, let us fix $\rho_k = \max\left(4, \frac{(\alpha_k)^{4-\beta}}{\sqrt{n}}\right) = o(\mu)$. For any $\epsilon > 0$ and choosing $\delta = 1 - \epsilon$, we have that, whenever $\rho_k < (1-\delta)\mu = \epsilon\mu$,

$$\mathbb{P}\left(\sum_{[l_1,l_2]\in\mathcal{P}_{k-1}} \mathbb{1}_{i,l} \leq \rho_k\right) < \exp((1-\epsilon)^2\mu/2).$$

It is straightforward to see that $\exp((1-\epsilon)^2\mu/2) < n^{-2}$ provided that $\mu > 4\log n/(1-\epsilon)^2$, which corresponds to $\alpha_k > \left(\frac{8\bar{w}\log n}{Cs^2(1-\epsilon)^2}\right)^{\frac{1}{3-\beta}}$.

Then, we have that $\mathbb{P}\left(\sum_{[l_1,l_2]\in\mathcal{P}_{k-1}} \mathbb{1}_{i,l} \leq \rho_k\right) < n^{-2}$ provided that, for some $\epsilon > 0$, jointly $\alpha_k > \alpha_k^* = \left(\frac{8\bar{w}\log n}{Cs^2(1-\epsilon)^2}\right)^{\frac{1}{3-\beta}}$ and $\rho_k < (1-\delta)\mu = \epsilon\mu$. The last condition can be reformulated in terms of n as[4]: $n > n_1 = \max\left\{\exp\left[\left(\frac{8\bar{w}}{Cs^2}\right)^{2-\beta}\epsilon^{\beta-3}\right], \left(\frac{2\bar{w}}{Cs^2\epsilon}\right)^{\frac{2}{1-2\gamma}}\right\}$.

At last, jointly considering all pairs in $\mathcal{P}_k$, the probability that $\sum_{[l_1,l_2]\in\mathcal{P}_{k-1}} \mathbb{1}_{i,l} \leq \rho_k$ for some $[i_1, i_2] \in \mathcal{P}_k$, is:

$$\begin{aligned}&\mathbb{P}\left(\exists[i_1,i_2]\in\mathcal{P}_k | \sum_{[l_1,l_2]\in\mathcal{P}_{k-1}} \mathbb{1}_{i,l} \leq \rho_k\right) \\ &\leq \sum_{[i_1,i_2]\in\mathcal{P}_k} \mathbb{P}\left(\sum_{[l_1,l_2]\in\mathcal{P}_{k-1}} \mathbb{1}_{i,l} \leq \rho_k\right) < nn^{-2} = n^{-1} \end{aligned} \tag{5}$$

provided that jointly $n > n_1$ and $\alpha_k^* < \alpha_k < n^\gamma$, as immediate consequence of probability sub-additivity. ■

Similarly, the theorem below proves that the probability that a bad pair has a number of neighboring good pairs greater than, or equal to, a given threshold tends to zero.

Theorem 4: Consider the bad pairs $[i_1, j_2]$, with vertex weights $w_i, w_j < \alpha_k$, being $\alpha_k < n^\gamma$ ($\gamma < 1/2$). Uniformly over such pairs $[i_1, j_2]$, for any $n > n_2 = \max\left\{\left(\frac{272Cs^4}{\bar{w}^2}\right)^{\frac{2(4-\beta)}{3-\beta}}, \left(\frac{36Cs^4}{\bar{w}^2}\right)^{\frac{2}{1-2\gamma}}\right\}$, with probability greater than $1 - n^{-1}$, the number of their neighboring good pairs $[l_1, l_2] \in \mathcal{P}_k$ is smaller than $\rho_k = \max\left(4, \frac{(\alpha_k)^{4-\beta}}{\sqrt{n}}\right)$.

The proof follows the same lines as in Theorem 3, thus it is omitted for brevity. We only remark that the average number of good pairs in $\mathcal{P}_k$, which are neighbors of a bad pair $[i_1, j_2]$, is $\mu = \Theta\left(\frac{s^2(\alpha_k)^{5-\beta}}{n}\right) = O(\frac{\alpha_k}{\sqrt{n}}\rho_k)$ with $\frac{\alpha_k}{\sqrt{n}} < n^{\gamma-1/2}$.

Theorems 3 and 4 provide the basic ingredients to show that the DDM algorithm can match all good pairs in slices $\mathcal{P}_k$, for $k \geq 2$, with $\alpha_k > \alpha_k^* = (\frac{8\bar{w}\log n}{Cs^2(1-\epsilon)^2})^{\frac{1}{3-\beta}}$. Indeed, we can show that the identification of good pairs successfully propagates from one slice to the next (up to α_k^*), even without requiring a "local" seed set in $\mathcal{P}_k$. Specifically, the matching

[4] The second term in the right hand side of the inequality can be easily obtained by upper bounding α_k with n^γ.

process performed by the DDM algorithm can be divided into stages. At stage $k+1$, we fix $r_{k+1} = \rho_k = \max(4, \frac{\alpha_k^{4-\beta}}{\sqrt{n}})$ and match all of the candidate unmatched pairs of vertices, with weight smaller than α_k, that have at least r_{k+1} neighbors among the already matched pairs in $\mathcal{P}_k$. Observe that the success of the whole recursion through k is guaranteed again by sub-additivity of probability. Indeed, given that the number of stages is by construction upper bounded by $\frac{\log n}{2}$, for $n > \max(n_1, n_2)$:

$$\mathbb{P}\left(\exists k | \text{either not all good pairs in } \mathcal{P}_k \text{ are matched} \right. \\ \left. \text{or some bad pair is matched}\right) \leq n^{-1} \log n \,. \quad (6)$$

Next, we consider slices $\mathcal{P}_h$ with $\alpha_h \leq \alpha_k^*$. The same algorithm with $r_h = 4$ can be applied, however only a weaker form of percolation occurs in this case.

Theorem 5: Consider the good pairs $[i_1, i_2] \in \mathcal{P}_h$, with vertex weight $w_i \in [\alpha_{h+1}, \alpha_h]$. Also, assume that, for some $\eta > 0$, at least a fraction η of neighboring good pairs, $[l_1, l_2] \in \mathcal{P}_{h-1}$, have been previously identified. Then, for any $0 < \epsilon < 1$, at least a fraction $(1-\epsilon)$ of pairs $[i_1, i_2] \in \mathcal{P}_h$ have a number of neighbors among the identified pairs $[l_1, l_2] \in \mathcal{P}_{h-1}$ greater than 4 w.h.p., as long as $\alpha_h \to \infty$.

Proof: We employ again the indicator function $\mathbb{1}_{i,l}$ and repeat the same arguments as in the proof of Theorem 3. Given any $0 < \eta < 1$, we define $\mu = \eta N_{h-1} p_{\min} s^2 = \eta C n s^2 \alpha_h^{1-\beta} \frac{\alpha_h^2}{2n\bar{w}} = \Theta\left(s^2 (\alpha_k)^{3-\beta}\right)$. Since $4 \ll \mu$, we have:

$$\mathbb{P}\left(\sum_{l \in \mathcal{P}_{h-1}, l \text{ identified}} \mathbb{1}_{i,l} \leq 4 \right) < \\ \mathbb{P}(\text{Bi}(\eta N_{h-1}, p_{\min}) \leq 4) \leq \exp(-\delta^2 \mu/2) \quad (7)$$

with $\delta = \frac{\mu-4}{\mu}$ and as long as $\alpha_h \gg 1$.

Let us denote by Y_h the random variable indicating the number of vertices in $\mathcal{P}_h$ that have at least 4 neighbors among the vertices in $\mathcal{P}_{h-1}$, which have been previously identified. Then, the above result implies that: $\mathbb{E}[Y_h] \geq (1 - \exp(-\delta^2\mu/2))N_h = N_h - o(N_h)$. Thus, for a sufficiently large n such that $\exp(-\delta^2\mu/2) < \epsilon/2$, (i.e., $\mu > \max\left(8, -4 \log \frac{\epsilon}{2}\right)$ and $\mathbb{E}[Y_h] > (1-\epsilon/2)N_h$), recalling that $0 < \epsilon < 1$, we have: $\mathbb{P}(Y_h \leq (1-\epsilon)N_h) < e^{[-\epsilon^2 (1-\frac{\epsilon}{2}) \frac{N_h}{8}]} \to 0$, as $\alpha_h \to \infty$. ■

Furthermore, consider slices in the interval $h \in [h_{\min}, h_{\max}]$, where $h_{\min}$ has been chosen so as to guarantee $\alpha_{h_{\min}} \geq (\frac{8\bar{w} \log n}{Cs^2(1-\epsilon)^2})^{\frac{1}{3-\beta}}$, while $h_{\max}$ is such that $\alpha_{h_{\max}} \to \infty$. Then, a sufficiently large n_3 can be found such that, uniformly on $h \in [h_{\min}, h_{\max}]$, we have $\mu_h > \max\left(8, -4\log\frac{\epsilon}{2}\right)$ (i.e., $\exp(-\delta^2\mu_h/2) < \epsilon/2$). This because, by construction, for every n, μ_h is decreasing with h. Thus, if for a given n the expression $\mu_{h_{\max}} > \max\left(8, -4\log\frac{\epsilon}{2}\right)$ holds, the relationship is automatically satisfied for any $h < h_{\max}$. Now, for $n \geq n_3$, by sub-additivity of probability we can bound the probability that the DDM algorithm at some stage fails to identify at least a fraction $1-\epsilon$ of good pairs. Specifically, the bound is given by: $\sum_{h_{\min}}^{h_{\max}} \exp\left(-\epsilon^2\left(1-\frac{\epsilon}{2}\right)N_h/8\right) = \sum_{h_{\min}}^{h_{\max}} \exp\left(-\epsilon^2\left(1-\frac{\epsilon}{2}\right)N_{h_{\min}} 2^{(h-h_{\min})(\beta-1)}/8\right) = \Theta(\exp(-\epsilon^2(1-\epsilon/2)N_{h_{\min}+1}/8)) \to 0$. We conclude that, for any $\epsilon > 0$, we can iteratively identify at least a fraction $1-\epsilon$ of good pairs jointly in all slices w.h.p., as long as for each slice h the assumptions of Theorem 5 are satisfied for some $\eta > 0$.

At last, we consider slices $\mathcal{P}_q$ such that $\alpha_q = \Theta(1)$.

Theorem 6: Consider the good pairs $[i_1, i_2] \in \mathcal{P}_q$, with vertex weight $w_i \in [\alpha_{q+1}, \alpha_q]$. A finite fraction $f(\alpha_q)$ ($0 < f(\alpha_q) < 1$) of such pairs have a number of neighbors among the identified pairs $[l_1, l_2] \in \mathcal{P}_{q-1}$ greater than 4, with a probability at least $1 - n^{-1}$. This result holds provided that at least a fraction $f(\alpha_{q-1}) \geq f(\alpha_q)$ of neighboring good pairs $[l_1, l_2] \in \mathcal{P}_{q-1}$ (i.e., pairs whose vertices have weight $w_j \in [\alpha_q, \alpha_{q-1}]$) have been previously identified. The above property holds for properly selected values of $f(\alpha_q)$, whenever $\alpha_q > (\frac{32\bar{w}}{Cs^2 f(\alpha_q)})^{\frac{1}{3-\beta}}$ and $n > \frac{2\alpha_q^{\beta-1}}{10^4 Cs^2 f(\alpha_q)}$.

Proof: Define Y_q as in the proof of Theorem 5. If $\mathbb{E}[Y_q] > (1+\epsilon)f(\alpha_q)N_q$, for some $\epsilon > 0$, we can claim:

$$\mathbb{P}(Y_h \leq f(\alpha_q)N_q) < \exp\Big(-\epsilon^2 \mathbb{E}[Y_q]/2\Big) < n^{-1} \quad (8)$$

as long as $n > \left(\frac{4E[w]}{\epsilon^2 Cs^2 f(\alpha_{q-1})}\right)^2$. Now, $\mathbb{E}[Y_q] > N_q(1 - \exp(-\delta^2 f(\alpha_q)\mu_q/2))$ with $\mu_q \geq Cs^2 \alpha_q^{1-\beta} \frac{\alpha_q^2}{2\bar{w}}$ and $\delta = \frac{f(\alpha_q)\mu_q - 4}{f(\alpha_q)\mu_q}$. Thus, to guarantee $\mathbb{E}[Y_q] > (1+\epsilon)f(\alpha_q)N_q$, we impose $N_q(1-\exp(-\delta^2 f(\alpha_q)\mu_q/2)) \geq (1+\epsilon)f(\alpha_q)N_q$, i.e., $1 - \exp(-\delta^2 f(\alpha_q\mu_q/2)) \geq (1+\epsilon)f(\alpha_q)$, from which we can derive the minimal value of μ_q and the maximal $f(\alpha_q)$ for which the previous inequality holds. ■

As before, the joint application of Theorem 6 to all slices $\mathcal{P}_{q-1}$ with $\alpha_q > (\frac{32\bar{w}}{Cs^2 f(\alpha_q)})^{\frac{1}{3-\beta}}$ permits concluding that at least a fraction of good pairs in each slice $\mathcal{P}_{q-1}$ is matched w.h.p., while no bad pairs are matched (again thanks to Theorem 4). In conclusion, a fraction $\Theta(n)$ of vertices is successfully identified by our algorithm.

At last, the DDM algorithm recovers the pairs initially left out in slice $\mathcal{P}_0$. Theorem 7 (whose proof is omitted for brevity) guarantees that all good pairs in $\mathcal{P}_0$ (and only them) can be matched.

Theorem 7: Consider a generic pair $[i_1, i_2] \in \mathcal{P}_0$ with $w_i > n^{\gamma}/2$, and a slice $\mathcal{P}_k$ such that $\alpha_k \leq \log^2 n$. For a sufficiently large n, with probability greater than $1 - n^{-1}$, the number of good pairs $[l_1, l_2] \in \mathcal{P}_k$ that are neighbors of $[i_1, i_2]$ is greater than $\rho_0 = n^{\gamma/2}$. Also, for sufficiently large n, with probability greater than $1 - n^{-2}$, the number of neighboring good pairs $[l_1, l_2] \in \mathcal{P}_k$ of bad pair $[i_1, j_2] \in \mathcal{P}_0$ is smaller than ρ_0, The above properties hold uniformly over all good pairs in $\mathcal{P}_0$ w.h.p.

A. *Uniformly distributed seeds*

Up to know we have assumed that all the initial seeds in $\mathcal{A}_0$ belongs to $\mathcal{P}_1$. Now we show that the DDM algorithm can properly percolate when seeds are uniformly distributed over the slices. Note that, although the uniform distribution is probably the most interesting one, our results hold for an arbitrary distribution of the seeds over the nodes. We start introducing the key parameter that characterizes the ability to start the bootstrap percolation process over $\mathcal{P}_1$ (and then over the whole $\mathcal{P}(\mathcal{G}_\text{T})$):

Definition 1: We denote by $\partial\mathcal{A}_0$ the set of edges between the seed set $\mathcal{A}_0$ and the rest of pairs $\mathcal{P}(\mathcal{G}_{\rm T}) \setminus \mathcal{A}_0$.

Theorem 8: Whenever the seed set $\mathcal{A}_0$ is chosen in such a way that:

$$|\partial\mathcal{A}_0| \gg n^{\gamma + \frac{(1-2\gamma)r+\gamma(\beta-1)-1}{r-1}},$$

our DDM algorithm successfully percolates identifying $\Theta(n)$ good pairs.

Proof: We proceed as follows. By exploiting the monotonicity property of the percolation process, we can show that a properly dimensioned set of seeds belonging to slice $\mathcal{P}_k$, with $k > 1$, is equivalent to a single seed belonging to $\mathcal{P}_1$. Similar arguments can be used to show that a group of seeds in $\mathcal{P}_1$ behaves as a seed in $\mathcal{P}_0$. More formally, we consider the evolution of the DDM algorithm operating on a seed set $\mathcal{A}_0$ of pairs in $\mathcal{P}_1$. Then, we compare it to the evolution of a modified version of the DDM algorithm operating on a seed set $\mathcal{A}_0^*$, which differs from $\mathcal{A}_0$ in that a fraction of seeds in $\mathcal{P}_1$ is replaced with a group of seeds, S_k, in $\mathcal{P}_k$.

The modified version of the DDM algorithm handles every group of seeds belonging to $\mathcal{P}_k$ as a single seed (i.e., all the seeds in the same group are selected by the algorithm at the same time and simultaneously included in $\mathcal{Z}$). Also, while proceeding, the two versions of the algorithm process exactly the same sequence of seeds. We show that, by properly setting S_k, we can guarantee that the identification of good pairs spreads faster starting from $\mathcal{A}_0^*$ than from $\mathcal{A}_0$.

Consider a generic good pair $[i_1, i_2]$ in $\mathcal{P}_1$. Note that, by construction, the number of edges between $[i_1, i_2]$ and a given pair $[l_1, l_2] \in \mathcal{A}_0$ is either 0 or 1. The probability that such edge exists in $\mathcal{P}(\mathcal{G}_{\rm T})$ is upper-bounded by $p_{1,1} = \frac{w_i \alpha_1}{n\bar{w}}$. Instead, the probability that at least an edge exists between $[i_1, i_2]$ in $\mathcal{P}_1$ and the corresponding group of S_k seeds in $\mathcal{P}_k$ is lower-bounded by $p_{1,S_k} = 1 - (1 - \frac{w_i \alpha_{k+1}}{n\bar{w}})^{S_k}$. By setting $S_k > \frac{\alpha_1}{\alpha_{k+1}} + \epsilon$ for any $\epsilon > 0$, it can be easily shown that, for sufficiently large n, $p_{1,S_k} > p_{1,1}$, i.e., the group of S_k seeds belonging to $\mathcal{P}_k$ in $\mathcal{A}_0^*$ distributes to any good pair in $\mathcal{P}_1 \setminus \mathcal{A}_0$ a number of marks that upper bounds those distributed by the corresponding seed in $\mathcal{A}_0$. This immediately implies that $\mathcal{B}'_t(\mathcal{A}_0^*) \setminus \mathcal{A}_0 \supseteq \mathcal{B}'_t(\mathcal{A}_0) \setminus \mathcal{A}_0$ for any t. Therefore, at t_1 defined as in Theorem 2, $\mathcal{B}'_{t_1}(\mathcal{A}_0^*)$ must necessarily include all pairs in $\mathcal{P}_1 \setminus \mathcal{A}_0$. In addition, it is straightforward to show that every pair in $\mathcal{A}_0 \setminus \mathcal{A}_0^*$ has at least r neighbors among good pairs in $\mathcal{P}_1 \setminus \mathcal{A}_0$ and, thus, it is included in $\mathcal{B}'_{t_1}(\mathcal{A}_0^*)$.

To conclude the proof, we have to show that $\mathcal{B}''_{t_1}(\mathcal{A}_0^*) = \emptyset$. This can be done by following the lines of Theorem 2, i.e., by uniformly upper-bounding the probability of adding marks at any time t to bad pairs in $\mathcal{P}_1$, and, then, repeating the arguments of Corollary 1. Iterating the previous argument for all slices containing seeds, we get the assertion. ■

From Theorem 8, it immediately descends that, for any choice of seeds, we can correctly match $\Theta(n)$ good pairs provided that the size of the seed set is at least of order $n^{\frac{1}{2}+\epsilon}$, for an arbitrarily small ϵ.

V. Experimental validation

Our results hold asymptotically as the number of nodes tends to infinite, thus it is difficult to validate them considering networks of finite size. Nevertheless, in this section, we show that the dramatic impact of power-law degree on the performance of graph matching algorithms is already evident on small-scale systems. Another important goal of this section is to check whether Chung-Lu graphs, which only capture effects due to the (marginal) degree distribution of the nodes, can indeed predict the performance achievable in real social networks, which possess several other features not accounted for by the simple Chung-Lu model.

In our first experiment, we took a publicly available, early snapshot of Facebook containing friendship data of users [10]. This graph contains 63,371 nodes, the average node degree is 25.64, and the power law exponent, estimated using the maximum-likelihood approach [11], is 2.9412 (quite large). To understand the impact of network structure, we proceed as follows: we generate a $G(n,p)$ (Erdös-Rényi) graph with the same average degree as the Facebook snapshot, and a Chung-Lu graph which, besides the average, reproduces also the power-law exponent of the Facebook snapshot, using the simple weight sequence introduced in Sec. II-A. We obtain three graphs, which are used, in turn, as groundtruth network $\mathcal{G}_{\rm T}$. We fix the edge sampling probability to $s = 0.7$.

We run the PGM algorithm on the $G(n,p)$ graph, and a simplified version of the DDM algorithm on both the Chung-Lu and the Facebook graphs, considering either the case of seeds uniformly distributed, or seeds selected only among nodes whose degree lies in the interval $[\sqrt{n}/2, \sqrt{n}]$. I.e., we take $\gamma = 1/2$ for the first slice, even though in theory we should take a value slightly smaller than $1/2$. For a more meaningful comparison, our simplified version of DDM employs a constant threshold $r = 4$ for all slices, the same used in PGM. Results are reported on Fig. 1, in which, for each considered number of seeds, we average the total number of matched nodes obtained in 100 different runs[5].

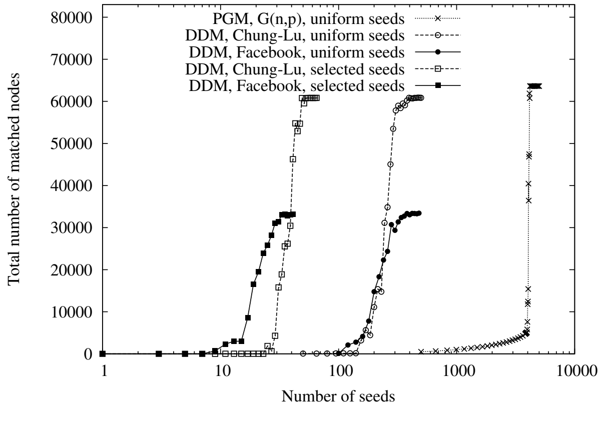

Fig. 1. Total number of matched nodes vs number of seeds, for different graphs and algorithms, in the case of $s = 0.7$

We clearly see a phase transition effect in all cases, but the position of the transition changes dramatically (notice the log x scale). Even a power-law exponent of 2.9 can reduce the threshold associated to a $G(n,p)$ graph by more than one order of magnitude, still considering uniformly distributed seeds. A reduction of another order of magnitude is gained by selecting

[5] The three graphs are fixed, but randomness is present in the identity of the initial seeds and within the algorithms themselves.

all seeds in the initial slice of DDM. Very interestingly, the position of the threshold is more or less the same in the Chung-Lu graph and in the real Facebook snapshot, meaning that taking into account the power-law exponent alone allows us to predict the performance of graph matching algorithms in a real social network quite well.

Note that, using the Facebook snapshot, the total number of matched nodes does not go beyond 33K. This is due to the fact that a large fraction of nodes in this graph have degree smaller than 4, hence they cannot be matched in any case[6]. At last, we report some figures for the fraction of bad pairs matched by our algorithm in the above experiment (negligible errors were produced by PGM in the Erdös-Rényi graph). We consider only the fraction of bad pairs at the phase transition point, because here the error is known to be maximum [4]. We observed about 0.001 (0.0002) fraction of bad pairs using the Chung-Lu graph, respectively with uniform and selected seeds. The Facebook snapshot produced slightly more matching errors, 0.05 and 0.02, respectively. However, we do not consider these errors really significant, as they could be reduced by a more careful selection of threshold r, without affecting the scaling-order performance gains of our algorithm.

In our second experiment, we used a much larger Youtube graph with 3.2M nodes and $\beta = 2.2$. Results of this experiment, similar to those in Fig. 1, are reported in [8].

VI. Conclusions

We analytically investigated the de-anonymization problem in social networks represented by scale-free graphs, by exploiting bootstrap percolation results and a novel graph slicing technique. Our main finding is that, to successfully identify most of the nodes, the seed set can be as small as n^ϵ (for any $\epsilon > 0$) when seeds are properly selected, and of the order of $n^{\frac{1}{2}+\epsilon}$ when they are uniformly distributed among the nodes. Our asymptotic results, experimentally validated by simulation experiments with real social networks, suggest that taking into account the power-law degree distribution alone effectively allows us to predict the surprising performance achievable by graph matching algorithms in realistic social networks.

Appendix

In Sec. IV we assumed that the pairs graph $\mathcal{P}(\mathcal{G}_{\mathrm{T}})$ is directly sliced into subgraphs $\mathcal{P}_k$. In practice, only $\mathcal{G}_1$ and $\mathcal{G}_2$ (and the corresponding pairs graph) can be sliced according to the observed degree of the nodes. Here, we show that the impact of such imperfect knowledge can be made negligible.

As a first step, we observe that the original vertex weight in $\mathcal{G}$ can be simply estimated from the observed degree in $\mathcal{G}_1$ (or $\mathcal{G}_2$). Indeed, given a vertex i_1 in $\mathcal{G}_1$ with degree D_1^i, the estimated weight associated to it is just $\hat{w}_i^1 = D_1^i/s$. By slicing $\mathcal{G}_1$ (or $\mathcal{G}_2$) on the basis of such estimate, it is clear that any slice may include vertices with different weights than those expected. We now show how to build an imperfect slice $\mathcal{P}_k'$ with estimated weights in the range $[\alpha_{k+1}, \alpha_k]$, such that the following three conditions are satisfied: 1) only pairs formed by vertices whose actual weight is in the interval $[\alpha_{k+1}, \alpha_k]$ are included in $\mathcal{P}_k'$; 2) only a finite fraction of good pairs of $\mathcal{P}_k$ is not included in $\mathcal{P}_k'$; 3) the following event occurs with negligible probability: a bad pair $[i_1, j_2]$ is included in the slice, while at the same time neither of pairs $[i_1, i_2]$ or $[j_1, j_2]$ is included. The third condition ensures that every bad pair in $\mathcal{P}_k'$ conflicts with at least one good pair in $\mathcal{P}_k'$, thus it cannot be matched by the DDM algorithm when it (eventually) reaches the threshold.

To guarantee that the above conditions hold, we build $\mathcal{P}_k'$ as follows. We partition the interval $[\alpha_{k+1}, \alpha_k]$, into two sub-intervals: $[\alpha_{k+1}(1+\epsilon), \alpha_k(1-\epsilon)]$, with $0 < \epsilon \leq 1/4$, is called *inner* region, while the remaining range of values is called *outer* region. The idea is to include in $\mathcal{P}_k'$ pairs of vertices whose weights fall either in the inner or in the outer region, adding the extra constraint that only pairs for which at least one vertex falls in the inner region are included in $\mathcal{P}_k'$. This implies that $[i_1, j_2]$ is included in $\mathcal{P}_k'$ only if i_1 (j_2) falls in the inner region and i_2 (j_1) falls in the inner plus outer region.

Then, by applying standard concentration results, we can easily show that, as long as $\alpha_{k+1} > \frac{65}{\epsilon^2} \log n$, for sufficiently large n the above conditions 1), 2) and 3) are satisfied with probability greater than $1 - n^{-1}$.

Theorem 2 can then be extended to show that our DDM algorithm correctly percolates within slice $\mathcal{P}_1'$ (a detailed proof is reported in [8]). Similarly, it can be shown (by slightly generalizing and strengthening Theorems 3-6) that the cascading process through slices successfully takes place when slices are imperfect. The only requirement is that the seed set at *every* stage of the algorithm is properly adjusted so as to ensure that conditions 1), 2) and 3) are met (see [8] for a detailed explanation).

References

[1] A. Narayanan, V. Shmatikov, “De-anonymizing social networks,” *IEEE Symp. on Security and Privacy*, 2009.

[2] P. Pedarsani, M. Grossglauser, “On the privacy of anonymized networks,” *SIGKDD*, 2011.

[3] N. Korula, S. Lattanzi, “An efficient reconciliation algorithm for social networks,” *PVLDB*, 2014.

[4] L. Yartseva, M. Grossglauser, “On the performance of percolation graph matching,” *ACM COSN*, 2013.

[5] A.-L. Barabási, R. Albert, “Emergence of scaling in random networks,” *Science*, vol. 286, pp. 509–512, 1999.

[6] F. Chung, L. Lu, “The average distance in a random graph with given expected degrees,” *Internet Mathematics,* vol. 1, no. 1, pp. 91–113, 2003.

[7] S. Janson, T. Luczak, T. Turova, T. Vallier, “Bootstrap percolation on the random graph $G_{n,p}$,” *The Annals of Applied Probability,* vol. 22, no. 5, pp. 1989–2047, 2012.

[8] Anonymous Technical Report https://www.dropbox.com/s/t8x8pfqk3n52rlj/Tech_rep.pdf

[9] H. Amini, N. Fountoulakis, “Bootstrap percolation in power-law random graphs,” http://arxiv.org/abs/1111.1339, accessed in June 2014.

[10] B. Viswanath, A. Mislove, M. Cha, K. P. Gummadi, “On the evolution of user interaction in Facebook,” *WONS*, 2009.

[11] A. Clauset, C.R. Shalizi, and M.E.J. Newman, “Power-law distributions in empirical data,” *SIAM Review*, vol. 51, no. 4, pp. 661–703, 2009.

[6] This does not occur with the Chung-Lu graph, in which low-degree nodes are almost not present, since we decided to reproduce just the tail behavior (power-law exponent) of the Facebook degree distribution.