Fragmentation properties of two-dimensional proximity graphs considering random failures and targeted attacks

C. Norrenbrock,* O. Melchert,† and A. K. Hartmann‡
Institut für Physik, Universität Oldenburg, 26111 Oldenburg, Germany
(Received 17 September 2015; revised manuscript received 15 June 2016; published 19 December 2016)

The pivotal quality of proximity graphs is connectivity, i.e., all nodes in the graph are connected to one another either directly or via intermediate nodes. These types of graphs are often robust, i.e., they are able to function well even if they are subject to *limited* removal of elementary building blocks, as may occur for random failures or targeted attacks. Here, we study how the structure of these graphs is affected when nodes get removed successively until an extensive fraction is removed such that the graphs fragment. We study different types of proximity graphs for various node-removal strategies. We use different types of observables to monitor the fragmentation process, simple ones like the number and sizes of connected components and more complex ones like the hop diameter and the backup capacity, which is needed to make a network $N-1$ resilient. The actual fragmentation turns out to be described by a second-order phase transition. Using finite-size scaling analyses we numerically assess the threshold fraction of removed nodes, which is characteristic for the particular graph type and node deletion scheme; this suffices to decompose the underlying graphs.

DOI: 10.1103/PhysRevE.94.062125

I. INTRODUCTION

Robustness describes the ability of networks to withstand random failures and targeted attacks. This quantity is of particular importance for many real-world networks, which are crucial for the proper functioning of modern human societies. This means that these networks must work well even if some network components malfunction. Examples are electrical power grids [1], urban road networks [2], airline networks [3], and communication networks like the Internet [4].

In the last decade, various studies have been published that focus on networks which exhibit a scale-free degree distribution. In particular, the fragmentation properties of scale-free Barabási-Albert (BA) networks [5–10] and other variants [11–15] have been subjected to scrutiny. In the aforementioned articles, different node-removal strategies have been considered to investigate the robustness of the considered networks. It turns out that scale-free networks are robust against random node removals but very vulnerable to intentional attacks targeting particularly "important" nodes. Note that there are many local and global measures to quantify whether a node is important. Popular choices are, e.g., the degree of a node, its betweenness centrality [16] (subject to a particular metric used to measure the length of the shortest paths between pairs of nodes), or, somewhat more specific to the hyperlink structure of the Internet, the PageRank [17] relevance measure for Web pages.

In the presented work we focus on types of networks which are completely different from scale-free graphs. The networks considered here are constructed from sets of points distributed in the two-dimensional (2D) Euclidean plane. Therefore, they are particularly suited to model networks which are physically embedded on the earth's surface, like road networks, power grids, water pipelines, and grids of antennas for mobile phones.

More precisely, we consider four types of networks. Three of them belong to the class of *proximity graphs*, namely, *relative neighborhood graphs* (RNGs) [18], *Gabriel graphs* (GGs) [19], and *Delaunay triangulations* (DTs) [20]. These are graphs where pairs of nodes are connected by undirected edges if they are considered to be *close* in some sense; see the definitions in Sec. II. Also, these graphs are *planar* [21], i.e., the edges do not cross. The fourth network type is a certain type of geometric random graph, termed a *minimum-radius* (MR) graph, where each pair of nodes is connected if their mutual distance does not exceed a particular threshold value. Therefore, these graphs are also embedded in a finite-dimensional space but they are usually not planar. Thus, in contrast to the other three types, the MR graph is not a proximity graph.

Proximity graphs extract the relevant structure of point sets, thus they find application in fields where the structure of an associated point set is important. Several applications such as pattern recognition and computer vision are reviewed in [18]. Furthermore, they have been studied in scientific fields such as the simulation of epidemics [22], percolation [23–26], and message routing and information dissemination in *ad hoc* networking [27–30]. To elaborate on the latter point, proximity graphs find application in the construction of planar "virtual backbones" for *ad hoc* networks, i.e., collections of radio devices without a fixed underlying infrastructure, along which information can be efficiently transmitted [27,31–34]. Routing with guaranteed node-to-node connectivity (at least in a multihop manner) is especially important to ensure a complete broadcast of information in *ad hoc* networks [27]. Malfunction of some network components might lead to a drastic decrease in its functionality, e.g., when node-to-node connectivity gets lost, so we find it interesting to determine the stability of those networks regarding different kind of breakdowns.

*christoph.norrenbrock@uni-oldenburg.de
†oliver.melchert@uni-oldenburg.de
‡alexander.hartmann@uni-oldenburg.de

Here, we consider three types of node-removal strategies with different levels of severity (see Sec. III), and we numerically assess the threshold fraction of removed nodes (characteristic for the particular graph type and node deletion scheme), which suffices to decompose the underlying graphs into "small" clusters.

The remaining article is organized as follows. In Sec. II we introduce the four graph types that are considered in the presented study. In Sec. III we describe the three node-removal strategies that were used in order to characterize the fragmentation process for each of these graph types. In Sec. IV we introduce the observables that were recorded during the fragmentation procedure and we list the results of our numerical simulations. Finally, Sec. V concludes with a summary.

II. GRAPH TYPES

Subsequently we introduce four types of graphs for a planar set of, say, N points and we characterize the fragmentation process in each of these graph types following three node-removal strategies, detailed in Sec. III. Three of these graph types, introduced in Secs. II A through II C, belong to the class of proximity graphs [35]. The fourth graph type, detailed in Sec. II D, is a particular type of *random geometric graph*. Below, a graph is referred to as $G=(V,E)$, where V comprises its node set ($N=|V|$; where N is also referred to as the "system size") and where E ($M=|E|$) signifies the respective edge set [21]. Each of the N nodes $u\in V$ represents a point in the 2D unit square for which the coordinates u_x and u_y are drawn uniformly and independently at random. So as to compute the distance $\mathrm{dist}(u,v)$ between two nodes $u,v\in V$ we consider the Euclidean metric under which $\mathrm{dist}(u,v)=[(u_x-v_x)^2+(u_y-v_y)^2]^{1/2}$. We further consider open boundary conditions. Thus an increase in the system size corresponds to an increase in the density of nodes on the unit square. On the other hand, so as to maintain the density of nodes while increasing N, the networks can be pictured as having an effective side length $L=\sqrt{N}$. A common feature of these four types of graphs is that their edge set encodes proximity information regarding the close neighbors of the terminal nodes of a given edge. The different graph types can be distinguished by the precise linking rule that is used to construct the edge set for a given set of nodes. In this section the linking rules that define the four types of proximity graphs are detailed.

A. Relative neighborhood graphs

One particular proximity graph type that is considered subsequently is the relative neighborhood graph [18]. In order to determine whether, in the construction procedure for an instance of an RNG, two nodes $u,v\in V$ need to be connected to each other, it is necessary to check whether there is a third node $w\in V\setminus\{u,v\}$ with $\mathrm{dist}(u,w)\leqslant\mathrm{dist}(v,u)$ and $\mathrm{dist}(v,w)\leqslant\mathrm{dist}(v,u)$. If such a node w does not exist, u and v will get linked. In geometrical terms, for each pair u and v of points, the respective distance $\mathrm{dist}(u,v)$ can be used to construct the *lune* $\mathrm{lune}(u,v)$. The lune is given by the intersection of two circles of equal radius $\mathrm{dist}(u,v)$, centered at u and v, respectively. If no other point $w\in V\setminus\{u,v\}$ lies within $\mathrm{lune}(u,v)$, i.e., if the lune is empty, both nodes are connected by means of an edge. To facilitate intuition, an example of an RNG for a small set of $N=5$ nodes is sketched in Fig. 1(a). A larger example that illustrates the principal structure of a RNG is shown in Fig. 2(a).

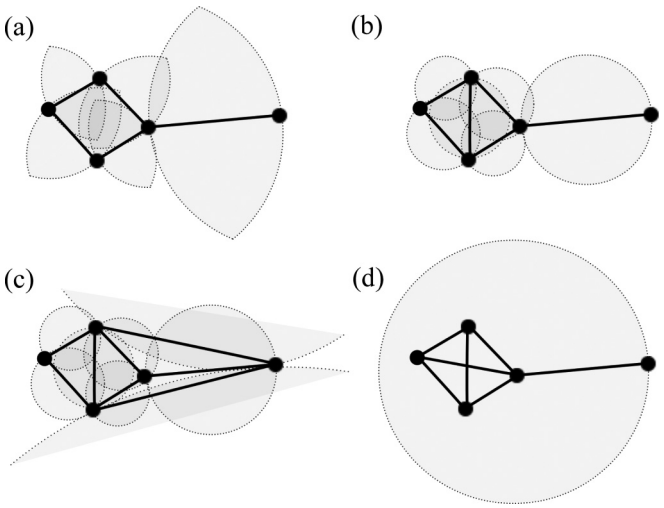

FIG. 1. Examples of the four graph types for a small set of $N=5$ nodes (see text for details). (a) Instance of an RNG, where for all pairs of nodes that will be connected under the respective linking rule, the respective lune is depicted in gray. (b) Instance of a GG, where for all pairs of nodes that will be connected under the respective linking rule, the circle that helps in the decision-making process is depicted in gray. (c) Instance of a DT, where the shaded gray circles are exemplary of those that might aid in the decision-making process. (d) Instance of an MR graph, where the linking range r is depicted (gray circle) for a single node only (all other nodes exhibit the same linking range).

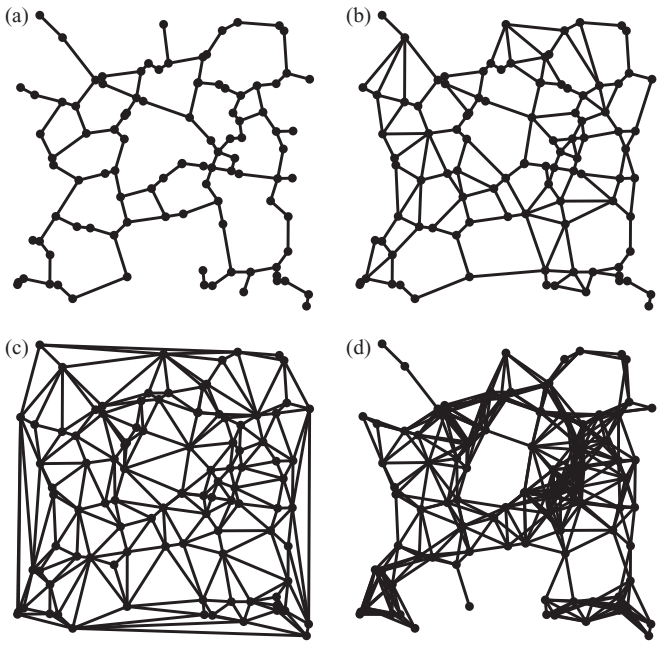

FIG. 2. Examples of the four graph types for the same set of $N=100$ nodes, distributed uniformly at random in the 2D unit square. (a) RNG, (b) GG, (c) DT, and (d) MR.

B. Gabriel graphs

Another proximity graph that is considered in this article is the Gabriel graph [19,23]. To determine whether, in the construction procedure for an instance of a GG, two nodes $u,v \in V$ need to be connected, circ(u,v), i.e., the smallest of all possible circles which embeds both nodes is considered, which has a diameter dist(u,v). These two nodes will be connected unless there is another node w which is located within the area enclosed by circ(u,v). To facilitate intuition, the linking rule for the GG is illustrated in Fig. 1(b). A larger example that illustrates the principal structure of a GG is shown in Fig. 2(b). Further, note that the GG is a supergraph of the RNG. This is due to the circumstance that circ(u,v), which is relevant in the construction procedure of a GG instance for a given set of nodes, encloses a subarea of lune(u,v), being relevant in the construction procedure of the corresponding RNG instance [compare the shaded gray surfaces in Figs. 1(a) and 1(b)]. Therefore, all edges contained in the RNG are also included in the GG. Note that this canis also shown in Figs. 2(a) and 2(b).

C. Delaunay triangulations

The construction of the Delaunay triangulation (also a type of proximity graph) [20] is quite similar. Two nodes $u,v \in V$ will be connected if any circle exists which embeds u as well as v but no further nodes. To facilitate intuition, the result of this linking rule is shown in Fig. 1(c). A larger example that illustrates the principal structure of a DT is shown in Fig. 2(c). From the definition of these linking rules, since the GG also involves the construction of a circle, it is evident that an instance of a DT for a given set of nodes must be a supergraph of the corresponding GG instance. As a consequence, being a subgraph of the GG, the RNG is also a subgraph of the DT. This is shown in Figs. 1(a)–1(c) [Figs. 2(a)–2(c)], where the RNG, GG, and DT are illustrated for the same set of $N = 5$ (100) nodes.

D. Minimum-radius graphs

The fourth network topology considered is the minimum-radius graph, which is not a proximity graph, but a geometric random graph. This means that two nodes $u,v \in V$ will be joined by an edge if dist$(u,v) \leqslant r$. Therein, the "connectivity radius" r is equal to the length of the longest edge in the minimum spanning tree [36]. By this means, it is guaranteed that a finite path between each node pair exists, i.e., all nodes are connected to each other in a multihop manner. It becomes evident from Figs. 1(d) and 2(d) that, in contrast to the previous graphs, the MR graph might feature crossing edges.

E. Graph construction

In order to construct the RNG and GG, we made use of the subgraph hierarchy RNG $\subset$ GG $\subset$ DT. That is, for a given set of nodes we first obtained the DT by means of the Qhull computational geometry library [37] (the DT for a set of N points can be computed in time $O(N \log(N))$ [37,38]) and then pruned the resulting edge set E until the linking requirements of GG or RNG are met. Here, we amend the naive implementation of this two-step procedure [18], yielding an algorithm with running time $O(N^2)$, by means of the "cell-list" method [26], resulting in a subquadratic running time. In this regard, note that Ref. [35] provides an overview of several algorithmic approaches for the construction of RNGs and GGs. Finally, note that RNGs and GGs can be found as the limiting cases of a parameters family of proximity graphs, termed β skeletons [39].

At this point, note that due to a yet unmentioned property of minimum-weight spanning trees (MSTs; i.e., a spanning tree in which the sum of Euclidean edge lengths is minimal; see Ref. [36]), we can set the "connectivity radius" of MR graphs, i.e., a geometric random graph, in the context of proximity graphs. Bear in mind that the longest edge present in any instance of an MR graph specifies the smallest possible edge length which ensures that all nodes are connected to one another. This edge length exactly characterizes the longest edge in the MST of the corresponding node set. For a given set of nodes, an MST is a spanning subgraph of the RNG [18,26]. Thus, considering MSTs, the previously mentioned subgraph hierarchy can be extended to MST $\subset$ RNG $\subset$ GG $\subset$ DT. This allows for the fast construction of an MR instance for a given set of points via a convenient three-step procedure: (i) compute the DT for the given set of points, (ii) filter the edge set of the DT to determine the corresponding MST, and, (iii) use the length of the longest MST edge as the "connectivity radius" to construct the respective MR graph. Therein, the overall running time is dominated by step (iii), which, in its most naive implementation has computational cost $O(N^2)$. Note that during the latter step, the previously mentioned "cell list" method can be used to achieve an improved running time.

Subsequently we introduce the node-removal strategies considered in the numerical simulations carried out to characterize the fragmentation process for the above graph types.

III. NODE-REMOVAL STRATEGIES

As pointed out above, in the presented article we aim at characterizing the fragmentation processes for the graph types introduced in Sec. II. Therefore we consider threet types of node-removal strategies that are used throughout the literature [5–9]. For convenience these are detailed subsequently. Therefore, note that the basic procedure for studying the fragmentation process for a single network instance consists of successively removing nodes until the network is decomposed into many small clusters of nodes, thereby recording observables that provide information about the current characteristics of the network (see Sec. IV).

The most simplistic node-removal strategy followed here is termed *random failure*. According to this strategy, a node is picked uniformly at random and deleted from the network (along with all its incident edges).

Depending on the context into which the networks are set, it might be useful to associate a measure of relevance to each node. Then it is also intuitive to ask for node-removal strategies that preferentially target the most relevant nodes. Removal strategies that capitalize on the relevance of a node are termed *targeted attacks*. Here, we consider two targeted attack strategies:

(i) Degree-based attack (abbreviated "attack 1"), where the relevance of a node is simply measured by its degree (i.e., the number of its incident edges). The higher the degree of a

node, the more relevant it is assumed to be. Accordingly, at each elementary node-removal step during the fragmentation process, the node with the currently highest degree is selected for deletion. If, at a given step, there are many nodes exhibiting the currently highest degree, one of these nodes is chosen uniformly at random. Note that the degree of a node is a local property only, i.e., for a given node one only has to determine the number of its nearest neighbors. Thus, from a computational point of view the node degree is a very inexpensive relevance measure.

(ii) Betweenness-based attack (abbreviated "attack 2"), where the relevance of a node is measured by its betweenness centrality [16]. The betweenness centrality of node u is the number of shortest paths between all node pairs (v,w) $(v,w \neq u)$ that pass through u. The larger the value of the betweenness centrality, the more relevant a node is assumed to be. In some applications, the Euclidean distance along the edges is relevant for determining the shortest paths [36]. However, here we instead considered the hop metric, where distances are simply measured in terms of node-to-node hops. Consequently, the shortest path problem can be solved by means of a breadth-first search [36]. During each elementary node-removal step, the node exhibiting the currently highest value of betweenness centrality gets removed. As before, if several nodes have the same value, one of them is chosen uniformly at random. Note that the betweenness centrality is a global property deduced from the underlying network, i.e., for the betweenness centrality of a particular node, the configuration of shortest paths between all pairs of nodes is of relevance. From a computational point of view this is, of course, considerably more expensive than the computation of the local node degree.

Subsequently, we use the above node-removal strategies in order to characterize the fragmentation process for the graph types described in Sec. II by means of numerical simulations.

IV. RESULTS

In Sec. II E it is illustrated how the subgraph hierarchy RNG $\subset$ GG $\subset$ DT can be used to make the simulations run faster. Additionally, the hierachy can also be used to constrain topological properties such as the average degree and percolation threshold, which is shown in this section.

We report on numerical simulations for the different graph types for planar sets of $N = 144\,(= 12^2)$ up to $36\,864\,(= 192^2)$ points, where results are averaged over 2000 independent graph instances. In Sec. IV A we first report on some topological properties of the graphs; in Sec. IV B the analysis of the fragmentation procedure is summarized. In Sec. IV C further issues concerning the resilience of the networks seen as transport networks ("$N-1$ resilience") are discussed. Finally, in Sec. IV C, the networks are compared under the assumption that they all exhibit the same summed-up edge length.

Subsequently, albeit we present results for all relevant combinations of the four graph types and three node-removal strategies, we so not show figures with results for all these combinations. Instead, so as to illustrate the analyses performed in the following section, we mainly present figures for the RNG proximity graphs subject to a degree-based node-removal strategy.

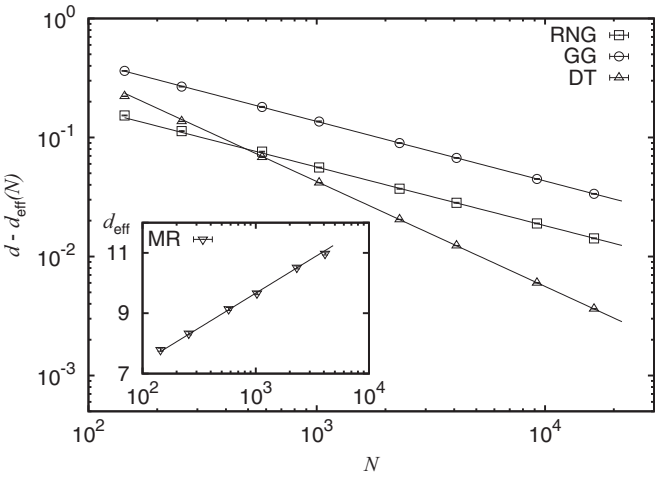

FIG. 3. Finite-size scaling behavior of the average degree for the four graph types. The scaling behavior found for the three types of proximity graph, i.e., RNG, GG, and DT (see text for details). Inset: The logarithmic scaling found in the case of the MR graphs (see text for details).

A. Topological properties

To emphasize structural differences between the graph types of the subgraph hierarchy RNG $\subset$ GG $\subset$ DT we first consider the respective average node degree. Therefore, the scaling behavior of the effective, i.e., system-size-dependent, average degree $d_{\rm eff}(N)$ is considered and analyzed using a fit to the function $d_{\rm eff}(N) = d - aN^{-b/2}$. For the three graph types RNG, GG, and DT the fits yield asymptotic degrees d and scaling exponents b, where $d_{\rm RNG} = 2.557(1)$ and $b_{\rm RNG} = 0.99(4)$ (with a reduced chi-square $\chi^2_{\rm red} = 0.87$; note that both the asymptotic average degree and the scaling exponent compare well to the estimates reported in Ref. [26]), $d_{\rm GG} = 3.999(1)$ and $b_{\rm GG} = 1.00(1)$ (reduced chi-square $\chi^2_{\rm red} = 0.70$), $d_{\rm DT} = 6.0001(1)$ and $b_{\rm DT} = 1.76(1)$ (for a reduced chi-square $\chi^2_{\rm red} = 1.87$; note that the average degree of the DT is known to be $d_{\rm DT} = 6$). In Fig. 3 the correction to scaling, i.e., $d - d_{\rm eff}(N) \propto N^{-b/2}$, is shown for the three types of proximity graphs. It is interesting to note that the RNG and GG exhibit a similar scaling, involving a correction of the form $N^{-1/2}$, whereas the scaling behavior for the average degree for the DT graphs is governed by a significantly larger exponent. Also, note that instances of the three types of proximity graphs are planar, i.e., there are no crossing edges. While the bounding cycles of the finite faces for the instances of RNGs and GGs might consist of an even or odd number of edges, all inner faces for instances of DTs are bounded by three edges.

Further, for the minimum-radius graph we found that the effective average degree fits best to a logarithmic scaling function of the form $d_{\rm eff}(N) = \log(aN)$ (see inset in Fig. 3), where $a = 15.7(4)$ (reduced chi-square $\chi^2_{\rm red} = 1.42$; however, note that the data can also be fit by a scaling function with a small power-law correction as above, where $d_{\rm MR} \approx 57$ and $b_{\rm MR} \approx 0.04$).

Regarding MR graphs, consider that the longest edge present in any instance of an MST (which specifies the length of the longest edge in the respective MR instance; see

discussion above) can by no means exceed the length of the longest edge of any of its supergraphs. Due to the geometric restrictions imposed by going from an instance of a DT to an RNG, it is thus plausible that the maximal edge length found for any MR instance is much shorter than, say, for the corresponding DT instance. This holds in particular for the case of open boundary conditions, where the outer faces of the DT instances feature rather long edges [see Fig. 2(c)]. For a set of 500 instances of point sets consisting of $N = 16\,384$ nodes (i.e., for systems of effective side length $L = 128$) we found that the highest edge length ratios $r_{\max}/L$ for the four graph types read $r^{\mathrm{DT}}_{\max}/L = 0.714(6)$, $r^{\mathrm{GG}}_{\max}/L = 0.029\,27(8)$, $r^{\mathrm{RNG}}_{\max}/L = 0.023\,35(8)$, and, $r^{\mathrm{MR}}_{\max}/L = 0.015\,48(6)$. For the first three graph types, these values should be more or less independent of the system size. On the other hand, for the minimum-radius graphs we found that the finite-size scaling behavior of the connectivity radius $r^{\mathrm{MR}}_{\max}(L)$ as a function of the effective system length L exhibits a logarithmic scaling of the form $r^{\mathrm{MR}}_{\max}(L) = a + b\log(L)$, where $a = 1.334(8)$ and $b = 0.133(2)$ (reduced chi-square $\chi^2_{\mathrm{red}} = 0.59$), supporting the logarithmic scaling of the average degree. That is, the respective "connectivity area" $A_r = \pi r^2_{\max}$, which, if centered at the position of a given node, specifies the area in which all its nearest neighbors can be found, should be almost equal to the previously discussed average degree d_{eff}, because the density of nodes is unity; e.g., at $N = 2304$ (i.e., $L = 48$) we find $A_r = 10.7(2)$ and $d_{\mathrm{eff}} = 10.52(5)$.

B. Fragmentation analysis

For the fragmentation analysis we consider instances of the four dgraph types introduced in Sec. II and successively remove nodes according to one of the node-removal strategies presented in Sec. III until the initially connected graph decomposes into small clusters. So as to determine the critical fraction p of nodes that need to be removed until the graph decomposes we perform a finite-size scaling (FSS) analysis for different observables that are commonly used in studies of percolation [40,41] in Sec. IV B 1. In addition, in Sec. IV B 2 we consider the scaling behavior of the hop diameter, i.e., the longest among all shortest paths measured in terms of node-to-node hops, which, e.g., is relevant in the context of broadcasting problems on networks [27].

1. *Analysis of typical percolation observables*

The observables we consider below can be rescaled following a common scaling assumption. Below, this is formulated for a general observable $y(p,N)$. This scaling assumption states that if the observable obeys scaling, it might be written as

$$y(p,L) = L^{-b} f[(p - p_c)N^{1/(2\nu)}], \tag{1}$$

wherein ν and b represent dimensionless critical exponents (or ratios thereof; see below) and $f[\cdot]$ denotes an unknown scaling function [41,42]. The critical point p_c marks the fraction of removed nodes where the phase transition takes place in the thermodynamic limit, i.e., $N \to \infty$. Following Eq. (1), data curves of the observable $y(p,N)$ recorded at different values of p and N collapse, i.e., fall on top of each other, if $y(p,N)N^{b/2}$ is plotted against $\epsilon \equiv (p - p_c)N^{1/(2\nu)}$ and if, further, the scaling parameters p_c, ν, and b that enter Eq. (1) are chosen properly. The values of the scaling parameters that yield the best data collapse determine the numerical values of the critical exponents that govern the scaling behavior of the underlying observable $y(p,N)$. In order to obtain a data collapse for a given set of data curves we here perform a computer-assisted scaling analysis (see Refs. [43] and [44]).

(a) Order parameter. As the first observable we consider $s_{\max}$, i.e., the relative size of the largest cluster of connected nodes. Averaged over different instances of, say, size N, at a given value of p this yields the order parameter

$$\langle P(p)\rangle = \langle s_{\max}(p)\rangle. \tag{2}$$

This observable scales according to Eq. (1), where $b = \beta/\nu$ and β is the order-parameter exponent. The data curves for the RNG proximity graphs for all three types of node-removal strategies are shown in Fig. 4(a).

For the RNG and GG, the random failure node-removal strategy simply corresponds to ordinary random percolation. An extended study of site and bond percolation for the RNG-type proximity graphs can be found in Ref. [26] and for the GG-type proximity graphs in Ref. [45], respectively. However, note that in these articles p signifies the fraction of occupied bonds/nodes as opposed to the fraction of deleted nodes. The respective values of p_c are listed in Table I. It is apparent that, in the order RNG, GG, DT, and MR, the graphs become less and less susceptible to fragment under random node removal. This correlates well with the average degree $d_{\mathrm{RNG}} < d_{\mathrm{GG}} < d_{\mathrm{DT}} < d_{\mathrm{eff,MR}}(N)$.

Regarding the *degree-based attack strategy* for the RNGs we found that the best data collapse (obtained for the three system sizes $N = 2304$, 4096, and 9126 in the range $\epsilon \in [-1,1]$) yields $p_c = 0.120(1)$, $\nu = 1.33(2)$, and $\beta = 0.148(8)$ with a quality $S = 3.63$ [see Refs. [43] and [44]), and Fig. 4(b)]. Note that the numerical values of the critical exponents match the expected values for 2D percolation, i.e., $\nu = 4/3 \approx 1.333$ and $\beta = 5/36 \approx 0.139$, quite well. Restricting the data analysis to the slightly smaller interval $\epsilon \in [-0.65,0.65]$, enclosing the critical point on the rescaled p axis, the optimal scaling parameters are found to be $p_c = 0.119(1)$, $\nu = 1.41(5)$, and $\beta = 0.14(1)$ with a quality $S = 0.98$. Further, fixing ν and β to their exact values, thus leaving only one parameter to adjust, yields $p_c = 0.119(1)$ with a data-collapse quality $S = 3.16$. Hence, for RNGs subject to a degree-based attack strategy, a fraction of $p_c = 0.119(1)$ seems to suffice in order to decompose the graph instance into small clusters. Note that this is already significantly smaller than the above value found for the case of random node failures.

The analyses for the proximity graph types GG and DT for the above two node-removal strategies (i.e., random failure and degree-based attack) were carried out in a similar fashion. For the DT ensemble, considering the degree-based node-removal strategy, the scaling parameters obtained by the FSS analysis read $p_c = 0.377(1)$, $\nu = 1.31(7)$, and 0.14(2) with a data-collapse quality $S = 0.89$ (obtained for the three system sizes $N = 2304,4096,9126$ in the range $\epsilon \in [-0.5,0.75]$). For comparison: the critical point for the random node-removal strategy is known to be $p_c = 0.5$; from our simulated data we find $p_c = 0.500(2)$, $\nu = 1.35(13)$, and $\beta = 0.13(2)$ with a

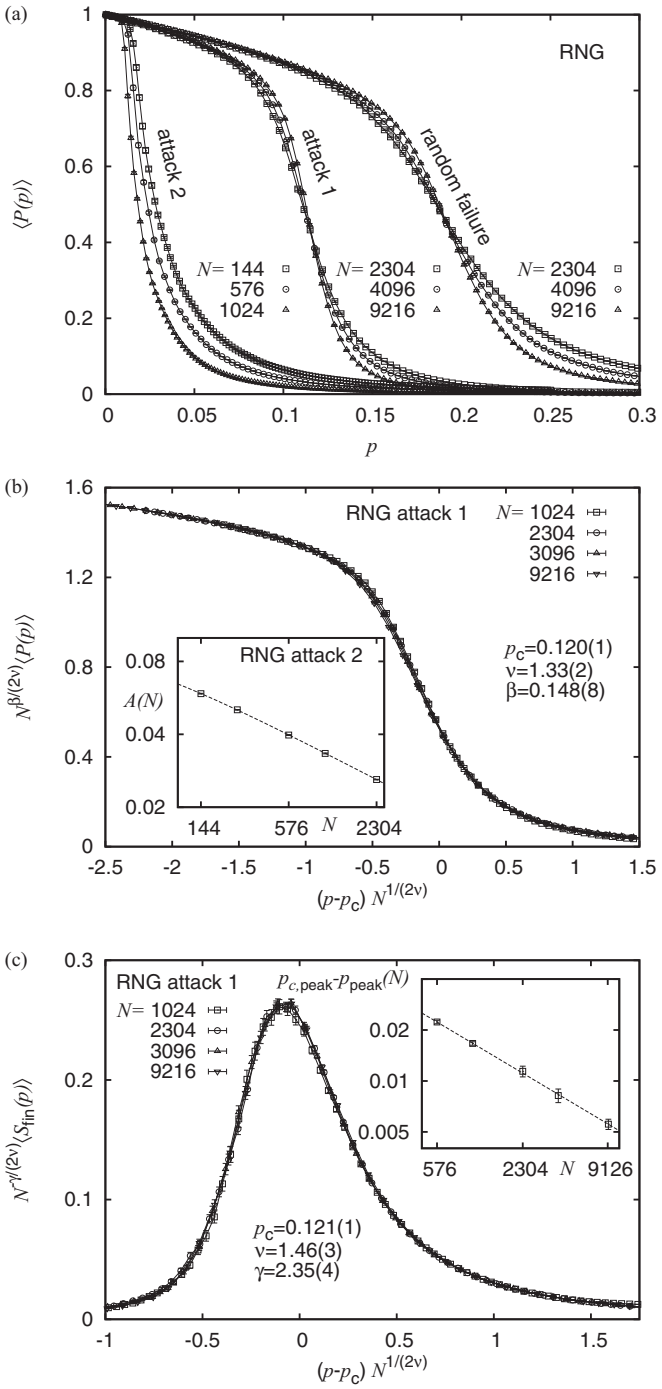

FIG. 4. Finite-size scaling analysis for the RNG proximity graphs. (a) Order parameter for RNGs subject to the three node-removal strategies discussed in Sec. III. (b) The best data collapse of the order parameter obtained for a degree-based node-removal strategyInset: Scaling of the area under the curve for a centrality-based node-removal strategy. The fact that the area seems to converge to 0 is compatible with a critical point $p_c = 0$. (c) The best data collapse for the average size of the finite clusters considering a degree-based node-removal strategy. Inset: Scaling of the peak position.

quality $S = 0.94$ (similar system sizes as above, only in the range $\epsilon \in [-0.25, 0.25]$).

For the case of the GG graphs, we found $p_c = 0.263(1)$, $\nu = 1.33(4)$, and $\beta = 1.4(2)$ with respect to the degree-based node-removal strategy (obtained for the system sizes $N = 2304$, 4096, and 9126 in the range $\epsilon \in [-0.2, 0.7]$ with quality $S = 1.08$).

However, note that for the geometric MR graphs, an analysis of the order parameter following a scaling assumption of the form of Eq. (1) did not lead to any conclusive results; i.e., the data curves did not give a satisfactory data collapse. Nevertheless, based on the analysis of the fluctuations of the order parameter, we were able to obtain estimates for the critical point (see below). In summary, as is obvious from Table I, degree-based attacks are more severe than random removals. Again, the resilience against attacks correlates well with the average degree.

TABLE I. Critical points p_c, i.e., fractions of removed nodes which indicate when the underlying network decomposes into "small" clusters for the different graph types and node-removal strategies, discussed in Secs. II and III (random failure, equivalent to random percolation; attack 1, degree-based node-removal strategy; attack 2, centrality-based node-removal strategy), respectively.

Strategy	RNG	GG	DT	MR
Random failure	0.205(1)	0.365(1)	0.500(2)	0.71(1)
Attack 1	0.120(1)	0.263(1)	0.377(1)	0.68(2)
Attack 2	0	0	0	0

Considering the *centrality-based attack strategy* for the RNGs, we start out with a more simplified initial analysis. As is evident from Fig. 4(a), the data curves that describe the scaling of the order parameter for this setup drop to 0 at rather small values of p. Thus, an FSS analysis (as carried out above) comes with several difficulties (related to the accessibility of data points in the critical scaling window). Hence, we first determine the area $A(N)$ under the order-parameter curves and assess its scaling behavior with increasing system size N to see whether it converges to a finite value at all. From a fit to the function $A(N) = a(N + \Delta N)^{-b}$ we find $a = 0.20(1)$, $\Delta N = 30(4)$, and $b = 0.315(3)$ [reduced chi-square $\chi^2_{\rm red} = 0.34$; see inset in Fig. 4(b)], indicating that indeed $A(N \to \infty) \to 0$. If we neglect the smallest system, we find that a pure power law $A(N) = 0.270(3)N^{-0.302(2)}$ fits the data well (reduced chi-square $\chi^2_{\rm red} = 0.65$). From this we conclude that for RNGs subject to a centrality-based attack strategy one has $p_c = 0$. Following this procedure, we also found that under this attack strategy $p_c = 0$ holds true for GGs, DTs, and MR graphs. Thus, due to its propensity to fragment graphs at negligible values of p, this strategy is much more efficient than the degree-based strategy, independent of the type of graph.

(b) Average size of finite clusters. As the second observable we consider the average size $\langle S_{\rm fin}(p)\rangle$ of all finite clusters for a particular graph instance, averaged over different graph instances. The definition of this observable reads [41]

$$S_{\rm fin}(p) = \frac{\sum_s' s^2\, n_s(p)}{\sum_s' s\, n_s(p)}, \tag{3}$$

where $n_s(p)$ signifies the probability mass function of cluster sizes for a single graph instance at a given value of p. The prime indicates that the sums run over all clusters excluding the largest cluster for each graph instance. The

average size of all finite clusters is expected to scale according to Eq. (1), where $b = -\gamma/\nu$. Therein, for 2D percolation, the critical exponent γ assumes a value of $\gamma = 43/18 \approx 2.389$.

Again, a detailed analysis of this observable for random percolation, which is equivalent to the random failure f strategy, can be found in Ref. [26] regarding RNGs and in Ref. [45] with respect to GGs.

Regarding the degree-based attack strategy for RNGs, considering systems of size $N = 1024$, 2304, 4096, and 9216 and restricting the data analysis to the interval $\epsilon \in [-0.5,0.5]$ on the rescaled p axis, the optimal scaling parameters are found to be $p_c = 0.120(1)$, $\nu = 1.46(3)$, and $\gamma = 2.35(4)$ with a collapse quality $S = 0.91$ [see Fig. 4(c)]. Note that here the estimated value of ν appears to overestimate the expected value somewhat. Apart from that, the numerical values of the extracted exponents are in reasonable agreement with their expected values and the estimate of the critical threshold p_c is consistent with the numerical value found from an analysis of the order parameter.

In addition to the full FSS analysis, we also performed a scaling analysis for the effective critical points $p_{\mathrm{peak}}(N)$ at which the curves of S_{fin} assume their maximum. Therefore, polynomials of fifth order were fitted to the data curves at different system sizes N in order to obtain an estimate $p_{\mathrm{peak},i}(N)$ of the peak position. Thereby, the index i labels independent estimates of the peak position as obtained by bootstrap resampling. For the analysis, we considered 20 bootstrap data sets, e.g., resulting in the estimate $p_{\mathrm{peak}}(N = 9216) = 0.1057(4)$ for the RNG regarding the degree-based attack strategy. Considering systems of size $N > 500$ and assuming the scaling form

$$p_{\mathrm{peak}}(N) = p_{c,\mathrm{peak}} - aN^{-b} \tag{4}$$

yields the fit parameters $p_{c,\mathrm{peak}} = 0.111(1)$, $b = 0.50(3)$, and $a = O(1)$ for a reduced chi-square $\chi^2_{\mathrm{red}} = 0.08$ [see inset in Fig. 4(c)]. This result indicates that the peak seems to be positioned off criticality at a value slightly below p_c [cf. Fig. 4(c)]. However, including also very small systems we yield $p_{c,\mathrm{peak}} = 0.119(5)$, $b = 0.29(6)$, and $a = O(1)$ for a reduced chi-square $\chi^2_{\mathrm{red}} = 3.89$, in good agreement with the value of p_c obtained from an analysis of the order parameter. Following this procedure by considering RNGs subject to a random node failure we yield $p_c = 0.196(8)$, which compares well to the estimate obtained from an analysis of the order parameter (see Table I). An analysis of the peak positions for all other types of proximity graphs led to qualitatively similar results. Hence, we do not elaborate on them here.

Whenever we analyzed the order parameter, we also analyzed the respective fluctuations, giving rise to the finite-size susceptibility $\chi(p)$

$$\chi(p) = N\left[\left\langle s^2_{\mathrm{max}}(p)\right\rangle - \left\langle s_{\mathrm{max}}(p)\right\rangle^2\right]. \tag{5}$$

These curves also feature a pronounced peak and exhibit the same scaling behavior as the average size of the finite clusters discussed above. Here, we also performed a scaling analysis of the peak positions of the $\chi(p)$ curves, similar to that performed for the peaks of the previous observable. Although this did not lead to new insight into the various types of proximity graphs, it was a valuable method to estimate critical points

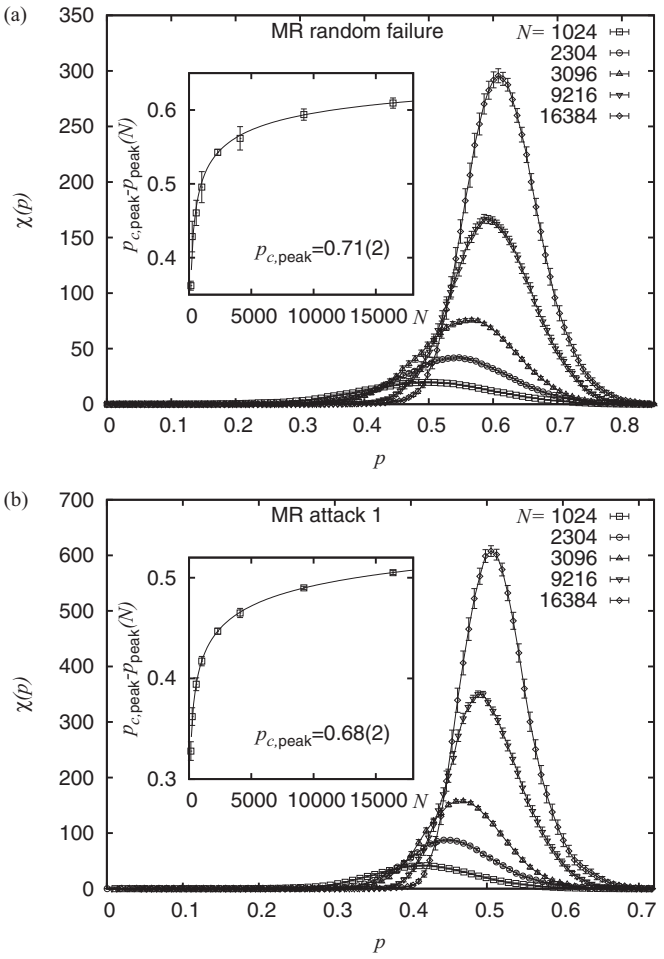

FIG. 5. Finite-size scaling analysis for the peak position of the finite-size susceptibility curves for the MR geometric graphs. (a) Data curves for the random node-removal strategy. Inset: Finite-size scaling of the respective peak positions (see text for details). (b) Data curves for the degree-based node-removal strategy: Finite-size scaling of the respective peak positions (see text for details).

for the MR geometric graphs. In this regard, for MR graphs subject to random node removal we find $p_c = 0.71(2)$ [see Fig. 5(a)]. Further, for MR graphs subject to the degree-based node-removal strategy we obtain $p_c = 0.68(2)$ [see Fig. 5(b)]. Hence, for the MR graphs we cannot rule out that the estimates for both critical points actually agree within error bars. This might be attributed to the rather high degree of the individual nodes and, from a statistical point of view, the extensive overlap of the individual node neighborhoods within the range of the underlying "connectivity radius." Hence, due to the high number of redundant node-to-node paths which easily allow compensation for deleted nodes, the effect caused by the removal of a randomly chosen node does not differ much from the effect caused by the removal of a node with a particularly high degree.

2. *Analysis of the hop diameter*

The last observable, studied in the context of the fragmentation analysis, is related to the diameter $R(p)$ of the graphs as a function of the fraction p of removed nodes. Here, the diameter

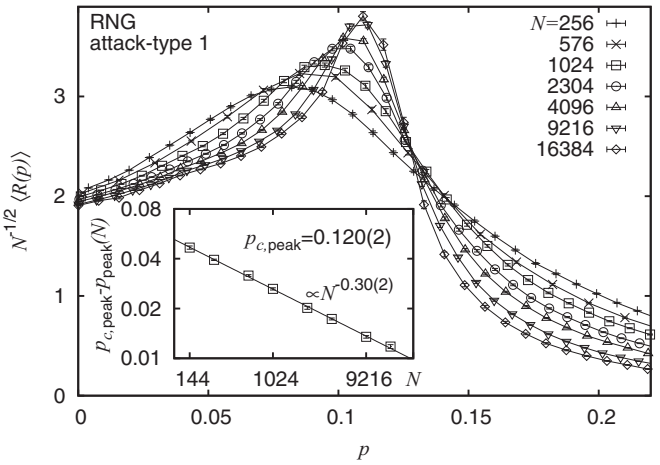

FIG. 6. Finite-size scaling analysis for the peak position of the diameter curves for the RNG proximity graphs subject to a degree-based attack strategy. The diameter of the graphs as a function of the removed fraction of nodes following the attack strategy "attack 1" (discussed in Sec. III). Inset: Scaling behavior of the effective peak position $p_{\mathrm{peak}}(N)$ as a function of the system size N.

of a graph indicates the longest among all finite shortest paths. In Fig. 6, the data curves of the diameter for the particular choice of RNG proximity graphs subject to a degree-based node-removal strategy are shown. For the particular case of nonfragmented RNGs, i.e., at $p = 0$, the diameter (averaged over different realizations of point sets) was previously found to scale as $\langle R\rangle \propto N^{1/2}$ [26]. In view of these prior results, the data curves in Fig. 6 are scaled so as to assume a fixed value at $p = 0$. As is evident from the figure the data curves assume a peak value when a certain fraction of nodes is removed. This appears to be quite intuitive: if nodes are removed from one of the graph instances introduced in Sec. II, redundant edges will disappear (on average), resulting in an increasing node-to-node distance. As soon as the value of p exceeds the percolation threshold of the respective setup (i.e., graph type and node-removal strategy), the graph instance decomposes into several "small" clusters accompanied by a decreasing node-to-node distance. With increasing system size N, the position $p_{\mathrm{peak}}(N)$ of the peak shifts towards larger values of p. For RNGs subject to degree-based node removal, a fit function of the form similar to Eq. (4) yields the fit parameters $p_c = 0.120(2)$, $a = O(10^{-1})$, and $b = 0.30(2)$ ($\chi^2_{\mathrm{red}} = 0.90$). Similarly, for DTs we find $p_c = 0.378(4)$, $a = O(10^{-2})$, and $b = 0.3(2)$ ($\chi^2_{\mathrm{red}} = 0.14$). The resulting asymptotic peak positions are in good agreement with the value obtained from an FSS analysis of the order parameter (cf. Table I). For the case of a random node failure, the results obtained from the scaling of the peak position fit the results from the order parameter analysis similarly well; e.g., for the case of RNGs we find $p_c = 0.199(2)$ (cf. Table I). Although we performed similar analyses for GGs and DTs, resulting in qualitatively similar results, we do not elaborate on them here. No analyses were performed for the centrality-based node-removal strategy.

C. $N-1$ resilience

The actual most important application example regarding proximity graphs is wireless *ad hoc* networks. Nevertheless, there might be some other fields of application for them. Proximity graphs ensure connectivity and the total length of all involved edges is small in comparison to many other networks that feature this quality. For applications where edges are expensive and connectivity is crucial, the topology of proximity graphs might be a good candidate to install. To this point, we have assumed that the capacities of the edges and nodes are infinitely large, so the network components do not overload, regardless of how intensively they get strained. In real scenarios, if some nodes or edges malfunction, network components which are hardly used under normal circumstances might become essential at once. Consequently, since the hardly used components are not designed to handle such a burden, this might trigger a cascading breakdown of the whole network [46–49]. Therefore, it is reasonable to equip all network components with sufficient capacity.

To ensure that the network operates in an orderly fashion under all circumstances when one component drops out, referred to as "$N-1$ resilience" (or $N-1$ stability or the $N-1$ criterion), it is necessary to know the most adverse scenario that can occur. With a transport model in mind, where some quantities have to be transported between all pairs of nodes [46,48,49], the betweenness centrality [50] is a measure of the capacity each node or edge has to provide in a well-functioning situation. When one node or edge fails, given that the network is still connected, the loads have to be redistributed, visible from a recalculation of the betweenness centrality. In some nodes or edges the centrality will increase [51,52], corresponding to the higher capacity these nodes or edges have to provide *a priori*. The value of the highest increment, which is called the backup capacity [52] Δb_{node} or Δb_{edge}, provides an estimate of the additional costs of each node or edge that must be invested to protect the network against cascading failures upon such an incident.

The betweenness centrality has been calculated based on Dijkstra's algorithm [36], i.e., the edge lengths have been taken into account in calculating the shortest path. The resulting probability mass functions of Δb_{node} and Δb_{edge} are illustrated in Figs. 7 and 8, respectively, for different network ensembles.

For the case of Δb_{edge}, but using a breadth-first search, i.e., without taking the actual edge lengths into account, the probability mass functions look qualitatively almost identical to the case of Δb_{node}, without notable distinctions. From the statistics mediated by the figures it is evident that the typical backup capacity of DT networks is the lowest, while that of RNG networks is the highest. This means that the structure of the network of RNG networks is more vulnerable, such that one has to invest more in the capacity of the edges in order to ensure $N-1$ resilience. This is not surprising, since the RNG is a subgraph of the GG and DT and includes fewer edges. On the other hand, due to the lack of the additional edges of the RNG in comparison to the others, the investment to provide the backup capacity must be applied to fewer edges. Thus, it makes sense to ask for the total backup capacity either per edge, if the investment costs are dominated by the number of connections, or per unit length, if the investments are dominated by the length of the edges. It becomes evident from Fig. 9(a) that for the former case, the typical total investment ($M \cdot \Delta b$) is, relatively speaking, still the same for all three ensembles. For the second case, i.e., also taking edge lengths into account

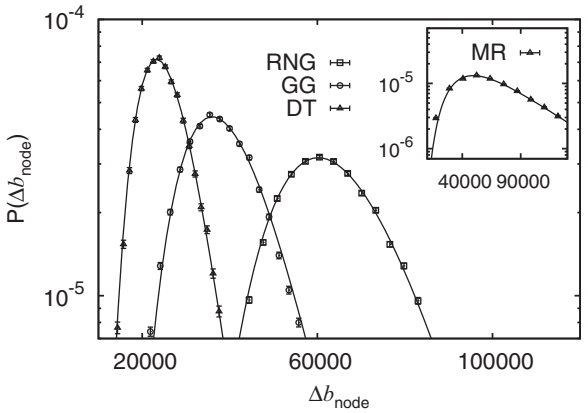

FIG. 7. Probability mass function of the backup capacity, which is needed to ensure an $N-1$ resilience (see text) Δb_{node} for different network topologies (initial system size: $N=1024$). The probability mass function concerning this measure was determined by analyzing 40 000 realizations of the disorder. The data were fitted by a log-normal distribution of moderate quality (reduced χ^2_{red} between 0.65 and 4.46).

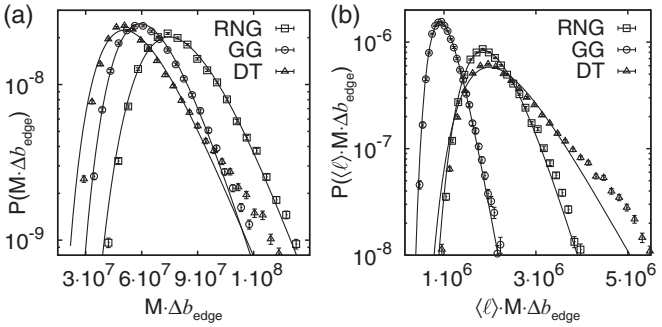

FIG. 9. Probability mass functions of (a) $M\cdot\Delta b_{edge}$ and (b) $\langle\ell\rangle\cdot M\cdot\Delta b_{edge}$ for different network topologies at $N=1024$. M describes the number of edges in the respective graph and $\langle\ell\rangle$ denotes their respective edge-length mean value. The data were fitted by a log-normal distribution of predominantly low quality: (a) reduced $\chi^2_{red}=6.5$ for GG data, reduced $\chi^2_{red}=4.99$ for RNG, data and reduced $\chi^2_{red}=12.75$ for DT; (b) reduced $\chi^2_{red}=2.9$ for GG data, reduced $\chi^2_{red}=5.34$ for RNG data, and reduced $\chi^2_{red}=20.09$ for DT.

[Fig. 9(b)], it turns out that the investment of the DT is at about the same level as that of the RNG. The GG appears to be the most cost-efficient graph under this scenario.

D. Networks of the same total length

To compensate for the simple resilience effect created by simply exhibiting more edges, we also compared the different topologies of the proximity graphs featuring the same total edge length ℓ_{tot}. Therefore, we measured the scaling behavior of this quantity for the different proximity graph types (see Fig. 10). The figure provides the number of nodes which have to be added to the RNG and GG in order to get same total edge length as the respective DT. For example, it is evident from the figure that a DT with $N=718$ nodes has the same total edge length $\ell_{tot}=100$ on average as a GG with $N=2625$ and an RNG with $N=9783$ nodes. Since the fragmentation thresholds for the different node-removal strategies are known (Table I), it can be calculated easily ($N\cdot p_c$) for each topology how many nodes must be removed until the respective network decomposes into small clusters; for example, if $\ell_{tot}=100$, the RNG will tolerate 2005 randomly removed nodes. In contrast, the GG tolerates 958 and the DT tolerates merely 359 nodes that fail randomly. As a consequence, implementing the topology of the RNG will be the most reasonable, if installing edges is much more expensive than adding further nodes. Certainly, the additional edges of the DT and GG increase

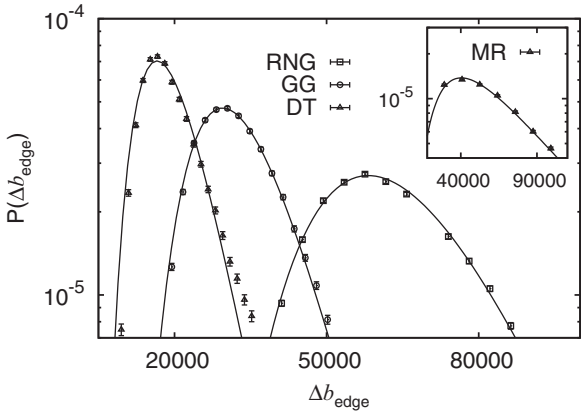

FIG. 8. Probability mass function of Δb_{edge} for different network topologies (initial system size: $N=1024$). Removing the edge featuring the largest betweenness centrality value, the largest increment of the betweenness centrality of the other edges Δb_{edge} was monitored. The probability mass function concerning this measure was determined by analyzing 40 000 realizations of the disorder. The data were fitted by a log-normal distribution of good quality for GG (reduced $\chi^2_{red}=0.26$) and RNG (reduced $\chi^2_{red}=2.71$ but poor quality for MR (reduced $\chi^2_{red}=5.2$) and DT (reduced $\chi^2_{red}=12.2$) graphs.

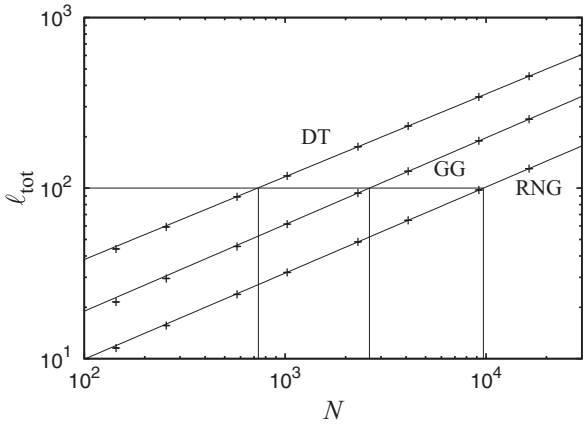

FIG. 10. Scaling behavior of the total edge length ℓ_{tot} of the RNG, GG, and DT. Each data point was created by averaging over 2000 instances. In each case the total edge length seems to scale according to $\ell_{tot}\sim N^{0.5}$ for large systems. We used the fit function $\ell_{tot}=a(N+b)^c$. By taking the system sizes $N=2304$, 4096, 9216, and 16 384 into account, we found $a=4.02(2)$, $b=19(5)$, $c=0.4868(4)$ for the DT ($\chi^2_{red}=2.44$), $a=1.91(1)$, $b=-47(5)$, $c=0.5039(5)$ for the GG ($\chi^2_{red}=1.31$), and $a=0.99(1)$, $b=-20(3)$, $c=0.5024(3)$ for the RNG ($\chi^2_{red}=0.46$).

the stability, but the benefit of these is small in comparison to that of the edges which are contained in the RNG anyway.

V. CONCLUSION

In this article, the robustness of three types of proximity graphs and a particular geometric random graph (see Sec. II), i.e., their ability to function well even if they are subject to random failures and targeted attacks, was put under scrutiny. For this purpose we generated instances of the considered graph types and successively removed nodes according to three node-removal strategies (see Sec. III). Once the fraction of removed nodes exceeds a certain threshold (characteristic of the particular graph type and node deletion scheme), the underlying graph instance decomposes into many small clusters. Using standard observables from percolation theory (see Sec. IV B), the critical node-removal thresholds were determined for the different graph types and deletion strategies (see Table I). Therein, so as to yield maximally justifiable results through numerical redundancy, we considered various observables to estimate the critical points and exponents. In order of increasing severity, these strategies have an intuitive order: a random node-removal mechanism, equivalent to ordinary random percolation, is less severe than a degree-based node-removal strategy which takes into account particular node-related local details (i.e., the node degrees) to optimize the order of node removals during the fragmentation procedure. As is evident from Table I, both removal schemes result in finite critical points. The latter strategy is again less severe than the centrality-based node-removal mechanism, which takes into account global information (i.e., the set of shortest paths that connect all pairs of nodes) that is used to impose a maximally efficient structural damage by preferentially removing nodes with maximal betweenness centrality (i.e., the most relevant nodes). As is evident from Table I and the discussion in Sec. IV B, the latter node-removal scheme requires the deletion of only a negligible number of nodes until the graph decomposes into small clusters. A peculiar result is the fragmentation thresholds related to the random failure and degree-based node removal for the MR geometric graph. As discussed in Sec. IV B we cannot rule out that the estimates for both critical points agree within the error bars. This might be attributed to the extensive overlap of the individual node neighborhoods within the range of the underlying "connectivity radius." Hence, due to the high number of redundant node-to-node paths which easily allow compensation for deleted nodes, the effect caused by the removal of a randomly chosen node does not differ much from the effect caused by the removal of a node of a particularly high degree.

For a given node-removal strategy, the sequence of critical points for the subgraph hierarchy RNG $\subset$ GG $\subset$ DT follows the commonly accepted belief that the percolation threshold (or, here, the fragmentation threshold) is a nondecreasing function of the average degree. This is in full accord with the containment principle due to Fisher [53], stating that if G' is a subgraph of G, then it holds that $p_c^{G'} \leqslant p_c^G$ for both bond and site percolation.

Finally, we considered the backup capacity, which is the largest betweenness-centrality increment of the nodes (or edges) after removing the most important node (or edge) beforehand, for the different graph types. Thus, via sufficient backup a graph is made $N-1$ resilient. Regarding the three studied proximity graph ensembles, it turned out that the DT is the most cost-efficient assuming that the backup investments are dominated by improvement of the nodes. On the other hand, if one has to back up the edges, the more cost-efficient one will be either the DT or the GG, depending on whether the investment depends mainly on the number or on the length of the edges.

For further studies, it would be very interesting to evaluate these simple spatial planar ensembles in the context of more complex transportation networks, as for steady-state power grids in the power-flow approximation [54], or for networks of truly dynamically coupled oscillators, as for Kurmatoto-like models [55–57].

ACKNOWLEDGMENTS

We thank Charlotte Beelen and Pascal Fieth for critically reading the manuscript. C.N. gratefully acknowledges financial support from the Lower Saxony research network Smart Nord, which acknowledges the support of the Lower Saxony Ministry of Science and Culture through the Niedersächsisches Vorab grant program (Grant No. ZN 2764/ZN 2896). O.M. gratefully acknowledges financial support from the Deutsche Forschungsgemeinschaft (DFG) under Grant No. HA3169/3-1. Simulations were performed at the HPC Cluster HERO, located at the University of Oldenburg (Germany), and funded by the DFG through its Major Instrumentation Programme (INST 184/108-1 FUGG) and the Ministry of Science and Culture (MWK) of Lower Saxony State.

[1] R. Albert, I. Albert, and G. L. Nakarado, Phys. Rev. E **69**, 025103 (2004).
[2] B. Jiang and C. Claramunt, GeoInformatica **8**, 157 (2004).
[3] J. H. Choi, G. A. Barnett, and B.-S. Chon, Global Networks **6**, 81 (2006).
[4] A.-L. Barabási, R. Albert, and H. Jeong, Physica A **281**, 69 (2000).
[5] R. Albert, H. Jeong, and A.-L. Barabasi, Nature **406**, 378 (2000).
[6] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts, Phys. Rev. Lett. **85**, 5468 (2000).
[7] P. Crucitti, V. Latora, M. Marchiori, and A. Rapisarda, Physica A **340**, 388 (2004).
[8] L. K. Gallos, R. Cohen, P. Argyrakis, A. Bunde, and S. Havlin, Phys. Rev. Lett. **94**, 188701 (2005).
[9] P. Holme, B. J. Kim, C. N. Yoon, and S. K. Han, Phys. Rev. E **65**, 056109 (2002).
[10] R. Cohen, K. Erez, D. ben-Avraham, and S. Havlin, Phys. Rev. Lett. **86**, 3682 (2001).
[11] X. Huang, J. Gao, S. V. Buldyrev, S. Havlin, and H. E. Stanley, Phys. Rev. E **83**, 065101 (2011).

[12] M. Kurant, P. Thiran, and P. Hagmann, Phys. Rev. E **76**, 026103 (2007).
[13] G. Paul, T. Tanizawa, S. Havlin, and H. E. Stanley, Eur. Phys. J. B **38**, 187 (2004).
[14] B. Shargel, H. Sayama, I. R. Epstein, and Y. Bar-Yam, Phys. Rev. Lett. **90**, 068701 (2003).
[15] T. Tanizawa, S. Havlin, and H. E. Stanley, Phys. Rev. E **85**, 046109 (2012).
[16] U. Brandes, Soc. Networks **30**, 136 (2008).
[17] L. Page, S. Brin, R. Motwani, and T. Winograd, Technical Report (Stanford InfoLab, 1999).
[18] G. T. Toussaint, Pattern Recognit. **12**, 261 (1980).
[19] K. R. Gabriel and R. R. Sokal, Syst. Biol. **18**, 259 (1969).
[20] R. Sibson, Comput. J. **21**, 243 (1978).
[21] J. W. Essam and M. E. Fisher, Rev. Mod. Phys. **42**, 272 (1970).
[22] Z. Toroczkai and H. Guclu, Physica A **378**, 68 (2007).
[23] E. Bertin, J.-M. Billiot, and R. Drouilhet, Adv. Appl. Prob. **34**, 689 (2002).
[24] A. M. Becker and R. M. Ziff, Phys. Rev. E **80**, 041101 (2009).
[25] J.-M. Billiot, F. Corset, and É. Fontenas, arXiv:1004.5292.
[26] O. Melchert, Phys. Rev. E **87**, 042106 (2013).
[27] E. Jennings and C. M. Okino, in *International Symposium on Performance Evaluation of Computer and Telecommunications Systems*, San Diego, CA (Jet Propulsion Laboratory, California Inst. of Tech.; Pasadena, CA, 2002).
[28] P. Santi, ACM Comput. Surv. **37**, 164 (2005).
[29] X.-Y. Li, W.-Z. Song, and Y. Wang, Wireless Networks **11**, 255 (2005).
[30] R. Rajaraman, ACM SIGACT News **33**, 60 (2002).
[31] B. Karp and H. T. Kung, in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking* (Association for Computing Machinery, New York, 2000), p. 243.
[32] P. Bose, P. Morin, I. Stojmenović, and J. Urrutia, Wireless Networks **7**, 609 (2001).
[33] C.-W. Yi, P.-J. Wan, L. Wang, and C.-M. Su, IEEE Trans. Wireless Commun. **9**, 614 (2010).
[34] F. Kuhn, R. Wattenhofer, and A. Zollinger, in *Proceedings of the 2003 Joint Workshop on Foundations of Mobile Computing* (Association for Computing Machinery, New York, 2003), p. 69.
[35] J. W. Jaromczyk and G. T. Toussaint, Proc. IEEE **80**, 1502 (1992).
[36] T. H. Cormen, C. Stein, R. L. Rivest, and C. E. Leiserson, *Introduction to Algorithms* (MIT Press, Cambridge, MA, 2001).
[37] C. B. Barber, Qhull computes the convex hull, Delaunay triangulation, Voronoi diagram, halfspace intersection about a point, furthest-site Delaunay triangulation, and furthest-site Voronoi diagram, http://www.qhull.org (1995).
[38] F. P. Preparata and M. I. Shamos, *Computational Geometry: An Introduction* (Springer-Verlag, Berlin, 1985).
[39] D. G. Kirkpatrick and J. D. Radke, in *Computational Geometry*, edited by G. Toussaint (Elsevier North-Holland, New York, 1985), p. 217.
[40] D. Stauffer, Phys. Rep. **54**, 1 (1979).
[41] D. Stauffer and A. Aharony, *Introduction to Percolation Theory* (Taylor & Francis, London, 1992).
[42] K. Binder and D. W. Heermann, *Monte Carlo Simulation in Statistical Physics: An Introduction* (Springer Science & Business Media, New York, 2010).
[43] J. Houdayer and A. K. Hartmann, Phys. Rev. B **70**, 014418 (2004).
[44] O. Melchert, arXiv:0910.5403v1.
[45] C. Norrenbrock, Eur. Phys. J. B **89**, 111 (2016).
[46] Z. J. Bao, Y. J. Cao, G. Z. Wang, and L. J. Ding, Phys. Lett. A **373**, 3032 (2009).
[47] P. Crucitti, V. Latora, and M. Marchiori, Phys. Rev. E **69**, 045104 (2004).
[48] A. E. Motter, Phys. Rev. Lett. **93**, 098701 (2004).
[49] A. E. Motter and Y.-C. Lai, Phys. Rev. E **66**, 065102 (2002).
[50] M. E. J. Newman, Phys. Rev. E **64**, 016132 (2001).
[51] B. Ouyang, X. Jin, Y. Xia, and L. Jiang, Eur. Phys. J. B **87**, 52 (2014).
[52] A. K. Hartmann, Eur. Phys. J. B **87**, 114 (2014).
[53] M. E. Fisher, J. Math. Phys. **2**, 620 (1961).
[54] P. Kundur, N. J. Balu, and M. G. Lauby, *Power System Stability and Control* (McGraw–Hill Education, New York, 1994).
[55] G. Filatrella, A. H. Nielsen, and N. F. Pedersen, Eur. Phys. J. B **61**, 485 (2008).
[56] M. Rohden, A. Sorge, M. Timme, and D. Witthaut, Phys. Rev. Lett. **109**, 064101 (2012).
[57] D. Witthaut and M. Timme, New J. Phys. **14**, 083036 (2012).