

Supplemental Material “Shortest-path percolation on random networks”

Minsuk Kim¹ and Filippo Radicchi^{1,*}

¹*Center for Complex Networks and Systems Research,
Luddy School of Informatics, Computing, and Engineering,
Indiana University, Bloomington, Indiana, 47408, USA*

I. THE SHORTEST-PATH-PERCOLATION MODEL ON ERDŐS-RÉNYI GRAPHS

All the results of the present paper are based on the application of the shortest-path-percolation (SPP) model to Erdős-Rényi (ER) graphs. Graphs \mathcal{G}_1 used as inputs of the SPP algorithm are generated by drawing exactly $E_1 = N\bar{k}/2$ edges between pairs of randomly chosen nodes. Here N is the size of the graph, and \bar{k} indicates the average degree of the graph. We use $\bar{k} = 4$ in all our simulations, although the choice affects only the location of the critical points and not the other critical properties of the SPP transition. In our simulations, pairs of origin-destination nodes are also extracted at random from all possible nodes in the graph. We extract random pairs of origin-destination nodes until all edges in the graph are removed.

The output of each realization of the SPP model is an ordered list containing all edges of the input graph \mathcal{G}_1 . These edges appear in the order in which they are removed according to the specific realization of the SPP model. Each edge has also associated its corresponding t value denoting the index of the origin-destination pair requested during the application of the SPP model. We use such an output list in all our subsequent analyses. To efficiently keep track of the cluster structure of the network during the SPP process, we take advantage of the Newman-Ziff algorithm [1].

Results are obtained on a large number of numerical simulations for each combination of the parameter C of the SPP model and size of the network N , see Table S1 for details.

N	$C = 1$	$C = 2$	$C = 3$	$C = N$
$2^{10} - 2^{17}$	5,000	5,000	5,000	5,000
$2^{18} - 2^{20}$	1,000	1,000	1,000	1,000
$2^{21} - 2^{25}$	1,000	1,000	1,000	-
2^{26}	200	200	200	-
2^{27}	20	20	20	-

TABLE S1. Numerical simulation of the shortest-path-percolation (SPP) model. For each size N of the network and value of the cost C , we report the corresponding number of numerical simulations that we performed.

II. CONTROL AND ORDER PARAMETERS

The natural control parameter of the shortest-path-percolation (SPP) model is t , i.e., the number of demanded pairs of origin-destination nodes. Another control parameter is the fraction of removed edges, namely p . This control parameter allows for immediate comparisons between the SPP model and other standard percolation models. The fraction of removed edges is associated to t as

$$p(t) = 1 - \frac{E_t}{E_1}, \quad (\text{S1})$$

where $E_t = |\mathcal{E}_t|$ is the size of the set of edges associated to the graph \mathcal{G}_t . Specifically, $\mathcal{G}_t = (\mathcal{V}, \mathcal{E}_t)$ is the graph available when the t -th pair of origin-destination nodes is demanded, composed of $N = |\mathcal{V}|$ nodes and $E_t = |\mathcal{E}_t|$ edges.

We consider two standard metrics in network percolation: the percolation strength P and the average cluster size S . These metrics serve to characterize the cluster structure of a network under the SPP model.

* fliradi@indiana.edu

Indicate with $s_1 \leq s_2 \leq \dots \leq s_{R-1} \leq s_R$ the size of the R connected components or clusters observed in the network at a certain stage of the SPP model. We clearly have that $\sum_{r=1}^R s_r = N$, i.e., the sum of all clusters sizes equals the network size N .

P is defined as the size of the largest cluster divided by the network size, thus

$$P = \frac{s_R}{\sum_{r=1}^R s_r}, \quad (\text{S2})$$

S is instead defined as

$$S = \frac{\sum_{r=1}^{R-1} s_r^2}{\sum_{r=1}^{R-1} s_r}, \quad (\text{S3})$$

where both the above sums runs over all clusters except the largest one.

III. POWER-LAW FIT

In our analysis, we systematically rely on power-law fits of data. We start from the hypothesis that variables x and y are related by

$$x = z y^\zeta. \quad (\text{S4})$$

To determine the best values of z and ζ , we first take the logarithms of both sides of Eq. (S4) to obtain

$$\log(x) = \log(z) + \zeta \log(y), \quad (\text{S5})$$

and then apply the simple linear regression algorithm. Such an algorithm provides us with the best estimates ($\log \hat{z}$ and $\hat{\zeta}$, their associated errors $d(\log z)$ and $d\zeta$ (i.e., standard errors under the assumption of residual normality), as well as the metric of goodness of the fit r (i.e., Pearson correlation coefficient). Our estimates of the exponent are in the format $\zeta = \hat{\zeta} \pm d\zeta$. For consistency with previous work, we report numerical values of these quantities with a precision of two decimals [2]. We compare r values to determine if one power-law scaling is better than another in some of our analyses.

For instance, in Fig. S1 (a), we plot the r^2 , i.e., the coefficient of determination of simple linear regression, as a function of p while estimating p_c of the SPP model with $C = 1$ by using the power-law fit. The yellow diamond symbol denotes the point where r^2 is maximized, $p_c = 0.75$. In Fig. S1 (b), we plot P as a function of N at $p = p_c$, $p = p_c - \Delta$, and $p = p_c + \Delta$, respectively. Here, we use $p_c = 0.75$ and $\Delta = 0.005$. We can clearly observe that $p = p_c$ provides the best power-law fit.

IV. FINITE-SIZE SCALING ANALYSIS

A. Conventional ensemble

We denote with $P(p, N)$ and $S(p, N)$ the values of the metrics of Eqs. (S2) and (S3), respectively, when exactly a fraction p of edges is removed from the graph. To quantify the critical properties of the SPP transition, we rely on finite-size scaling (FSS) analysis [3] based on the scaling ansatz

$$\langle P(p, N) \rangle = N^{-\beta/\bar{\nu}} F^{(1)}[(p - p_c)N^{1/\bar{\nu}}] \quad (\text{S6})$$

and

$$\langle S(p, N) \rangle = N^{\gamma/\bar{\nu}} F^{(2)}[(p - p_c)N^{1/\bar{\nu}}], \quad (\text{S7})$$

where $\langle P(p, N) \rangle$ and $\langle S(p, N) \rangle$ are the ensemble averages of $P(p, N)$ and $S(p, N)$, respectively, $F^{(1)}(\cdot)$ and $F^{(2)}(\cdot)$ are universal scaling functions, while β , γ , and $\bar{\nu}$ are critical exponents for the percolation strength, the average cluster size, and the correlation volume, respectively. In Eqs. (S6) and (S7), p_c is the critical value of the control parameter where the transition occurs in a network of infinite size. In a network of finite size N , the pseudo-critical threshold

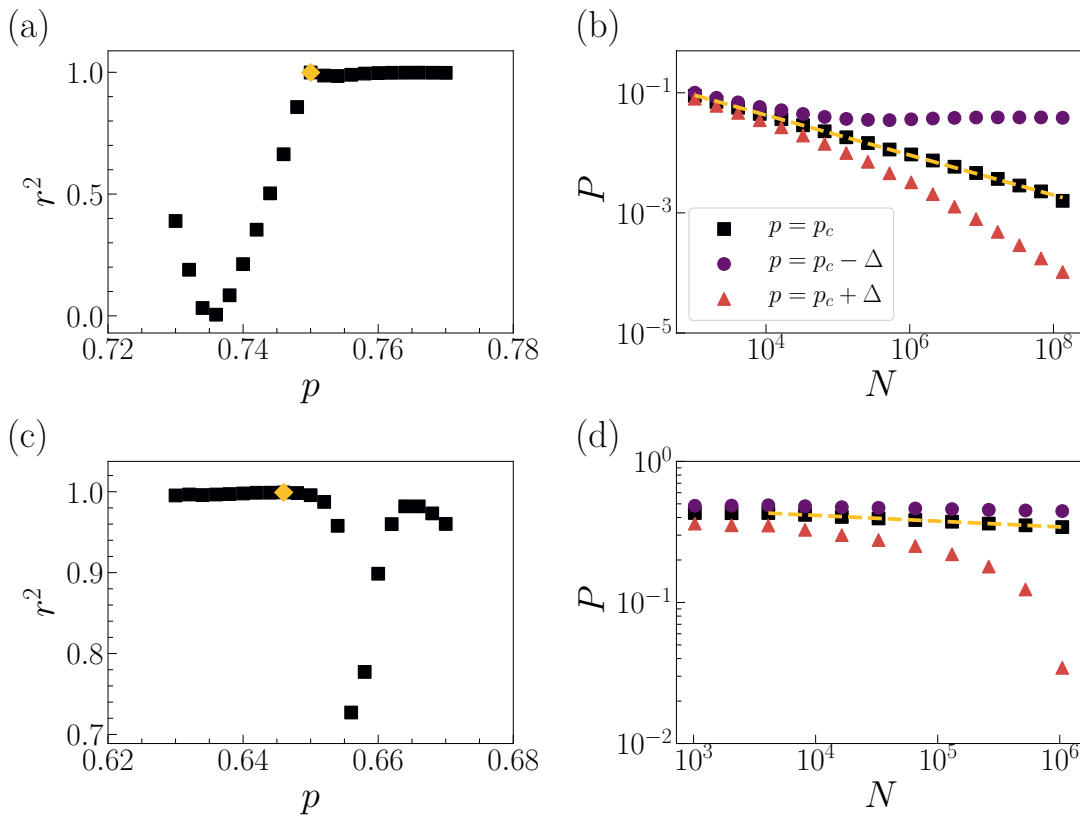


FIG. S1. Estimation of the critical threshold in the conventional ensemble. (a) Coefficient of determination r^2 as a function of p . Here $C = 1$. (b) Plot of P as a function of N at $p = p_c$, $p = p_c - \Delta$, and $p = p_c + \Delta$, respectively. We use $p_c = 0.75$ and $\Delta = 0.005$. Here also $C = 1$. (c) Similar to (a) but for $C = \infty$. (d) Similar to (b) but for $C = \infty$. We use $p_c = 0.646$ and $\Delta = 0.01$. Detailed results can be found in Table S2.

$p_c(N)$ indicates the value of the control parameter related to the biggest variation in the cluster structure of the network. There are multiple ways of defining this condition. We use

$$p_c(N) = \arg \max_p \langle S(p, N) \rangle . \quad (\text{S8})$$

Still according to FSS, critical and pseudo-critical points are related by

$$p_c = p_c(N) + bN^{-1/\bar{\nu}}, \quad (\text{S9})$$

where b is a constant.

B. Critical properties of the shortest-path-percolation in the conventional ensemble

Note that we use a simplified notation to denote $\langle P(p, N) \rangle$ and $\langle S(p, N) \rangle$ in the figures, and in the text. Details on the notation are provided below.

We determine the value of the critical threshold p_c using Eq. (S6). For $p = p_c$, the FSS ansatz predicts

$$P(p_c) = \langle P(p = p_c, N) \rangle \sim N^{-\beta/\bar{\nu}}, \quad (\text{S10})$$

with $P(p_c)$ highlighting the simplified notation adopted in the corresponding figure. Thus, we obtain the best estimator of the critical threshold as the value of the control parameter p leading to the best power-law fit, Fig. S2. The best estimate of the $\beta/\bar{\nu}$ is obtained from the very same power-law fit. The ansatz of Eq. (S7) together with the estimated p_c value further allows us to obtain the best estimate of $\gamma/\bar{\nu}$, i.e., $S(p_c) = \langle S(p = p_c, N) \rangle \sim N^{\gamma/\bar{\nu}}$, see Fig. S2(c).

For finite C , these scalings provide good estimates of the critical point and of the critical exponents. For infinite C , the estimates of the critical exponents are instead affected by strong finite size effects while the estimate of the critical point remains unaffected.

Also, we estimate $\beta/\bar{\nu}$ and $\gamma/\bar{\nu}$ using the scaling

$$P_c(N) = \langle P(p = p_c(N), N) \rangle \sim N^{-\beta/\bar{\nu}} , \quad (\text{S11})$$

$$S_c(N) = \langle S(p = p_c(N), N) \rangle \sim N^{\gamma/\bar{\nu}} , \quad (\text{S12})$$

as predicted by Eqs. (S6), (S7) and (S9). $p_c(N)$ is defined in Eq. (S8). See Fig. S2(d) and (e) for details. These provide us with accurate estimates of the critical exponents for any C value.

Finally, we perform the power-law fit of the expression $p_c - p_c(N) = bN^{-1/\bar{\nu}}$, see Eq. (S9), to determine the exponent $1/\bar{\nu}$ as well as the value of the constant b , Fig. S2(a). Numerical estimates of the critical parameters are reported in Table S2.

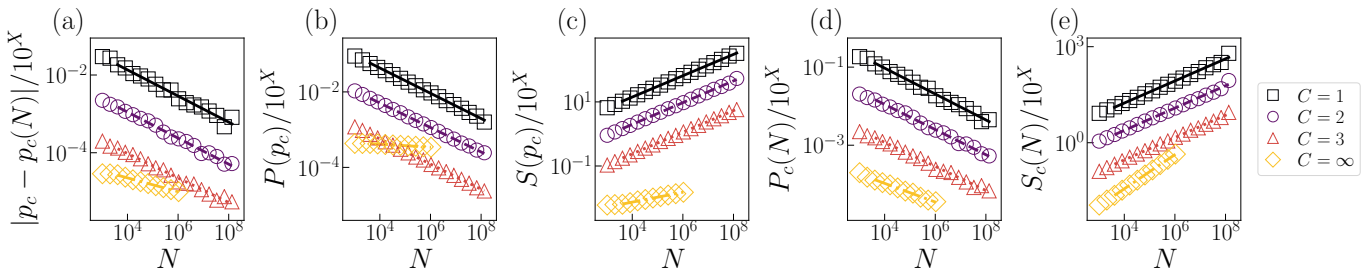


FIG. S2. Finite-size scaling analysis in the conventional ensemble. An additional factor 10^X is used for visual clarity. We use $X = 0, 1, 2, 3$ for $C = 1, 2, 3, \infty$, respectively. In each panel, we display as symbols the average value of the metric. Error bars are not displayed to avoid overcrowding the figure. Lines appearing in the plot are best fits obtained via simple linear regression on log-transformed variables, see text for details. (a) Scaling of Eq. (S9) to estimate $1/\bar{\nu}$. (b) Scaling at the critical point for the percolation strength, i.e., $P(p = p_c)$ vs N , to estimate $\beta/\bar{\nu}$. (c) Scaling at the critical point for the average cluster size, i.e., $S(p = p_c)$ vs N , to estimate $\gamma/\bar{\nu}$. (d) Scaling at the pseudo-critical point for the percolation strength, i.e., $P(p = p_c(N))$ vs N , to estimate $\beta/\bar{\nu}$. (e) Scaling at the pseudo-critical point for the average cluster size, i.e., $S(p = p_c(N))$ vs N , to estimate $\gamma/\bar{\nu}$. Estimated exponents are reported in Table S2.

C	critical		pseudo-critical			
	p_c	$1/\bar{\nu}$	$\beta/\bar{\nu}$	$\gamma/\bar{\nu}$		
1	0.750 ± 0.001	0.34 ± 0.02	0.33 ± 0.01	0.33 ± 0.01	0.33 ± 0.01	0.35 ± 0.001
2	0.701 ± 0.001	0.34 ± 0.01	0.32 ± 0.01	0.34 ± 0.01	0.32 ± 0.01	0.35 ± 0.01
3	0.682 ± 0.001	0.32 ± 0.01	0.33 ± 0.01	0.34 ± 0.01	0.31 ± 0.01	0.35 ± 0.01
∞	0.646 ± 0.001	0.18 ± 0.01	0.04 ± 0.01	0.15 ± 0.01	0.24 ± 0.01	0.53 ± 0.01

TABLE S2. Critical properties of the shortest-path-percolation (SPP) model. Estimates reported here are obtained via finite-size scaling analysis in the conventional ensemble.

C. Event-based ensemble

Also here, we denote with $P(p, N)$ and $S(p, N)$ the values of the metrics of Eqs. (S2) and (S3), respectively, when exactly a fraction of p edges is removed from the graph. The single-instance pseudo-critical point can be defined as in Ref. [2], i.e.,

$$p^+(N) = \arg \max_p [P(p - 2/N\bar{k}, N) - P(p, N)] , \quad (\text{S13})$$

or as in Ref. [4], i.e.,

$$p^-(N) = \arg \max_p [P(p, N) - P(p + 2/N\bar{k}, N)] . \quad (\text{S14})$$

In the above equations, the superscript $+$ ($-$) indicates whether the value of the control parameter p is measured immediately after (before) the largest drop, induced by the removal of a single edge, in the value of the order parameter

P is observed. Note that the values $p^+(N)$ and $p^-(N)$ are measured for each individual realization of the SPP model. For such an individual realization, we can also measure the single-instance pseudo-critical value of the percolation strength, i.e.,

$$P^+(N) = P(p = p^+(N), N). \quad (\text{S15})$$

Similar definitions are valid for $P^-(N)$, and also for $S^+(N)$ and $S^-(N)$.

The best estimate of the pseudo-critical point under the event-based ensemble is obtained by taking the ensemble average of the single-instance pseudo-critical points of Eq. (S13), thus

$$p_c(N) = \langle p^+(N) \rangle. \quad (\text{S16})$$

An analogous definition is obtained by using $p^-(N)$ in place of $p^+(N)$. The quantity defined in Eq. (S16) still obeys the same scaling relationship as of Eq. (S9). One can also measure the standard deviation of the single-instance pseudo-critical point, i.e.,

$$\sigma_{p_c}(N) = \sqrt{\langle [p^+(N)]^2 \rangle - [\langle p^+(N) \rangle]^2}. \quad (\text{S17})$$

Such a quantity obeys the scaling relation

$$\sigma_{p_c}(N) \sim N^{-1/\bar{\nu}}. \quad (\text{S18})$$

The single-instance pseudo-critical value $P^+(N)$ obeys the FSS distribution

$$\text{Prob.}[P^+(N)] = Q^{(1)}[P^+(N) N^{\beta/\bar{\nu}}]. \quad (\text{S19})$$

For the average cluster size, we have instead

$$\text{Prob.}[S^+(N)] = Q^{(2)}[S^+(N) N^{-\gamma/\bar{\nu}}]. \quad (\text{S20})$$

Here, $Q^{(1)}(\cdot)$ and $Q^{(2)}(\cdot)$ are universal scaling distributions. In particular, the average values of such distributions obey the scaling relationships

$$P_c(N) = \langle P^+(N) \rangle \sim N^{-\beta/\bar{\nu}} \quad (\text{S21})$$

and

$$S_c(N) = \langle S^+(N) \rangle \sim N^{\gamma/\bar{\nu}}, \quad (\text{S22})$$

respectively. Analogous scaling relationships are valid also for the standard deviation of the distributions $Q^{(1)}(\cdot)$ and $Q^{(2)}(\cdot)$, i.e.,

$$\sigma_{P_c}(N) = \sqrt{\langle [P^+(N)]^2 \rangle - [\langle P^+(N) \rangle]^2} \sim N^{-\beta/\bar{\nu}} \quad (\text{S23})$$

and

$$\sigma_{S_c}(N) = \sqrt{\langle [S^+(N)]^2 \rangle - [\langle S^+(N) \rangle]^2} \sim N^{\gamma/\bar{\nu}}. \quad (\text{S24})$$

All the above equations are valid also if we replace $P^+(N)$ with $P^-(N)$ and $S^+(N)$ with $S^-(N)$.

D. Critical properties of the shortest-path-percolation in the event-based ensemble

The following applies regardless if one consider the definition of single-instance pseudo-critical point of Eq. (S13) or of Eq. (S14). Different estimates of the critical exponents are obtained either by using the scaling of the average of the various observables, or their standard deviations. The numerical results summarized in Tables S3 and S4 are organized accordingly.

We first determine the best estimate of the critical threshold using Eq. (S9). Here, both p_c and $1/\bar{\nu}$ are treated as free parameters to perform the power-law fit, see Fig. S3 (a) and Fig. S4 (a). Also, we determine the critical exponent $1/\bar{\nu}$ using the scaling of Eq. (S18), see Fig. S3 (d) and Fig. S4 (d).

We then rely on the scaling laws of Eqs. (S21) and (S22) to determine the ratio between the critical exponents $\beta/\bar{\nu}$ and $\gamma/\bar{\nu}$, respectively, see Figs. S3 (b), (c), and Figs. S4 (b), (c). Additional estimates of these exponents are also obtained from Eqs. (S23) and (S24), see Figs. S3 (e), (f), and Figs. S4 (e), (f). Data collapse based on Eqs. (S19) and (S20) is provided in Fig. 3 of the main paper and in Fig. S5.

It's worth mentioning that the two estimates of $1/\bar{\nu}$ obtained from Eqs. (S17) and (S18) are identical for finite C , but they differ for $C = \infty$. The latter finding is an agreement with the results of Ref. [4] for explosive percolation, with the caveat that the SPP estimate obtained from Eq. (S17) is smaller than the one obtained from Eq. (S18). The exact opposite was instead found for the case of explosive percolation.

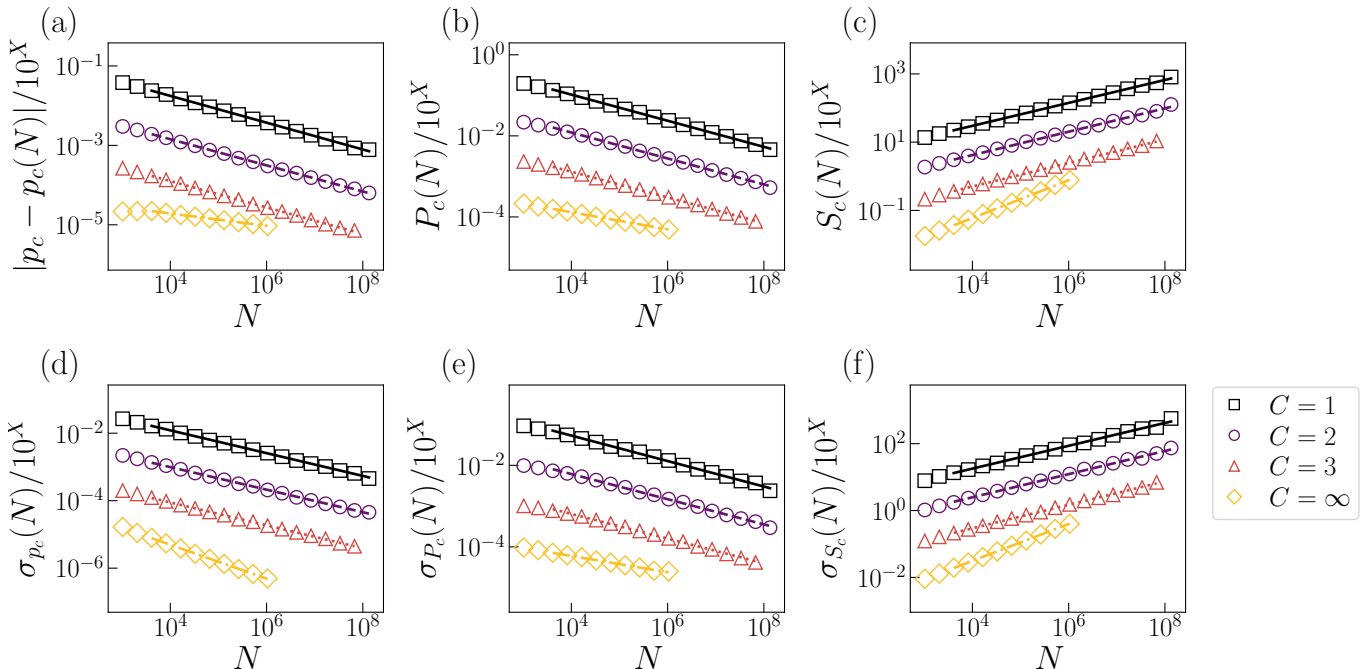


FIG. S3. Finite-size scaling analysis in the event-based ensemble using the pseudo-critical point of Eq. (S13). Note that an additional factor 10^X was used for visual clarity. We use $X = 0, 1, 2, 3$ for $C = 1, 2, 3, \infty$, respectively. In each panel, we display as symbols the average value of the metric. Error bars are not displayed to avoid overcrowding the figure. Lines appearing in the plot are best fits obtained via simple linear regression on log-transformed variables, see text for details. (a) Scaling of Eq. (S9) to estimate $1/\bar{\nu}$. (b) Scaling at the critical point for the percolation strength, i.e., $P(p = p_c)$ vs N , to estimate $\beta/\bar{\nu}$. (c) Scaling at the critical point for the average cluster size, i.e., $S(p = p_c)$ vs N , to estimate $\gamma/\bar{\nu}$. (d) Scaling of Eq. (S18) to estimate $1/\bar{\nu}$. (e) Scaling at the pseudo-critical point for the percolation strength, i.e., $P(p = p_c(N))$ vs N , to estimate $\beta/\bar{\nu}$. (f) Scaling at the pseudo-critical point for the average cluster size, i.e., $S(p = p_c(N))$ vs N , to estimate $\gamma/\bar{\nu}$. Estimated exponents are reported in Table S3.

C	p_c	average			fluctuation		
		$1/\bar{\nu}$	$\beta/\bar{\nu}$	$\gamma/\bar{\nu}$	$1/\bar{\nu}$	$\beta/\bar{\nu}$	$\gamma/\bar{\nu}$
1	0.750 ± 0.001	0.34 ± 0.01	0.32 ± 0.01	0.34 ± 0.01	0.34 ± 0.01	0.31 ± 0.01	0.34 ± 0.01
2	0.701 ± 0.001	0.33 ± 0.01	0.32 ± 0.01	0.34 ± 0.01	0.33 ± 0.01	0.30 ± 0.01	0.35 ± 0.01
3	0.682 ± 0.001	0.33 ± 0.01	0.32 ± 0.01	0.34 ± 0.01	0.35 ± 0.01	0.31 ± 0.01	0.36 ± 0.01
∞	0.646 ± 0.001	0.15 ± 0.01	0.21 ± 0.01	0.55 ± 0.01	0.50 ± 0.01	0.20 ± 0.01	0.55 ± 0.01

TABLE S3. Critical properties of the shortest-path-percolation (SPP) model in the event-based ensemble. Estimates reported here are obtained via finite-size scaling analysis in the event-based ensemble using the pseudo-critical point of Eq. (S13).

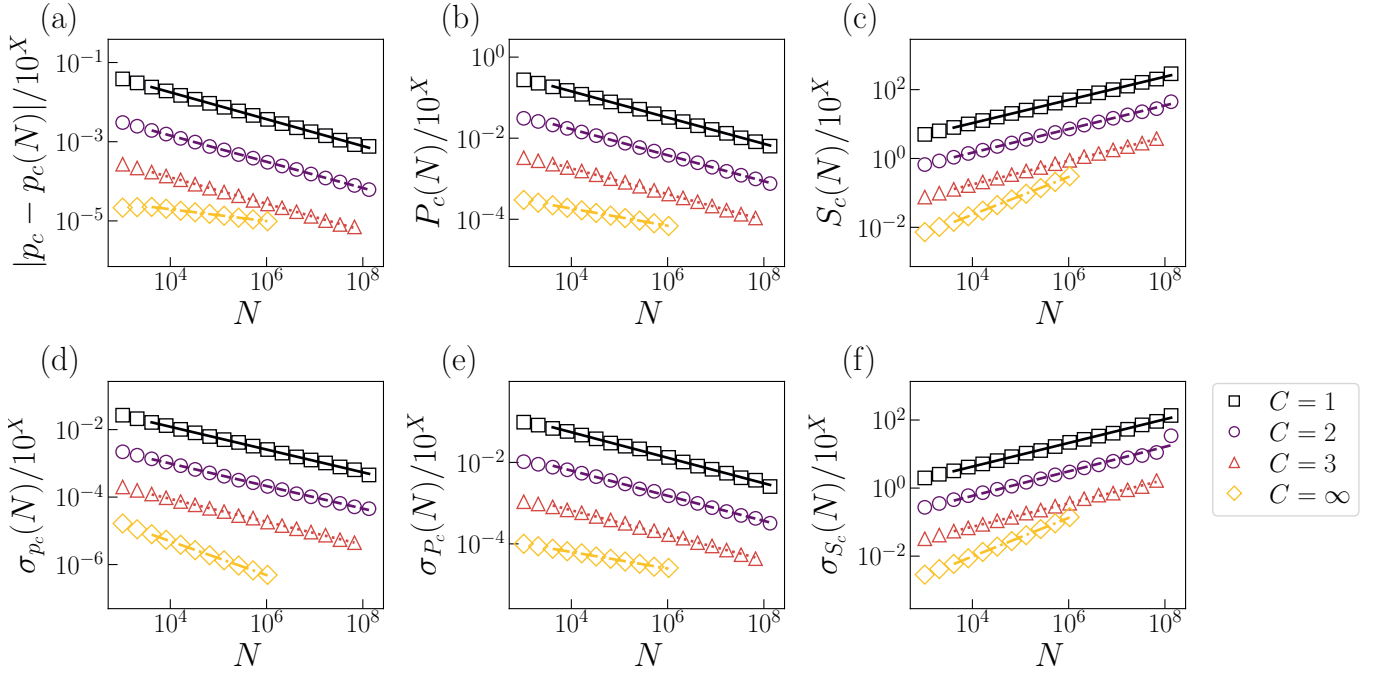


FIG. S4. Finite-size scaling analysis in the event-based ensemble using the pseudo-critical point of Eq. (S14). Note that an additional factor 10^X is used for visual clarity. We use $X = 0, 1, 2, 3$ for $C = 1, 2, 3, \infty$, respectively. In each panel, we display as symbols the average value of the metric. Error bars are not displayed to avoid overcrowding the figure. Lines appearing in the plot are best fits obtained via simple linear regression on log-transformed variables, see text for details. (a) Scaling of Eq. (S9) to estimate $1/\bar{\nu}$. (b) Scaling at the critical point for the percolation strength, i.e., $P(p = p_c)$ vs N , to estimate $\beta/\bar{\nu}$. (c) Scaling at the critical point for the average cluster size, i.e., $S(p = p_c)$ vs N , to estimate $\gamma/\bar{\nu}$. (d) Scaling of Eq. (S18) to estimate $1/\bar{\nu}$. (e) Scaling at the pseudo-critical point for the percolation strength, i.e., $P(p = p_c(N))$ vs N to estimate, $\beta/\bar{\nu}$. (f) Scaling at the pseudo-critical point for the average cluster size, i.e., $S(p = p_c(N))$ vs N , to estimate $\gamma/\bar{\nu}$. Estimated exponents are reported in Table S4.

		average			fluctuation		
C	p_c	$1/\bar{\nu}$	$\beta/\bar{\nu}$	$\gamma/\bar{\nu}$	$1/\bar{\nu}$	$\beta/\bar{\nu}$	$\gamma/\bar{\nu}$
1	0.750 ± 0.001	0.34 ± 0.01	0.32 ± 0.01	0.34 ± 0.01	0.34 ± 0.01	0.31 ± 0.01	0.35 ± 0.01
2	0.701 ± 0.001	0.33 ± 0.01	0.32 ± 0.01	0.34 ± 0.01	0.33 ± 0.01	0.30 ± 0.01	0.35 ± 0.01
3	0.682 ± 0.001	0.33 ± 0.01	0.32 ± 0.01	0.34 ± 0.01	0.35 ± 0.01	0.31 ± 0.01	0.33 ± 0.01
∞	0.646 ± 0.001	0.15 ± 0.01	0.21 ± 0.01	0.55 ± 0.01	0.50 ± 0.01	0.20 ± 0.01	0.57 ± 0.01

TABLE S4. Critical properties of the shortest-path-percolation (SPP) model in the event-based ensemble. Estimates reported here are obtained via finite-size scaling analysis in the event-based ensemble using the pseudo-critical point of Eq. (S14).

E. Changing control parameter

1. Mapping between control parameters

The two natural control parameters of the SPP model, i.e., p and t , can be mapped one to the other using Eq. (S1). Typical results of this mapping are displayed in Fig. 4(a) of the main paper. For finite C , we find that the map can be written as

$$p = V(t N^{-\theta}) . \quad (\text{S25})$$

Here, $V(\cdot)$ is a universal scaling function. An immediate way of estimating the exponent θ is to look at how $t_c(N)$ scales as function of N , see Fig. S6 (a). Also, we obtain another estimate of θ from the scaling of $t_{\max}(N)$, i.e., the total number of random origin-destination pairs that one has to extract in order to remove all edges from an ER network of size N , see Fig. S6 (b).

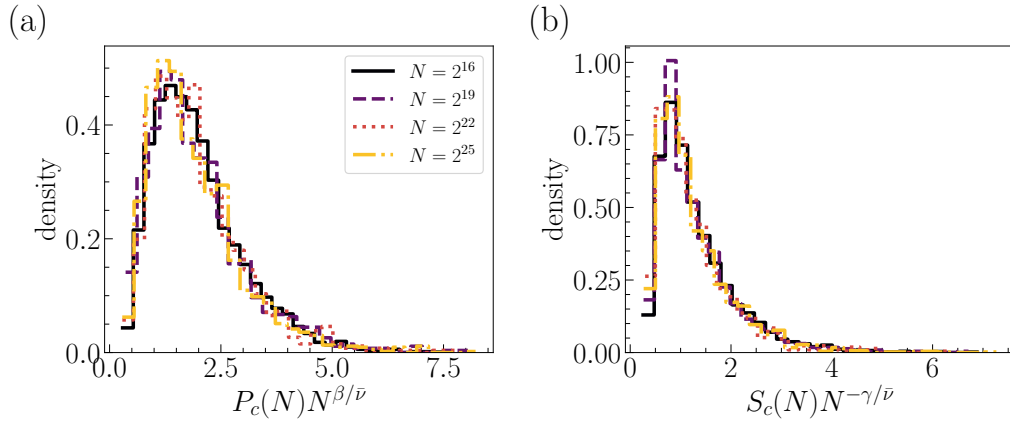


FIG. S5. Shortest-path-percolation transition in the event-based ensemble. Here, we used $C = 1$. (a) The distribution of $P_c(N) N^{\beta/\bar{\nu}}$ with $\beta/\bar{\nu} = 0.32$. Different curves correspond to different network sizes N . (b) Same as in (a) but for $S_c(N) N^{-\gamma/\bar{\nu}}$ with $\gamma/\bar{\nu} = 0.34$.

For infinite C , we find instead that

$$p = \begin{cases} V^-(tN^{-\theta^\ominus}) & \text{if } t \leq t_c(N) \\ V^+(tN^{-\theta^\oplus}) & \text{if } t > t_c(N) \end{cases}, \quad (\text{S26})$$

where $V^-(\cdot)$ and $V^+(\cdot)$ are universal scaling functions valid respectively for the supercritical and subcritical regimes of the SPP transition. A good estimate θ^\ominus can be obtained by still looking at how $t_c(N)$ scales with N , see Fig. S6 (a). We determine the best estimate of θ^\oplus from the scaling of $t_{\max}(N)$ vs N , see Fig. S6 (b).

We test the scaling of Eq. (S26) also for $C = N^{1/3}$ (Fig. S7) and $C = \log(N)$ (Fig. S8). We do not estimate the critical exponents θ^\ominus and θ^\oplus , rather simply verify that curve collapse occur for the same values of the critical exponents obtained for $C = N$. We see that, for sufficiently large N , the curves obtained for $C = N^{1/3}$ and $C = N$ are basically identical. Differences between the cases $C = \log(N)$ and $C = N$ are apparent, however, the corresponding curves are almost identical in both their leftmost and rightmost parts.

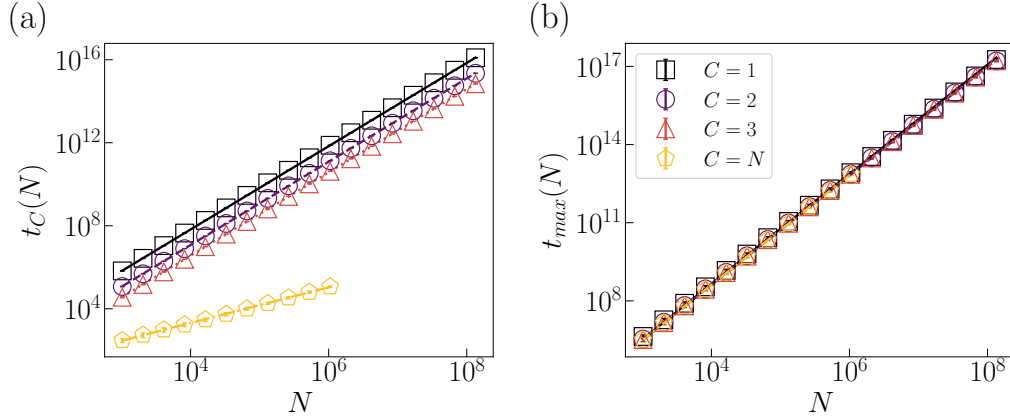


FIG. S6. (a) $t_c(N)$ as a function of network size N for different values of C . For each N value, we plot the average values of $t_c(N)$ as symbols and the associated standard deviation as error bars. The scaling relation is given by $t_c(N) \sim N^\theta$ with $\theta = 2.01 \pm 0.01$ for $C = 1$, $\theta = 2.01 \pm 0.01$ for $C = 2$, $\theta = 2.01 \pm 0.01$ for $C = 3$. For $C = \infty$, we fit $t_c(N) \sim N^{\theta^\ominus}$ and obtain $\theta^\ominus = 0.86 \pm 0.01$. (b) Maximum value of the raw number of demand $t_{\max}(N)$ as a function of the network size N for different values of C . The scaling relation is given by $t_{\max}(N) \sim N^\theta \log(N)$ with $\theta = 2.08 \pm 0.01$ for $C = 1$, $\theta = 2.07 \pm 0.01$ for $C = 2$, $\theta = 2.06 \pm 0.01$ for $C = 3$. For $C = \infty$, we look at $t_{\max}(N) \sim N^{\theta^\oplus} \log(N)$ and find $\theta^\oplus = 2.07 \pm 0.01$.

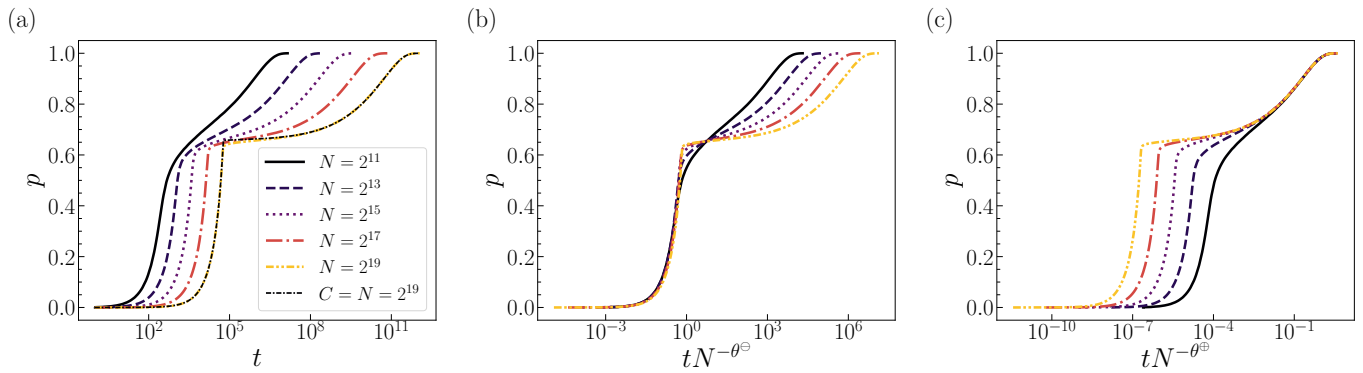


FIG. S7. Control parameters in the shortest-path percolation model. (a) The fraction of removed edges p is plotted as a function of number of demanding agents t for different values of N . Here $C = N^{1/3}$, but we display also the curve for $C = N = 2^{19}$. (b) Same as in (a), but the abscissas values are rescaled as $tN^{-\theta^\ominus}$, with $\theta^\ominus = 0.86$, to obtain a collapse between the various curves. (c) Same as in (a), but the abscissas values are rescaled as $tN^{-\theta^\oplus}$, with $\theta^\oplus = 2.00$.

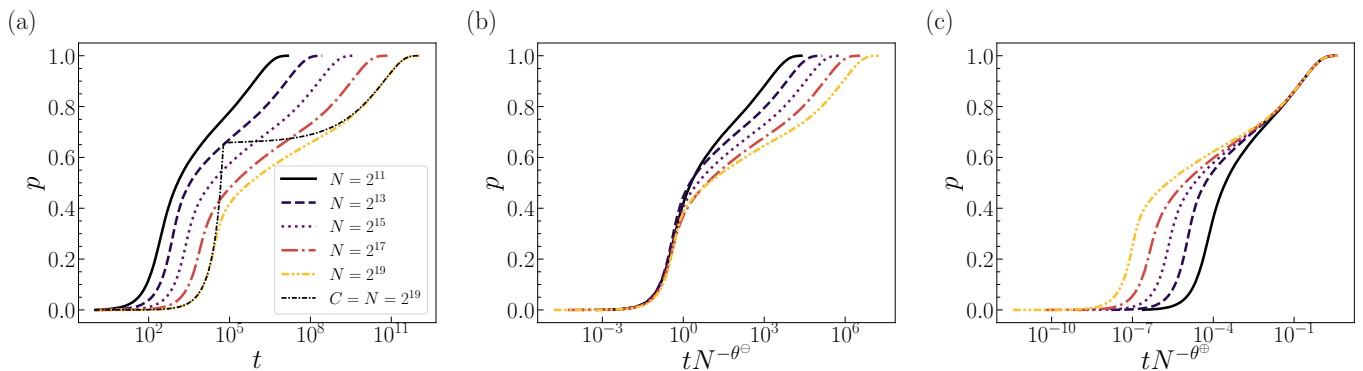


FIG. S8. Control parameters in the shortest-path percolation model. (a) The fraction of removed edges p is plotted as a function of number of demanding agents t for different values of N . Here $C = \log(N)$, but we display also the curve for $C = N = 2^{19}$. (b) Same as in (a), but the abscissas values are rescaled as $tN^{-\theta^\ominus}$, with $\theta^\ominus = 0.86$, to obtain a collapse between the various curves. (c) Same as in (a), but the abscissas values are rescaled as $tN^{-\theta^\oplus}$, with $\theta^\oplus = 2.00$.

2. Critical properties of the shortest-path-percolation transition using the control parameter t

All the above theoretical arguments and numerical analyses generalize immediately if one replaces the control parameter p with the control parameter t . Specifically, we look at the rescaled control parameter $\tilde{t} = tN^{-2}$ to deal with an intensive quantity similar to p .

Critical threshold values remain finite as long as C is finite. A vanishing threshold is instead obtained for infinite C .

Due to the one-to-one mapping between $p_c(N)$ and $\tilde{t}_c(N)$, critical exponent values concerning the order parameter and the average cluster size are unaffected by the specific choice of the control parameter. For finite C , some variations are observed for the value of the exponent $1/\bar{\nu}$; these variations are, however, sufficiently small and likely due to finite-size effects. For infinite C , we recover $1/\bar{\nu} = 2 - \theta^\ominus$. Results of the analysis are displayed in Fig. S9 and summarized in Table S5.

F. Scaling of the size of the k -largest cluster

The scaling of Eq. (S21) valid for the relative size of the largest cluster applies also to the relative size of the k -th largest cluster, i.e.,

$$P_c^{(k)}(N) = \langle \max_p P^{(k)}(p, N) \rangle \sim N^{\beta/\bar{\nu}}. \quad (\text{S27})$$

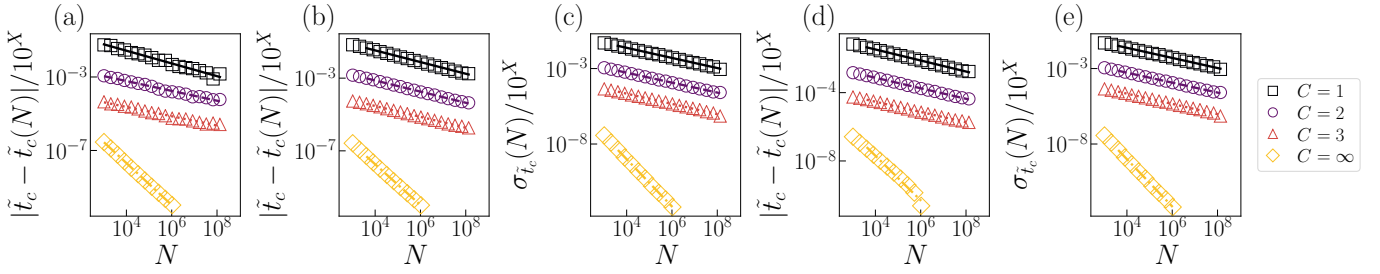


FIG. S9. Finite-size scaling analysis with the control parameter t . Note that an additional factor 10^X is used for visual clarity. We use $X = 0, 1, 2, 3$ for $C = 1, 2, 3, \infty$, respectively. In each panel, we display as symbols the average value of the metric. Error bars are not displayed to avoid overcrowding the figure. Lines appearing in the plot are best fits obtained via simple linear regression on log-transformed variables, see text for details. (a) Scaling of Eq. (S9) to estimate $1/\bar{\nu}$ in the conventional ensemble. (b) Similar to (a), but in the event-based ensemble using the pseudo-critical point $t_c(N)$ which corresponds to Eq. (S13). (c) Scaling of Eq. (S18) using the pseudo-critical point $t_c(N)$ which corresponds to Eq. (S13). (d) Similar to (b), but using the pseudo-critical point $t_c(N)$ which corresponds to Eq. (S14). (e) Similar to (c) but using the pseudo-critical point $t_c(N)$ which corresponds to Eq. (S14).

C	conventional		event-based (Eq. S13)		fluctuation	event-based (Eq. S14)		fluctuation
	\tilde{t}_c	$1/\bar{\nu}$	\tilde{t}_c	$1/\bar{\nu}$	$1/\bar{\nu}$	\tilde{t}_c	$1/\bar{\nu}$	$1/\bar{\nu}$
1	0.693 ± 0.001	0.35 ± 0.01	0.693 ± 0.001	0.32 ± 0.01	0.34 ± 0.01	0.696 ± 0.001	0.32 ± 0.01	0.34 ± 0.01
2	0.125 ± 0.001	0.27 ± 0.01	0.124 ± 0.001	0.31 ± 0.01	0.33 ± 0.01	0.124 ± 0.001	0.31 ± 0.01	0.33 ± 0.01
3	0.039 ± 0.001	0.25 ± 0.01	0.039 ± 0.001	0.30 ± 0.01	0.34 ± 0.01	0.039 ± 0.001	0.30 ± 0.01	0.34 ± 0.01
∞	0.000 ± 0.000	1.14 ± 0.01	0.000 ± 0.000	1.11 ± 0.01	1.57 ± 0.01	0.000 ± 0.000	1.31 ± 0.01	1.57 ± 0.01

TABLE S5. Estimation of the critical threshold \tilde{t}_c and of the critical exponent $1/\bar{\nu}$ when \tilde{t} is used as control parameter to monitor the transition. For $C = \infty$, we estimate very small values of the threshold \tilde{t}_c , well below the numerical accuracy reported in the table.

In Fig. S10, we show that the scaling holds for $C = 1, 2, 3$ and $C = \infty$, for $k = 2, 3, 4$, and 5. Estimated exponents are also reported in Table S6. As seen in Fig. S10 and Table S6, we could clearly obtain identical exponents from Eq. (S27).

C	$k = 2$	$k = 3$	$k = 4$	$k = 5$
1	0.31 ± 0.01	0.32 ± 0.01	0.32 ± 0.01	0.32 ± 0.01
2	0.32 ± 0.01	0.33 ± 0.01	0.32 ± 0.01	0.33 ± 0.01
3	0.30 ± 0.01	0.30 ± 0.01	0.30 ± 0.01	0.31 ± 0.01
∞	0.23 ± 0.01	0.23 ± 0.01	0.23 ± 0.01	0.23 ± 0.01

TABLE S6. Estimation of the exponent $\beta/\bar{\nu}$ from Eq. (S27).

V. SIMULATING THE SHORTEST-PATH-PERCOLATION MODEL

The inputs of the algorithm are: (i) a graph $\mathcal{G}_1 = (\mathcal{V}, \mathcal{E}_1)$ composed of $N = |\mathcal{V}|$ nodes and $E_1 = |\mathcal{E}_1|$ edges; (ii) a list of T origin-destination pairs, i.e., $(o_1 \rightarrow d_1, o_2 \rightarrow d_2, \dots, o_t \rightarrow d_t, \dots, o_T \rightarrow d_T)$; the maximum budget C available to each agent. We assume that the graph \mathcal{G}_1 is unweighted and symmetric.

We denote with $\mathcal{G}_t = (\mathcal{V}, \mathcal{E}_t)$ the graph available to the t -th agent when the agent demands an itinerary for its origin-destination $o_t \rightarrow d_t$. We set $t = 1$ and iterate the following:

1. We find all shortest paths between the origin o_t and the destination d_t ; the shortest paths are found on the graph \mathcal{G}_t using only edges in the set \mathcal{E}_t . Denote with Q_t the length of such shortest path. We copy the graph $\mathcal{G}_{t+1} \mapsto \mathcal{G}_t$.

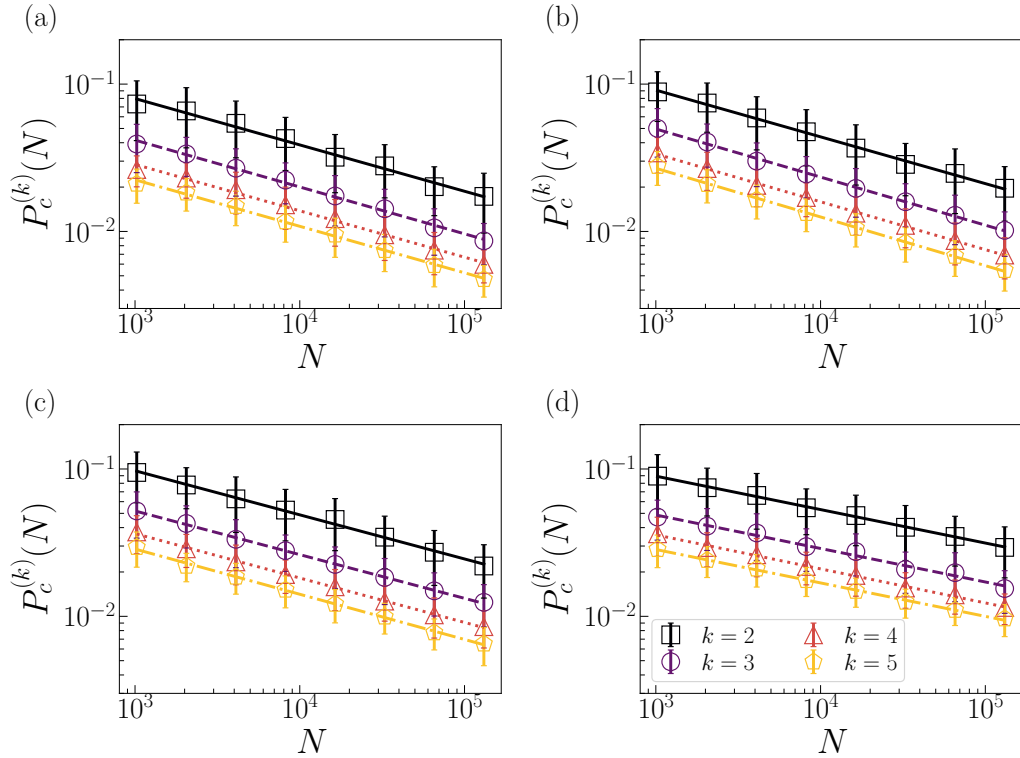


FIG. S10. Scaling of the relative size of the k -th largest cluster. Results reported here are valid for $k = 2, 3, 4$ and 5 . We display as points the average values of the relative size of the k -th largest cluster, and with the error bars their standard deviations. Here, we are using 100 samples. The different panels correspond to different values of the model parameter C . Specifically, (a) $C = 1$, (b) $C = 2$, (c) $C = 3$, and (d) $C = \infty$. See Table S6 for detailed information.

2. If there is no path from o_t to d_t or if $Q_t > C$, we proceed to point 4.
3. If a path from o_t to d_t exists and $Q_t \leq C$, then we select one of the shortest paths between o_t and d_t at random. We denote this path as $(i_0 = o_t, i_1, i_2, \dots, i_{Q_t-1}, i_{Q_t} = d_t)$. All edges in the path are removed from the current graph, i.e., $\mathcal{E}_{t+1} \mapsto \mathcal{E}_t \setminus \bigcup_{q=1}^{Q_t} (i_{q-1}, i_q)$.
4. We increase $t \mapsto t + 1$ and go back to point 1.

The former algorithm is iterated until $t \leq T$.

A random shortest path between a generic pair of nodes o and d can be efficiently selected by first applying a breadth-first search (BFS) from o to d . We perform a BFS with maximum depth C , as only shortest paths of length at most equal to C are considered. If node d is not reached with a BFS with maximum depth C then no shortest path shorter than C exists between o and d . While performing BFS, we construct a directed acyclic graph (DAG) that contains the entire information about all eventual shortest paths between o and d . Specifically, for a generic node i in the DAG, we store the information about its parents in the set \mathcal{P}_i . Also, we keep track of the number of shortest paths s_i that originate in node o and arrive to node i . Once BFS is completed, we sample one of the shortest paths between o and d by performing a random walk from d to o on the DAG. At each generic node i along the sampled walk, we walk back towards o by randomly selecting a parent of i from the set \mathcal{P}_i . The parent $j \in \mathcal{P}_i$ is selected proportionally to the pre-computed number of shortest paths s_j passing through it.

A. Shortest-path percolation for randomly selected pairs of nodes

In our paper, we study the case in which the SPP model assumes that a random origin-destination pair is selected at a time. The naive implementation of this procedure could be computationally inefficient as potentially many of the randomly generated pairs may not satisfy the above requirement, thus leading to a very large number of attempts before a suitable pair is effectively generated. The issue is exacerbated for finite C and when the graph progressively becomes sparser/disconnected due to the removal of edges.

To avoid the computational issue and be able to simulate the model in sufficiently large networks, we proceed as follows. We first construct the set of potentially suitable origin-destination pairs, i.e., $\mathcal{L} = \{o_1 \rightarrow d_1, o_2 \rightarrow d_2, \dots, o_z \rightarrow d_z, \dots, o_Z \rightarrow d_Z\}$, as the set of all origin-destination nodes at distance less than or equal to C in the graph \mathcal{G}_1 . Nodes in the network are labeled from 1 to N , in the set \mathcal{L} we have only pairs $o_z < d_z$ to avoid double counting. We set $t = 1$, and then apply the following:

1. We extract a random number τ from the geometric distribution

$$P(\tau) = \frac{2L}{N(N-1)} \left[1 - \frac{2L}{N(N-1)} \right]^{\tau-1}, \quad (\text{S28})$$

where L is the number of pairs in the set \mathcal{L} . The probability of success in the geometric distribution is $\frac{2L}{N(N-1)}$, and τ represents the number of random pairs that one needs to generate in order to find a pair that belongs to \mathcal{L} . We create the graph $\mathcal{G}_{t+\tau} \mapsto \mathcal{G}_t$. Then, we increase $t \mapsto t + \tau$.

2. We select a random pair, say $o_t \rightarrow d_t$, from the list \mathcal{L} . We apply the shortest-path-percolation algorithm by first determining whether o_t and d_t are in the same connected component and finding all eventual shortest paths connecting the two nodes. We then remove all edges along the shortest path between o_t and d_t only if such a shortest path exists and has length smaller or equal than C . Similarly to what we described in the preceding section, this step is performed on the graph \mathcal{G}_t . The result of the operation is the graph \mathcal{G}_t containing E_t edges.
3. We verify that the pair $o_t \rightarrow d_t$ can still belong to the list \mathcal{L} by measuring the shortest-path distance between o_t and d_t in the graph \mathcal{G}_t . If such a distance is larger than C , we remove the pair $o_t \rightarrow d_t$ from \mathcal{L} , i.e., $\mathcal{L} \mapsto \mathcal{L} \setminus \{o_t \rightarrow d_t\}$. If the pair stays in the set, it will be re-considered in next iterations of the algorithm; if the pair is removed from \mathcal{L} , then the probability of Eq. (S28) is automatically updated to account for it at the next stage of the algorithm.
4. Until the list \mathcal{L} is not empty, we go back to point 1.

To generate a random number from the distribution of Eq. (S28), we use

$$\tau = 1 + \left\lfloor \log(U) / \log \left[1 - \frac{2L}{N(N-1)} \right] \right\rfloor,$$

for $0 < \frac{2L}{N(N-1)} < 1$, and $\tau = 1$ for $\frac{2L}{N(N-1)} = 1$. In the above expression, U is a random variate extracted from the uniform distribution defined in the range $(0, 1)$, and $\lfloor \cdot \rfloor$ is the floor function.

We remark that the above procedure is more efficient than the naive implementation of the SPP model only if $Z \leq \binom{N}{2}$. This condition is true only when $C \leq \log(N/\bar{k})$ in ER graphs with size N and average degree \bar{k} . Specifically, the computational complexity of the two implementations of the model is identical even for $Z = \binom{N}{2}$, however, the proposed alternative implementation is subject to a greater space complexity since it relies on the fact that a list composed of Z elements is stored in memory.

For finite C , we use this efficient strategy from the beginning of the simulation. We have in fact that $L \simeq N\bar{k}C/2$, so that space complexity scales linearly with N ; we find that the amount of time required to simulate the SPP model scales slightly super-linearly with N , see Figure S11.

Mixed strategy for the simulation of the infinite- C SPP model

For the infinite- C SPP model, we adopt the following mixed strategy:

- A) Starting from the input network, we run the naive implementation of the SPP algorithm until no edges are deleted for T_* consecutive randomly selected pairs of nodes.
- B) We keep removing edges using the alternative version of the algorithm described above. This second part is started from the network remaining after the deletion of the edges occurred at point A.

The output is given by the concatenation of the two lists of edges identified at points A and B, respectively.

The value of the parameter T_* that determines the condition to pass from point A to point B is chosen in a way that optimizes the trade-off between time and space complexity. If, at a certain stage of the simulation, \mathcal{L} is the set

of pairs with distance shorter or equal than C , then the probability of extracting less than T_* consecutive random pairs of nodes outside this set is given by the cumulative distribution of Eq. (S28), thus

$$P(\tau < T_*) = \left[1 - \frac{2L}{N(N-1)} \right]^{T_*}, \quad (\text{S29})$$

where we indicated with $L = |\mathcal{L}|$ the size of the set \mathcal{L} . We remind that L does not increase during the course of a simulation. Ideally, we would like to set T_* such that we pass from point A to point B when $L \ll N^2$; this ensures that we are able to take advantage of the computational speed up without incurring the computational burden of pre-computing and storing the distance between a large number of pairs of nodes.

For infinite C , $L \sim N^2$ from the beginning of the simulation until the critical point is reached. At criticality, we have instead $L \sim N^{2-2\beta/\bar{\nu}} + N^{1+\gamma/\bar{\nu}} \ll N^2$, where β , γ and $\bar{\nu}$ are the critical exponents associated with the size of the largest cluster and the average cluster size. This follows from the fact that the size of the largest cluster scales as $N^{1-\beta/\bar{\nu}}$ at criticality [Eq. (S10)], and all the pairs of nodes within such a cluster are part of the set \mathcal{L} . Other elements of the set \mathcal{L} are pairs of nodes in finite clusters, which is proportional to the average cluster size $SN \sim N^{1+\gamma/\bar{\nu}}$, see Eq. (S3). We note that $2 - 2\beta/\bar{\nu} = 1 + \gamma/\bar{\nu}$, we can therefore write

$$P(\tau < T_*) \simeq \left[1 - N^{-2\beta/\bar{\nu}} \right]^{T_*} \simeq e^{-T_* N^{2\beta/\bar{\nu}}}, \quad (\text{S30})$$

where we used the approximation $\log(1-x) \simeq -x$ for $x \ll 1$. A reasonable way of stating that $P(\tau < T_*)$ is small enough is to impose $P(\tau < T_*) < 1/V$, with V number of simulations performed, leading to

$$T_* > \log(V) N^{2\beta/\bar{\nu}}. \quad (\text{S31})$$

We set $T_* = 7N^{0.42}$ corresponding to $V = 1,000$ (i.e., roughly equal to the number of our experiments for most of the N values considered, see Table S1) and $2\beta/\bar{\nu} = 0.42$ as measured in our numerical tests, see Table I of the main text. Before the critical point, the application of point A requires a time that scales as $N^{1+\theta^\ominus} \simeq N^{1.86}$, as we need to select $t_c(N) \sim N^{\theta^\ominus}$ random pairs of nodes to reach the critical point, and for each of them we need to find the shortest paths in a time that scales as N , see Eq. (S26) and Fig. S6. Once the critical point is reached, we need to determine the set \mathcal{L} ; the operation requires time and space complexity both growing as $N^{2-2\beta/\bar{\nu}} \simeq N^{1.58}$, see for example Fig. S2. The above predictions about time and space complexity of the mixed strategy used to simulate the infinite- C SPP model are validated in Fig. S11.

-
- [1] M. Newman and R. M. Ziff, Efficient monte carlo algorithm and high-precision results for percolation, *Physical Review Letters* **85**, 4104 (2000).
 - [2] J. Fan, J. Meng, Y. Liu, A. A. Saberi, J. Kurths, and J. Nagler, Universal gap scaling in percolation, *Nature Physics* **16**, 455 (2020).
 - [3] D. Stauffer and A. Aharony, *Introduction to percolation theory* (CRC press, 1985).
 - [4] M. Li, J. Wang, and Y. Deng, Explosive percolation obeys standard finite-size scaling in an event-based ensemble, *Physical Review Letters* **130**, 147101 (2023).

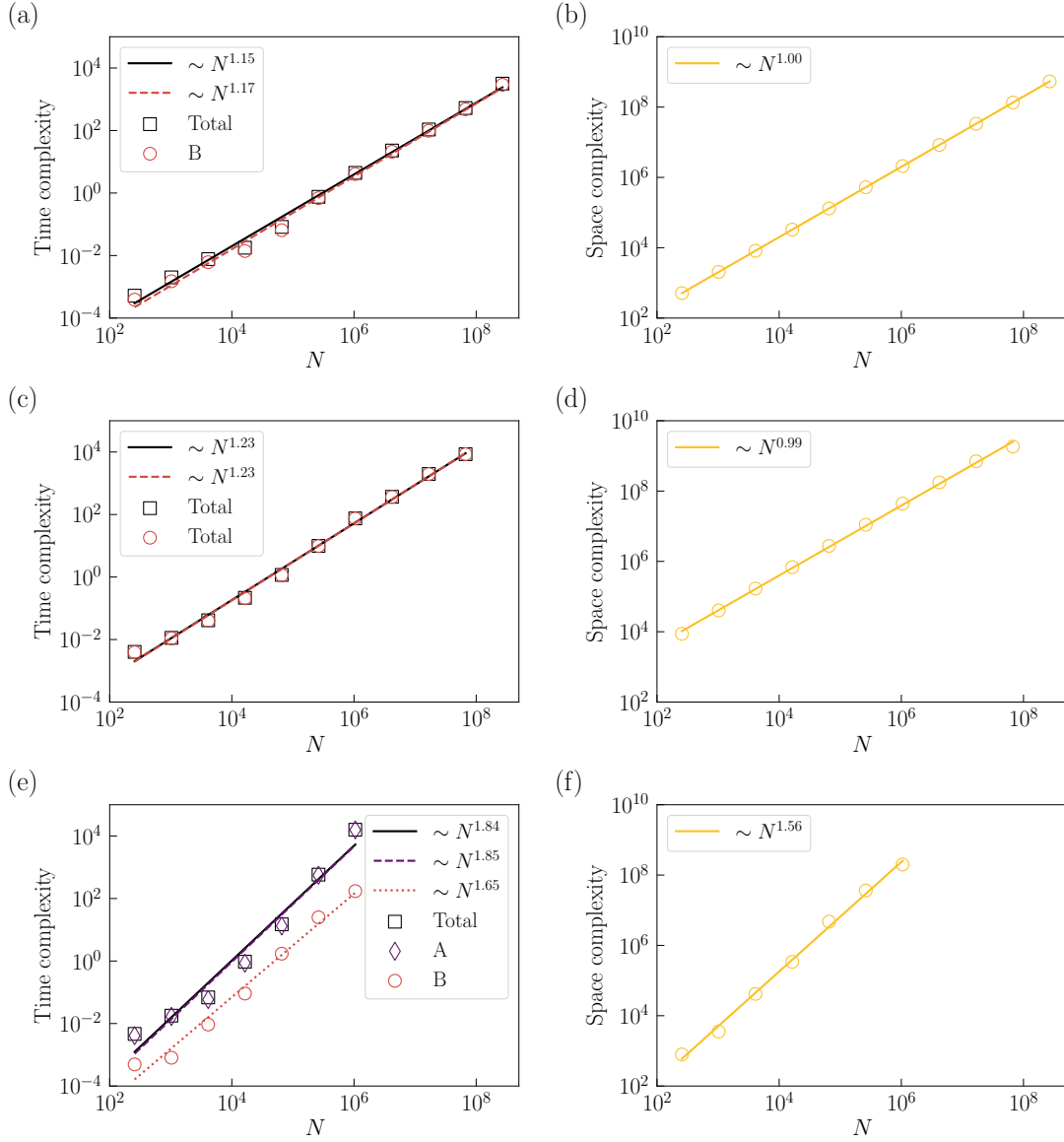


FIG. S11. Algorithmic complexity of the shortest-path-percolation (SPP) model. (a) We plot the amount of time necessary to run a single simulation of the SPP model as a function of the network size N . Here, $C = 1$. We plot separately the time required for point B of the mixed simulation strategy, and the total time required to draw the entire phase diagram. The total time includes all parts of the algorithm required to draw the phase diagram, including the generation of the graph and the Newman-Ziff portion that allows us to compute the percolation strength and the average cluster size. Data points are fitted with power laws to determine the time complexity of the algorithm. Simulations were performed on a Intel(R) Xeon(R) CPU E5-2690 v4 @ 2.60 GHz and time is measured in seconds. (b) We visualize the size (i.e., number of elements) of the set \mathcal{L} used in part B) of the mixed simulation strategy as a function of the network size for the same set of simulations as in panel (a). Data points are fitted with power laws to determine the space complexity of the algorithm. (c) and (d) are the same as panels (a) and (b), respectively, but for $C = 3$. (e) Same as in panels (a) and (c), but for $C = N$. Here, we report also the amount of time required to run part A) of the mixed simulation strategy. (f) Space complexity for the same set of simulations as in (e).