Factors affecting congestion-aware routing in complex networks

Juan Echagüe [a] Vicent Cholvi [b] Antonio Fernández [c]

[a] *Departamento de Ingeniería y Ciencias de los Computadores, Universitat Jaume I, 12071, Castellón (Spain).*

[b] *Departamento de Lenguajes y Sistemas Informáticos, Universitat Jaume I, 12071, Castellón (Spain).*

[c] *IMDEA Networks Institute (Spain).*

Abstract

An important issue in complex networks is to find efficient strategies for information delivery between a given sending node and its destination host. Whereas there is no doubt that the use of the congestion provides significant advantages for routing, it has been observed that there are some factors that greatly influence the behavior of the routing protocols that use it. In this paper we study the effect of two factors that greatly influence the behavior of congestion-aware routing protocols: updating all the paths at the same time and using information from a subset of the nodes to compute the paths. On one hand, we give explanations to these behaviors and, on the other hand, we quantitatively evaluate the effect of these factors on the performance of the routing protocols.

Key words: Complex Networks, Routing Strategies, Congestion Control
PACS: 89.75.Fb, 89.75.Hc, 89.75.Da

1 Introduction

Complex systems play an important role in our daily life, so that their understanding and control is one of the major intellectual and scientific challenges today. Indeed, traffic dynamics on complex networks have attracted much attention in the last years, especially as many real-world networks such as the Internet [14], city transport [18], epidemic spreading [15], management of organizations [19], etc., can be modeled by scale-free networks [3]. Since the main function of a network is for transporting various objects, such as data packets in telecommunication networks, and cars in city transport networks, then being able to guarantee a high traffic capacity is a critical issue to network service providers. This is even more important considering that, without any control, networks can be easily congested, beginning with rapid traffic aggregation on some central nodes and then later rapid spreading to the whole network. Therefore, an issue that has attracted the attention of the research community interested in complex networks is to find efficient strategies for information delivery between a given sending node and its destination host [1,6,13]. To this end, a number of protocols for the dissemination of information (i.e., routing protocols) has been developed in the last few years. Many networks use routing strategies that, either implicitly or explicitly, are weighted in nature. The main idea of such weighted routing is to assign a (possible different) weight to all the links in the network, and then to select a path that minimizes some function that will depend on the weights assigned to the links that form the path. Examples of such strategies are the *shortest path routing* between any two nodes of the network [10], which is implemented by assigning the same weight to all links. Yan et al. provide a routing strategy [20] in which the path between node i and node j corresponds to the route that makes the sum $\psi(\mu) = \sum(k_m)^\mu$ a minimum, where μ is a tunable parameter and k_m is the degree of node k for all the nodes in the path from node i to node j. The rationale behind this approach was to assign bigger weights to well connected nodes, which are expected to be more congested. A similar approach was proposed by Yang et al. [21], in which the weights are assigned taking into account the product degrees of the nodes that form each link (namely, $\sum(k_i k_j)^\theta$, for each pair of adjacent nodes i and j). Furthermore, they show how to determine the optimal value of the parameter θ. In [17], the authors extended the previous degree-based weighting scheme to interdependent networks and demonstrated that there still exists an optimal weighting parameter on interdependent networks, but it might shift as compared to the case in isolated networks. In order to make the costs of paths in networks more comparable, Pu et al. [16] proposed a similar routing strategy in which they used $\log(\log(k_m))$ instead of k_m. As a result, high-degree nodes have more opportunities to be part of the optimal paths to deliver packets, and they found that the traffic load is more evenly distributed.

Unlike the previous strategies, which assign weights based on some properties of the network (and, therefore, the routes between each pair of nodes remain fixed), in [5] the author proposed a routing strategy that selects the paths according to the congestion of nodes in the network. In the proposed algorithm the routes do not remain fixed, but they dynamically change over time to adapt to traffic changes. We say that such routing protocols are *congestion-aware*, since, at each time instant, the election of the paths followed by packet will depend on the congestion of the nodes/links that such a packet will traverse. In [12], the authors consider a congestion-aware strategy in which the routing process is decomposed into N (the network size) steps and, at each step, we compute all paths or calculate the spanning tree for one source node by considering dynamic betweenness centrality and degree information. Now, the goal is to minimize cost function $\sum(B^s_{x_v} D_{x_v})^\beta$ at each step, where $B^s_{x_v}$ is the *dynamic efficient betweenness* of node x_v in the computing process for source node s (which is calculated incrementally in the routing process for all nodes of the network), D_{x_v} is the degree of node x_v, and β is a tunable parameter. The found that traffic capacity can be enhanced by a substantial factor at the cost of a slight lengthening in transition delay. Recently, in [8] the authors introduced a new routing congestion-aware routing protocol that has been shown that provides a tolerance to *collapse* close to the optimal and short paths (the optimality of the protocol was obtained by comparing it against the global optimization performed by Danila et al. [7], which systematically adjusts the traffic routing to minimize the maximum congestion on the network).

Our contributions

Whereas there is no doubt that congestion-aware routing provides significant advantages (see the references presented above), we have identified two factors that can greatly influence the behavior of the routing protocols that use it. Namely:

(1) Updating all the paths at the same time: contrary to what one would expect, we have found that when all the paths that packets follow are calculated immediately (by using the current congestion of nodes), this causes a significant decrease in the collapse points.
(2) Using information from a subset of the nodes: although it seems clear that the more that is known about the congestion of the nodes the better the protocol will work, we have found that no relevant benefits are obtained unless the known nodes are chosen in a smart way; otherwise, the obtained benefits are marginal. This also applies when the protocol loses information from a subset of nodes: a few nodes, chosen in a smart way, could provoke a big degradation, which doesn't happens if they are chosen in a random way.

For these factors and by means of a series of experiments, we quantitatively evaluate how they affect the performance of the routing protocols and give explanations of the reasons for such behaviors. At this point, we would like to mention that, rather than proposing a new routing protocol, our goal in this paper is to look into some factors that can affect the behavior of all congestion-aware routing protocols (i.e., those that take into account the current congestion of the nodes to decide the route followed by packets). Therefore, to a greater or lesser extent, these factors affect all of them.

In Section 2, we describe both the system model and the congestion-aware routing protocol. A performance evaluation is carried out in Section 3, where we also analyze the reasons for the results obtained. In Section 4, we end the paper with some conclusions.

2 Model

We consider a system model similar to the one used in [2,11]. Such a system is formed by a network that contains a set of nodes/vertices interconnected by means of links/edges. In such a scenario, the information flowing through the network is formed by discrete packets sent from a source node to a destination node. Each node has a double function: (i) it acts as *source* and *sink* of packets, and (ii) it also acts as a *router* that receives packets from other nodes, selects output links according to a routing strategy, and forwards the packets through these links.

In the system, nodes are independent agents that have a limited capacity to process and forward packets, independently of their load (i.e., the volume of arriving packets). Such a limitation can result in packets to be stored at nodes, pending to be processed. We assume that nodes can store as many packets as necessary. We also assume that time is slotted, and that in each time slot a node can generate and send at most one packet.

In order to regulate the load of the system, we bound probabilistically the number of packets that nodes can generate with a parameter ρ. Hence, we consider that at each time t, a packet is created at every node with probability $0 \leq \rho \leq 1$: small values of ρ correspond to low density of packets and high values of ρ correspond to high density of packets. When a packet is created at a node, a destination node is chosen uniformly at random. Then, a path in the network from the node to the destination node is selected by the routing algorithm, and the packet is sent over this path. Once the packet reaches the destination node, it is delivered and disappears from the network.

Assuming the processing power (i.e., the capacity) of nodes to be one packet per unit of time, we have that node i *collapses* when the number of packets to be processed at node i exceeds, on average over time, the rate of one per unit of time. We say that the network *collapses* when at least one node collapses; otherwise, we say that the network is *fluid*. The value of ρ at which the network collapses is called its *collapse point* (denoted as ρ_c).

A *commodity* from node a to node b refers to the set of packets generated at source node a, that must travel to the destination node b. These packets will follow an associated *path* (the links along which these packets will be sent), which is computed by the routing algorithm and may vary over time. In our work, we will refer to the pair (a, b) as the *route* of the mentioned commodity, and the links along which the packets of the commodity are sent as the current path of the commodity route.

The routing protocol

An important challenge of routing is how to articulate control and routing strategies to prevent complex networks from collapsing. The routing protocol we consider here is based on a version of the weighted shortest path algorithm introduced in [8], called *congestion-aware-weights* routing protocol (CAW). Note that CAW is a loop-free routing protocol, so that each packet passes through the same node only once. The key aspect of the CAW protocol is how weights are assigned to the different links. Let us define the *congestion* c_i^t of node i at time t, denoted c_i^t, as the number of packets that arrive at node i at time t. Based on the current value of c_i^t, we assign a weight w_i^t to node i, as follows (initially w_i^t is set to 1):

$$w_i^t = (1 + c_i^t)^\gamma, \tag{1}$$

where γ is a non-negative real value. Observe that w_i^t grows with c_i^t and its increment depends on the value of γ. We define the weight of a link (i, j), denoted as w_{ij}^t, as the maximum of the two values: w_i^t and w_j^t.

The routing protocol works as follows:

(1) For each commodity from node i to j, the path that the packets follow is calculated at i by using the weighted shortest path algorithm with the weights of links assigned by using Equation 1.
(2) Each node, at given time intervals, communicates its congestion to the rest of nodes in the network.
(3) When a node i receives all the congestion values from the rest of the nodes, it updates the route of one or more of its commodities as follows. At time 0, node i generates a random permutation π of all nodes j in the network to which i will send packets. Then, i updates the commodity routes following this order π in round robin fashion.

We note that the parameter γ in Equation 1 can be seen as a parameter that characterizes the "selfishness" of the routes: when $\gamma = 0$ we have the classical *shortest path routing*, and the paths are the shortest ones, regardless of the resulting collapse point. In turn, when γ increases, routes become increasingly *altruistic*, in the sense that in order to decrease the collapse point (by avoiding congested nodes), they accept to increase the length of their paths. In [8], the authors presented an algorithm to assign the values of γ in a dynamic manner: in order to minimize the length of the paths, the algorithm assigns small values for low traffic, and for heavy traffic the algorithm provides bigger values, which increases the tolerance to collapse. By using that value assignment algorithm, the resulting routing protocol was shown to provide a tolerance to collapse close to the optimal value ρ_c, with a reasonable average length of the paths. Indeed, on one hand, its tolerance to collapse was shown to be close to the solution provided by Danila et al. in [7], which consisted of using a heuristic algorithm that systematically adjusts the traffic routing to minimize the maximum congestion on the network (although it needs to be performed beforehand, for each network); on the other hand, it was shown that the average length of the paths that grows in a linear fashion with the increase of ρ.

However, in our work we maintain the value γ fixed, which will allow us to analyze, more clearly, how the system evolves over time. More specifically, we take $\gamma = 7$, since it has been shown in [8] (see Figure 1(a)) that it provides a good balance between high collapse points (90% of the collapse point) and short paths. Nevertheless, we would like to note that the observed behaviors take place regardless of the values of γ, except for the case where $\gamma = 0$ (i.e., when all the weight are the same).

At this point, we would like to remark that, rather than proposing a new routing protocol, our goal is to look into some factors that can greatly affect the behavior of congestion-aware routing protocols.

Congestion and effective betweenness

It has been shown in previous works [11] that in the considered model the number of packets that arrive at a node i at each time instant is, on average,

$$c_i^t = \frac{\rho \, B_i^t}{n-1}, \tag{2}$$

where B_i^t is the *effective betweenness* of node i at time t, defined as the number of routes whose paths pass through node i at time t, and n is the number of nodes in the system. Observe that the knowledge of the effective betweenness of a node allows us to identify when such a node is collapsed (i.e., when c_i^t exceeds, on average over time, the rate of one per unit of time).

Taking this into account, our simulations are based on calculating the effective betweenness and then obtaining the value of the congestion of each node using Equation 2, rather than making nodes to inject packets into the network and measuring the congestion of each node. That approach allows us to perform the simulations without having to model all the details. At this point, we note that this approach has been followed by a number of authors in the past [11,2,7,5,6,8], and it has been shown to provide accurate results.

3 Using congestion awareness for routing

Whereas there is no doubt that the use of congestion (i.e., the current congestion at each node) provides advantages for routing, here we will show that, in order to obtain the maximum possible profit from it, it is necessary to carefully select which information must be taken into account, and when.

In this section, we consider two factors that we have found have a great influence on the behavior of the routing protocol. Namely, the number of routes updated at the same time and the use of information from a limited number of nodes. We analyze the performance of the routing strategy by means of a series of experiments carried out by using simulations.

In our experiments, we consider both scale-free [3] and random networks [4] formed of $n = 256$ (i.e., the number of nodes) with $m = 3$ (i.e., the number of outgoing edges), and assume that all nodes communicate with each other (i.e., there is a path for each pair of nodes). At this point, we note that, although the results are similar in all scale networks, we have also considered other values of n and m. As expected, the shape of the curves are the same; the only thing that changes are the points at which the system collapses: the more nodes the lower the collapse point, and the more links the greather the collapse point.

Furthermore, ρ is a control parameter that, since we want to be able to inject as many packets as possible, needs to be as large as possible. Therefore, for any given value of ρ, the best strategy with respect to collapse of the network would be the one that minimizes, over all nodes, the maximum value of B_i^t (and consequently, the maximum value of c_i^t).

In each simulation, we increase the value of ρ by 0.001 (initially 0) when all possible routes in the network have been updated using that ρ (we call it, a *round*). This process is repeated until the network collapses. We have performed 100 simulations for each experiment, averaging values.

3.1 On updating the paths of several commodities at the same time

A behavior that we observed when using congestion-aware routing is that, when all the paths that packets follow are calculated immediately using the current congestion of nodes, this caused a significant decrease in the collapse points. That seems contrary to what one would expect since paths are chosen

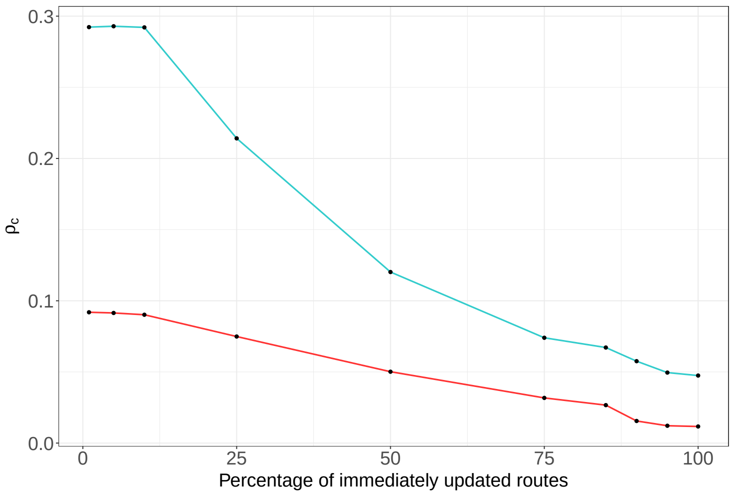

Fig. 1. Experiment that shows how the collapse point varies with the percentage of paths updated at each step. The blue line corresponds to random networks and the red line to scale-free networks.

in order to avoid currently congested nodes, which should result in an increase in the collapse points.

Therefore and in order to quantitatively evaluate the effect of using immediately the current congestion of nodes, in our first experiment we vary the percentage of routes (starting at each node) that immediately update their paths by using the most recent information regarding the congestion of nodes. The rest of the routes also update the paths of their routes by using the congestion of nodes, but not immediately; namely, each node only updates one route at each step. For each scenario, we measure the collapse points.

In Fig. 1, it can be seen that there is an important decrease on the values of such congestion points when we increase the percentage of immediately updated routes, which confirms our previous observations.

In order to provide an explanation to this counter-intuitive behavior, in Fig. 2 we introduce an illustrative scenario in which the set of source nodes in subnetwork $\mathcal{S}_1$ are assumed to send packets to node d_1 and the set of source nodes in subnetwork $\mathcal{S}_2$ are assumed to send packets to node d_2. Blue dashed lines denote the set of links (one for each node) that go form nodes in $\mathcal{S}_1$ and $\mathcal{S}_2$ toward nodes f_1 and f_2. Let us respectively denote as $\mid \mathcal{S}_1 \mid$ and $\mid \mathcal{S}_2 \mid$ the number of nodes in $\mathcal{S}_1$ and $\mathcal{S}_2$ that send packets to nodes d_1 and d_2. Let us assume that $\mid \mathcal{S}_1 \mid = \mid \mathcal{S}_2 \mid -1$, and that collapse occurs when the betweenness of some node is $2 \cdot \mid \mathcal{S}_1 \mid$. It is easy to see that, choosing the paths carefully, the system could work in a fluid fashion (e.g., routes from $\mathcal{S}_1$ choose node f_1, and routes from $\mathcal{S}_2$ choose node f_2). However, if all nodes in $\mathcal{S}_1$ and $\mathcal{S}_2$ chose initially routing via f_1, and they change their paths at the same time, then it is easy to see that nodes f_1 and f_2 will alternatively become fluid and

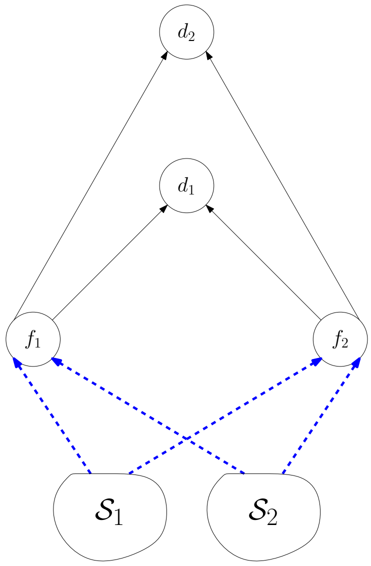

Fig. 2. Scenario used to illustrate how immediately using the current congestion of nodes to calculate the paths may cause a decrease in the performance of the routing protocol.

collapsed at each step: at a given step, all routes will choose the node with the smallest betweenness, which will become collapsed; then, at the subsequent step and by the same reason, the other node will become collapsed. That will occur in a cyclic fashion, which will make the overall system collapsed all the time. On the contrary, if the routes paths are updated in a gradual way (i.e., one route at each step), that allows the routing protocol to adapt to the current congestion conditions in a smoothly manner, therefore avoiding the above mentioned harmful behavior.

We note that this counter-intuitive behavior has been observed in other scenarios, such as in games with strict strategic complementarities, which have shown to be unstable for a broad class of learning dynamics [9].

3.2 On using information from a subset of the nodes

In this section, we study how the knowledge on the congestion of a subset of nodes of the networks affects the performance of the routing protocol. In order to do this, it is necessary to determine how to choose the nodes in that subset (which we will refer as *known nodes*). Here, we consider three approaches: the first one consists of choosing the nodes with higher *betweenness centrality* (i.e., defined as the number of routes whose paths pass through a node by using shortest paths), the second one is simply the inverse of the first one (i.e., choosing the nodes with lower betweenness centrality), and the third one consists of choosing the nodes in a random fashion.

The first and second approaches have been chosen since they provide the best/worst results in terms of higher and lower collapse points, while the third one was chosen in order to show the effect of an election of nodes that is oblivious to the collapse points. As for the congestion of the nodes not included in the above mentioned subset, it is assumed to be the average congestion, at each time, of all the known nodes.

Similar to the previous experiment, here we vary the percentage of nodes whose congestion is known, and for each scenario, we measure the collapse points. The obtained results are shown in Fig. 3. It can be seen that there is a big difference on the values obtained for the collapse points depending on how the election of the known nodes is performed.

The best results are obtained when these nodes are those with the highest betweenness centrality. This is not surprising since the nodes with higher betweenness centrality are the ones that will have the most congestion, thus driving the system to collapse. Therefore, it seems evident that if the routing protocol can avoid, as much as possible, such congested nodes, then this will increase the collapse points. Furthermore, it can also be seen that the collapse points curve grow faster when considering scale-free networks than with random ones. The reason for such behavior is due to the fact that scale-free networks are characterized by having a small number of nodes with very high betweenness centrality (i.e., high congestion), whereas in random networks the betweenness centrality of nodes is more uniformly distributed (and smaller). So, the effect of acting on these nodes is greater on scale-free than on random networks.

Regarding the other two other approaches (i.e., choosing the known nodes at random and these with lower betweenness centrality), it can be seen that the increase of the collapse points is marginal. Whereas we foresaw a lower improvement, we found that it was unexpectedly small. In case of choosing as known nodes those with lower betweenness centrality no growth was observed in the collapse point curve until more than 95% of the nodes were known, and this percentage is only slightly better when the known nodes are chosen at random (for instance, until the congestion of 75%-85% of the nodes is known, no significant increase in the collapse points was observed).

These results confirm our previous observations regarding the fact that the number of known nodes plays an important role on the performance of the routing protocol. However, we have found that no relevant benefits are obtained unless the known nodes are chosen in a *smart* way; otherwise, the obtained benefits are marginal.

The study of how the knowledge on the congestion of a subset of nodes of the networks affects the performance of the routing protocol can also be analyzed

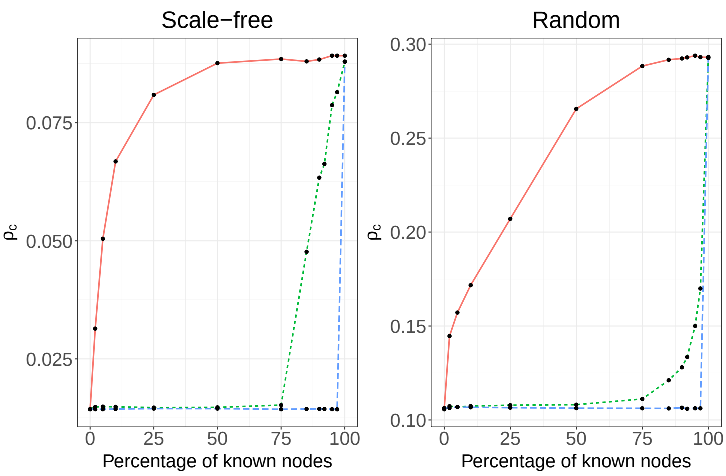

Fig. 3. Collapse points vs percentage of known nodes. Red lines represent the case where nodes are chosen in accordance with their greatest betweenness centrality, lines in blue in accordance with their smallest betweenness centrality and lines in green when they are chosen at random.

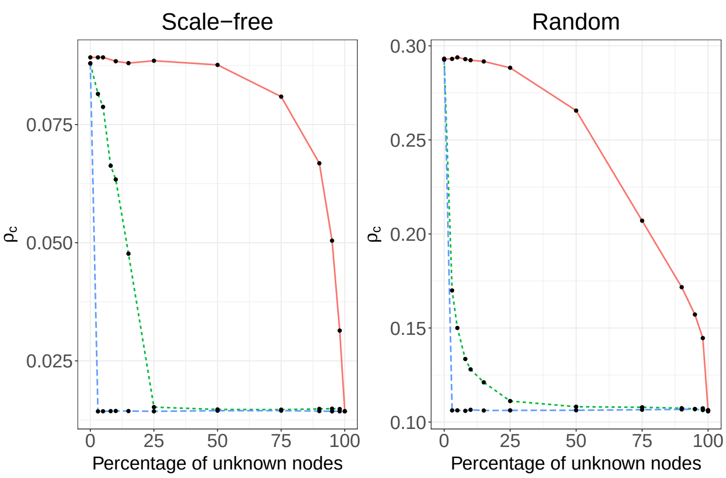

Fig. 4. Collapse points vs percentage of unknown nodes. Red lines represent the case where nodes are chosen in accordance with their smallest betweenness centrality, lines in blue in accordance with their greatest betweenness centrality and lines in green when they are chosen at random.

from a different point of view. Namely, when the routing protocol, who is assumed to use the congestion of all nodes, loses information from a subset of them. That could happen for several reasons: technical problems, sabotages, etc. For this, we redraw Fig. 3 as in Fig. 4, so that we start from a percentage of 0% of unknown nodes towards 100% of them.

Now, we see that the best results are obtained when the unknown nodes are chosen based on the lowest betweenness centrality, and the worst results when are chosen based on the highest betweenness centrality; the results by choosing nodes at random is in between, although closest to the case where they are chosen based on the highest betweenness centrality. Quantitatively, by taking the nodes with the highest betweenness centrality the collapse points decrease to the minimum just with less of 5% of unknown nodes, with slightly better values by taking nodes at random. In turn, by taking the nodes with the lowest betweenness centrality the values are much better: for instance, scale-free networks can tolerate up to 50% of unknown nodes without any significant degradation, and with a degradations smaller than 20% with random networks.

These results also confirm that the manner in which unknown nodes are chosen plays an important role on the performance of the routing protocol. Namely, the decrease in the collapse points ranges from a scenario where there is a relevant degradation only when most of the nodes are unknown (i.e., when they are chosen based on the lowest betweenness centrality), to a scenario where a few nodes provoke a big degradation (i.e., when they are chosen either at random or based on the highest betweenness centrality).

4 Conclusion

In summary, in this paper we have studied the effect of two factors that greatly influence the behavior of congestion-aware routing protocols. On one hand, we have given explanations to these behaviors and, on the other hand, we have quantitatively evaluated the effect of these factors on the performance of the routing protocols.

Acknowledgement

This work has been partially funded by the Spanish Ministry of Science and Innovation grant PID2019-109805RB-I00 (ECID) cofunded by FEDER.

References

[1] Reka Albert and A.-L. Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74:47–97, 2002.

[2] A. Arenas, A. Díaz-Guilera, and R. Guimerà. Communication in networks with hierarchical branching. *Phys. Rev. Lett.*, 86:3196–3199, Apr 2001.

[3] Albert-Laszlo Barabasi and Reka Albert. Emergence of scaling in random networks. *Science*, 286:509–512, October 1999.

[4] B. Bollobas. *Random Graphs.* Academic Press, 1985.

[5] Vicent Cholvi. Dissemination of information in complex networks with congestion. *Physica A*, 366(1):571–577, 2006.

[6] Vicent Cholvi, Vctor Laderas, Luis López, and Antonio Fernández. Self–adapting network topologies in congested scenarios. *Physical Review E*, 71(3):035103, 2005.

[7] Bogdan Danila, Yong Yu, John A Marsh, and Kevin E Bassler. Optimal transport on complex networks. *Physical review. E, Statistical, nonlinear, and soft matter physics*, 74 4 Pt 2:046106, 2006.

[8] Juan Echage, Vicent Cholvi, and Dariusz R. Kowalski. Effective use of congestion in complex networks. *Physica A: Statistical Mechanics and its Applications*, 494:574 – 580, 2018.

[9] Federico Echenique and Aaron Edlin. Mixed equilibria are unstable in games of strategic complements. *J. Economic Theory*, 118(1):61–79, 2004.

[10] Bernard Fortz and Mikkel Thorup. Optimizing OSPF/IS-IS weights in a changing world. *IEEE Journal on Selected Areas in Communications*, 20(4):756–767, 2002.

[11] R. Guimerà, A. Díaz-Guilera, F. Vega-Redondo, A. Cabrales, and A. Arenas. Optimal network topologies for local search with congestion. *Physical Review Letters*, 89:248701, 2002.

[12] Zhong-Yuan Jiang and Man-Gui Liang. Incremental routing strategy on scale-free networks. *Physica A: Statistical Mechanics and its Applications*, 392(8):1894–1901, 2013.

[13] M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45:167–256, 2003.

[14] R. Pastor-Satorras and A. Vespignani. *Evolution and Structure of the Internet.* Cambridge University Press, 2004.

[15] Romualdo Pastor-Satorras and Alessandro Vespignani. Epidemic spreading in scale-free networks. *Phys. Rev. Lett.*, 86:3200–3203, Apr 2001.

[16] Cun-Lai Pu, Si-Yuan Zhou, Kai Wang, Yi-Feng Zhang, and Wen-Jiang Pei. Efficient and robust routing on scale-free networks. *Physica A: Statistical Mechanics and its Applications*, 391(3):866 – 871, 2012.

[17] Yuzhuo Qiu. Optimal weighting scheme and the role of coupling strength against load failures in degree-based weighted interdependent networks. *Physica A: Statistical Mechanics and its Applications*, 392(8):1920–1924, 2013.

[18] Jianjun Wu, Ziyou Gao, Huijun Sun, and Hai-Jun Huang. Urban transit system as a scale-free network. *Modern Physics Letters B*, 18:1043–1049, 01 2012.

[19] Lynn Wu, Benjamin N. Waber, Sinan Aral, Erik Brynjolfsson, and Alex Pentland. Mining face-to-face interaction networks using sociometric badges: Predicting productivity in an IT configuration task. In *Proceedings of the International Conference on Information Systems, ICIS 2008, Paris, France, December 14-17, 2008*, page 127. Association for Information Systems, 2008.

[20] Gang Yan, Tao Zhou, Bo Hu, Zhong-Qian Fu, and Bing-Hong Wang. Efficient routing on complex networks. *Physical Review E*, 73(046108), 2006.

[21] Rui Yang, Wen-Xu Wang, Ying-Cheng Lai, and Guanrong Chen. Optimal weighting scheme for suppressing cascades and traffic congestion in complex networks. *Physical Review E*, 79(026112), 2009.