Effect of Degree-Preserving, Assortative Rewiring on OSPF Router Configuration

Rogier Noldus
Ericsson Eurolab Deutschland
Herzogenrath, Germany
rogier.noldus@ericsson.com

Piet Van Mieghem
Delft University of Technology
Delft, The Netherlands
P.F.A.VanMieghem@tudelft.nl

***Abstract*—We study the impact of degree-preserving rewiring on the routing table configuration in an OSPF network. Degree-preserving rewiring may be applied to optimize specific network characteristics, such as assortativity.**

Rewiring has significant impact on IP router tables. This manifests itself through the number of nodes that are affected by the rewiring and the average number of routing table entries that are modified per affected node. Networks with high link density are generally less impacted. Through the use of Software Defined Networks (SDN), a number of rewiring steps can be aggregated, leading to fewer routing table updates in total. For aggregation of 15 - 20 rewiring steps, depending on network class and characteristics, almost all nodes in the network are affected.

Keywords—OSPF, assortativity, rewiring, degree-preserving, network

I. INTRODUCTION

Rewiring can be applied for improving certain network characteristics, whereby the network is represented as a graph $G(N,L)$ with N i nodes and L links. One objective may be to increase the assortativity of the network or to reduce the effective graph resistance. Increasing the assortativity influences the spectral radius λ_1 (largest eigenvalue of adjacency matrix of graph G) or the algebraic connectivity $\mu_{N\text{-}1}$ (second smallest eigenvalue of Lapacian matrix of G) of the graph [7], [10]. Degree-preserving rewiring implies that two links are rewired such that the degree of the involved nodes remains constant. When links are rewired, the topology of the network changes. Changes in topology are detected by the routers in that network. Routers in OSPF[1] will update their link-state, shortest path tree and routing tables. Rewiring in one area of the network may affect routing tables in a large portion of the network.

Network rewiring may be applied within Software Defined Networks (SDN)[2]. SDN implies that routing tables in IP routers in a network are controlled through centralized control logic. Rewiring is not the only means of improving network characteristics through topology changes. Another way of changing the topology, and improving designated network characteristics, is link addition (between two routers). Network rewiring has the advantage that it may be done with existing cabling infrastructure, for IP routers in physical proximity.

We apply degree-preserving rewiring on a set of simulated networks. We study the number of affected nodes (for which at least one routing table entry is changed) per rewiring step, as well as the average number of affected entries per affected node, per rewiring step. We also monitor the accumulative impact from an aggregate series of rewirings.

The effect from rewiring in one part of the network may ripple through the entire network. The spread of the effect, i.e. how far the effect from rewiring reaches, depends on aspects such as the class of network, the size of the network, the link density, etc.

II. REWIRING

The rationale of rewiring a network whilst keeping the *degree* of the nodes constant, is that the degree distribution is an essential network characteristic. Van Mieghem *et al.* [7] devise a method to increase assortativity through degree-preserving rewiring and show that the degree assortativity ρ_D of a graph is directly related to the spectral radius λ_1 and the algebraic connectivity $\mu_{N\text{-}1}$. The λ_1 and $\mu_{N\text{-}1}$ characterize a graph's vulnerability to processes such as virus spreading. Van Mieghem *et al.* rewrite the assortativity formula, originally defined by Newman [5], to derive a degree-preserving rewiring algorithm. This algorithm can be used to influence ρ_D and hence λ_1 and $\mu_{N\text{-}1}$ (and the network's dynamics). The formula for assortativity is rewritten as:

$$\rho_D = \frac{N_1 N_3 - N_2^{\,2}}{N_1 \sum_{i=1}^{N} d_i^{\,3} - N_2^{\,2}} \tag{1}$$

where N_k is the total number of walks in G with k hops.

Whilst the set of node degrees remains constant, increasing ρ_D will lead to a decrease of $\mu_{N\text{-}1}$. The ρ_D of a graph can be increased or decreased, as required, through degree-preserving rewiring [7]. Increasing λ_1 through increased assortativity, may however lead to graph disconnectivity. Increasing $\mu_{N\text{-}1}$ through decreased assortativity, however, leads to improved topological robustness of the graph. This observation provides us a tool to

[1] IETF RFC 2328; Open Shortest Path First (OSPF); 1998
[2] Open Network Foundation; www.opennetworking.org/

change the properties of a network, through iterative, degree-preserving rewiring. Whilst we may affect the assortativity of a graph for topological robustness, other graph metrics should also be considered. For example, a rewired graph may exhibit improved topological robustness, but the topological robustness may decrease more rapidly under random or targeted attack (on a node or on a link) than for the original graph.

Increasing ρ_D through degree-preserving rewiring may increase a network's modularity. Modularity reflects how strongly the network is constructed from 'communities' with dense intra-community connections and sparse inter-community connections. Van Mieghem *et al.* [8] also study the relation between assortativity and average hop count $E[H]$. For Barabási-Albert (BA) [1] graphs as well as for Erdős-Rényi (ER) [3] graphs, increasing assortativity above 0 leads to increase of $E[H]$. It is also shown that when ρ_D for such graph reaches its minimum value, $E[H]$ also increases, but only slightly. A larger assortativity of a network leads to a faster information spread through the network [9]. Since larger ρ_D results in higher $E[H]$, one would expect information (including viruses!) to spread more slowly. A large assortativity, with high-degree nodes being well connected, has the effect that information spreads easily across different regions of the network, but will take longer to reach low-degree nodes. It is, furthermore, shown in [8] that the increase in ρ_D results in a steady increase in effective graph resistance R_G with a sharp rise of R_G when ρ_D reaches its maximum value for the class of graphs. The full paper[3] shows graphical examples of the relation between R_G and ρ_D.

A. Degree-preserving rewiring

Degree preserving rewiring entails that the network is rewired such that the degree vector $d^T = [d_1, d_2, d_3, \dots , d_N]$ is preserved. Rewiring, generally, means that a link is moved, at one of its end points, from one node to another node. This obviously affects d^T; for one node the degree will decrease by 1, for another node the degree will increase by 1. For preserving the degree vector, a pair of links is rewired. The rewiring is done by swapping the end points of two links (Figure 1).

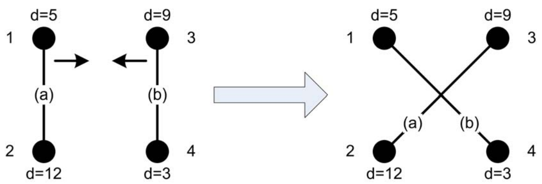

Figure 1: Degree-preserving rewiring

The degrees of the nodes are not affected. Nodes 2 and 3 both become more assortative, as they are now connected to other nodes that have degree closer to their own degree. Nodes 1 and 4 also become more assortative, being rewired to other nodes with degree closer to their own degree.

[3] http://www.nas.its.tudelft.nl/index.php/people/rogier-noldus

Through successive degree-preserving rewiring, we explore the ensemble of graphs with the same degree sequence as G. This set of graphs is denoted as $\mathscr{G}(G)$. Any rewired graph of G is denoted as $G' \in \mathscr{G}(G)$. Finding the complete set $\mathscr{G}(G)$ is hard. It is bounded by a theoretical maximum of the binomial of $(N*(N\text{-}1)/2 , L)$. Degree-preserving rewiring further reduces the upper bound. According to Bienstock and Günlük [2], any G' can be obtained through a finite set of degree-preserving rewiring steps. Holme and Zhao [6] have studied $\mathscr{G}(G)$, to obtain the minimum and maximum assortativity values for $G' \in \mathscr{G}(G)$.

For each rewiring goal, different strategies may be devised, in order to let the network topology converge in as few rewiring steps as possible towards the desired state. The rewiring strategy should also find a suitable link pair in as few attempts as possible. Winterbach *et al.* [11] as well as Trajanovski *et al.* [6] outline methodology for exhaustive rewiring, studying whether a greedy approach yields optimum assortativity for a network within reasonable time. When rewiring is applied on large networks (N>1000), an optimized algorithm for finding rewirable links becomes crucial in order to curb computation time.

III. Network Rewiring Simulation

For different classes of networks, degree-preserving, assortative rewiring is performed. Rewiring continues until no further rewiring is possible that increases assortativity or until a defined maximum number of rewirings is reached. We apply the following assumptions: (i) links have unit weight, (ii) links are bi-directional and (iii) a single shortest path is determined for each node pair (the last shortest path found when no shorter paths can be found for this link pair); routing tables have, for each destination, a single next hop only.

After each rewiring step on a graph, we recalculate all shortest paths and the routing table for each node. We consider a node in the network to be *affected* when at least one entry in the routing table of that node is changed due to the rewiring.

A. Rewiring algorithm

We follow the principle described in Lemma 1 of Van Mieghem *et al.* [7], but apply a targeted selection of link pairs, in order to quickly find a link pair that is (a) rewirable and that (b) will lead to an increase of assortativity. We extrapolate the approach depicted in Figure 1 into a rewiring algorithm. This algorithm is explained in the full paper.

A link pair is classified as 'rewirable' when: (i) links do not have a node in common, and (ii) rewiring does not lead to multiple links between one node pair, and (iii) rewiring does not result in network disconnection.

B. Effect from rewiring

For the simulated networks, we analyse specifically the following information:

- Routing table (next hop for each destination IP address);

- Routing table update (indicating which entries in the routing table have changed);
- List of affected nodes, indicating per affected node the average hop count between that node and the rewired links;
- List of affected nodes, indicating per affected node the number of affected entries.

IV. TEST RESULTS AND EVALUATION

We perform tests on graphs of the following classes:

- Erdős-Rényi random graph (ER);
- Barabási-Albert scale-free graph (BA).

The ER graphs are generated with $50 \leq N \leq 1500$ and $0.01 \leq p \leq 0.50$. The BA graphs are generated by starting from a complete network with $1 \leq N_0 \leq 10$, target network size $50 \leq N \leq 1500$ and link attachment rate $1 \leq n \leq 5 \mid n \leq N_0$.

The tests are divided in two test groups (A and B). We provide test results for a selected number of graph classes, size and link probability. A more elaborate set of test results for both test groups, is contained in the full paper.

A. Iterative rewiring

We apply iterative rewiring on a simulated network and record the impact on the routing tables per rewiring step.

A1. Number of affected nodes

We consider ER graphs with N=250, $0.05 \leq p \leq 0.50$. We study the number of affected nodes per iterative rewiring step. Simulations are repeated 100 times on networks with the same N and p.

Figure 3 shows the number of affected nodes per rewiring step, for N=250 and p=0.05. The figure shows results over all simulation runs, divided in 10%-quantiles. The *quantiles* represent the (empirically determined) probability that for particular rewiring step (X-axis), the number of affected nodes (Y-axis) is in certain value range ('band'). The *Average* represents the average number of affected nodes, for certain rewiring step, when calculated over all simulations.

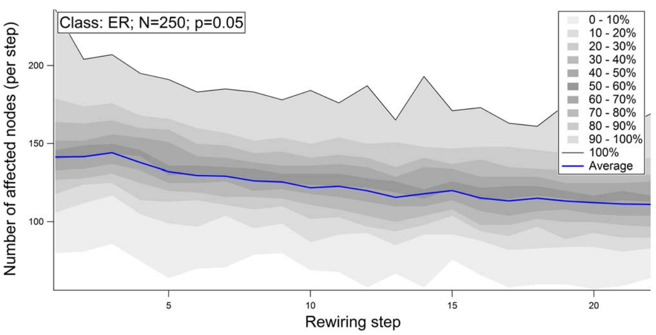

Figure 3: Rewiring for ER graph, N=250, p=0.05 (number of affected nodes per individual rewiring step)

Due to low p (p=0.05), the average hop count of the shortest path is high. This has the effect that a rewiring action, involving four nodes, affects a large number of shortest paths in the network. Hence, the number of nodes of which the routing table is impacted by the rewiring is large. With increasing p we observe through testing that fewer nodes, and on average fewer routing table entries per node, are affected. This is attributed to the fact that large p generally leads to small average hop count of the shortest path in the network. More nodes have direct connection with each other. In addition, smaller average shortest path means that fewer nodes are, on average, traversed for the shortest path between any two randomly chosen nodes i and j. Rewiring has therefore smaller chance of affecting the shortest path between two randomly chosen nodes.

A2. Number of affected entries per affected node

We consider the same ER graphs as in section A1. This time, we study the number of *affected entries* in the routing table of an *affected node*. We determine the average number of affected entries in the affected nodes. Figure 4 shows results for p=0.25, reflecting 10%-quantiles.

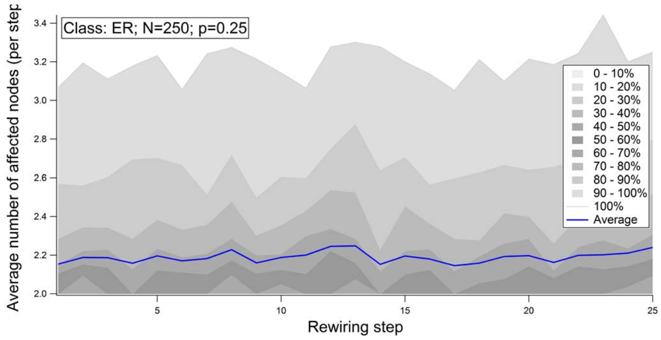

Figure 4: Rewiring for ER graph, N=250, p=0.25 (average number of affected entries per affected node per rewiring step)

We observe that the number of affected entries per affected node is small (for p=0.25, it is less than 1%). With increasing p, the effect of the network rewiring, both in terms of number of affected nodes and in terms of average number of affected entries per affected node, decreases.

A3. Spread of the impact

We consider BA graphs with $100 \leq N \leq 1500$. We study the spread of the impact from rewiring. We determine the average hop count of the affected node to the rewired links. Hereto, we take the average hop count, $< l(i\text{-}j)$, between the affected node and the respective nodes involved in the rewiring (Figure 5). Hereby, l represents hop count, i denotes a node involved in the rewiring and j denotes the affected node.

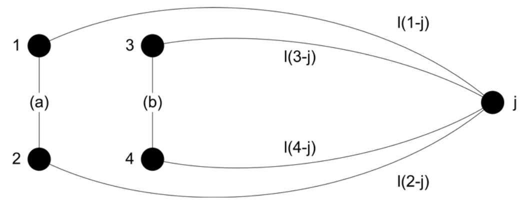

Figure 5: Average distance of affected node to the rewired links

Figure 6 shows average distance of affected nodes, for each rewiring step. We consider a BA network, with N=1500.

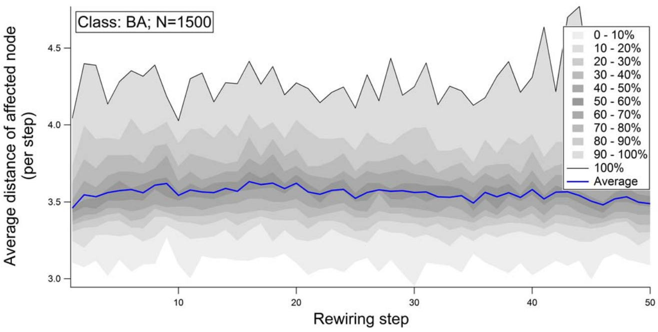

Figure 6: Average distance of affected node, BA graph, N=1500 (average hop count per rewiring step)

With varying N, average distance between affected nodes and rewired links increases with N, but not proportionally.

B. *Aggregate effect from rewiring*

We apply rewiring iteratively to obtain a maximum assortativity. The impact on the routing table is determined over the aggregate set of rewirings. A single routing table update action is applied after these n rewiring steps. This reflects the change in topology of the network resulting from the n rewiring steps.

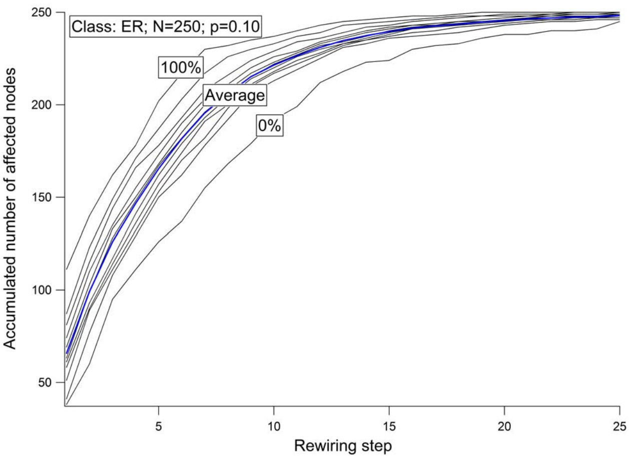

Figure 7: Accumulated number of affected nodes

Figure 7 shows the accumulated effect when an ER graph with N=250 and p=0.10 is successively rewired. The simulations are repeated 50 times. For every rewiring step, the accumulated number of affected nodes is reflected (average and 10%-quantiles). On average, 15 rewiring steps have the effect that almost all nodes in the network are affected. The accumulated average number of affected entries per affected node increases also, as the rewiring continues; refer Figure 8 (full paper). The *quantiles* show the probability that the Accumulated average number of affected entries per affected node (Figure 7) or the Accumulated number of affected nodes (Figure 8) is below certain value, after n rewiring steps. The *Average* reflects the average calculated over all simulations.

Figure 9 and Figure 10 (full paper) show results for a network of class BA, N=1000, k=2 (links preferentially attached for each added node). The simulations are repeated 100 times. Average and 10%-quantiles are provided.

We notice again that, on average, after small number of rewiring steps (appr. 20), all nodes in the network are affected. The average number of affected entries per affected node, calculated over all test runs, rises almost linearly.

V. CONCLUSIONS

Rewiring in a particular part of the network generally leads to updating the routing tables in a large portion of the other nodes in the network. The number of affected nodes depends on the size of the network and on the class and characteristics of the network, like link density. With increasing link density, the number of affected nodes generally decreases. The number of affected entries per affected node is generally low, even for large networks. In addition, the number of affected entries per affected node decreases with increasing p (for ER graphs). The average hop count between affected nodes and the rewired links increases gradually with increasing network size. When considering the aggregate impact from a set of successive rewiring steps, a relative small number of rewirings results in routing table update for almost all nodes in the network. The impact per affected node (number of affected routing table entries) increases for increasing number of rewiring steps.

The overall conclusion is that network rewiring has extensive impact on routing tables in OSPF based network. A great portion of the network nodes need to update their routing tables, resulting from a rewiring action. Refined insight in the impact from rewiring on routing tables may help in devising algorithms and methodology for centrally updating the routing tables, as in SDN.

REFERENCES

[1] Barabási, A.-L., & Albert, R. (1999). Emergence of scaling in random networks. Science, 286 (October), 509–512.

[2] Bienstock, D., & Gunluk, O. (1994). A Degree Sequence Problem Related to Network Design. John Wiley & Sons, 195–205.

[3] Erdős, P., & Rényi, A. (1959). On random graphs. Publicationes Mathematicae (Debrecen) 6, 290-297.

[4] Holme, P., & Zhao, J. (2007). Exploring the assortativity-clustering space of a network's degree sequence. Phys. Rev. E, 75(4), 046111.

[5] Newman, M. E. J. (2002). Assortative mixing in networks. Phys. Rev. Lett. 89, 2(4), 1–5, 208701.

[6] Trajanovski, S., Martín-Hernández, J., Wynand, W., & Van Mieghem, P. (2013). Robustness Envelopes of Networks. Journal of Complex Networks, Vol. 1, 44-62.

[7] Van Mieghem, P., Wang, H., Ge, X., Tang, S., & Kuipers, F. A. (2010). Influence of assortativity and degree-preserving rewiring on the spectra of networks. The European Physical Journal B, 76(4), 643–652.

[8] Van Mieghem, P., Ge, X., Schumm, P., Trajanovski, S., & Wang, H. (2010). Spectral graph analysis of modularity and assortativity. Physical Review E, 82(5), 056113.

[9] Van Mieghem, P. (2011). The N-intertwined SIS epidemic network model. Computing, 93(2-4), 147–169. doi:10.1007/s00607-011-0155-y.

[10] Van Mieghem, P. (2011). Graph Spectra for Complex Networks. Cambridge University Press.

[11] Winterbach, W., Ridder, D., Wang, H. J., Reinders, M., & Mieghem, P. (2012). Do greedy assortativity optimization algorithms produce good results? The European Physical Journal B, 85(5), article 151.