

# On the possibility of turning an arbitrary graph into a complex network, using rewiring mechanisms

Michael Shell, *Member, IEEE*, John Doe, *Fellow, OSA*, and Jane Doe, *Life Fellow, IEEE*

**Abstract**—The abstract goes here.

**Index Terms**—Computer Society, IEEE, IEEEtran, journal, LATEX, paper, template.



## 1 INTRODUCTION

## 2 RELATED WORK

## 3 METHODS

### 3.1 General operation

- Frequency of use of dynamic links during the exploration phase ( $f_e$ ).
- Frequency of visits to candidate nodes during the exploration phase ( $f_n$ ).

### 3.2 Initial conditions

#### 3.2.1 Initial topology

The initial topology, is given by a 2D grid, where each node has a Von Neumann neighborhood consisting of the top, bottom, left and right neighbors, except for the nodes lying on the borders and corners of the grid, which have 3 or 2 neighbors, respectively.

This initial structure is only tied by its fixed links. Meanwhile, in this very beginning, the outgoing dynamic edges are initially untied. We proposed a grid of two dimensions because in this topology is easy to know the position of each node.

#### 3.2.2 Compass routing

we want to travel from an initial vertex  $s$  to a destination vertex  $t$ , and that all the information available to us at any point in time is: the coordinates of our destination, our current position, and the directions of the edges incident with the vertex at which we are located. Starting at  $s$ , we will in a recursive way choose to traverse the edge of the geometric graph incident to our current position and with the closest slope to that of the line segment connecting the vertex of our current position to  $t$ . Ties are broken randomly.

Each node forwards a tracer packet based on its local implementation of the compass routing algorithm [32]. To choose the next step of a packet, the routing is performed as

follows: we draw a main straight line between the current node in charge and the final destination. Also, we draw a secondary line between the node in charge and each of its immediate neighbors. Each secondary line crosses the main line with a different angle. The neighbor corresponding to the smallest angle will be designated as the next node in the route of the tracer

#### 3.2.3 Dynamic links

#### 3.2.4 Coordinator election

As is explained in the following section, the coordinator node have two tasks: 1) to count the number of cycles of the rewiring mechanism, and 2) to coordinate the start and end of each cycle phase.

## 3.3 Rewiring mechanism

The rewiring mechanism is a distributed process that is executed in a determined number of cycles. The coordinator node starts each cycle spreading notification messages through a PI algorithm. Each cycle consists of the following four phases:

### 3.3.1 Exploration

Each node starts its exploration phase when it receives a notification from the coordinator node. During this phase, each node sequentially sends twenty tracer packets to arbitrary destination nodes. Packets are routed applying compass routing. Destination nodes respond to a tracer packet with an acknowledgment packet containing the path followed by the tracer packet. When a node receives an acknowledgment packet, it updates its vectors  $f_e$  and  $f_n$ . A node finishes its exploration phase when it has received acknowledgments packets for its twenty tracer packets.

### 3.3.2 Exploration synchronization

This phase starts when the coordinator node has finished its exploration phase. Then it starts a PIF algorithm to spread messages asking if the rest of nodes have finished its exploration phase. When the coordinator node receives an acknowledgment from all the nodes in the network, it starts a PI algorithm to spread messages notifying that nodes must start its rewiring phase.

- 
- M. Shell was with the Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, 30332.  
E-mail: see <http://www.michaelshell.org/contact.html>
  - J. Doe and J. Doe are with Anonymous University.

Manuscript received April 19, 2005; revised August 26, 2015.

### 3.3.3 Rewiring

Each node starts its rewiring phase when it receives a notification from the coordinator. Then, the notified node uses the information in its vectors  $f_e$  and  $f_n$  to take a rewiring decision according to one of the following rules:

- **Rule 1.** A node rewires its least used dynamic link (according with  $f_e$ ) to the most visited node (according to  $f_n$ ).
- **Rule 2.** A node rewires its least used dynamic link (according with  $f_e$ ) to the first node at distance 2 from it, which is the first node in  $f_n$ .

### 3.3.4 Rewiring synchronization

Analogous to the *Exploration Synchronization phase*, this phase starts when the coordinator node has finished its rewiring phase. Then it starts a PIF algorithm to spread messages asking if the rest of nodes have finished its rewiring phase. When the coordinator node receives an acknowledgment from all the nodes in network, it checks if all cycles have been completed. If so, the rewiring mechanism finishes. Otherwise, the coordinator node updates the cycle counter and starts a PI algorithm to spread messages notifying that the cycle has finished and nodes must start the *Exploration phase* of a new cycle.

## 4 RESULTS

### 4.0.1 Initial parameters

- Initial graph: 2D-grid of size  $50 \times 50$ .
- 50 cycles
- 20 tracer packets.
- length of dynamic links.
- Vectors  $f_e, f_n$

### 4.1 Topological properties of the resulting graphs

#### 4.1.1 Average clustering coefficient

#### 4.1.2 Centrality distributions

- Degree centrality
- Betweenness centrality
- Closeness centrality

#### 4.1.3 Distance metrics

- Average path length
- Diameter

#### 4.1.4 Community metrics

- Modularity
- Number of communities

### 4.2 Robustness against failures

- Average clustering coefficient
- Number of connected components
- Proportion of the largest connected component
- Diameter of the largest connected component
- Average path length of the largest connected component

### 4.3 Robustness against attacks

- Average clustering coefficient
- Number of connected components
- Proportion of the largest connected component
- Diameter of the largest connected component
- Average path length of the largest connected component

## 5 DISCUSSION

## 6 FUTURE WORK

- Impact of the initial graph
- Impact of the number of the routing algorithm
- Impact of the number of exploration packets
- Determine the number of exploration packets
- Epidemic spreading

## APPENDIX A

### PROOF OF THE FIRST ZONKLAR EQUATION

Appendix one text goes here.

## APPENDIX B

Appendix two text goes here.

## ACKNOWLEDGMENTS

The authors would like to thank...

## REFERENCES

- [1] H. Kopka and P. W. Daly, *A Guide to L<sup>A</sup>T<sub>E</sub>X*, 3rd ed. Harlow, England: Addison-Wesley, 1999.

**Michael Shell** Biography text here.

PLACE  
PHOTO  
HERE

**John Doe** Biography text here.

**Jane Doe** Biography text here.