Paper

Diffusion in Networks

Rafał Kasprzyk

Faculty of Cybernetics, Military University of Technology, Warsaw, Poland

Abstract—In this paper a concept of method and its application examining a dynamic of diffusion processes in networks is considered. Presented method was used as a core framework for system CARE (Creative Application to Remedy Epidemics).

Keywords—complex networks, diffusion, probabilistic finite-state machine.

1. Introduction

Diffusion is a process, by which information, viruses, gossips and any other behaviors spread over networks [1]–[5], in particular, over social networks.

The standard approach is a simplified assumption that behaviors (information, viruses, gossips) spread in the environment, which is modeled, using very simple construction of *Regular Graphs* like GRID-based graph or similar, very rarely *Random Graphs*. Standard approaches do not explain the real dynamic of diffusion in real-world networks, in particular:

- why even slightly infectious behavior (e.g., contagious diseases) can spread over a network for a long time;
- how to choose nodes to maximize or minimize diffusion range (e.g., how to choose individuals to vaccinate, in order to minimize the epidemic's range);
- what is the mechanism of arising secondary behaviors centres.

The drawbacks of the standards diffusion models is that they do not take into account an underling real-world networks topology. Who (or what) is connected to whom (what), seems to be a fundament question. Apparently, networks derived from data on real life cases (most often: networks growing spontaneously) are neither *Regular Graphs* nor *Random* ones. As it turned out, real networks, which have been intensively studied recently have some interesting features. These features, which origins are nowadays discovered, modeled [6]–[11] and examined [12]–[15] significantly affect dynamics of the diffusion processes within real-world networks. Three very interesting models of real-world networks which have been introduced recently, e.g., *Random Graphs*, *Small World* and *Scale Free*, will be described later in this paper.

We have to also remember that all kinds of behavior spreading over the network have their unique properties, and we should be able to model them. The notion of a state machine seems to be useful in this modeling situation. Using probabilistic finite-state machines [16], [17] we can model a spreading of vast variety of behaviors. For example, we are able to build models of diseases with any states (e.g., *susceptible*, *infected*, *carrier*, *immunized*, *dead*, etc.), and probabilities of transitions from one state to another, resulting from social interactions (contacts). Again, the underling contacts (social network topology) seem to have a huge impact on the dynamic of diffusion processes, what has been already mentioned.

2. Definitions and Notations

Let's define network as follows:

$$Net(t) = \Big\langle G(t) = \langle V(t), E(t)\rangle, \{f_i(v,t)\}_{\substack{i\in\{1,\ldots,NF\}\\ v\in V(t)}}, \{h_j(e,t)\}_{\substack{j\in\{1,\ldots,NH\}\\ e\in E(t)}} \Big\rangle,$$

where:

$G(t) = \langle V(t), E(t)\rangle$ – simple dynamic graph, $V(t), E(t)$ – sets of graph's vertices and edges, $E(t) \subset \big\{\{v,v'\} : v,v' \in V(t)\big\}$ (the dynamic [18] means that $V(t)$ and $E(t)$ can change over time);

$f_i : V(t) \to Val_i$ – the i-th function describe on the graph's vertices, $i = 1,\ldots NF$, (NF – number of vertex's functions), Val_i – is a set of f_i values;

$f_j : E(t) \to Val_j$ – the j-th function describe on the graph's edges, $j = 1,\ldots NH$, (NH – number of edge's functions), Val_j – is a set of h_j values.

We assume that values of function's ($f_i(\cdot)$ and $h_j(\cdot)$ can also change over time.

In this paper we were particularly interested in relationship between the structure of real-world networks and the dynamic of any behaviors on them. Due to this fact, we focused on the characteristics of the graph $G(t)$, while functions on the graph's vertices (nodes) and edges (links) were omitted.

Simple dynamic graphs are very often represented by a matrix $A(t)$, called adjacency matrix, which is a $V(t) \times V(t)$ symmetric matrix. The element $a_{ij}(t)$ of adjacency matrix equals 1 if there is an edge between vertices i and j, and 0 otherwise.

The first-neighborhood of a vertex v_i denote as $\Gamma_i^1(t)$ is defined as set of vertices immediately connected with v_i, i.e.,

$$\Gamma_i^1(t) = \big\{v_j \in V(t) : \{v_i, v_j\} \in E(t)\big\}.$$

The degree $k_i(t)$ of a vertex v_i is the number of vertices in the first-neighborhood of a vertex v_i, i.e.,

$$k_i(t) = \big|\Gamma_i^1(t)\big|.$$

The path starting in vertex v_i and ending in vertex v_j is a sequence of $\langle v_0, v_1, \ldots, v_{k-1}, v_k \rangle$, where $\{v_{i-1}, v_i\} \in E(t) \forall\ i = 1, \ldots, k$. The length of a path is defined as the number of links in it. The shortest path length starting in vertex v_i and ending in vertex v_j is denoted as $d_{ij}(t)$.
Now we can define diameter D as the longest shortest path, i.e.,

$$D(t) = \max_{v_i, v_j \in V(t)} \{d_{ij}(t)\}.$$

Let's denote the number of existing edges between the first-neighborhood of a vertex v_i as $N_i(t)$, i.e.,

$$N_i(t) = \left|\{v_l, v_k\} : v_l, v_k \in \Gamma_i^1(t) \wedge \{v_l, v_k\} \in E(t)\right|.$$

Now, we can define a very important concept, called as the local clustering coefficient C_i for a vertex v_i, which is then be given by the proportion of $N_i(t)$ and divided by the number of edges that could possible exist between first-neighborhood of a vertex v_i (every neighbor of v_i is connected to every other neighbor of v_i). Formally:

$$C_i(t) = \begin{cases} \dfrac{2N_i(t)}{k_i(t)\left(k_i(t)-1\right)}, & \left|\Gamma_i^1(t)\right| > 1 \\ 0, & \left|\Gamma_i^1(t)\right| \leq 1. \end{cases}$$

The clustering coefficient C for the whole network is define as the average of C_i overall $v_i \in V$, i.e.,

$$C(t) = \frac{1}{|V(t)|} \sum_{v_i \in V(t)} C_i(t).$$

The degree distribution $P(k,t)$ of a network is defined as the fraction of nodes in the network with degree k. Formally:

$$P(k,t) = \frac{|V_k(t)|}{|V(t)|},$$

where: $|V_k(t)|$ is the number of nodes with degree k; $|V(t)|$ is the total number of nodes.

2.1. Models of Real-World Networks

Most of the real-world networks are found to have: small average path length, relatively small diameter, high clustering coefficient, and degree distributions that approximately follow a power law, i.e., $P(k,t) \sim k^{-\gamma}$, where γ is a constant. These features, which origins are nowadays discovered indeed affect dynamic of the diffusion processes within networks. Understanding the balance of order and chaos in real-world networks is one of the goals of the current research on so called complex networks.
Identifying and measuring properties of a real-world networks is a first step towards understanding their topology. The next step is to develop a mathematical model, which typically takes a form of an algorithm for generating networks with the same statistical properties.
For a long time real networks without visible or known rule of organization were described using Erdös and Rényi model of *Random Graphs* [8], [9]. Assuming equal probability and independent random connections made between any pair of vertices in initially not connected graph, they proposed a model suffering rather unrealistic topology. Their model has now only a limited usage for modeling real-world network.
Not long ago Watts and Strogatz proposed *Small World* model [11] of real-world networks as a result of simple observation that real networks have topology somewhere between regular and random one. They began with *Regular Graph*, such as a *Ring*, and then "rewire" some of the edges to introduce randomness. If all edges are rewired a *Random Graph* appears. The idea of this method was depicted in Fig. 1.

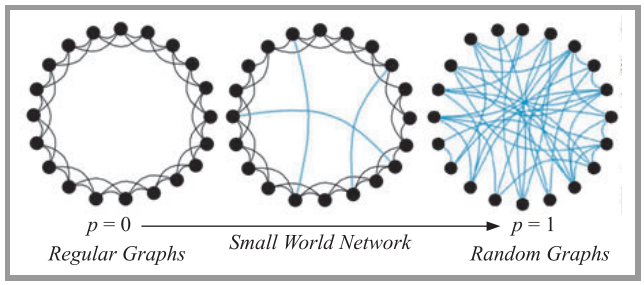

Fig. 1. The idea of *Small World* network model.

The process of rewiring affects not only the average path length but also clustering coefficient. Both of them decrease as probability of rewiring increases. The interesting property of this procedure is that for a wide range of rewiring probabilities the average path length is already low, while clustering coefficient remains high. This correlation is typical for real-world networks.
Barabási and Albert introduced yet another model [6] of real-world networks so called *Scala Free* network as a result of two main assumptions: constant growth and preferential attachment. They showed why the distribution of nodes degree is described by a power law. The process of network generation is quite simple. The network grows gradually, and when a new node is added, it creates links (edges) to the existing nodes with probability proportional to their connectivity. In consequence nodes with very high degree appears (so called *hubs* or *super-spreaders*), which are very important for communication in networks.

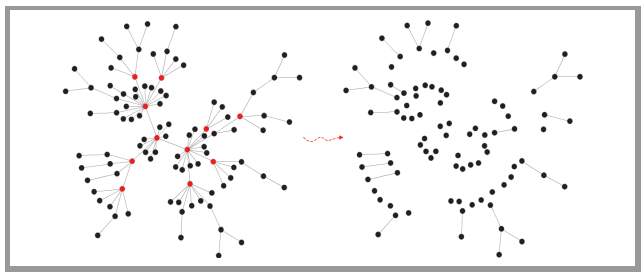

Fig. 2. The role of hubs in *Scale Free* network.

There are many modification of this basic procedure for generating networks. Now it is considered that *Scale Free* models of real-world networks are the best ones (Fig. 2).

2.2. Measures of Nodes Importance

In Fig. 3, there is an example of real social network. Nodes represent individuals and link social interactions.

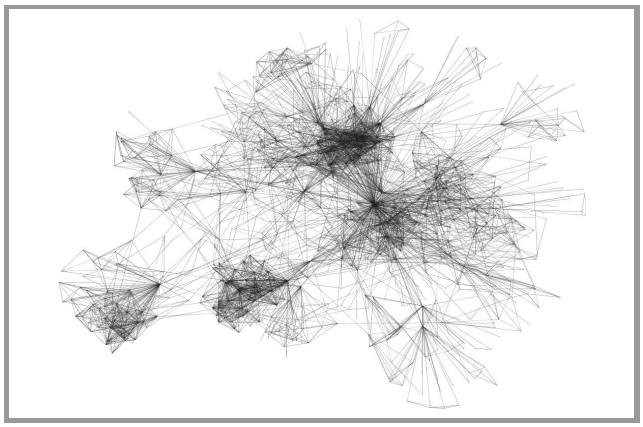

Fig. 3. An example of real social network.

The most basic and frequently asked question is how to identify the most important nodes. The answer can help maximize or, on the other hand, minimize diffusion dynamic of any behaviors within networks. We decided to use the so called centrality measures to assess nodes importance. No single measure of centre is suited for the application. Sever noteworthy measures are: degree centrality, radius centrality, closeness centrality, betweenness centrality, eigenvector centrality. Thanks to these measures we can show, for example, how to disintegrate the network with minimum number of steps and in consequence minimize diffusion area, in particular how to optimize vaccination strategies [19].

Degree centrality. The degree centrality (Fig. 4) gives the highest score of influence to the vertex with the largest number of first-neighbors. It is traditionally defined analogous to the degree of a vertex, normalized over the maximum number of neighbors this vertex could have:

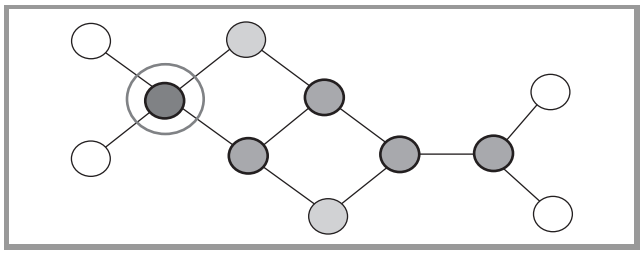

Fig. 4. Importance of nodes according degree centrality.

$$dc_i(t) = \frac{k_i(t)}{|V(t)|-1}.$$

Radius centrality. It chooses the vertex with the smallest value of the longest shortest path starting in each vertex (Fig. 5). So, if we need to find the most influential node for the most remote nodes, it is quite natural and easy to use this measure:

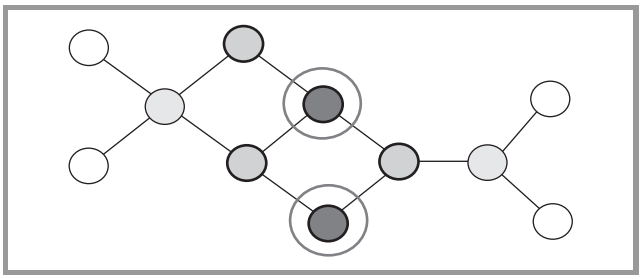

Fig. 5. Importance of nodes according radius centrality.

$$rc_i(t) = \frac{1}{\max\limits_{v_j \in V(t)} d_{ij}(t)}.$$

Closeness centrality. The closeness centrality (Fig. 6) focuses on the idea of communications between different vertices and the vertex, which is "closer" to all vertices and gets the highest score:

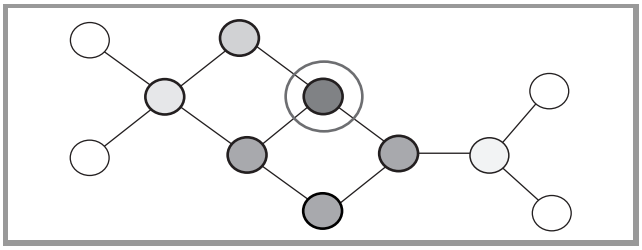

Fig. 6. Importance of nodes according closeness centrality.

$$cc_i(t) = \frac{|v(t)|-1}{\sum\limits_{v_j \in V(t)} d_{ij}(t)}.$$

Betweenness centrality. It can be defined as the percent of the shortest paths connecting two vertices that pass through the considered vertex (Fig. 7). If $p_{l,i,k}(t)$ is the set of all shortest paths between vertices v_l and v_k passing through vertex v_i and $p_{l,k}(t)$ is the set of all shortest paths between vertices v_l and v_k then:

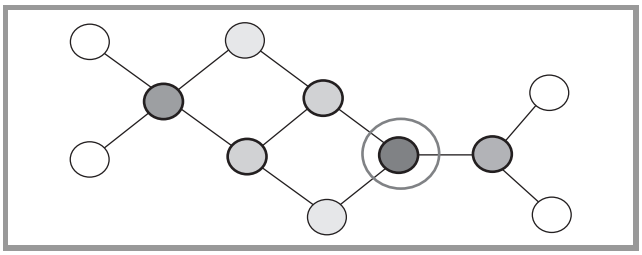

Fig. 7. Importance of nodes according betweeness centrality.

$$bc_i(t) = \frac{\sum\limits_{l<k} \frac{p_{l,i,k}(t)}{p_{l,k}(t)}}{\big(|V(t)|-2\big)\big(|V(t)|-1\big)}.$$

Eigenvector centrality. While degree centrality gives a simple count of the number of connection, a vertex has eigenvector centrality acknowledges that not all connections

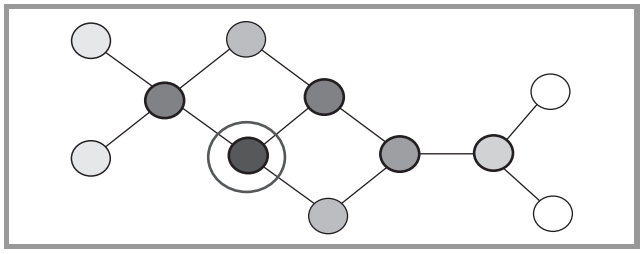

Fig. 8. Importance of nodes according eigenvector centrality.

are equal (Fig. 8). If we denote the centrality of vertex v_i by $ec_i(t)$ then we can allow for this effect by making $ec_i(t)$ proportional to the centralities of the v_i's first-neighbors,

$$ec_i(t) = \frac{1}{\lambda} \sum_{j=1}^{|V(t)|} a_{ij}(t) ec_j(t) .$$

Using matrix notation, we have as follows:

$$\overrightarrow{ec(t)} = \frac{1}{\lambda} A(t) \overrightarrow{ec(t)} .$$

So we have $A(t)\overrightarrow{ec(t)} - \lambda I \overrightarrow{ec(t)} = 0$ and the λ value we can calculate using $\det\left(A(t) - \lambda I\right) = 0$. Hence, $\overrightarrow{ec(t)}$ is an eigenvector of adjacency matrix with the largest value of eigenvalue λ.

2.3. Model of Diffusion

All in all, who is connected to whom seems to be crucial for diffusion in networks, but all kinds of behaviors have their unique properties. In consequence, we defined the model of diffusion in network as a vector, with three elements:

$$Diff(t) = \langle Net(t), PSM_{x=1,2,\ldots,N}, Gen(v,t) \rangle ,$$

where:

$Net(t)$ – network model of system constitutes diffusion environment;

PSM_x – probabilistic finite-state machine model of considered behavior (information, virus, gossip and so on);

$Gen : V(t) \rightarrow \text{SIG}$ – specific function for simulation needs (generator of signals), which assigns for each vertex in each simulation step a set of signals as a result of vertices' first-neighborhood and theirs states. These signals are received and processed by *PSM* on each vertex.

Thus, both concepts, i.e., probabilistic state machine models and real-world networks topology are highly pertaining to the presented idea subject and objectives. The aim is to uncover the diffusion mechanisms hidden in the structure of networks.

3. Simulation Environment

Our simulation environment is based on well known *Gephi* platform [20] for interactive visualization and networks exploration. The simulation environment has been implemented as a set of plugins. This kind of extensions is feasible thanks to the *Gephi* architecture based on MVC (*Model-View-Controller*) and *Service Locator* patterns. MVC pattern isolates algorithms and data from GUI (Fig. 9), permitting independent development, testing and maintenance of each one. *Service Locator* is an implementation of the IoC (*Inversion of Control*) pattern. It is a technique that allows removing dependencies from the code.

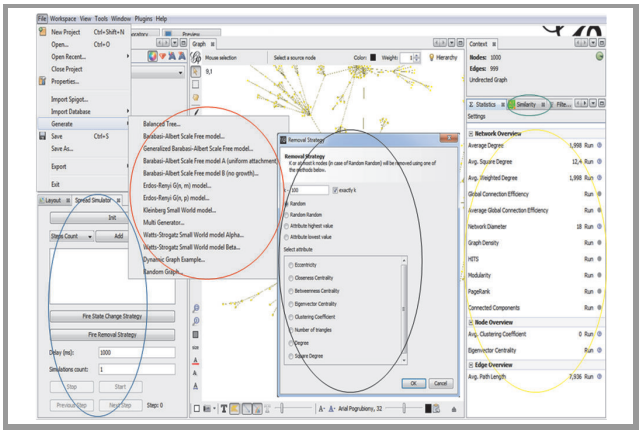

Fig. 9. GUI of simulation environment.

We added to *Gephi* new functionalities, such as: complex networks generators, scenarios for centrality measures utilization in simulation of diffusion, and finally the ability to simulate diffusion of any behaviors in any networks.
Gephi architecture allows us to develop the code according to SOLID principles (*Single responsibility*, *Open-closed*, *Liskov substitution*, *Interface segregation*, *Dependency inversion*) that is five basic principles of object-oriented programming and design. It makes the code very extensible and scalable.

4. Simple Case Study

Let us now analyze a very simple case study of the diffusion process from the field of epidemiology. One of the most extensively studied epidemic models is SIS (*Susceptible-Infected-Susceptible*). In each time step, the susceptible individuals are infected by each infected neighbors with probability *beta* and the recovering rate of infected individuals to susceptible ones is *alfa*. Parameter *lambda* is known in literature as speed of spreading or virulence of the disease and is define as:

$$lambda = beta / alfa .$$

Figure 10 representing PSM_1 diagram of SIS model of a disease prepared in our simulation environment with $lambda = 0.5/0.1 = 5$.

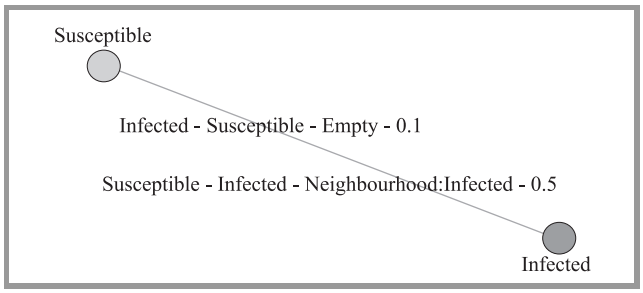

Fig. 10. SIS model of a disease.

The central question then becomes: how network topology may affect diffusion process. We focus on the SIS model of a disease spreading in networks with different topology. We use three networks: *Scale Free* (SF), *Random Graphs* (RG) and *Regular Graphs* that is exactly GRID-base one (very popular graph used in cellular automata). All networks consist of 10 000 nodes and about 20 000 edges. Average degree of nodes are similar and close to 4.

At time 0 small number of nodes (1%) is chosen randomly and infected. Then, the simulation of diffusion process is started. Each simulation was repeated 1000 times. Dynamic of disease diffusion in different networks as a function of *lambda* is presented in Figs. 11–15.

We can see that if *lambda* is high (e.g., *lambda* = 5), topology of networks have small impact on diffusion dynamic. According to Fig. 11, the number of infected individuals rose sharply and flattened out at a very high level (about 90%).

When *lambda* parameter decreases diffusion dynamic are more and more dependent on network topology. For *lambda* = 0.5 (Fig. 12) diffusion dynamic in GRID-based graph is significantly different from diffusion in *Scale Free* and *Random Graphs*. First of all, the number of infected

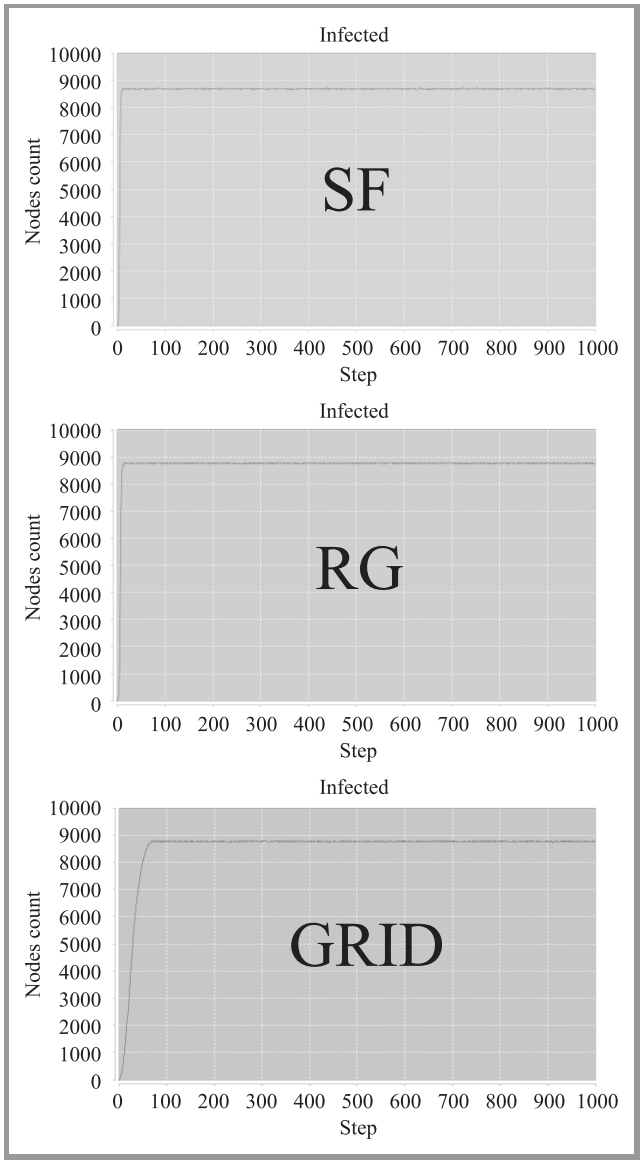

Fig. 11. SIS model of a disease with *lambda* = 5 in networks with different topology.

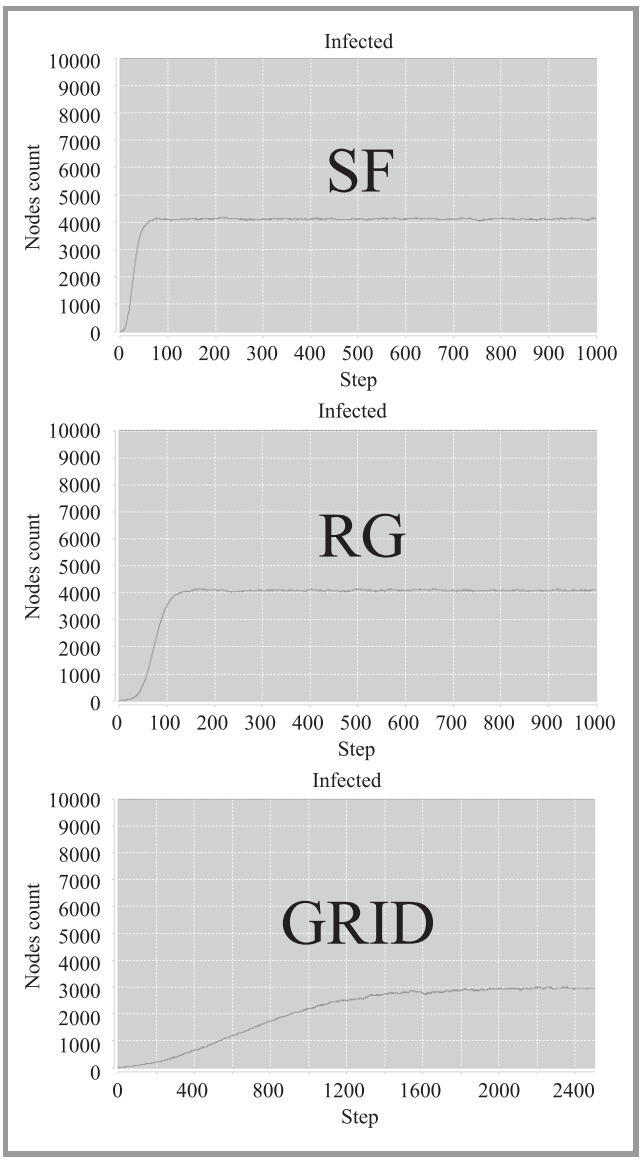

Fig. 12. SIS model of a disease with *lambda* = 0.5 in networks with different topology.

individuals rose slower, secondly flattened out at a lower level (about 30% by contrast with 40% for *Scale Free* and *Random Graphs*).

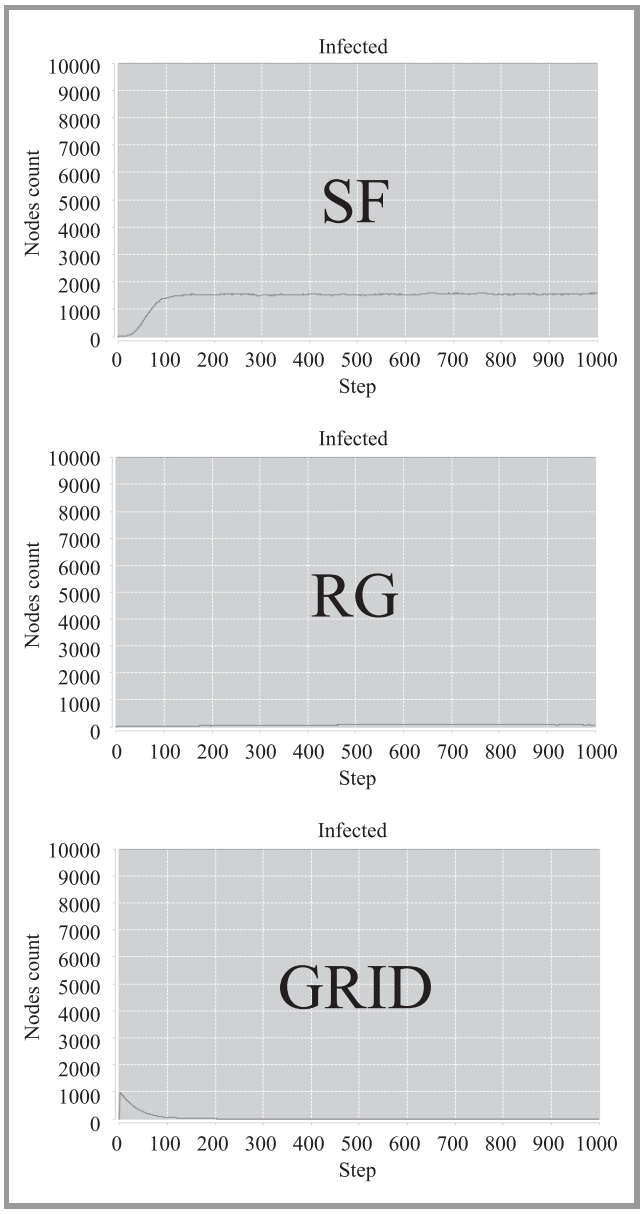

Fig. 13. SIS model of a disease with *lambda* = 0.25 in networks with different topology.

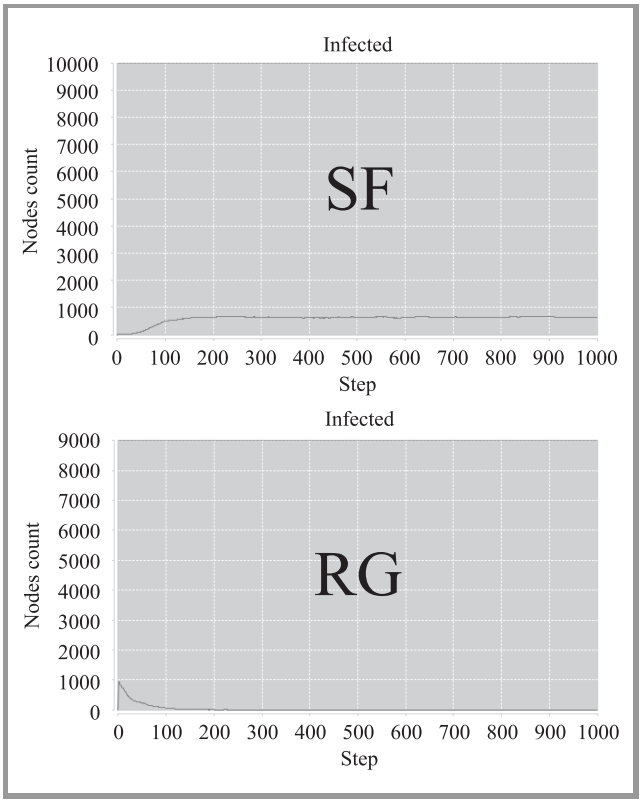

Fig. 14. SIS model of a disease with *lambda* = 0.2 in networks with different topology.

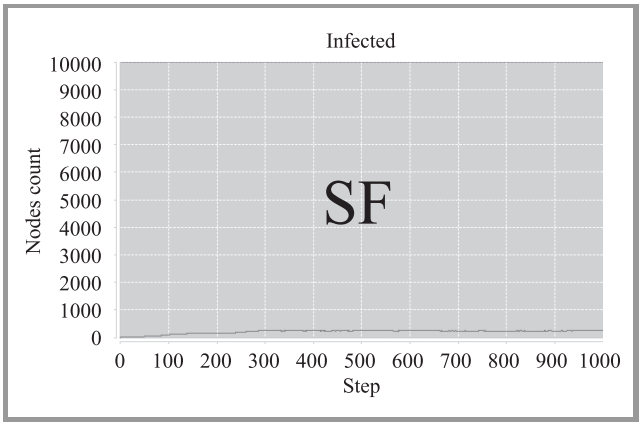

Fig. 15. SIS model of a disease with *lambda* = 0.15 in network with different topology.

It turn out that for *lambda* = 0.25 (Fig. 13) the virus of infection disease disappear from population modeled as GRID-base graph (even though 10% individuals were infected at start time).

For *lambda* = 0.2 (Fig. 14) the virus of infection diseases also disappear from population modeled as *Random Graphs* (even though 10% individuals were infected at start time).

For *lambda* = 0.15 (Fig. 15) the virus is able to spread only in *Scale Free* network. It is an answer to the question: Why even slightly contagious diseases can plague human population over a long time without being epidemic. Not long ago it was also analytically proved that in *Scale Free* network there is no epidemic threshold for *lambda* value [5].

5. System CARE

As practical utilization of our research system called CARE (*Creative Application to Remedy Epidemics*) was developed [21]–[23]. CARE is *Decision Support System*, which help decision makers to fight with epidemic. CARE con-

tains five modules: *Disease Modeling*, *Social Network Modeling*, *Simulation*, *Vaccination* and *Questionnaires*.

In the *Disease Modeling* module, using probabilistic finite-state machine approach, we can model any kind of disease based on knowledge from the field of epidemiology. We allow to build the models of diseases with any states and transitions in the editor we have proposed.

Fig. 16. CARE user interface.

In *Social Network Modeling* module we can model and generate social networks using complex network theory. Using proposed generators we obtain synthetic networks but with the same statistical properties as real-world social networks. The algorithms generate networks that are *Regular Graphs*, *Random Graphs*, *Small World* networks, *Scale Free* networks or modifications thereof.

Using *Simulation* module we can visualize and simulate how the epidemic will spread in a given population. The system proposes two ways of information visualization. The first way is called "*Layout*" and helps user to manipulate networks and to set up some parameters of simulation. The alternative way is "*Geo-contextual*" one which allows to visualize networks on the world map. The system estimates the expected outcomes of different simulation scenarios and generate detailed reports. The user can assess the results and the effectiveness of the chosen vaccination strategy.

Based on the centrality measures *Vaccination* module helps the user to identify so called "*super-spreaders*" and to come up with the most efficient vaccination strategy [19]. The identification and then vaccination or isolation of the most important individuals of a given network helps decision makers to reduce the consequence of epidemics, or even stop them early in the game.

The crucial step in fighting against a disease is to get information about the social network subject to that disease. *Questionnaires* module helps building special polls based on sociological knowledge to help discover network topology. Polls designed in this way are deployed on mobile devices to gather data about social interaction.

6. Conclusion

In this paper we presented the model of diffusion in networks and the simulation environment based on *Gephi* platform. We would like to admit that we are a little bit closer to understand diffusion in networks. The solutions presented in the paper have practical implementation as a system to fight with infection diseases called CARE. Now CARE is a subsystem of monitoring, early warning and forecasting system SARNA, which was build at MUT and was put into practice in the Government Safety Centre in Poland [24]. It is worth to mentioned that CARE has its counterpart to fight with malwares in the Internet called VIRUS [25].

Acknowledgements

This work was partially supported by the research project GD-651/2011/WCY of Cybernetic Faculty at Military University of Technology.

References

[1] Godin S.: Unleashing the Ideavirus, Hyperion, New York, 2001.

[2] J. Leskovec, L. Adamic, and B. A. Huberman, "The dynamics of viral marketing", *ACM Trans. Web*, vol. 1, no. 1, article 5, 2007.

[3] A. L. Lloyd and R. M. May, "How viruses spread among computers and people", *Science*, vol. 292, no. 5520, pp. 1316–1317, 2001.

[4] D. López-Pintado, "Diffusion in complex social networks", *GAMES and Economic Behavior*, vol. 62, no. 2, pp. 573–590, 2008.

[5] R. Pastor-Satorras and A. Vespignani, "Epidemic spreading in scale-free networks", *PRL*, vol. 86, no. 14, pp. 3200–3203, 2001.

[6] A. L. Barabási and R. Albert, "Emergency of scaling in random networks", *Science*, vol. 286, pp. 509–512, 1999.

[7] A. L. Barabási and R. Albert, "Topology of evolving networks: local events and universality", *PRL*, vol. 85, no. 24, pp. 5234–5237, 2000.

[8] P. Erdös and A. Rényi, "On random graphs", *Publicationes Mathem.*, vol. 6, pp. 290–297, 1959.

[9] P. Erdös and A. Rényi, "On the evolution of random graphs", Publications of the Mathematical Institute of the Hungarian Academy of Sciences 5, pp. 17–61, 1959.

[10] M. E. J. Newman, "Models of the small world: A review", *J. Stat. Phys.*, vol. 101, pp. 819–841, 2000.

[11] D. J. Watts and S. H. Strogatz, "Collective dynamics of "small-world" networks", *Nature*, vol. 393, pp. 440–442, 1998.

[12] A. L. Barabási and R. Albert, "Statistical mechanics of complex networks", *Rev. Modern Phys.*, vol. 74, pp. 47–97, 2002.

[13] M. E. J. Newman, "The structure and function of complex networks", *SIMA Rev.*, vol. 45, no. 2, pp. 167–256, 2003.

[14] S. H. Strogatz, "Exploring complex networks", *Nature*, vol. 410, pp. 268–276, 2001.

[15] X. Wang and G. Chen, "Complex networks: Small-world, scale-free and beyond", *IEEE Circ. Sys. Mag.*, vol. 3, no. 1, pp. 6–20, 2003.

[16] A. Sokolova and E. P. de Vink, *Probabilistic Automata: System Types, Parallel Composition and Comparison*, LNCS 2925, Heidelberg: Springer, 2004, pp. 1–43.

[17] E. Vidal, F. Thollard, C. de la Higuera, F. Casacuberta, and R. C. Carrasco, "Probabilistic Finite-State Machines – Part I", *IEEE Trans. Pattern Anal. Machine Intel.*, vol. 27, no. 7, pp. 1013–1025, 2005.

[18] F. Harary and G. Gupta, "Dynamic Graph Models", *Mathl. Comput. Modelling*, vol. 25, no. 7, pp. 79–87, 1997.

[19] R. Kasprzyk, "The vaccination against epidemic spreeding in complex networks", *Biuletyn ISI*, no. 3(1/2009), pp. 39–43, 2009.

[20] M. Bastian, S. Heymann, and M. Jacomy, "Gephi: an open source software for exploring and manipulating networks", in *Proc. Int. AAAI Conf. Weblogs Social Media*, San Jose, CA, USA, 2009.

[21] R. Kasprzyk, A. Najgebauer, and D. Pierzchała, "Modelling and simulation of infection disease in social networks", *Comput. Collective Intel.*, LNAI 6922, pp. 388–398, 2011.

[22] R. Kasprzyk, D. Pierzchała, and A. Najgebauer, "Creative application to remedy epidemics", in *Risk Analysis VII & Brownfields V*, C. Brebia, Ed. WIT Press, 2010, pp. 545–562.

[23] R. Kasprzyk, B. Lipiński, K. Wilkos, M. Wilkos, and C. Bartosiak, "CARE – creative application to remedy epidemics", *Biuletyn ISI*, no. 3(1/2009), pp. 45–52, 2009.

[24] A. Najgebauer, D. Pierzchała, and R. Kasprzyk, "A distributed multilevel system for monitoring and silumation of epidemics", in *Risk Analysis VII & Brownfields V*, C. A. Brebbia, Ed. WIT Press, 2010, pp. 583–596.

[25] R. Kasprzyk, "Symulator rozprzestrzeniania się złośliwego oprogramowania w sieciach komputerowych", *Symulacja w badaniach i rozwoju*, vol. 1, no. 2, pp. 139–150, 2010 (in Polish).

Rafał Kasprzyk was commissioned as 2Lt. in 2004 and one year later received his M.Sc. Eng. degree after individual studying in DSS (Decision Support System) at Cybernetics Faculty as the 1st place graduate from Military University of Technology in 2005. He was promoted to the rank of Lt. and Capt. in 2008 and 2010 respectively. In 2012 he received Ph.D. degree in the field of computer science. He has worked as a lecturer at Cybernetics Faculty since 2005. He has participated in many scientific projects connected with combat simulation and crisis management. His main interest are graph and network theory, decision support systems, computer simulation, homeland security and cyber-security.
E-mail: rkasprzyk@wat.edu.pl
Faculty of Cybernetics
Military University of Technology
Gen. S. Kaliskiego st 2
00-908 Warsaw, Poland