Automation, Control and Intelligent Systems
2015; 3(4): 56-62
Published online July 29, 2015 (http://www.sciencepublishinggroup.com/j/acis)
doi: 10.11648/j.acis.20150304.12
ISSN: 2328-5583 (Print); ISSN: 2328-5591 (Online)

Evolutionary Model for Virus Propagation on Networks

Arnold Adimabua Ojugo[1], Fidelis Obukowho Aghware[2], Rume Elizabeth Yoro[3], Mary Oluwatoyin Yerokun[4], Andrew Okonji Eboka[4], Christiana Nneamaka Anujeonye[4], Fidelia Ngozi Efozia[5]

[1]Dept. of Math/Computer, Federal University of Petroleum Resources Effurun, Delta State, Nigeria
[2]Dept. of Computer Science Education, College of Education, Agbor, Delta State, Nigeria
[3]Dept. of Computer Sci., Delta State Polytechnic, Ogwashi-Uku, Delta State, Nigeria
[4]Dept. of Computer Sci. Education, Federal College of Education (Technical), Asaba, Delta State, Nigeria
[5]Prototype Engineering Development Institute, Fed. Ministry of Science Technology, Osun State, Nigeria

Email address:

arnoldojugo@yahoo.com (A. A. Ojugo), ojugo_arnold@yahoo.com (A. A. Ojugo), aghwarefo@yahoo.com (F. O. Aghware), rumerisky@yahoo.com (R. E. Yoro), an_drey2k@yahoo.com (A. O. Eboka), agapenexus@hotmail.co.uk (M. O. Yerokun), anujeonyechristy@gmail.com (C. N. Anujeonye), fenngo31@yahoo.com (F. N. Efozia)

To cite this article:

Arnold Adimabua Ojugo, Fidelis Obukowho Aghware, Rume Elizabeth Yoro, Mary Oluwatoyin Yerokun, Andrew Okonji Eboka, Christiana Nneamaka Anujeonye, Fidelia Ngozi Efozia. Evolutionary Model for Virus Propagation on Networks. *Automation, Control and Intelligent Systems*. Vol. 3, No. 4, 2015, pp. 56-62. doi: 10.11648/j.acis.20150304.12

Abstract: The significant research activity into the logarithmic analysis of complex networks will yield engines that will minimize virus propagation over networks. This task of virus propagation is a recurring subject and design of complex models will yield solutions used in a number of events not limited to and include its propagation, network immunization, resource management, capacity service distribution, dataflow, adoption of viral marketing amongst others. Machine learning, stochastic models are successfully employed to predict virus propagation and its effects on networks. This study employs SI-models for independent cascade and the dynamic models with Enron dataset (of e-mail addresses) and presents comparative result using varied machine models. It samples 25,000 e-mails of Enron dataset with Entropy and Information Gain computed to address issues of blocking, targeting and extent of virus spread on graphs. Study addressed the problem of the expected spread immunization and the expected epidemic spread minimization; but not the epidemic threshold (for space constraint).

Keywords: Stochastic, Immunize, Network, Vertices, SIS, SIR, Search Space, Solution, Models

1. Introduction

Networks are dynamic and their normal operation is continually threatened by unethical users referred to as hackers. They employ use of harmful and malicious programs called malware to wreak havoc to users of networks. Today, Internet has become a high target for the spread of such malwares – as hackers do damage globally, much more easily and faster. Thus, its early detection is imperative to minimize damage caused by it (Desai, 2008).

Malwares are known to attach copies of itself, alters the behaviour as well as modifies attributes of its host machine's files without user's knowledge (Szor, 2005). Malwares also can sometimes, modify their codes as they infect, to include an evolved copy (Dawkins, 1989). Malwares are grouped into simple, encrypted, polymorphics and metamorphic viruses. Malware are considered malicious software if they consist of codes, scripts, active contents and other software – designed to disrupt or deny operations, gather information that tends to loss of privacy or exploitation, gain unauthorized access to system resources, and other abusive behaviour (Singhal and Raul, 2012). Thus, software codes are considered a malware based on the perceived intentions of its creator rather than any particular feats.

Antivirus (AVs) is designed to detect, prevent and remove malwares such as viruses, worms, Trojans, spyware and adware. AVs detection mechanism are broadly grouped into: (a) signature-based scans for signature, and to evade it – virus makers create new virus strings that can alter their structure while keeping its functionality via code obfuscation, and (b) code emulation creates sandbox so that files are executed within it while scanning for virus. If virus is detected, it is no threat as it is running in controlled environment that limit damage to host machine (Singhal and

Raul, 2012).

AVs can often impair system performance as any incorrect decision may lead to security breach since it runs at the operating system's kernel. If an AV uses heuristics, its success depends on a right balance between positives and negatives. Today, virus may no longer be executables but macros, which present security risk and antivirus heavily relies on signature-detection. Some viruses evade detection effective (Filiolel, 2005) via code obfuscation and encryption methods. Studies show the best AVs can never yield a perfect detection since all scanners can yield a false positive result and identify benign files as malware (Bishop, 2005).

The idea is to model a network using graph theory with emails as nodes. It implements SI (susceptible-infect) design for virus propagation on graph with a view of immunization problem that helps to deal with extent, targeting and blocking of virus propagation on a network. SIR model with independent cascade is used as it aid inoculated nodes stay completely invulnerable to viral attacks.

1.1. Network Topologies

Networks are used for spreading of data – making it easier for users to disseminate useful data as well as viruses. The problem of virus propagation has been a recurring subject and ongoing research notes that every harmful data spread over such networks are considered as malware or viruses as can be interchangeably used; while the process of impeding the spread of such harmful data (malware) over such social network is referred to as network immunization. This aims to prevent the spread of such malwares, protect such networks from virus attacks and control data and sensitive information leakages – while at same time noting that our resources such as vaccination, AVs and influences are quite costly and limited in their capability to discover such malware. With such AVs and vaccinations, users aim to achieve the best effect; while still allocating the least resources possible (Ojugo et al 2013).

Hackers (or adversary) wreak more havoc being aware of the propagation model used to avert such attacks. In simplest form, a social network is seen as a complex graph. Thus, the propagation model has as input a graph G = (V,E), state vector $S_v^{(t)}$ for each node vertex v ∈ V at t, and an internal parameter vector P. Based on the states of all interacting nodes, it outputs a new state vector $S_v^{(t+1)}$ for each node at t+1. Models are applied to synthetic data with graph types (Mitchell, 1997; Giakkoupis et al, 2010; Kermach and McKendrick, 1927; Pastor-Satorras and Vespignani, 2002) as:

a. Scale-Free Networks: Probability that node *x* in network is of degree k is proportional to $k^{-\gamma}$ with $\gamma > 1$. Scale free graph are modeled as by Barabasi and Albert (1999). It inserts nodes sequentially with each node linked to an existing one chosen with a probability proportional to its current degree in a tree-fashion with grandparent-parent-children-grandchildren structure and it builds graph with exponent $\gamma = 3$ denoted with G_{sf}. Each node in the graph can be autonomous but must be connected to an existing one. Thus, two nodes are connected together on the graph via physical link between two corresponding autonomous systems. Such is referred to as Autonomous Systems. Another scale free graph consists of undirected edges between nodes, also termed Co-Author graphs.
b. Small World Networks are those with small characteristics path length L (the average shortest path between any pair of vertices) and large clustering coefficient C (the average fraction of pairs of neighbours of a node also connected to each other). We generate small-world graphs using the generating model proposed in Watts (1999). G_{swL} denotes small-world graphs with path length feat; while G_{swC} to denote those of large clustering coefficient.

Graphs G_{swL} are influenced by α, which intuitively determines the probability of two nodes being connected given a number of their common neighbours. It controls to what extent the graph has small and densely connected components. As α nears infinity, G_{swL} becomes a random graph. Conversely, graphs of G_{swC} are influenced by q, which determines the probability of an edge in the lattice being rewired to connect to a random node in the graph. Thus, initialized on a ring lattice, each node is of degree k. Small values of q entails G has high clustering coefficient and large average path length; while large values q creates random graphs. For q-values close to 0.01, the generated graphs are small-world graphs. Note that G_{swL}, G_{swC} and G_{sf} are quite distinct graphs.

1.2. SI-Models for Epidemic Spread

Two major models are: Susceptible-Infect-Remove (SIR) and Susceptible-Infect-Susceptible (SIS). In SIR, a node may be in any of these states: (a) susceptible: if the node has no virus but will become infected if it is exposed to it, (b) infected: if the node has the virus and can pass it on to others, and (c) removed: if the node had the virus but has been recovered or virus dies. The node is permanently immunized and can no longer participate in propagation, and a particular node cannot be infected twice. Conversely SIS, a node can be cured but not immunized. Thus, it can be infected again. Such node switches between susceptible and immunized.

Giakkoupis et al (2010) and Lahiri and Cebrian (2010) A graph holds these definitions as true:

a. Network is directed or undirected graph for propagation of virus. A node is represented as v ∈ V; and edge (u,v) ∈ E represents interactions between two individuals or nodes in the system. It also assume that the graph is drawn from a specific family (algorithm consider all possible graphs). For G = (V,E) as a dynamic network, E is set of edges that are time-stamped, $(u,v)_t \in E$ are interactions at $t \in Z^+$. In a typical SI setting, set of nodes are initialized as *activated*. The propagation process proceeds in discrete time-steps such that at each time-step, an activated vertex may come in contact with *inactive* vertices. This continues till a stop criterion is satisfied or there are no more inactive vertices.
b. The virus propagation model that determines how the virus is spread on the network.

c. Immunization model aims to minimize viruses spread and an immunized node cannot transfer or receive a virus. It is conceptually removed from graph. Cost of immunization model is, number of nodes immunized.
d. Adversary with knowledge of the propagation model, plants *d* copies of virus in G, to maximize speed of spread is denoted as F_d. An adaptive adversary is one who has knowledge of choices made by immunization algorithm; while a randomized adversary places copies of virus, uniformly at random on the network.

1.3. Independent Cascade Model

It is a discrete-time special case SIR model in which at t = 0, an adversary inserts *d* copies of virus to some nodes on graph. If node *x* is infected the first time at t, it has single chance to infect any neighbours *y* currently uninfected. Probability that *x* succeeds with *y* is P_{xy}. If *x* succeeds, *y* is infected at t+1; Else, *x* tries again in the future (even if *y* gets infected by another neighbour). This process continues and stops after *n*-steps if no more infections are possible. It requires a nodes stay infected exactly once and it is the independent cascade model following Kempe et al (2003). Graph of size M, has M_d subset of nodes and *d* copies of virus placed on the network. With propagation complete, $S(M_d,G)$ is expected number of infected nodes. Expectation exceeds all random choices made by propagation model. Eq. 1 is maximum expected number of infected nodes and maximum exceeds all possible initial virus placements.

$$S_d(G) = \max_{M_d} S(M_j, G) \quad (1)$$

$A_d = arg\max_{M_d} S(M_d, G)$ corresponds to choices made by an adaptive adversary. $S_d(G)$ is epidemic spread in G. A similar definition of epidemic spread of randomize adversary is Eq. 2, that defines expected epidemic spread as where the expectation takes over all possible positions of the *d* viruses placed on the network and given by:

$$S'_d(G) = \mathrm{E}_{\mathrm{M_d}}[S(M_d, G) \quad (2)$$

1.4. Dynamic Propagation Model

In SIS, viruses are seen as dynamic birth-death process that evolves overtime. It continues to either propagate or eventually die. An infected node *x* spreads virus to node *y* in time t with infection rate of $\frac{\beta}{\delta}$ and probability β. At same time, an infected node may recover with probability δ. With adjacency matrix T, $\lambda_1(T)$ is largest eigen-value of T. The condition $\frac{\beta}{\delta} < \frac{1}{\lambda_1(T)}$ holds true as epidemic threshold and is sufficient for quick recovery, easily proven (Ganesh et al, 2005; Wang et al, 2003.

2. Statement of Problem

While networks are an effective way to spread data, they also help with spread and propagation of malwares and viruses. These have significant implication on the network as it can destroy user data and/or become a means to retrieve useful, confidential data from unsuspecting users. It thus becomes imperative to deal with means that help user curb the spread of viruses on networks.

3. Network Immunization Problem: Proposed Framework / Design

Typical challenges in SI propagation model are:

1. Extent: With specific subset of initially activated vertices in network and propagation model used, how many vertices are expected to be activated after a specific time/period?
2. Targeting: Which vertices are targeted as initiators by an adversary to result in max extent of spread (Cohen et al, 2003)? This is a hard NP to solve optimally, regardless of propagation model used (Kempe et al, 2003)
3. Blocking: Which vertices are targeted for immunization to minimize the expected number of activated vertices (Singhal and Raul, 2012; Dezso and Barabasi, 2002)?

The immunization problem is thus defined as thus:

Problem 1 – Spread Immunization: Given graph G, a number of *d* initial copies of viruses and number *k*. We immunize *k* nodes in G such that expected spread $S_d(G')$ in the immunized graph is minimized. The role of the adversary is played by the influence-maximization model of Kempe et al (2003), whose proof is omitted due to space constraint.

Problem 2 – Expected Epidemic Spread Minimization: Given graph G, a number of *d* initial copies of viruses and a number *k*. We immunize *k* nodes in G such that the expected epidemic spread $S'_d(G')$ in the immunized graph is minimized. As a hard NP-complete task that attempts to immunize G with random strategy for influence spread and closely related to the sum-of-squares partition task as studied in Aspnes et al (2005).

Problem 3 – Threshold Maximization: Given G, a number of *d* copies of viruses and an infection rate of $\frac{\beta}{\delta}$, we immunize the minimum number of *k* nodes in G so that $\frac{\beta}{\delta} < \frac{1}{\lambda_1(T)}$ holds true. Thus, the epidemic spread $S'_d(G')$ in the graph is minimal. The task attempts to immunize G with influence spread while seeking the minimal number of nodes that can be immunized.

4. Experimental Framework

Machine learning as a branch of artificial intelligence is a scientific discipline that deals with development and design of algorithms that allows machines (computers) to evolve its behaviour based on empirical data such as sensors data and databases. A learner takes advantage of data to capture its characteristics of interest of their underlying and unknown probability distribution. Such data may illustrate relationships between observed variables. Major focus on machine learning is to automatically learn to recognize complex patterns and make intelligent decisions from it (Singhal and Raul, 2012).

4.1. Dataset

The Enron e-mail dataset is one of the largest, e-mail dataset available, representing a dynamic social network. Each node in network is an e-mail address with a directed time-stamped edge as e-mail sent between two addresses. Lahiri and Cebrain (2010) obtained all e-mail headers of 1,326,771 time-stamped e-mails from 84,716 addresses and 215,841 unique timestamps as non-uniformly covering a period of approximately 4years.

We sampled a subset of 25000 addresses representing about 30% for the graphs G_{sf}, G_{swL} and G_{swC} families. In all cases, we used p = 0.25, q = 0.009 and α = 6 to generate the graphs. These result in models' graph having low average path length and high clustering coefficient. There exists the relationship between parameters (p, q and α) and the clustering coefficient as studied in (Watts, 1999). α starts with value 1 till it reaches 6. The clustering coefficient drop as α increase and for small values of q, high clustering coefficient is observed while clustering coefficient drops as q tends to 1.

4.2. Genetic Algorithm (GA)

GA is inspired by Darwinian evolution (survival of fittest) and consists of a pool of solutions chosen for natural selection to a specific task. Each potential solution is an individual for which an optimal is found via four operators namely: initialize, select, crossover and mutation (Coello et al, 2004). Individuals with genes close to its solution's optimal is said to be fit, and the fitness function determines how close an individual is to the optimal solution.

Theorem 1: With G, adjacency matrix T and infection rate $\frac{\beta}{\delta}$. $\frac{\beta}{\delta} < \frac{1}{\lambda_1(T)}$ is true, if expected time for virus to die is logarithmic. This is a function of the number of nodes in the graph against an adversary. Many interesting families of graphs holds too that $\frac{\beta}{\delta} > \frac{1}{\lambda_1(T)}$ is recovery rate – so that expected time at which virus dies out is exponential, known as Epidemic threshold.

GA achieves its fitness function as it finds solution to the network. Its dynamic, non-linear model can be made linear so as to resolve it analytically. The dynamic nature of graph as social network makes them impossible to resolve analytically using non-linearity (if considered as a multiple copies model). Let v^t be an n-dimensional vector of states at t-steps and $v_d^t$is number of virus copies at node *x* at t-steps. Initialized at t = 0, $v_d^0$is number of *d* copies planted by an adversary. At t+1, the model evolves for (all) nodes *x,y,z* in the network, and for each v_d^tcopies of virus planted at node *x*, virus is propagated to node *y* with probability β. Virus dies with probability 1 – δ, and if Δ = βT + diag(1 – δ,…, 1 – δ) is true, $v^t$is the expected state of system at time t. Then, model is completely linear if $\Delta v^t = \Delta v^{t+1}$ proven as in (Giakkoupis et al, 2010; Kempe et al, 2003; Kleinberg, 2007 and Hethcotee, 1989).

For GA, operators are (Lahiri and Cebrian, 2010):

a. Initialize/Select: For edge (u,v) at time t, let its corresponding state string be coded as $S_u^{(t)}$ and $S_v^{(t)}$ vectors respectively, which are interactions between the nodes. Thus, we select $S_u^{(t+1)} = S_u^{(t)}$ and $S_v^{(t+1)} = S_v^{(t)}$.
b. A crossover point C is randomly and uniformly selected from the interval [1,β]. Two new states strings or vectors are created by swapping the tails $S_u^{(t)} and S_v^{(t)}$ - where tail is defined by all positions including and after the index C. let these two new vector states strings be denoted as st_1 and st_2 respectively.
c. Objective score of each new state vector is then evaluated according to the fitness function f(x). if any of them have a greater fitness value that either of their parent node, the corresponding parent nodes state vector string is replaced by its offspring for the next iteration, achieved via:

$$S_u^{(t+1)} = \text{argmax}_{x \in \{S_u^{(t)}, S_u^{(t+1)}, st_1, st_2\}} f(x) \quad (1)$$

$$S_v^{(t+1)} = \text{argmax}_{x \in \{S_v^{(t)}, S_v^{(t+1)}, st_1, st_2\}} f(x) \quad (2)$$

In the case of ties in fitness score between original and a new string vector, its original string vector is retained – as the offspring cannot outperform its parent. This model is close to GA with spatially distributed population GASDM (Min et al, 2006; Payne and Eppstein, 2006) – except that the GA's selection operator is replaced with real social network data that dictates the sequence of mating operation. The propagation in GASDM occurs as states vector and are modified using crossover. After which, they are subsequently adopted based on fitness value. Major missing components to add meaning to this mapping is the choice of its fitness function, f(x).

Study proposes that the objective/fitness function be achieved via Information Gain.

4.3. Decision Tree / IDA

It uses hill-climbing to search a space for optima. Once a peak is found, it restarts with another randomly chosen starting point (as such peak may not be the only one that exists). Its merit is simplicity with functions with too many maxima. Each random trial done in isolation helps immunize the nodes and overall shape of the domain is transparent to an adversary – because, as random search progresses, it continues to allocate its trials evenly over the space and evaluates as many points in the both regions found with low- and high-fitness values. Its choice is in selecting feats and attributes in graph to test is via information gain at each step while it grows the graph. The algorithm as Mitchell (1997) and Ojugo et al, (2012) is thus:

1. DT (Examples, Target_Attribute, Attributes)
2. //Data Attributes are feats to be tested. Target_Attribute are
3. //values predicted. Return is decision to correctly detect Example
4. Create a Root node of Graph
5. If Examples are positive, Return single_node Root with label = +
6. If Examples are negative, Return single_node Root with label = -

7. If Attribute is empty, Return single_node Root, with label = most
8. common value of Target_Attribute in Examples
9. Otherwise Begin
   a. A ← attribute from attributes that best* classifies Examples
   b. The decision attribute for Root ← A
   c. For each possible value v_i, of A,
   d. Add new branch to G below Root, corresponding to A = v_i
   e. Let Examples v_i be subset of Examples with value v_i for A
   f. IF Examples v_i = empty
   g. THEN add leaf to new branch with label = most common
   h. value of Target_Attribute in Examples
   i. Else below this new branch, add the subtree
10. IDA(Examples v_i, Traget_Attributes, Attributes – {A})
11. End
12. Return Root

Random Forest Algorithm as a decision tree predictor in which each individual is trained on partially, independently sample set of instances selected from the complete training dataset. Predicted output of a classified instance is the most frequent class output of the individual trees (Szor, 2005; McGraw and Morrisett, 2002; Mitchell, 1997).

Bayesian Belief Model describes probability distribution of a set of nodes on G by specifying a set of conditional independent assumptions along with a set of conditional probabilities. Thus, allows stating conditional assumptions that applies to a subset of nodes on the network by providing an intermediate and more tractable solution unlike Naïve Bayes that applies to each instance that assumptions of each graph attribute values are conditionally independent of the target value. Thus, the assumptions is that given target value of an instance, the probability of observing the interactions between nodes in the graph is the product of their probabilities from the individual attributes (Szor, 2005; Alpaydin, 2010; Mitchell, 1997, Harrington, 2012).

Entropy characterizes impurity of an arbitrary collection of nodes on G, which contains both activated (infected) and inactive (uninfected) node. The Entropy is a Boolean classification given by:

$$Entropy\ (E) \equiv\ -p_{\oplus} log_2\, p_{\oplus} -\ p_{\ominus} log_2\, p_{\ominus} \qquad (3)$$

Sample consists of n=25000 e-mail address from which we have normal/infected nodes to form G. Normal (inactive/p+) = 20000, infected (activated/p-) nodes where adversary plants viruses p- = 5000. To compute Entropy, we have:

$$Entropy\ (E) \equiv\ -\frac{20000}{25000} log_2 \frac{20000}{25000} - \frac{5000}{25000} log_2 \frac{5000}{25000}$$

$$E \equiv [-(0.8) log_2\,(0.8)] - [(0.2) log_2\,(0.2)] = 0.0775 + 0.1398 = 0.22$$

Information Gain is the expected reduction in entropy caused by partitioning the network according to its attributes (infected and uninfected) nodes. IG is info about target function value, given the value of another attribute A. IG of attribute (A) is given by Eq. 4. The Values(A) is set of all possible values of Attribute A, E_v is E subset of attributes A with value v. Our second is the expected entropy after partitioning with attribute A (sum of all entropies of each subset E_v weighted by fraction of $\frac{E_v}{E}$ of E_v).

$$Gain(E, A) \equiv Entropy(E) - \sum_{v \in Values(A)} \frac{|E_v|}{|E|} Entropy(E_v) \qquad (4)$$

$$\equiv Entropy - \sum_{v \in \{inactive, immunized\}} \frac{|E_v|}{|E|} Entropy(E_v)$$

$$Gain(E, A) \equiv 0.220 - \left\{\frac{8000}{25000} * 0.811\right\} - \left\{\frac{23000}{25000} * 0.921\right\}$$
$$= 0.220 - \{-0.587\} \equiv 0.220 + 0.58$$
$$= 0.807$$

Thus, we choose only top 80% of nodes that are most likely to be infected. IG is updated as below:

$$Gain(E, A) \equiv Gain(E, A) \pm \left[\frac{\sum_{i=o}^{n} Gain(X_i)}{n}\right]$$

4.4. Result Findings and Discussion

After training/testing, model results discovered that with the same amount of seed nodes (that is, viruses planted in the same number of nodes in this case, 5000nodes, on a network), the extent of the network that is blocked from virus attack is 22%; while 81% of the nodes are targeted before a complete network immunization is performed. However:

a. GA took 21seconds to find the solution after 98 iterations (best). CGANN was run 15-times and it found optima each time. Its convergence time that varied between 21seconds and 4 minutes and depends on how close the initial population is to the solution as well as on mutation applied to the individuals in the pool. The model is able to immunized 90% of the nodes before the virus eventually dies out.
b. IDA (at best) took 18seconds after 321 iterations. It was run 25 times and solution found each time on a range between 4seconds and 3minutes. In addition to the facts, the model is able to immunized 94% of the nodes before the virus eventually dies out
c. RFA arrived at solution 2.112seconds after 401 iterations. In addition to the facts as stated earlier on its extent and targeting, the model is able to immunized 97% of the nodes before the virus eventually dies out.

4.5. Rationale for Choice of Algorithms

The comparisons are as follows:

- Stochastic Model: are mostly inspired by evolution laws and biological population cum behaviors. They are heuristics that search a domain space for optimal solution to a task. They use hill-climbing method that are flexible, adaptive to changing states and suited for real-time application. GA guarantees high global convergence to

optimal point for multimodal tasks. It initializes with a random population, allocates increasing trials to regions of space found with high fitness and finds optimal in time. Its demerit is that they are not good with linear systems in that if the optimal is in a small region surrounded by regions of low fitness – the function becomes difficult to optimize.

- Gradient/Greedy Search: A number of different methods for optimizing well-behaved continuous functions have been developed which rely on using information about the gradient of the function to guide the direction of search. If the derivative of the function cannot be computed, because it is discontinuous, for example, these methods often fail. Such methods are generally referred to as *hill-climbing*. They can perform well on functions with only one peak (*unimodal* functions). But on functions with many peaks, (multimodal functions), they suffer from the problem that the first peak found will be climbed, and this may not be the highest peak. Having reached the top of a local maximum, no further progress can be made.
- Iterative Search is a combined random and gradient search that also employs an *iterated hill-climbing* search. Once one peak has been located, the hill-climb is started again, but with another, randomly chosen, starting point. This technique has the advantage of simplicity, and can perform well if the function does not have too many local maxima. However, since each random trial is carried out in isolation, no overall picture of the shape of the domain is obtained. As random search progresses, it continues to allocate its trials evenly over the search space. This means that it will still evaluate just as many points in regions found to be of low fitness as in regions found to be of high fitness.

5. Conclusion

Models have been successfully used today to determine epidemic spread of viruses. Many studies recently on the mathematical epidemiology is focusing on the analytic epidemic thresholds for varying propagation models and different families of network – seeking insight into the nature of such epidemic existence, its threshold and to unveil if such epidemic will continue to spread or eventually die out (Bougna et al, 2003; Barthelemy et al, 2005; Barabasi and Albert, 1999). Models serve as educational, predictive tools to compile knowledge about a task. They also serve as a new language to communicate hypotheses, investigate parameters crucial in estimation and help us gain better insight to a problem domain. Thus, their growth, development, sensitivity and failure analysis helps reflect on the theories and functioning of nature systems.

References

[1] Alpaydin, E., (2010). *Introduction to Machine Learning*, McGraw Hill publications, ISBN: 0070428077, New Jersey.

[2] Aspnes, J., Chang, K and Yampolskiy, A., (2005). *Inoculation strategies for victims of viruses and the sum-of-squares partition problem*. In *SODA*.

[3] Barabasi, A.L and Albert, R., (1999). *Emergence of scaling in random networks*. Science, 286, p23.

[4] Barthelemy, M., Barrat, A., Pastor-Satorras, R and Vespignani, A. (2005). *Dynamical patterns of epidemic outbreaks in complex heterogeneous networks*. Journal of Theoretical Biology, p54.

[5] Boguna, M., Pastor-Satorras, R and Vespignani, A., (2003). *Epidemic spreading in complex networks with degree correlations*. Statistical Mechanics of Complex Networks, p36.

[6] Cohen, R., Havlin, S and Ben-Avraham, D., (2003). *Efficient immunization strategies for computer networks and populations*. Phys Rev Letters, p232.

[7] Dezso, Z and Barabasi, A.L., (2002). *Halting viruses in scale-free networks*. Phys. Rev. E 66, p67.

[8] Filiol, E., (2005). *Computer Viruses: from Theory to Applications*, Springer, ISBN 10: 2287-23939-1.

[9] Ganesh, A., Massouli, L and Towsley, D., (2005). *The effect of network topology on the spread of epidemics*. In *IEEE INFOCOM*.

[10] Harrington, P., (2012). *Machine Learning in action*, Manning publications, ISBN: 9781617290183, NY.

[11] Kempe, D., Kleinberg, J and Tardos, E., (2003). *Maximizing the spread of influence through a social network*. In *SIGKDD*.

[12] Kermack, W and McKendrick, A., (1927). *A contribution to the mathematical theory of epidemics*. Proceedings Royal Society London.

[13] Mitchell, T.M., (1997). *Machine Learning*, McGraw Hill publications, ISBN: 0070428077, New Jersey.

[14] Newman, M.E., (2003). *The structure and function of complex networks*. SIAM Reviews, 45(2), p167.

[15] Ojugo, A., Eboka, A., Okonta, E., Yoro, R and Aghware, F., (2012). *GA rule-based intrusion detection system*, J. of Computing and Information Systems, 3(8), p1182.

[16] Ojugo, A.A., and Yoro, R., (2013a). *Computational intelligence in stochastic solution for Toroidal Queen task,* Progress in Intelligence Computing Applications, 2(1), 10.4156/pica.vol2.issue1.4, p46.

[17] Ojugo, A.A., Emudianughe, J., Yoro, R.E., Okonta, E.O and Eboka, A.O., (2013b). *Hybrid artificial neural network gravitational search algorithm for rainfall,* Progress in Intelligence Computing and Applications, 2(1), 10.4156/pica.vol2.issue1.2, p22.

[18] Pastor-Satorras, R and Vespignani, A., (2002). *Epidemics and immunization in scale-free networks*. Handbook of Graphs and Networks: From the Genome to the Internet.

[19] Singhal, P and Raul, N., (2012). *Malware detection module using machine learning algorithm to assist centralized security in Enterprise networks*, Int. J. Network Security and Applications, 4(1), doi: 10.5121/ijnsa.2012.4106, p61.

[20] Szor, P., (2005). *The Art of Computer Virus Research and Defense*, Addison Wesley Symantec Press. ISBN-10: 0321304543, New Jersey.

[21] Wang, Y., Chakrabarti, D., Wang, C and Faloutsos, C., (2003). *Epidemic spreading in real networks: An eigenvalue viewpoint*. In *SRDS*.

[22] Watts, D.J., (1999). *Networks, dynamics and the small world phenomenon*. American Journal of Sociology, 105, p234-245.