



Science

REDUCTION OF ACTIVE POWER LOSS & STATIC VOLTAGE STABILITY MARGIN ENHANCEMENT BY VIRAL SYSTEM ALGORITHM

Dr.K.Lenin ^{*1}

^{*1} Professor, Department of EEE, Prasad V.Potluri Siddhartha Institute of Technology, Kanuru,
Vijayawada, Andhra Pradesh -520007, India

Abstract

This paper presents Viral Systems Algorithm (VSA) for solving optimal reactive power problem. VSA have proven to be very efficient when dealing with problems of high complexity. The virus infection expansion corresponds to the feasibility region exploration, and the optimum corresponds to the organism lowest fitness value. Many available algorithms usually present weaknesses and cannot guarantee the optimum output for the problem in a bounded time. Projected Viral Systems Algorithm (VSA) has been tested on standard IEEE 30 bus test system and simulation results show clearly about the superior performance of the proposed Viral Systems Algorithm (VSA) in reducing the real power loss and static voltage stability margin (SVSM) index has been enhanced.

Keywords: Viral System; Transmission Loss; Reactive Power Problem.

Cite This Article: Dr.K.Lenin. (2017). “REDUCTION OF ACTIVE POWER LOSS & STATIC VOLTAGE STABILITY MARGIN ENHANCEMENT BY VIRAL SYSTEM ALGORITHM.” *International Journal of Research - Granthaalayah*, 5(12), 275-290. <https://doi.org/10.5281/zenodo.1134582>.

1. Introduction

Main objective of the Optimal reactive power dispatch problem is to minimize the real power loss and to enhance the voltage stability index. A variety of numerical techniques like the gradient method [1-2], Newton method [3] and linear programming [4-7] have been adopted to solve the optimal reactive power dispatch problem. Both the gradient and Newton methods have the complexity in controlling inequality constraints. If linear programming is applied, then the input-output function has to be articulated as a set of linear functions which predominantly lead to loss of accuracy. The difficulty of voltage stability and fall down, play a major role in power system planning and operation [8]. Global optimization has received wide-ranging research responsiveness, and enormous number of methods has been applied to solve this problem. Evolutionary algorithms such as genetic algorithm have been already proposed to solve the reactive power flow problem [9,10]. Evolutionary algorithm is a heuristic approach used for

minimization problems by utilizing nonlinear and non-differentiable incessant space functions. In [11], Genetic algorithm has been used to solve optimal reactive power flow problem. In [12], Hybrid differential evolution algorithm is proposed to perk up the voltage stability index. In [13], Biogeography Based algorithm is planned to solve the reactive power dispatch problem. In [14], a fuzzy based method is used to solve the optimal reactive power scheduling method. In [15], an improved evolutionary programming is used to solve the optimal reactive power dispatch problem. In [16], the optimal reactive power flow problem is solved by integrating a genetic algorithm with a nonlinear interior point method. In [17], a pattern algorithm is used to solve ac-dc optimal reactive power flow model with the generator capability limits. In [18], proposes a two-step approach to evaluate Reactive power reserves with respect to operating constraints and voltage stability. In [19], a programming based proposed approach used to solve the optimal reactive power dispatch problem. In [20], presents a probabilistic algorithm for optimal reactive power requirement in hybrid electricity markets with uncertain loads. This paper presents Viral Systems Algorithm (VSA) for solving optimal reactive power problem. VSA have proven to be very efficient when dealing with problems of high complexity. The virus infection [21-24] expansion corresponds to the feasibility region exploration, and the optimum corresponds to the organism lowest fitness value. Many available algorithms usually present weaknesses and cannot guarantee the optimum output for the problem in a bounded time. Projected Viral Systems Algorithm (VSA) has been tested on standard IEEE 30 bus test system and simulation results show clearly about the superior performance of the proposed Viral Systems Algorithm (VSA) in reducing the real power loss and static voltage stability margin (SVSM) index has been enhanced.

2. Voltage Stability Evaluation

2.1. Modal Analysis for Voltage Stability Evaluation

Modal analysis is one among best methods for voltage stability enhancement in power systems. The steady state system power flow equations are given by.

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} J_{p\theta} & J_{pv} \\ J_{q\theta} & J_{qv} \end{bmatrix} \begin{bmatrix} \Delta\theta \\ \Delta V \end{bmatrix} \quad (1)$$

Where

ΔP = Incremental change in bus real power.

ΔQ = Incremental change in bus reactive Power injection

$\Delta\theta$ = incremental change in bus voltage angle.

ΔV = Incremental change in bus voltage Magnitude

$J_{p\theta}$, J_{pv} , $J_{q\theta}$, J_{qv} jacobian matrix are the sub-matrixes of the System voltage stability is affected by both P and Q.

To reduce (1), let $\Delta P = 0$, then.

$$\Delta Q = [J_{qv} - J_{q\theta} J_{p\theta}^{-1} J_{pv}] \Delta V = J_R \Delta V \quad (2)$$

$$\Delta V = J^{-1} - \Delta Q \quad (3)$$

Where

$$J_R = (J_{QV} - J_{Q\theta} J_{P\theta}^{-1} J_{PV}) \quad (4)$$

J_R is called the reduced Jacobian matrix of the system.

2.2. Modes of Voltage Instability

Voltage Stability characteristics of the system have been identified by computing the Eigen values and Eigen vectors.

Let

$$J_R = \xi \Lambda \eta \quad (5)$$

Where,

ξ = right eigenvector matrix of J_R

η = left eigenvector matrix of J_R

Λ = diagonal eigenvalue matrix of J_R and

$$J_R^{-1} = \xi \Lambda^{-1} \eta \quad (6)$$

From (5) and (8), we have

$$\Delta V = \xi \Lambda^{-1} \eta \Delta Q \quad (7)$$

or

$$\Delta V = \sum_i \frac{\xi_i \eta_i}{\lambda_i} \Delta Q \quad (8)$$

Where ξ_i is the i th column right eigenvector and η the i th row left eigenvector of J_R .

λ_i is the i th Eigen value of J_R .

The i th modal reactive power variation is,

$$\Delta Q_{mi} = K_i \xi_i \quad (9)$$

where,

$$K_i = \sum_j \xi_{ij}^2 - 1 \quad (10)$$

Where

ξ_{ji} is the j th element of ξ_i

The corresponding i th modal voltage variation is

$$\Delta V_{mi} = [1/\lambda_i] \Delta Q_{mi} \quad (11)$$

If $|\lambda_i| = 0$ then the i th modal voltage will collapse .

In (10), let $\Delta Q = e_k$ where e_k has all its elements zero except the k th one being 1. Then,

$$\Delta V = \sum_i \frac{\eta_{1k} \xi_i}{\lambda_i} \quad (12)$$

η_{1k} k th element of η_1
V –Q sensitivity at bus k

$$\frac{\partial V_K}{\partial Q_K} = \sum_i \frac{\eta_{1k} \xi_1}{\lambda_1} = \sum_i \frac{P_{ki}}{\lambda_1} \quad (13)$$

3. Problem Formulation

The objectives of the reactive power dispatch problem is to minimize the system real power loss and maximize the static voltage stability margins (SVSM).

3.1. Minimization of Real Power Loss

Minimization of the real power loss (Ploss) in transmission lines is mathematically stated as follows.

$$P_{\text{loss}} = \sum_{\substack{k=1 \\ k=(i,j)}}^n g_k (V_i^2 + V_j^2 - 2V_i V_j \cos \theta_{ij}) \quad (14)$$

Where n is the number of transmission lines, g_k is the conductance of branch k, V_i and V_j are voltage magnitude at bus i and bus j, and θ_{ij} is the voltage angle difference between bus i and bus j.

3.2. Minimization of Voltage Deviation

Minimization of the voltage deviation magnitudes (VD) at load buses is mathematically stated as follows.

$$\text{Minimize VD} = \sum_{k=1}^{nl} |V_k - 1.0| \quad (15)$$

Where nl is the number of load busses and V_k is the voltage magnitude at bus k.

3.3. System Constraints

Objective functions are subjected to these constraints shown below.

Load flow equality constraints:

$$P_{Gi} - P_{Di} - V_i \sum_{j=1}^{nb} V_j \begin{bmatrix} G_{ij} & \cos \theta_{ij} \\ +B_{ij} & \sin \theta_{ij} \end{bmatrix} = 0, i = 1, 2, \dots, nb \quad (16)$$

$$Q_{Gi} - Q_{Di} - V_i \sum_{j=1}^{nb} V_j \begin{bmatrix} G_{ij} & \sin \theta_{ij} \\ +B_{ij} & \cos \theta_{ij} \end{bmatrix} = 0, i = 1, 2, \dots, nb \quad (17)$$

where, nb is the number of buses, PG and QG are the real and reactive power of the generator, PD and QD are the real and reactive load of the generator, and G_{ij} and B_{ij} are the mutual conductance and susceptance between bus i and bus j.

Generator bus voltage (VGi) inequality constraint:

$$V_{Gi}^{\min} \leq V_{Gi} \leq V_{Gi}^{\max}, i \in ng \quad (18)$$

Load bus voltage (VLi) inequality constraint:

$$V_{Li}^{\min} \leq V_{Li} \leq V_{Li}^{\max}, i \in nl \quad (19)$$

Switchable reactive power compensations (QCi) inequality constraint:

$$Q_{Ci}^{\min} \leq Q_{Ci} \leq Q_{Ci}^{\max}, i \in nc \quad (20)$$

Reactive power generation (QGi) inequality constraint:

$$Q_{Gi}^{\min} \leq Q_{Gi} \leq Q_{Gi}^{\max}, i \in ng \quad (21)$$

Transformers tap setting (Ti) inequality constraint:

$$T_i^{\min} \leq T_i \leq T_i^{\max}, i \in nt \quad (22)$$

Transmission line flow (SLi) inequality constraint:

$$S_{Li}^{\min} \leq S_{Li} \leq S_{Li}^{\max}, i \in nl \quad (23)$$

Where, nc, ng and nt are numbers of the switchable reactive power sources, generators and transformers.

4. Viral Systems Algorithm (VSA)

The concept of viruses' analogies has been mainly used as part of genetic algorithms. It use of genetic algorithms which make use of a virus evolutionary theory (GAV), and an algorithm based on the conception of horizontal evolution caused by virus infections. GAV is carried out by attacking a chromosome by a number of viruses, and having the genes of the chromosome recombined by the attack. The infection is allowed when the evaluation value goes up, but it falls into local minima easily. In order to escape from these local minima, an infection which makes the evaluation value worse in a small rate under small probability is allowed as well.

Viruses, viral infections and organism antigenic response Viruses are intracellular parasites shaped by nucleic acids, such as DNA or RNA, and proteins. The protein generates a capsule, called a capsid, where the nucleic acid is located. The capsid plus the nucleic acid shape the nucleus capsid, defining the virus. There are a high number of different types of viruses, each of them showing a different and autonomous behavior. However, the simplest and most common type of virus is the phage, a type of virus infecting bacteria. Figure 1 depicts a traditional representation for such structure.

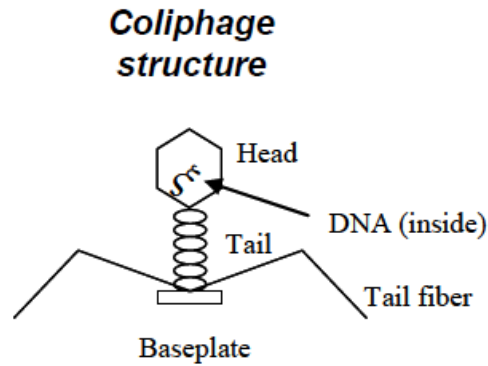


Figure 1: Coli phage structure

One of the main characteristics of viruses is the replication mechanism. The phage (a common type of virus) does follow lytic replication process. Left side of Figure 2 depicts the biological evolution of the virus infection following the next steps:

1. The virus is adhered to the border of the bacterium. After that, the virus penetrates the border being injected inside this one, (1) and (2) in Figure 2.
2. The infected cell stops the production of its proteins, beginning to produce the phage proteins. So, it starts to replicate copies of the virus nucleus-capsids, (3a) in Figure 2.
3. After replicating a number of nucleus-capsids, the bacterium border is broken, and new viruses are released, (4a), which can infect near cells, (1), in Figure 2.

The life cycle of the virus can be developed in more than one step. Some viruses are capable of lodging in cells giving rise to the lysogenic replication.

This case is shown in the right side of Figure 2. It follows:

1. The virus infects the host cell, being lodged in its genome, (3b) in Figure 2 where a pro-phage (mutation) can arise.
2. The virus remains hidden inside the cell during a while until it is activated by any cause, for example ultraviolet irradiation or X-rays, (i) in Figure 2. During such time the cell reproduces itself normally.
3. The replication of cells altered, with proteins from the virus, starts. So, lysogenic replication produces the genome alteration of the cell leading to a procedure similar to a mutation process.

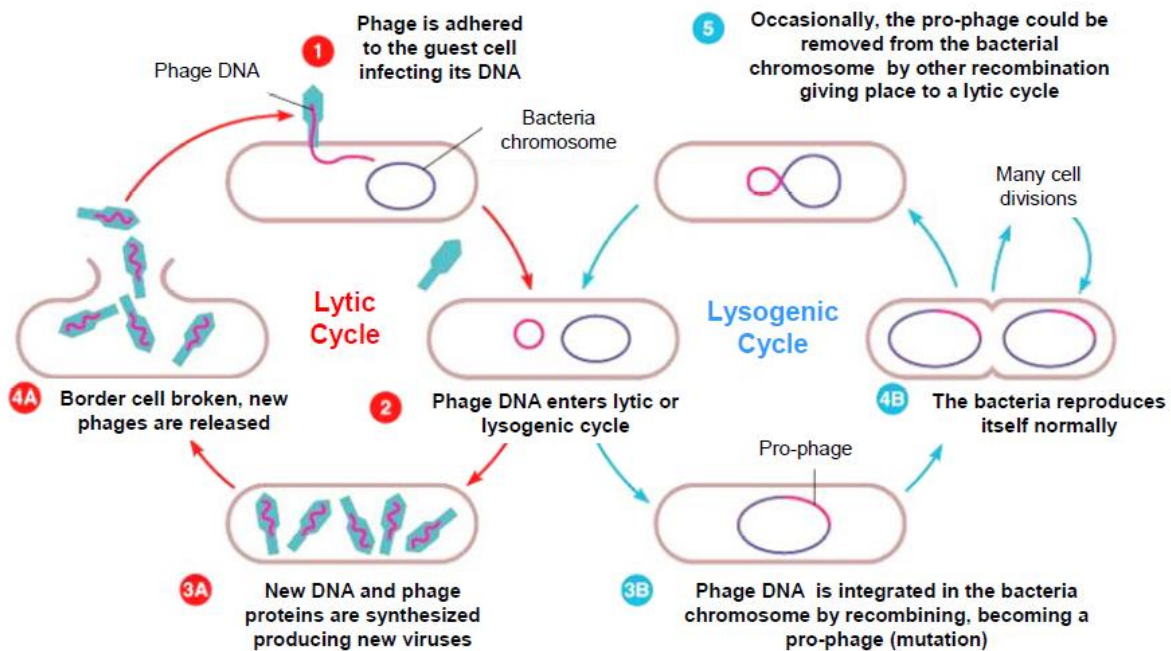


Figure 2: Lytic (left) and lysogenic (right) replication of viruses

However, some viruses have the property of leading an antigenic response in the infected organism. In these situations an immune response is originated causing the creation of antibodies. This is the specific case of phages. So VS follows an exploration process that combines lytic replication to search the neighbourhood of the existing solutions (which is one of the main features of Tabu Search) and a mutation process (which is a characteristic of Genetic Algorithms).

4.1. Computational Description of Viral Systems

VS are an iterative method that runs during a maximum number of iterations, or until the optimum are reached in case of a known optimum. VS define the clinical picture of an infected population as the description of all the cells infected by viruses. Computationally, it includes the encoding of the solution that is being explored (the genome of the cell that is infected, in biological terms) and the number of nucleus-capsids being replicated, NR, (for lytic replications) or the number of hidden generations, IT, (for lysogenic replications). Thus the state of each virus is given by the three-tuple “cell genome-NR-IT”. All these three-tuples corresponding to the cells infected by viruses define the clinical picture.

Every cell infected by a virus develops a lytic or a lysogenic replication according to a probability plt (for lytic replication) or plg otherwise, where $plt + plg = 1$. In case of lysogenic replications, the activation of the mutation process takes place after a limit of iterations has passed (LIT). The value of LIT depends on the cell's health conditions, so a healthy cell (high value of the objective function being minimized, $f(x)$) will have a low infection probability, i.e. the value of LIT will be higher. An unhealthy cell, on the contrary, will have a lower value of LIT. In case of lytic replications, a number of virus replications (NR) is calculated for each

iteration as a function of a binomial variable, Z , adding its value to the current NR in the clinical picture. Z is calculated using a Binomial distribution given by the maximum level of nucleus-capsids replicated, LNR, and the single probability of one replication, pr ; $Z = \text{Bin}(\text{LNR}, pr)$. LNR represents the limit to break the cell border and to release the lodged viruses. As in the lysogenic cycle, the value of LNR is set depending on the value of the objective function being minimized, $f(x)$.

Thus cells with higher $f(x)$ have lower probability of getting infected, and therefore the value of LNR will be higher. Two infections process have been defined for VS: massive infections where a devastating infection reaches a high number of cells, and selective infection where a parsimonious infection following a like-elistist process takes place. An example of the first case is the Ebola virus with a rapid and massive infection that very often produces the death of the patient in a few days, and an example of the second one is the HIV virus, which through a step-by-step evolution destroys the immune system during a process that can take years.

4.2. Massive Infection

Once a massive infection takes place and viruses are liberated inside the organism, each liberated virus will have a probability, pi , of infecting other new cells of the neighbourhood. If the neighbourhood cardinality of x is defined as $|V(x)|$, the number of cells infected by the virus in the neighbourhood can be calculated as a binomial distribution given by $Y = \text{Bin}(|V(x)|, pi)$. On the other hand, in order to defend itself from the growth of the viral infection, the Organism (the set of cells) responds by releasing antigens.

In the clinical picture, each one of the infected cells generates antibodies according to a Bernoulli probability distribution $A(x) = \text{Ber}(pan)$, where pan is the unitary probability of generating antibodies by the cell x in the clinical picture. Hence, the total population of infected cells generating antibodies is characterized by a Binomial distribution of parameters: the size of the clinical picture, n , and the probability of generating antibodies, pan : $A(\text{population}) = \text{Bin}(n, pan)$. Also, the antigenic response for every cell in the neighbourhood of an active virus is estimated as a Bernoulli probability distribution given by the probability of generating antibodies, pan : $A(x') = \text{Ber}(pan) : x' \in V(x)$.

Therefore, the total number of cells with antibodies in the neighbourhood will follow a Binomial probability distribution given by the total size of the neighbourhood for all the active viruses, $|V(x)|$, and the probability of generating antibodies, pan : $A = \text{Bin}(|V(x)|, pan)$. In this situation, a Markovian Process defines the evolution of the clinical picture

Let, $\pi = (\pi_0, \pi_1, \dots, \pi_{LNR})$ be the probability of a cell with 0, 1, ..., LNR nucleus-capsids replicated. Equations (1-3) are satisfied in steady state.

$$\pi = \pi.p \tag{24}$$

$$\pi_J = \frac{1}{1-p_0} \left(\sum_{K=0}^{J-1} P_{J-K} \cdot \pi_K \right) \forall J = 1, 2, \dots, LNR \tag{25}$$

$$\pi_0 + \pi_1 + \dots + \pi_{LNR} = 1 \tag{26}$$

To ensure computational control of the infection evolution, we can give (27) as an adequate value for pan .

$$P_{an} > \frac{n \cdot \pi_{LNR} \cdot (P_i \cdot \left| \frac{\leftarrow}{\rightarrow} \right| - 1)}{n \cdot \pi_{LNR} \cdot (P_i \cdot \left| \frac{\leftarrow}{\rightarrow} \right| - 1) + n} \tag{27}$$

Where $|V(x)|$ is the average neighbourhood size for a specific problem. However, we do not use the same value of pan for all the cells. In fact, a higher value of $f(x)$ implies a healthy cell and therefore this cell will have a higher probability of developing an antigenic response.

On the contrary, a cell with a low value of $f(x)$ represents an unhealthy cell with a lower probability of developing an antigenic response. Thus we define for each cell its specific $pan(x)$. Figure 3 describes the algorithmic process. The original state is depicted by the clinical picture on the left-hand side. The viruses reaching the level of nucleus capsids (LNR) break the border and start infecting new cells in their neighbourhoods. The response of the Organism is characterized by the antigenic response, liberating space in the clinical picture, and by creating antibodies in cells located in the virus neighbourhood. This situation leads to a new clinical picture, depicted on the right-hand side of the figure, with new infected cells lodging viruses.

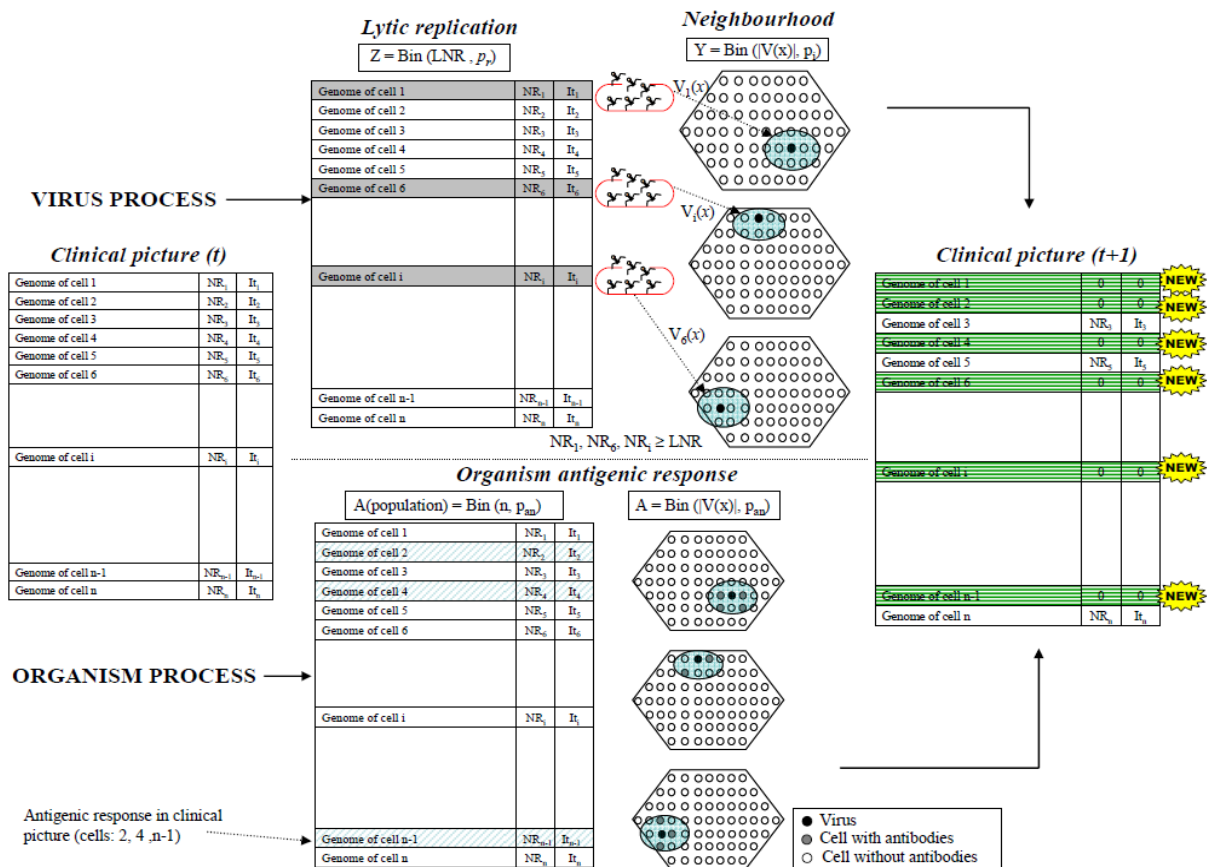


Figure 3: Algorithm for lytic replication case in massive infections

4.3. Selective Infection

Once a selective infection takes place and viruses are liberated inside the organism, the virus selects a cell with a low value of $f(x)$ in the neighbourhood. However, the virus will not be able to infect those cells that have developed antigens. Higher values of $f(x)$ imply healthy cells and therefore cells that have a higher probability of developing antigenic responses. On the contrary, cells with low value of $f(x)$ imply unhealthy cells with lower probability of developing antigenic responses. This effect is represented by the previously introduced hyper geometric function. Then, if the probability of generating antibodies for the case of cell x is $pan(x), nA(x)$ is defined as a Bernoulli random variable: $A(x) = Ber(pan(x))$. If cell x generates antibodies, the cell is not infected and it is therefore not included in the new clinical picture. For recording this clinical picture we use the original cell (that was infected by the virus and that reached the LNR limit) and we initiate a lysogenic cycle for that cell. Figure 4 defines the algorithm evolution for the infection. The initial state is on the left-hand side: the virus process starts with viruses breaking the border and starting the infection of new cells in their neighbourhoods. Each virus selects the most promising cell, which is the least healthy cell. The Organism process is characterized by the probability of antigenic response in the least healthy cell. Those cells developing antibodies are not infected. Finally, the interaction (right hand side of the figure) defines the new clinical picture, with new infected cells lodging viruses. The cells generating antibodies follow a new lysogenic replication.

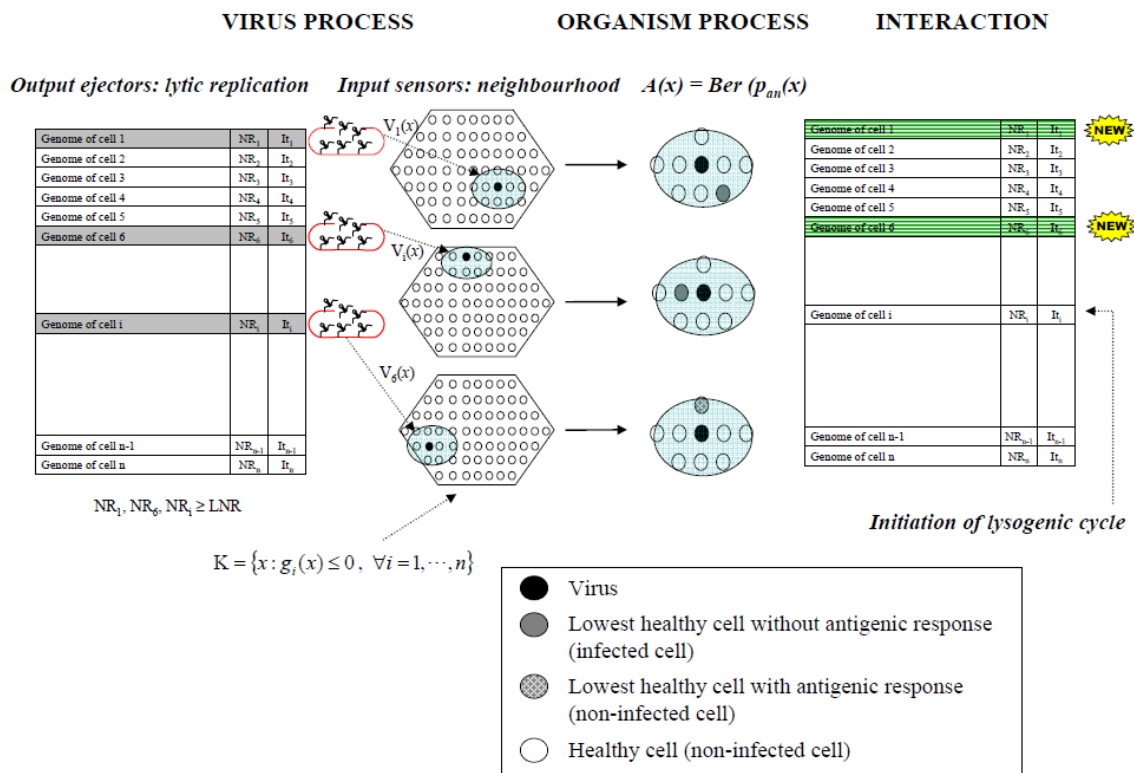


Figure 4: Algorithm for lytic replication case in selective infection

The main difference between massive and selective infection processes is the infection activity every time the algorithm makes iteration. In the selective infection case, only a single cell is infected whereas in the massive one, all cells are infected at each iteration. However, lytic and lysogenic replications are the same for both processes. Therefore, the differences in the pseudo code of the massive process respect to the selective process only appear in the main procedure.

Virus System Algorithm for Solving Optimal Reactive Power Problem

- 1) Determine the required inputs for the problem
- 2) Determine the value of Viral Systems parameter, i.e. the maximum number of iterations (ITER), the size of clinical picture (POB), the probability of occurring lytic replication (plt), the probability of infecting a neighboring cell (π_i), the probability of replicating a virus (pr), probability of producing antigene (pan), the constant multiplier for computing LNR (LNR0), and the constant multiplier for computing LIT (LIT0).
- 3) Create infected cells as many as POB and put them in the initial clinical picture. Each infected cell has a unique genome, has NR equals 0 and IT equals 0.
- 4) Start the first iteration; $t = 1$.
- 5) Start checking the objective function value and antigene production for the first infected cell, $m = 1$.
- 6) Compute the objective function value for cell m . Update the best objective function value found so far, if the objective function value of cell m is better (lower) than the previously recorded.
- 7) Check the ability of cell m to produce antigene, based on the probability of producing antigene is pan. If cell m is able to produce antigene, the cell is deleted from the clinical picture.
- 8) If $m \neq POB$, then $m = (m+1)$ and go back to step 6. Otherwise, go to the next step.
- 9) Start the replication for the first cell in the clinical picture, $m = 1$.
- 10) Determine the replication type for the virus inside cell m , based on the probability of occurring lytic replication is plt. If the replication type is lytic, then go to the next step. Otherwise, go to Step 15 for lysogenic replication.
- 11) Update the NR of cell m ; $NR = NR + Z$.
- 12) Compute the LNR for cell m . If its NR is larger than or equal to LNR, then go to the next step. It means, the replicated viruses break the cell's border. Otherwise, go to Step 19.
- 13) Delete cell m from clinical picture. Start infecting neighboring cells, based on the probability of infecting a neighboring cell is π_i .
- 14) For each infected neighboring cell, check whether it is able to produce antigene. Keep the neighboring cells that are unable to produce antigene. Go to Step 18.
- 15) Update the IT of cell m ; $IT = IT + 1$.
- 16) Compute the LIT for cell m . If LIT is larger than or equal to LIT, then go to the next step. It means, the virus is activated and thus it will mutate the cell's genome. Otherwise, go to Step 19.
- 17) Delete cell m from clinical picture. Perform the mutation procedure by selecting the first gene and another gene randomly. Swap those genes and conform the last gene to the new first gene.
- 18) Store either neighboring cells or mutated cell in a temporary storage.
- 19) If $m \neq POB$, then $m = (m+1)$ and go back to step 10. Otherwise, go to the next step.

- 20) If there is any duplication between the genome of cells in the temporary storage, remove the duplicates. If there is any duplication with the cells in the clinical picture, delete the duplicates as well.
- 21) Insert the cells in the temporary storage into the clinical picture. If the number of cells in the temporary storage is larger than POB, then delete all cells in the clinical picture and select cells from the temporary storage starting from the best to enter the new clinical picture.
- 22) If $t \neq \text{ITER}$, then $t = (t+1)$ and go back to step 5. Otherwise, the algorithm stops with the best solution found is the one whose objective function value is recorded.

5. Simulation Results

The efficiency of the proposed Viral Systems Algorithm (VSA) is demonstrated by testing it on standard IEEE-30 bus system. The IEEE-30 bus system has 6 generator buses, 24 load buses and 41 transmission lines of which four branches are (6-9), (6-10), (4-12) and (28-27) - are with the tap setting transformers. The lower voltage magnitude limits at all buses are 0.95 p.u. and the upper limits are 1.1 for all the PV buses and 1.05 p.u. for all the PQ buses and the reference bus. The simulation results have been presented in Tables 1, 2, 3 & 4. And in the Table 5 shows the proposed algorithm powerfully reduces the real power losses when compared to other given algorithms. The optimal values of the control variables along with the minimum loss obtained are given in Table 1. Corresponding to this control variable setting, it was found that there are no limit violations in any of the state variables.

Table 1: Results of VSA – ORPD optimal control variables

Control Variables	Variable Setting
V1	1.022
V2	1.028
V5	1.031
V8	1.029
V11	1.000
V13	1.026
T11	1.00
T12	1.00
T15	1.00
T36	1.00
Q10	2
Qc12	2
Qc15	2
Qc17	0
Qc20	2
Qc23	2
Qc24	3
Qc29	2
Real power loss	4.2348
SVSM	0.2478

Optimal Reactive Power Dispatch (ORPD) problem together with voltage stability constraint problem was handled in this case as a multi-objective optimization problem where both power loss and maximum voltage stability margin of the system were optimized simultaneously. Table 2 indicates the optimal values of these control variables. Also it is found that there are no limit violations of the state variables. It indicates the voltage stability index has increased from 0.2478 to 0.2486, an advance in the system voltage stability. To determine the voltage security of the system, contingency analysis was conducted using the control variable setting obtained in case 1 and case 2. The Eigen values equivalents to the four critical contingencies are given in Table 3. From this result it is observed that the Eigen value has been improved considerably for all contingencies in the second case.

Table 2: Results of VSA -Voltage Stability Control Reactive Power Dispatch (VSCRPD)
Optimal Control Variables

Control Variables	Variable Setting
V1	1.038
V2	1.036
V5	1.042
V8	1.030
V11	1.001
V13	1.030
T11	0.090
T12	0.090
T15	0.090
T36	0.090
Qc10	3
Qc12	2
Qc15	2
Qc17	3
Qc20	0
Qc23	2
Qc24	2
Qc29	3
Real power loss	4.9891
SVSM	0.2486

Table 3: Voltage Stability under Contingency State

Sl.No	Contingency	ORPD Setting	VSCRPD Setting
1	28-27	0.1419	0.1434
2	4-12	0.1642	0.1650
3	1-3	0.1761	0.1772
4	2-4	0.2022	0.2043

Table 4: Limit Violation Checking Of State Variables

State Variables	Limits		ORPD	VSCRPD
	Lower	Upper		
Q1	-20	152	1.3422	-1.3269
Q2	-20	61	8.9900	9.8232
Q5	-15	49.92	25.920	26.001
Q8	-10	63.52	38.8200	40.802
Q11	-15	42	2.9300	5.002
Q13	-15	48	8.1025	6.033
V3	0.95	1.05	1.0372	1.0392
V4	0.95	1.05	1.0307	1.0328
V6	0.95	1.05	1.0282	1.0298
V7	0.95	1.05	1.0101	1.0152
V9	0.95	1.05	1.0462	1.0412
V10	0.95	1.05	1.0482	1.0498
V12	0.95	1.05	1.0400	1.0466
V14	0.95	1.05	1.0474	1.0443
V15	0.95	1.05	1.0457	1.0413
V16	0.95	1.05	1.0426	1.0405
V17	0.95	1.05	1.0382	1.0396
V18	0.95	1.05	1.0392	1.0400
V19	0.95	1.05	1.0381	1.0394
V20	0.95	1.05	1.0112	1.0194
V21	0.95	1.05	1.0435	1.0243
V22	0.95	1.05	1.0448	1.0396
V23	0.95	1.05	1.0472	1.0372
V24	0.95	1.05	1.0484	1.0372
V25	0.95	1.05	1.0142	1.0192
V26	0.95	1.05	1.0494	1.0422
V27	0.95	1.05	1.0472	1.0452
V28	0.95	1.05	1.0243	1.0283
V29	0.95	1.05	1.0439	1.0419
V30	0.95	1.05	1.0418	1.0397

Table 5: Comparison of Real Power Loss

Method	Minimum Loss
Evolutionary programming [25]	5.0159
Genetic algorithm [26]	4.665
Real coded GA with Lindex as SVSM [27]	4.568
Real coded genetic algorithm [28]	4.5015
Proposed VSA method	4.2348

6. Conclusion

In this paper, a new approach Viral Systems Algorithm (VSA) successfully solved optimal reactive power problem. VSA have proven to be very efficient when dealing with problems of high complexity. The virus infection expansion corresponds to the feasibility region exploration, and the optimum corresponds to the organism lowest fitness value. Many available algorithms usually present weaknesses and cannot guarantee the optimum output for the problem in a bounded time. The projected Viral Systems Algorithm (VSA) has been tested on standard IEEE 30 bus test system and simulation results shows clearly about the superior performance of the proposed Viral Systems Algorithm (VSA) in reducing the real power loss and static voltage stability margin (SVSM) index has been enhanced.

References

- [1] O.Alsac, B. Scott, "Optimal load flow with steady state security", IEEE Transaction. PAS -1973, pp. 745-751.
- [2] Lee K Y ,Paru Y M , Ortiz J L –A united approach to optimal real and reactive power dispatch , IEEE Transactions on power Apparatus and systems 1985: PAS-104 : 1147-1153
- [3] A.Monticelli , M .V.F Pereira ,and S. Granville , "Security constrained optimal power flow with post contingency corrective rescheduling" , IEEE Transactions on Power Systems :PWRS-2, No. 1, pp.175-182.,1987.
- [4] Deeb N, Shahidehpur S.M, Linear reactive power optimization in a large power network using the decomposition approach. IEEE Transactions on power system 1990: 5(2) : 428-435
- [5] E. Hobson ,'Network consrained reactive power control using linear programming, ' IEEE Transactions on power systems PAS -99 (4) ,pp 868=877, 1980
- [6] K.Y Lee, Y.M Park, and J.L Ortiz, "Fuel –cost optimization for both real and reactive power dispatches", IEE Proc; 131C,(3), pp.85-93.
- [7] M.K. Mangoli, and K.Y. Lee, "Optimal real and reactive power control using linear programming" , Electr.Power Syst.Res, Vol.26, pp.1-10,1993.
- [8] C.A. Canizares, A.C.Z.de Souza and V.H. Quintana, "Comparison of performance indices for detection of proximity to voltage collapse ," vol. 11. no.3, pp.1441-1450, Aug 1996.
- [9] K.Anburaja, "Optimal power flow using refined genetic algorithm", Electr.Power Compon.Syst , Vol. 30, 1055-1063,2002.
- [10] D. Devaraj, and B. Yeganarayana, "Genetic algorithm based optimal power flow for security enhancement", IEE proc-Generation.Transmission and. Distribution; 152, 6 November 2005.
- [11] A. Berizzi, C. Bovo, M. Merlo, and M. Delfanti, "A ga approach tocompare orpf objective functions including secondary voltage regulation,"Electric Power Systems Research, vol. 84, no. 1, pp. 187 – 194, 2012.
- [12] C.-F. Yang, G. G. Lai, C.-H.Lee, C.-T. Su, and G. W. Chang, "Optimalsetting of reactive compensation devices with an improved voltagestability index for voltage stability enhancement," International Journalof Electrical Power and Energy Systems, vol. 37, no. 1, pp. 50 – 57, 2012.
- [13] P. Roy, S. Ghoshal, and S. Thakur, "Optimal var control for improvementsin voltage profiles and for real power loss minimization usingbiogeography based optimization," International Journal of ElectricalPower and Energy Systems, vol. 43, no. 1, pp. 830 – 838, 2012.
- [14] B. Venkatesh, G. Sadasivam, and M. Khan, "A new optimal reactivepower scheduling method for loss minimization and voltage stabilitymargin maximization using successive multi-objective fuzzy lp technique,"IEEE Transactions on Power Systems, vol. 15, no. 2, pp. 844 –851, may 2000.

- [15] W. Yan, S. Lu, and D. Yu, "A novel optimal reactive power dispatch method based on an improved hybrid evolutionary programming technique," IEEE Transactions on Power Systems, vol. 19, no. 2, pp. 913–918, may 2004.
- [16] W. Yan, F. Liu, C. Chung, and K. Wong, "A hybrid genetic algorithm interior point method for optimal reactive power flow," IEEE Transactions on Power Systems, vol. 21, no. 3, pp. 1163–1169, aug. 2006.
- [17] J. Yu, W. Yan, W. Li, C. Chung, and K. Wong, "An unfixed piecewise optimal reactive power-flow model and its algorithm for ac-dc systems," IEEE Transactions on Power Systems, vol. 23, no. 1, pp. 170–176, feb. 2008.
- [18] F. Capitanescu, "Assessing reactive power reserves with respect to operating constraints and voltage stability," IEEE Transactions on Power Systems, vol. 26, no. 4, pp. 2224–2234, nov. 2011.
- [19] Z. Hu, X. Wang, and G. Taylor, "Stochastic optimal reactive power dispatch: Formulation and solution method," International Journal of Electrical Power and Energy Systems, vol. 32, no. 6, pp. 615–621, 2010.
- [20] A. Kargarian, M. Raoofat, and M. Mohammadi, "Probabilistic reactive power procurement in hybrid electricity markets with uncertain loads," Electric Power Systems Research, vol. 82, no. 1, pp. 68–80, 2012.
- [21] Saito, S. 2003: A Genetic Algorithm by Use of Virus Evolutionary Theory for Combinatorial Problems. Optimization and Optimal Control, Vol.1, World Scientific Publishing Co.; 10: 251-268.
- [22] Cutello V., Nicosia G., Pavone M. 2007: An Immune Algorithm with Stochastic Aging and Kullback Entropy for the Chromatic Number Problem. Journal of Combinatorial Optimization 14(1), 9-33.
- [23] Kubota, N., Fukuda, T., Shimojima, K. 1996: Virus-evolutionary genetic algorithm for a self-organizing manufacturing system. Computers and Industrial Engineering 30 (4), 1015-1026.
- [24] Cutello V., Nicosia G., Pavone M. And Timmis J. 2007: An Immune Algorithm for Protein Structure Prediction on Lattice Models. IEEE Transaction on Evolutionary Computation 11(1), 101-117.
- [25] Wu Q H, Ma J T. "Power system optimal reactive power dispatch using evolutionary programming", IEEE Transactions on power systems 1995; 10(3): 1243-1248 .
- [26] S. Durairaj, D. Devaraj, P. S. Kannan, "Genetic algorithm applications to optimal reactive power dispatch with voltage stability enhancement", IE(I) Journal-EL Vol 87, September 2006.
- [27] D. Devaraj, "Improved genetic algorithm for multi – objective reactive power dispatch problem", European Transactions on electrical power 2007; 17: 569-581.
- [28] P. Aruna Jeyanthi and Dr. D. Devaraj "Optimal Reactive Power Dispatch for Voltage Stability Enhancement Using Real Coded Genetic Algorithm", International Journal of Computer and Electrical Engineering, Vol. 2, No. 4, August, 2010 1793-8163.

*Corresponding author.

E-mail address: gklenin@ gmail.com