Contents lists available at SciVerse ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Virus detection using clonal selection algorithm with Genetic Algorithm (VDC algorithm)

Suha Afaneh[a], Raed Abu Zitar[b,*], Alaa Al-Hamami[c]

[a] Department of Computer Sciences, Isra University, Amman, Jordan
[b] College of Information Technology, American University of Madaba, Jordan
[c] Department of Computer Sciences, Amman Arab University, Amman, Jordan

ARTICLE INFO

Article history:
Received 12 January 2012
Received in revised form 18 July 2012
Accepted 7 August 2012
Available online 21 August 2012

Keywords:
Artificial immune system
Virus detection algorithm
Clonal selection

ABSTRACT

This paper presents a novel approach for computer viruses detection based on modeling the structures and dynamics of real life paradigm that exists in the bodies of all living creatures. It aims to develop an algorithm based on the concept of the artificial immune system (AIS) for the purpose of detecting viruses. The algorithm is called Virus Detection Clonal algorithm (VDC), and it is derived from the clonal selection algorithm. The VDC algorithm consists of three basic steps: cloning, hyper-mutation and stochastic re-selection. In later stage, the developed VDC algorithm is subjected to validation, which consists of two phases; learning and testing. Two main parameters are determined; one of them is setting the number of signatures per clone (*Fat*), while the other defines the hypermutation probability (*Pm*). Later on, the Genetic Algorithm (GA) is used as a tool, to improve the developed algorithm by searching the values of the main parameters (*Fat* and *Pm*) to reproduce better results. The results have shown that the detection rate of viruses, by using the developed algorithm, is 94.4%, whereas the detection rate of false positives has reached 0%. These percentages indicate that the VDC algorithm is sufficient and usable in this field. Moreover, the results of employing the GA to optimize the VDC algorithm have shown an improvement in the detection speed of the algorithm.

1. Introduction

Different artificial intelligence based techniques are used nowadays in all areas of computer security [1]. Techniques such as swarm intelligence, Genetic Algorithms, and ant colony optimization have different applications in pattern classification and image and signal processing [2–4]. The artificial immune system (AIS), on the other hand, is very similar to those paradigms in structure and mechanism, however, it is quite recent, and has not been matured yet. The AIS has been applied in different fields, most notably in computer viruses' detection field. The protection against viruses is becoming more and more difficult day after day, and they constitute a threat for every one who uses computers. The viruses' intelligence is escalating by the time, and their signatures are changing continuously [5,6]. That has made the anti-viruses mission more difficult [7]. The (AIS) has several concepts: clonal selection, negative selection and network immune theory. This paper proposes the (VDC) algorithm which is inspired by the clonal selection algorithm and more precisely by the CLONALG [8] in detecting viruses.

Studies have shown that 25% of people using computers are infected by some sort of malwares, while the commercial PC sector is suffering from around the half of this percentage [9]. The simplest and the most common method to protect networks from the viral attacks is to use the signature technology.

This paper should offer a helping hand by proposing a Virus Detection Clonal (VDC) algorithm then optimizing the parameters using the GA, the VDC algorithm is a modern field, despite the fact that the virus issue is an aged issue. However, the problem we are solving can be considered as a growing problem because it affects every individual that uses computers.

The Negative Selection Algorithm (the self-non-self algorithm) has been used for virus detection [10–12,1], but the clonal selection algorithm has not been used yet with this type of application, after making a wide web search and investigating a wide range of specialized journals, it has been found that applying the clonal selection algorithm is a brand new contribution. The clonal selection principle describes the approach of an immune response to an antigenic stimulus. Which can be explained as the following: only the cells that recognize the antigen do proliferate and are selected against those that do not. These generated B-cells, which are copies of their parents, are mutated. When the antibody strongly matches the antigen, then these B-cells will be stimulated to produce clones of themselves [13]. In this paper the antigens represent the

[*] Corresponding author.
E-mail address: rzitar@nyit.edu (R.A. Zitar).

1568-4946/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.asoc.2012.08.034

computer viruses in the infected files and the antibodies represent the signatures. The signatures with high matching values (fitness) are selected to the cloning, the hypermutation and the reselection processes; so that the cloning produces copies of the signatures with Best fitness, then they are mutated to provide the ability of detecting viruses which are different in some characters (genes), even if these viruses have not attacked previously (just like the adaptive defense in the Immune System). In this research, the reselection is stochastically added to the clonal selection algorithm in order to guarantee choosing the best mutated signatures.

2. The proposed VDC algorithm

The research consists of two stages (MATLAB 7.1 is used); first, the design and implementation of the Virus Detection Clonal (VDC) algorithm, second, the validation of the VDC algorithm.

2.1. The design and implementation of the VDC algorithm

Fig. 1 illustrates the flowchart of the VDC algorithm. The pseudo code of the VDC algorithm is illustrated at Fig. 2.

After loading the Signatures' pool and the files' pool, the loop condition is defined as the *Learning Gen* parameter, which is determined in Table 2. This loop goes through the 3 main steps: Cloning (making copies of the signatures with highest fitness), Hypermutation (the mechanism of making random changes to the virus signatures with higher fitness) and Reselection (choosing the next generation of signatures according to their fitness stochastically). The fitness is calculated according to Eq. (1) below.

$$f(x) = f_0(x) + \delta \sum_{i=1}^{z} \text{match function}(x, y_i) + \sum_{j=1}^{t} D_j \qquad (1)$$

where: $f_0(x)$: the initial fitness for signature x and it is a random number determined in the initialization of the algorithm when loading the signatures' pool.

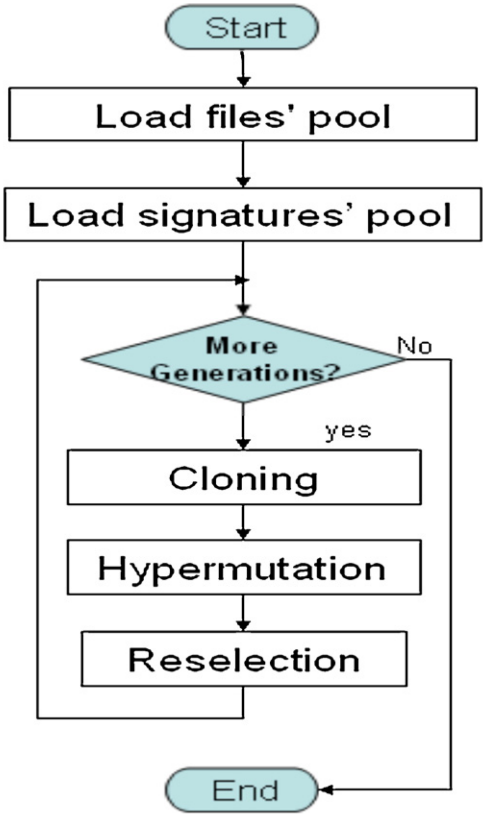
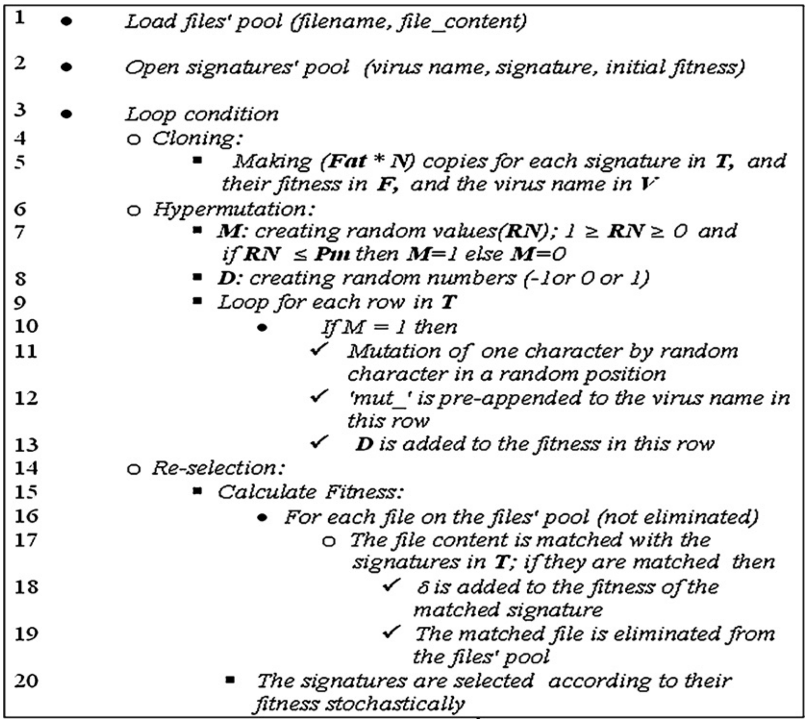


Fig. 1. The VDC algorithm flowchart.

y_i: the ith file.
δ: a multiplying factor with a value of 10.
z: the number of all files in the files' pool.
D_j: if the mutation is performed on the signature j then $D_j = (0, 1$ or $-1)$ uniformly random.



```
• Load files' pool (filename, file_content)
• Open signatures' pool (virus name, signature, initial fitness)
• Loop condition
  o Cloning:
      ▪ Making (Fat * N) copies for each signature in T, and
        their fitness in F, and the virus name in V
  o Hypermutation:
      ▪ M: creating random values(RN); $1 \geq RN \geq 0$ and
        if $RN \leq Pm$ then $M=1$ else $M=0$
      ▪ D: creating random numbers (-1or 0 or 1)
      ▪ Loop for each row in T
          • If $M = 1$ then
              ✓ Mutation of one character by random
                character in a random position
              ✓ 'mut_' is pre-appended to the virus name in
                this row
              ✓ D is added to the fitness in this row
  o Re-selection:
      ▪ Calculate Fitness:
          • For each file on the files' pool (not eliminated)
              o The file content is matched with the
                signatures in T; if they are matched then
                  ✓ $\delta$ is added to the fitness of the
                    matched signature
                  ✓ The matched file is eliminated from
                    the files' pool
        ▪ The signatures are selected according to their
          fitness stochastically
```

Fig. 2. The VDC algorithm pseudo code.