Security Establishment In Cybersecurity Environment Using PSO Based Optimization

J Priyanka (priyankajaya86@gmail.com)
Madurai Kamaraj University
M Ramakrishnan
Madurai Kamaraj University

Research Article

Keywords: Cyber-Security, Meta-heuristic, Particle Swarm Optimization, Parento-approximation, anomaly classification

Posted Date: June 28th, 2021

DOI: https://doi.org/10.21203/rs.3.rs-530371/v1

Security Establishment In Cybersecurity Environment Using PSO Based Optimization

J.Priyanka

Ph.D. Research Scholar - Department of Computer Applications, School of Information Technology, Madurai Kamaraj University, Madurai-625021, Tamil Nadu, India
EMail: priyankajaya86@gmail.com

Dr.M. Ramakrishnan

Professor & Head, Department of Computer Applications, School of Information Technology, Madurai Kamaraj University, Madurai-625021, Tamil Nadu, India Email: ramkrishod@gmail.com

Abstract: ***Cybersecurity based significant data context is considered a challenge in the research community. Machine Learning approaches are considered for dealing with the big data-based security problem. Here, Particle Swarm Optimization (PSO) is used for configuring a massive amount of data. This work formulates a solution for Multi-objective problems to fulfill accuracy, computational and model complexities. A novel Meta-heuristic framework for multi-objective optimization is developed for dealing with lower levels and higher-level heuristics. In the former group, various rules are generated for configuring PSO, and in the latter model, search performance to control the selection process is used for newer configurations of PSO, deal with this multi-objective function. Parento-Approximation (PA) approach is used for strengthening this framework. The proposed optimization approach can be used in cybersecurity problems like anomaly classification. The proposed model is expected to provide better results in contrast to other models.***

Keywords- Cyber-Security; Meta-heuristic; Particle Swarm Optimization; Parento-approximation; anomaly classification

1.Introduction

With the growing advancements and the integration of social life and the Internet, there are substantial human activities. As the Internet is changing people by providing how to learn and work, also it leads to crucial security threats drastically [1]. Therefore, predicting various kinds of network attacks and the non-seen attacks is considered a key issue to be resolved directly [2].

Cyber-security is a process and set of technologies modeled to preserve networks, computers, data, and programs from attacks, destruction, alteration, and unauthorized access. The security system comprises of computer security and network security system [3]. These systems mentioned above include anti-virus software, firewall, and intrusion detection system (IDS) [4]. It helps to determine, discover, and predict the unauthorized system characteristics like copying, using, modifying, and destruction.

Cyber-security is an extensive research field, and the prediction of malicious network activities is one of the standards and older issues [5]. Moreover, intrusion detection is exceptionally reactive and responsive to certain observed anomalies and patterns [6]. The successive intuitive step is the consideration of a pro-active approach, where there is a necessity to infer preemptively with the upcoming malicious activities; thus, it can reach various events before causing any harm. Research progression and efforts in forecasting and prediction in cyber-security are not so famous with attack detection [7]. Moreover, it acquires huge attention, and advancements in this field can benefit the entire cyber-security discipline.

Before commencing the prediction process of cybersecurity, there is a necessity to determine what is generally to be predicted and the obstacles that lead to challenging problems. Initially, it is probable to identify the successive steps [8]. This process is known as attack prediction. A task like this process is known as intention recognition. The investigators evaluate the adversaries' foremost objective, which can also help recognize adversaries in successive moves. The next task is to predict cyber-attacks that are initiated to occur in a particular region.

In some cases, intrusion prediction is also considered, even though there are enormous techniques to predict vulnerabilities [9]. Finally, it is exciting to perform overall statistics of attacks, threat presence, and other essential information to form a network security situation. This work concentrates on providing network security using PSO by fulfilling multi-objective problems. This work focuses on network intrusion prediction. Also, various techniques and systems are anticipated to deal with these issues as analyzed in the literature, and it shares some common theoretical background also deal with this issue. Various factors influence the cause of threat over the network. The problem formulation is discussed based on this work.

2.Problem formulation

Recapitulate the open issues. This work emphasizes the following research challenges of forecasting and predictions in cybersecurity:

1) What is predicted from the cyber-security environment? Is it related to adversary appearance over incoming attack or cybersecurity from a global perspective?

2) How the prediction of cyber-security contributes the society? Whether it can be used for attack mitigation effectually? Else, it should be prepared for handling the upcoming security threats?

3) How the prediction evaluation of cyber-security and the metrics are utilized? Is that more appropriate in the assessment with testbeds and datasets or the appropriate prediction accuracy is measured with the available network settings? To deal with this, whether both practical and theoretical perspectives influence these research challenges? Here, postulates are provided to forecast and predict the possibilities and concentrate on evaluating and applying theoretical outcomes.

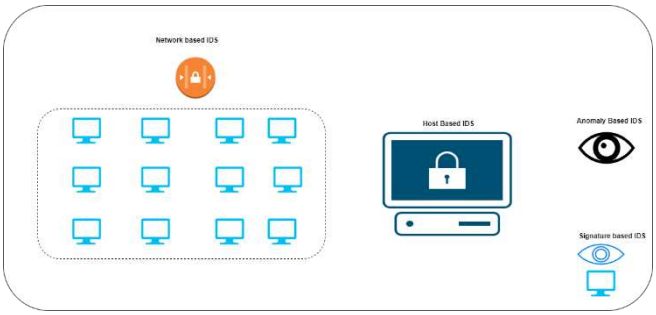

Fig 1: Generic view of Intrusion detection System (IDS)

resolve these issues as mentioned above, Machine Learning (ML) approaches are anticipated for categorizing the malicious software and unknown attacks. The utilization of ML approaches provides promising solutions to identify and categorize new malicious software. Support Vector Machines (SVMs) are amongst the standard ML algorithms and provides a great evaluation of the real-world environment. SVM's popularity is based on scalability and performance. Moreover, indeed of enormous benefits, the SVM performance is strongly influenced by the chosen configurations.

The accessible optimization techniques comprise gradient-based techniques, grid-search techniques, and meta-heuristic optimization approaches. The grid searching methods are significantly easier to execute and give better results [9]. Moreover, they are computationally higher with constraints in significant data applicability issues. Similarly, gradient-based techniques are extremely effectual; but there are certain substantial short-comings and need an objective function to be differentiable and ultimately depend on the primary function [10]. Various Meta-heuristic optimization approaches are also recommended to eliminate the drawbacks over the grid searching and gradient-based techniques. Moreover, the meta-heuristic approaches' functionality is strongly based on the chosen operators and parameters, where the selection is considered extraordinarily efficient and time-consuming.

This work concentrates on a novel Particle Swarm Optimization approach with Parento-approximation for configuring effectual optimization. This optimization process is extremely productive as they do not rely on any specific tasks and can acquire superior competitive configuration. The anticipated model merges various vital components that can differentiate it from prevailing works to find a productive cyber-security configuration. The expected model's performance is compared and validated with the prevalent approaches over cyber-security issues like malware over big-data classification and suitable for anomaly intrusion detection. The experimental outcomes were completely based on the efficiency of the anticipated model on both the problems.

This work is partitioned into five different sections. Section 2 is the problem formulation to carry out further research. Section 3 explains the background studies related to the optimization approaches. Section 4 is PSO with the Parento-optimization process for predicting intrusion in the cyber-security environment. Section 5 depicts the numerical outcomes and the appropriate discussion with prevailing methods like AASR, SRPGM, and OPTIMAL-EERP-SIGNCRYPTION. Section 6 is the conclusion and future directions.

3. Related works

This section discusses certain related works on malware detection techniques and optimization techniques. Also, it includes optimization to perform a classification crisis.

A recent study by Yu et al., [11] classifies malware detection techniques into three kinds: pattern-based detection, signature-based detection, and cloud-based detection techniques. The most prevailing detection approaches utilize a signature to identify malware software. It is a unique short string byte for known malware software; therefore, it is utilized to identify potential unknown software [12]. But, signature-based detection techniques can identify malware software; they need consistent revision over newer malware software signature into the signature database. It is simply evaded by malware developers using polymorphism, encryption. Moreover, signature databases are generally produced through the manual processing by diverse domain experts, and it is considered a time-consuming task [12].

Patterns-based detection approaches validate whether the provided malware software comprises a set of patterns. These patterns are extracted using domain experts to differentiate non-benign files and malware software [13]. Moreover, the malware software analysis and pattern extraction are performed by domain experts who are error-prone and need a huge amount of time. This specifies that manual analysis and extraction are crucial issues in constructing pattern-based detection techniques as malware software grows extremely faster.

Cloud-based detection approaches utilize the server for preserving detection software; therefore, malware detection is performed with the client-server process with

a cloud-based framework [14]. Moreover, these detection techniques are extremely influenced by the available number of clustered nodes and the detection processes running time. It can slow down the detection process, and therefore multi-able malware software cannot be recognized easily.

In general, due to the economic advantages, malware software acquires extreme complexity, and the malware developers use automated malware development toolkits for writing malware codes to avoid detection techniques [15]. Also, existing approaches are not scalable to handle these big data and less responsive to some threats due to the quick changes in malware software nature. ML algorithms are recommended to be utilized as malware detection methods to identify the malware software automatically [16]. Moreover, modeling an effectual detection process uses an ML process that is a confronting task due to the massive number of possible design options and lacks an appropriate intelligent manner to select and merge prevailing possibilities [17]. This work helps to resolve the various challenges by anticipating PSO based Parento-optimization process to search space and to design the options with diverse values. It iteratively integrates and uses different options for diverse problem instances.

A conventional SVM possess diverse tunable factors that have to be required to be optimized to acquire higher-quality outcomes. Meta-heuristic techniques extensively utilized determine the finest parameter finest combination of parameters and SVM values. It is an approach that targets understanding the problem features and the finest algorithm that fits it. Specifically, it attempts to learn and identify the problem elements merge to algorithm performance and evaluate suitable algorithms for this crisis. In [18], Vladimir anticipated meta-learning techniques to determine parameter values of the Gaussian kernel for SVM to resolve the regression problem. The author utilized k-NN as ranking techniques to choose the finest value for the kernel width parameter.

Lin et al., in [19], anticipated a meta-learning and case-based reasoning to produce initial starting solutions. The expected genetic algorithm is utilized to determine suitable parameter values to address the problem instance. Huang et al., in [20], uses a heuristic model to suggest kernel techniques for SVM. In [21], he et al. anticipated a hybrid technique that merges the meta-learning and searching process to choose parameter values.

Even though meta-learning techniques have been provided to be efficient in tuning SVMs parameter values, they still face the enormous over-fitting problem. This is due to the extracted problem features that capture instances that are utilized during the training process. As well, most existing techniques are utilized to tune single kernel techniques and were validated on small scale instances. The anticipated model uses kernel-based techniques and selection procedures, which are modeled with bi-objective optimization to handle the big data issues effectually.

Various heuristic algorithms are considered an emergent searching technique that determines the automated combination processor produces an effectual problem solver. The conventional hyper-heuristic models are completely modeled with an option as input and then decide which processes have to be used. The outcomes of a diverse hyper-heuristic structure are a problem solver indeed of its solution. Bao et al., in [22] anticipated the hyper-heuristic model to produce a set of attributes that characterizes the given instances for one-dimensional bin-packing issues. The author's utilized the hyper-heuristic model to identify which heuristic should be utilized to address the present problem instance.

Iqbal et al., in [23], anticipated learning vector quantization NN-based hyper-heuristics structure for addressing constraint satisfaction crisis. The hyper-heuristic structure was trained to determine which heuristic needs to be chosen based on given instances of hand. Ahmed et al., in [24], offered a stochastic hyper-heuristic approach for unsupervised matching towards partial information. This framework is implemented as a feature selection approach to demonstrate the feature sub-set that has to be chosen. Zhao et al., in [25], anticipate the hyper-heuristic model to deal with decision-tree for predicting software efforts.

IV. Methodology

This section discusses a novel Meta-heuristic framework for multi-objective optimization problems with lower-level and higher-level heuristics. In the former level, various rules are generated for configuring PSO, and in the latter model, search performance to control the selection process is used for newer configurations of PSO. Deal with this multi-objective function. An approximation approach is used for strengthening the proposed framework. The flow diagram of the anticipated model is given in Fig 2.

a. Standard Particle Swarm Optimization (PSO)

Generally, PSO is depicted as a parallel evolutionary computational approach modeled by Eberhart and Kennedy [26]. This is developed based on the social behavior of particles. It is greatly influenced by the tuning parameters known as exploitation and exploration. The former depicts the ability to concentrate on searching the candidate solution's vicinity for locating the optimal solution quickly and faster. The latter defines the ability to evaluate diverse regions of problems to optimum, which is preferably a global solution. However, the selections of parameters are considered to be empirical to more extent. The objective function is utilized to compute solutions and operate based on fitness values. Every particle has to save its position, composed of a candidate solution, and to calculate velocity and fitness. It is used in various applications to address many problems. The velocity and position based modifications is a process for attaining optimal solution at all iterations based on Equation (1) & (2):

$v_i\ (k+1) = wv_ik + c_1r_1(x_{best}, local - x_i) + c_2r_2\ (x_{best,\ global-}\ x_i)$	(1)
$x_i(k+1) = x_k + v(k+1)$	(2)

The position and velocity of particles are specified as vectors $x_i = (x_{i1}, \dots, x_{id})$ and $v_k = (v_{k1}, \dots, v_{kd})$respectively. Here, $'x'$ vectors are specified as the best global and local positions. c_1 And c_2 are accelerating factors termed as cognitive and social parametersr_1? Andr_2 are a random number that ranges from 0 and 1. $'k'$ is iteration index. $'w'$ is inertia weight parameters and update $x_i$for particle.

b. Improved Particle Swarm Optimization

Here, an optimization approach known as Improved Particle Swarm Optimization (IPSO) is used to address the problem identified in Cyber-Physical system based intrusion detection []. This method helps to recognize the malicious user (attack scenario) is encountered in cyber-systems. This is done to improve the performance of the system. It comprises of two phases known as the ranking phase (RP) and Grouping phase (GP). The ranking phase is used for a classifier from well-known (labeled) network traffic data. The Grouping phase is applied for classifying incoming patterns (unlabelled or labeled) using Particle Swarm Optimization. The grouping phase makes the classifier more dynamic.

1) Ranking phase

This phase is used for determining the attributes of labeled patterns (fields). This is because there are several available features. Here, classification is considered to be a time-consuming process. Time-consuming is due to a larger number of patterns. The ranking phase has to choose only the attributes subset with sampling approaches or Principle Component Analysis to mitigate computational load. The ranking process is explained in Algorithm 1. The initial process is done to perform classification based on a set of available labeled patterns. All the designs are labeled in prior. It is known that the accuracy is improved by ranking and with the use of algorithms like IPSO.

Algorithm 1:

1. Labeled data pre-processing
2. Perform ranking based on pre-processing data

//Grouping phase

1. Unlabeled data pre-processing
2. Create groups after successive ranking of pre-processed data using IPSO

Show the classifier output

2) Grouping phase

Here, IPSO is applied for the grouping process. It is utilized to identify the unknown and known network traffic patterns more dynamically. Based on the prior evaluation, the standard PSO uses clustering to emulate the social characteristics. Every particle's velocity and position give a grouping solution (centroids) and searching mechanism, respectively. The speed and work at all iterations $t+1$ is depicted as in Eq. (3) & (4):

$$P_i^{t+1} = p_i^t + v_i^{t+1} \tag{3}$$

$$v_i^{t+1} = \omega v_i^t + a_1\varphi_1(pb_i^t - p_i^t) + a_2\varphi(gb^t - p_i^t) \tag{4}$$

Here, pb_i^tit is the best particle position; gb^tit is in the global best p ω art; it is inertial weight; $\varphi_1\varphi_2$ it is distributed uniformly to determine $gbpb_ia_1a_2$ and two constant values. The PSO based grouping process is adjusted dynamically by IDS classifiers. The concept behind this grouping is simpler. It encodes k −centroids as particle position $p_i = (c_1^i, c_2^i, \dots, c_k^i)c_j^ij^{th}$ s. Where is centroid encoded in i^{th} particle The particle fitness is depicted as in Eq. (5):

$$fitness = \sum_{j=1}^{k_i} \sum_{\forall x \in \pi_j^i} \left|\left|x - c_j^i\right|\right|^2 \tag{5}$$

Herek_i the number of clusters ini^{th} particle; π_j^i is j^{th}a cluster ofi^{th} particle. The process is to initialize particles with the use of 'perturbed' outcomes. It can increase the population diversity. Perform the initial process, and the assignment process is used to determine the types of attacks from input patterns dependent on threshold values. The threshold is $\overline{d}_\iota$ the average distance among the cluster centroid and patterns that comes under the cluster $'i'$ attained from the grouping phase. The successive threshold values are dynamically adjusted. With these threshold values, the anticipated algorithm is competent to verify whether the patterns can be allocated to every cluster or newer class has to be generated.

Therefore, it automatically determines the number of attacks. Eliminate the anticipated approach from creating more number of clusters from the smaller number of

patterns; this process is applied. Next, the IPSO grouping uses the outcome of the initially assigned procedure to generate particles first and create successive particles like p_b particles that comprise k_{max} the maximal number of clusters from the assignment process. Here, $k_{min} = k_{max}$; the value k_{max} is set as $k_{min} + 2$. The standard PSO is used to search optimization solutions. Provide better PSO solutions. The k-centroid is utilized as a local search operator. Here, two processes comprise the core algorithm process, which shows solutions and thus attains better IDS performance to predict whether the incoming traffic is normal or abnormal.

Algorithm 2:

1. Define the attributes of labeled incoming patterns
2. For all labeled patterns
3. Allocate group that comes under a common label
4. For all group
5. Count set of patterns
6: Evaluate the centroid
7: Return the evaluated value as initial classifier

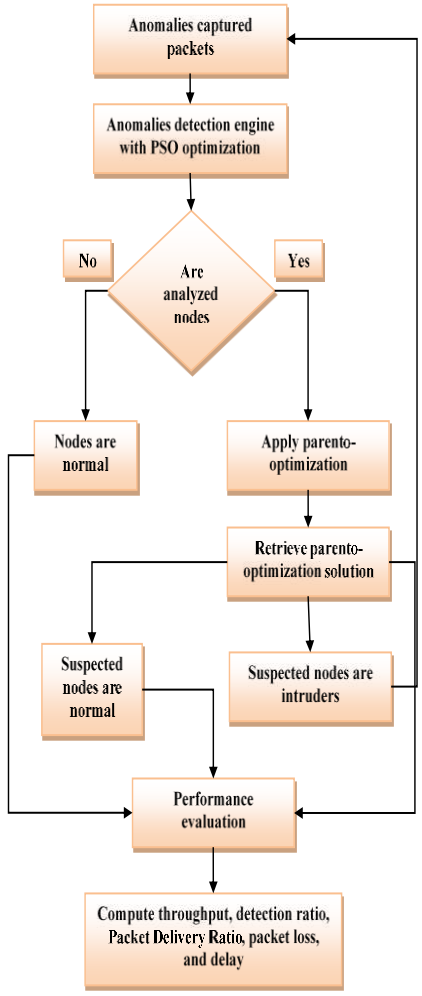

Fig 2: Flow diagram of secure PSO

To validate and analyze the new incoming traffic that enters the cyber-system. Also, prior works cannot re-train some classifiers by performing grouping algorithms for over-all data when a new pattern enters the IDS. It will take a huge amount of computational time. Thereby, it slows down the system performance. To resolve this crisis, the anticipated model initially verifies the incoming patterns of the existing group (classifier) or checks whether it is a new type of traffic. It compares the distance of incoming ways with the threshold T_d. When space is comparatively smaller than the threshold($d_x < T_d$) where the incoming patterns belong to the prevailing classifier. Else, it is considered as a new type of traffic. Therefore, the new classifier is added to IDS. When a certain amount of incoming patterns (25%) of labeled patterns are belonging to newer classifiers, it is added to the IDS. The IPSO is used to perform pattern grouping to adjust IDS classifier. Therefore, it resolves the problem identified during clustering and reduces the computation complexity identified during IDS performance.

Algorithm 3: Allocating clusters to IDS

1. Particle initialization with centroids of known value (grouping phase)
2. For all particles
3. Consider $T_d = 0$;
4. For all unlabeled patterns
5. Compute distance among the centroids
6. Find centroid nearer to $'x'$ values
7. When $d_x < \overline{d_\iota}$ the
8. Allocate patterns to group 1
9. Else
10. Add a new group and allocate $'x'$ values
11. Consider T_d value
12. End for
13. End for

//validation

14. Create particles from the grouping phase and generate successive particles from $k_{min}\ to\ k_{max}$
15. Perform computation with updating and changing operators to modify positions and velocities.
16. Perform centroid based computation
17. When stopping criteria is fulfilled, then stop the process
18. Obtain the best particle
19. Else move to step 14.

c. Multi-objective optimization

As a multi-objective problem to be resolved in cyber-system by analyzing the huge incoming data as below in Eq. (6):

$$\min_x f(x) = (f_1(x), \dots, f_m(x))\hat{}T \qquad (6)$$

Subject to $x \in X = \{x \in R^n | g_j(x) \leq 0; j = 1,..,l$

Here, $x = (x_1, \dots, x_n)^T$ the design variable vector and $'X'$ the set of feasible solutions. In general, unlike conventional optimization approaches with a single objective function, there prevails an optimal solution that reduces all objective functions $f_i(x), i = 1,..,m$ simultaneously. Therefore, the idea behind an optimal solution is related to Parento dominance.

d. Parento optimal solution

In MOP, $\hat{x} \in X$ is considered as a Parento Optimal solution; when there is no $x \in X$ such that;

$$f_i(x) \leq f_i\widehat{(x)}\ for\ all\ i = 1,..,m \qquad (7)$$

$$f_j(x) \neq f_j(\hat{x}) forall\ j = 1, \dots, m \qquad (8)$$

The optimal parent solution is set to attain an objective solution known as Parento frontier. The solution for the multi-objective problem may be attained from a set of solutions in existing approaches. Using multi-objective problems, two or three objectives are fulfilled using Parento frontiers and assists in decision making and to make preferable solutions. It is easier to make better analysis by visualizing the problem. For the construction of Parento Optimal solutions, various meta-heuristic approaches like PSO and GA have been extended. More specifically, meta-heuristic processes are observed from all cases to attain two or more functionalities. There are some essential factors to be analyzed:

1) How to assist population-based on Parento boundaries faster and closer (convergence rate).

2) How to determine well-established solutions by spreading complete Parento boundaries.

3) How to select the best solution

4) How to determine the parameters like velocity and position

The Parento optimal solution is to determine self-adaptive parameters and to enhance the convergence by attaining better solutions. The general multi-objective optimization problem is of minimization type, which is specified as in Eq. (9) – Eq. (11):

$$\min F(x) = [f_1(x), f_2(x), .., f_m(x)] \qquad (9)$$

$$\zeta(x) = [\zeta_1(x), \zeta_2(x), \dots, \zeta_c(x)] \geq 0 \qquad (10)$$

$$x_i^{(L)} \leq x_i \leq x_i^{(U)} \qquad (11)$$

Here, $X = (x_1, x_2, .. x_N)$ the set of decision variables, $'m'$the number of objectives, $'c'$the number of constraints, $\zeta(x)$the lower bound of decision variable, $x_i^{(U)}$and the decision variable's upper bound. In a multi-objective optimization problem, two diverse solutions are compared based on dominance. Consider two solution$'x'$s and$'y'$ which is said to dominate $'y'$. When$'x'$ is superior or equal to $'y'$in all common objectives, and it is strictly superior to all other variables $'b'$with least one goal.

$$F(x) \prec f(y) iff \begin{cases} f_i(x) \leq f_i(y) & for\ all\ i = 1,..,m \\ \exists i \in 1,..,m & f_i(x) < f_i(y) \end{cases} \qquad (12)$$

Here $'x'$ is a parento optimal solution when there is no dominant solution. Based on the ser of parento optimal solutions, it is known as Pareto-optimal set, and the objective space is known as Pareto front or parento frontier. The ultimate goal is to attain an optimization algorithm for finding a solution.

In PSO's case, the accuracy is measured with the complexity of the position and velocity of particles. A larger number of particles leads to an over-fitting problem when the value of 'C' is increased due to the generalized ability that leads to inappropriate classification for every sample. This is managed by controlling the selection of particles (kernel parameters and kernel type). Here, accuracy and complexity are considered as major training instances.

The accuracy specifies the classification performance of the instance. The PSO is trained K-times. In all iterations, K-1 sets are considered for training, and others are used for testing. The error specifies the average number of misclassification set with training iterations.

Similarly, complexity specifies particles or upper bound with the expected number of errors. The configuration comprises of decision variables (velocity and positions). The bounding of all decision variables lies in the range of possible items. The optimization is done for two objectives thus$(m = 2)$. It is formulated as in Eq. (13):

$$\min F(x) = [f_1(x), f_2(x)] \qquad (13)$$

$$where\ f_1(x) = \ error$$

$$f_2(x) = \ number\ of\ variables$$

Here, the error is due to the number of misclassified datasets, and the number of variables is specified as the variables related to PSO configuration. The instances are partitioned into K disjoint sets of the same size, and the heuristic framework is applied for lower-level and higher-level strategies. The former selects heuristics from existing lower-level heuristics where the higher level works on heuristic space indeed of solution space.

Resolve the multi-objective problem. This work uses PSO with Pareto-optimal solutions. It works based on the population of solutions and utilizes archive to store the non-dominate solutions. The anticipated model merges Pareto dominance and decomposition to approximate the configuration set effectually. The concept is to connect the diversity ability with a decomposition approach with immense convergence power. It operates on a population solution, and the framework generates a newer solution using an older population. It facilitates appropriate balance among diversity and convergence. When the finest intersection is used to reduce the distance among solutions, it provides higher diversity involvement towards distribution maximization with the appropriate solution.

V. Numerical results and discussion

Here, simulations are performed using Network Simulator-NS2, and the results are mentioned in the below table. Mainly the performance and effectiveness of the proposed Secure-PSO algorithm are analyzed in the simulator. Two protocols, namely AASR, SRPGM, and OPTIMAL-EERP-SIGNCRYPTION, are also studied in this experiment and represented through graphs. The below parameters are being analyzed in the conducted simulation study: Packet Delivery Ratio (PDR), Average Delay, Average Energy Consumption, Packet loss, and Detection ratio, respectively.

Table 1: simulation parameters

PARAMETER	**VALUE**
Simulator	NS-2.34
Topology	Random node placement
Number of nodes	50
Bandwidth	2 Mbps
Propagation Model	Two Ray Ground
Physical Model	Wireless
Antenna model	Omni Antenna
Queue Size	50
Traffic type	CBR,UDP
Attacker nodes	5,10,15,20,25 and 30
Routing Algorithm	Secure-PSO
Packet size	512
Mac protocol	802.11 standard
Simulation Time	200Sec

From Table I, it is known that random node placement topology is considered for node placement with 50 nodes. Here, a wireless physical model is used with the Omni antenna. 802.11 MAC protocol standard is considered with a secure-PSO routing algorithm. Fig 3 depicts the node placement over the simulated network environment, and the establishment of node connectivity is shown in Fig 4 and Fig 5. The source and destination nodes are shown in highlighted colors.

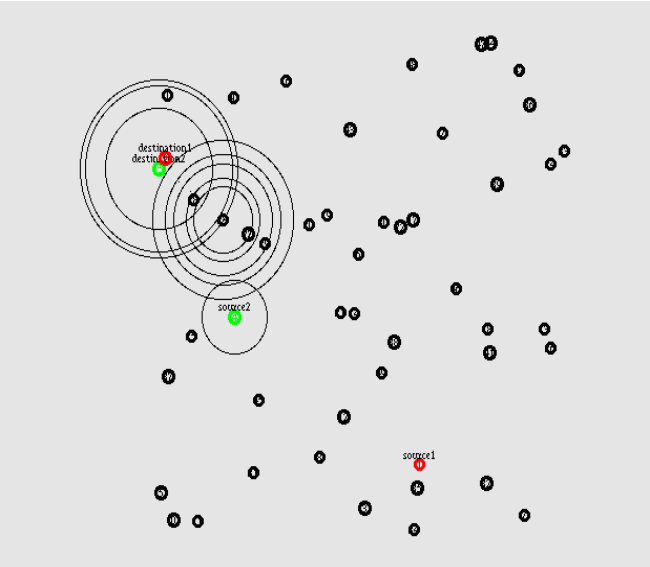

Fig 3: Node creation

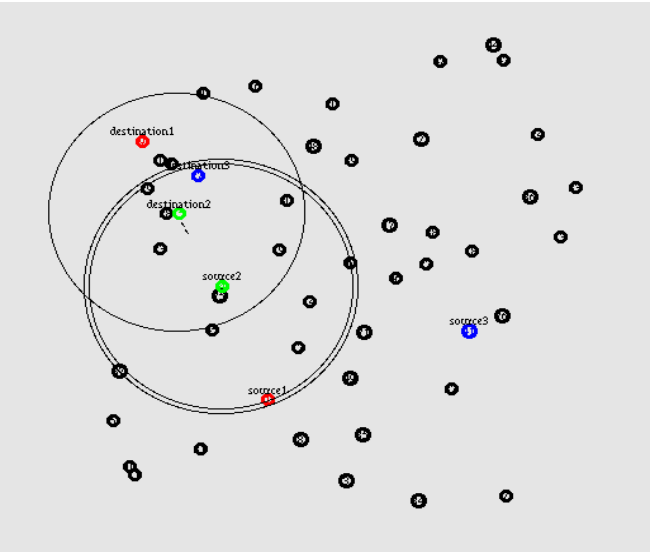

Fig 4: Analyzing malicious node activities

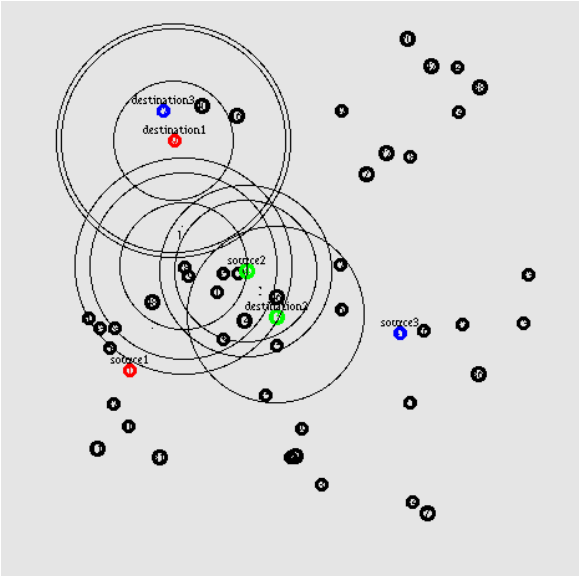

Fig 5: Route establishment from source to destination

Table II: Delay comparison

No of nodes	AASR	Optimal EERP	SRPGM	Secure PSO
5	6.90	3.90	2.90	1.90
10	11.22	5.22	3.22	2.22
15	14.6	7.6	5.6	4.6
20	18.88	8.88	7.88	5.88
25	22.90	10.90	9.90	7.90
30	26.90	12.90	11.90	9.90

Table II depicts the delay comparison of anticipated Secure PSO with AASR, optimal EERP, and SRPGM, respectively. Here, 5 to 30 nodes are considered where the anticipated model shows reduced delay than other approaches, while AASR shows higher delay than optimal EERP, SRPGM, and secure PSO. Table III depicts the attack detection ratio comparing anticipated Secure PSO with AASR, optimal EERP, and SRPGM, respectively. The proposed secure PSO gives a higher detection rate towards the attackers than SRPGM, optimal EERP, and AASR methods. It gives 99% detection towards the malicious nodes, which is 6% higher than SRPGM, 2% higher than optimal EERP, and 5% higher than AASR.

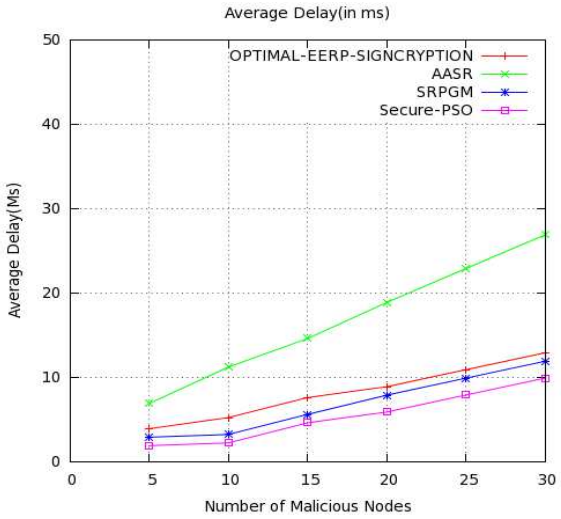

Fig 6: Delay comparison

Table III: Detection ratio comparison

No of nodes	AASR	Optimal EERP	SRPGM	Secure PSO
5	0.94	0.97	0.93	0.99
10	0.94	0.92	0.90	0.98
15	0.92	0.90	0.85	0.96
20	0.90	0.88	0.82	0.93
25	0.88	0.86	0.79	0.92
30	0.86	0.85	0.76	0.90

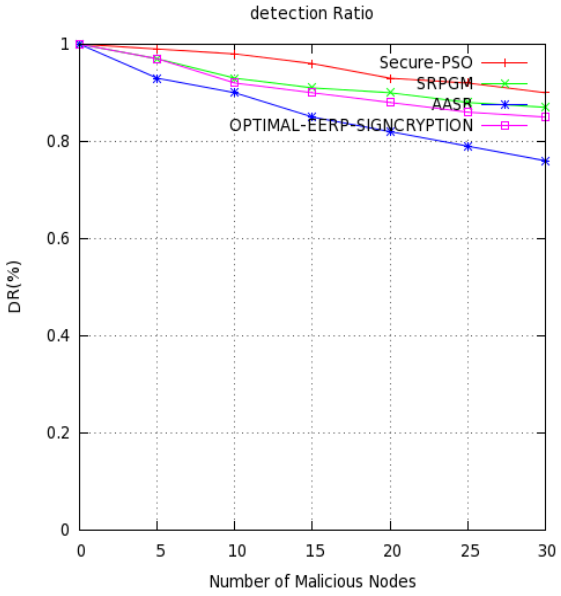

Fig 7: Detection ratio comparison

Table IV: Energy consumption (J) comparison

No of nodes	AASR	Optimal EERP	SRPGM	Secure PSO
5	6.90	5.90	3.90	2.90
10	10.22	7.22	4.22	3.22
15	14.6	10.6	6.6	4.6
20	17.43	12.43	7.43	5.43
25	21.55	13.55	8.55	6.55
30	26.43	15.43	9.43	7.43

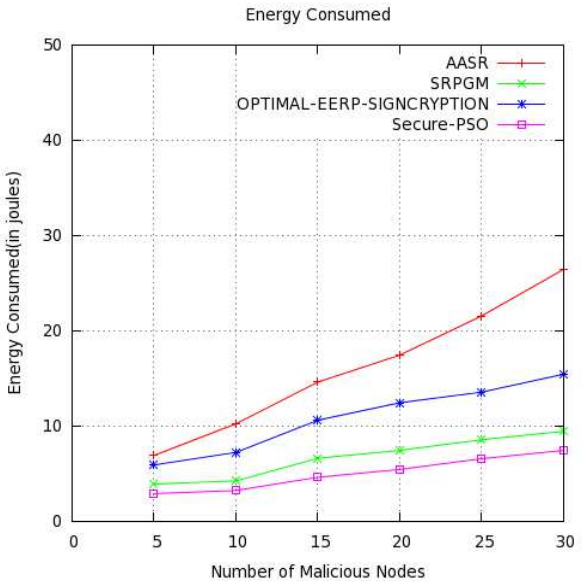

Fig 8: Energy consumption (J)

Table V: Packet Loss comparison

No of nodes	AASR	Optimal EERP	SRPGM	Secure PSO
5	6.90	3.11	2.11	1.11
10	9.22	5.12	4.12	3.12
15	14.6	6.54	5.54	4.54
20	23.43	8.13	7.13	5.13
25	35.55	8.55	7.55	6.55
30	47.43	9.43	8.43	7.43

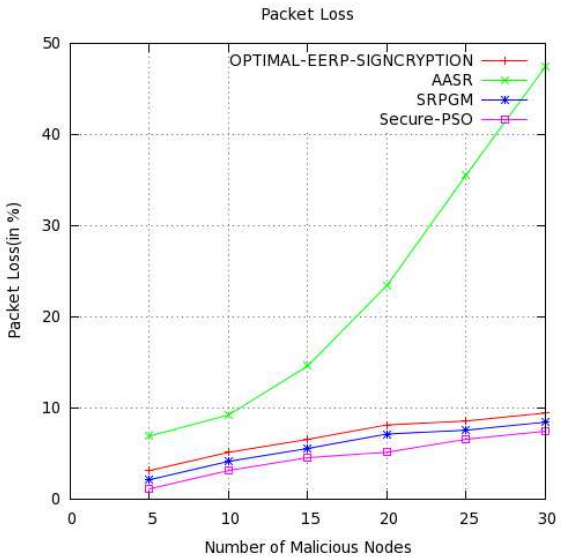

Fig 9: Packet loss comparison

Table VI: PDR comparison

No of nodes	AASR	Optimal EERP	SRPGM	Secure PSO
5	97.28	98.28	99.28	99.96
10	90.58	95.58	96.58	98.32
15	85.14	93.14	94.14	96.17
20	77.38	91.38	90.38	93.38
25	64.89	89.89	89.89	90.89
30	50.13	86.13	87.13	89.13

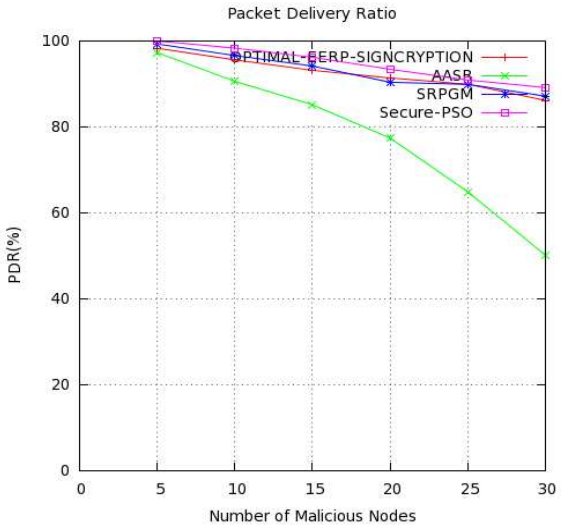

Fig 10: PDR comparison

Table VII: Throughput comparison

No of nodes	AASR	Optimal EERP	SRPGM	Secure PSO
5	102.11	108.90	109.90	112.90
10	95.23	106.22	107.22	110.22
15	90.6	102.6	104.6	108.6
20	85.43	100.43	102.43	105.43
25	80.55	98.55	100.55	103.55
30	75.43	95.43	98.43	101.43

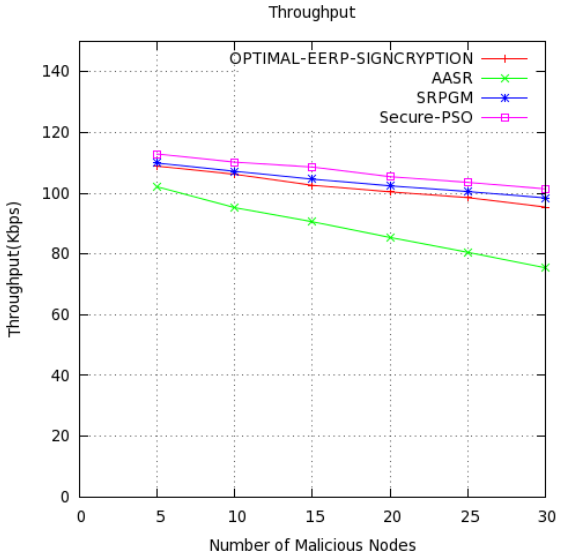

Fig 11: Throughput comparison

Table IV depicts the energy consumed by malicious nodes and the comparison of anticipated Secure PSO with AASR, optimal EERP, and SRPGM. The proposed secure PSO consumes less power of 2.90J with five nodes than SRPGM, optimal EERP, and AASR methods. It consumes 2.90J energy towards node placement, which is 1J lesser than SRPGM, 3J lesser than optimal EERP, and 4J lesser than AASR. Table V depicts the packet loss comparison of anticipated Secure PSO with AASR, optimal EERP, and SRPGM. The proposed secure PSO lesser packet loss during identifying attackers compared to SRPGM, optimal EERP, and AASR methods. It gives 1.11 bytes loss towards the malicious nodes, which is 1 byte lesser than SRPGM, 2 bytes lesser than optimal EERP, and 5 bytes lesser than AASR, respectively. Table VI depicts the packet delivery ratio comparison of anticipated Secure PSO with AASR, optimal EERP, and SRPGM. The

proposed secure PSO shows a higher delivery ratio during identifying attackers than SRPGM, optimal EERP, and AASR methods. It gives 99.96% delivery towards the malicious nodes, which is 0.38% higher than SRPGM, 1.38% higher than optimal EERP, and 2.38% higher than AASR, respectively. Table VII depicts the throughput comparison of anticipated Secure PSO with AASR, optimal EERP, and SRPGM. The proposed secure PSO shows higher throughput during identifying attackers than SRPGM, optimal EERP, and AASR methods. It gives 112.90 bits/sec towards the malicious nodes, which is 3 bits higher than SRPGM, 4 bits higher than optimal EERP, and 10.79 bits higher than AASR. Fig 6 to Fig 11 depicts the performance metrics comparison of Secure PSO with other models.

From the above analysis, it is known that the anticipated Secure PSO with Parento optimization gives better results with metrics like throughput, loss, PDR, delay, and detection ratio. This proves that this model works effectually in predicting the malicious nodes over the cyber-security network than the other models like SRPGM, EERP, and AASR, respectively.

VI . Conclusion

The proposed model concentrates on handling the cyber-security framework issues that occur due to the malicious nodes. The challenges are addressed using Machine Learning techniques to enhance security. Here, Improved Particle Swarm Optimization (IPSO) is applied for configuring the number of incoming data packets to the network. The proposed Meta-Heuristic model is used for improving the performance measures compared to other existing approaches. The proposed meta-heuristic framework for addressing multi-objective optimization is developed for dealing with lower-level and higher-level heuristics. In lower-level heuristics, various rules are generated for configuring PSO, while in higher-level heuristics, the control selection process is applied for PSO configurations. Also, the multi-objective problems are handled by the Parento-approximation approach for strengthening the proposed framework. Also, metrics like throughput, PDR, packet loss, detection ratio, and energy consumption are measured. With this observation, it is known that the anticipated model works well compared to other approaches. In the future, hybrid optimization approaches are used and tested with security measures.

DECLARATIONS

-Ethical Approval: not applicable

. **Funding** UGC-RGNF(Award number: 201516-RGNF-2015-17-SC-TAM-25217)

Conflicts of interest/Competing interests 'Not applicable'

Availability of data and material 'Not applicable'

Author's contribution

Priyanka Jayachandran: contributed with full support as the technical and development.

Dr.M. Ramakrishnan: contributed with full support as the guidance as well as development.

ACKNOWLEDGMENT:(**M. Ramakrishnan** is a co-author)

REFERENCES

[1] Y. Yan, Y. Qian, H. Sharif, and D. Tipper, "A survey on cybersecurity for smart grid communications," *IEEE Commun. Surveys & Tuts.*, vol. 14, no. 4, pp. 998–1010, 4th Quart. 2012.

[2] Y. Yan, Y. Qian, H. Sharif, and D. Tipper, "A survey on cybersecurity for smart grid communications," *IEEE Commun. Surveys Tuts.*, vol. 14, no. 4, pp. 998–1010, 4th Quart. 2012.

[3] W. Wang and Z. Lu, "Cybersecurity in the smart grid: Survey and challenges," *Comput. Netw.*, vol. 57, no. 5, pp. 1344–1371, 2013.

[4] S. Bi and Y. J. Zhang, "Using covert topological information for defense against malicious attacks on DC state estimation," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 7, pp. 1471–1485, Jul. 2014.

[5] Z. Ismail, J. Leneutre, D. Bateman, and L. Chen, "A game-theoretical analysis of data confidentiality attacks on smart-grid AMI," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 7, pp. 1486–1499, Jul. 2014.

[6] Y. Guo, C.-W. Ten, S. Hu, and W. W. Weaver, "Preventive maintenance for advanced metering infrastructure against malware propagation," *IEEE Trans. Smart Grid*, vol. 7, no. 3, pp. 1314–1328, May 2016.

[7] Y. Mo, S. Weerakkody, and B. Sinopoli, "Physical authentication of control systems: Designing watermarked control inputs to detect counterfeit sensor outputs," *IEEE Control Syst.*, vol. 35, no. 1, pp. 93–109, Feb. 2015.

[8] H. Lin, A. Slagell, Z. Kalbarczyk, P. W. Sauer, and R. K. Iyer, "Runtime semantic security analysis to detect and mitigate control-related attacks in power grids," *IEEE Trans. Smart Grid*, vol. 9, no. 1, pp. 163–178, Jan. 2016.

[9] Y. Fan, Z. Zhang, M. Trinkle, A. D. Dimitrovski, J. B. Song, and H. Li, "A cross-layer defense mechanism against GPS spoofing attacks on PMUs in smart grids," *IEEE Trans. Smart Grid*, vol. 6, no. 6, pp. 2659–2668, Nov. 2015

[10] M. Ozay, I. Esnaola, F. T. Y. Vural, S. R. Kulkarni, and H. V. Poor, "Machine learning methods for attack detection in the smart grid," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 8, pp. 1773–1786, Aug. 2016.

[11] Yanfang Ye, Tao Li, Donald Adjeroh, and S Sitharama Iyengar. A survey on malware detection using data mining techniques. *ACM Computing Surveys (CSUR)*, 50(3):41, 2017.

[12] Eric Filiol. Malware pattern scanning schemes secure against black-box analysis. *Journal in Computer Virology*, 2(1):35–50, 2006.

[13] Eric Filiol, Gr' egoire Jacob, and Micka¨el Le Liard. Evaluation methodology and theoretical model for antiviral behavioural detection strategies. *Journal in Computer Virology*, 3(1):23–37, 2007.

[14] Mohsen Damshenas, Ali Dehghantanha, and Ramlan Mahmoud. A survey on malware propagation, analysis, and detection. *International Journal of Cyber-Security and Digital Forensics (IJCSDF)*, 2(4):10–29, 2013.

[15] Min Chen, Shiwen Mao, and Yunhao Liu. Big data: A survey. *Mobile Networks and Applications*, 19(2):171–209, 2014.

[16] Luba Gloukhov, Cody Wild, and David Reilly. Malware classification: Distributed data mining with spark. In *Association for the Advancement of Artificial Intelligence*, pages 1–6. www.aaai.org, 2015

[17] Jos'e Carlos Ortiz-Bayliss, Hugo Terashima-Mar' ıN, and Santiago Enrique Conant-Pablos. Learning vector quantization for variable ordering in constraint satisfaction problems. *Pattern Recognition Letters*, 34(4):423–432, 2013.

[18] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 2013.

[19] J. Liu, Y. Xiao, S. Li, W. Liang, and C. L. P. Chen, "Cyber Security and Privacy Issues in Smart Grids," Commun. Surv. Tutor., vol. 14, no. 4, pp. 981–997, Fourth 2012

[20] Y. Huang, J. Tang, Y. Cheng, H. Li, K. A. Campbell, and Z. Han, "Real-Time Detection of False Data Injection in Smart Grid Networks: An Adaptive CUSUM Method and Analysis," IEEE Syst. J., vol. 10, no. 2, pp. 532–543, Jun. 2016.

[21] H. He. J. Yun, "Cyber-physical attacks and defences in the smart grid: a survey," IET Cyber-Phys. Syst., Theory Appl., vol. 1, no. 1, pp. 13–27, Nov.2016.

[22] H. Bao, R. Lu, B. Li, and R. Deng, "BLITHE: Behavior Rule-Based Insider Threat Detection for Smart Grid," IEEE Internet Things J., vol.3, no.2, pp.190–205, Apr. 2016.

[23] S. Iqbal et al., "On cloud security attacks: A taxonomy and intrusion detection and prevention as a service," J. Netw. Comput. Appl., vol.74, no.2, pp.98–120, Oct. 2016.

[24] S. Ahmed, Y. Lee, S.-H. Hyun, and I. Koo, "Feature selection–based detection of covert cyber deception assaults in smart grid communications networks using machine learning," *IEEE Access*, vol. 6, pp. 27518–27529, 2018.

[25] L. Zhao and B. Zeng, "Vulnerability analysis of power grids with line switching," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 2727–2736, Aug. 2013.

[26] James Kennedy' and Russell Eberhart, "Particle Swarm Optimization", IEEE, 1995.