IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 23 June 2023, accepted 14 August 2023, date of publication 22 August 2023, date of current version 30 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3307492

RESEARCH ARTICLE

An Integral Cybersecurity Approach Using a Many-Objective Optimization Strategy

OMAR SALINAS[1], **RICARDO SOTO**[2], **BRODERICK CRAWFORD**[2], **AND RODRIGO OLIVARES**[1]

[1]Escuela de Ingeniería Informática, Universidad de Valparaíso, Valparaíso 2362735, Chile
[2]Escuela de Ingeniería Informática, Pontificia Universidad Católica de Valparaíso, Valparaíso 2362807, Chile

Corresponding authors: Omar Salinas (omar.salinas@postgrado.uv.cl) and Ricardo Soto (ricardo.soto@pucv.cl)

The work of Omar Salinas was supported by Doctorado en Ingeniería Informática Aplicada, Universidad de Valparaíso, under Grant 101.016/2020. The work of Rodrigo Olivares was supported by Grant ANID/FONDECYT/INICIACIÓN/11231016. The work of Broderick Crawford was supported by Grant ANID/FONDECYT/REGULAR/1210810.

ABSTRACT Data networks and computing devices have experienced exponential growth. Within a short span of time, they have opened new digital frontiers while also bringing forth new threats. These threats have the potential to increase costs and disrupt regular operations. Choosing a cybersecurity plan to address these threats requires balancing direct and indirect costs against the benefits of implementation and subsequent operation. In this study, we propose an efficient strategy for designing networking topologies by incorporating a Security Information and Event Management System. This system consists of a central server and Network Intrusion Detection Sensors, which gather data and promptly transmit information regarding suspicious activities to the server. The server then takes immediate action in case of incidents. To determine the optimal number and placement of sensors, a many-objective optimization approach is employed. The problem is mathematically modeled using linear programming. To solve the optimization problem, swarm intelligence techniques such as the particle swarm optimizer, the bat algorithm, and the black hole method are utilized. Various test scenarios were created by presenting low, medium, and complex instances of conventional networks. The results obtained using the black hole bio-inspired algorithm were particularly satisfying, surpassing the performance and resolution of the other methods.

INDEX TERMS Security information and event management, network intrusion detection system, cybersecurity, many-objective optimization strategy, metaheuristics.

I. INTRODUCTION

Since the advent of personal computing and the subsequent emergence of data networks, the landscape of information technologies has undergone a profound transformation, fundamentally altering our ways of interaction [1]. Today, the irruption of the internet has become an essential part of human life [2]. We now rely on the internet for various purposes, accessing it from our homes, offices, and portable devices, regardless of our location. Within organizations and companies, information systems facilitate the exchange of data, supporting critical business operations and contributing to the overall mission of each entity [3]. Consequently, the reliance on information technologies has grown substantially, necessitating the implementation of robust security risk management measures to safeguard sensitive data and information. Failure to address these measures may lead to unintended consequences. Moreover, organizations and companies are increasingly cognizant of the potential impacts that a security breach in their computing infrastructure could have on their business continuity and overall reputation [4].

An organization's effective cyber risk management involves the identification of critical assets that are essential for operational and productive continuity, as well as the evaluation of measures to protect them against potential threats [5]. Risk, in this context, refers to the possibility of experiencing damage or loss. Threats, on the other hand, are components of risk and encompass threat agents, whether

The associate editor coordinating the review of this manuscript and approving it for publication was Frederico Guimarães.

human or non-human, capable of triggering actions such as identifying and exploiting vulnerabilities, resulting in unexpected and unwanted outcomes [6]. However, the current state of cyber risk management falls short of delivering the desired results and requires improvement. This issue stems from various factors, including the ever-changing landscape of new attacks, which go undetected by existing security technology systems due to novel patterns of incidence. It is worth noting that organizations worldwide are making unprecedented investments in cybersecurity, yet they still struggle to achieve their desired outcomes due to a misaligned focus on priorities [7]. Reports indicate a steady increase in successful malware attacks over the past decade, causing damages exceeding 7.5 billion. In 2017, a survey estimated that a typical financial institution faced an average of 85 percent of cyberattacks annually, with one-third of them resulting in successful breaches [8], [9]. In summary, organizations face continuous cyber-attacks, necessitating more effective mitigation actions for defense [10].

Cyber risk management endeavors to safeguard information assets through the utilization of cutting-edge techniques, disciplines, policies, firewalls, and established intrusion detection and prevention systems. However, selecting an appropriate cybersecurity plan from the available options can be a daunting undertaking. Numerous security controls are at our disposal, and strategically placing them within the network can offer some defense against the exploitation of vulnerabilities. Nonetheless, these approaches often find themselves in a state of constant conflict, where implementing one plan invariably diminishes the viability of alternative strategies.

Operating systems, software, and applications often have known vulnerabilities that cybercriminals can exploit, including firewalls, antivirus software, and other security mechanisms. Therefore, it is crucial to regularly apply security updates provided by vendors. Additionally, implementing a vulnerability management program is essential as it helps detect and resolve issues before cybercriminals can take advantage of them. Make sure that technologies such as firewalls and intrusion detection systems (IDS) are properly configured and regularly updated to protect against emerging threats. Consider leveraging cloud-based security solutions that offer ongoing, up-to-date protection against both known and emerging threats. These solutions often provide advanced detection and response capabilities, along with automatic security updates. However, it is important to remember that cybersecurity is an ongoing effort that requires constant updates and improvements to stay ahead of new and emerging threats and vulnerabilities.

In light of the aforementioned arguments, it becomes imperative to undertake research that employs a strategy to address cyber risk management as a many-objective optimization problem, aiming to discover effective and efficient solutions. The multicriteria approach has previously been studied to support cybersecurity issues, with many of them being framed as modeling problems in binary domains [11]. For example, in [12], three control methods are proposed for addressing anti-phishing measures, vulnerabilities, and attack paths. It is important to note that attack modeling techniques play a vital role in understanding, exploring, and validating security threats in the cyber world, as highlighted in the bibliographic review [13]. In [14], a suitable trade-off between cyber risk and investment is proposed using the mixed-integer paradigm. Recently, [15] published another work that employs a bi-objective formulation, where the primary optimization components considered are the service cost and multi-cloud risk. The reported works generally focus on direct costs, risks, vulnerabilities, and similar issues. In our proposal, we also consider indirect costs associated with networking performance and the benefits derived from its installation and subsequent operation.

Our research proposes the development of a cyber risk management strategy, which entails the implementation of a security control known as a security information and event management system. This system consists of a server and sensors distributed throughout the network. To achieve this, we mathematically model the network's requirements based on various functional factors. These two aspects of development converge to accomplish a shared objective: establishing a protected network with strategically positioned and technologically advanced resources.

SIEM solutions have evolved into comprehensive systems that offer extensive visibility, enabling the identification of high-risk areas and proactive mitigation strategies to minimize costs and incident response time [16]. Furthermore, in [17], the authors assert that SIEM enables the storage and monitoring of unwanted events and unauthorized access to computer system records, which can lead to various security threats, including information leakage and breaches of privacy and confidentiality.

As mentioned earlier, the SIEM system will consist of a central server and sensors responsible for collecting log information from multiple sources. These sensors will correlate events to identify malicious activities or cyber-attacks. Determining the optimal number of sensors and their installation locations will be approached as a multi-objective problem, mathematically modeled using linear programming based on the technical and functional aspects of the network. The solution will be obtained through the utilization of metaheuristic techniques and bio-inspired algorithms. The main contributions are related to: (a) increasing the coverage of the NIDS across the network with the minimal number of sensors, (b) maximizing the performance of the network, and (c) minimizing the deployment cost of the NIDS, particularly the number of required sensors. Efficient solutions will be weighted and scaled using the traditional linearization method.

This research article is structured as follows: Section II discusses the bibliographic search conducted to identify relevant works in the field of study, along with fundamental concepts pertaining to multi-objective and many-objective optimization problems in cybersecurity, cyber risk management, and the implementation of security controls. In Section III, the formal statement of the problem to be addressed is presented. Section IV provides detailed insights into the developed solution, highlighting key aspects of modeling and problem-solving. Section V elaborates on the experimental setup employed for evaluation purposes. Moving forward to VI we delve into a comprehensive discussion of the main results obtained. Finally, Section VII presents the conclusions drawn from the study, along with avenues for future research.

II. RELATED WORK

The document [12] represents the closest work to the ongoing research, as it puts forth a strategy aimed at effectively addressing many-objective optimization problems within the realm of cybersecurity defense. Specifically, this work focuses on formulating the defense problem by identifying and selecting security controls that simultaneously minimize security risk as well as direct and indirect costs. The approach adopted in this article involves modeling the problem as a min-max many-objective optimization, employing techniques such as binary linear programming.

In [18], a previous mention was made of optimizing the security of dynamic networks through the utilization of probabilistic graphs and linear programming. This method offers an approximate solution to the internal optimization problem, employing Taylor expansion and sequential linear programming. The primary focus of this work is to address the challenge of rigorously evaluating a range of network security defense strategies with the aim of reducing the probability of successful large-scale attacks on complex and dynamically evolving network architectures. To analyze the security of intricate networks and diminish the likelihood of successful attacks, the study introduces a probabilistic graph model and corresponding algorithms. Sequential linear programming, a scalable optimization technique, is employed, while a probabilistic model is utilized to account for uncertainties in network configurations. In [19], multiple objective functions encompassing security risk, direct costs, and indirect costs are proposed. However, these models primarily consider single-stage attacks rather than sequences of steps. Conversely, [20] introduces an approach that employs mixed-integer linear programming for optimal defense, which is further extended in [21] to include robustness and sensitivity analyses.

In the article [22], an alternative proposal is presented, utilizing three widely recognized metaheuristics to showcase the effectiveness of the technique in resolving optimization problems related to cybersecurity prevention in the Internet of Things. Furthermore, in the context of information security systems, [23] puts forth multi-criteria cost optimization method, employing the vector-evaluated genetic algorithm. Researchers in [24], driven by the escalating interaction in cyberspace, have devised optimization techniques for both attack and defense. These optimization approaches leverage artificial intelligence techniques powered by metaheuristics to identify and detect threats and attacks. Another study [25] proposes the utilization of particle swarm optimization as a technique to determine the optimal number of ad hoc mobiles within a network. These mobiles are grouped as sensors, functioning as intrusion detectors.

In [26], the authors also adopt the structure of the attack graph and put forth novel coverage models aimed at selecting an optimal portfolio of security controls to mitigate threat vulnerability. They propose a polynomial-time heuristic and a Bender's branch-and-cut algorithm, which efficiently provide near-optimal and exact solutions for large-scale scenarios. The study incorporates three robustness models that account for modeling uncertainty, assuming that the attack paths are enumerated and provided in a complete list. In contrast, our approach relaxes this assumption by allowing the attacker to exploit the most advantageous attack path among numerous alternatives. Similar, in [27], researchers propose a new framework for cybersecurity planning. The objective of the defense strategy is to prolong the time it takes for attacks to succeed, while the attacker aims for the shortest completion time. In our approach, we assume that the attacker selects the attack path with the highest probability of success.

In the realm of implementing a SIEM, the following research [28] emerges as a notable source, showcasing a practical illustration of monitoring security events within an organization's network. It is worth mentioning that security information and event management systems have gained widespread adoption as a robust tool for preventing, detecting, and responding to cyber-attacks, as reaffirmed by [29]. Over time, SIEM solutions have evolved into comprehensive, end-to-end systems that offer extensive visibility, enabling the identification of high-risk areas and proactive mitigation strategies to reduce costs and incident response time. The study also highlights that various companies have developed SIEM software products to detect network attacks and anomalies in IT system infrastructures. Noteworthy entities in this domain include esteemed IT companies such as HP, IBM, Intel, McAfee, as well as visionary options like AT&T Cybersecurity/AlienVault's SIEMs, and promising tools worth considering within the SIEM context, such as Splunk.

Another article highlights the challenges posed by the uncertainty, complexity, and diversity of data traffic in network intrusion behaviors. To tackle this issue, the article [30] employs a network-based intrusion detection algorithm and the particle optimizer algorithm as a detection method. In addressing the problem of ensuring information security in wireless sensor networks, the document [31] introduces a security information and event management methodology in conjunction with a multi-sensor or agent approach. The proposed approach seamlessly integrates the SIEM methodology with a multi-agent architecture comprising data collection agents, coordinator agents, and local intrusion detection systems. A server undertakes correlation analysis, identifies the most significant incidents, and assists in prioritizing incident response.

Numerous studies have explored the utilization of sensor networks for diverse purposes. In relation to this, the article [32] highlights the significance of the concept of Smart Cities and the monitoring of environmental parameters, which has garnered considerable scientific interest over the past decade. The study further emphasizes that the emergence of recent computing technologies, coupled with low-cost and low-power devices, has expanded the scope of research, enabling the development of monitoring devices and the implementation of Sensor Networks using Raspberry PI in a more accessible and expansive manner.

Regarding cybersecurity and sensor technology, the study [33], focuses on the development of a lightweight Intrusion Detection System (IDS) based on machine learning. This IDS incorporates a novel feature selection algorithm and is specifically designed and implemented on the Raspberry Pi platform. The performance of the system is assessed using a dataset obtained from an IoT environment, demonstrating that the detection system is lightweight enough to operate effectively within the Raspberry Pi environment, without compromising its detection performance.

Finally, works that include artificial intelligence algorithms, principally machine and deep learning mechanisms, can be seen in [34], [35], and [36]. Here, temporal networks -designed as a classical networking topology, are studied using well-known learning-based methods. Feature selection is used as an optimization problem to reduce the time of the training phase and keep the algorithm's performance. Last, interesting works were published in [37] and [38]. Here, again the optimization paradigm and the machine learning techniques cross their paths to model real-world applications.

III. PRELIMINARIES

In the current community, cybersecurity can benefit from applying optimization methods to address various challenges. The most common issue is intrusion detection and prevention, where optimization methods can be used to enhance intrusion detection and prevention systems [39]. Network traffic analysis and anomaly detection are other cases where optimization methods can attend to cybersecurity topics by analyzing network flow analysis, traffic pattern recognition, or anomaly detection algorithms. Thus, cybersecurity professionals can identify potential security breaches, unauthorized access attempts, or abnormal behaviors within the network [40]. Another open challenge is malware detection and classification. Here, optimization techniques can support the feature selection strategy, model parameters, ensemble methods, and machine learning algorithms for improving the effectiveness of cybersecurity defenses [41]. Finally, we consider vulnerability assessment and patch management as hot topics. Again, optimization mechanisms can be utilized to prioritize vulnerability assessments and patch management efforts. By considering factors like risk severity, system criticality, and available resources, optimization algorithms can help organizations allocate limited resources efficiently, ensuring that the most critical vulnerabilities are addressed promptly [42].

A. TECHNICAL CHARACTERISTICS OF A SIEM

Today, all computer systems face the constant threat of cyberattacks, necessitating ongoing security measures to mitigate potential risks [43]. As a result, technological infrastructures are fortified with various security components, including firewalls, intrusion detection systems, intrusion prevention systems, and security software installed on terminal devices [44]. However, these security controls operate independently, and comprehensive attack recognition requires the combination and correlation of logs and events from different security components [45]. This is precisely where a security information and event management (SIEM) system prove valuable. A SIEM system serves as the central platform within a security operation center, gathering events from multiple sensors such as intrusion detection systems, antivirus software, and firewalls. By correlating these events, it provides a unified view of alerts for managing threats and generating security reports. In line with this, the present research proposes the implementation of a SIEM as a centralized system, utilizing a server and distributed sensors, including intrusion detection systems, capable of collecting and identifying malicious activities and anomalous traffic.

In the cited study [28], a basic SIEM architecture consists of distinct blocks, namely source devices, log collection, parsing normalization, rules engine, log storage, and event monitoring. Each component operates independently, ensuring their individual functionality. Figure 1 visually depicts the fundamental constituents of a SIEM solution.

As stated in [46], the pivotal component of the SIEM system is the central engine responsible for tasks such as log filtering, analysis, monitoring, policy application, and alert generation. Additionally, it facilitates the transmission of logs to storage and forwards the generated information to the presentation layer. This enables security officers to view real-time network activities. Figure 2 illustrates the physical architecture of the SIEM, providing a visual representation of its structure.

B. TECHNICAL CHARACTERISTICS OF THE SENSOR MODULE

The sensor module utilizes a Raspberry Pi as its primary component. To achieve the desired outcomes, the sensor control core necessitates two Ethernet network ports—one connected to the internal subnet and the other linked to the central server housing the SIEM application. Operating in monitor mode, it can detect all incoming and outgoing data flow within the subnet. The details provided in the referenced article suggest that Raspberry Pi serves as a fully functional single-board computer, integrated onto a circuit board system. It operates on a Linux operating system and can be easily modified by replacing the board's memory. Like a computer, it can handle multiple tasks concurrently, including networks, data

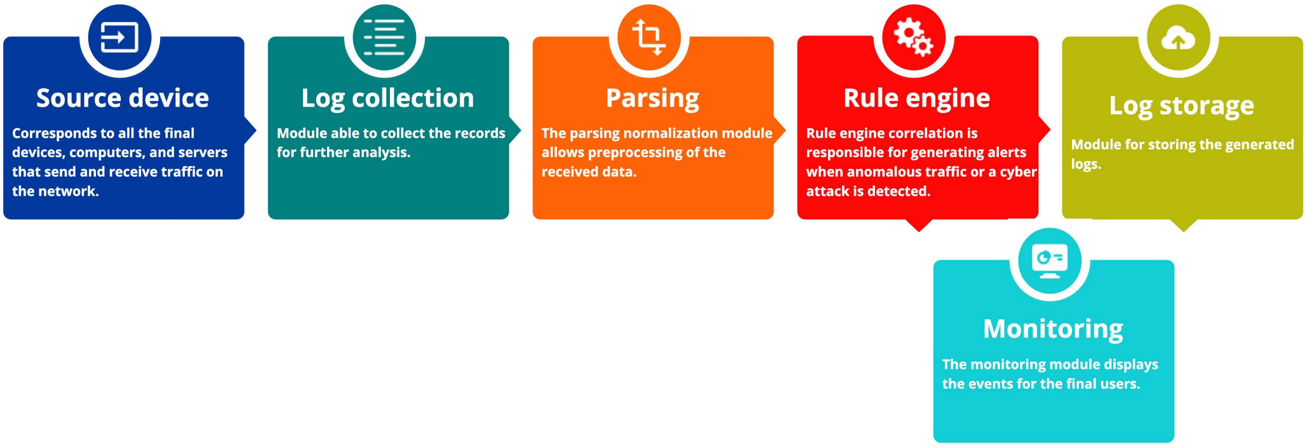

Source device
Corresponds to all the final devices, computers, and servers that send and receive traffic on the network.
Log collection
Module able to collect the records for further analysis.
Parsing
The parsing normalization module allows preprocessing of the received data.
Rule engine
Rule engine correlation is responsible for generating alerts when anomalous traffic or a cyber attack is detected.
Log storage
Module for storing the generated logs.
Monitoring
The monitoring module displays the events for the final users.

FIGURE 1. SIEM provides real-time monitoring, analysis of events, tracking, and logging security data for compliance or auditing purposes.

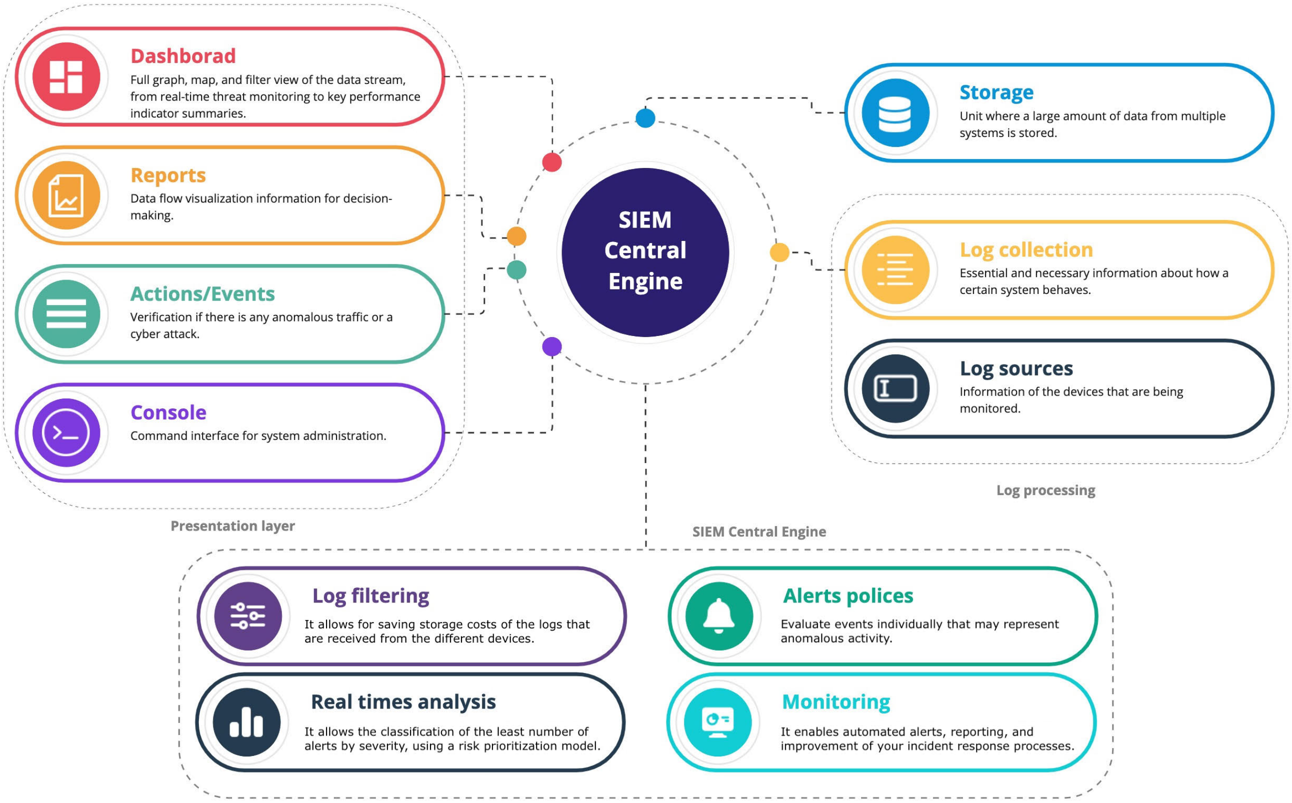

Dashborad
Full graph, map, and filter view of the data stream, from real-time threat monitoring to key performance indicator summaries.
Reports
Data flow visualization information for decision-making.
Actions/Events
Verification if there is any anomalous traffic or a cyber attack.
Console
Command interface for system administration.
Presentation layer
SIEM Central Engine
Storage
Unit where a large amount of data from multiple systems is stored.
Log collection
Essential and necessary information about how a certain system behaves.
Log sources
Information of the devices that are being monitored.
Log processing
SIEM Central Engine
Log filtering
It allows for saving storage costs of the logs that are received from the different devices.
Alerts polices
Evaluate events individually that may represent anomalous activity.
Real times analysis
It allows the classification of the least number of alerts by severity, using a risk prioritization model.
Monitoring
It enables automated alerts, reporting, and improvement of your incident response processes.

FIGURE 2. Physical architecture of the SIEM.

transmission, databases, and web servers, making it suitable for Raspberry Pi-based applications (see Figure 3). Furthermore, remote access is possible through Secure Shell. Given its affordability, small-scale and micro-scale entrepreneurs can leverage Raspberry Pi's integrated control kernel for real-world applications.

C. DESCRIPTION OF A ORGANIZATION'S TRADITIONAL NETWORK TOPOLOGY

In line with contemporary design principles, a modern traditional network topology is structured to address the demands of uptime and scalability. It adheres to a hierarchical model, encompassing intermediate devices like routers, switches, edge firewalls, and end devices such as computers and servers. Additionally, it incorporates redundant connectivity to the Internet through two Internet Service Providers (ISPs) for enhanced reliability [47]. Figure 4 shows a typical traditional network.

This proposal encompasses multiple network topologies, referred to as instances, beginning with the zero-network instance. The zero instance consists of five distinct subnets, each serving specific production areas with their respective

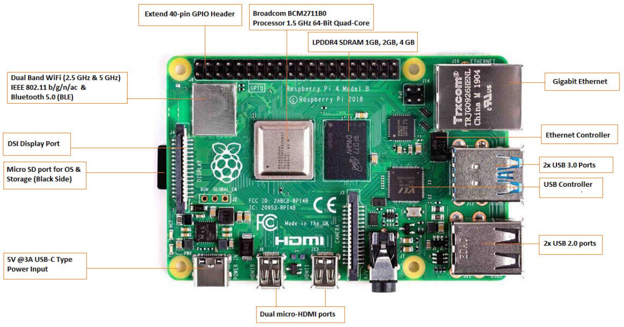


FIGURE 3. Ports, connectors, and chips of the Raspberry Pi 4 board.

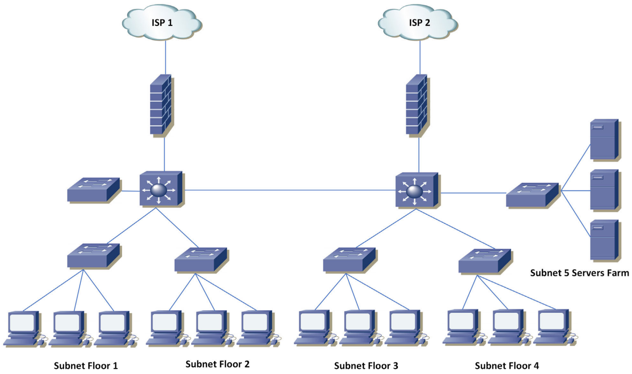


FIGURE 4. Traditional network topology.

technical and operational prerequisites. The first four subnets are dedicated to accommodating various production areas, while the fifth subnet is designated as the server farm, housing the organization's data center.

D. CYBER RISK MANAGEMENT PROBLEM

As computer networks continue to expand and become more intricate, the task of monitoring and safeguarding them becomes increasingly challenging [48]. Cyber risk management aims to protect valuable information assets through the application of effective techniques, disciplines, policies, and existing infrastructure. Selecting an appropriate cybersecurity plan can be a complex endeavor, given the multitude of security controls available, each capable of providing defense against potentially overlapping vulnerabilities. Examples of such security controls include inventory management of authorized and unauthorized devices and software, identity management and access control, adherence to password policies, patch management, configuration of network and/or application firewalls, deployment of antimalware and antiphishing software, staff training to counter social engineering attacks, data backup practices, resource redundancies, and physical security measures [49].

However, the management of cyber risks described above is deficient and fails to deliver the expected results. This inadequacy can be attributed to several factors, including the constant evolution and variation of new attacks that go undetected by conventional technological security systems due to their novel patterns of incidence. Moreover, existing regulations and established best practices often prove ineffective in mitigating security breaches.

It is important to recognize that cyber risk management is a process aimed at achieving an optimal balance between leveraging opportunities and minimizing vulnerability losses. This is typically accomplished by ensuring that the impact of threats exploiting vulnerabilities remains within acceptable limits and at a reasonable cost [50]. In the current cybersecurity landscape, organizations have been compelled to incorporate a set of robust security practices into their information management systems. These protective measures have become widely adopted, prompting various organizations to establish and implement information security standards. Traditional cyber risk management relies on qualitative instruments and expert judgments to address risk mitigation and resource optimization. However, some authors argue that such approaches [14] are subjective and incomplete, and therefore, should be complemented by a quantitative and objective approach.

This research proposes an approach to cyber risk management using a many-optimization strategy. The objective is to determine the optimal or near-optimal placement of network components to design an efficient network topology. The network topology instance depicted in 4 presents a solution for placing devices while ensuring specific constraints, such as availability, are met. This strategy is scalable and applicable to various network instances.

By employing linear programming modeling and optimization algorithms, this project aims to investigate whether the utilization of these techniques is suitable for enhancing the effectiveness of cyber risk management through the design of an efficient network topology. Additionally, it seeks to determine if the proposed many-objective optimization approach can effectively mitigate cyber risks within organizations.

Linear programming is a mathematical modeling technique that aims to accurately represent reality to understand its behavior and obtain solutions for specific actions. It has been successfully applied in various studies to address similar issues, demonstrating its effectiveness [51], [52], [53], [54]. Therefore, we believe that leveraging linear programming as a modeling process to optimize multiple information security variables can enhance cyber risk management. By maximizing benefits or minimizing costs, it enables better decision-making regarding threat mitigation, vulnerability remediation, and investment in monitoring technologies and security controls. Additionally, we recognize the value of utilizing bio-inspired algorithms to achieve near-optimal results, which can provide advanced planning for the placement of network intrusion detection systems (NIDS) within the organization's network.

Finally, we adopt the linear programming paradigm to model the multiple variables of cybersecurity, aiming to find solutions that maximize the benefits and minimize the costs associated with cyber risk management. By formulating the problem in this way, we can effectively optimize various aspects of cybersecurity and make informed decisions to enhance the overall management of cyber risks.

IV. DEVELOPED SOLUTION

Cybersecurity threats are ever evolving, with new strategies and techniques emerging constantly. Models and optimization techniques may prove less effective in defending against fresh and unforeseen threats and attacks due to their reliance on historical knowledge and data. As a result, it is crucial to frequently update and modify these models and optimization techniques to address emerging threats. However, some exact optimization techniques may demand extensive computational power and processing time to yield accurate results. This can pose challenges when rapid responses to security incidents are necessary or when operating in resource-constrained environments. Overcoming this limitation can involve leveraging simplified methods or more efficient optimization techniques capable of producing immediate results. By combining models, regularly validating with real-world data, and collaborating with cybersecurity experts, these constraints can be overcome, leading to improved cybersecurity outcomes.

Organizations across various sectors are increasingly encountering advanced threats, which drive up their operational costs. In response, IT security teams are implementing appropriate systems capable of promptly responding to these new threats. In this regard, it is imperative to establish a Security Information and Event Management (SIEM) system for managing and monitoring security events and incidents [55]. SIEM, when deployed on a server and complemented by distributed NIDS (Network Intrusion Detection System) sensors throughout the network, has evolved into a mature, reliable, and easy-to-use technology. Numerous solution providers offer scalable SIEM and NIDS solutions that can be tailored to different needs and budgets. The proposed solution can scale and adapt to networks of varying sizes and complexities, capable of adjusting to dynamic environments with multiple devices, users, and network segments. Furthermore, it can seamlessly integrate with other network infrastructure components such as firewalls, antivirus systems, and logging services, ensuring comprehensive visibility and threat detection capabilities.

A. PROBLEM MODELING

In this section, we detail how the modeling of the problem regarding the management of cyber risks is carried out, and we describe the characteristics of bio-inspired methods to solve this problem and find near-optimal solutions.

The proposed solution aims to optimize the management of cybernetic risk by modeling various instances of cybersecurity that encompass it. This modeling process allows us to derive objective functions that maximize the benefits of deploying sensors in specific locations within the organization's network while simultaneously minimizing the monetary costs associated with each sensor and the indirect costs incurred by not installing them. Additionally, the solution determines the corresponding decision variables that describe these objective functions and the constraints that define the feasible solution space. The optimization results obtained will provide valuable insights for decision-making, informing choices regarding the number of NIDS to utilize and their optimal placement within the network.

To effectively model the cyber risk management problem, the initial step involves assessing the organization's vital assets, ensuring the continuity of productive operations. This assessment adheres to the guidelines outlined in the ISO 27005 standard [56]. The primary objective is to identify and effectively manage the cyber risks that these information assets face, thereby safeguarding their confidentiality, integrity, and availability. An asset, in this context, refers to any element within the organization that possesses value and requires protection, as it contributes to the achievement of the organization's objectives [57]. Once the assets have been identified, each one will be assigned a value based on its significance in relation to business productivity. The availability of these assets assumes particular importance, as any disruption or unavailability could result in significant repercussions on the organization's overall productive capacity.

An optimization problem generally includes an objective function to be minimized, maximized, or both, subject to different constraints. In this research, we explore minimizing the monetary costs of the number of sensors used. Likewise, we also seek to maximize the benefits given that the sensor is installed in a particular place. Finally, we study to minimize the indirect effects of the sensor not being installed in a specific place. In short, we have a many-objective optimization, whose mathematical formulation we will describe below.

First, Equation (1), described as a vector, represents the modeling of the many-objective problem:

$$F(\vec{x}) = \langle f_1(\vec{x}), f_2(\vec{x}), f_3(\vec{x}) \rangle \tag{1}$$

where $\vec{x} = \langle x_{i1}, x_{i2}, \ldots, x_{in} \rangle^T$ defines a binary vector of n dimensions and represents the set of decision variables, therefore, $x_{ij} \in \{0, 1\}$. There can be several types of sensors, so we will use subindex i to denote it. Thus, $F(\vec{x})$ represents the set of mono-objective functions to be optimized.

$$f_1(\vec{x}) : \min_{x_{ij} \in X} \sum_{i=1}^{s} \sum_{j=1}^{n} x_{ij} c_{ij} \tag{2}$$

$$f_2(\vec{x}) : \max_{x_{ij} \in X} \sum_{j=1}^{n} x_{ij} d_{ij}, \ \forall i \tag{3}$$

$$f_3(\vec{x}) : \min_{x_{ij} \in X} \sum_{j=1}^{n} (1 - x_{ij}) i_{ij}, \ \forall i \tag{4}$$

Equation (2) defines the minimization of the cost of the sensors or acquisition price, represented by the variable c_{ij}. This research considers two types of NIDS ($s = 2$), whose value depends on their performance, technical characteristics, and manufacturer.

Equation (3) models the maximization of the direct benefits of a sensor by being located in a certain place. Variable d_{ij} takes values from a qualitative range of four-point scale. The variable d_{ij} will receive a very high rating when the operational importance of a subnet is very significant and represents high availability, whose assigned value for this condition is twenty. Similarly, d_{ij} will take a high value when the importance of a subnet is significant, receiving the value of fifteen. For a d_{ij} with medium importance, the variable will obtain the value of ten, a d_{ij} with low importance will receive the value of five, and a d_{ij} with very low importance it will get the value one. The values of parameters are assigned by using ISO 27005 as a reference.

Equation (4) expresses the objective function that models the minimization of indirect costs, given that the sensor is not located in a certain place. From a qualitative scale, the assigned values determine the impact of not having a sensor on a specific subnet.

The variable i_{ij} establishes the value of the indirect cost, assigning it a value of seven when it is a catastrophic impact that is not installed, a value of five for a severe impact, a value of three when the impact is medium, and a value of one when the impact is minimal. Finally, Equations (5) and (6) describe the set of constraints of the problem.

$$\sum_{j=1}^{n} x_{ij} \geq 1, \ \forall i \tag{5}$$

$$\frac{\sum_{j=1}^{n} p_j(1 - x_{ij})}{\sum_{j=1}^{n} p_j} \leq (1 - u), \ \forall i \tag{6}$$

Regarding the mathematical modeling constraints, Equation (5) tells us that the network must have at least one NIDS. The importance of having more than one NIDS distributed in the network is that it allows us to correlate events from different devices in the network, detecting threats and validating the SIEM implementation as a security control.

Concerning Equation (6), p_j represents the probability of non-operation for a given subnet. Therefore, these restriction forces have a NIDS in the subnets with the highest probability of failure, according to a general probability of network uptime represented by u.

In addition to edge firewalls and all the network design features described above, this research proposes implementing a security information and event management system, known by its acronym SIEM, for centralizing the storage and interpretation of security data. Security to control the management of cyber risks. This system comprises a central server where the SIEM solution and sensors called NIDS are installed, which are placed at specific points in the network. This research aims to optimize resources, such as the number of NIDS to use. This strategy will be detailed later.

Given the zero instance described above, a survey of the functional and operational characteristics of each subnet is carried out to determine if it is necessary to place a NIDS. The management of the assets of this zero instance is carried out to protect the operational continuity that guarantees availability in their respective subnets.

Given the qualitative scale described above to categorize by values the direct benefits given that the sensor is installed and the indirect costs given that the sensor is not installed, instance zero is described with their respective values by subnet. In addition, it is based on the premise that each subnet has a sensor corresponding to one of the two types. Once the optimization strategy has been carried out, leave the corresponding ones. Therefore, for each subnet, the following values are defined:

- Subnet Floor 1: The direct benefit d_{ij} is assigned the value 15, representing a high benefit if the sensor is placed in this subnet. The parameter i takes the value 3, corresponding to a half indirect cost if the sensor is not placed. This subnet has a drop probability of 0.65. A type 1 sensor is placed if required.
- Subnet Floor 2: The direct benefit d_{ij} is assigned the value 1, representing a very low benefit if the sensor is placed in this subnet. The parameter i takes the value 1, corresponding to a minimum indirect cost if the sensor is not placed. This subnet has a drop probability of 0.50. A type 1 sensor is placed if required.
- Subnet Floor 3: The direct benefit d_{ij} is assigned the value 10, representing a half benefit if the sensor is placed in this subnet. The parameter i takes the value 3, corresponding to a half indirect cost if the sensor is not placed. This subnet has a drop probability of 0.10. A type 2 sensor is placed if required.
- Subnet Floor 4: The direct benefit d_{ij} is assigned the value 5, representing a low benefit if the sensor is placed in this subnet. The parameter i takes the value 1, corresponding to a half indirect cost if the sensor is not placed. This subnet has a drop probability of 0.50. A type 1 sensor is placed if required.
- Subnet 5 Server Farm: The direct benefit d_{ij} is assigned the value 20, representing a very high benefit if the sensor is placed in this subnet. The parameter i takes the value 7, corresponding to a catastrophic indirect cost if the sensor is not placed. This subnet has a drop probability of 0.7. A type 2 sensor is placed if required.

Figure 5 shows instance zero of network topology.

B. BIO-INSPIRED METHODS

Bio-inspired optimization methods are optimization techniques that draw inspiration from nature and the behaviors of living beings to solve complex problems [58], [59]. These methods simulate biological processes and strategies developed in nature for adaptation and survival [60].

In this research, we utilize three population-based metaheuristic algorithms: particle swarm optimization, black hole algorithm, and bat optimization. These metaheuristic techniques are selected due to their popularity in swarm

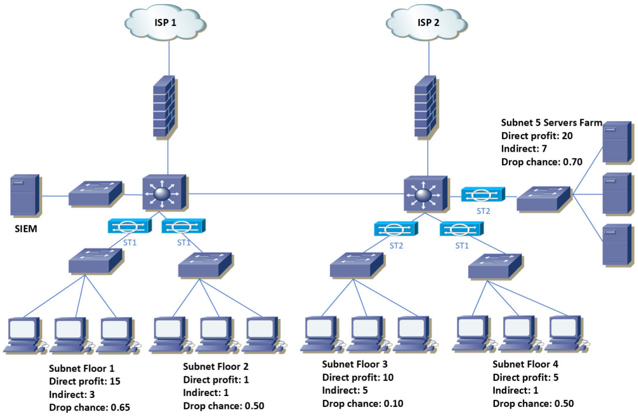


FIGURE 5. Zero instance of network topology.

intelligence methods and their similar working principles. Initially, the algorithms generate a random initial population, which is then improved by modifying the velocities of the swarm.

Solving multi-objective optimization problems related to locating security devices in networks for cybersecurity issues can present challenges in terms of computational complexity due to the combinatorial nature and multiple objectives involved. However, by carefully designing efficient strategies, it is possible to address this complexity and find quality solutions in a reasonable amount of time. In our case, computational complexity is kept in $O(kn)$ because we consider n as the number of subnetworks, and metaheuristics operate k-times in this dimension iteratively.

1) PARTICLE SWARM OPTIMIZATION

Particle swarm optimizer (PSO) is a population-based optimization technique inspired by the social behavior of birds flocking or fish schooling. It was introduced in 1995 by Kennedy and Eberhart as a simple yet powerful algorithm for solving optimization problems [61]. The main idea behind PSO is to simulate the social behavior of a group of individuals (particles) that move and search for the optimal solution in a search space.

PSO algorithm starts by randomly initializing a population of particles in the search space. Each particle represents a candidate solution to the optimization problem. The particles have a position and a velocity vector, updated at each iteration of the algorithm based on the best position found by each particle and the best position found by the entire swarm.

The update rule for the velocity of a particle i is given by:

$$v_i(t+1)=wv_i(t)+c_1r_1(pbest_i-x_i(t))+c_2r_2(gbest-x_i(t)) \tag{7}$$

where $v_i(t)$ is the velocity vector of particle i at time t, $x_i(t)$ is the position vector of particle i at time t, $pbest_i$ is the best position found by particle i so far, $gbest$ is the best position found by the entire swarm so far, w is the inertia weight, c_1 and c_2 are the cognitive and social parameters that control the influence of personal and global bests, and r_1 and r_2 are two random numbers in [0, 1].

The update rule for the position of a particle i is given by:

$$x_i(t+1)=x_i(t)+v_i(t+1) \tag{8}$$

where $x_i(t)$ is the current position of particle i at time t.

The inertia weight w controls the trade-off between the exploration and exploitation of the search space. A high value of w promotes exploration, whereas a low value of w promotes exploitation. The cognitive and social parameters influence personal and global bests, respectively. A high value of c_1 promotes exploitation, whereas a high value of c_2 promotes exploration.

PSO algorithm iteratively updates the position and velocity vectors of the particles until a stopping criterion is met. The stopping criterion can be a maximum number of iterations, a maximum computational time, or a target fitness value.

2) BLACK HOLE ALGORITHM

The Black Hole Optimizer (BHO) is a metaheuristic optimization algorithm inspired by the behavior of black holes in space. Hatamlou first proposed the algorithm in 2013 [62].

In the BHO algorithm, solutions to an optimization problem are represented as stars that are attracted toward a black hole, which represents the best solution found so far. Each particle has a position in the search space, and a velocity determines its movement (see Equation (9)) that is updated based on its current position and the positions of the other stars (see Equation (10)).

$$v_i^j(t+1)=r(bh^j-x_i^j(t)) \tag{9}$$

$$x_i^j(t+1)=x_i^j(t)+v_i^j(t+1) \tag{10}$$

The attraction towards the black hole is determined by the mass of each star, which is calculated based on its fitness value. The more fit a star is, the greater its mass and the stronger its attraction towards the black hole. As stars move toward the black hole, they can exchange information with each other and explore different regions of the search space.

Using a random component that reacts when a probability of involvement is attained, this method aims to break the deadlock. This mechanism, known as the event horizon, is crucial for regulating global and local searches. If any star crosses the black hole's event horizon will be absorbed by it. The radius of the event horizon is computed by Equation (11):

$$E=\frac{f(bh)}{\displaystyle\sum_{i=1}^{s} f(x_i)} \tag{11}$$

where s represents the number of stars, $f(bh)$ is the fitness value of the black hole, and $f(x_i)$ is the fitness value of the ith star. When the distance between a ith star and the black hole ($diff_i$) is less than the event horizon at an instant t, the star collapses into the black hole. The separation between a star and a black hole is the Euclidean distance calculated by

Equation (12) as follows:

$$diff_i(t) = \sqrt{[bh^1 - x_i^1(t)]^2 + \cdots + [bh^n - x_i^n(t)]^2} \quad (12)$$

This approach operates according to the eventual guidelines mentioned above. For example, if an actual random number between 0 and 1 is more significant than an input parameter, the event horizon provides variety among the solutions. Otherwise, the solutions will continue to intensify the current search area.

3) BAT OPTIMIZATION ALGORITHM

Bat optimization algorithm (BAT) is a metaheuristic optimization algorithm inspired by the echolocation behavior of micro-bats. The algorithm was proposed by Xin-She Yang in 2010 [63].

In the bat algorithm, solutions to an optimization problem are represented as virtual bats that fly through the search space. Each bat has a position and a velocity and emits ultrasonic pulses to search for food (i.e., optimal solutions). The loudness of the pulses represents the energy level of the bat, while the frequency represents the pulse rate.

The algorithm starts with a population of bats randomly distributed in the search space. Each bat then flies towards a random position in the search space, with its velocity (see Equations (13) and (14)) and direction determined by its current position (see Equation (15)), the position of the best bat found so far, and a random noise term. If a bat finds a new solution that is better than the current best solution, it updates its position and energy level and adjusts its pulse rate and loudness.

$$f_i = f_{min} + (f_{max} - f_{min})\beta \quad (13)$$

$$v_i^j(t+1) = (x_{best}^j - x_i^j(t))f_i \quad (14)$$

$$x_i^j(t+1) = x_i^j(t) + v_i^j(t+1) \quad (15)$$

where β is a uniformly distributed random value in the range $[0, 1]$, f_{min} is set to have a small value, and f_{max} varies according to the *max* variance allowed in each time step. Next, x_{best} describes the global best solution all bats generate during the search process.

The pulse rate and loudness of each bat are updated dynamically during the search process based on the quality of the solutions found so far. The algorithm also includes a mechanism for controlling the exploration-exploitation trade-off, which balances the search for new solutions with the exploitation of the best solutions found so far.

In this optimizer, the random walk mechanism leads the branching phase to alter a solution. The solution is generated by the current volume of the bat A_i and the maximum variation allowed $max(var)$ during a time step. This procedure is calculated by Equation (16).

$$x_{new}^j = x_{old}^j + \epsilon A_i max(var) \quad (16)$$

where ϵ is a random value in $[-1, 1]$.

Finally, the variation between loudness and pulse emission drives the intensification phase. This initiation occurs from the hunter's behavior when the bats recognize their prey. When it happens, it attenuates the volume and intensifies the pulse emission rate. This approach is calculated by Equation (17).

$$A_i = \alpha A_i, \quad r_i = r_i^{time=0}(1 - e^{-\gamma(time=t)}) \quad (17)$$

where α and γ are ad-hoc constants to control the intensification phase. For $0 < \alpha < 1$ and $\gamma > 0$, we get $A_i \to 0$, $r_i \to r_i^{(time=0)}$, $t \to 0$.

4) COMMON BEHAVIOR

Population-based metaheuristics show a common behavior. For that, Algorithm 1 details how bio-inspired solvers execute their search procedures. Now, to implement these algorithms, we adjust the particularities of each of them.

Algorithm 1 Common Work Scheme Used to Implement the Population-Based Algorithms

```
Input: popSize: the population size; $T$: the maximum
    time; Lb: Lower bound; Ub: Upper bound; particular
    parameters; $n$: dimensionality.
$(f_k, n) \leftarrow loadProblemData()$
objective functions $f_k(\vec{x})$, $x = \langle x^1, \ldots, x^n \rangle$
    $(\forall k = \{1, \ldots, K\})$
// produce the first generation of popSize agents
    (particles, stars or bats), randomly.
foreach agent $a$, $(\forall a = \{1, \ldots, popSize\})$ do
    foreach variable $j$, $(\forall j = \{1, \ldots, n\})$ do
        position $x_i^j(0) \leftarrow Random[Lb, Ub]$;
        velocity $v_i^j(0) \leftarrow 0$;
    end
    compute $f(x_i(0))$;
end
// produce $T$-generations of popSize agents.
$t \leftarrow 1$;
while $t < T$ do
    foreach agents $a$, $(\forall i = \{1, \ldots, popSize\})$ do
        if $f(x_i(t))$ is better than $f(x_g)$ then
            $x_g \leftarrow x_i(t)$;
        end
    end
    foreach variable $j$, $(\forall j = \{1, \ldots, n\})$ do
        update $v_i^j(t+1)$;
        update $x_i^j$;
    end
    compute $f(x_i(t))$;
    $t \leftarrow t + 1$;
end
return post-process results and visualization;
```

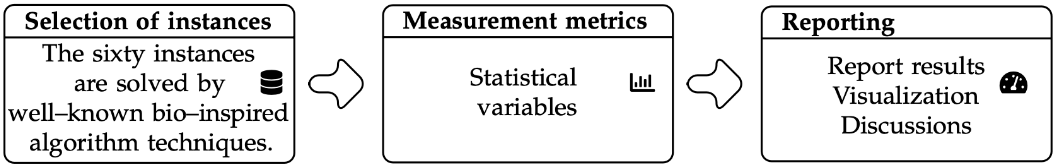


FIGURE 6. Schema of the experimental phase applied to this work.

V. EXPERIMENTAL SETUP

As part of this experimental stage, sixty instances have been created. These instances have random operational parameters, including the number of subnets, direct benefits, sensor costs, and indirect costs associated with the placement or absence of a sensor. The purpose is to cover a wide range of scenarios, ensuring the inclusion of various possibilities. (see Table 1).

After the solution vector changes, a binarization step is necessary to apply continuous metaheuristics in a binary domain [64]. Sigmoid function is contrasted against a uniform random value δ between 0 and 1. Next, a transformation function, i.e., $[1/(1+e^{-x_i^j})] > \delta$, is used as a discretization method, in this case, if the sentence is true, then $x_i^j \leftarrow 1$. Otherwise, $x_i^j \leftarrow 0$.

To properly assess the performance of swarm intelligence methods, robust performance analysis is required. For that, we compare the best solutions achieved by the metaheuristics with the result of the sixty instances. Figure 6 describes the procedures involved in the experiments.

We design goals and suggestions for the experimental phase to show that the proposed approach is a viable alternative for solving the location of NIDS. Solving time is computed to determine how long metaheuristics take to reach the best solutions. We employ the best value as a vital indicator for assessing future results, computed by Equation (18).

$$\sum_{(p,q)_{p\neq q}\in K} \underbrace{\frac{f_p(\vec{x})}{e_p(\vec{x}^{best})}\omega_p}_{max} + \underbrace{\frac{\widehat{c}-f_q(\vec{x})}{\widehat{c}-e_q(\vec{x}^{best})}\omega_q}_{min}, \qquad \omega_{(p,q)} \geqslant 0 \tag{18}$$

where $\omega_{(p,q)}$ represents weight of objective functions and $\sum \omega_{(p,q)} = 1$ must be satisfied. Values of $\omega_{(p,q)}$ is defined by analogous estimating. $f_{(p,q)}(\vec{x})$ is the single-objective function and $e_{(p,q)}(\vec{x}^{best})$ stores the best value met independently. Finally, $\widehat{c}$ is an upper bound of minimization single-objective functions.

Next, we apply ordinal analysis to evaluate whether the strategy is proper. Finally, we detail the hardware and software used to replicate computational experiments. Results will visualize in tables and graphics.

We highlight that test scenarios are created from conventional simulated networks that are prototyped to emulate the behavior and characteristics of real-world networks, which reflect the execution characteristics of any network of a given organization, including small, medium, and large networks. Depending on its scope and size, determined by the number of subnets, every network comprises devices such as computers, routers, switches, server farms, and their corresponding interconnections. This research considers test networks ranging from small networks of five subnets, through medium networks of fifteen subnets, to large networks of up to thirty-nine subnets. The emulation of the test networks, given their operational and functional characteristics, considers the limitations of bandwidth, latency, packet loss, and network congestion, which among other factors, determine the uptime of each subnet. Since the uptime of a network refers to the duration or percentage of time that the network remains operational and accessible without experiencing significant downtime, for this investigation, the networks must have a minimum uptime of 90% since interruptions and downtime may occur—downtime caused by equipment failures, network congestion, connectivity failures. Given the proactive monitoring of the SIEM implementation, it will ensure high uptime on the network.

All algorithms were finally coded in the Java 1.8 programming language. The infrastructure was a workstation running Windows 11 Pro operating system with seven processors i7 8700, and 16 GB of RAM. Parallel implementation was not required.

VI. DISCUSSION

The main results are illustrated in Tables 2-4 which correspond to the executions of the PSO, BAT, and BH algorithms, respectively. These tables provide an overview of the statistical variables that reflect the performance of the sixty instances used in the experiments. For each instance, the tables display the number of subnets, the best solution, the worst solution, the average, the median, the standard deviation, the interquartile range, the best time, and the average time of the set of solutions obtained. Furthermore, the tables indicate the optimal number of sensors to be placed in each instance, allowing for a comprehensive assessment of the algorithms' performance in achieving the desired objectives. The statistical information presented in these tables facilitates a thorough evaluation and comparison of the algorithms' effectiveness in solving the problem.

According to the analysis of Tables 2-4, it is evident that the three algorithms (PSO, BAT, and BH) produce comparable results for the initial instances. The solutions obtained by these algorithms are identical, indicating their effectiveness in finding the best solution. Additionally, when considering the convergence times of the algorithms, it is observed that from the first to the nineteenth instance, all three algorithms

TABLE 1. Description of the instances.

Instance	Number of subnets	Type of sensors	Uptime	Range of direct costs	Qualitative profit-range	Range of indirect costs	Performance of subnets
1	5	2	90%	[100-150]	[10-20]	[1-5]	[0.55-0.90]
2	5	2	90%	[100-150]	[5-15]	[1-7]	[0.23-0.92]
3	5	2	90%	[100-150]	[1-20]	[1-5]	[0.28-0.73]
4	6	2	90%	[100-150]	[10-20]	[1-5]	[0.17-0.98]
5	7	2	90%	[100-150]	[1-20]	[1-7]	[0.12-0.67]
6	8	2	90%	[100-150]	[1-20]	[1-7]	[0.17-0.98]
7	9	2	90%	[100-150]	[1-20]	[1-7]	[0.25-0.58]
8	9	2	90%	[100-150]	[1-20]	[1-7]	[0.25-0.85]
9	9	2	90%	[100-150]	[1-20]	[1-7]	[0.55-0.90]
10	10	2	90%	[100-150]	[1-20]	[1-7]	[0.05-0.96]
11	10	2	90%	[100-150]	[1-20]	[1-7]	[0.39-0.80]
12	10	2	90%	[100-150]	[1-20]	[1-7]	[0.10-0.80]
13	10	2	90%	[100-150]	[1-20]	[1-7]	[0.02-0.80]
14	10	2	90%	[100-150]	[1-20]	[1-5]	[0.11-0.80]
15	10	2	90%	[100-150]	[1-20]	[1-7]	[0.14-0.94]
16	11	2	90%	[100-150]	[1-20]	[1-7]	[0.14-0.94]
17	12	2	90%	[100-150]	[1-20]	[1-7]	[0.14-0.94]
18	12	2	90%	[100-150]	[1-20]	[3-5]	[0.07-0.96]
19	13	2	90%	[100-150]	[1-20]	[3-7]	[0.23-0.96]
20	14	2	90%	[100-150]	[10-20]	[1-7]	[0.50-0.89]
21	15	2	90%	[100-150]	[1-15]	[1-7]	[0.02-0.96]
22	15	2	90%	[100-150]	[1-20]	[1-7]	[0.07-0.97]
23	15	2	90%	[100-150]	[1-20]	[1-7]	[0.20-0.98]
24	15	2	90%	[100-150]	[1-20]	[1-7]	[0.05-0.98]
25	15	2	90%	[100-150]	[1-20]	[1-7]	[0.16-0.73]
26	16	2	90%	[100-150]	[1-20]	[1-7]	[0.10-0.99]
27	17	2	90%	[100-150]	[1-20]	[1-7]	[0.28-0.90]
28	18	2	90%	[100-150]	[1-20]	[1-7]	[0.10-0.96]
29	19	2	90%	[100-150]	[1-20]	[1-7]	[0.08-0.98]
30	20	2	90%	[100-150]	[1-20]	[1-7]	[0.06-0.98]
31	20	2	90%	[100-150]	[1-20]	[1-7]	[0.06-0.98]
32	20	2	90%	[100-150]	[1-20]	[1-7]	[0.01-0.98]
33	20	2	90%	[100-150]	[1-20]	[1-7]	[0.01-0.99]
34	20	2	90%	[100-150]	[1-20]	[1-7]	[0.10-0.82]
35	21	2	90%	[100-150]	[1-20]	[1-7]	[0.08-0.94]
36	22	2	90%	[100-150]	[1-20]	[1-7]	[0.07-0.97]
37	23	2	90%	[100-150]	[1-20]	[1-7]	[0.15-0.95]
38	24	2	90%	[100-150]	[1-20]	[1-7]	[0.02-0.97]
39	25	2	90%	[100-150]	[1-20]	[1-7]	[0.09-0.84]
40	25	2	90%	[100-150]	[1-20]	[1-7]	[0.09-0.91]
41	25	2	90%	[100-150]	[1-20]	[1-7]	[0.01-0.95]
42	26	2	90%	[100-150]	[1-20]	[1-7]	[0.02-0.96]
43	27	2	90%	[100-150]	[1-20]	[1-7]	[0.05-0.85]
44	28	2	90%	[100-150]	[1-20]	[1-7]	[0.17-0.96]
45	29	2	90%	[100-150]	[1-20]	[1-7]	[0.05-0.92]
46	30	2	90%	[100-150]	[1-20]	[1-7]	[0.06-0.92]
47	30	2	90%	[100-150]	[1-20]	[1-7]	[0.01-0.87]
48	30	2	90%	[100-150]	[1-20]	[1-7]	[0.02-0.98]
49	30	2	90%	[100-150]	[1-20]	[1-7]	[0.10-0.99]
50	30	2	90%	[100-150]	[1-20]	[1-7]	[0.02-0.96]
51	30	2	90%	[100-150]	[1-20]	[1-7]	[0.06-0.90]
52	31	2	90%	[100-150]	[1-20]	[1-7]	[0.04-0.95]
53	32	2	90%	[100-150]	[1-20]	[1-7]	[0.02-1.00]
54	33	2	90%	[100-150]	[1-20]	[1-7]	[0.05-0.93]
55	34	2	90%	[100-150]	[1-20]	[1-7]	[0.06-0.99]
56	35	2	90%	[100-150]	[1-20]	[1-7]	[0.03-0.98]
57	36	2	90%	[100-150]	[1-20]	[1-7]	[0.05-0.96]
58	37	2	90%	[100-150]	[1-20]	[1-7]	[0.08-0.93]
59	38	2	90%	[100-150]	[1-20]	[1-7]	[0.01-0.97]
60	39	2	90%	[100-150]	[1-20]	[1-7]	[0.01-0.99]

TABLE 2. PSO results table for sixty instances.

ID	Best Solution	Worst Solution	Average	Median	Standard Deviation	Interquartile Range	Better Time	Average Time	Number of sensors to place
1	654	654	654	654	0	0	1	8.10	5
2	426	426	426	426	0	0	0	9.10	4
3	555	555	555	555	0	0	1	3.93	4
4	602	602	602	602	0	0	0	13.33	5
5	634	634	634	634	0	0	0	3.23	5
6	641	641	641	641	0	0	1	4.93	6
7	846	846	846	846	0	0	1	14.17	7
8	854	854	854	854	0	0	1	20.83	8
9	935	935	935	935	0	0	1	25.60	8
10	765	765	765	765	0	0	1	13.03	7
11	950	950	950	950	0	0	2	15.70	9
12	773	733	733	733	0	0	1	12.17	7
13	822	822	822	822	0	0	1	10.63	7
14	872	872	872	872	0	0	2	10.73	8
15	807	807	807	807	0	0	1	13.87	8
16	1003	1003	1003	1003	0	0	1	13.40	11
17	1098	1098	1098	1098	0	0	2	22.10	10
18	1101	1101	1101	1101	0	0	2	22.87	12
19	1969	1069	1069	1069	0	0	3	29.37	10
20	380	638	507.17	469.50	79.16	84	1	26.13	4
21	1083	1136	1091.83	1083	20.1	0	3	19.93	11
22	1189	1260	1217.40	1189	35.38	71	3	21.90	10
23	1263	1267	1264.07	1263	1.80	4	3	19.10	11
24	1187	1223	1188.73	1187	6.62	0	2	24.77	11
25	1215	1215	1215	1215	0	0	3	27.10	12
26	1278	1285	1279.17	1278	2.65	0	4	78.60	12
27	1394	1406	1394.80	1394	3.04	0	9	96.27	14
28	1373	1448	1409.13	1427	28.44	54	6	88.03	18
29	1437	1489	1449	1437	19.04	19	11	76.03	14
30	1520	1609	1535.23	1528	22.83	18	17	95.57	14
31	1556	1637	1 588.97	1580	30.12	72	13	101.53	15
32	1417	1508	1442.53	1431	28.03	50	11	63.53	14
33	1481	1609	1531.67	1541	31.07	20.75	17	72.30	14
34	1535	1622	1583.23	1591	33.17	75	20	144.50	15
35	1311	1451	1355.07	1341.50	46.72	95	7	50.40	14
36	1544	1689	1625.60	1627	22.93	16.25	10	51.83	14
37	1737	1830	1780.37	1775	35.14	77.25	117	326.23	17
38	1593	1766	1681	1693	45.22	54.75	94	175.07	16
39	1980	2069	2011.23	1991	31.01	55.75	192	355.40	19
40	1737	1889	1820.70	1832.50	49.07	75	101	176.77	17
41	1699	1901	1800.47	1795.50	62.87	37.75	219	421.80	17
42	1761	1863	1817.87	1825.50	35.03	59.75	77	331.07	18
43	1922	2068	1981.57	2001	44.98	88	515	1137.40	20
44	2079	2188	2119.10	2117	38.24	61	325	780.53	21
45	2025	2162	2120.77	2140	40	70.75	222	517.23	20
46	2022	2235	2116.20	2136	48.32	85	372	665.90	20
47	1993	2183	2099.87	2085	41.85	42	406	1041.97	21
48	1928	2235	2083.63	2081	70.45	104.75	122	370.07	19
49	2238	2354	2295.23	2295	37.31	71	223	630.27	22
50	2078	2325	2234.70	2225	62	112.25	249	665.43	20
51	2500	2608	2545.83	2532	31.92	51.50	541	1270.57	23
52	2214	2402	2318.33	2328.50	47.01	52	910	2891.23	21
53	2234	2445	2328.93	2318.50	58.57	94.75	405	1460.93	22
54	2223	2431	2336.87	2329.50	48.57	57.75	828	1847.07	24
55	2738	2950	2854.83	2846	52.32	68.50	1376	3115.77	25
56	2721	2921	2852.87	2856.50	43.98	50.50	1171	3323.37	26
57	2799	2970	2898	2921.50	52.68	70.25	3521	5054.53	27
58	2704	2980	2854.87	2854.50	64.95	85.50	7509	14508.23	27
59	2755	2978	2889.90	2899.50	59.51	88.25	2690	5624.80	26
60	2761	2976	2854.60	2846.50	59	82	2048	4215.77	28

TABLE 3. BAT results table for sixty instances.

ID	Best Solution	Worst Solution	Average	Median	Standard Deviation	Interquartile Range	Better Time	Average Time	Number of sensors to place
1	654	654	654	654	0	0	1	8.53	5
2	426	426	426	426	0	0	<0	7.13	4
3	555	555	555	555	0	0	<0	12.40	4
4	602	602	602	602	0	0	1	22.13	5
5	634	634	634	634	0	0	<0	11.30	5
6	641	641	641	641	0	0	1	12.97	6
7	749	749	749	749	0	0	1	15.30	7
8	854	854	854	854	0	0	1	32.80	8
9	935	935	935	935	0	0	1	32.60	8
10	765	765	765	765	0	0	2	11	7
11	950	950	950	950	0	0	2	17.50	9
12	773	773	773	773	0	0	2	23	7
13	822	822	822	822	0	0	2	13.37	7
14	872	872	872	872	0	0	4	20.47	8
15	807	888	809.70	807	14.79	0	2	22.50	8
16	1003	1003	1003	1003	0	0	2	18.10	9
17	1098	1098	1098	1098	0	0	5	25.13	10
18	1101	1101	1101	1101	0	0	2	36.97	10
19	1069	1157	1080	1069	30.43	0	2	43.10	10
20	380	638	507.17	469.50	79.16	135	1	33.27	4
21	1083	1143	1104.90	1083	27.35	53	4	46.03	11
22	1189	1349	1233.27	1260	42.45	71	4	37.83	10
23	1263	1296	1269.87	1263	10.87	23	3	32.13	11
24	1187	1273	1202.07	1187	26.63	36	5	45.20	11
25	1215	1234	1216.03	1215	3.72	0	8	52.33	12
26	1278	1285	1278.70	1278	2.14	0	9	100.27	12
27	1394	1406	1394.40	1394	2.19	0	93	224.17	14
28	1373	1449	1402.87	1400	31.11	54	89	164.53	13
29	1437	1535	1461.90	1444	33.25	46.25	96	203.37	14
30	1520	1632	1553.17	1553.50	30.14	31	83	173.27	14
31	1556	1634	1588.67	1580	27.85	72	102	196.63	15
32	1417	1513	1467.47	1473	35.60	78.25	11	153.23	14
33	1481	1631	1516.23	1481	42.43	66.50	23	143.20	14
34	1535	1674	1579.80	1591	40.04	75	109	293.53	15
35	1311	1454	1356.13	1324	48.96	97.50	20	106.37	14
36	1531	1779	1637.63	1629	56.89	68.75	26	108.67	14
37	1737	1830	1775	1784	30.77	39	99	413.30	17
38	1593	1830	1684.47	1692.50	51.46	75.50	101	257.73	16
39	1946	2081	2006.70	1985	44.32	75.50	90	502.07	18
40	1743	1976	1842.10	1845	55.53	63.50	135	341.43	17
41	1699	1821	1755.70	1786	44.16	67.50	120	492.93	17
42	1761	1941	1846.77	1851	49.74	80	280	591.03	18
43	1922	2098	1996.17	2001	48.13	35.25	500	1951.03	20
44	1984	2247	2107.40	2114.50	81.77	138	514	1223.47	21
45	1995	2242	2137.67	2148.50	64.43	98.75	221	783.83	19
46	2022	2235	2116.20	2136	48.32	85	372	665.99	20
47	1989	2191	2112.27	2091	46.90	76.50	607	1509.77	21
48	2016	2325	2113.97	2104.50	82.78	122.25	184	493.73	20
49	2238	2345	2283.83	2282	35.33	61.50	298	818.77	22
50	2119	2331	2235.50	2234.50	65.36	120	364	991.20	20
51	2504	2614	2563.47	2571	35.90	72.25	1027	2148.80	23
52	2145	2462	2324.37	2323	80.33	103.25	585	1877.20	21
53	2187	2442	2329.27	2340	66.41	104.75	1261	2290.43	22
54	2225	2422	2337.50	2323	48.50	59	1601	3375.93	24
55	2739	2936	2847.10	2838	49.34	61.50	3193	5617.13	25
56	2746	2945	2863.07	2876	49.51	55.25	2559	5716.70	26
57	2716	3057	2896.97	2901	65.33	75.25	5184	10194.57	26
58	2670	2972	2827.87	2834	76.49	114.50	9818	28525.57	26
59	2747	3036	2889.97	2915	91.04	164.25	5920	11411.67	26
60	2748	2992	2856.57	2860	60.38	96	4318	8849.777	28

TABLE 4. BH results table for sixty instances.

ID	Best Solution	Worst Solution	Average	Median	Standard Deviation	Interquartile Range	Better Time	Average Time	Number of sensors to place
1	654	654	654	654	0	0	2	11.63	5
2	426	426	426	426	0	0	1	6.70	4
3	555	555	555	555	0	0	<0	5.53	4
4	602	602	602	602	0	0	2	19.13	5
5	634	634	634	634	0	0	1	10.97	5
6	641	641	641	641	0	0	1	16.03	6
7	749	749	749	749	0	0	2	22.67	7
8	854	854	854	854	0	0	3	24.07	8
9	935	935	935	935	0	0	3	36.07	8
10	765	765	765	765	0	0	2	11	7
11	950	950	950	950	0	0	6	28.77	9
12	773	773	773	773	0	0	3	24.67	7
13	822	822	822	822	0	0	2	19.23	7
14	872	872	872	872	0	0	4	20.47	8
15	807	807	807	807	0	0	3	26.03	8
16	1003	1003	1003	1003	0	0	4	29.87	9
17	1098	1098	1098	1098	0	0	7	39.37	10
18	1101	1101	1101	1101	0	0	10	66.10	10
19	1069	1069	1069	1069	0	0	8	77.30	10
20	380	550	460	463	52.72	39.75	2	33.17	4
21	1083	1136	1084.77	1083	9.68	0	87	132	11
22	1189	1189	1189	1189	0	0	12	55.87	10
23	1263	1263	1263	1263	0	0	10	44.40	11
24	1187	1187	1187	1187	0	0	88	126	11
25	1215	1215	1215	1215	0	0	80	153.63	15
26	1278	1285	1278.23	1278	1.28	0	96	251.80	12
27	1394	1394	1394	1394	0	0	292	442.20	14
28	1373	1428	1392.83	1373	26.51	54	126	365.93	13
29	1437	1448	1439	1437	3.44	7	288	479.37	14
30	1520	1550	1523.33	1520	6.17	4	198	305.90	14
31	1556	1630	1562.47	1556	15.65	0	319	541.47	15
32	1417	1506	1426.47	1420.50	18.28	12	209	429.80	14
33	1481	1549	1519.93	1541	30.21	60	288	351.70	14
34	1535	1610	1556.03	1535	30.57	56	400	868.53	15
35	1311	1415	1332	1316	30.37	48	108	233.33	14
36	1531	1632	1589.33	1611.50	40.71	81	109	282.03	14
37	1737	1831	1766.40	1747	31.35	40.25	675	1090.07	17
38	1593	1718	1647.83	1650	39.09	76	329	861.83	16
39	1946	2030	1983.47	1984	14.34	8	840	1225.30	18
40	1737	1890	1798.43	1803	34.19	48	401	733.77	17
41	1699	1821	1755.70	1786	44.16	86	597	967.10	17
42	1761	1811	1788.10	1794	17.42	35	812	1384.87	18
43	1922	2011	1970.93	1991.50	36.20	75.50	1980	2911.03	20
44	1984	2140	2075.83	2084	40.99	13	2494	3237.10	21
45	1990	2153	2072.90	2073	45.40	45.75	1340	2042.83	19
46	1971	2150	2072.53	2066.50	54.21	88.50	1484	2103.87	19
47	1991	2147	2067.37	2085	44.93	37.75	3700	4563.17	21
48	1921	2122	2021.77	2022	53.28	51.25	864	1232.07	19
49	2238	2342	2278.30	2280.50	29.90	42	1852	2496.23	22
50	2073	2281	2188.77	2183	51.57	55.50	2282	2890.27	20
51	2500	2571	2522.83	2514.50	21.67	24.75	5472	6748.80	23
52	2148	2357	2266.17	2256.50	44.20	71.75	3926	4966.77	21
53	2176	2357	2277.97	2279.50	42.26	32.25	6657	14253.40	22
54	2223	2340	2310.97	2319	29.27	17.75	8586	10542.10	24
55	2739	2879	2818.97	2834.50	44.84	58	17794	20314	25
56	2748	2870	2813.70	2813	33.22	41.25	17486	21963.80	26
57	2773	2918	2848	2850.50	36.48	75.25	30885	46754.20	26
58	2701	2898	2809.28	2829	53.28	78	69809	86407.56	27
59	2748	2929	2827.63	2835.50	35.05	34	17659	26008.90	26
60	2761	2896	2822.55	2821	39.98	67.75	26687	30998.77	28

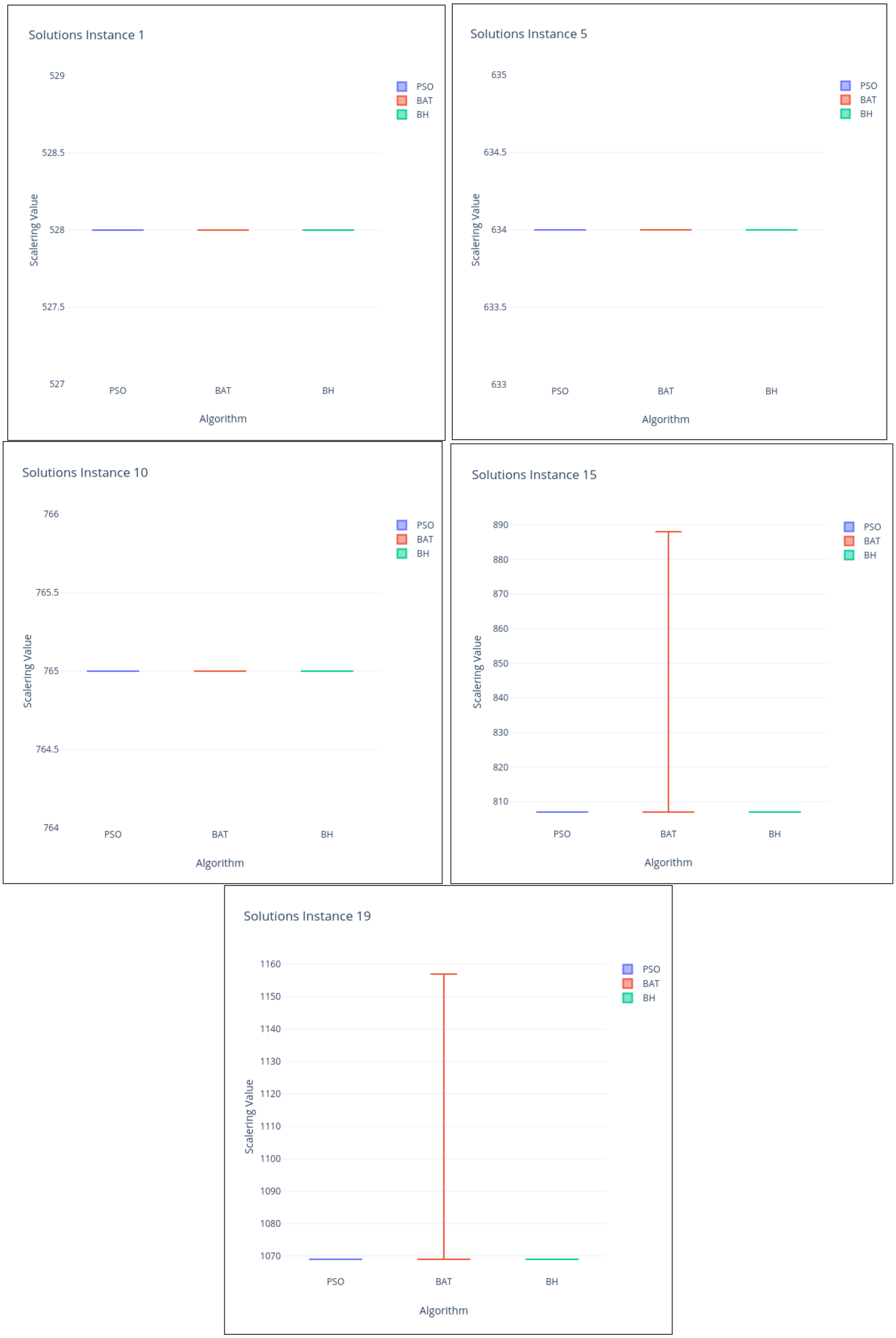


FIGURE 7. Distributions of best solutions (1/3).

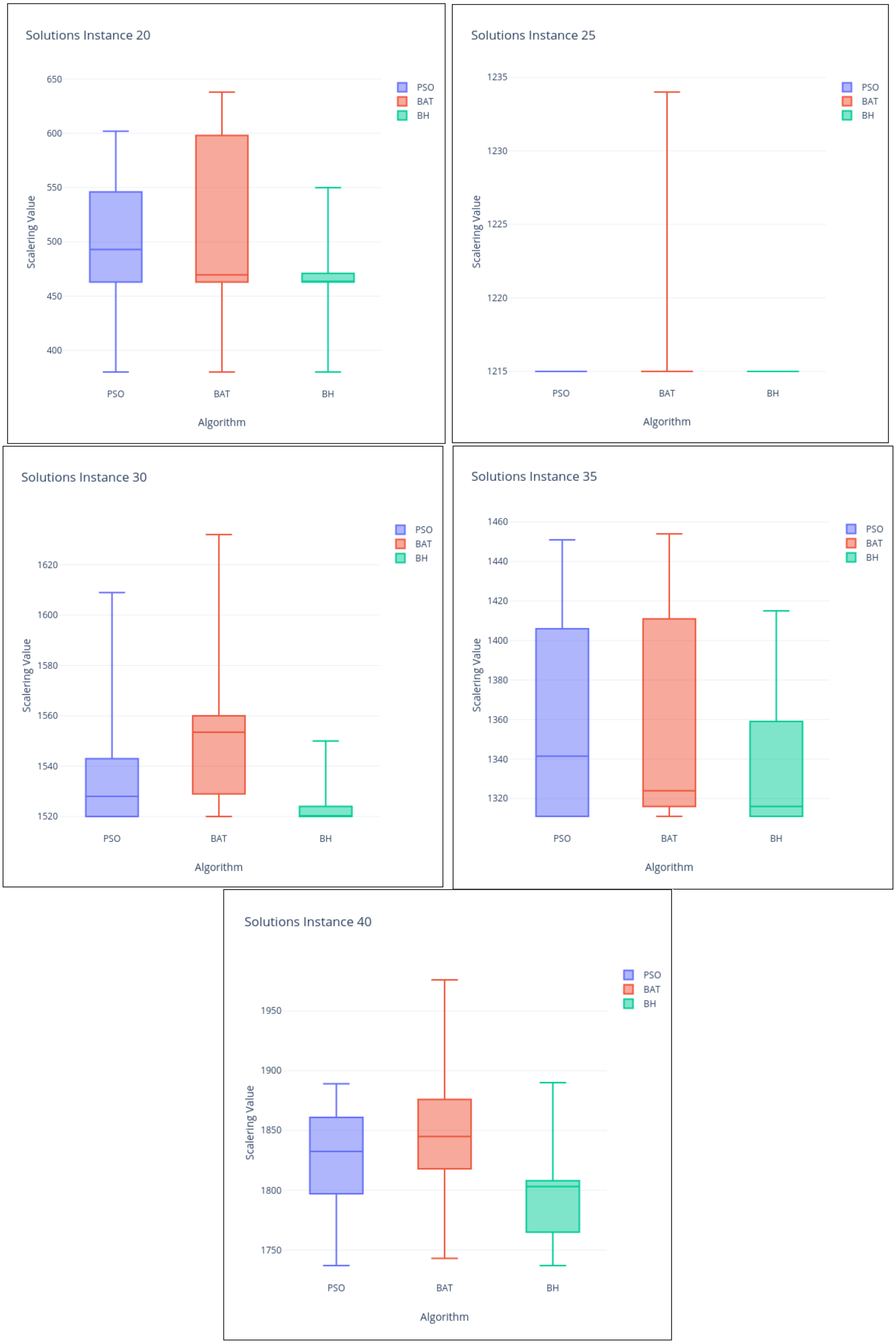


FIGURE 8. Distributions of best solutions (2/3).

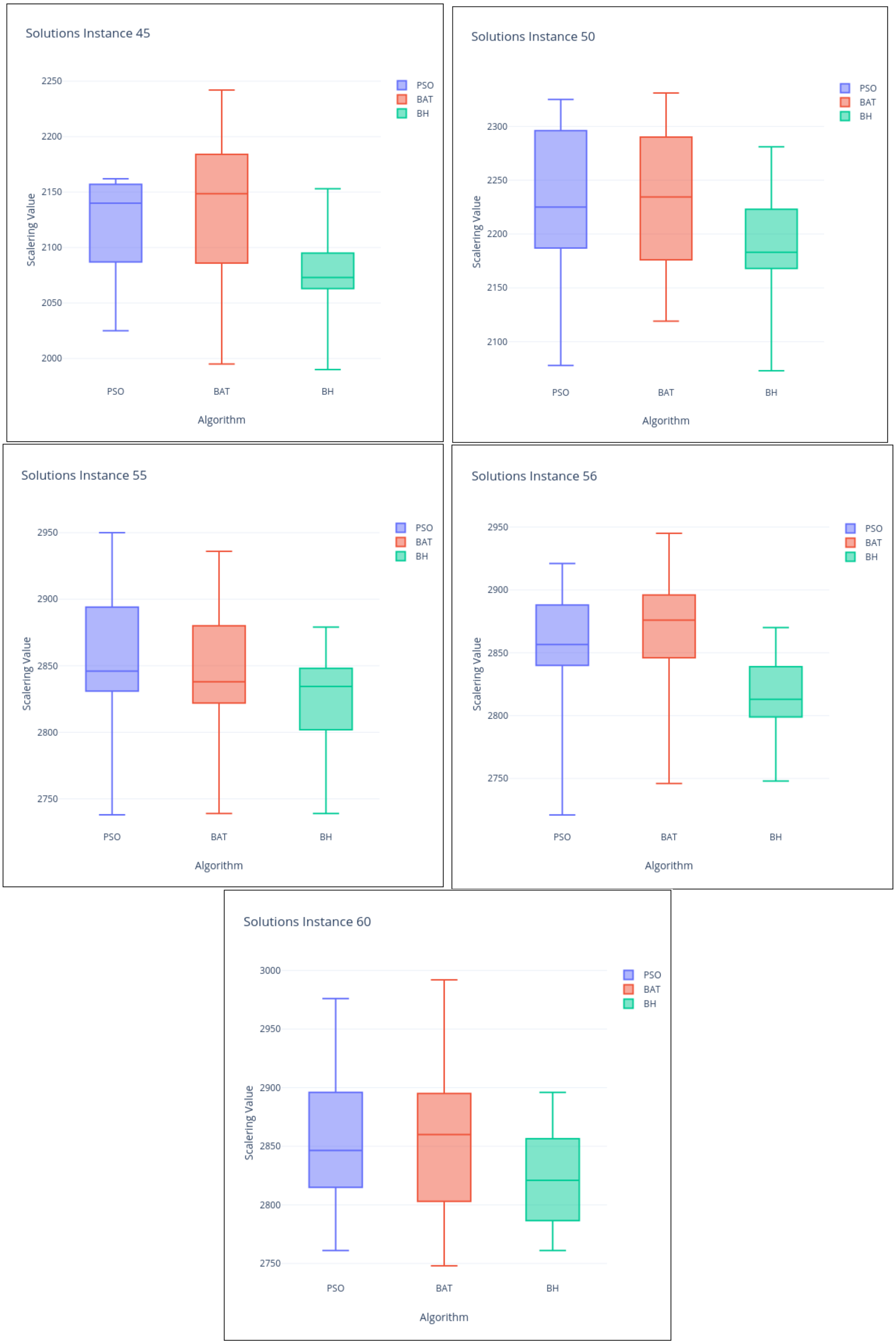


FIGURE 9. **Distributions of best solutions (3/3).**

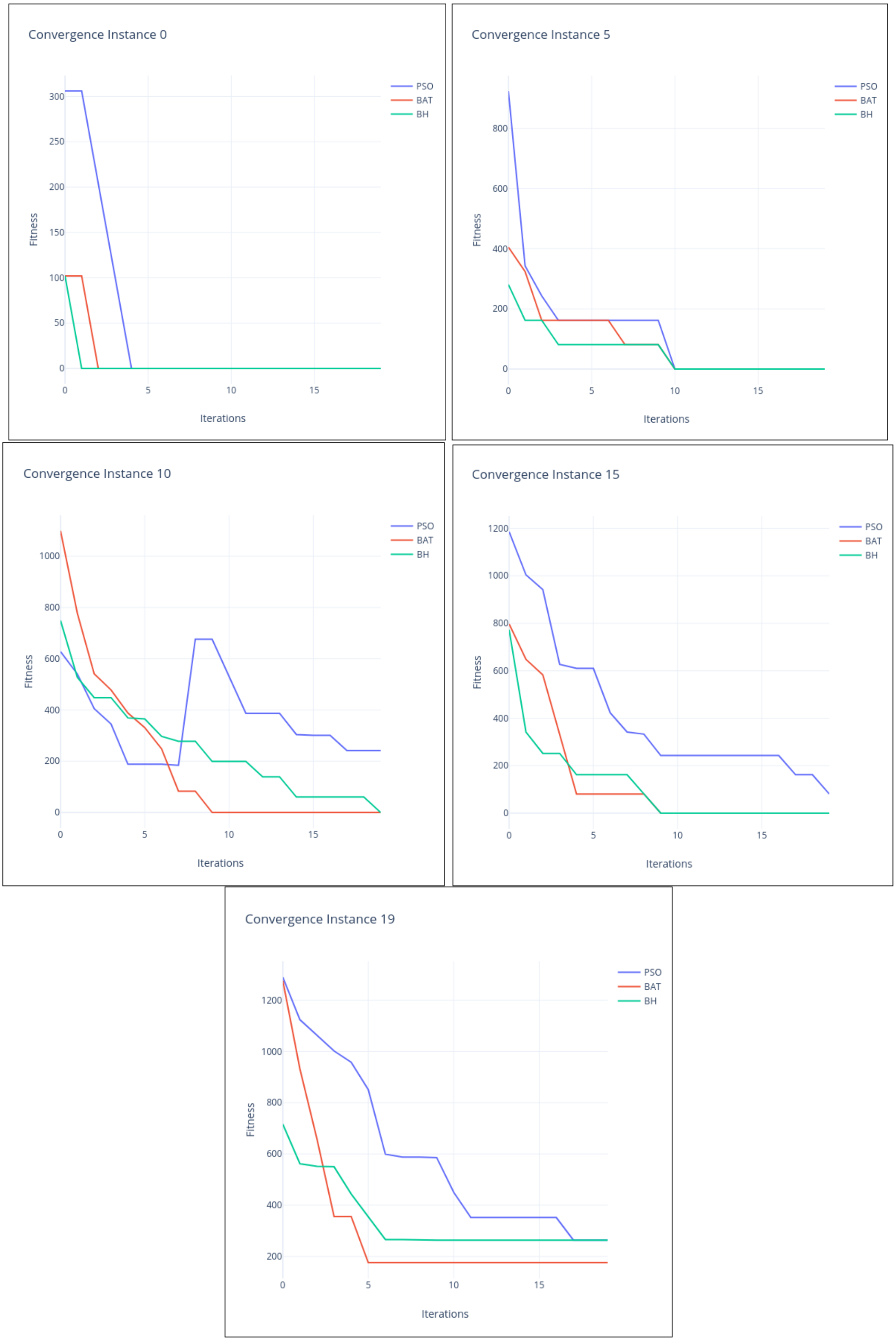
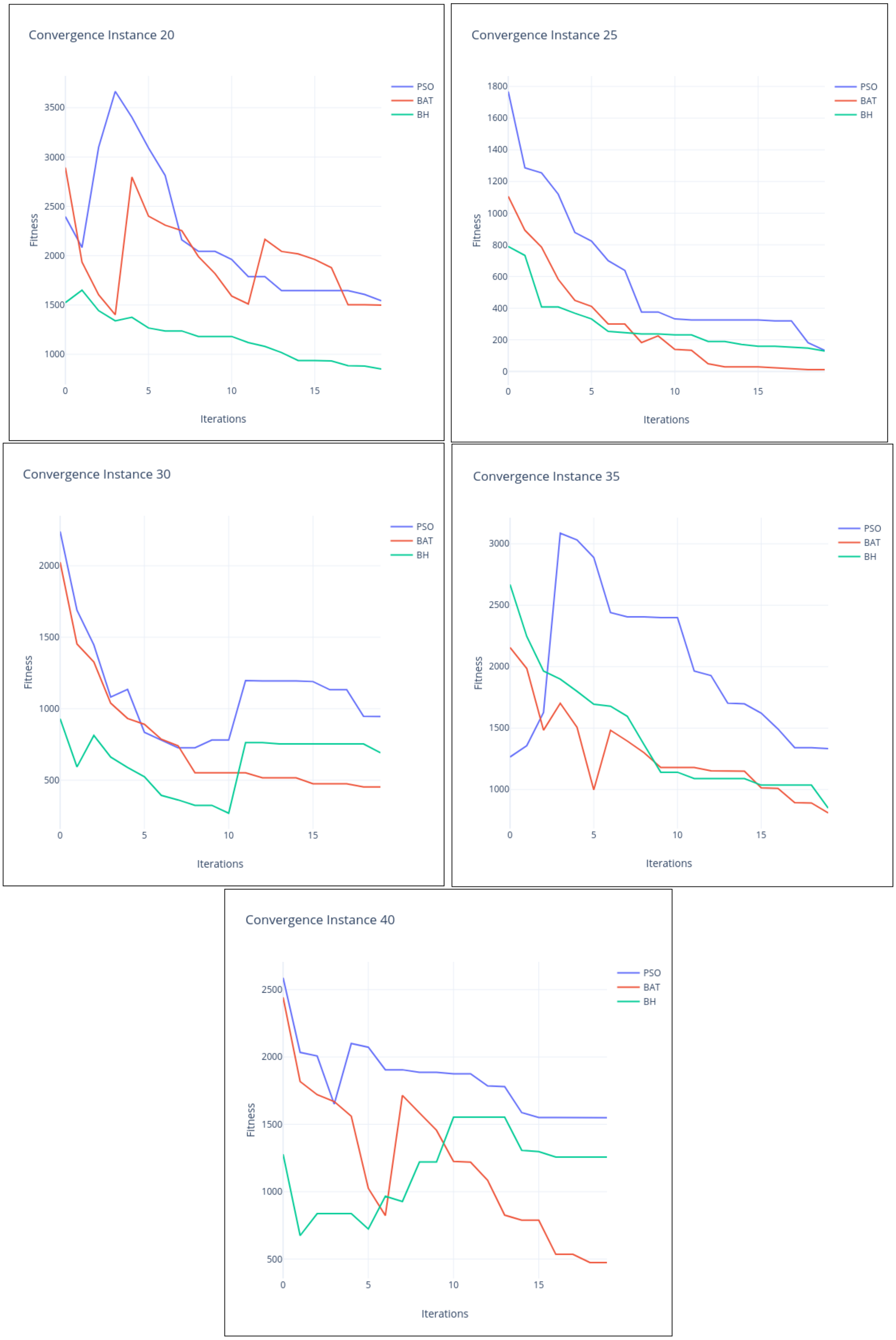


FIGURE 10. Convergence of diff amount the population solutions (1/3).



Convergence Instance 20

Convergence Instance 25

Convergence Instance 30

Convergence Instance 35

Convergence Instance 40

FIGURE 11. Convergence of diff amount the population solutions (2/3).

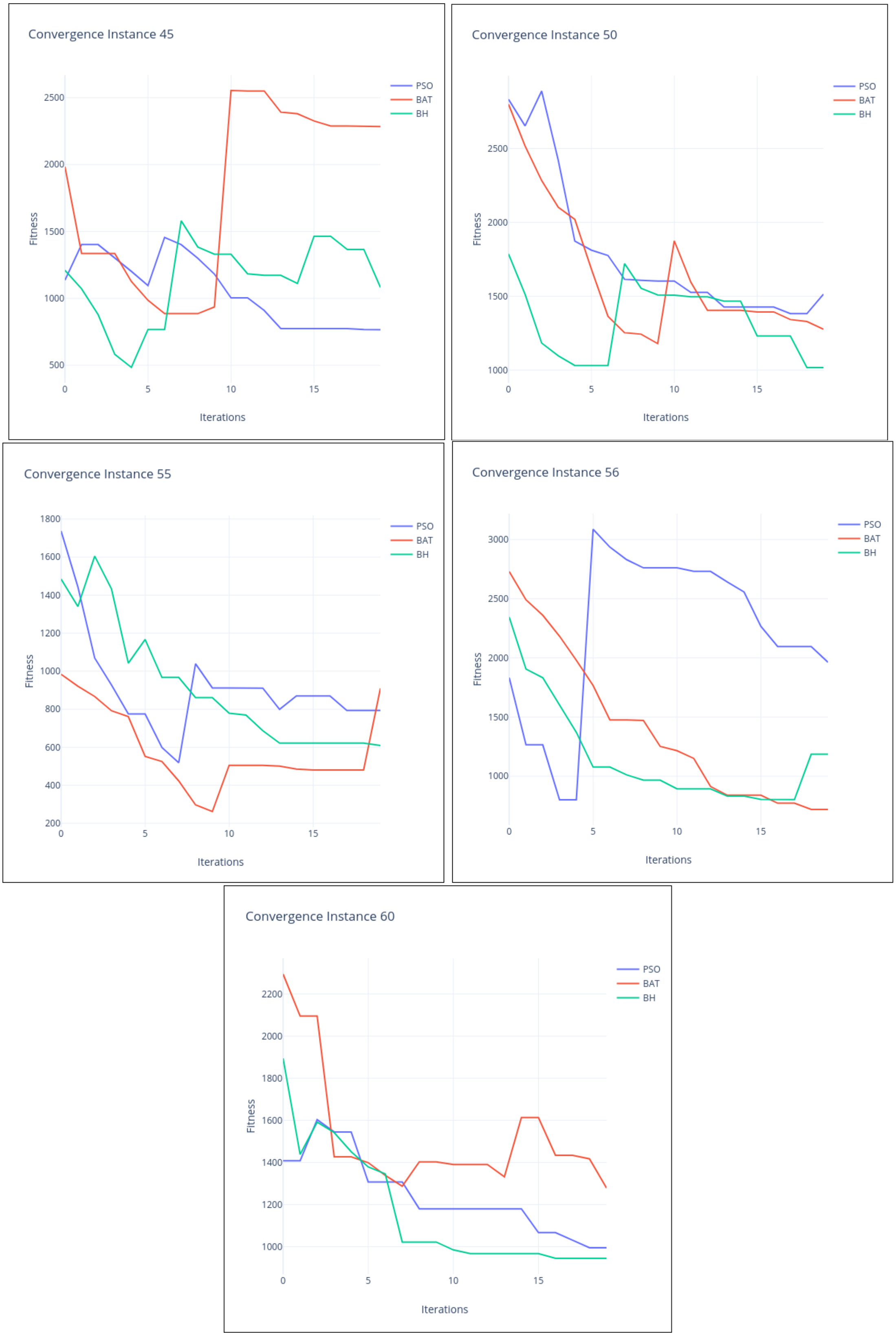


FIGURE 12. **Convergence of diff amount the population solutions (3/3).**

exhibit similar behavior, consistently achieving the same results within a comparable timeframe. Unlike the rest of the instances, the results are usually different but within the same range.

A notable finding of the experimental phase is the robustness demonstrated by the consistent solutions obtained across multiple executions of each instance. This is evident from the results depicted in Figures 7, 8, and 9, which showcase the solution quality achieved by each algorithm. It is observed that the BH algorithm outperforms both PSO and BAT in terms of solution quality, as it consistently attains the minimum value, representing the best optimal solution. Furthermore, the standard deviations associated with BH are generally lower compared to the other algorithms. This finding indicates that BH produces more homogeneous solutions, as supported by the interquartile range and the observation of the box plots. 10, 11, and 12 illustrate the convergence behavior of the PSO, BAT, and BH algorithms. It is observed that these algorithms exhibit similar convergence patterns for instances one to nineteen, with rapid convergence. However, for instances twenty to sixty, the convergence becomes slower. This can be attributed to the increased complexity of these instances, characterized by a higher number of subnets. Consequently, the algorithms require more time to find efficient solutions. Overall, these findings highlight the effectiveness of the BH algorithm in terms of solution quality and homogeneity, particularly in more challenging instances. Additionally, they emphasize the impact of instance complexity on convergence speed.

Finally, We also tested new metaheuristics recently reported but we did not get the expected results. All implementations can be downloaded from [65].

VII. CONCLUSION

This research presents a cyber risk management approach that incorporates the optimal distribution of NIDS (Network Intrusion Detection System) intrusion detection sensors, subordinated to a security information and event management (SIEM) tool, which allows continuous monitoring. of events associated with cyber risks. Furthermore, the approach emphasizes the importance of a containment strategy to mitigate cyber risk attacks, ensuring that organizations can sustain their operations and maintain service availability.

Also, development of a strategy for the management of cyber risks, which mathematically model with linear programming the requirements of said implementation according to the functional characteristics of the network. Our proposal covers some of the most relevant topics to consider when locating security sensors, such as costs and uptime. Here, we must prioritize and not include, for example, the number of end-points for subnetworks, wireless networking, virtual networking, and among others. These topics can be included in future works. For experiments, sixty network instances were created, ranging in complexity from networks of five subnets to thirty-nine subnets. It was possible to determine the efficient number of sensors that allows maximizing the benefits of its function, minimizing the indirect costs of not having it, and minimizing the costs of the number of NIDS.

Many works deal with multicriteria optimization from cost and investment perspectives. Our proposal covers not only direct costs, but also indirect costs related to network performance and the benefits of identifying the best location to increase sensor performance. In this context, the efficient quantity and where to place the NIDS for the centralized SIEM tool was approached as a multi-objective problem mathematically modeling through linear programming the technical and functional characteristics of the network and solved with bio-inspired algorithms such as PSO, BAT, and Black Hole. Subsequently, once the results were obtained by applying the three bio-inspired algorithms, it can be confirmed that the Black Hole algorithm achieved the best efficient solutions. The above given that PSO is deterministic since the solution it generates is not better than the previous one, so it does not change it. For its part, BAT is semi-deterministic since it changes a solution but with a certain probability. If that probability is not exceeded, it does not change. Therefore, it is not deterministic in the face of a better chance. Black Hole is non-deterministic. If it finds a better solution, it changes it immediately. It should be noted that Black Hole will not always provide better solutions. It will depend on the problem.

The scalability of the research proposal is addressed in the experimentation phase, working with sixty test instances ranging from small networks of five to fifteen subnets, through medium networks of fifteen to twenty-five subnets, up to large networks of twenty-five to thirty-nine. subnets. Therefore, the methodology can efficiently scale as network size, complexity, and requirements evolve. In addition to different network sizes, different scenarios include factors such as latency, throughput, and response times, all of which influence the probability of a network outage.

It should be also noted that the limitations of the proposed method are manifested from instance fifty-one when the number of subnets begins to grow. Thus, after thirty-one subnets, the corresponding bio-inspired algorithms take a long time to deliver their respective results, and only up to sixty with thirty-nine subnets is it possible to obtain the expected results. To solve these problem limitations, it is proposed to increase the computing capacity by processing hard instances with cloud computing.

For future work, we propose to develop a methodology that allows solving more complex instances, that is, more than thirty-nine subnets. Furthermore, we plan to use knowledge extraction intelligence mechanisms for boosting the search procedures of bio-inspired algorithms in order to they can find efficient solutions in a large-scale instances of this problem. Finally, we also try to solve this problem using machine learning techniques on generated data by bio-solvers. Here, we employ data from smaller instances to treat the hardest.

ABBREVIATIONS

NIDS: Network Intrusion Detection Sensors; SIEM: Security Information and Event Management System; PSO: Particle swarm optimization; BHO: Black Hole Optimizer; BAT: Bat Optimization Algorithm;

AVAILABILITY OF DATA AND MATERIAL

Please contact the authors for data requests.

REFERENCES

[1] D. C. Marinescu, *Cloud Computing: Theory and Practice*. San Mateo, CA, USA: Morgan Kaufmann, 2022.

[2] S. Kim, M. Tentzeris, and A. Georgiadis, "Hybrid printed energy harvesting technology for self-sustainable autonomous sensor application," *Sensors*, vol. 19, no. 3, p. 728, Feb. 2019, doi: 10.3390/s19030728.

[3] R. K. Rainer and B. Prince, *Introduction to Information Systems*. Hoboken, NJ, USA: Wiley, 2021.

[4] A. Yeboah-Ofori and S. Islam, "Cyber security threat modeling for supply chain organizational environments," *Future Internet*, vol. 11, no. 3, p. 63, Mar. 2019.

[5] F. Hoppe, N. Gatzert, and P. Gruner, "Cyber risk management in SMEs: Insights from industry surveys," *J. Risk Finance*, vol. 22, nos. 3–4, pp. 240–260, Nov. 2021.

[6] A. A. Ganin, P. Quach, M. Panwar, Z. A. Collier, J. M. Keisler, D. Marchese, and I. Linkov, "Multicriteria decision framework for cybersecurity risk assessment and management," *Risk Anal.*, vol. 40, no. 1, pp. 183–199, Jan. 2020.

[7] D. W. Hubbard, *The Failure of Risk Management: Why It'S Broken and How to Fix It*. Hoboken, NJ, USA: Wiley, 2020.

[8] M. Ahsan, K. E. Nygard, R. Gomes, M. M. Chowdhury, N. Rifat, and J. F. Connolly, "Cybersecurity threats and their mitigation approaches using machine learning—A review," *J. Cybersecurity Privacy*, vol. 2, no. 3, pp. 527–555, Jul. 2022.

[9] S. Mansfield-Devine, "Cyber security breaches survey 2022," Tech. Rep., 2022.

[10] M. Tvaronavičienė, T. Plėta, S. D. Casa, and J. Latvys, "Cyber security management of critical energy infrastructure in national cybersecurity strategies: Cases of USA, UK, France, Estonia and Lithuania," *Insights into Regional Develop.*, vol. 2, no. 4, pp. 802–813, Dec. 2020.

[11] G. Kavallieratos, G. Spathoulas, and S. Katsikas, "Cyber risk propagation and optimal selection of cybersecurity controls for complex cyberphysical systems," *Sensors*, vol. 21, no. 5, p. 1691, Mar. 2021.

[12] M. Khouzani, Z. Liu, and P. Malacaria, "Scalable min-max multi-objective cyber-security optimisation over probabilistic attack graphs," *Eur. J. Oper. Res.*, vol. 278, no. 3, pp. 894–903, Nov. 2019, doi: 10.1016/j.ejor.2019.04.035.

[13] S. Sarkar, "Detecting vulnerabilities of web application using penetration testing and prevent using threat modeling," in *Advances in Electronics, Communication and Computing*. Cham, Switzerland: Springer, 2021, pp. 21–32.

[14] M. Shukla, S. Sarmah, and M. K. Tiwari, "A multi-objective framework for the identification and optimisation of factors affecting cybersecurity in the industry 4.0 supply chain," *Int. J. Prod. Res.*, vol. 61, pp. 1–16, Jul. 2022, doi: 10.1080/00207543.2022.2100840.

[15] M. Hosseini Shirvani, "Bi-objective web service composition problem in multi-cloud environment: A bi-objective time-varying particle swarm optimisation algorithm," *J. Experim. Theor. Artif. Intell.*, vol. 33, no. 2, pp. 179–202, Feb. 2020, doi: 10.1080/0952813x.2020.1725652.

[16] A. Fausto, G. B. Gaggero, F. Patrone, P. Girdinio, and M. Marchese, "Toward the integration of cyber and physical security monitoring systems for critical infrastructures," *Sensors*, vol. 21, no. 21, p. 6970, Oct. 2021, doi: 10.3390/s21216970.

[17] L. Rikhtechi, V. Rafe, and A. Rezakhani, "Secured access control in security information and event management systems," *J. Inf. Syst. Telecommun.*, vol. 9, no. 1, p. 33, 2021.

[18] H. M. J. Almohri, L. T. Watson, D. Yao, and X. Ou, "Security optimization of dynamic networks with probabilistic graph modeling and linear programming," *IEEE Trans. Dependable Secure Comput.*, vol. 13, no. 4, pp. 474–487, Jul. 2016, doi: 10.1109/tdsc.2015.2411264.

[19] M. Khouzani, P. Malacaria, C. Hankin, A. Fielder, and F. Smeraldi, "Efficient numerical frameworks for multi-objective cyber security planning," in *Proc. Eur. Symp. Res. Comput. Secur.* Cham, Switzerland: Springer, 2016, pp. 179–197.

[20] A. Schilling and B. Werners, "Optimal selection of IT security safeguards from an existing knowledge base," *Eur. J. Oper. Res.*, vol. 248, no. 1, pp. 318–327, Jan. 2016.

[21] A. Schilling, "A framework for secure IT operations in an uncertain and changing environment," *Comput. Operations Res.*, vol. 85, pp. 139–153, Sep. 2017.

[22] H. A. Alterazi, P. R. Kshirsagar, H. Manoharan, S. Selvarajan, N. Alhebaishi, G. Srivastava, and J. C.-W. Lin, "Prevention of cyber security with the Internet of Things using particle swarm optimization," *Sensors*, vol. 22, no. 16, p. 6117, Aug. 2022.

[23] T. Llanso, M. McNeil, and C. Noteboom, "Multi-criteria selection of capability-based cybersecurity solutions," in *Proc. Annu. Hawaii Int. Conf. Syst. Sci.*, 2019, pp. 1–9, doi: 10.24251/hicss.2019.879.

[24] A. Salas-Fernández, B. Crawford, R. Soto, and S. Misra, "Metaheuristic techniques in attack and defense strategies for cybersecurity: A systematic review," in *Artificial Intelligence for Cyber Security: Methods, Issues and Possible Horizons or Opportunities*, 2021, pp. 449–467.

[25] G. Rajeshkumar, M. V. Kumar, K. S. Kumar, S. Bhatia, A. Mashat, and P. Dadheech, "An improved multi-objective particle swarm optimization routing on MANET," *Comput. Syst. Sci. Eng.*, vol. 44, no. 2, pp. 1187–1200, 2023.

[26] K. Zheng and L. A. Albert, "A robust approach for mitigating risks in cyber supply chains," *Risk Anal.*, vol. 39, no. 9, pp. 2076–2092, Sep. 2019.

[27] K. Zheng and L. A. Albert, "Interdiction models for delaying adversarial attacks against critical information technology infrastructure," *Nav. Res. Logistics (NRL)*, vol. 66, no. 5, pp. 411–429, Jun. 2019.

[28] G. González-Granadillo, S. González-Zarzosa, and R. Diaz, "Security information and event management (SIEM): Analysis, trends, and usage in critical infrastructures," *Sensors*, vol. 21, no. 14, p. 4759, Jul. 2021.

[29] A. R. Muhammad, P. Sukarno, and A. A. Wardana, "Integrated security information and event management (SIEM) with intrusion detection system (IDS) for live analysis based on machine learning," *Proc. Comput. Sci.*, vol. 217, pp. 1406–1415, 2023.

[30] M. Mahmood and B. Al-Khateeb, "Review of neural networks and particle swarm optimization contribution in intrusion detection," *Periodicals Eng. Natural Sci.*, vol. 7, no. 3, pp. 1067–1073, 2019.

[31] V. Vasilyev and R. Shamsutdinov, "Security analysis of wireless sensor networks using SIEM and multi-agent approach," in *Proc. Global Smart Ind. Conf. (GloSIC)*, Nov. 2020, pp. 291–296.

[32] J. Noriega-Linares and J. N. Ruiz, "On the application of the raspberry Pi as an advanced acoustic sensor network for noise monitoring," *Electronics*, vol. 5, no. 4, p. 74, Oct. 2016.

[33] Y. N. Soe, Y. Feng, P. I. Santosa, R. Hartanto, and K. Sakurai, "Towards a lightweight detection system for cyber attacks in the IoT environment using corresponding features," *Electronics*, vol. 9, no. 1, p. 144, Jan. 2020.

[34] Z. Xiao, X. Xu, H. Xing, S. Luo, P. Dai, and D. Zhan, "RTFN: A robust temporal feature network for time series classification," *Inf. Sci.*, vol. 571, pp. 65–86, Sep. 2021, doi: 10.1016/j.ins.2021.04.053.

[35] Z. Xiao, H. Zhang, H. Tong, and X. Xu, "An efficient temporal network with dual self-distillation for electroencephalography signal classification," in *Proc. IEEE Int. Conf. Bioinf. Biomed. (BIBM)*, Dec. 2022, pp. 1759–1762, doi: 10.1109/bibm55620.2022.9995049.

[36] H. Xing, Z. Xiao, D. Zhan, S. Luo, P. Dai, and K. Li, "SelfMatch: Robust semisupervised time-series classification with self-distillation," *Int. J. Intell. Syst.*, vol. 37, no. 11, pp. 8583–8610, Jul. 2022, doi: 10.1002/int.22957.

[37] N. Q. K. Le, T.-T. Huynh, E. K. Y. Yapp, and H.-Y. Yeh, "Identification of clathrin proteins by incorporating hyperparameter optimization in deep learning and PSSM profiles," *Comput. Methods Programs Biomed.*, vol. 177, pp. 81–88, Aug. 2019, doi: 10.1016/j.cmpb.2019.05.016.

[38] N. Q. K. Le, D. T. Do, T.-T.-D. Nguyen, and Q. A. Le, "A sequence-based prediction of Kruppel-like factors proteins using XGBoost and optimized features," *Gene*, vol. 787, Jun. 2021, Art. no. 145643, doi: 10.1016/j.gene.2021.145643.

[39] Y. Li, S.-M. Ghoreishi, and A. Issakhov, "Improving the accuracy of network intrusion detection system in medical IoT systems through butterfly optimization algorithm," *Wireless Pers. Commun.*, vol. 126, no. 3, pp. 1999–2017, Aug. 2021, doi: 10.1007/s11277-021-08756-x.

[40] R. Duo, X. Nie, N. Yang, C. Yue, and Y. Wang, "Anomaly detection and attack classification for train real-time Ethernet," *IEEE Access*, vol. 9, pp. 22528–22541, 2021, doi: 10.1109/access.2021.3055209.

[41] S. K. Smmarwar, G. P. Gupta, S. Kumar, and P. Kumar, "An optimized and efficient Android malware detection framework for future sustainable computing," *Sustain. Energy Technol. Assessments*, vol. 54, Dec. 2022, Art. no. 102852, doi: 10.1016/j.seta.2022.102852.

[42] W. Xia, R. Neware, S. D. Kumar, D. A. Karras, and A. Rizwan, "An optimization technique for intrusion detection of industrial control network vulnerabilities based on BP neural network," *Int. J. Syst. Assurance Eng. Manage.*, vol. 13, no. S1, pp. 576–582, Jan. 2022, doi: 10.1007/s13198-021-01541-w.

[43] S. Tufail, I. Parvez, S. Batool, and A. Sarwat, "A survey on cybersecurity challenges, detection, and mitigation techniques for the smart grid," *Energies*, vol. 14, no. 18, p. 5894, Sep. 2021.

[44] N. Thapa, Z. Liu, A. Shaver, A. Esterline, B. Gokaraju, and K. Roy, "Secure cyber defense: An analysis of network intrusion-based dataset CCD-IDSv1 with machine learning and deep learning models," *Electronics*, vol. 10, no. 15, p. 1747, Jul. 2021.

[45] K.-O. Detken, T. Rix, C. Kleiner, B. Hellmann, and L. Renners, "SIEM approach for a higher level of IT security in enterprise networks," in *Proc. IEEE 8th Int. Conf. Intell. Data Acquisition Adv. Comput. Syst., Technol. Appl. (IDAACS)*, vol. 1, Sep. 2015, pp. 322–327.

[46] V. Casola, A. De Benedictis, A. Riccio, D. Rivera, W. Mallouli, and E. M. de Oca, "A security monitoring system for Internet of Things," *Internet Things*, vol. 7, Sep. 2019, Art. no. 100080.

[47] B. B. Gupta and M. Quamara, "An overview of Internet of Things (IoT): Architectural aspects, challenges, and protocols," *Concurrency Comput., Pract. Exper.*, vol. 32, no. 21, p. e4946, Nov. 2020.

[48] P. Mukherjee and A. Mukherjee, "Advanced processing techniques and secure architecture for sensor networks in ubiquitous healthcare systems," in *Sensors for Health Monitoring*. Amsterdam, The Netherlands: Elsevier, 2019, pp. 3–29.

[49] L. Al Sardy, T. Tang, M. Spisländer, and F. Saglietti, "Analysis of potential code vulnerabilities involving overlapping instructions," in *Proc. Int. Conf. Comput. Saf., Rel., Secur.* Cham, Switzerland: Springer, 2017, pp. 103–113.

[50] H. Almarabeh and A. Sulieman, "The impact of cyber threats on social networking sites," *Int. J. Adv. Res. Comput. Sci.*, vol. 10, no. 2, pp. 1–9, Apr. 2019.

[51] Q.-V. Pham, S. Mirjalili, N. Kumar, M. Alazab, and W.-J. Hwang, "Whale optimization algorithm with applications to resource allocation in wireless networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4285–4297, Apr. 2020.

[52] K. Zenitani, "A multi-objective cost–benefit optimization algorithm for network hardening," *Int. J. Inf. Secur.*, vol. 21, pp. 813–832, Mar. 2022.

[53] K. Massey, N. Moazen, and T. Halabi, "Optimizing the allocation of secure fog resources based on QoS requirements," in *Proc. 8th IEEE Int. Conf. Cyber Secur. Cloud Comput. (CSCloud)/ 7th IEEE Int. Conf. Edge Comput. Scalable Cloud (EdgeCom)*, Jun. 2021, pp. 143–148.

[54] F. Enayaty-Ahangar, L. A. Albert, and E. DuBois, "A survey of optimization models and methods for cyberinfrastructure security," *IISE Trans.*, vol. 53, no. 2, pp. 182–198, Feb. 2021.

[55] A. Skendžic, B. Kovacic, and B. Balon, "Management and monitoring security events in a business organization–SIEM system," in *Proc. 45th Jubilee Int. Conv. Inf., Commun. Electron. Technol. (MIPRO)*, May 2022, pp. 1203–1208.

[56] R. Fauzi, S. H. Supangkat, and M. Lubis, "The PDCA cycle of ISO/IEC, 27005: 2008 maturity assessment framework," in *Proc. Int. Conf. User Sci. Eng.* Cham, Switzerland: Springer, 2018, pp. 336–348.

[57] A. Corallo, M. Lazoi, and M. Lezzi, "Cybersecurity in the context of industry 4.0: A structured classification of critical assets and business impacts," *Comput. Ind.*, vol. 114, Jan. 2020, Art. no. 103165.

[58] S. J. Nanda and G. Panda, "A survey on nature inspired metaheuristic algorithms for partitional clustering," *Swarm Evol. Comput.*, vol. 16, pp. 1–18, Jun. 2014, doi: 10.1016/j.swevo.2013.11.003.

[59] J. Krause, J. Cordeiro, R. S. Parpinelli, and H. S. Lopes, "A survey of swarm algorithms applied to discrete optimization problems," in *Swarm Intelligence and Bio-Inspired Computation*. Amsterdam, The Netherlands: Elsevier, 2013, pp. 169–191, doi: 10.1016/b978-0-12-405163-8.00007-7.

[60] M. Mavrovouniotis, C. Li, and S. Yang, "A survey of swarm intelligence for dynamic optimization: Algorithms and applications," *Swarm Evol. Comput.*, vol. 33, pp. 1–17, Apr. 2017, doi: 10.1016/j.swevo.2016.12.005.

[61] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. Int. Conf. Neural Netw. (ICNN)*, 2002, pp. 1–10, doi: 10.1109/icnn.1995.488968.

[62] A. Hatamlou, "Black hole: A new heuristic optimization approach for data clustering," *Inf. Sci.*, vol. 222, pp. 175–184, Feb. 2013, doi: 10.1016/j.ins.2012.08.023.

[63] X.-S. Yang, "A new metaheuristic bat-inspired algorithm," in *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*. Berlin, Germany: Springer, 2010, pp. 65–74, doi: 10.1007/978-3-642-12538-6_6.

[64] B. Crawford, R. Soto, G. Astorga, J. García, C. Castro, and F. Paredes, "Putting continuous metaheuristics to work in binary search spaces," *Complexity*, vol. 2017, pp. 1–19, 2017, doi: 10.1155/2017/8404231.

[65] R. Olivares. (2023). *MOMHs-Codes for NIDS Projects*. [Online]. Available: https://figshare.com/articles/software/MOMHs-Codes_for_NIDS_projects/23535609

OMAR SALINAS is currently pursuing the Ph.D. degree in computer sciences with Universidad de Valparaíso. His research interests include optimization, machine learning, networking, and cybersecurity.

RICARDO SOTO received the Ph.D. degree in computer science from the University of Nantes, France, in 2009. He is currently a Full Professor and the Head of the Computer Science Department, Pontifical Catholic University of Valparaso, Chile. His research interests include metaheuristics, global optimization, and autonomous search. In this context, he has published more than 200 scientific papers in different international conferences and journals in computer science, operational research, and artificial intelligence. Most of these papers are based on resolving real-world and academic optimization problems related to industry, manufacturing, rostering, and seaports.

BRODERICK CRAWFORD received the Ph.D. degree in computer science from Universidad Técnica Federico Santa María, Valparaso, Chile, in 2011. He is currently a Full Professor with the Computer Science Department, Pontifical Catholic University of Valparaso, Chile. His research interests include combinatorial optimization, metaheuristics, global optimization, and autonomous search. In this context, he has published about more than 300 scientific papers in different international conferences and journals in computer science, operational research, and artificial intelligence. Most of these papers are based on the resolution of benchmark and real-world optimization problems.

RODRIGO OLIVARES received the Ph.D. degree in computer science from Pontificia Universidad Católica de Valparíso, Chile, in 2019. He is currently an Assistant Professor and the Head of the Pregraduate Program of Informatics Engineering with Universidad de Valparaíso. His research interests include reactive and self-adaptive metaheuristics, global optimization, and machine learning. He has contributed to relevant scientific journals and prestigious conferences about optimization, artificial intelligence, and swarm intelligence algorithms. Most of these works are based on treating real-world and academic optimization problems related to the industry, health care, education, and agile practices.