# MS-NHHO: A Swarm Intelligence Optimization Algorithm Incorporating Cognitive Science for Malicious Traffic Detection

**Ziang Li (liziang@iie.ac.cn)**
Institute of Information Engineering, Chinese Academy of Sciences
School of Cyber Security, University of Chinese Academy of Sciences
Beijing, China

**Zhou Zhou (zhouzhou@iie.ac.cn)**
Institute of Information Engineering,
Chinese Academy of Sciences
Beijing, China

**Chengxiang Si✉ (sichengxiang@cert.org.cn)**
National Computer Network Emergency Response Technical Team/ Coordination Center of China (CNCERT/CC)
Beijing China

**Qingyun Liu (liuqingyun@iie.ac.cn)**
Institute of Information Engineering,
Chinese Academy of Sciences
Beijing, China

## Abstract

The diversification of attacks jeopardizes cyberspace's normal operation. This paper proposes a new Harris Hawks Optimization Based on Multiple Strategies (MS-NHHO), inspired by humans' limited cognitive load, collective decision-making, and dynamic learning mechanisms for processing complex information. This paper utilizes the elite chaos reverse learning strategy to improve the algorithm's convergence speed and population diversity. Then, the dynamic adaptive weights are introduced into the escape energy decline mechanism to improve the algorithm's global exploration and local exploitation ability. Finally, the Gaussian random walk strategy enhances the algorithm's anti-stagnation ability. The experimental results confirm the usefulness of the three optimization strategies. Meanwhile, MS-NHHO exhibits satisfactory performance in terms of computational cost, detection performance, and efficiency in several scenarios.

**Keywords:** cognitive science; malicious traffic detection; feature selection; swarm intelligence optimization algorithm

## Introduction

Phishing attacks, denial-of-service attacks, and other methods are increasingly frequent threats to the network environment, and traditional security protection measures appear to be incompetent in the face of complex and changing attacks. Swarm intelligence optimization algorithms have become a hotspot of extensive research in recent years because of their powerful global search capability, adaptability, and intelligent features. Enhancing the intelligence level of swarm intelligence algorithms so that they can better simulate how human beings cope with uncertainty, complexity, and variability in the decision-making process has become the key to improving the efficiency and accuracy of attack detection systems.

Cognitive science can expose the transmission and decision-making process of information, help to better simulate human coping strategies in the face of cyber threats, and improve the intelligence and flexibility of detection algorithms (Andrade & Yoo, 2019; Andrade, Fuertes, Cazares, Ortiz-Garcés, & Navas, 2022; Rodgers, Attah-Boakye, & Adams, 2020). Therefore, we introduce cognitive load theory, collective decision theory, and learning mechanisms, and propose MS-NHHO. The contributions are as follows:

(i) Cognitive load and information processing. Too much information when individuals process complex information can lead to decision-making errors. We introduce the cognitive load theory into MS-NHHO and use the elite chaotic inverse learning strategy to improve the population diversity and optimize the information processing to avoid information overload, to improve MS-NHHO's efficiency and accuracy.

(ii) Collective Decision Making and Information Sharing. Information exchange and collaboration among individuals are the key to achieving global optimization. By introducing collective decision theory, we can simulate how group members can work together to identify attacks through information fusion and avoid individuals falling into local optimal solutions when facing complex attack patterns.

(iii) Learning Mechanism and Adaptive Optimization. We simulate human learning when facing new environments and problems and introduce a Gaussian random walk strategy to improve the algorithm's anti-stagnation ability. This enables MS-NHHO to dynamically adjust and adapt to new attacks.

The combination of cognitive science theory and a swarm intelligence optimization algorithm can not only provide new ideas for network attack detection but also promote the development of network security protection technology in the direction of intelligence, adaptability, and dynamization. Meanwhile, the research results will also provide a new theoretical foundation and technical route for the design of future intelligent network security protection systems and further promote the deep integration of artificial intelligence and cognitive science in the field of network security.

## Related Work

### Malicious Traffic Detection and Cognitive Science

Static detection is accomplished through reverse engineering decompiled applications and extracting static features. W. Li, Ge, and Dai (2015) developed a model using sensitive permissions. Arp et al. (2014) combined static features to achieve better performance. However, these methods require human involvement and prior knowledge, and may not be timely.

Dynamic detection makes judgments based on the actual running results by simulating a scenario in which a user actually runs the program to be detected. Behiry and Aly (2024) adopted feature reduction techniques and machine learning algorithms to identify cyberattacks. Z. Li, Cheng, Zang, and Li (2023) proposed a two-layer model to achieve malicious traffic detection more accurately through a hierarchical feature learning method. Rustam, Aljedaani, Elsayed, and Jurcut (2024) proposed a security vulnerability attack detection system called FAMTDS by analyzing network devices and



In D. Barner, N.R. Bramley, A. Ruggeri and C.M. Walker (Eds.), *Proceedings of the 47th Annual Conference of the Cognitive Science Society* 

IP-based network traffic. Although dynamic detection methods are effective, they require deep system modifications or virtual environments such as sandboxes to accomplish monitoring. Meanwhile, the operation efficiency is low and cannot meet the demand for real-time detection of Internet traffic.

Cognitive science brings a new perspective to malicious traffic detection. Veksler et al. (2020) examined how two cognitive modeling approaches can be helpful to cybersecurity professionals, exploring the potential for automatically constructing and predicting attacker preferences in real-time attack scenarios. White (2023) explores how humans are affected by unconscious biases in cybersecurity decision-making and how insights from cognitive science can be used to mitigate these effects. Orun, Orun, and Kurugollu (2023) proposed a remote threat identification method based on cognitive psychology and artificial intelligence.

### Swarm Intelligence Optimization Algorithms

To better address feature selection and model parameter optimization, researchers have carried out some attempts using swarm intelligence algorithms.Jia, Li, and Sun (2022) considered the correlation between features and data, and proposed a feature selection method that fused genetic algorithm and sooty tern optimization algorithm, which greatly improved the performance of the feature subset. Xue and Shen (2020) designed a sparrow search algorithm (SSA), which possesses better convergence speed and stability. Aljarah et al. (2018) proposed an encapsulated feature selection method based on the locust optimization algorithm.

Although each of the above methods has its strengths, the excessive time and computational costs can affect their usability. Swarm Intelligence Optimization Algorithms, all suffer from the shortcomings of being prone to local optimum and the difficulty of developing balanced exploration, and their optimization capabilities also have the potential to be further improved. Based on this, this paper proposes MS-NHHO, which adopts multiple optimization strategies to achieve a certain degree of improvement in terms of computational cost, detection efficiency, and detection performance.

## MS-NHHO

To enhance efficiency and effectively identify large-scale malicious Internet traffic, we adopted XGBoost, whose complexity is impacted by the number of base classifiers and the tree's complexity. Thus, parameter optimization is needed.

Harris Hawks Optimization (HHO) (Heidari et al., 2019) exhibits a stronger ability for optimization and higher accuracy, with the potential to achieve superior detection performance by using fewer features, which can be used in traffic detection module to improve efficiency. However, HHO is also susceptible to falling into local optimum. Therefore, we propose a New Harris Hawks Optimization Based on Multiple Strategies (MS-NHHO) to achieve fast and accurate detection of traffic through multiple optimization strategies.

### Harris Hawks Optimization (HHO)

HHO simulates various hunting behaviors of Harris's hawks, which are mainly divided into exploration and exploitation.

**Exploration Phase.** When the absolute value of the prey's escape energy $|E| \geq 1$, HHO engages in global exploration using two strategies with equal probability and executes a population position update as per equation (1).

$$L_{i+1} = \begin{cases} [G_i - A_i] - p_3[B_l + p_4(B_u - B_l)], & \text{if } q < 0.5 \\ L_i^{rand} - p_1[L_i^{rand} - 2p_2 L_i], & \text{if } q \geq 0.5 \end{cases} \quad (1)$$

$L_i$ is the population position obtained after the $i$-th iteration. $L^{rand}$ refers to the position of the randomly selected individual in the population. $B_u$ and $B_l$ indicate the upper and lower boundaries of the search space, respectively. $G_i$ denotes the position of the prey, which is also the position of the current optimal individual. $p$ and $q$ are random numbers in the interval of $(0,1)$. $A_i$ represents the mean position of the population, which is calculated using equation (2).

$$A_i = \frac{1}{N}\sum_{j=1}^{N} L_i^j \quad (2)$$

$N$ is the total number of individuals, and $L_i^j$ is the location of each individual in the population after the $i$-th iteration.

**Phase Transition.** Maintaining an appropriate balance between global exploration and local exploitation is essential to improving the performance of swarm intelligent optimization algorithm. During the iterative search, HHO conducts global exploration or local exploitation based on the escape energy of the prey. The value of the prey's escape energy is updated after each iteration in accordance with equation (3).

$$E = 2E_0(1 - \frac{i}{I}) \quad (3)$$

Where, $E_0$ is the initial value of escape energy, which is a random number in the interval $(0,1)$. $I$ is the maximum number of iterations. If the absolute value of the escape energy is less than 1, the algorithm enters the exploitation phase; otherwise, it continues with global exploration.

**Exploitation Phase.** HHO chooses four strategies to emulate the attack behavior of the Harris's hawks. Selection is based on the probability of prey capture $C$ and escape energy value $E$, which enables completion of the position update.

(i) *Strategy 1: Soft surround.* When $|E| \geq 0.5$ and $C \geq 0.5$, the prey has sufficient energy, and Harris's hawks hunt with a soft surround strategy while continuously adjusting its position based on equations (4) to (6).

$$L_{i+1} = D_t - E|R \cdot G_i - L_i| \quad (4)$$

$$D_t = G_i - L_i \quad (5)$$

$$R = 2(1 - p_5) \tag{6}$$

$R$ is the distance at which the prey escaped, $D_t$ is the difference between the optimal individual and the current individual, and $p_5$ is a random number in the interval $(0,1)$.

(ii) *Strategy 2: Hard surround.* When $|E| < 0.5$ and $C \geq 0.5$, the prey is low on energy, and Harris's hawks hunt with a hard surround strategy while continuously adjusting its position based on equations (7).

$$L_{i+1} = G_i - E|D_t| \tag{7}$$

(iii) *Strategy 3: Soft surround with rapid dives.* When $|E| \geq 0.5$ and $C < 0.5$, the prey is highly energetic and evasive, making it difficult to catch. The population position is updated in accordance with equation (8). This strategy consists of two ways, and the second way is used when the first way does not improve the fitness value.

$$L_{i+1} = \begin{cases} Y, & \text{if } F(Y) < F(L_i) \\ Z, & \text{if } F(Z) < F(L_i) \end{cases} \tag{8}$$

$$Y = G_i - E|R \cdot G_i - L_i| \tag{9}$$

$$Z = Y + \boldsymbol{V} \cdot F_{Levy}(D_Q) \tag{10}$$

$F(\cdot)$ is the fitness function and $F_{Levy}(\cdot)$ represents the Levy flight. $D_Q$ indicates the dimension of the problem and $\boldsymbol{V}$ is a random vector of dimension $D_Q$.

(iv) *Strategy 4: Hard surround with rapid dives.* When $|E| < 0.5$ and $C < 0.5$, the prey is not easily captured, but its escape energy is also insufficient. In this case, HHO updates the population position according to equation (11).

$$L_{i+1} = \begin{cases} Y, & \text{if } F(Y) < F(L_i) \\ Z, & \text{if } F(Z) < F(L_i) \end{cases} \tag{11}$$

$$Y = G_i - E|R \cdot G_i - A_i| \tag{12}$$

$$Z = Y + \boldsymbol{V} \cdot F_{Levy}(D_Q) \tag{13}$$

## New Harris Hawks Optimization Based on Multiple Strategies (MS-NHHO)

During HHO algorithm iterations, the population solely utilizes information from optimal individuals leading to reduced diversity, and ultimately, HHO falls into premature convergence. Additionally, the linear variation of the escape energy can result in an imbalanced exploration phase versus the exploitation phase of the algorithm. Therefore, we adopt the elite chaos reverse learning strategy to enhance both population diversity and the number of elite groups in the population. In addition, dynamic adaptive weights are integrated into the escape energy iteration mechanism to balance the algorithm's global exploration and local exploitation ability. Finally, we utilize the Gaussian random walk strategy to improve the algorithm's anti-stagnation capability. The flow of the MS-NHHO is shown in figure 1.

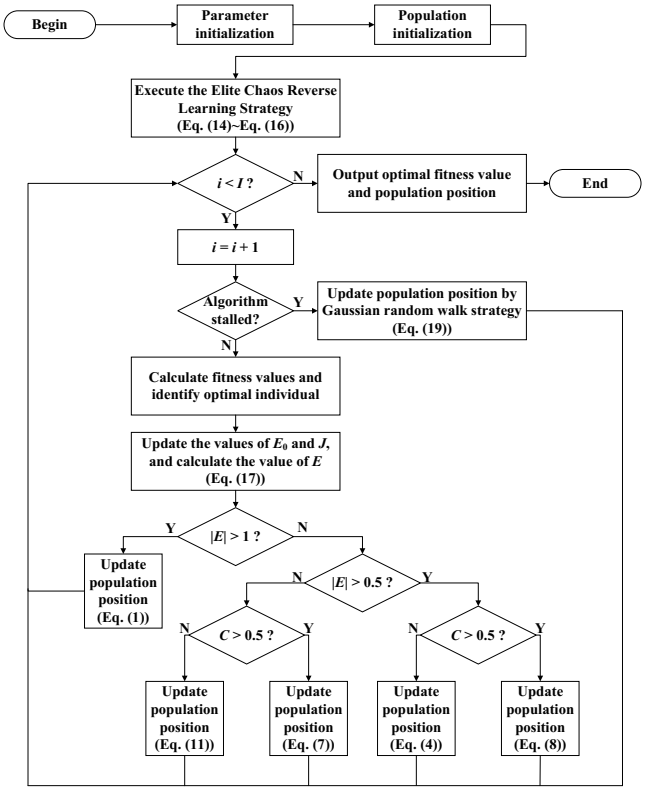


Figure 1: The flow chart of the MS-NHHO.

**Elite Chaos Reverse Learning Strategy.** Sufficient population diversity ensures that individuals are widely distributed in the solution space, allowing the algorithm to maintain its ability to explore better solutions and avoid premature convergence. Second, a diverse population can cover a larger solution space, which gives the algorithm a better chance of discovering a globally optimal solution. Finally, the population's diversity helps the algorithm strike a balance between the exploration and exploitation phases.

The elite group represents the current optimal solutions, and by increasing the number of elite individuals, the algorithm can focus more on the fine-tuning of these high-quality solutions, and thus move closer to the global optimal solution more effectively. In addition, in the iterative process of the algorithm, randomness factors may lead to the replacement or loss of quality solutions, and retaining more elite individuals can effectively reduce this risk.

Therefore, for the swarm intelligence optimization algorithms, increasing the population diversity and the number of elite groups in the population can enhance the algorithm's exploration ability and exploitation ability.

Chaotic mapping produces sequences with random, ergodic properties, which are used to initialize the popula-

tion, resulting in a more uniform distribution of the population in the search space. Elite reverse learning, on the other hand, utilizes the information of the current elite population to guide other individuals to search for better solutions during the iteration process of the algorithm. Combining the two can realize the improvement of population diversity and the number of elite groups in the population, which in turn enhances the algorithm's searchability and convergence speed on complex problems.

Tent chaotic mapping is ergodic, where almost every point in the definition domain is mapped, which helps the algorithm to explore uniformly throughout the search space. In addition, its randomness helps in generating uncorrelated initial populations, thus increasing the diversity of the population. Therefore, Tent chaotic mapping is used to initialize the population to prevent the MS-NHHO from falling into the local optimum, as shown in equation (14).

$$S_{i+1} = \begin{cases} 2S_i + \frac{r}{N_t}, & \text{if } 0 \le S_i < \frac{1}{2} \\ 2(1-S_i) + \frac{r}{N_t}, & \text{if } \frac{1}{2} \le S_i < 1 \end{cases} \tag{14}$$

$N_t$ is the number of particles within the Tent sequence and $r$ is a random number between 0 and 1. The population individual $X_i$ can be obtained by inverse mapping the obtained chaotic sequence $S_i$, as demonstrated in equation (15).

$$X_i = S_i(B_u - B_l) + B_l \tag{15}$$

To broaden the scope of the search, we use the small-hole imaging reverse learning strategy to increase the diversity of alternative locations in MS-NHHO's optimization process as well as the number of elite individuals. It uses the principle of small-hole imaging to reflect the solutions of elite individuals to symmetric positions, using the center of the solution space as the axis of symmetry. Through such reverse generation, under-explored regions of the solution space can be explored and better solutions can be found faster, ultimately realizing an increase in the number of elite individuals in the population, as shown in equation (16).

$$A_d{}' = \frac{(u_d + l_d)}{2} + \frac{(u_d + l_d)}{2n} - \frac{A_d}{n} \tag{16}$$

$A_d$ is the current optimal solution and $A_d{}'$ denotes the reverse optimal solution obtained after processing by the small hole imaging reverse learning strategy, respectively. $u_d$ and $l_d$ stand for the upper and lower bounds of the solution in the $d$-th dimension, respectively. $n$ is the moderating factor.

**Nonlinear Escape Energy Update Strategy.** In complex optimization problems, purely local search may not be able to find a good enough solution, and some global search needs to be continued at a later stage to help cope with these complexities and uncertainties. Dynamic tuning of the search strategy can control the balance between exploration and exploitation by adjusting the parameters. Therefore, to address the shortcomings of HHO that only performs local searches in the later stages, we introduce dynamic adaptive weights into the escape energy update strategy.

In the early stages, the adaptive weights can be given higher values to enhance the initial value of the escape energy, making the algorithm more inclined to global exploration. As the number of iterations increases, the adaptive weights can be progressively reduced, thereby decreasing the effect of the escape energy. The algorithm gradually shifts from a global search to a search pattern that continues to explore to some extent and at the same time begins to exploit the potentially optimal regions that have been found. In the later stages of the algorithm, the adaptive weights can be further reduced or even approach zero. At this point, the escape energy is also reduced. In this way, the algorithm can take full advantage of the previously searched favorable regions to carefully optimize the current solution and further approximate the global optimal solution. The nonlinear escape energy update strategy is shown in equation (17). Where, $\lambda_0$ and $\lambda_f$ denote the initial and final values of the weights, respectively. $\delta$ is a random number within the interval of [0,1].

$$E = 2E_0\lambda(1 - \frac{i}{I}) \tag{17}$$

$$\lambda = \delta[\lambda_0 - (\lambda_0 - \lambda_f) \times \frac{1}{e-1} \times (e^{\frac{i}{I}} - 1)] \tag{18}$$

**Gaussian Random Walk Strategy.** The individuals in the population stop moving or fail to update effectively in the search space can cause the algorithm to slow down its convergence or even fail to find the global optimal solution. Therefore, improving the anti-stagnation ability of the algorithm can enable the population to quickly adjust its search strategy when the environment changes, ensuring that the algorithm can continue to explore and exploit the search space and enhance its robustness.

In general, the average value of the dominant population does not change during two consecutive iterations can be considered that the algorithm is falling into stagnation. The Gaussian random walk strategy makes the search process more random through the random step size generated by the Gaussian distribution, thus allowing the algorithm to randomly jump in the search space and break the stagnation. The strategy is shown in equation (19).

$$L_{i+1} = Gaussian(L_i, \sigma) \tag{19}$$

$$\sigma = \cos[\frac{2}{\pi} \times (\frac{i}{I})^2] \times (L_i - L_i{}^*) \tag{20}$$

$L_i{}^*$ is a randomly chosen individual from the dominant population. The Gaussian random walk's step size is adjusted using a cosine function so that a substantial perturbation during the early iteration and a quick reduction during the later part of the iteration.

**Time Complexity.** The time complexity of HHO and MS-NHHO primarily depends on the population initialization, fitness assessment, and population position update phases.

We assume that the number of individuals in the population is $N$, the dimension of the fitness function is $D_Q$, and the upper limit of the number of iterations is $I$, then the time complexity of HHO is shown in equation (21).

$$C_{HHO} = O[N \times (I + I \times D_Q + 1)] \tag{21}$$

In the population initialization phase of the MS-NHHO, we assume that the time for Tent chaotic mapping according to equation (14) is $T_1$, and the time for population elitist processing according to equation (16) is $T_2$, then the time complexity of the population initialization phase of the MS-NHHO can be expressed as shown in equation (22).

$$C_1 = O(N) + O[(T_1 + T_2) \times N \times D_Q] = O(N \times D_Q) \tag{22}$$

When conducting the fitness assessment by equation (17), i.e., the escape energy update phase, we assume that the time of each of its updates is $T_3$, and the time complexity of the fitness assessment phase is shown in equation (23).

$$C_2 = O(T_3 \times N \times I) = O(N \times I) \tag{23}$$

We assume that the time for Gaussian random walk strategy, as per equation (19), is $T_4$, $T_5$, $T_6$, and $T_7$ under the four hunting strategies. So, the time complexity of the population position update phase is shown in equation (24).

$$C_3 = O[(T_4 + T_5 + T_6 + T_7) \times N \times I \times D_Q] = O[N \times I \times D_Q] \tag{24}$$

In summary, the time complexity of the MS-NHHO is shown in equation (25).

$$C_{MS-NHHO} = C_1 + C_2 + C_3 = O[I + I \times D_Q + D_Q] \tag{25}$$

$D_Q$ denotes the dimension of the fitness function, which is a constant when the fitness function is determined. Therefore, we can see that the time complexity of the MS-NHHO and the HHO is essentially the same, and the computational cost of introducing multiple optimization strategies is essentially negligible relative to a computer.

**Fitness Function.** The fitness function measures the superiority or inferiority of individuals within the population and guides the optimization direction of the algorithm in subsequent stages. In this paper, we adopt the MS-NHHO to optimize the feature selection process and XGBoost's parameter optimization process and design the fitness function as in equation (26) by considering the number of selected features and the detection accuracy.

$$F = \alpha(\text{Acc} - 1) - \beta \cdot \frac{n}{N} \tag{26}$$

Both $\alpha$ and $\beta$ are adjustable parameters, and $n$ and $N$ are respectively the number of feature subsets selected by the algorithm and the total number of feature subsets. When the detection performance of XGBoost is better and the dimension of the feature subset is lower, the value of the fitness function is larger.

## The Experimental Results

### Experimental Configuration

Python3.7 is used to build the MS-NHHO, which run on Windows10 OS. The computer has an Intel (R) Core (TM) I7-6700HQ 2.60GHz CPU and 16GB memory. In order to increase the speed of the MS-NHHO, a NVIDIA GeForce GTX 960M GPU is used as the accelerator. We have set the problem dimension to be solved by MS-NHHO as $D_Q = 30$ and the prey escape distance interval as $R \in [0, 2]$. We have also set the total number of individuals in the population to $N = 50$ and the maximum number of iterations to $I = 300$.

### Dataset (CIC-IDS2017)

CIC-IDS2017 comprises benign traffic and common attack-generated traffic, published by the Canadian Institute for Cybersecurity in 2018. We select some of the benign and malicious traffic samples in PCAP format with a size of 7.8 GB.

### Ablation Experiments

Three important strategies are used in the optimization process of MS-NHHO, i.e., Chaotic Elite Reverse Learning Strategy (CERL), Nonlinear Escape Energy Updating Strategy (NEEU), and Gaussian Random Walk Strategy (GRW). We conducted experiments using a subset of samples from the CIC-IDS2017 dataset, and each set of experiments is performed five times to take the average value to validate the effectiveness of these three strategies. The performance of the algorithms in different scenarios is shown in Table 1.

Table 1: Results of ablation experiments of the MS-NHHO.

| | **Acc (%)** | **CPU usage (%)** | **Time (s)** |
|---|---|---|---|
| w/o CERL | 85.49 | 76 | 46 |
| w/o NEEU | 91.26 | 82 | 43 |
| w/o GRW | 93.11 | 81 | 83 |
| MS-NHHO | 98.73 | 48 | 24 |

The experimental results reflect that all three optimization strategies have a positive impact on the performance of MS-NHHO in the malicious traffic detection task.

The removal of the CERL resulted in a decrease in accuracy. This is because this strategy improves the population diversity, and the high diversity gives the algorithm a better chance to find the best combination among different feature combinations, thus selecting the subset of features that contributes the most to the performance.

The presence of the NEEU reduces the computational resource consumption. Effectively balancing the exploration

and exploitation phases allows the algorithm to avoid premature convergence and optimize deeply in the appropriate region, ultimately obtaining a better-quality feature subset.

The algorithm's performance is also further facilitated by the GRW, whose improved anti-stagnation ability can optimize the search path, enabling the algorithm to search effectively in complex high-dimensional feature spaces and improve the efficiency of feature selection.

Together, these strategies enable the MS-NHHO to select better features among a large number of samples, which achieves more accurate detection with fewer resources.

## Effect of the MS-NHHO on Detection Performance

During the detection of malicious traffic, selecting a high-quality subset of features can achieve the goal of fast and accurate detection. In this section, we propose the MS-NHHO for feature selection, parameter optimization and training.

We selected multiple sets of data from the CIC-IDS2017 dataset with different sample sizes, different number of features, and different number of categories in multiple times as the experimental dataset, as shown in Table 2.

Table 2: Details of the experimental dataset.

| Scenarios | Samples | Features | Categories | Proportion (Benign: Malicious) |
|---|---|---|---|---|
| CIC-1 | 2312 | 19 | 6 | 1: 1 |
| CIC-2 | 5000 | 42 | 4 | |
| CIC-3 | 27200 | 78 | 8 | |
| CIC-4 | 35714 | 94 | 11 | |

We chose HHO (Heidari et al., 2019), IHHO (Xu & Liu, 2021), BAS (Jiang & Li, 2018) and SSA (Xue & Shen, 2020) as a control group. Accuracy and number of feature subsets selected in the detection process are used as evaluation metrics. Each set of experiments is performed five times to take the average value, as shown in figure 2 and figure 3.

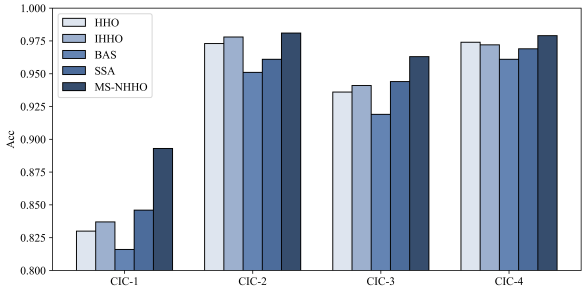


Figure 2: The accuracy of each algorithm.

Compared to the control group, MS-NHHO is able to achieve more accurate detection using fewer features. In these five sets of experiments, MS-NHHO only uses about 20% to 25% of these features for detection, significantly reducing the feature dimension, which also reflects the algorithm's advantage in feature selection. In addition, MS-NHHO also has the highest detection accuracy, which also indicates that its detection capability is very impressive.

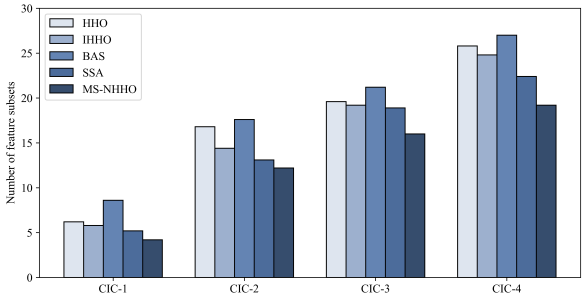


Figure 3: The number of features selected by each algorithm.

## Effect of the MS-NHHO on Costs and Efficiency

We designed multiple sets of experiments based on two experimental scenarios, CIC-1 and CIC-3, to verify the effect of the MS-NHHO on the computational cost and efficiency of the malicious traffic detection process. During the experiments, each group of experiments is conducted five times and averaged as the final results, as shown in Table 3.

Table 3: The results in terms of cost and efficiency.

| Methods | CIC-1 | | CIC-3 | |
|---|---|---|---|---|
| | CPU usage (%) | Time (s) | CPU usage (%) | Time (s) |
| HHO | 46 | 8 | 72 | 73 |
| IHHO | 31 | 5 | 62 | 54 |
| BAS | 57 | 11 | 78 | 112 |
| SSA | 39 | 10 | 67 | 85 |
| MS-NHHO | 17 | 2 | 51 | 26 |

MS-NHHO significantly decreases CPU usage and time costs. This is mainly attributed to CERL, NEEU, and GRW, which motivate MS-NHHO to select higher-quality feature subsets and thus have a lower computational cost.

## Conclusion

Aiming at the phenomena of high cost and low efficiency in the process of malicious traffic detection, this paper proposes the MS-NHHO with comprehensive consideration of feature selection and parameter optimization.

We use the CIC-IDS2017 dataset to carry out validation and testing from multiple dimensions. The results show that MS-NHHO is able to select a better-quality feature subset and thus reduce the feature dimensions compared to the existing methods. In addition, MS-NHHO possesses lower CPU usage and time cost. In summary, MS-NHHO effectively reduces the computational cost of the detection, while playing a very positive role in operational efficiency.

In future research, we will focus on fine-grained detection in few-shot scenarios to achieve higher-level detection goals.

## Acknowledgments

This work is supported by the Scaling Program of Institute of Information Engineering, CAS (Grant No. E3Z0041101).

## References


Aljarah, I., Al-Zoubi, A. M., Faris, H., Hassonah, M. A., Mirjalili, S., & Saadeh, H. (2018). Simultaneous feature selection and support vector machine optimization using the grasshopper optimization algorithm. *Cognitive Computation*, *10*, 478–495.

Andrade, R. O., Fuertes, W., Cazares, M., Ortiz-Garcés, I., & Navas, G. (2022). An exploratory study of cognitive sciences applied to cybersecurity. *Electronics*, *11*(11), 1692.

Andrade, R. O., & Yoo, S. G. (2019). Cognitive security: A comprehensive study of cognitive science in cybersecurity. *Journal of Information Security and Applications*, *48*, 102352.

Arp, D., Spreitzenbarth, M., Hubner, M., Gascon, H., Rieck, K., & Siemens, C. (2014). Drebin: Effective and explainable detection of android malware in your pocket. In *Ndss* (Vol. 14, pp. 23–26).

Behiry, M. H., & Aly, M. (2024). Cyberattack detection in wireless sensor networks using a hybrid feature reduction technique with ai and machine learning methods. *Journal of Big Data*, *11*(1), 16.

Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., & Chen, H. (2019). Harris hawks optimization: Algorithm and applications. *Future generation computer systems*, *97*, 849–872.

Jia, H., Li, Y., & Sun, K. (2022). Simultaneous feature selection optimization based on hybrid sooty tern optimization algorithm and genetic algorithm. *Acta Autom. Sin*, *48*, 1601–1615.

Jiang, x., & Li, S. (2018). Bas: Beetle antennae search algorithm for optimization problems. *International Journal of Robotics and Control*, *1*, 1–5.

Li, W., Ge, J., & Dai, G. (2015). Detecting malware for android platform: An svm-based approach. In *2015 ieee 2nd international conference on cyber security and cloud computing* (pp. 464–469).

Li, Z., Cheng, Z., Zang, T., & Li, Y. (2023). Mtcd-model: A two-layer model for malicious traffic classification and detection based on hierarchical feature learning. In *2023 international joint conference on neural networks (ijcnn)* (pp. 1–8).

Orun, A., Orun, E., & Kurugollu, F. (2023). Recognition of cyber-intrusion patterns in user cognitive behavioural characteristics for remote identification. *arXiv preprint arXiv:2401.04111*.

Rodgers, W., Attah-Boakye, R., & Adams, K. (2020). Application of algorithmic cognitive decision trust modeling for cyber security within organisations. *IEEE Transactions on Engineering Management*, *69*(6), 3792–3801.

Rustam, F., Aljedaani, W., Elsayed, M. S., & Jurcut, A. D. (2024). Famtds: A novel mfo-based fully automated malicious traffic detection system for multi-environment networks. *Computer Networks*, *251*, 110603.

Veksler, V. D., Buchler, N., LaFleur, C. G., Yu, M. S., Lebiere, C., & Gonzalez, C. (2020). Cognitive models in cybersecurity: learning from expert analysts and predicting attacker behavior. *Frontiers in Psychology*, *11*, 1049.

White, C. A. (2023). *Mixed method exploration of cybersecurity executive decisions and cognitive bias*. Unpublished doctoral dissertation, Marymount University.

Xu, G., & Liu, M. (2021). Malware detection method based on improved harris hawks optimization synchronization optimization feature selection. *Netinfo Security*, *21*, 9–18.

Xue, J., & Shen, B. (2020). A novel swarm intelligence optimization approach: sparrow search algorithm. *Systems science & control engineering*, *8*(1), 22–34.