fire

MDPI

Article

A Fire Evacuation and Control System in Smart Buildings Based on the Internet of Things and a Hybrid Intelligent Algorithm

Ali Mohammadiounotikandi [1], Hassan Falah Fakhruldeen [2,3,4], Maytham N. Meqdad [5], Banar Fareed Ibrahim [6], Nima Jafari Navimipour [7,8,*] and Mehmet Unal [9]

1 Department of Computer and IT Engineering, Faculty of Engineering, South Tehran Branch, Islamic Azad University, Tehran 1584743311, Iran
2 Computer Techniques Engineering Department, Faculty of Information Technology, Imam Ja'afar Al-Sadiq University, Baghdad 10011, Iraq
3 Electrical Engineering Department, College of Engineering, University of Kufa, Kufa 540011, Iraq
4 Computer Technical Engineering Department, College of Technical Engineering, The Islamic University, Najaf 54001, Iraq
5 Intelligent Medical Systems Department, Al-Mustaqbal University, Hillah 51001, Iraq
6 Department of Information Technology, College of Engineering and Computer Science, Lebanese French University, Erbil 44001, Iraq
7 Department of Computer Engineering, Faculty of Engineering and Natural Sciences, Kadir Has University, 34083 Istanbul, Turkey
8 Future Technology Research Center, National Yunlin University of Science and Technology, Douliou 64002, Taiwan
9 Department of Computer Engineering, Nisantasi University, 34485 Istanbul, Turkey
* Correspondence: nima.navimipour@khas.edu.tr or jnnima@yuntech.edu.tw

Abstract: Concerns about fire risk reduction and rescue tactics have been raised in light of recent incidents involving flammable cladding systems and fast fire spread in high-rise buildings worldwide. Thus, governments, engineers, and building designers should prioritize fire safety. During a fire event, an emergency evacuation system is indispensable in large buildings, which guides evacuees to exit gates as fast as possible by dynamic and safe routes. Evacuation plans should evaluate whether paths inside the structures are appropriate for evacuations, considering the building's electric power, electric controls, energy usage, and fire/smoke protection. On the other hand, the Internet of Things (IoT) is emerging as a catalyst for creating and optimizing the supply and consumption of intelligent services to achieve an efficient system. Smart buildings use IoT sensors for monitoring indoor environmental parameters, such as temperature, humidity, luminosity, and air quality. This research proposes a new way for a smart building fire evacuation and control system based on the IoT to direct individuals along an evacuation route during fire incidents efficiently. This research utilizes a hybrid nature-inspired optimization approach, Emperor Penguin Colony, and Particle Swarm Optimization (EPC-PSO). The EPC algorithm is regulated by the penguins' body heat radiation and spiral-like movement inside their colony. The behavior of emperor penguins improves the PSO algorithm for sooner convergences. The method also uses a particle idea of PSO to update the penguins' positions. Experimental results showed that the proposed method was executed accurately and effectively by cost, energy consumption, and execution time-related challenges to ensure minimum life and resource causalities. The method has decreased the execution time and cost by 10.41% and 25% compared to other algorithms. Moreover, to achieve a sustainable system, the proposed method has decreased energy consumption by 11.90% compared to other algorithms.

Keywords: Internet of Things; fire evacuation system; emergency rescue; energy consumption; smart buildings; fire; metaheuristic algorithms

check for updates

Citation: Mohammadiounotikandi, A.; Fakhruldeen, H.F.; Meqdad, M.N.; Ibrahim, B.F.; Jafari Navimipour, N.; Unal, M. A Fire Evacuation and Control System in Smart Buildings Based on the Internet of Things and a Hybrid Intelligent Algorithm. *Fire* **2023**, *6*, 171. https://doi.org/10.3390/fire6040171

Academic Editors: Guowei Zhang, Diping Yuan, Guoqing Zhu and Hongyong Liu

Received: 23 February 2023
Revised: 6 April 2023
Accepted: 13 April 2023
Published: 20 April 2023

1. Introduction

Because of increased industrialization, large industries are gradually expanding. In most situations, industries do not follow the usual criteria for constructing high-rise buildings, resulting in various accidents, including fires [1]. Large public structures are densely occupied and comprise numerous structures and intricate functions. Evacuation is ineffective in the event of abrupt calamities (fire, earthquake, gas leak, etc.) because of the absence of an appropriate evacuation direction [2,3]. Electric fires are most frequent during disasters because of the excessive use of electric systems in buildings. Therefore, we need very intelligent systems to control the situation in these cases. Fires are one of the most frequent disasters with destructive effects, which cause astonishingly direct economic losses and threaten personal safety. Therefore, the correct evacuation plan would reduce casualties when a fire broke out. Currently, most buildings use smoke alarms as fire warnings but lack an evacuation guidance system when a fire breaks out. The traditional one-way instruction fire evacuation indicator system is no longer suitable for current building structures; it makes things worse in the most critical moments, leading people to the wrong escape route. A key technology to develop an intelligent evacuation system is using the basic principles of safety evacuation route instruction requirements combined with the typical fire scenarios and architectural space structures to establish a smart evacuation system [4]. Additionally, although historic buildings' elevators and other components are unsuitable for evacuation purposes due to weak planning and the risk of damaging electric power, electric controls, and fire and smoke protection, several investigations have been conducted to find a solution.

On the other hand, innovative technologies such as the Internet of Things (IoT) and cloud computing have played vital roles in many applications [5–7]. As one of the prominent technologies, the IoT plays a vital role in fire evacuation systems. It enables direct human-free communication between them by combining numerous sensors and items. The "things" in the IoT are tangible objects such as sensors that track and collect data [8]. With the advent of the IoT, people, things, sensors, and services are constantly connected around the globe [9]. The primary goal of the IoT is to offer network architecture with open communication standards and applications to use real and virtual sensors everywhere in a network [10]. Many smart products and devices have sensors, allowing them to perceive information about their surroundings in real-time. This occurrence has ultimately resulted in the fascinating idea of the IoT, which connects all smart things and equips them with data analytics to change how we live, work, and play forever. The IoT includes wearable technology, smart cars, sensors, laptops, industrial and utility components, and wearable components [11,12]. In recent years, numerous entrepreneurs have adopted and put into practice the idea of the IoT in fields such as intelligent cities, intelligent healthcare, intelligent surroundings, intelligent traffic, etc. [13].

Building fire disaster management systems in smart cities have become required due to the worldwide phenomenon of fire [14] and the increasing significance of fire accidents in structures [15]. So, this paper introduces a new energy-aware method to solve the problem of intelligent fire evacuation systems based on the IoT in smart buildings. This paper proposes a hybrid method. The suggested technique finds the quickest and most energy-efficient route to the exit and directs the evacuees there using Particle Swarm Optimization (PSO) and Emperor Penguin Colony (EPC) methods. It combines the PSO with the standard EPC algorithm to reach the best particles sooner. The following are the contributions of this study:

- Decreasing cost in an IoT-based fire evacuation system using a hybrid EPC-PSO method;
- Decreasing energy consumption in an IoT-based fire evacuation system using a hybrid EPC-PSO method;
- Decreasing execution time in an IoT-based fire evacuation system using a hybrid EPC-PSO method.

The following parts make up the rest of the article: There are related works in the second section. The proposed method is outlined in Section 3. Section 4 explores the suggested method and compares it to alternative approaches. Finally, a conclusion and suggested projects are provided in Section 5.

2. Related Work

In this section, some of the existing methods and algorithms proposed by authors for solving problems in the field of fire evacuation systems are mentioned.

Acharjya and Koley [16] recommended an IoT-based fire crisis architecture addressing fire hazards. The structure has a few sensors that can spot smoke, hazardous gas, and fire within a building. The use of Bluetooth modules for distant sensor connection was also shown. A single unit comprises a Bluetooth module, sensors, and a microcontroller board. The mobile application chooses a safe exit from the building after receiving data from the sensors that recognize the fire hazards. The recommended approach made it feasible to alert disaster management teams and local emergency agencies by sharing the hazardous area via a cloud service. Additionally, in order to assure the promptness and dependability of network-wide communication, a lightweight data-oriented publish–subscribe messaging protocol was used. The outcomes demonstrated that the model might prevent a person's death in a fire.

Mekni [17] proposed a low-cost, intelligent IoT-based fire detection and monitoring system design. Their solution allowed users to remotely monitor the status of a building via an intuitive dashboard that displays data for all metrics on a single page. When a gathered parameter exceeds its threshold, a buzzer, a red light, and a text message are delivered to the homeowner's mobile phone so she/he can phone the authorities to arrive promptly. The proposed method counted the number of individuals at the fire scene, facilitating the evacuation procedure.

For real-time fire detection, monitoring, and evacuation aid, Singh, Birajdar [18] recommended a hybrid solution using 2.4 GHz Zigbee and Long-Range (LoRa). The architecture's five main components are the end device, the safety operation controller, the evacuation path display controller, the gateway, and the vision node. The end device and vision node provide real-time data and visual representations of a fire's progression. In addition, an OPNET simulator Zigbee simulation was performed to examine the network characteristics. The evacuation path display controller implemented Dijkstra's shortest-path algorithm to offer the quickest escape route in the event of a fire. The results showed that the proposed approach for deploying customized hardware for fire detection was superior to prior studies.

Zualkernan, Aloul [19] showed a system that intelligently steered building inhabitants toward a safe evacuation by using IoT technology to track the location of the fire and the occupants. The technique makes use of Bluetooth Low Energy (BLE) beacons to locate people inside using their mobile phones. It also kept an eye on dangerous areas using smoke and temperature sensors. WiFi and DigiMesh were only two of the networks used to increase the system's fire resistance. The recommended technology might support a network of emergency communications covering the whole city.

Khan, Aesha [1] suggested developing an IoT-based intelligent fire evacuation system that could efficiently direct individuals along an escape route in a fire situation. The A* search algorithm controls the primary module of the suggested model. By directing individuals along the quickest, safest route feasible, their suggested approach assisted people in navigating out of danger. The second-best path was displayed if the first ideal path was already occupied. The findings demonstrated that the suggested approach assisted evacuees in reaching the exit and instantly alerted the fire department to the rescue effort.

Aymaz, Çavdar [20] proposed a technique to explore wayfinding during the fire. Wayfinding depends on the building type. Moreover, they investigated the possible influence of smoke, light, and distance on route determination for fire evacuation. When a fire occurs, the system provides evacuation route guidance to people so that they are able to avoid the hazard. Here, Particle Swarm Optimization (PSO) was used to optimize the evacuation route. The results showed that the fire evacuation system could recommend the shortest and safe route.

Lozowicka and Nikonczuk [21] presented the method of using the Genetic Algorithm (GA) to improve the system of evacuation in case a fire were to grow on the ferry "Polonia". The information from the fire detector system concerning the zone cut-off was used. The two-dimensional grid of escape routes was described as a matrix of coordinates. The GA looked for the best evacuation route while taking into consideration the zone where the fire had already grown. The authors presented that the best evacuation way was from the business-class cabin. The fire started at one of the cabins. The results showed that the system signaled escape routes with power-supplied double-direction luminous arrows that could offer the passengers the optimal escape route.

Table 1 compares existing techniques for the fire evacuation and control systems in smart buildings based on the IoT, detailing each technique's objective, the strategy employed, evaluation tools, and performance measures.

Table 1. Side-by-side comparison of the studied articles.

Article	Goal	Used Technique	Evaluation Tools
Acharjya, Koley [16]	Proposing a fire crisis leave framework for fire risks	Using IoT sensors and Bluetooth module	Sensors, Bluetooth modules, and cell phone
Mekni [17]	Designing and implementing an intelligent system to monitor the building's status remotely in the case of fire	Using IoT sensors	Testing under different scenarios in real-time
Singh, Birajdar [18]	Assisting in the safe evacuation of the building and real-time fire detection, monitoring	Proposing an architecture based on Zigbee and LoRa	Zigbee simulation and LoRa-based hardware
Zualkernan, Aloul [19]	Tracking the location of the fire and building occupants	Proposing a system based on BLE, WiFi, and DigiMesh	Testing the system over an area of 1600 m^2
Khan, Aesha [1]	Proposing an IoT-based intelligent fire evacuation system	Using JavaScript and MySQL database	Dijkstra's algorithm
Aymaz, Çavdar [20]	Exploring the best escape way during a fire evacuation	Using PSO	Matlab
Lozowicka and Nikonczuk [21]	Improving the system of evacuation	Using GA method in the optimization of escape routes	Matlab

3. Methodology

Building fires result in a great number of fatalities and property losses; hence, adequate action must be taken. The effort to create such disaster management systems using information and communication technology has grown, and several studies have been carried out in real-world settings [22]. This paper introduces a new method for a fire evacuation and control system in smart buildings based on the IoT to efficiently direct individuals along an evacuation route in the event of fire incidents. First, the system architecture is explained, then a hybrid method is described.

3.1. System Architecture

There are various vital issues in integrated smart buildings, such as energy management and fire evacuation systems. The subject here is developing an intelligent fire evacuation system. Figure 1 depicts the system architecture and operating procedure of this work. The system's operation is broken down into the following steps: (1) A fire breaks out within the structure. (2) The building's smoke and fire detectors identify a fire, and an evacuation is initiated. (3) The IoT and Dynamic Cellular Automata (DCA) analysis is initialized. The IoT system begins to monitor smoke and fire alarm signals, analyzes the data of the surveillance camera system, and then uploads the data to the edge computing gateway via the LoRa IoT wireless communication module. (4) Using a BIM-based visualization system, the DCA model is executed on the edge gateway to determine evacuation routes depending on the current fire scenario and manpower situations. On this basis, the system assigns each sign its direction value. (5) Using IoT LoRa connectivity, the system transmits signage direction data to smart signage devices in order to direct the audience to evacuate via the ideal route [23].

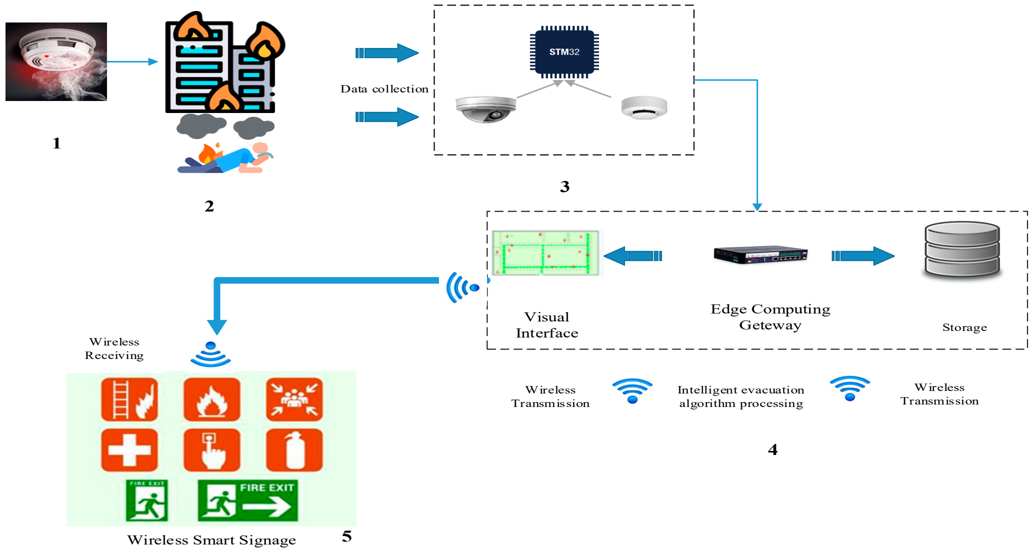

Figure 1. A fire evacuation and control system architecture in smart buildings based on IoT.

To create the initial population, a number of separate routes from the departure node to the destination node are created. Population size is the total number of people in a population. In this study, the population size is fixed at 100. Every person symbolizes the arrangement of nodes along a path that connects a departure node with a destination node [24]. A departure node ID, through-node IDs, and a destination node ID are therefore used to denote the persons. Variable lengths are seen in the people. The person is represented as follows: for instance, if the departure node ID is 1, the destination node ID is 7, and the order of the through-node IDs is 2-4-5 [1,2,4,5,7] (see Figure 2).

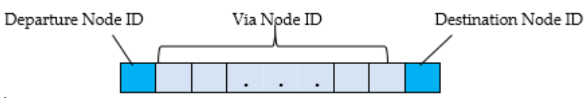

Figure 2. Individual route planning method.

3.2. Problem Formulation

The current analysis is based on Francis's [25] initial formulation of the building evacuation issue, which took into account the enclosure evacuation optimization. Nevertheless, by narrowing the scope of the issue, we may utilize more accurate models of people's mobility and include behavior models. Consider an enclosure with k inhabitants and n potential exits that are positioned so that there is no obstruction to occupant movement to any of them. We may assume a steady flow via each exit because of how the people are dispersed around the space. Each exit also has an evacuation path; therefore, there will be n distinct evacuation routes, each of length lj and having an access area of aj, for n exits. When every inhabitant of the enclosure has reached his or her destination, the evacuation of the enclosure is complete. The time required for the evacuation is determined by the lengthiest path and is shown by the moment when the final passenger arrives at his or her destination.

To reduce the cost, energy usage, and execution time (z), the following model may be used to predict how many individuals are required [26]:

$$\text{min}z = Max\left[t_j(x_j)\right] \qquad j = 1.\ \ldots.\ n \tag{1}$$

$$s.t.\ \sum\nolimits_{j=1}^{n} x_j = k. \tag{2}$$

$$x_j \geq 0. \tag{3}$$

where $\boldsymbol{j}$ is the exit number, $\boldsymbol{xj}$ is the number of individuals who will evacuate the enclosure by exit $\boldsymbol{j}$, and $\boldsymbol{t_j}$ $(\boldsymbol{x_j})$ is the evacuation function determining how long it will take $\boldsymbol{x_j}$ individuals to exit the enclosure. Although the variable $\boldsymbol{x_j}$ (the number of persons) should be an integer, most problems with large values of $\boldsymbol{k}$ are regarded as real variables, which are then adjusted to the nearest integer value once the solution is found. In contrast to the negativity requirement of the variables, which states that the number of persons who select a certain route cannot be negative, constraint requires that all $\boldsymbol{k}$ individuals depart the enclosure.

3.3. Emperor Penguins Colony (EPC) Algorithm

Algorithm 1 describes the EPC method's pseudo-code [27]. These are the guidelines for this algorithm:

(1) All initial-population penguins radiate heat and are attracted to one another according to their absorption coefficient.
(2) It is believed that every penguin's body surface area is the same.
(3) Penguins ignore the impacts of the earth's surface and atmosphere and absorb all heat radiation.
(4) The penguins' thermal radiation is considered linear.
(5) Two penguins are attracted to one another based on the quantity of heat in the space between them. The amount of heat received decreases with increasing distance; on the other hand, it increases with decreasing distance.

Throughout the absorption procedure, the penguin spiral moves irregularly and with evenly distributed variance.

Algorithm 1: Pseudo-code of the EPC algorithm.

```
Begin
    generate initial population array of EPs (Colony Size); generate position of each EP;
    generate cost of each EP;
    determine initial heat absorption coefficient;
    for it = 1 to MaxIteration do
        generate repeat copies of population array;
        for i = 1 to n population, do
                    if cost_j < cost_i then
                        calculate heat radiation (Equation (9)); calculate attractiveness
(Equation (10));
                        calculate coordinated spiral movement; determine new position;
                        evaluate new solutions;
                    end for
      end for
    end for
        sort the solutions and find the best one;
End
```

Heat radiation transfer can calculate the heat intensity and attraction. Each penguin's body surface area must be calculated to add the heat radiation from each penguin. The subsections that follow provide more information on the aforementioned amounts. Additionally, penguins' synchronized spiral motions have an impact on their indirect movement.

- ➢ Body surface area

This procedure must calculate thermal radiation, attraction, heat absorption, and spiral movement coordination. The body surface area is required to calculate heat radiation, which is vital to determine heat radiation transmission and compute attractiveness [28]. Using Equation (4), the main body surface area *Atrunk* (m^2) is obtained [29].

$$A_{trunk} = 2\pi \frac{ab}{e} sin^{-1} e + 2\pi b^2 \tag{4}$$

where $e = a^2 - b^2$, a denotes half the body length of the neck, and b is equal to the main trunk's half diameter. Through the cone equation, the beak area A_{beak} is calculated by Equation (5):

$$A_{beak} = \pi rs \tag{5}$$

where the beak length is defined as the biggest sectional area of the beak divided by a radius. Equation (6) determines the head area using the sphere equation [27]:

$$A_{head} = \pi d^2 - \pi r^2 \tag{6}$$

Ultimately, the flippers are seen as rectangles with dimensions of l and w, which are calculated using Equation (7).

$$A_{flipper} = l \times w \tag{7}$$

- Heat transfer

A route or process function is the transport of heat. As a result, how this procedure proceeds and the system's initial and end states determine how much heat is transported during the thermodynamic process that affects the system's state. The heat transfer coefficient computes the heat flux ratio to the thermodynamic heat. A quantitative, vectorial description of heat movement via a surface is called a heat flux. When objects and their environment reach identical temperatures, thermal equilibrium is reached. There are three different types of heat transfer: thermal conduction, convection, and thermal radiation [30]. It is the direct transfer of kinetic energy particles over a barrier between two systems at the microscopic level. Heat convection happens when a fluid's bulk flow transmits heat together with the flow of its constituent parts. The EPC technique uses the radiation heat transfer criteria while disregarding the conduction and convectional heat transfer requirements. The amount of radiation a penguin receives is a determinant of their attractiveness. The warmer penguin is more appealing when there is less distance between a penguin and a warmer penguin. The heat exchange of an emperor penguin is determined using a model of distributed heat transfer. This model assumes that the penguin is in thermal equilibrium with its surroundings and assembles the heat transfers from each part of its body, including its feet, head, and flippers, according to Equation (8) [31]:

$$q_{total} = q_{trunk} + q_{head} + q_{flippers} + q_{feet} \tag{8}$$

Equation (9) can be used to calculate the radiation emitted from each body component of the surface area.

$$Q_{penguin} = A\varepsilon\sigma T_S^4 \tag{9}$$

where $Q_{penguin}$ is heat transfer per unit time (W), and A is the total surface area. ε is the emissivity of bird plumage, σ is the Stefan–Boltzmann constant, and Ts is the absolute temperature in Kelvin (K) [32].

- Heat intensity and attractiveness

Penguins are considered linear heat sources based on their bodies' geometry. Ultimately, we can determine the allure of the penguin by Equation (10) by combining the radiation equation with the linear heat source equation:

$$Q = A\varepsilon\sigma T_s^4 e^{-\mu x} \tag{10}$$

The attenuation factor is μ, and the separation between two linear sources is x. The μ is considered a positive value for determining the convergence rate. The concept of heat absorption grows as the attenuation coefficient falls [27].

- Coordinated spiral-like movements

Coordinated spiral-like movements are huddles in which clockwise movement is performed around a fixed center. In this instance, the system's structure has indeterminate limits and a spiral pattern around its core. The temperature is warmest at the huddle's core and significantly colder at its periphery. Penguins do not compete for personal advantage. The entire huddle has a slow spiral motion, allowing each penguin to rotate through all positions [33]. Let's imagine that there are two penguins, i and j. The penguin that requires heat is continually moving toward the warmer penguin. Because j in this instance is warmer, the spiral movement is from i to j in this instance. The spiral type in this study is referred to as a logarithmic spiral. Equation (11) takes the logarithmic spiral equation into account:

$$r = ae^{b\theta} \tag{11}$$

where θ is the angle of the x-axis, a and b are constant and are selected randomly, and r shows the distance from the origin.

According to [34], the logarithmic spiral may be created from equally spaced rays when drawing the perpendicular to a neighboring ray after starting along one. As the number of rays approaches infinity, the segmentation pattern resembles a smooth logarithmic spiral. The pace at which penguin i travels a distance in the direction of j is determined by the attraction value, Q. Due to its attraction rating, it cannot reach its destination and stops after traveling a considerable distance. The new location of i is k. This individual knowledge of spiral-like movement can be utilized in future relocations or shared with the entire community. The distance between i and j can be obtained using Equation (12) to determine the equation for this spiral-like motion.

$$\begin{aligned} &D_{ij=\int_i^j ds=\int_{\theta_i}^{\theta_j} \sqrt{\left(\frac{dr}{d\theta}\right)^2+r^2 d\theta}} \\ &= \int_{\theta_i}^{\theta_j} \sqrt{a^2b^2e^{2b\theta} + a^2e^{2b\theta} d\theta} \\ &= a\sqrt{b^2+1} \int_{\theta_i}^{\theta_j} e^{b\theta} d\theta \\ &= \frac{a}{b}\sqrt{b^2+1} \left(e^{b\theta_j} - e^{\theta_i}\right) \end{aligned} \tag{12}$$

The result is multiplied by attractiveness Q to get the equation of distance i to k according to Equation (13):

$$\begin{aligned} &D_{ik} = Q\frac{a}{b}\sqrt{b^2+1}\left(e^{b\theta_j} - e^{\theta_i}\right) \\ &= \int_i^k ds \\ &= Q\frac{a}{b}\sqrt{b^2+1}\left(e^{b\theta_k} - e^{\theta_i}\right) \end{aligned} \tag{13}$$

The relation between Cartesian and Polar coordinates (θ) is calculated by Equations (14) and (15) [35]:

$$\theta_k = tan^{-1}\frac{y}{x} \tag{14}$$

where

$$\begin{aligned} x_k &= a\ cos\ \theta_k e^{b\theta_k} \\ y_k &= a\ sin\ \theta_k e^{b\theta_k} \end{aligned} \tag{15}$$

3.4. Particle Swarm Optimization (PSO)

PSO is a population-based optimization tool that can be deployed and utilized to tackle various function optimization issues [36]. In the simulation, the behavior of particles is influenced by the best local particle or the best global particle. If the current velocity is added to the current position, a new (future) state will appear. It uses the "particle" to introduce solutions to optimization problems. The location and velocity vectors determine the distance and orientation of particles. The relevant fitness function defines the fitness function related to particles. The particle alters its position persistently by looking for the local optimal. Particles are evaluated iteratively in the treatment area to determine the appropriate treatment.

Imagine that m particles exist in a D-dimensional space, and the particle locations are $x_i = (x_{i1}.x_{i2}. \ldots .x_{id})$; the fitness function is f_i, and the velocity of a particle at the same time is $v_i = (v_{i1}.v_{i2}. \ldots .v_{id})$. The best position for a particle i is the value of $p_i = (p_{i1}.p_{i2}. \ldots .p_{id})$, also called p_{best}. The best location for whole particles is g_{best}. The speed and location of PSO are calculated by Equations (16) and (17) [37]:

$$v_{id}^{t} = wv_{id}^{t-1} + c_1 rand()\left(p_{id} - x_{id}^{t-1}\right) + c_2 rand()\left(p_{gd} - x_{id}^{t-1}\right) \tag{16}$$

$$x_{id}^{t} = x_{id}^{t-1} + v_{id}^{t-1} * \Delta t \tag{17}$$

where v_{id}^{t} is the speed amount of the particle i in the t iteration. x_{id}^{t} is the location amount of the particle i in the t iteration. w is the inertia weight utilized to regulate the speed of particle update. c_1, c_2 are stable and frequently are the amounts of 2. *Rand ()* is a random number in [0, 1]. Δt is the time interval, often $\Delta t = 1$.

PSO has several advantages, such as easily adjustable parameters, high effectiveness, quick search times, etc., but it is vulnerable to the local optimal solution issue when working with discrete problems. The PSO algorithm must be improved to solve the work planning issue in the multi-robot system. Three methods are used to promote PSO. The PSO method was initially intended to be adapted to discrete treatment space issues through the particle coding and decoding procedure. The "stop" speed would be wrongly reset to evade the particle being stuck in the local ideal. The particle's location would then wrongly reset as it approached the best local solution, enlarging the particle swarm search space and raising the possibility of discovering the g_{best}.

To find the appropriate place, particles are accidentally examined over the probable treatment range. The particle currently scouts the area around the barrier when it flies over the potential boundary. The particle just needs to be moved to the potential side. The particle can keep searching for the optimum answer no matter how close it is to the border. The calculated function is performed to set the amount to the total of the exceeded boundary amount when the particle is outside the permitted range in order to keep it within the acceptable border location [37]. Algorithm 2 depicts the pseudo-code of PSO [38].

Algorithm 2: Pseudo-code of the PSO algorithm.

```
Begin
initialize the population with n particles;
while (termination creation is not met)
        for i = 1:n
            Xi←particle i;
                if (f(Xi) > pbest)
                    pbest = f(Xi);
              end if
                    gbest ←best Xi among all solutions in the population;
                    Xi←calculate the velocity based on Equation (16);
                    update the particles based on Equation (17);
            end for
      end while
End
```

3.5. Hybrid Method

Scholars have used hybrid algorithms to handle optimization problems throughout the last few decades. These hybrid algorithms perform better than their counterparts in solving multiple complex issues [39]. Intelligent heuristic algorithms like EPC and PSO have been widely used in optimization. The performance of EPC and PSO has advanced based on ongoing enhancements made to their initial versions. Figure 3 depicts the suggested flowchart. The steps of the proposed combined method are described as follows:

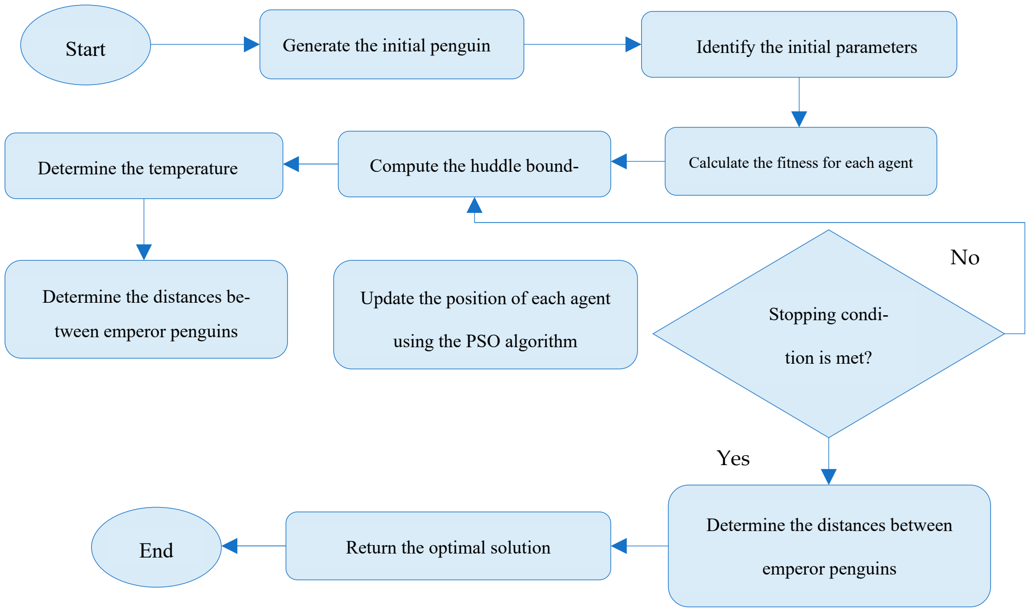

Figure 3. Hybrid EPC and PSO flowchart.

Step 1: Population initialization progress: smart building fire evacuation and control system based on IoT control parameters.

Step 2: Choose the basic parameters that will constrain the intelligent fire evacuation system's capabilities.

Step 3: Calculate the cost, energy use, execution time, and fitness metrics.

Step 4: Determine the emperor penguin behaviors' boundaries of huddle circumstances.

Step 5: The relationship between the huddling behavior and the temperature profile may be calculated.

Step 6: Determine the distance between emperor penguins in order to update their locations and choose the best search agents found using the PSO algorithm.

Step 7: The distance and PSO algorithm update the penguins' positions.

Step 8: The emperor penguins' and the fire evacuation and control system in smart buildings based on the IoT boundary circumstances are examined.

Step 9: Update the search agent's location and fitness value.

Step 10: When the algorithm completes the most iterations, it becomes stationary.

Step 11: Show the best-obtained route.

A fitness function processes an individual positive integer to represent the route's appropriateness. We have used three main objectives to produce the fitness value achieved from Equation (18).

$$F = \alpha_1 \times cost + \alpha_2 \times execution\ time + \alpha_3 \times energy\ consumption \quad (18)$$

where F is between 0 and 1. The constants α_1, α_2, and α_3 are weight coefficients [40]. The sum of weights equal to one:

$$\sum_{i=1}^{3} \alpha_i = 1 \quad (19)$$

The elements of the fitness function should be normalized by Equation (20):

$$N\,(V) = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (20)$$

where *N(V)* is the normalized value of either cost, execution time, or energy consumption.

4. Simulation

The presentation of the suggested approach is supported in this section. The success of the suggested strategy must be demonstrated using the performance evaluation. The specifics of the empirical settings, including the simulation environment, dataset, and simulated parameters, are presented in the remaining paragraphs of this section. Lastly, this section describes comparisons and results.

4.1. Simulation Environment

Arduino UNO was coupled to an nRF24L01 transmitter for the prototype workstation. In this instance, the Arduino UNO code was configured so that the transmitter functioned as a receiver. Python was employed to collect the sensor data from the mesh, while MATLAB® was utilized to evaluate the data and activate the emergency system. When executed, the Python script establishes a serial communication port between the Arduino and itself. It constantly verifies serial data. If any data are received, they are transformed into a specified three-byte format before being sent to MATLAB®. We used MATLAB version R2020a for path planning. We used some libraries and toolboxes in MATLAB, such as Open Genetic Algorithm Toolbox, Genetic and Evolutionary Algorithm Toolbox (GEATbx), and some custom libraries. The MATLAB® script developed an occupancy grid matrix where the system was installed upon input from Python (based on pyeasyga, pygalib, and DEAP libraries). The grid matrix can be produced using vectors as inputs in MATLAB's robotics program. MATLAB® would generate barriers to mimic fire during fire occurrences and then utilize a path-planning algorithm to determine all viable departure routes from each room. The path-planning algorithm computes the lengths to each exit from each room by connecting all the dots, avoiding barriers (walls and spots where smoke detectors detect smoke), and generating numbers based on this path computation.

4.2. Dataset and Simulation Parameters

This section tests the four algorithms (EPC, GA, PSO, and ALO) on various test scenarios. The basic idea for the GA and PSO are derived from [21], and [20], respectively. Moreover, we used the basic algorithms for EPC and ALO since they had not been used in this problem yet. There are three examples in the literature, and eighty instances with various dimensions and complexity levels were picked randomly. Each of the four algorithms has several parameters. The parameters used in the research are listed in Table 2.

4.3. Comparisons

In this section, the obtained results are evaluated. The values obtained from the hybrid method are compared with the results of other algorithms (GA, PSO, and ALO). Calculating the objective function, as illustrated in Figure 4, allows a comparison of the relative performance for various iteration counts. According to Figure 4, the fitness amount ranges from 0 to 1. This simulation looks at the fitness of 200 iterations. The fitness value declines as the number of iterations rises. As can be seen, the fitness amount between the 100th and 200th iterations falls to its lowest point of 0.24.

Another crucial test for metaheuristic algorithms is to examine the stability of the associated algorithm. Metaheuristic algorithms are unexpected and uncertain; hence, it is important to look at their stability. If an algorithm returns the same or similar responses for different performances, it is said to be stable. Figure 5 shows that the suggested approach was used 48 times to assess its stability. In the diagrams, the horizontal axis represents the algorithm's execution order, while the vertical axis represents the fitness in each execution order. The recommended algorithm is extremely stable, according to stability-related statistics.

Table 2. Used parameters for the experiment.

Parameter	Value
The number of IoT nodes	10–200
Number of routes available	50–120
Area	About 1000 m^2
The primary energy of nodes	1–10 mj
Requested time	1–10 ms
The number of servers per IoT node	1–5
Initial cost	1–10 $
EPC algorithm parameters	
Colony size	100
Heat radiation damping ratio	0.9995
Attenuation coefficient	1
Attenuation coefficient damping ratio	0.9998
Mutation coefficient	0.2
Mutation coefficient damping ratio	0.03
a	0.2
b	0.5
α_1, α_2, α_1	1/3
Genetic Algorithm (GA) parameters	
Number of chromosomes (solutions)	100
Possibility of crossover	0.7
Mutation rate	0.1
PSO parameters	
Number of particles	100
C_1	0.5
C_2	0.5
r_1	0.5
r_2	0.5
Inertia weight (W_{max})	0.9
Inertia weight (W_{min})	0.1
Maximum particle velocity (V_{max})	3
Minimum particle velocity (V_{min})	−3
Ant Lion Optimizer (ALO) parameters	
Max no. of iterations	100
No. of dim	5
Lower bound	1
Upper bound	8–13
Best score	Elite antlion fitness
Best position	Elite antlion position

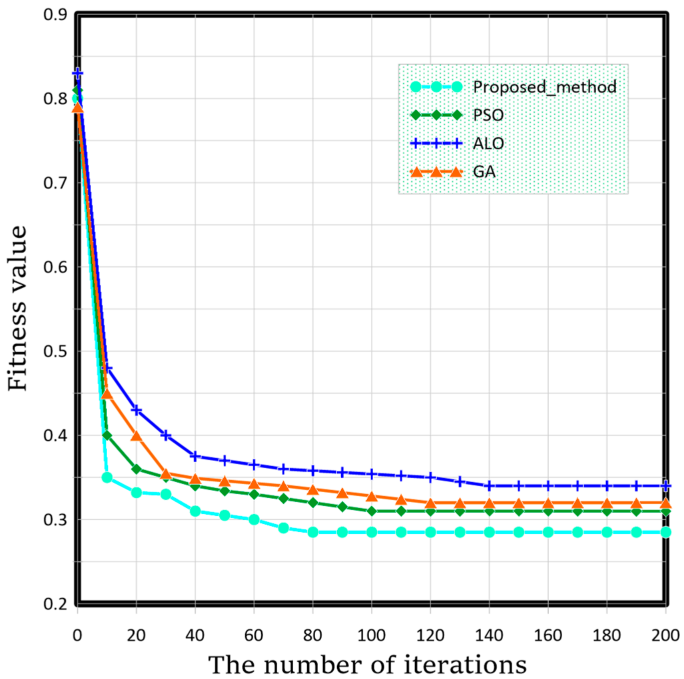

Figure 4. The result of convergence in 200 iterations.

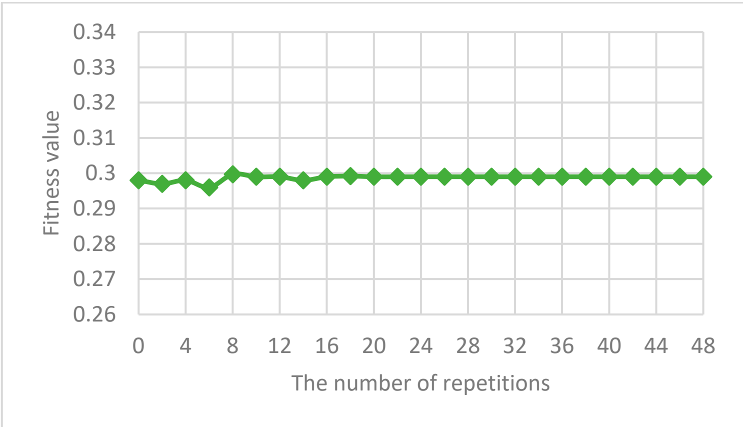

Figure 5. Stability test in 48 iterations.

The obtained results to assess the criteria parameters of the energy consumption, execution time, and cost are shown in Figures 6–8. Many studies have paid particular attention to energy consumption [41,42]. The test of average energy consumption was proposed to measure the routing model's effectiveness in the network's energy efficiency. Figure 6 illustrates the outcomes of this test of the proposed method. They confirm that the hybrid method can deliver better results than other methods due to considering the residual energy of the node, the amount of traffic in the routes of the parent node (which delays queue processing, transmission, and distribution), and also the amount of higher-level parent access. The mean energy usage of the nodes is included in the network time intervals, which indicates the knowledge of the suggested method regarding the energy and status of the network. Moreover, the energy consumption trend in the proposed network will decrease over time. The energy consumption rate of our method decreased at the end of the simulation time compared to other methods.

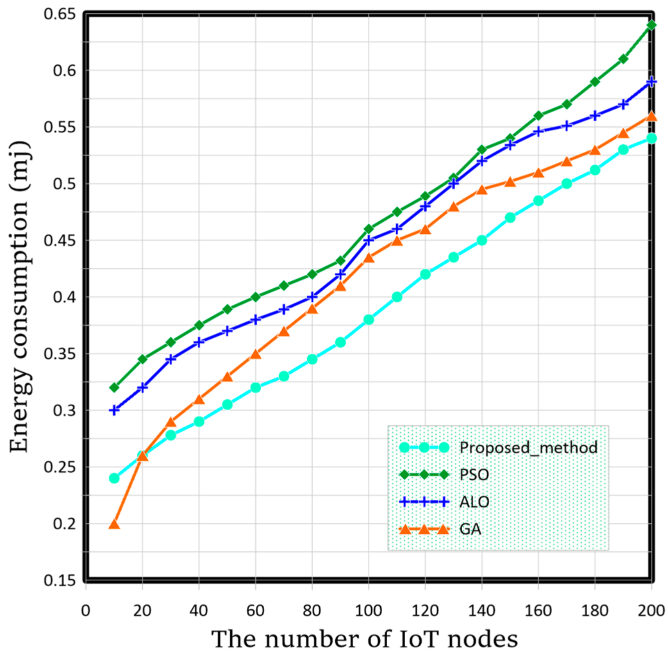

Figure 6. Comparison of the suggested method's energy consumption with that of the PSO, ALO, and GA algorithms.

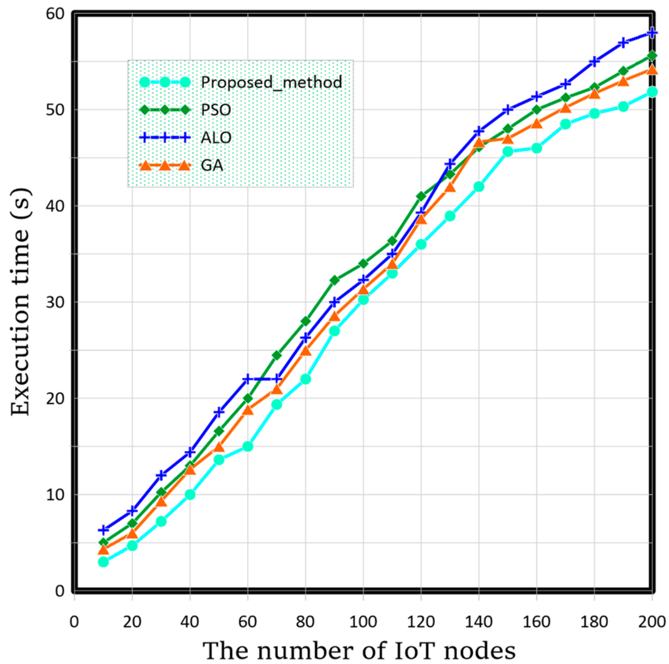

Figure 7. Comparison of the execution time among the proposed method, PSO, ALO, and GA algorithms.

The obtained results for the execution time criterion are illustrated in Figure 7. This figure shows that the hybrid method has the shortest execution time among the investigated methods since it can converge faster than others. Therefore, the proposed methods significantly reduce the time to perform the whole process compared to other methods. The results obtained in the experiments of this research show that the cost of the proposed method is reduced compared to the compared methods, i.e., the GA, ALO, and PSO algorithms. As shown in Figure 7, the PSO algorithm, ALO, and GA have the best results after the proposed method.

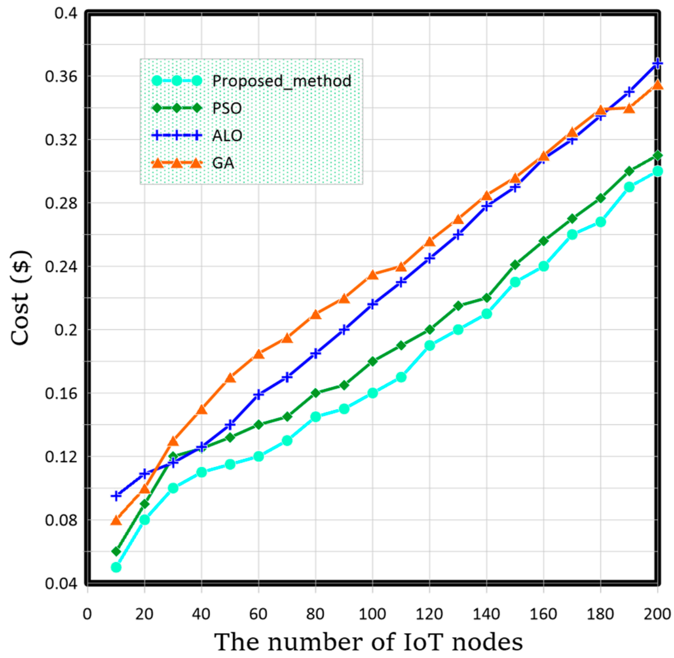

Figure 8. Comparison of the cost between the proposed method, PSO, ALO, and GA algorithms.

The results of the proposed method for reducing cost compared to other methods are shown in Figure 8. The method was compared to the PSO, ALO, and GA algorithms. The obtained results show that the cost is significantly decreased compared to the previous methods. In the experiment conducted in this research, the results obtained for the cost of the ALO algorithm were better than those of the PSO and GA algorithms.

5. Conclusions and Future Work

Recently, numerous cities have seen the construction of intricate and lofty structures. Fire safety should be a top priority for architects, engineers, and governments. We understand the value of gathering information in the foreground to help firefighting operations, evacuation protocols, etc., through prior fire experience. Recent developments in IT, data analytics, and other detection and monitoring technologies have allowed academics who study fire safety to rethink fire safety procedures in the built environment. IoT-assisted building fire evacuation is a novel method that enhances evacuation by relying on real-time fire–ground data, such as the nature of the fire and the people, to make escape decisions. This research proposes a hybrid nature-inspired optimization algorithm (EPC-PSO) for smart buildings' fire evacuation and control systems based on the IoT. During fire incidents, MATLAB would produce barriers to simulate flames and then use a path-planning algorithm to identify all feasible exit routes from each room. The path-planning algorithm connects all the dots, avoids obstacles (walls and areas where smoke detectors detect smoke), and generates values based on this path calculation to determine the distances to each exit from each room. Simulated fires were created near two sensors to test the path-planning algorithm. Our proposed method was successfully tested in multiple iterations. Future research will also concentrate on broadening the applicability of this technology to other domains, such as maritime vessels and evacuation within buildings, disaster safety through Web or mobile application services, and preventative measures for optimal catastrophe recovery. We want to retrofit a conventional fire alarm with our prototype and test it to determine how the system would respond in real-world scenarios. Moreover, in the future, we can reduce the number of computation nodes to increase efficiency.

Author Contributions: Conceptualization, A.M., H.F.F., M.N.M. and B.F.I.; formal analysis, H.F.F. and M.N.M.; funding acquisition, N.J.N. and M.U.; investigation, A.M.; project administration, N.J.N. and M.U.; resources, H.F.F. and B.F.I.; software, A.M., H.F.F. and M.N.M.; supervision, N.J.N. and M.U.; validation, A.M., H.F.F. and M.N.M.; visualization, B.F.I., N.J.N. and M.U.; writing—original draft, A.M., H.F.F. and M.N.M.; writing—review and editing, M.N.M., B.F.I., N.J.N. and M.U. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: All data are reported in the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Khan, A.; Aesha, A.A.; Aka, J.S.; Rahman, S.F.; Rahman, M.J.-U. An IoT based intelligent fire evacuation system. In Proceedings of the 2018 21st International Conference of Computer and Information Technology (ICCIT), Dhaka, Bangladesh, 21–23 December 2018.
2. Jiang, H. Mobile fire evacuation system for large public buildings based on artificial intelligence and IoT. *IEEE Access* **2019**, *7*, 64101–64109. [CrossRef]
3. Fang, H.; Lo, S.; Lo, J.T. Building fire evacuation: An IoT-aided perspective in the 5G era. *Buildings* **2021**, *11*, 643. [CrossRef]
4. Zheng, W.; Cai, Q. Implementation of intelligent fire evacuation route based on Internet of things. In Proceedings of the 2015 IEEE Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chongqing, China, 19–20 December 2015.
5. Darbandi, M. Proposing New Intelligence Algorithm for Suggesting Better Services to Cloud Users based on Kalman Filtering. *J. Comput. Sci. Appl.* **2017**, *5*, 11–16.
6. Darbandi, M. Kalman filtering for estimation and prediction servers with lower traffic loads for transferring high-level processes in cloud computing. *HCTL Int. J. Technol. Innov. Res.* **2017**, *23*, 10–20.
7. Rejeb, A.; Rejeb, K.; Simske, S.; Treiblmaier, H.; Zailani, S. The big picture on the internet of things and the smart city: A review of what we know and what we need to know. *Internet Things* **2022**, *19*, 100565. [CrossRef]
8. Zhang, G.; Navimipour, N.J. A comprehensive and systematic review of the IoT-based medical management systems: Applications, techniques, trends and open issues. *Sustain. Cities Soc.* **2022**, *82*, 103914. [CrossRef]
9. Ghaffar, Z.; Alshahrani, A.; Fayaz, M.; Alghamdi, A.M.; Gwak, J. A topical review on machine learning, software defined networking, internet of things applications: Research limitations and challenges. *Electronics* **2021**, *10*, 880.
10. Heidari, A.; Jabraeil Jamali, M.A.; Jafari Navimipour, N.; Akbarpour, S. Deep Q-Learning technique for offloading offline/online computation in blockchain-enabled green IoT-Edge scenarios. *Appl. Sci.* **2022**, *12*, 8232. [CrossRef]
11. Khraisat, A.; Gondal, I.; Vamplew, P.; Kamruzzaman, J.; Alazab, A. A novel ensemble of hybrid intrusion detection system for detecting internet of things attacks. *Electronics* **2019**, *8*, 1210. [CrossRef]
12. Dang, L.M.; Piran, M.J.; Han, D.; Min, K.; Moon, H. A survey on internet of things and cloud computing for healthcare. *Electronics* **2019**, *8*, 768. [CrossRef]
13. Sun, X.; Ansari, N. EdgeIoT: Mobile edge computing for the Internet of Things. *IEEE Commun. Mag.* **2016**, *54*, 22–29. [CrossRef]
14. Bilbao, B.; Mistry, J.; Millán, A.; Berardi, A. Sharing multiple perspectives on burning: Towards a participatory and intercultural fire management policy in Venezuela, Brazil, and Guyana. *Fire* **2019**, *2*, 39. [CrossRef]
15. Hassan, M.K.; Hossain, M.D.; Gilvonio, M.; Rahnamayiezekavat, P.; Douglas, G.; Pathirana, S.; Saha, S. Numerical investigations on the influencing factors of rapid fire spread of flammable cladding in a high-rise building. *Fire* **2022**, *5*, 149. [CrossRef]
16. Acharjya, P.P.; Koley, S.; Chattopadhyay, S. A Smart Emergency Exit System Based on IoT for Fire Perils. In Proceedings of the International Conference on Advanced Computing Applications, Coimbatore, India, 4–5 March 2022; Springer: Berlin/Heidelberg, Germany, 2022.
17. Mekni, S.K. Design and Implementation of a Smart fire detection and monitoring system based on IoT. In Proceedings of the 2022 4th International Conference on Applied Automation and Industrial Diagnostics (ICAAID), Hail, Saudi Arabia, 29–31 March 2022.
18. Singh, R.; Birajdar, G.S.; Rashid, M.; Gehlot, A.; Akram, S.V.; AlGhamdi, A.S.; Alshamrani, S.S. Hybrid Architectural Network Implementation to Realize a Fire Evacuation Path with 2.4 GHz Zigbee and LoRa. *Sustainability* **2021**, *13*, 13238. [CrossRef]
19. Zualkernan, I.A.; Aloul, F.A.; Sakkia, V.; Al Noman, H.; Sowdagar, S.; Al Hammadi, O. An IoT-based emergency evacuation system. In Proceedings of the 2019 IEEE International Conference on Internet of Things and Intelligence System (IoTaIS), Bali, Indonesia, 5–7 November 2019.
20. Aymaz, Ş.; Çavdar, T.; Çavdar, A.D. Fire evacuation route determination based on particle swarm optimization. In Proceedings of the 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), Malatya, Turkey, 16–17 September 2017.
21. Lozowicka, D.; Nikonczuk, P. Genetic Method of Optimization of Evacuation Ways in Cases the Fire Growth at Ferryboat. *IFAC Proc. Vol.* **2001**, *34*, 233–235. [CrossRef]
22. Park, S.; Park, S.H.; Park, L.W.; Park, S.; Lee, S.; Lee, T.; Lee, S.H.; Jang, H.; Kim, S.M.; Chang, H. Design and implementation of a smart IoT based building and town disaster management system in smart city infrastructure. *Appl. Sci.* **2018**, *8*, 2239. [CrossRef]

23. Ji, Y.; Wang, W.; Zheng, M.; Chen, S. Real Time Building Evacuation Modeling with an Improved Cellular Automata Method and Corresponding IoT System Implementation. *Buildings* **2022**, *12*, 718. [CrossRef]
24. Ikeda, Y.; Inoue, M. An evacuation route planning for safety route guidance system after natural disaster using multi-objective genetic algorithm. *Procedia Comput. Sci.* **2016**, *96*, 1323–1331. [CrossRef]
25. Francis, R.L. *A Simple Graphical Procedure to Estimate the Minimum Time to Evacuate a Building*; Society of Fire Protection Engineers: Gaithersburg, MD, USA, 1979.
26. Pursals, S.C.; Garzón, F.G. Optimal building evacuation time considering evacuation routes. *Eur. J. Oper. Res.* **2009**, *192*, 692–699. [CrossRef]
27. Harifi, S.; Mohammadzadeh, J.; Khalilian, M.; Ebrahimnejad, S. Hybrid-EPC: An Emperor Penguins Colony algorithm with crossover and mutation operators and its application in community detection. *Prog. Artif. Intell.* **2021**, *10*, 181–193. [CrossRef]
28. Kumar, P.V.; Marimuthu, C.; Sivaranjani, S.; Punitha, S. Reconstruction of Multi-channel ECG using Compressive Sensing based Emperor Penguin Colony in WBSN. *Int. J. Mech. Eng.* **2021**, *6*, 3.
29. Pinshow, B.; Fedak, M.A.; Battles, D.R.; Schmidt-Nielsen, K. Energy expenditure for thermoregulation and locomotion in emperor penguins. *Am. J. Physiol. -Leg. Content* **1976**, *231*, 903–912. [CrossRef] [PubMed]
30. Du, N.; Fan, J.; Wu, H.; Chen, S.; Liu, Y. An improved model of heat transfer through penguin feathers and down. *J. Theor. Biol.* **2007**, *248*, 727–735. [CrossRef] [PubMed]
31. McCafferty, D.; Gilbert, C.; Paterson, W.; Pomeroy, P.; Thompson, D.; Currie, J.; Ancel, A. Estimating metabolic heat loss in birds and mammals by combining infrared thermography with biophysical modelling. *Comp. Biochem. Physiol. Part A Mol. Integr. Physiol.* **2011**, *158*, 337–345. [CrossRef] [PubMed]
32. Hammel, H.T. Infrared emissivities of some arctic fauna. *J. Mammal.* **1956**, *37*, 375–378. [CrossRef]
33. Harifi, S.; Khalilian, M.; Mohammadzadeh, J.; Ebrahimnejad, S. Optimizing a neuro-fuzzy system based on nature-inspired emperor penguins colony optimization algorithm. *IEEE Trans. Fuzzy Syst.* **2020**, *28*, 1110–1124. [CrossRef]
34. Weisstein, E.W. Logarithmic Spiral. 2002. Available online: https://mathworld.wolfram.com/ (accessed on 23 February 2023).
35. Harifi, S.; Khalilian, M.; Mohammadzadeh, J.; Ebrahimnejad, S. Emperor Penguins Colony: A new metaheuristic algorithm for optimization. *Evol. Intell.* **2019**, *12*, 211–226. [CrossRef]
36. Premalatha, K.; Natarajan, A. Hybrid PSO and GA for global maximization. *Int. J. Open Probl. Compt. Math* **2009**, *2*, 597–608.
37. Kong, X.; Gao, Y.; Wang, T.; Liu, J.; Xu, W. Multi-robot Task Allocation Strategy based on Particle Swarm Optimization and Greedy Algorithm. In Proceedings of the 2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC), Chongqing, China, 24–26 May 2019.
38. Alzaqebah, M.; Jawarneh, S.; Mohammad, R.M.A.; Alsmadi, M.K.; Al-Marashdeh, I.; Ahmed, E.A.; Alrefai, N.; Alghamdi, F.A. Hybrid feature selection method based on particle swarm optimization and adaptive local search method. *Int. J. Electr. Comput. Eng.* **2021**, *11*, 2414. [CrossRef]
39. Arora, S.; Singh, H.; Sharma, M.; Sharma, S.; Anand, P. A new hybrid algorithm based on grey wolf optimization and crow search algorithm for unconstrained function optimization and feature selection. *IEEE Access* **2019**, *7*, 26343–26361. [CrossRef]
40. Ren, X.; Zhang, Z.; Arefzadeh, S.M. An energy-aware approach for resource managing in the fog-based Internet of Things using a hybrid algorithm. *Int. J. Commun. Syst.* **2021**, *34*, e4652.
41. Sadrishojaei, M.; Jafari Navimipour, N.; Reshadi, M.; Hosseinzadeh, M.; Unal, M. An energy-aware clustering method in the IoT using a swarm-based algorithm. *Wirel. Netw.* **2022**, *28*, 125–136. [CrossRef]
42. Zanbouri, K.; Razoughi Bastak, M.; Alizadeh, S.M.; Jafari Navimipour, N.; Yalcin, S. A New Energy-Aware Method for Gas Lift Allocation in IoT-Based Industries Using a Chemical Reaction-Based Optimization Algorithm. *Electronics* **2022**, *11*, 3769. [CrossRef]