Robotica (2025), 1–14
doi:10.1017/S0263574724002200

RESEARCH ARTICLE

Multi-robot movement based on a new modified source-seeking algorithm

Morsy A. Morsy[1,2], Humaid Eqab Al-Otaibi[3], Yasser Bin Salamah[3] and Irfan Ahmad[3]

[1]Department of Electrical Engineering, Shaqra University, Ar. Riyadh, Saudi Arabia
[2]Department of Electronics and Communication Engineering, Ain Shams University, Cairo, Egypt
[3]Department of Electrical Engineering, King Saud University, Ar. Riyadh, Saudi Arabia
Corresponding author: Morsy A. Morsy; Email: morsy_ismail@su.edu.sa

Received: 26 April 2024; **Revised:** 2 November 2024; **Accepted:** 10 December 2024

Keywords: multi-robot system; cooperative control; collision avoidance; source-seeking algorithm; noise cancelation; performance enhancement

Abstract
The two main sources of difficulty for a group of mobile robots employing sensors to find a source are robot collisions and wireless ambient noise, such as light, sound, and other sounds. This paper introduces a novel approach to multi-robot system cooperation and collision avoidance: the new modified source-seeking control with noise cancelation technology. The robot team works together on an incline of a light source field; the team's mobility is dependent upon following the upward gradient's direction and forming a particular movement pattern. The proposed program also takes into account each robot's size, speed limit, obstacles, and noise. The noise cancelation technique has been used to avoid the delay and false decisions to find the target point of the source. When the noise is canceled, all control inputs to the algorithm are accurate, and the feedback decision will be true. In this study, we use the MATLAB simulation tools to test the velocity, position, time delay, and performance of each robot in the used group of robots. The simulation and practical results of the robots in searching for a light source showed very satisfactory performance compared with the results in the literature.

1. Introduction
A collection of autonomous robots that cooperate to accomplish a shared objective is referred to as a multi-robotic system, also known as a multi-robot team or multi-robot system [1, 2]. These systems have attracted a lot of interest in robotics research and development because they can improve a variety of applications' efficiency, scalability, robustness, and flexibility. The ability of robots to coordinate, communicate, and work together allows multi-robotic systems to perform complex tasks that would be challenging or impossible for a single robot to complete. To achieve a shared goal, they are able to collaborate, divide the effort, and share information. The following are some essential features and uses for multi-robotic systems [3, 4]:

Task allocation: In multi-robot systems, tasks are distributed among robots according to their qualifications, available resources, and mission specifications. Algorithms for task allocation seek to give jobs to robots as efficiently as possible while taking into account variables including energy efficiency, robot capabilities, communication limitations, and task complexity.

Cooperation: Multi-robot systems facilitate cooperative problem-solving among robots. To enhance system performance as a whole, they can communicate data, coordinate their actions, and share information. A variety of activities, including work sharing, coordination, formation control, and cooperative sensing, can be included in collaboration.

Exploration and mapping: Tasks involving exploration and mapping frequently make use of multi-robot systems. They can map unknown settings more rapidly and cover greater regions more efficiently when a team of robots is deployed. The robots are able to work together to create maps, exchange sensor data, and update the maps in real time.

Monitoring and surveillance: Applications involving surveillance and monitoring can benefit from multi-robot systems. They can be used in environments, disaster relief efforts, security surveillance, and other areas that call for constant observation. To deliver thorough coverage and prompt response, the robots can disperse, synchronize their movements, and exchange information.

Search and rescue: In the event of a disaster, multi-robot systems may improve search and rescue efforts. Together, the robots may search for survivors, investigate dangerous areas, and carry out various tasks like communication relay, object detection, and mapping. They can lessen the risk to human responders while increasing the efficacy and efficiency of rescue operations.

Swarm robotics: Swarm robotics is a branch of multi-robotic systems that specializes in large-scale systems that draw inspiration from natural swarms, including insect colonies or bird flocks. Swarm robotic systems are usually composed of a large number of basic robots that self-organize and interact locally to demonstrate collective behaviors. Applications for swarm robotics can be found in distributed sensing, pattern generation, and collective transport [5].

Multi-robotic system design and control provide a number of difficulties, including fault tolerance, communication, coordination, path planning, and task allocation. Scholars are formulating formulas and methodologies to tackle these obstacles and unleash the complete capabilities of multi-robotic systems across several fields [6–8].

Source-seeking algorithms (SSAs) for multi-robot applications refer to a set of techniques and strategies used by a group of robots to locate and approach a specific target or source in their environment. These algorithms are designed to enable cooperation and coordination among multiple robots to efficiently search for and converge on the desired source. There are several SSAs commonly used in multi-robot applications. Here are a few examples [9, 10]:

Gradient-based algorithms: These algorithms direct the robots toward a source by using information about the gradient of a field or signal that the source emits. After determining the signal strength, each robot advances in the direction of the strongest signal. The robots can collaborate to increase the precision and effectiveness of source localization by exchanging measurements and modifying their movements together.

Swarm intelligence algorithms: Modeled after the group actions of social insects, swarm intelligence algorithms use several robots working together to find and search for the source. Every robot abides by basic guidelines determined by its interactions and information sharing with nearby robots. Particle swarm optimization, artificial bee colony methods, and ant colony optimization are a few examples.

Algorithms for distributed exploration: By distributing the search space among the robots, these algorithms seek to efficiently cover it. Every robot investigates a certain area, and they periodically communicate with one another regarding sources they have found or haven't yet examined. This makes it possible for the robots to more successfully converge on the source and jointly create a map of the surroundings.

Algorithms based on communication: In these algorithms, information about the source location is exchanged between robots through communication. To increase the precision and speed of source localization, the robots can exchange measurements, estimated source positions, and other pertinent data. Wireless communication, ad hoc networks, and other methods can all be used to accomplish communication.

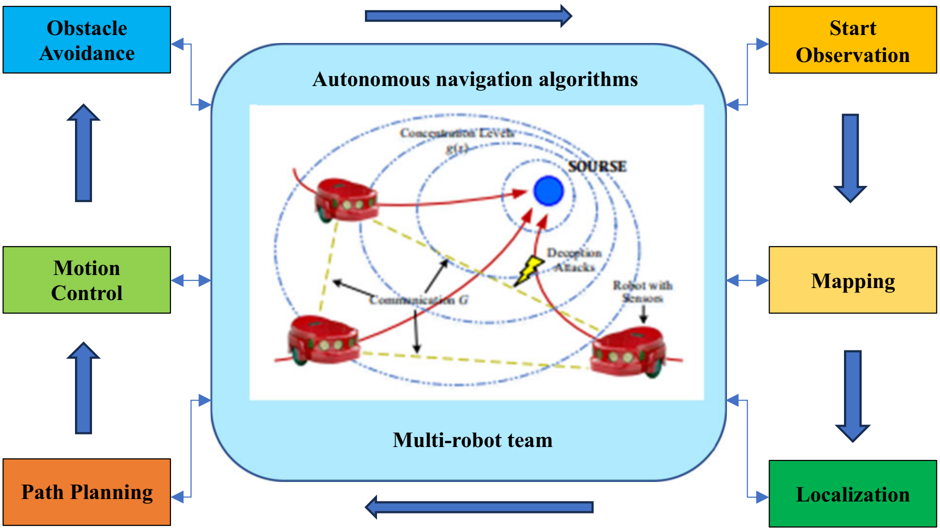

Figure 1. *The common modular design for mobile robot autonomous navigation.*

Potential field algorithms: They represent the source as an alluring potential and the environment as a potential field. Every robot moves through the field by following the potential gradient, and by minimizing a potential function, they converge at the source. Cooperative source finding can be achieved by combining potential field algorithms with communication-based methods.

These are a few instances of SSAs used in applications involving multiple robots. The type of source, the surroundings, the robots' capabilities, and the particular needs of the application all play a role in the algorithm selection. In order to increase source seeking's efficacy and efficiency in a variety of settings, multi-robot system engineers and researchers are always creating and improving algorithms. The following modules/subsystems make up the most widely used system design for multi-mobile robot autonomous navigation, which is shown in Figure 1. The multi-robot movements, obstacle avoidance, mapping, and path planning are all managed by the autonomous navigation algorithms.

Mobile robots are controlled using an intelligent control method called extremum seeking control (ESC). ESC can be used in a multi-robot system to arrange the robots such that they work together to find and maximize a certain target or source. An outline of the application of ESC in a multi-robot system is provided below: define an objective function that accurately represents the performance metric you wish to optimize. This function could stand in for the distance to the source, the concentration of a material, or any other pertinent metric in the context of multi-robot source seeking. A control parameter, or a group of control parameters, which affects how the robots behave, is perturbed.

The robots' activities change as a result of this parameter perturbation, which affects their capacity to accomplish the goal. Measurement and feedback: the multi-robot system's performance is assessed using the specified objective function. In further iterations, the control parameter is adjusted based on this feedback. Every robot has sensors to sense its surroundings and identify the features of the source [11]. Cameras, gas detectors, microphones, or specialized sensors are examples of sensors, depending on the source (chemical, acoustic, or visual). Robots must interact in order to exchange data regarding the location of the source, their surroundings, and their own positions [12]. In addition to preventing needless investigation, this communication facilitates cooperative decision-making. Robots must accurately map their surroundings and understand their own locations in order to perform localization and mapping.

Modern methods such as simultaneous localization and mapping guarantee that robots always know where they are. Algorithms for path planning and exploration are required so that robots can efficiently explore their surroundings and converge near the source. The ratio of exploration to exploitation should be balanced along these routes [13–15]. Distribution control: every robot decides what to do based on information from its surroundings and interactions with other robots. Without a central controller, coordination is ensured using consensus algorithms and distributed control mechanisms. Collaborative decision-making: robots can decide on the best course of action for the group by voting, negotiating, or using consensus algorithms. By utilizing swarm intelligence concepts, robots can collectively display emergent behaviors that improve the efficiency of source seeking.

It is possible to use methods like particle swarm optimization or ant colony optimization. Algorithms for obstacle avoidance and collision avoidance are crucial in ensuring that robots avoid obstacles and avoid collisions when approaching the source. Resource management is essential for ensuring the lifetime of multi-robot operations, including controlling resources like battery life and communication bandwidth. Adaptation and learning: in order to improve source-seeking performance over time, the control system can integrate learning algorithms to modify methods in response to the robots' experiences [16–19]. Assigning tasks and roles to specific robots can optimize coverage of the environment and prevent redundancy. Roles can also be assigned to individual robots.

Through sensor fusion, robots can fuse data from multiple sensors to improve the accuracy of source localization and reduce uncertainties. Scalability and robustness: the system should be scalable to handle varying numbers of robots and robust enough to handle dynamic environments and changing source conditions [20–23].

Numerous attempts have been documented in the literature to address the issue of noise in multi-robot systems induced by light, sound, and other sources in wireless environments [6–10]. These sounds impair system effectiveness and performance and lengthen the robots' search time for the target source [11, 16]. Moreover, the results that have been published do not take these noises into account. There have been some intriguing findings regarding this issue, which are crucial for designing the SSAs in multi-robot systems [17–23].

This study aims to design a new modified source-seeking control (NM-SSC) algorithm to improve the performance, stability, and time delay between the target source point and the start point of a multi-robot system. The purpose behind this effort is explained above. Additionally, we provide a balanced detector and light sensor noise cancelation method. In comparison to previous systems in the literature, all considered robots may eventually move to the target source place efficiently with least time delay by employing the noise cancelation technique in the modified algorithm.

Simulation results demonstrate the usefulness of the suggested noise cancelation technique and improved algorithms. In order to prevent robot collisions when many robots approach the source point simultaneously, the NM-SSC algorithm has an extra two steps above the ESC method. This paper's reminder is created as follows: the suggested NM-SSC method for the multi-robot model is introduced in Section 2. Section 3 then presents the simulation findings and discussions. In Section 4, the work's conclusions are demonstrated.

2. The proposed model

In this section, we will discuss the proposed model steps to reach the source of mult-robots based on the NM-SSC technique as a smart control approach for moving multi-robots.

2.1. System description

The multi-robot model for an efficient and stable multi-vehicle system is a mathematical representation of the dynamics of this team of vehicles that can move randomly in any direction and has a multiple degree of freedom for its vehicle orientation. The proposed model contains a set of matrices based on nonlinear equations that describe the dynamics of the multi-vehicle motion. These equations include the position, velocity, orientation, and noise level of each vehicle. To find the optimal control inputs for the

https://doi.org/10.1017/S0263574724002200 Published online by Cambridge University Press

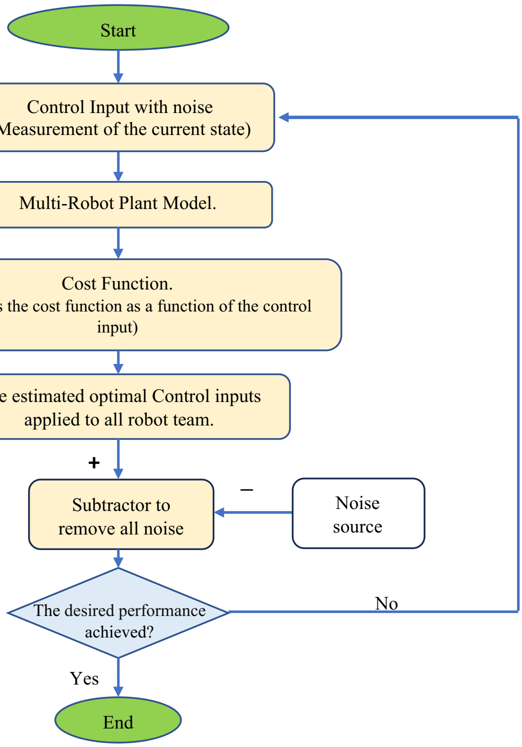

Figure 2. *The NM-SSC operation steps.*

multi-vehicle model to achieve the desired performance, we use the NM-SSC algorithm. The structure of the proposed NM-SSC algorithm is represented by the flowchart of Figure 2 and introduced into eight steps as shown in Table I.

2.2. New modified source-seeking control algorithm (NM-SSC)

The proposed NM-SSC technique is the ESC algorithm plus a noise cancelation function, which is used to track each robot in the team and consequently find the optimum operating point. This leads to minimize or maximize the system performance function. The NM-SSC algorithm works based on offering a small periodic perturbation signal without any noise and uses the system feedback controller response to adjust the control inputs with the existence of noise to eliminate it and achieve the optimum desired performance.

The mathematical model of the NM-SSC algorithm for the team of multi-vehicle can be represented by a nonlinear set of differential equations as in refs. [1–5] plus some modification parameters to avoid the noise and delay. Therefore, each robot in the team of multi-robot has position coordinate parameters $x'(t)$, $y'(t)$ and heading angle θ'. These parameters are used to represent the robot dynamics as follows [5]:

$$x'(t) = v * \cos(\theta) \tag{1}$$

$$y'(t) = v * \sin(\theta) \tag{2}$$

$$\theta'(t) = \omega_o \tag{3}$$

https://doi.org/10.1017/S0263574724002200 Published online by Cambridge University Press

Table I. *Steps description of the NA-SSC algorithm.*

Step#	Step name	Description
1	State measurement	Each robot i in the group of n robot measures its position and velocity $x_i(t)$ and $v_i(t)$, respectively, and then sends the measured position to the neighbors' robots and uses these values as the inputs to the NA-ESC algorithm.
2	Collision avoidance	The $j \in n$ neighbors' robot sends signal $x_j^i(t)$ to the robot i to prevent the collision, if the value of this signal less than some thresholds $x_{th}$of these values are stringently larger than the position value of the robot i. Otherwise, the robot i ignores the x_{th} largest values.
3		Repeat step 2 for the values smaller than $x_i(t)$.
4	Plant model	Predict the next state of each vehicle based on the mathematical model of multi-vehicle dynamics and the current control inputs.
5	Cost function	Define the cost function that represents the optimum system performance and use the NA-ESC algorithm to determine the optimal control inputs that can minimize this function.
6	Control inputs	The NA-ESC generates the optimum control inputs and feeds the vehicles to achieve the optimum performance.
7	Noise subtraction and cancelation	The noise is measured and fed as a control input to the algorithm, and the algorithm subtraction function subtracts it from the control input that has the same amount of noise to remove it. This step is important for choice the true decision without noise.
8	Iteration	Repeat the above five steps based on the new measurements as a new control input.

where v is the robot forward velocity and ω_o is the robot angular velocity. Then the team of robots' model can be described by the following matrices:

$$\begin{bmatrix} x_1'(t) \\ x_2'(t) \\ \vdots \\ x_{n-1}'(t) \\ x_n'(t) \end{bmatrix} = \begin{bmatrix} v_1 \sin(\theta_1) \\ v_2 \sin(\theta_2) \\ \vdots \\ v_{n-1} \sin(\theta_{n-1}) \\ v_n \sin(\theta_n) \end{bmatrix} \tag{4}$$

$$\begin{bmatrix} y_1'(t) \\ y_2'(t) \\ \vdots \\ y_{n-1}'(t) \\ y_n'(t) \end{bmatrix} = \begin{bmatrix} v_1 \cos(\theta_1) \\ v_2 \cos(\theta_2) \\ \vdots \\ v_{n-1} \cos(\theta_{n-1}) \\ v_n \cos(\theta_n) \end{bmatrix} \tag{5}$$

$$\begin{bmatrix} \theta_1'(t) \\ \theta_2'(t) \\ \vdots \\ \theta_{n-1}'(t) \\ \theta_n'(t) \end{bmatrix} = \begin{bmatrix} \omega_{01} \\ \omega_{02} \\ \vdots \\ \omega_{0n-1} \\ \omega_{0n} \end{bmatrix} \tag{6}$$

https://doi.org/10.1017/S0263574724002200 Published online by Cambridge University Press

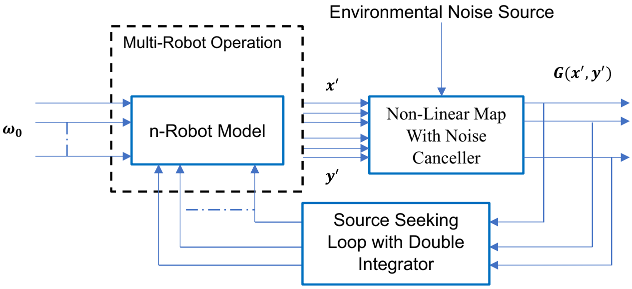

***Figure 3.** Multi-robot team NM-SSC model.*

The NM-SSC algorithm computes the control inputs of each robot in the team and uses a reference sinusoidal signal of these inputs to search for the optimal point of the source. The NM-SSC algorithm can be represented by the following mathematical equation:

$$\begin{bmatrix} \omega_1(t) \\ \omega_2(t) \\ \vdots \\ \omega_{n-1}(t) \\ \omega_n(t) \end{bmatrix} = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_{n-1}(t) \\ u_n(t) \end{bmatrix} + \begin{bmatrix} A_1 \sin(2\pi f_1 t + \varnothing_1) \\ A_2 \sin(2\pi f_2 t + \varnothing_2) \\ \vdots \\ A_{n-1} \sin(2\pi f_{n-1} t + \varnothing_{n-1}) \\ A_N \sin(2\pi f_n t + \varnothing_n) \end{bmatrix} \tag{7}$$

where $u_1(t)$ to $u_n(t)$ and $A_1 \sin(2\pi f_1 t + \varnothing_1)$ to $A_1 \sin(2\pi f_1 t + \varnothing_1)$ are the control inputs and the modified reference signals of robot number 1 to robot number n, respectively. Here, the modified reference signals are not constant like in the ESC. Therefore, these signals are based on the modified amplitude, frequency, and phase A_i, f_i, and $\varnothing_i$ parameters, respectively, which are varied according to the control inputs variations. Based on (7), we can represent the multi-robot team model with the NM-SSC algorithm by a set of nonlinear differential Eqs. (4), (5), and (8).

$$\begin{bmatrix} \theta'_1(t) \\ \theta'_2(t) \\ \vdots \\ \theta'_{n-1}(t) \\ \theta'_n(t) \end{bmatrix} = \begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_{n-1}(t) \\ u_n(t) \end{bmatrix} + \begin{bmatrix} A_1 \sin(2\pi f_1 t + \varnothing_1) \\ A_2 \sin(2\pi f_2 t + \varnothing_2) \\ \vdots \\ A_{n-1} \sin(2\pi f_{n-1} t + \varnothing_{n-1}) \\ A_N \sin(2\pi f_n t + \varnothing_n) \end{bmatrix} \tag{8}$$

Figure 3 shows the modified extremum seeking scheme based on the extremum seeking loop, which is used to modify the objective function $q_i = g(x_{s_i}, y_{s_i})$ to optimize the value of the forward velocity of robot number i. Also, the extremum seeking loop must ensure that the control input $[x_{c_i}, y_{c_i}]$ converges asymptotically toward the maximizer $[x_i^*, y_i^*]$.

The nonlinear map controller is used to drive the system to the desired optimum position coordinate and cancel the noise coming from the multi-robot model from the environmental noise, and the output objective function can be expressed as in Eq. (9).

$$\begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_{n-1} \\ q_n \end{bmatrix} = \begin{bmatrix} g_1(x_1, y_1) \\ g_2(x_2, y_2) \\ \vdots \\ g_{n-1}(x_{n-1}, y_{n-1}) \\ g_n(x_n, y_n) \end{bmatrix} = \begin{bmatrix} G_1^* \\ G_2^* \\ \vdots \\ G_{n-1}^* \\ G_n^* \end{bmatrix} - \begin{bmatrix} d_{x_1}(x_1 - x_1^*) \\ d_{x_2}(x_2 - x_2^*) \\ \vdots \\ d_{x_{n-1}}(x_{n-1} - x_{n-1}^*) \\ d_{x_n}(x_n - x_n^*) \end{bmatrix} - \begin{bmatrix} d_{y_1}(y_1 - y_1^*) \\ d_{y_2}(y_2 - y_2^*) \\ \vdots \\ d_{y_{n-1}}(y_{n-1} - y_{n-1}^*) \\ d_{y_n}(y_n - y_n^*) \end{bmatrix} \tag{9}$$

where d_{x_i} and d_{y_i} are unknown constants for the robot number i.

https://doi.org/10.1017/S0263574724002200 Published online by Cambridge University Press

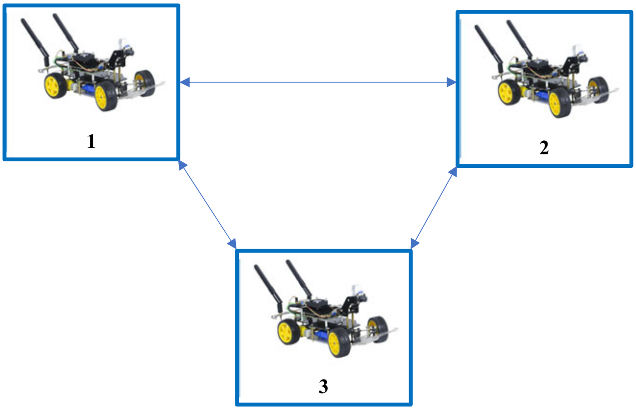

Figure 4. *Three-robot communication network.*

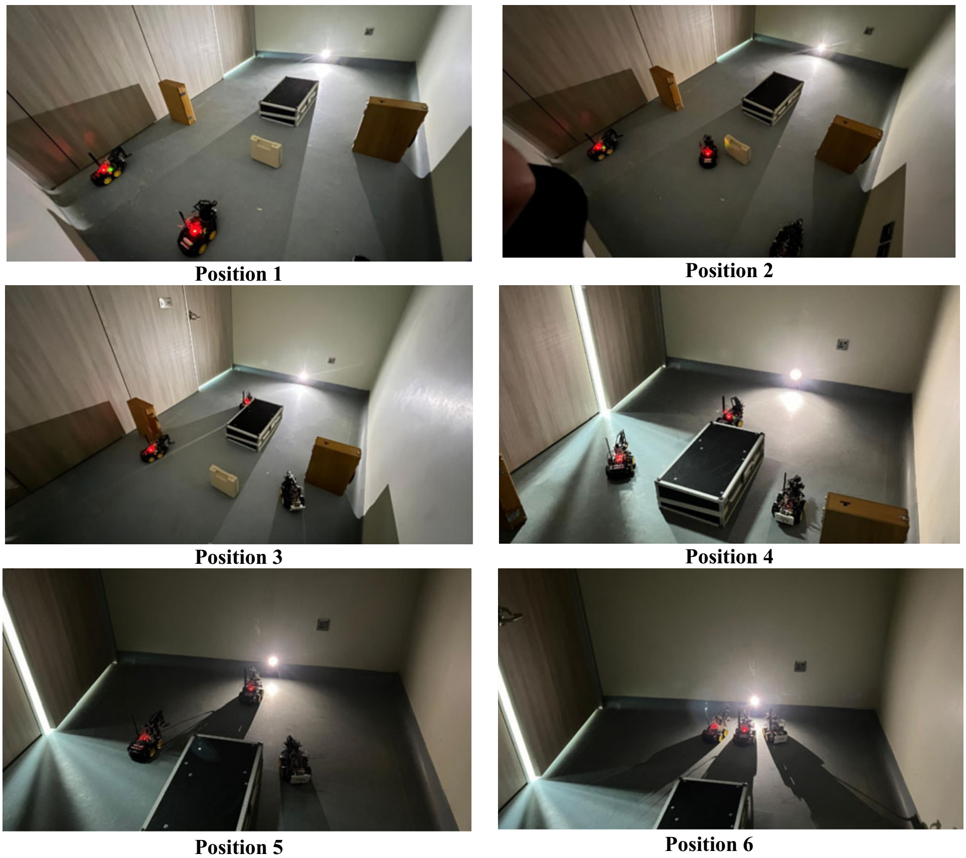

Figure 5. *Different position for the multi-robot movement from the start position to the source position.*

3. Simulation and practical results and discussion

In this section, we are providing a simulation result of three-robot team to validate the proposed NM-SSC algorithm. The group of three robots are communicating as in Figure 4, assuming random initial position and velocities from $[-1.5, 1.5]$ m and $[-1, 1]$ m/s, respectively. In addition, we assume $\omega_0 = 1$ rad/s for all robots and the reference signal is the same in all robots and equal to $(0.5 + 1.5\sin(\pi t))$. Also, Figure 5 presents multi-robot movement at different positions from the starting point to the light

https://doi.org/10.1017/S0263574724002200 Published online by Cambridge University Press

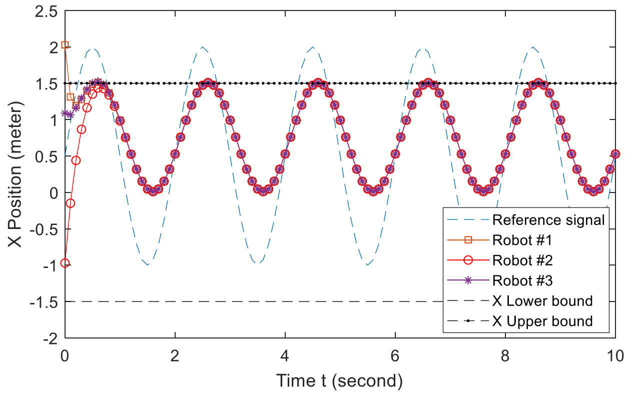

Figure 6. *The positions of three-robot group versus time when applying the NM-SSC algorithm without steps 2 and 3 in Table I.*

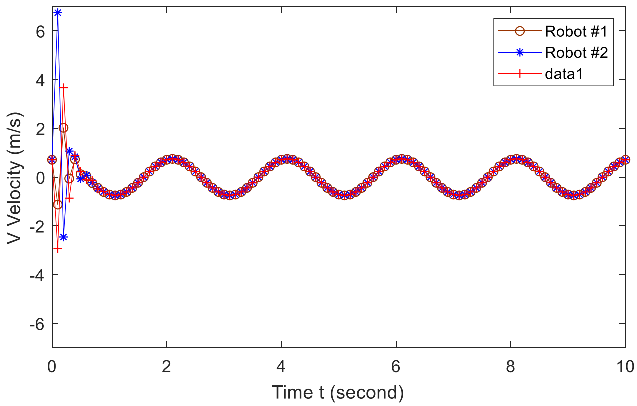

Figure 7. *The measured velocities of three-robot group versus time using the NM-SSC algorithm without applying steps 2 and 3 in Table I.*

source. We put four obstacles to show how the robots avoid these obstacles and to minimize the time that used to avoid it. We keep the door of the room open 5 cm to see the effect of the environment light noise on the robot movement.

3.1. Movement scenario

Position 1: this is the starting position where the robots take a decision to start movement. Every robot has a certain path according to its initial position.

Position 2: the robots start movement and communicate to decide how each one can avoid the obstacle.

Position 3: all robots avoid the obstacles and move in the right path to the source before the fourth obstacle.

Position 4: all robots take different paths to avoid the fourth obstacle (the big box) in the room.

Position 5: each robot takes the final decision to reach the light source.

Figures 6 and 7 illustrate the simulated positions and measured velocities versus time, respectively, for three robots in case of all communications without applying steps 2 and 3 in Table I. The result shows that without applying steps 2 and 3, all robots follow the reference signal within the theoretical safety condition represented by the dash black lines at 1.5 and −1.5. Also, we can see that the system

https://doi.org/10.1017/S0263574724002200 Published online by Cambridge University Press

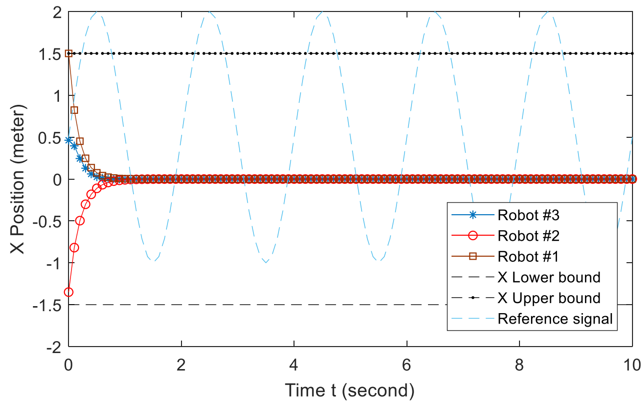

Figure 8. *The positions of three-robot group versus time using the NM-SSC algorithm with applying steps 2 and 3 in Table I.*

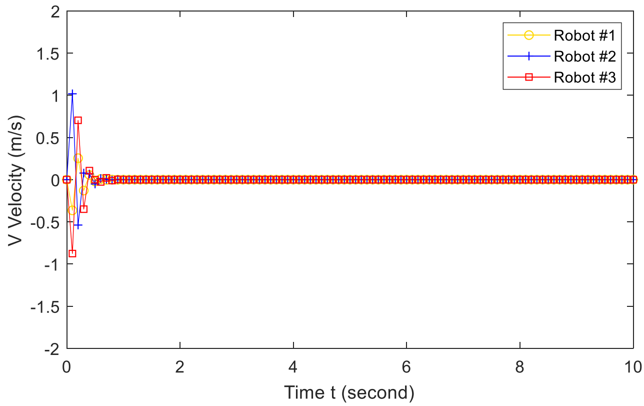

Figure 9. *The measured velocities of three-robot group versus time using the NM-SSC algorithm with applying steps 2 and 3 in Table I.*

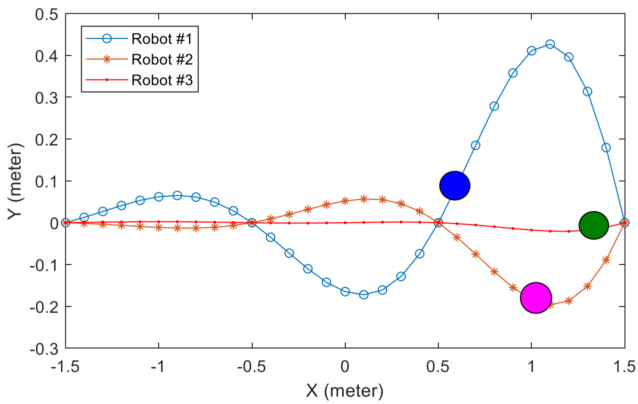

Figure 10. *Collision free of three-robot movement without using steps 2 and 3 in Table I.*

https://doi.org/10.1017/S0263574724002200 Published online by Cambridge University Press

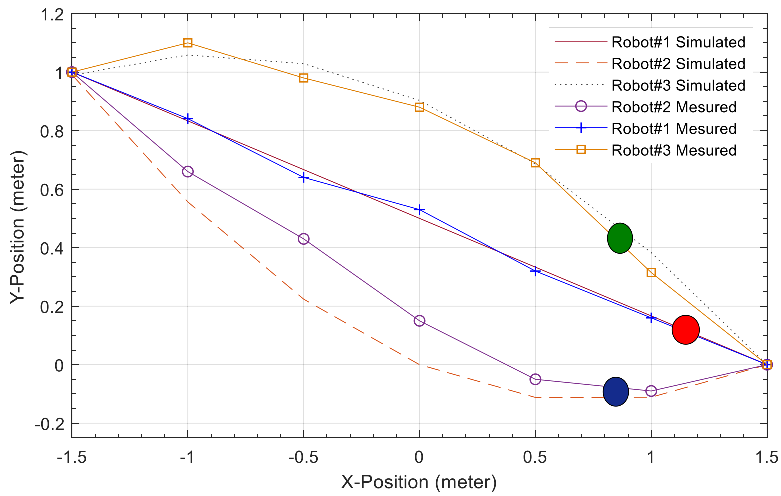

Figure 11. *Collision free of three-robot movement using steps 2 and 3 in Table I.*

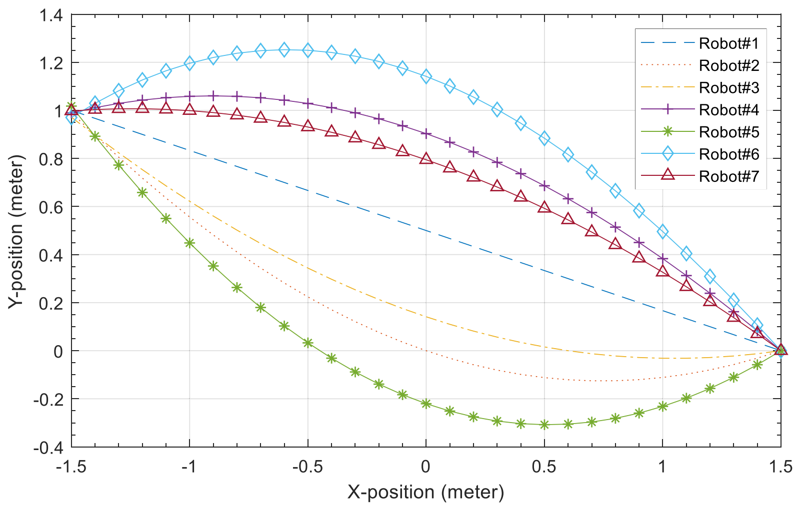

Figure 12. *Collision free of seven robots' movement using steps 2 and 3 in Table I.*

can achieve the resilient consensus by applying the other steps. On the other hand, Figures 8 and 9 illustrate the simulated positions and measured velocities versus time, respectively, for the three robots with applying steps 2 and 3. In this case, we can observe that all robots tend to the center point of the source [0, 0] and also the resilient consensus is achieved and the target has been achieved by the system. In all results, we can see that the time needed to achieve the resilient consensus does not exceed 1 s.

Figure 10 presents the three-robot movement without using steps 2 and 3 in Table I. These results show that the robots move without collision or collision free, but the movement for each robot is random, depending on the presence of the reference signal when the desired robot need to moves.

This causes the robot to take more time compared to the straight path from the start point to the source or target point. The second problem is that there is no minimum distance condition between robots, and as a result, the collision can occur. When steps 2 and 3 are considered, each robot moves in a desired path from the start point to the source without overlapping as shown in Figure 11. Also, each path has different time delay compared to the other paths, and this make sure no collision will occur.

https://doi.org/10.1017/S0263574724002200 Published online by Cambridge University Press

https://doi.org/10.1017/S0263574724002200 Published online by Cambridge University Press

Table II. *Comparison between the proposed word and literature.*

Ref. #	Developed algorithm	Objective	Type of model	Results and number of robots	steady state time
[1]	Modified gravitational search algorithm and dynamic window approach	Three different neural oscillators are chosen and applied to the path planning	Dynamic	Simulation and experiment of multi-robot in different ranges	30% improvement in time 11 s
[8]	Point measurement algorithm	Correct the wrong movement caused by the sensor noise	Dynamic	Simulation of two robots in the range of 10 m versus time	Time is variable Average ≅ 40 s
[13]	Distributed algorithms for stochastic source seeking with all to all communication	Guide the robot movement by using the finite-difference scheme to correctly estimate the signal gradient	Dynamic	Simulation of five robots in the range of 4.62 m	Time NA. But the number of iterations from 30 to 50
[14]	Source seeking of all to all communication and limited communication	Guide the robot movement by gradient estimation technique	Dynamic	Simulation and experiment of five robots in the range of 2 m versus time.	20 s
[15]	Resilient cooperation control (RCC)	Develop the RCC algorithm suffering from local deception attack	Dynamic	Simulation and experiment of 7 and 15 robots in the range of 2 m versus time	3.5 s
Proposed work	New modified source-seeking control (NM-SSC)	Develop the NM-ESC algorithm with noise canceler, resilient consensus and all to all communication	Dynamic	Simulation experiment of three robots in the range of 3 m versus time	≅ 0.8 s evaluated and 1.1 s measured

The result in Figure 11 also compares between the simulation results and the experimental results of the three-robot movements. Only the experimental result of robot #2 outperforms the simulation result due to the decision path defined by the simulation parameters when robot #2 starts movement. However, the experimental movement updated every moment according to the sensors mesuremnts. For robots #1 and 2, the simulation results match with the experimental results.

To address scalability, we have increased the number of robots in the simulation, demonstrating the algorithm's performance under larger team conditions. This provides preliminary insights into its scalability and effectiveness in more complex scenarios, though we recognize that further real-world testing would enhance the understanding of its practical applications. Figure 12 presents seven robots' movements under applying steps 2 and 3 in Table I. The results showed that only one robot moved from the starting point to the source in a straight line, while three of the robots took slower paths at the beginning and faster paths near the light source. The remaining three robots, however, took faster paths at the beginning and slower paths at the end before the source. This is to avoid any collisions, and at the same time, they complete the journey in very similar times. When returning to the starting point, each robot follows the path that complements the shape of the eye, as shown in the figure (Table II).

4. Conclusion

In this paper, the new modified extremum seeking control (NM-SSC) for multi-robot team cooperation based on light sensors has developed. For the performance criterion measurements and the information provided to the NM-SSC algorithm, we use the light sensors for updating the control inputs. The light sensors are also used for environmental noise cancelation to prevent the error that can occur in the control inputs provided to the algorithm, which improves the system performance in terms of time delay, movement error, and optimal source seeking. The proposed algorithm also works on the distance measurement between robots and uses the feedback to prevent the collision between robots. Additionally, we use the measured distance between robots to control position and velocity of each robot to reduce the time delay between the transient response and steady-state response in the presence of reference signal. All results focus on comparing measurements using the ESC algorithm with those from the proposed algorithm to investigate the differences and present solutions to the challenges faced by the multi-robot team.

Availability of data and materials. This declaration is not applicable.

Author contributions. The authors confirm their contributions to the paper as follows: study conception and design: Humaid Eqab, M. A. Morsy, and Yasser Bin Salamah; data collection: Humaid Eqab and M. A. Morsy; analysis and interpretation of results: Humaid Eqab, M. A. Morsy, Yasser Bin Salammah, and Irfan Ahmad; draft manuscript preparation: M. A. Morsy, Yasser Bin Salamah, and Irfan Ahmad. All authors reviewed the results and approved the final version of the manuscript.

Funding. The research was funded by King Saud University through Researchers Supporting Project number (RSP2024R270), King Saud University, Riyadh, Saudi Arabia.

Competing interests. The authors declare that there's no conflict of interest.

Ethical approval. This declaration is not applicable.

References

[1] P. D. R. Vikas, "Multi-robot path planning using a hybrid dynamic window approach and modified chaotic neural oscillator-based hyperbolic gravitational search algorithm in a complex terrain," *Intel. Serv. Robotics* **16**(2), 213–230 (2023). doi: 10.1007/s11370-023-00460-y.

[2] H. M. La and W. Sheng, "Distributed sensor fusion for scalar field mapping using mobile sensor networks," *IEEE Trans. Cybern.* **43**(2), 766–778 (2013).

- [3] Z. Liu, J. Wang, J. Li, P. Liu and K. Ren, "A novel multiple targets detection method for service robots in the indoor complex scenes," *Intel. Serv. Robotics* **16**(4), 453–469 (2023). doi: 10.1007/s11370-023-00471-9.
- [4] S. Secil and M. Ozkan, "A collision-free path planning method for industrial robot manipulators considering safe human-robot interaction," *Intel. Serv. Robotics* **16**(3), 323–359 (2023). doi: 10.1007/s11370-023-00465-7.
- [5] H. Eqab, Y. B. Salamah, I. Ahmad and M. A. Morsy, "Development of source seeking algorithm for mobile robots," *Intel. Serv. Robotics* **16**(3), 393–401 (2023). doi: 10.1007/s11370-023-00470-w.
- [6] A. Basiri, V. Mariani, G. Silano, M. Aatif, L. Iannelli and L. Glielmo, "A survey on the application of path-planning algorithms for multi-rotor UAVs in precision agriculture," *J. Navig.* **75**(2), 364–383 (2022). doi: 10.1017/S0373463321000825.
- [7] A. D. Sabiha, M. A. Kamel and E. Said, "Hussein WM real-time path planning for autonomous vehicle based on teaching-learning-based optimization," *Intell. Serv. Robot.* **15**, 381–398 (2022). doi: 10.1007/S11370-022-00429-3/FIGURES/19.
- [8] Y. Elor and A. M. Bruckstein, "Two-robot source seeking with point measurements," *Theor. Comput. Sci.* **457**, 76–85 (2012). doi: 10.1016/j.tcs.2012.06.036.
- [9] Vikas, D. R. Parhi and A. K. Kashyap, "Humanoid robot path planning using memory-based gravity search algorithm and enhanced differential evolution approach in a complex environment," *Expert. Syst. Appl.* **215**, 119423 (2023). doi: 10.1016/J.ESWA.2022.119423.
- [10] S. Mirjalili and A. Lewis, "Adaptive gbest-guided gravitational search algorithm," *Neural Comput. Appl.* **25**, 1569–1584 (2014). doi: 10.1007/S00521-014-1640-Y/TABLES/10.
- [11] H. Xu, S. Jiang and A. Zhang, "Path planning for unmanned aerial vehicle using a mix-strategy-based gravitational search algorithm," *IEEE Access* **9**, 57033–57045 (2021). doi: 10.1109/ACCESS.2021.3072796.
- [12] T. A. Khan and S. H. Ling, "A novel hybrid gravitational search particle swarm optimization algorithm," *Eng. Appl. Artif. Intell.* **102**, 104263 (2021). doi: 10.1016/J.ENGAPPAI.2021.104263.
- [13] N. A. Atanasov, J. Leny and G. J. Pappas, "Distributed algorithms for stochastic source seeking with mobile robot networks," *J. Dyn. Syst. Meas. Control* **137**(3), 031004 (2015).
- [14] S. Li, R. Kong and Y. Guo, "Cooperative distributed source seeking by multiple robots: Algorithms and experiments," *IEEE/ASME Trans. Mechatron* **19**(6), 1810–1820 (2014). doi: 10.1109/TMECH.2013.2295036.
- [15] W. Fu, J. Qin, W. X. Zheng, Y. Chen and Y. Kang, "Resilient cooperative source seeking of double-integrator multi-robot systems under deception attacks," *IEEE Trans. Ind. Electron.* **68**(5), 4218–4227 (2021). doi: 10.1109/TIE.2020.2987270.
- [16] E. Dönmez, A. F. Kocamaz and M. Dirik, "A vision-based real-time mobile robot controller design based on Gaussian function for indoor environment," *Arab. J. Sci. Eng.* **43**(12), 7127–7142 (2018). doi: 10.1007/s13369-017-2917-0.
- [17] E. Dönmez and A. F. Kocamaz, "Design of mobile robot control infrastructure based on decision trees and adaptive potential area methods," *Iran. J. Sci. Technol. Trans. Electr. Eng.* **44**(1), 431–448 (2020). doi: 10.1007/s40998-019-00228-0.
- [18] K. M. Lynch and F. C. Park. *Modern Robotics: Mechanics, Planning, and Control*. 1st edition, Cambridge University Press, Cambridge, MA, (2017).
- [19] M. Dirik, F. Kocamaz and E. Dönmez, "Visual servoing based control methods for non-holonomic mobile robot," *J. Eng. Res.* **8**(2), (2020).
- [20] W. Sun and Y. Tan, "Model guided extremum seeking control of electromagnetic micromirrors," *Sci. Rep.* **11**(1), 1–15 (2021). doi: 10.1038/s41598-021-97098-6.
- [21] M. Diachuk and S. M. Easa, "Improved technique for autonomous vehicle motion planning based on integral constraints and sequential optimization," *Vehicles* **4**(4), 1122–1157 (2022). doi: 10.3390/VEHICLES4040060.
- [22] D. González, J. Pérez, V. Milanés and F. Nashashibi, "A review of motion planning techniques for automated vehicles," *IEEE Trans. Intell. Transp. Syst* **17**(4), 1135–1145 (2016). doi: 10.1109/TITS.2015.2498841.
- [23] J. Betz, H. Zheng, A. Liniger, U. Rosolia, P. Karle, M. Behl, V. Krovi and R. Mangharam, "Autonomous vehicles on the edge: A survey on autonomous vehicle racing," *IEEE Open J. Intell. Transp. Syst.* **3**, 458–488 (2022). doi: 10.1109/OJITS.2022.3181510.

Cite this article: M. A. Morsy, H. Eqab Al-Otaibi, Y. Bin Salamah and I. Ahmad, "Multi-robot movement based on a new modified source-seeking algorithm", Robotica. https://doi.org/10.1017/S0263574724002200