14

Glowworm Swarm Optimization - Modifications and Applications

Krishnanand Kaipa

Department of Mechanical and Aerospace Engineering
Old Dominion University, Norfolk, Virginia, USA

Debasish Ghose

Department of Aerospace Engineering
Indian Institute of Science, Bangalore, India

CONTENTS

14.1 Introduction

Glowworm Swarm Optimization (GSO) is a unique swarm intelligence algorithm that aims to capture all the local optima rather than just the global optimum as most other swarm intelligence algorithms do. There are several applications where this objective makes sense [1, 2, 3]. For instance, the problem of identifying multiple sources of signals in the environment needs each signal source to be identified. As it is more than likely that the location of each signal source will correspond to a local optimum, while the global optimum will merely correspond to the strongest signal source, searching for the local optimum will lead to a satisfactory solution. Another example where searching for local optima is desired is when the global optimum might be too costly to implement and a local optimum, which may be less optimum in terms of some performance measure could be a cheaper alternative. A third possibility is when the set of optima are contiguous points on the search space. This can happen when one is looking for the boundary of, say, a level set. In this chapter, many of these classes of engineering applications, which also require modifications in the basic GSO that become necessary due to the unique requirement of the application, are presented [4, 5, 6]. The chapter also presents a few modifications that show very innovative application of GSO to clustering and wireless network applications.

14.2 Brief description of GSO

GSO starts by placing a population of n glowworms randomly in the search space. Each cycle of the algorithm consists of a luciferin update phase, a movement phase, and a neighborhood range update phase. A detailed description of the original GSO algorithm was described in the chapter 14 of the companion tutorial book on swarm intelligence algorithms [19]. The equations representing the steps of GSO are repeated here.

1. Luciferin update:

$$\ell_i(t+1) \quad = \quad (1-\rho)\ell_i(t) + \gamma J(x_i(t+1)) \tag{14.1}$$

where $\ell_i(t)$ represents the luciferin level associated with glowworm i at time t, ρ is the luciferin decay constant $(0 < \rho < 1)$, γ is the luciferin enhancement constant and $J(x_i(t))$ represents the value of the objective function at agent i's location at time t.

2. Movement update: For each glowworm i, the probability of moving toward a neighbor j is given by

$$p_{ij}(t) \quad = \quad \frac{\ell_j(t) - \ell_i(t)}{\sum_{k \in N_i(t)} \ell_k(t) - \ell_i(t)} \tag{14.2}$$

where $j \in N_i(t)$ and

$$N_i(t) = \{j : d_{ij}(t) < r_d^i(t) \text{ and } \ell_i(t) < \ell_j(t)\} \tag{14.3}$$

is the set of neighbors of glowworm i at time t, $d_{ij}(t)$ represents the Euclidean distance between glowworms i and j at time t, and $r_d^i(t)$ represents the variable neighborhood range associated with glowworm i at time t. Let glowworm i select a glowworm $j \in N_i(t)$ with $p_{ij}(t)$ given by (14.2). Then, the movement update of each glowworm is given by

$$x_i(t+1) = x_i(t) + s\left(\frac{x_j(t) - x_i(t)}{\|x_j(t) - x_i(t)\|}\right) \tag{14.4}$$

where $x_i(t) \in R^m$ is the location of glowworm i, at time t, in the $m-$dimensional real space R^m, $\|\cdot\|$ represents the Euclidean norm operator, and s (> 0) is the step size.

3. Neighborhood range update phase: To adaptively update the neighborhood range of each glowworm, the following rule is applied:

$$r_d^i(t+1) = \min\{r_s, \max\{0, r_d^i(t) + \beta(n_t - |N_i(t)|)\}\} \tag{14.5}$$

where β is a constant parameter and n_t is a parameter used to control the number of neighbors.

14.3 Modifications to GSO formulation

14.3.1 Behavior switching modification

Localization of sources using mobile robot swarms has received considerable attention in the collective robotics community. Examples of such sources include sound, heat, light, leaks in pressurized systems, hazardous plumes/aerosols resulting from nuclear/chemical spills, fire-origins in forest fires, deep-sea hydrothermal vent plumes, hazardous chemical discharge in water bodies, oil spills, etc. This problem has also been recognized as one that involves significant risks to humans. Most research in this area has dealt with single sources, and relatively less research effort has been devoted to multiple source localization [7]. The problem is compounded when there are multiple sources. In all the above situations, there is an imperative need to simultaneously identify and neutralize all the sources before the emissions cause harm to the environment and people in the vicinity. In addition to this, mapping of the contaminant boundary facilitates a rapid planning effort to move people and valuable property out of the affected region [8].

The problem of identifying multiple source location should deal with the issue of how to automatically partition the robots into subgroups in order to ensure that each source is captured by one of the subgroups. Thomas and Ghose [9, 10] proposed a swarm algorithm that intelligently combines chemotactic, anemotactic, and spiralling behaviors in order to locate multiple odor sources. The chemotactic behavior was achieved by using GSO. Agents switch between the three behaviors based on the information available from the environment for optimal performance. The proposed algorithm was achieved by incorporating the following modifications into the GSO framework. In turbulent flows, the peak concentration value within a patch and the frequency of encountering a patch increase as the glowworm gets closer to the source. However, the instantaneous value might mislead the movement decision of the glowworms. Therefore, the authors defined the luciferin of each glowworm to be the maximum odor concentration encountered in the last N_{mem} seconds in its trajectory. This change was seen to improve algorithmic performance significantly. Accordingly, the luciferin update equation was modified as below:

$$\ell_i(t) \quad \leftarrow \quad \max\{C(x_i(t - N_{mem} + 1)), \ldots, C(x_i(t))\} \tag{14.6}$$

where, $C(x_i(t))$ is the instantaneous odor concentration at glowworm i's location at time t.

A glowworm without a neighbor, but with a nonzero luciferin value switches to the anemotactic behavior: it takes a step in the upwind direction, as given by (14.7), only when the measured concentration is above a threshold. This condition prevents a glowworm from leaving the plume and proceeding upwind away from the source. In case the concentration measured at its current position is below the threshold value, the glowworm stays at its current position.

$$x_i(t) \quad \leftarrow \quad x_i(t) - sw \tag{14.7}$$

where, w is the wind direction and s is the step size. A glowworm without a neighbor and with a zero luciferin value switches to a spiralling behavior until it either finds a neighbor or measures non-zero luciferin.

Most methods have addressed the problems of either (multiple) source localization or boundary mapping separately. However, [8] proposed a novel algorithm that enables a robotic swarm to achieve the following dual goals simultaneously: localization of multiple sources of contaminants spread in a region and mapping of the boundary of the affected region. The algorithm uses the basic GSO and modifies it considerably to make it suitable for both these tasks. Two types of agents, called the source localization agents (or S-agents) and boundary mapping agents (or B-agents) are used for this purpose. Whereas the S-agents behave according to the basic GSO, thereby achieving source localization, new behavior patterns are designed for the B-agents based on their terminal performance as well as interactions between them that help these agents to reach the boundary. The B-agents follow a luciferin update rule depending on the instantaneous point measurements of the level of contamination (a function value at a point in $x - y$ plane) made by the B-agent. Compared to the S-agents, the luciferin update rule is different for the B-agents. The acceptable level of contamination is defined as $\tilde{J}$. The instantaneous point measurement, which an i-th agent makes at a location $x_b^i(t)$ is

$J(x_b^i(t))$. If the value $J(x_b^i(t))$ is higher than $\tilde{J}$, the luciferin value of the i-th agent is updated as follows:

$$l_b^i(t) = (1-\rho)l_b^i(t-1) - \gamma J(x_b^i(t)); \quad \forall i = 1, \ldots, n_b \tag{14.8}$$

where, ρ and γ are the luciferin update scalar parameters in basic GSO and fixed at the same values as that of the S-agents. When $J(x_b^i(t)) \leq \tilde{J}$, the luciferin value associated with the ith agent is updated as

$$l_b^i(t) = 0 \tag{14.9}$$

By assigning the luciferin value to zero, those agents which are on the boundary are restricted from moving further to a safe region where the instantaneous point measurement could be less than $\tilde{J}$. By this method, the B-agents all converge on the boundary and mark the boundary by their presence.

14.3.2 Local optima mapping modification

The basic GSO was formulated to seek the local optima of a given multimodal function, where the fitness (luciferin level) of each glowworm is based on the objective function value at its current location. Standard benchmark multimodal functions were used to test the working of GSO. Aljarah and Ludwig [14] proposed Clustering-GSO, which involves a modification where the multimodal function value $J(g_j)$ of each glowworm g_j, $j = 1, \ldots, m$, where m is the swarm-size, is constructed as a function of its distances to data instances distributed in an d-dimensional space, where d is the size of each data instance and also the dimension of the position vector of each glowworm:

$$J(g_j) = \frac{InterDist \times \frac{1}{n}|cr_j|}{SSE \times \frac{intraD_j}{\max\limits_j(intraD_j)}} \tag{14.10}$$

$$SSE = \sum_{j=1}^{k}\sum_{i=1}^{|C_j|}(Distance(x_i, c_j))^2 \tag{14.11}$$

$$InterDist = \sum_{i=1}^{k}\sum_{j=i}^{|C_j|}(Distance(c_i, c_j))^2 \tag{14.12}$$

$$IntraD_j = \sum_{i=1}^{|Cr_j|} Distance(cr_{ji}, g_j) \tag{14.13}$$

where SSE is the sum of squared errors, InterDist is a function of inter distance between centroids, IntraD_j is a function of the distance of each glowworm g_j to the data instances covered by it, n is the number of data instances, x_j is the location of glowworm g_j, cr_j is the cluster of data instances covered by glowworm g_j, $C = \{C_1, C_2, \ldots, C_k\}$ is the set of clusters, $c = \{c_1, c_2, \ldots, c_k\}$

is the set of corresponding centroids of the clusters, and k is the number of clusters.

This modification enables the algorithm to identify different types of data instances, where each local optimum of the resulting multimodal function corresponds to the optimal centroid of a different cluster of data instances.

14.3.3 Coverage maximization modification

Wireless sensor networks (WSNs) are large collections of sensor nodes with capabilities of perception, computation, communication, and locomotion. They are usually deployed in outdoor fields to carry out tasks like climate monitoring, vehicle tracking, habitat monitoring, earthquake observation, and surveillance. The performance of a WSN is mainly influenced by its coverage of the service area. This problem deals with finding an efficient deployment of the sensor nodes so that every location in the region of interest is sampled by a minimum of one node.

Liao et al. [17] proposed a sensor deployment scheme based on GSO that maximizes the coverage of the sensors with limited movement after an initial random deployment. The decentralized nature of the GSO based approach leads to scalable WSNs. They presented simulation results to show that their approach outperforms the virtual force algorithm (VFA) in terms of coverage rate and sensor movement.

They modeled the sensor deployment problem in the framework of GSO as follows. Each sensor node is considered as a glowworm emitting luciferin whose intensity is a function of its distance from its neighbors. Each glowworm has a sensing range r_s and a communication radius r_c. In original GSO, a glowworm is attracted toward a neighbor of brighter luminescence. On the contrary, in the proposed approach, a glowworm is attracted toward its neighbors having dimmer luminescence and decides to move toward one of them. These local movement rules enable the glowworms to gradually distribute themselves within the sensing field so that coverage is maximized. The luciferin of each glowworm i at time t is computed as below:

$$\ell_i(t) = \ell_i(t-1) + \sum_{j=1}^{|N_i(t)|} \frac{\ell_j(t)}{d_{ij}^2(t)} \tag{14.14}$$

During the movement phase, each glowworm selects to move toward a neighbor with the following probability:

$$p_{ij}(t) = \frac{\ell_i(t) - \ell_j(t)}{\sum_{k \in N_i(t)} \ell_i(t) - \ell_k(t)} \tag{14.15}$$

$$j \in N_i(t) \neq \phi$$

$$N_i(t) = \{j : d_{ij}(t) < r_c \text{ and } \ell_j(t) < \ell_i(t)\} \tag{14.16}$$

where $N_i(t)$ is the set of neighbors of glowworm i at time t.

The optimal distance between neighboring sensors for maximum coverage is $\sqrt{3}r_s$ [18]. Therefore, the distance moved by the glowworm toward its neighbor is chosen as $\frac{\sqrt{3}r_s - d_{ij}(t)}{2}$. Therefore, the movement update for glowworm i is given by:

$$x_i(t+1) \quad = \quad x_i(t) + \left(\frac{\sqrt{3}r_s - d_{ij}(t)}{2}\right)\left(\frac{x_j(t) - x_i(t)}{||x_j(t) - x_i(t)||}\right) \quad (14.17)$$

14.3.4 Physical robot modification

It may be recalled that the original GSO was devised with the express intent to make it implementable on swarm robotic platforms. However, in order to implement GSO on a robotic platform it needs to be modified in order to make the implementation possible, without compromising with the working of the basic GSO. The swarm robotics based approach to source localization usually involves a swarm of mobile robots that search a given area for previously unknown signal-sources. These robots use cues such as their perception of signal-signatures at their current locations, and any information available from their neighbors, in order to guide their movements toward, and eventually to converge at, the signal-emitting sources. The GSO algorithm serves as an effective swarm robotics approach to the source localization problem.

Certain algorithmic aspects need modifications while implementing in a robotic swarm mainly because of the point-agent model of the basic GSO algorithm as against the physical dimensions and dynamics of a real robot. The modifications incorporated into the algorithm in order to make it suitable for a robotic implementation are the subject matter of this subsection.

There are mainly three issues that arise during a robotic implementation that are not taken into account in the algorithmic description of GSO:

1. The linear and angular movements of agents in the algorithm are instantaneous in nature. However, physical robots spend a finite time in performing linear and angular movements based on their prescribed speeds and turn rate capabilities.
2. In the algorithm, agents are considered as points and agent collisions are ignored. These are acceptable assumptions for numerical optimization problems. However, real robots have a physical shape and foot print of finite size, and cannot have intersecting trajectories. Thus, robots must avoid collisions with other robots in addition to avoiding obstacles in the environment.
3. In the point model implementation of GSO agents move over one another to perform a local search. Since this is not possible for real robots, they have to perform collision avoidance maneuvers around other robots.

The above issues are very important and need careful consideration as they call for changes in the agent movement models and alter the working of the basic GSO algorithm when it is implemented on a swarm of real robots.

Thus, each member of the mobile robot swarm used to implement GSO should possess the following capabilities:

1. Sensing and broadcasting of profile-value (luciferin level) at its location.
2. Detection of the number of its neighbors and their locations relative to its own location.
3. Reception of profile-values (luciferin levels) from its neighbors.
4. Selection of a neighbor using a probability distribution (based on the relative luciferin levels of its neighbors) and making a step-movement toward it.
5. Variation of the neighborhood range.
6. Avoiding collisions with obstacles and other robots in the environment.

As a real-robot simulation platform is used and perfect sensing and broadcast are assumed in the initial experiments, the first four robot capabilities are rather straightforward to implement. However, the mechanisms of implementing obstacle avoidance is non-trivial and are described below. They also serve to modify the GSO behavior.

The footprint of the robot is considered to be an octagon with a circumcircle radius r_{robot}. A pair of sonar based proximity-detection sensors with a range r_{sonar} and field of view θ_{fov} are mounted in the frontal region of the robot at a distance d_{sonar} from the robot-center. A simple obstacle avoidance rule is used where the robot decides to turn right (left) if the left (right) sensor detects the other robot. The robot performs the collision-avoidance maneuver by moving along an arc, whose radius of curvature is r_{cv}, and never reaching closer than a safe-distance d_{safe} to the other robot/obstacle. Using simple geometry, the radius of curvature r_{cv} can be calculated as below:

$$r_{cv} = \frac{(d_{sonar} + r_{sonar} + r_{robot})^2 - (2r_{robot} + d_{safe})^2}{2(r_{robot} + d_{safe})} \qquad (14.18)$$

14.4 Engineering applications of GSO

14.4.1 Application of behavior switching to multiple source localization and boundary mapping

In [9, 10] a number of experiments were conducted to validate the algorithm's ability to simultaneously capture multiple odor sources. The proposed approach was later tested on data obtained from a dye mixing experiment. It was also seen capable of locating an odor source under varying wind conditions. A detailed survey on robot algorithms for localization of multiple emission sources was presented by McGill and Taylor [11]. The survey recognized that GSO directly addresses the issue of automatically partitioning a swarm into subgroups as required by the multiple source localization problem. This is achieved by the adaptive local decision domain that facilitates the formation of subgroups in order to locate multiple sources simultaneously. In a comparative analysis, the authors reported that GSO can capture the highest number of sources (100 sources and swarm size of 1000), when compared to other state-of-the-art algorithms. Their experiments considered a gradient field consisting of ten Gaussian sources on a continuous 1000-1000 unit space [12]. Dead space was created by setting the field strength to zero if it was below a threshold value of 0.5. Three initial robot distributions were devised: uniform (agents are deployed at the node locations of a 2D grid spanning the search area), drop (all agents are deployed at a single location in the search area), and line (agents are deployed at one of the edges of the search area). In this, a glowworm employs original GSO once it acquires at least one neighbor.

For the boundary mapping problem [8], an area R in which there are three cumulative Gaussian distribution functions representing basically three sources of contamination. The simulations are done on this static function profile with these three contaminant sources with the assumption that the function values represent the time averaged contaminant intensities at that point. The contour plot of the contaminant spread is considered such that the outer contour corresponds to a value $\tilde{J} = 0.39$ representing an acceptable level of the contamination spread over the region. The objective of the problem is to determine simultaneously an approximation to the outer contour and the sources. The parameters for the modified GSO algorithm are set as follows: sensor range $r_s = 1$, local decision range the same as sensor range, repulsion distance is 0.5, which is less than the sensor range, and maximum number of iterations is 1500. For the experiment, 50 S-agents and 100 B-agents are randomly deployed initially over the region. Luciferin value is initialised with zero. The luciferin decay and enhancement constants are fixed at $\gamma = 0 : 6$ and $\rho = 0 : 4$. The number of desired neighbouring agents is 2. The decision range gain is defined as 0.08. Several cases were considered and it was shown that a good spread of B-agents was achieved while the S-agents converged to multiple sources.

Another very interesting engineering application of GSO to the boundary mapping problem is the implementation of the above algorithm for a boundary mapping problem and the development of a Swarm Algorithm Test-bed (SAT) [13]. The implementation involves the use of three robots, the SAT and a planner.

14.4.2 Application of local optima mapping modification to clustering

The problem of clustering deals with partitioning a set of objects into clusters so that the objects in the same cluster are more similar to each other than to those in other clusters. Clustering has important applications in exploratory data analysis, pattern recognition, machine learning, and other engineering fields. Aljarah and Ludwig [14] applied the local optima mapping modification to the clustering problem. The goal was to partition a set of data instances D into different clusters by having the glowworm swarm split into subswarms, and each subswarm converge to the centroid of each cluster. The GSO objective was modified to locate multiple optimal centroids such that each centroid represents a sub-solution and the combination of these sub-solutions formulate the global solution for the clustering problem. The proposed implementation consisted of four main phases: initialization phase, luciferin level update, glowworm movement, and candidate centroids set construction.

During initialization, a glowworm swarm of size m was created using uniform randomization within the given search space within the minimum and the maximum values that are calculated from the data set D. Then, the luciferin level was initialized using the initial luciferin level ℓ_0. The fitness function value $J(g_j)$ was initialized to zero. The local range r_s was set to an initial constant range r_0. Next, the set of data instances cr_j covered by g_j was extracted from D. The fitness function is evaluated for each glowworm g_j by using eq. (14.11). After this, the steps of the original GSO are applied.

The authors presented the results obtained using their modified algorithm on well-known data sets to conduct a reliable comparison. They presented experimental results to show that GSO with the proposed modification is efficient compared to other well-known clustering algorithms like K-Means clustering, average linkage agglomerative Hierarchical Clustering, Furthest First, and Learning Vector Quantization.

14.4.3 Application of coverage maximization modification to wireless networks

Liao et al. [17] evaluated the performance of the proposed GSO-based sensor deployment scheme by using simulations. The authors considered a variable number of sensor nodes (n = 50, 100, and 200) with two different initial deployments: center and random. A 2D obstacle-free environment was considered for deployment of sensors. The first result showed that the coverage rate increased with an increase in number of nodes and the GSO-based scheme achieved a higher coverage (96 % for n = 200) compared to VFA (40 % for n = 200) in all the cases. The second result showed that the coverage rate is higher in the case of random deployment than center deployment irrespective of the number of sensors in the network. The third result showed that the GSO-based scheme achieved a lower moving distance compared to VFA.

14.4.4 Application of physical robot modification to signal source localization

Several real-robot experiments are carried out on actual robot platforms. For this purpose, wheeled robots, called Kinbots [15], that were originally built for experiments related to robot formations, are used. By making necessary modifications to the Kinbot hardware, the robots are endowed with the capabilities required to implement the various behavioral primitives of GSO. These robots have been used for sound source localization [7] and light source localization experiments [16] demonstrate the potential of robots using GSO for localizing signal sources.

Sound source localization [7]: Each Glowworm is equipped with a sound pick up sensor in order to measure the intensity of sound at its location. A PIC16F877 microcontroller is used as the robot's processing unit.

The hardware used to implement the modules of luciferin broadcast/reception is as follows: The glow consists of an infrared light modulated by an 8-bit serial binary signal that is proportional to the Glowworm's luciferin value at the current sensing-decision-action cycle. Four emitters that are mounted vertically and symmetrically about the Glowworm's central axis cast the infrared-light onto a buffed aluminium conic reflector (with an azimuth of 45^o) in order to obtain a near circular pattern of luciferin emission in the Glowworm's neighborhood. Two infrared receivers mounted on a sweep-platform are used as luciferin receptors. In order to avoid problems due to interference between data signals coming from different neighbors, the receiver sweeps and aligns along the line-of-sight with a neighbor before reading the luciferin data transmitted by it. Using the above scheme, a minimum threshold separation of 10 cm between neighbors was observed to be sufficient in order to distinguish data coming from different glowworm neighbors.

The hardware used to implement the modules of luciferin broadcast/reception and relative localization of neighbors is as follows: Two photodiodes mounted on the rotary platform perform a 180^o sweep and record the intensity of the received infrared light as a function of the angle made by the direction of the source with the heading direction of a Glowworm. Local peaks in the intensity pattern indicate the presence and location of other glowworms. The received intensity of infrared light follows an inverse square law with respect to distance which is used to compute range information to other robots. Even though the Glowworm locates all others within the perception range of the distance-sensor (excepting those that are eclipsed by other glowworms), it identifies them as neighbors only when they are located within its current variable local-decision domain.

In the robotic platform experiment, localization of a single sound source was demonstrated. The sound source was a loud speaker activated by a square wave signal of frequency 28 Hz. A microphone based sound sensor enables each Glowworm to measure the sound intensity at its current location. The sound-intensity pattern in the workspace, is obtained by taking measurements at a sufficiently large number of locations. At first a Glowworm (A) is placed near the sound source and a dummy Glowworm (B) away from the source which is kept stationary but made to emit luciferin proportional to the intensity measurement at its location. A is already located at the source and doesn't get a direction to move and hence, remains stationary. Initially, since B is in the vicinity of C (while A is not), it moves towards B. However, as it reaches closer to B it senses A and hence, changes direction in order to move towards A. Since D is closer to A, it makes deterministic movements towards A at every step. In this manner, the glowworms localize the sound source eventually.

Light source localization [16]: A PIC16F877 microcontroller is used as the robot's processing unit. Experiments are conducted in which robots use GSO to localize a light source. A tethered power supply was used in these experiments.

As GSO is used in these experiments for locating a light source, a light pick-up sensor is used to measure the light intensity at the robot's location. The light sensor output is fed to an in-built analog-to-digital (A/D) convertor of the microcontroller. The output of the A/D module is converted into a luciferin glow that consists of an infrared emission modulated by an 8-bit serial binary signal that is proportional to the robot's luciferin value at the current sensing-decision-action cycle. Eight IR emitters that are mounted symmetrically about the robot's central axis are used to broadcast luciferin data in its neighborhood. An infrared receiver mounted on a sweep platform is used as a luciferin receptor. In order to avoid interference between data signals coming from different neighbors, the receiver sweeps and aligns along the line-of-sight with a neighbor to read the luciferin data broadcast by it.

For relative localization of neighbors, an improved neighbor localization hardware module is built with a circular array of sixteen infrared LEDs, placed radially outward. This array serves as a beacon to obtain a near circular emission pattern around the robot. These IR LEDs are actuated by a 1 KHz square wave signal. A photodiode, mounted on a rotary platform, performs a sweep and records the intensity of the received infrared light as a function of the angle made by the direction of the source with the heading direction of the robot. Local peaks in the intensity pattern indicate the presence of neighbors and serve to compute their location in polar coordinates. As in the sound localization problem, the received intensity of infrared light follows an inverse square law with respect to distance, which is used to compute range information to robot-neighbors and the angular position of the local peak is approximated as the relative angle to a neighbor. The photo-diode output is passed through a bandpass filter, centered around a frequency of 1 KHz, in order to make it sensitive only to the frequency of the actuation signal of the infrared-beacons and reject noise due to ambient light conditions and IR signals used to broadcast luciferin information.

The obstacle avoidance maneuver is achieved by the movement of the robot along an arc in physical simulations. However, in real-robot implementation, when a robot approaches closer than a safe distance $d_{safe} < s$ to another robot, it performs a simple collision-avoidance maneuver through a discrete sequence of point-turn and straight-line movements.

Localization of a single light source was demonstrated through an experiment, in which two Kinbots K_A and K_B implement GSO to detect, taxi toward, and co-locate at a light source. The robots provide simple light intensity measurements at their current locations as inputs to the GSO algorithm running on their onboard processors. The robots use a photodiode based light sensor for this purpose.

In this experiment, the robots are initially deployed in such a way that K_B is closer to the light source. Both the robots start scanning their respective neighborhoods and sense each other. As $\ell_A(0) < \ell_B(0)$, K_B remains stationary and K_A moves toward K_B until $t = 10$ sec. However, between $t = 10$ sec and $t = 30$ sec, K_A remains stationary. This can be attributed to the fact that the luciferin value received by K_A is corrupted. After $t = 30$ sec K_A resumes moving toward K_B until $t = 35$ sec when it switches to obstacle avoidance behavior. Now K_B starts moving toward K_A at $t=$ 45 sec. However, at $t =$ 45 sec, K_A is still relatively far from the light source than K_B. This can be attributed to the fact that when the robots are very close to each other, the difference in luciferin values interferes with the range of sensor noise. K_B stops moving at $t = 65$ sec). However, at $t=$ 110 sec, K_A is closer to the source and $\ell_{K_A}(110) > \ell_{K_B}(110)$. Therefore, K_B moves toward K_A. This results in the robot pair moving closer to the light source.

The paths traced by the two robots, as they execute the modified GSO algorithm, show that toggling between the basic GSO and obstacle avoidance behaviors of the two robots, eventually leads to their localization at the light source. The robots remain idle during certain time intervals, which can be attributed to one of the following reasons:

1. A robot is isolated.
2. A robot is measuring the highest light intensity.
3. Both the robots are measuring almost the same light intensity (A robot is made to move toward another one only when $|\ell_A - \ell_B| > 3$).
4. Reception of luciferin data is corrupted.

From the experimental results it is also clear that the sensing-decision-action cycles of the robots are asynchronous with respect to each other.

14.5 Conclusions

This chapter presented a few modifications that researchers have suggested to the original GSO algorithm in order to make the algorithm suitable to the unique needs of that application. The applications themselves are also presented and discussed. These modifications and applications provide a glimpse into the diverse use of GSO in solving engineering problems. Apart from these, in the literature, there are a large number of other modifications of different kinds that combine GSO with other swarm intelligence algorithms and obtain superior results. These results are indicative of the usefulness of the basic philosophy of GSO to engineering applications and its possible application in many other practical problems.

References

1. K.N. Krishnanand and D. Ghose. Detection of multiple source locations using a glowworm metaphor with applications to collective robotics. In *Proceedings of IEEE Swarm Intelligence Symposium*, Pasadena, California, June 8−10, 2005, pp. 84−91.
2. K.N. Krishnanand and D. Ghose. Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions. *Swarm Intelligence*, Vol. 3, No. 2, pp. 87−124, 2009.
3. K.N. Kaipa and D. Ghose. *Glowworm Swarm Optimization: Theory, Algorithms, and Applications*. Studies in Computational Intelligence. Vol. 698. Springer, 2017.
4. Kalaiselvi, T., Nagaraja, P., & Basith, Z. A. A comprehensive study on Glowworm Swarm optimization. *Computational Methods, Communication Techniques and Informatics*, pp. 332-337.
5. Kalaiselvi, T., Nagaraja, P., & Basith, Z. A. A review on glowworm swarm optimization. *International Journal of Information Technology*, 3(2), 2017, pp. 49-56.
6. Singh A., Deep K. How Improvements in Glowworm Swarm Optimization Can Solve Real-Life Problems. In: *Das K., Deep K., Pant M., Bansal J., Nagar A. (eds) Proceedings of Fourth International Conference on Soft Computing for Problem Solving. Advances in Intelligent Systems and Computing*, vol. 336, 2015, Springer, New Delhi.
7. K.N. Krishnanand, P. Amruth, M.H. Guruprasad, S.V. Bidargaddi, and D. Ghose. Glowworm-inspired robot swarm for simultaneous taxis toward multiple radiation sources. In *Proceedings of IEEE International Conference on Robotics and Automation*, Orlando, Florida, May 15−19, 2006, pp. 958−963.
8. P.P. Menon and D. Ghose. Simultaneous source localization and boundary mapping for contaminants. *American Control Conference*, Montreal, Canada, June 27-29, 2012, pp. 4174-4179.
9. J. Thomas and D. Ghose. Strategies for locating multiple odor sources using glowworm swarm optimization. In *Proceedings of*

Indian International Conference on Artificial Intelligence, Tumkur, India, December 16−18, 2009, pp. 842−861.

10. J. Thomas and D. Ghose. A GSO-Based Swarm Algorithm for Odor Source Localization in Turbulent Environments. In *Handbook of Approximation Algorithms and Metaheuristics: Contemporary and Emerging Applications*, Vol. 2, pp. 711-738, Chapman & Hall/CRC Press, 2018.

11. K. McGill and S. Taylor. Robot algorithms for localization of multiple emission sources. *ACM Computing Surveys*, Vol. 43, No. 3, 2011.

12. K. McGill and S. Taylor. Comparing swarm algorithms for multi-source localization. *IEEE International Workshop on Safety, Security, and Rescue Robotics*, Denver, Colorado, November 3-6, 2009.

13. A. Arvind, D. Ghose, P. Menon. Development of a Robotic Platform to Implement a Boundary Mapping Algorithm. International Conference on Advantages in Control and Optimization of Dynamical Systems, March 2014, Special Issue of IFAC Proceedings Volumes, 47(1), 2014, pp. 484–490.

14. I. Aljarah and S. Ludwig. A new clustering approach based on glowworm swarm optimization. *2013 IEEE Congress on Evolutionary Computation*, June 2013, pp. 2642-2649.

15. K.N. Krishnanand and D. Ghose. Formations of minimalist mobile robots using local templates and spatially distributed interactions. *Robotics and Autonomous Systems*, Vol. 53, No. 3−4, pp. 194−213, 2005.

16. K.N. Krishnanand and D. Ghose. A glowworm swarm optimization based multi-robot system for signal source localization. In *Design and Control of Intelligent Robotic Systems*, Springer-Verlag, Vol. 177, pp. 53−74, 2008.

17. W-H. Liao, Y. Kao, Y-S. Li. A sensor deployment approach using glowworm swarm optimization algorithm in wireless sensor networks. *Expert Systems with Applications*, Vol. 38, pp. 12180−12188, 2011.

18. H. Zhang and J.C. Hou. Maintaining sensing coverage and connectivity in large sensor networks. *Ad Hoc and Sensor Wireless Networks*, Vol. 1(1−2), 2005, pp. 89−124.

19. K.N. Krishnanand, D. Ghose. Glowworm Swarm Optimization - A Tutorial. In *Swarm Intelligence Algorithms: A Tutorial*, CRC Press, Taylor & Francis Group, 2020.