Hindawi
Journal of Sensors
Volume 2020, Article ID 3482463, 20 pages
https://doi.org/10.1155/2020/3482463

Research Article

Passive Target Localization Problem Based on Improved Hybrid Adaptive Differential Evolution and Nelder-Mead Algorithm

Maja B. Rosić, Mirjana I. Simić, and Predrag V. Pejović

University of Belgrade, School of Electrical Engineering, Bulevar Kralja Aleksandra 73, Belgrade, Serbia

Correspondence should be addressed to Maja B. Rosić; rosic.maja9@gmail.com

Received 29 October 2019; Revised 20 January 2020; Accepted 24 January 2020; Published 27 February 2020

Academic Editor: Antonio Lazaro

This paper considers a passive target localization problem in Wireless Sensor Networks (WSNs) using the noisy time of arrival (TOA) measurements, obtained from multiple receivers and a single transmitter. The objective function is formulated as a maximum likelihood (ML) estimation problem under the Gaussian noise assumption. Consequently, the objective function of the ML estimator is a highly nonlinear and nonconvex function, where conventional optimization methods are not suitable for this type of problem. Hence, an improved algorithm based on the hybridization of an adaptive differential evolution (ADE) and Nelder-Mead (NM) algorithms, named HADENM, is proposed to find the estimated position of a passive target. In this paper, the control parameters of the ADE algorithm are adaptively updated during the evolution process. In addition, an adaptive adjustment parameter is designed to provide a balance between the global exploration and the local exploitation abilities. Furthermore, the exploitation is strengthened using the NM method by improving the accuracy of the best solution obtained from the ADE algorithm. Statistical analysis has been conducted, to evaluate the benefits of the proposed modifications on the optimization performance of the HADENM algorithm. The comparison results between HADENM algorithm and its versions indicate that the modifications proposed in this paper can improve the overall optimization performance. Furthermore, the simulation shows that the proposed HADENM algorithm can attain the Cramer-Rao lower bound (CRLB) and outperforms the constrained weighted least squares (CWLS) and differential evolution (DE) algorithms. The obtained results demonstrate the high accuracy and robustness of the proposed algorithm for solving the passive target localization problem for a wide range of measurement noise levels.

1. Introduction

Determining the position of a passive target using time of arrival (TOA) measurements has become an important issue for a number of different applications, such as radar, sonar, telecommunications, mobile communications, and navigation [1, 2]. In general, localization systems can be classified into active and passive [1]. With the active localization system, the target actively participates in the localization process. Compared to the active case, in passive localization, the target is not involved in the localization process and can only reflect or scatter the signals from the transmitter to receivers [2]. Therefore, the passive localization has been widely applied to different fields, such as robots, underwater acoustics, radar, crime-prevention, surveillance, and urban environments [3, 4].

Hence, this paper proposes a passive target localization problem using noisy TOA measurements. In this way, the signal propagation time from the transmitter via the target to the receiver can be measured and used to determine range measurements, e.g., transmitter-target-receiver distances. Due to the nonlinear relationship between the target position and measurements, various efficient estimation methods have been proposed, such as nonlinear least squares (NLS) and maximum likelihood (ML) [5]. To obtain a closed-form solution, the NLS problem has been linearized using the linear least squares (LLS) and the weighted least squares (WLS) algorithms [6, 7]. In order to improve the accuracy, especially in the case of high measurement noise, the constrained weighted least squares (CWLS) is introduced [8]. Hence, the CWLS problem can be formulated as a quadratically constrained quadratic program (QCQP), which is converted to an unconstrained optimization problem using the method of Lagrange multipliers [8]. Another widely used estimation method is the ML estimator, which is commonly applied when the measurement error distribution is previously known [2]. In general, the ML objective function is highly nonlinear, with multiple local optima, and thus, the solution in closed-form cannot be obtained. Therefore, finding the global optimal solution by conventional optimization methods is difficult, due to the multimodal objective function [9]. For this reason, various efficient optimization algorithms have been derived to overcome these types of difficulties and to answer the challenges of complex optimization problems [10].

Motivated by these facts, this paper proposes evolutionary algorithms (EAs) and their hybrid variants to overcome these drawbacks and improve the localization performance [11]. Generally, the optimization process of EAs consists of two phases, namely, exploration and exploitation [10]. In such a context, the first phase consists of exploring the search space and locating the region of the global optimum, while the exploitation phase investigates the promising area to refine the solution around the current global best solution. Therefore, finding an appropriate balance between the exploration and exploitation abilities during the evolution process is a significant challenge for optimization algorithms.

Various EAs have been successfully applied to solve different complex optimization problems, such as the particle swarm optimization (PSO) [9], the differential evolution (DE) [12], the artificial bee colony (ABC) [13], and the cuckoo search algorithm (CS) [14]. Among these algorithms, the DE has emerged as a very efficient and robust EA for global optimization with successful applications in finding a global optimum [15].

The DE is a population-based EA, which has been widely used to solve numerous optimization problems in different fields of science and engineering [15, 16]. Easy implementation, compact structure, reliability, and robustness are the main advantages of the DE algorithm [15]. However, some difficulties occur, such as a weak local search and slow convergence [17]. In general, the main factors that affect the performance are mutation and crossover operators and their corresponding control parameters, such as the scale factor (F), crossover rate (CR), and population size (NP). In this direction, various empirical guidelines have been provided for selecting the most suitable values for control parameters such as $F \in [0.4, 1]$ and $\mathrm{CR} \in [0, 1]$ [18]. To obtain acceptable results for a given problem, different control parameter values at different stages of the evolution process are needed. In this way, a large F is required at the early stage of the evolution process for the purpose of preventing premature convergence, while a smaller F is preferred to accelerate the convergence at the later stage of the evolution process [19]. A large value of CR in the early stage of evolution can increase the population diversity. On the contrary, a smaller CR can improve the local exploitation ability and convergence at the later stage of evolution [19]. In this regard, different adaptive mechanisms eliminate the need for manual tuning of control parameters, which are adjusted adaptively using the feedback from the evolution process [20, 21].

Furthermore, choosing the appropriate mutation operator can significantly affect the optimization performance of the DE algorithm. In this way, a number of different mutation operators have been developed, such as DE/rand/1, DE/rand/2, DE/best/1, DE/best/2, DE/current-to-best/1, DE/current-to-rand/1, and different variants of them [15]. The DE/rand/1 and DE/rand/2 generally have powerful global exploration ability, while the DE/best/1 and DE/current-to-best/1 have strong local exploitation ability [20].

Hybridization of DE with other algorithms is another way to overcome the drawbacks of both algorithms and further enhance the optimization performance. Depending on the type of algorithm, the DE can be hybridized with other EAs, such as ABC, CS, and PSO [13, 22] or with different local search methods such as Powell's method, the Hook-Jeeves (HJ), and the Nelder-Mead (NM) [23–25]. Among them, the NM method has been chosen, due to its excellent local search ability. However, the convergence of the NM method is extremely sensitive to the choice of the initial point [25], and thus, it cannot be used alone to find the global optimum of a multimodal objective function.

Based on the above considerations, this paper is aimed at proposing an improved HADENM algorithm, in order to efficiently find the estimated position of a passive target. The proposed algorithm is a two-stage method, where in the first stage, the adaptive differential evolution (ADE) algorithm is used as the global optimizer, to quickly locate the promising region containing the global optimum. Then, in the second stage, the NM method is employed to improve the accuracy of the best solution obtained from the ADE algorithm. Moreover, an adaptive adjustment parameter is introduced in the mutation phase of the ADE algorithm to automatically apply the appropriate mutation operator, in order to avoid the problem of stagnation and premature convergence. In addition, to further improve the optimization performance, the control parameters of the ADE algorithm are automatically and adaptively adjusted during the evolution process. Therefore, the purpose of this paper is to propose a robust optimization algorithm for the passive target localization problem for a wide range of measurement noise levels.

Finally, the Cramer-Rao lower bound (CRLB), which provides a lower bound on the variance of any unbiased estimator, has been derived and used as a benchmark to evaluate the localization performance [26]. Hence, the CRLB for the passive target localization problem is compared with the root mean square error (RMSE) performance of the proposed HADENM algorithm and the existing CWLS and DE algorithms.

The contributions of this paper are summarized as follows:

(i) The passive target localization problem in LOS (line-of-sight) environment is formulated using the TOA measurements obtained from a set of receivers and a single transmitter, in Wireless Sensor Networks (WSNs)

(ii) An improved HADENM algorithm, as the hybridization of the ADE and NM algorithms, is proposed in order to efficiently solve the passive target localization problem. Furthermore, the scale factor and the crossover rate are updated using an adaptive strategy. In addition, an adaptive mutation operator has been designed in order to maintain a balance between the exploration and exploitation. To further increase the exploitation ability, the NM method is employed with the aim to enhance the accuracy of the best solution previously obtained by the ADE algorithm

(iii) The experiments are carried out to evaluate the benefits of the proposed modifications on the optimization performance of the proposed HADENM algorithm. Based on the statistical analysis of the proposed algorithm and its versions, it can be concluded that the modifications proposed in this paper can improve the overall optimization performance. Furthermore, the simulation results show the effectiveness of the HADENM algorithm for a wide range of measurement noise levels compared to the CWLS and DE algorithms. Additionally, the proposed algorithm attains the CRLB accuracy and shows better robustness against variations in network topology

The structure of the rest of the paper is organized as follows. Section 2 presents a review of background and related work. In Section 3, the passive target localization problem using noisy TOA measurements is presented. Section 4 gives the derivation of the ML objective function for the passive target localization problem. In Section 5, the CWLS algorithm is described. Section 6 introduces the DE algorithm followed by corresponding modifications. A local search NM method is described in Section 7. In Section 8, the HADENM algorithm is introduced to solve the considered passive target localization problem. In Section 9, the derivation of the CRLB is developed. To evaluate the localization accuracy, experimental results are presented in Section 10. The conclusion and future work are given in Section 11.

2. Background and Related Work

Highly accurate passive target localization can be considered as one of the most significant and challenging issues in WSNs [26]. The accuracy of the estimated target position depends on the geometric configuration of sensors and the measurement accuracy [1]. The localization algorithms in WSNs can be divided into range-based and range-free [5]. Range-based algorithms use range measurements including distance or angle between the target and the receivers, such as TOA [26], time difference of arrival (TDOA) [8], received signal strength (RSS) [27], angle of arrival (AOA) [28], or a combination of them [29]. Among them, TOA is the most commonly used algorithm for solving the localization problems, which can achieve high localization accuracy [26]. On the contrary, the range-free algorithms do not measure the distance or angle information; these algorithms use connectivity information in WSNs to estimate sensor positions [30]. Compared with range-based algorithms, the range-free algorithms do not require complex hardware structure. Hence, the range-free algorithms are cost effective and easy to implement; however, they are less accurate in estimating the sensor position [30].

There are many applications, in which global positioning system (GPS) or other navigation systems cannot be used, such as in indoor, underwater acoustics, and urban environments [31]. In these cases, when GPS signals are not available or do not have sufficient accuracy, the passive localization system can be employed as an efficient alternative. In this regard, the passive target localization system has become an attractive solution for determining the unknown position of the passive target under various circumstances and environments [32]. In this way, a novel two-step algorithm for TOA passive target localization has been proposed for synchronous networks [26]. Furthermore, the two new algorithms based on belief propagation on a factor graph have been developed to localize multiple passive targets, in the case where the receiver position errors exist [2]. In addition, the two-step linear algorithm has been proposed to simultaneously estimate the position of the passive target and the unknown time offset in a quasisynchronous network using TOA measurements [33].

Numerous estimation methods have been proposed to find the position of a target, where the well-known NLS estimation method is commonly employed. To obtain a closed-form solution, the LLS and WLS [6, 7] methods have been widely applied and implemented. However, these linear estimation methods cannot provide high localization accuracy of the target for different measurement noise levels [34]. To further improve the localization performance, especially in the presence of high noise levels, the CWLS algorithm is developed [8]. An additional widely used estimation method in the literature is the ML estimator, which maximizes the likelihood function of the unknown target position. However, the corresponding ML objective function is a highly nonlinear function, under the Gaussian noise assumption [9]. In order to avoid difficulties related to the multimodal nature of the ML objective function, a semidefinite programing (SDP) relaxation is applied to transform the nonconvex problem into a convex problem [35]. However, in the presence of significant measurement noise, the SDP has some disadvantages in terms of accuracy; thus, it can only provide a near-global optimal solution [35]. Thus, to solve the ML estimation problem with high accuracy and improve the convergence, for a given localization problem, researchers have developed and applied various efficient EAs [10].

A number of different EAs like DE, PSO, ABC, and CS have been widely applied in order to solve the localization problem, in terms of determining the unknown position of the target [36]. In this regard, the PSO algorithm and its improved variants have been applied to accurately estimate the position of the passive target using TDOA measurements [37]. The simulation results have shown that improved variants of PSO provide better localization performance compared to the conventional PSO and well-known LLS and WLS algorithms. In addition, the cuckoo search (CS) algorithm has been used to estimate the position of the target in the passive localization system based on TDOA measurements [14]. Based on the simulation results, it can be concluded that the CS algorithm has faster convergence speed and more efficient global exploration ability compared to the PSO and Newton iteration algorithms.

In recent years, a number of hybrid variants of the DE algorithm have been created in order to efficiently balance the exploration and exploitation abilities during the evolution process [23–25]. With respect to this, the direct search Powell's method has been combined with DE (DESAP) [23], where the near-global solution obtained by the DE algorithm is improved using Powell's method. To enhance the performance, the hybridization between DE and a local search algorithm (MLSHADE-SPA), with linear population size reduction and semiparameter adaptation, has been proposed [38]. Furthermore, the DE algorithm has been hybridized with Hook-Jeeves in distributed memetic DE algorithm (DMDE), to efficiently find the global optimum and achieve a better trade-off between the exploration and the exploitation [24]. In addition, a new reflection-based mutation operation, inspired by the reflection operations in the NM method, has been incorporated into the DE algorithm, with a combination of multiple mutation strategies based on roulette wheel selection (MM-RDE) [25].

Therefore, this paper proposes an improved HADENM algorithm, based on the hybridization of ADE and NM algorithms, to solve the multimodal passive target localization problem with high accuracy even in the presence of high measurement noise.

3. Localization Problem

This section considers passive target localization problem in two dimensions, using noisy TOA measurements. The unknown position of a passive target can be determined using one transmitter T_x and a set of receivers $\{R_i\}_{i=1}^N$, where N is the number of receivers, such that $N \geq 3$. Let $\mathbf{x} = [x\, y]^T \in \mathbb{R}^2$ be the unknown position of a passive target, $\mathbf{x}_i^r = [x_i^r\, y_i^r]^T \in \mathbb{R}^2$, $\forall i \in \{1, 2, \cdots, N\}$ be the known coordinates of the ith receiver, and $\mathbf{x}_T = [0\, 0]^T \in \mathbb{R}^2$ be the coordinates of the transmitter, as shown in Figure 1.

The passive target localization system starts with the transmitter emitting a signal, and upon arrival, the signal is reflected by the target. The considered TOA algorithm employs the information of the absolute signal travel time from the transmitter to the receivers to obtain range measurements. Therefore, it is necessary to know the signal departure time, which is achieved by the synchronization between the transmitter and receivers [26]. In addition, the Gaussian noise is widely used in the localization algorithms under LOS environment [2]. In this case, the noisy TOA measurements are obtained as follows:

$$t_\mathrm{i} = \frac{1}{c}\left(\|\mathbf{x}\|_2 + \|\mathbf{x} - \mathbf{x}_i^r\|_2\right) + \bar{n}_i, \quad \forall i \in \{1, 2, \cdots, N\}, \tag{1}$$

where $\|\cdot\|_2$ denotes the Euclidean distance, c is the speed of light, and $\bar{n}_i$ denotes the zero-mean Gaussian measurement noise with variance σ_i^2. Then, multiplying t_i by the speed of light c, the range measurements (transmitter-target-receiver distances), denoted by $\{r_i\}_{i=1}^N$, can be written as

$$r_i = c \cdot t_i = d_i + n_i, \quad \forall i \in \{1, 2, \cdots, N\}, \tag{2}$$

where $d_i = \|\mathbf{x}\|_2 + \|\mathbf{x} - \mathbf{x}_i^r\|_2$ is the true range measurement and $n_i = c\bar{n}_i$ is the measurement noise, which follows the Gaussian distribution $\mathcal{N}(0, \sigma_{ni}^2)$, with zero-mean and variance $\sigma_{ni}^2 = c^2\sigma_i^2$.

From geometric interpretation in Figure 1, in the absence of measurement noise, the true range measurement d_i defines an ellipse E_i with focal points placed at $\mathbf{x}_T$ and $\mathbf{x}_i^r$, respectively. Therefore, the corresponding ellipse E_i can be defined as

$$E_i = \left\{(x, y) \in \mathbb{R}^2 : d_i = \sqrt{x^2 + y^2} + \sqrt{(x - x_i^r)^2 + (y - y_i^r)^2}\right\}. \tag{3}$$

In this regard, the true passive target position $\mathbf{x} = [x \quad y]^T \in \mathbb{R}^2$ is determined at the intersection point of at least three solid line ellipses $\{E_i\}_{i=1}^{N\geq 3}$, corresponding to the TOA measurements, as shown in Figure 1.

Then, in the presence of measurement noise, the vector form of Equation (2) can be written as

$$\mathbf{r} = \mathbf{d}(\mathbf{x}) + \mathbf{n}, \tag{4}$$

in which the vector of true range measurements is

$$\mathbf{d}(\mathbf{x}) = \begin{bmatrix} \|\mathbf{x}\|_2 + \|\mathbf{x} - \mathbf{x}_1^r\|_2 \\ \vdots \\ \|\mathbf{x}\|_2 + \|\mathbf{x} - \mathbf{x}_N^r\|_2 \end{bmatrix}, \tag{5}$$

$\mathbf{n} = [n_1 \quad \cdots \quad n_N]^T$ is the corresponding measurement noise vector. In this case, three or more ellipses derived from measurements do not have a unique intersection point. Thus, an estimated position of the passive target can be found inside the bounded region, i.e., the region surrounded by the bold black curve, as depicted in Figure 1. In this way, the main goal is to estimate the unknown position of the passive target based on noisy TOA measurements, which involves solving a highly nonlinear and multimodal ML estimation problem explained in the following section.

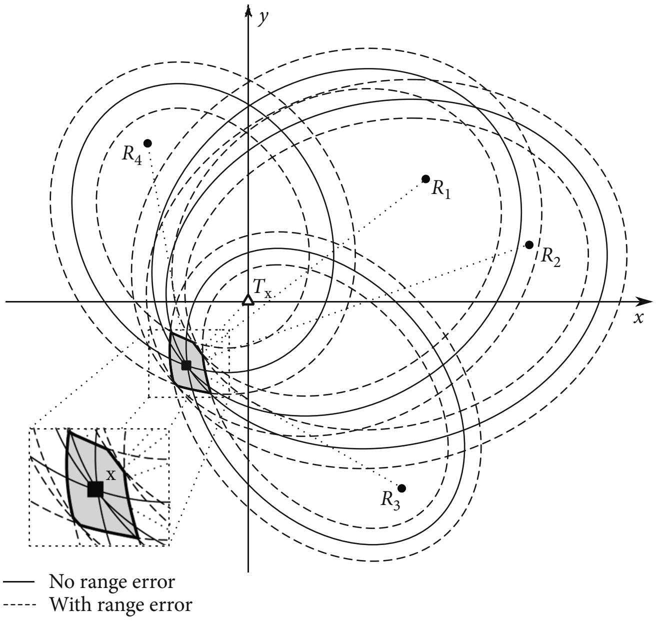

Figure 1: Passive target localization system using noisy TOA measurements.

4. Maximum Likelihood Method

The unknown position of the passive target can be estimated through maximizing the likelihood function. Under the assumptions of Gaussian noisy TOA measurements, which are independent and identically distributed, the likelihood function $L(\mathbf{x})$ for the passive target position [26] can be expressed as

$$L(\mathbf{x}) = f(\mathbf{r}|\mathbf{x}) = \frac{1}{(2\pi)^{N/2} \det(\mathbf{C})^{1/2}} \exp \cdot \left(-\frac{1}{2} (\mathbf{r} - \mathbf{d}(\mathbf{x}))^{\mathrm{T}} \mathbf{C}^{-1} (\mathbf{r} - \mathbf{d}(\mathbf{x})) \right), \tag{6}$$

where $f(\mathbf{r}|\mathbf{x})$ denotes the probability density function and $\mathbf{C} = \mathrm{diag}\,\{\sigma_1^2, \cdots, \sigma_N^2\}$ is a diagonal covariance matrix.

In order to simplify the maximization problem, it is often convenient to take the logarithm of the likelihood function, which can be written as

$$\ln L(\mathbf{x}) = c - \frac{1}{2} (\mathbf{r} - \mathbf{d}(\mathbf{x}))^{\mathrm{T}} \mathbf{C}^{-1} (\mathbf{r} - \mathbf{d}(\mathbf{x})), \tag{7}$$

where $c = \ln\,(1/((2\pi)^{N/2} \det\,(\mathbf{C})^{1/2}))$ is a constant independent of $\mathbf{x}$. Consequently, the ML estimator requires the maximization of the log-likelihood function, which is equivalent to minimizing the negative logarithm of the likelihood function. Then, after neglecting the constant terms, the ML estimation problem can be formulated as

$$\min_{\mathbf{x} \in R^2} J_{\mathrm{ML}}(\mathbf{x}) = \min_{\mathbf{x} \in R^2} (\mathbf{r} - \mathbf{d}(\mathbf{x}))^{\mathrm{T}} \mathbf{C}^{-1} (\mathbf{r} - \mathbf{d}(\mathbf{x})), \tag{8}$$

where $J_{\mathrm{ML}}(\mathbf{x})$denotes the corresponding ML objective function.

Thus, the goal is to find the optimal solution $\hat{\mathbf{x}}$ that minimizes the objective function $J_{\mathrm{ML}}(\mathbf{x})$ with respect to $\mathbf{x}$, such that

$$\hat{\mathbf{x}} = \underset{\mathbf{x} \in R^2}{\operatorname{argmin}} (J_{\mathrm{ML}}(\mathbf{x})). \tag{9}$$

Hence, the $J_{\mathrm{ML}}(\mathbf{x})$ for the passive target localization problem is depicted in Figure 2.

Figure 2 provides information about the nature of the optimization problem and shows that $J_{\mathrm{ML}}(\mathbf{x})$ is a highly nonlinear and nonconvex function with multiple local optima. Therefore, finding the global optimal solution for a given optimization problem becomes a significant challenge. Thus, in order to solve this kind of complex optimization problem, it is necessary to employ different optimization algorithms, which are described in the next sections.

5. Constrained Weighted Least Squares Algorithm

As a well-known and widely used localization algorithm in WSNs, the CWLS algorithm is considered in this paper in order to compare and evaluate the localization performance of the proposed HADENM algorithm. The passive target position is estimated using CWLS algorithm by squaring both sides of Equation (2) and introducing $R^t = \|\mathbf{x}\|_2 = \sqrt{x^2 + y^2}$ [8], which yields

$$2x_i^r x + 2y_i^r y - 2r_i R^t = (x_i^r)^2 + (y_i^r)^2 - r_i^2 - 2n_i R^t + 2r_i n_i, \quad \forall i \in \{1, 2, \cdots, N\}, \tag{10}$$

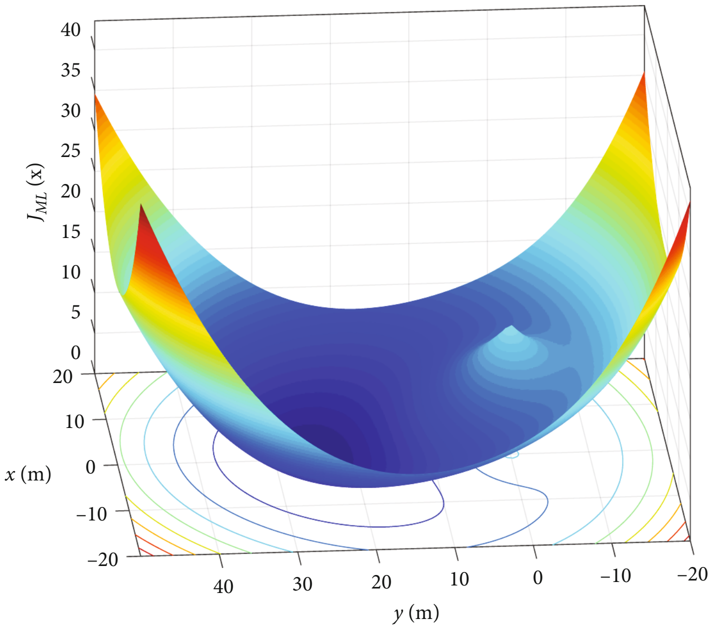

Figure 2: Illustration of the objective function $J_{\mathrm{ML}}(\mathbf{x})$.

where the second-order term of noise n_i^2 can be neglected for small noise levels. Hence, Equation (10) is linear, which can be written as

$$\mathbf{A}\boldsymbol{\theta} = \mathbf{b} + \mathbf{m}, \tag{11}$$

where

$$\mathbf{A} = 2\begin{bmatrix} x_1^r & y_1^r & -r_1 \\ \vdots & \vdots & \vdots \\ x_N^r & y_N^r & -r_N \end{bmatrix}, \tag{12}$$

$$\boldsymbol{\theta} = \begin{bmatrix} x & y & R^t \end{bmatrix}^T, \tag{13}$$

$$\mathbf{b} = \begin{bmatrix} (x_1^r)^2 + (y_1^r)^2 - r_1^2 \\ \vdots \\ (x_N^r)^2 + (y_N^r)^2 - r_N^2 \end{bmatrix}, \tag{14}$$

$$\mathbf{m} = 2\left[\left(r_1 - R^t\right)n_1 \quad \cdots \quad \left(r_N - R^t\right)n_N \right]^T. \tag{15}$$

Then, based on Equation (11) through Equation (15), the following WLS optimization problem can be formulated as

$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} (\mathbf{A}\boldsymbol{\theta} - \mathbf{b})^T \mathbf{W} (\mathbf{A}\boldsymbol{\theta} - \mathbf{b}), \tag{16}$$

where, under the small noise assumption, $\mathbf{W} \in R^{N\times N}$ is a symmetric weighting matrix, which can be defined as

$$\mathbf{W} = \left[E\left\{\mathbf{m}\mathbf{m}^T\right\}\right]^{-1} = \left(\mathbf{D}^T\mathbf{C}\mathbf{D}\right)^{-1}, \tag{17}$$

in which $\mathbf{D} = \operatorname{diag}\left\{2(r_1 - R^t), \cdots, 2(r_N - R^t)\right\}$.

It should be noted that since the measurement noise $\mathbf{m}$ from Equation (15) is Gaussian distributed and due to the linear relationship in Equation (11), the objective function of the ML estimator $J_{\mathrm{ML}}(\mathbf{x})$, given in Equation (8), is equivalent to that of the WLS estimator [8]. Then, by incorporating the relationship between unknown target position $\mathbf{x}$ and an auxiliary variable R^t as a second-order equality constraint in the WLS estimator, the following CWLS target localization problem is obtained as

$$\begin{aligned} \min_{\boldsymbol{\theta}} \quad & J(\boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} (\mathbf{A}\boldsymbol{\theta} - \mathbf{b})^T \mathbf{W} (\mathbf{A}\boldsymbol{\theta} - \mathbf{b}), \\ \text{s.t.} \quad & \boldsymbol{\theta}^T \mathbf{S} \boldsymbol{\theta} = 0, \end{aligned} \tag{18}$$

where $\mathbf{S} = \operatorname{diag}\{1, 1,-1\}$ is a diagonal matrix.

The closed-form solution is not available due to the nonlinearity of the constrained problem. In this regard, an unconstrained optimization problem is obtained by forming the auxiliary Lagrange function $\mathscr{L}(\boldsymbol{\theta}, \lambda)$, as follows:

$$\min_{\boldsymbol{\theta},\lambda} \mathscr{L}(\boldsymbol{\theta}, \lambda) = \min_{\boldsymbol{\theta},\lambda} (\mathbf{A}\boldsymbol{\theta} - \mathbf{b})^T \mathbf{W} (\mathbf{A}\boldsymbol{\theta} - \mathbf{b}) + \lambda \boldsymbol{\theta}^T \mathbf{S} \boldsymbol{\theta}, \tag{19}$$

where λ is the Lagrange multiplier associated with the equality constraint. The corresponding optimal solution $\widehat{\boldsymbol{\theta}}_{\mathrm{CWLS}}$ to the CWLS problem can be obtained as

$$\widehat{\boldsymbol{\theta}}_{\mathrm{CWLS}} = \left(\mathbf{A}^T\mathbf{W}\mathbf{A} + \lambda\mathbf{S}\right)^{-1} \mathbf{A}^T\mathbf{W}\mathbf{b}. \tag{20}$$

Hence, the Lagrange multiplier λ is found as the root of a 4th-order polynomial [8], as follows:

$$\sum_{i=1}^{3} \frac{\alpha_i \beta_i}{(\zeta_i + \lambda)^2} = 0, \tag{21}$$

where $\boldsymbol{\alpha} = \boldsymbol{\Sigma}^T \mathbf{S}\mathbf{A}^T \mathbf{W}\mathbf{b} = [\boldsymbol{\alpha}_1 \; \boldsymbol{\alpha}_2 \; \boldsymbol{\alpha}_3]^T$, $\boldsymbol{\beta} = (\boldsymbol{\Sigma})^{-1}\mathbf{A}^T\mathbf{W}\mathbf{b}$, and $\mathbf{A}^T \mathrm{WAS} = \boldsymbol{\Sigma} \operatorname{diag}\{\zeta_1, \zeta_2, \zeta_3\}\boldsymbol{\Sigma}^{-1}$ [8]. Then, based on the CWLS method, the algorithm for the passive target localization can be stated as follows:

Step 1. Set the symmetric weighting matrix $\mathbf{W}$ as unit matrix $\mathbf{I}$, i.e., $\mathbf{W} = \mathbf{I}$.

Step 2. Find all roots of Equation (21) taking into consideration only real roots.

Step 3. For each λ, obtained in Step 2, determine subestimates of $\widehat{\boldsymbol{\theta}}_{\mathrm{CWLS}}$ using Equation (20). Then, separate subestimates for which $J(\boldsymbol{\theta})$ is minimal.

Step 4. Determine a weighting matrix $\mathbf{W}$ using Equation (17).

Step 5. Repeat Steps 2–4, while the predefined termination criterion is satisfied.

6. Differential Evolution Algorithm and the Proposed Modified Version

6.1. Differential Evolution Algorithm. The DE is a population-based EA successfully applied for global optimization, which is introduced by Storn and Price [12]. The evolution process of the DE algorithm is based on four basic steps, i.e., initialization, mutation, crossover, and selection, which are described below.

6.1.1. Initialization. The evolution process of the DE algorithm starts with an initial population of n-dimensional N_P individuals, in which each individual represents a candidate solution of the problem, i.e., $\{\mathbf{x}_i^{(G)} : x_{i,j}^{(G)} \in [x_{i,j}^L, x_{i,j}^U]\}$, $\forall i \in \{1, 2, \cdots, N_P\}$, $\forall j \in \{1, 2, \cdots, n\}$, where G denotes the current generation. For the considered localization problem, every individual $\mathbf{x}_i^{(G)}$ in two-dimensional space (i.e., $n = 2$) has two variables corresponding to x and y coordinates of the target position. Here, $x_{i,j}^{\mathrm{L}}$ and $x_{i,j}^{\mathrm{U}}$ are the lower and upper bounds of the jth component of the ith individual $x_{i,j}$, respectively. At $G = 0$, every individual is randomly generated as

$$x_{i,j}^{(0)} = x_{i,j}^{\mathrm{L}} + \mathrm{rand}_j\left(x_{i,j}^{\mathrm{U}} - x_{i,j}^{\mathrm{L}}\right), \tag{22}$$

where rand_j is a uniformly distributed random number in the range $[0, 1]$.

6.1.2. Mutation. The mutation operator is employed, after initialization, to generate a new mutant vector $\mathbf{v}_i^{(G)} = \left[v_{i,1}^{(G)} \; \cdots \; v_{i,j}^{(G)} \; \cdots \; v_{i,n}^{(G)}\right]^T$ for each target vector $\mathbf{x}_i^{(G)}$. The most widely used mutation operators in the literature [15] are

DE/rand/1:

$$\mathbf{v}_i^{(G)} = \mathbf{x}_{r_1}^{(G)} + F\left(\mathbf{x}_{r_2}^{(G)} - \mathbf{x}_{r_3}^{(G)}\right), \tag{23}$$

DE/rand/2:

$$\mathbf{v}_i^{(G)} = \mathbf{x}_{r_1}^{(G)} + F\left(\mathbf{x}_{r_2}^{(G)} - \mathbf{x}_{r_3}^{(G)}\right) + F\left(\mathbf{x}_{r_4}^{(G)} - \mathbf{x}_{r_5}^{(G)}\right), \tag{24}$$

DE/best/1:

$$\mathbf{v}_i^{(G)} = \mathbf{x}_{\mathrm{best}}^{(G)} + F\left(\mathbf{x}_{r_1}^{(G)} - \mathbf{x}_{r_2}^{(G)}\right), \tag{25}$$

DE/current-to-best/1

$$\mathbf{v}_i^{(G)} = \mathbf{x}_i^{(G)} + F\left(\mathbf{x}_{\mathrm{best}}^{(G)} - \mathbf{x}_i^{(G)}\right) + F\left(\mathbf{x}_{r_1}^{(G)} - \mathbf{x}_{r_2}^{(G)}\right), \tag{26}$$

where r_1, r_2, r_3, r_4, and r_5 are distinct integers randomly selected from $\{1, 2, \cdots, N_P\} \setminus \{i\}$, $\mathbf{x}_{\mathrm{best}}^{(G)}$ is the individual with the lowest objective function value, and $F \in [0, 1]$ is the scale factor.

6.1.3. Crossover. In order to increase the diversity of the population, a binomial crossover operator is applied on the target vector $\mathbf{x}_i^{(G)}$ and the corresponding mutant vector $\mathbf{v}_i^{(G)}$ to produce a trial vector $\mathbf{u}_i^{(G)}$, which can be expressed as follows:

$$u_{i,j}^{(G)} = \begin{cases} v_{i,j}^{(G)} & \text{if}\left(\mathrm{rand}_{i,j} \le CR \vee j = j_{\mathrm{rand}}\right) \\ x_{i,j}^{(G)} & \text{otherwise} \end{cases}, \quad \forall j \in \{1, 2, \cdots, n\}, \; \forall i \in \{1, 2, \cdots, N_P\}, \tag{27}$$

where $\mathrm{rand}_{i,j}$ is an uniform number generated within $[0, 1]$, $CR \in [0, 1]$ is the crossover rate, and $j_{\mathrm{rand}} \in \{1, 2, \cdots, n\}$ is an integer selected randomly.

6.1.4. Selection. After crossover, the selection operator is employed to compare the objective function values of the corresponding trial vector $\mathbf{u}_i^{(G)}$ with the objective function value of the target vector $\mathbf{x}_i^{(G)}$. The objective function value of the ith individual in the Gth generation has been calculated using the ML objective function in (8), as follows:

$$f\left(\mathbf{x}_i^{(G)}\right) = J_{\mathrm{ML}}\left(\mathbf{x}_i^{(G)}\right), \tag{28}$$

$$f\left(\mathbf{u}_i^{(G)}\right) = J_{\mathrm{ML}}\left(\mathbf{u}_i^{(G)}\right). \tag{29}$$

In this way, the vector with a better objective function value is selected, according to

$$\mathbf{x}_i^{(G+1)} = \begin{cases} \mathbf{u}_i^{(G)} & \text{if}\, f\left(\mathbf{u}_i^{(G)}\right) \le f\left(\mathbf{x}_i^{(G)}\right), \\ \mathbf{x}_i^{(G)} & \text{otherwise.} \end{cases} \tag{30}$$

6.2. A Modified Differential Evolution Algorithm. In order to improve the performance of the conventional DE algorithm, this subsection introduces modifications to the scale factor and crossover rate. Furthermore, an automatically adapted mutation operator is proposed, to select an appropriate mutation strategy based on the current optimization state.

6.2.1. Adaptive Scale Factor. The performance of the DE algorithm depends largely on the proper choice of the scale factor, which can affect the convergence. In this context, a higher value of the scale factor, in the early stage, improves the exploration ability, which has a positive effect on the population diversity and, thus, avoids premature convergence at local optima [18, 39]. In contrast, a smaller value of the scale factor, in the later stage, improves the exploitation ability, which can enhance the convergence speed [18, 39].

Based on the above analysis, an adaptive scale factor $F_i^{(G)}$, which is dynamically adapted in each generation for each individual, is introduced as follows:

$$F_i^{(G)} = \frac{F_{\max} - F_{\min}}{\exp\left(-10G\left(\left(f_i^{(G)} - f_{\text{best}}^{(G)}\right)/\left(f_{\text{worst}}^{(G)} - f_{\text{best}}^{(G)}\right)\right)\right)}, \quad \forall i \in \{1, 2, \cdots, N_P\}, \tag{31}$$

where $F_{\min}$ and $F_{\max}$ are minimum and maximum values of $F_i^{(G)}$, and $f_{\text{best}}^{(G)}$, $f_{\text{worst}}^{(G)}$, and $f_i^{(G)}$ are the objective function values of the best, worst, and ith individual, respectively.

In this respect, Figure 3 illustrates the changes of the proposed adaptive scale factor $F_i^{(G)}$ defined in Equation (31) during the evolution process, where $F_{\max} = 0.9$ and $F_{\min} = 0.5$.

According to Figure 3, it is evident that in the early stage, the adaptive scale factor $F_i^{(G)}$ has a larger value, which is suitable for exploration. During the later stage, with the increase of generations, the adaptive scale factor $F_i^{(G)}$ starts to decrease, which improves exploitation and accelerates the local convergence of the algorithm.

6.2.2. Adaptive Crossover Rate. According to the analysis in [18, 40], adapting the suitable value of the crossover rate can maintain the diversity of the population and improve the quality of the solution. From Equation (30), it is evident that for the large value of the crossover rate, the mutant vector $\mathbf{v}_i^{(G)}$ has a greater contribution to the trial vector $\mathbf{u}_i^{(G)}$. In this case, a higher crossover rate increases the population diversity and enhances the global search. In contrast, for a smaller crossover rate, any trial vector $\mathbf{u}_i^{(G)}$ keeps the previous state $\mathbf{x}_i^{(G)}$ with a large probability. This can further refine the trial vector, which is beneficial for improving the quality of the solution.

In this regard, an adaptive crossover rate $CR^{(G)}$ is proposed here, to address the above mentioned issues, which is defined as

$$CR^{(G)} = (CR_{\max} - CR_{\min}) \cdot 2^{-\exp(3-(G_{\max}/(G+1)))} + CR_{\min}, \tag{32}$$

where $G_{\max}$ is the maximum number of generations and $CR_{\max}$ and $CR_{\min}$ are the maximum and minimum values of $CR^{(G)}$, respectively.

In this way, Figure 4 shows the proposed adaptive crossover rate $CR^{(G)}$ versus the number of generations, where $CR_{\max} = 0.9$ and $CR_{\min} = 0.1$.

As can be seen from Figure 4, the $CR^{(G)}$ has a large value in the initial stage of the evolution process and gradually decreases with the increase of generations. In this way, larger $CR^{(G)}$ can advance the population diversity and strongly enhance the exploration ability. During the later stage of the evolution process, the $CR^{(G)}$ takes a small value, which is beneficial for improving the quality of the solution.

6.2.3. The Adaptive Mutation Operator. The performance of the DE algorithm, such as convergence, population diversity, and exploration ability, is greatly affected by mutation operators. The appropriate mutation operators for different evolutionary stages have been proposed here, with the aim to avoid premature convergence and prevent stagnation. In this manner, the DE/rand/1 and DE/rand/2 have strong global exploration ability, while the DE/best/1 operator has a good local exploitation ability [20]. In order to further improve the local search ability, the DE/current-to-best/1 mutation operator can be employed [20].

Based on the above considerations, to dynamically adjust the global exploration and local exploitation abilities, this paper proposes an adaptive adjustment parameter $\delta^{(G)}$, as follows:

$$\delta^{(G)} = \left|\frac{f_{\text{best}}^{(G)} - f_{\text{mean}}^{(G)}}{f_{\text{best}}^{(G)} - f_{\text{worst}}^{(G)}}\right|, \tag{33}$$

where $f_{\text{mean}}^{(G)}$ is mean value of the objective function. Thus, the pseudocode of the proposed adaptive mutation operator is shown in Algorithm 1.

As a matter of fact, the adaptive parameter $\delta^{(G)} \in [0, 1]$ has an important influence on the identification of evolutionary stages in the search process. According to Equation (33), in the early stage, the value of $\delta^{(G)}$ is close to 1, which indicates that the population is far from the region of the global optimum, and this corresponds to the global exploration in the search space. Hence, the DE/rand/1 and DE/rand/2 will be chosen randomly with the probability of 0.5, in order to improve the exploration and find the region of the global optimum. In the later stage, the value of $\delta^{(G)}$ is close to 0, which shows that the population is near the region of the global optimum, and this corresponds to the local exploitation. Therefore, the DE/best/1 and DE/current-to-best/1 will be randomly selected, with the probability of 0.5, to strengthen the exploitation ability and further improve the solution quality.

7. Nelder-Mead Method

The Nelder-Mead is a local search method that does not require the derivative information of the objective function

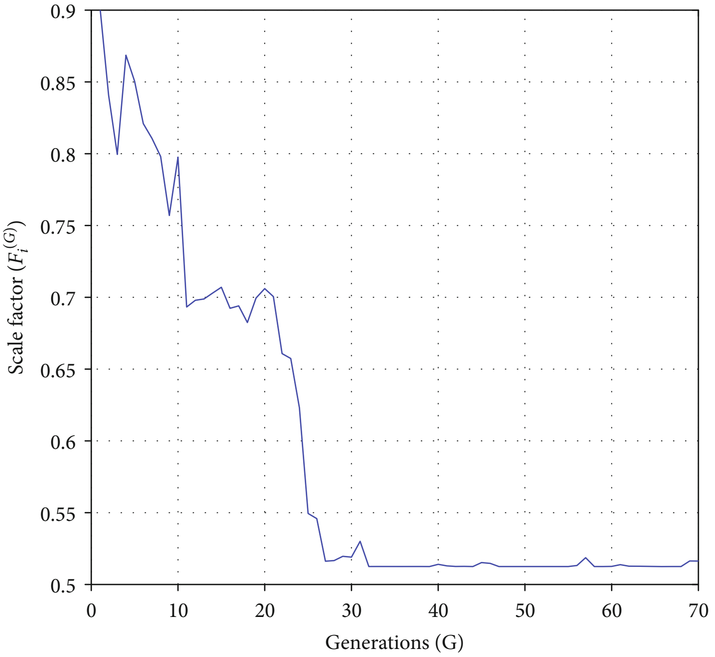

Figure 3: The changes of the proposed adaptive scale factor $F_i^{(G)}$ during the evolution process.

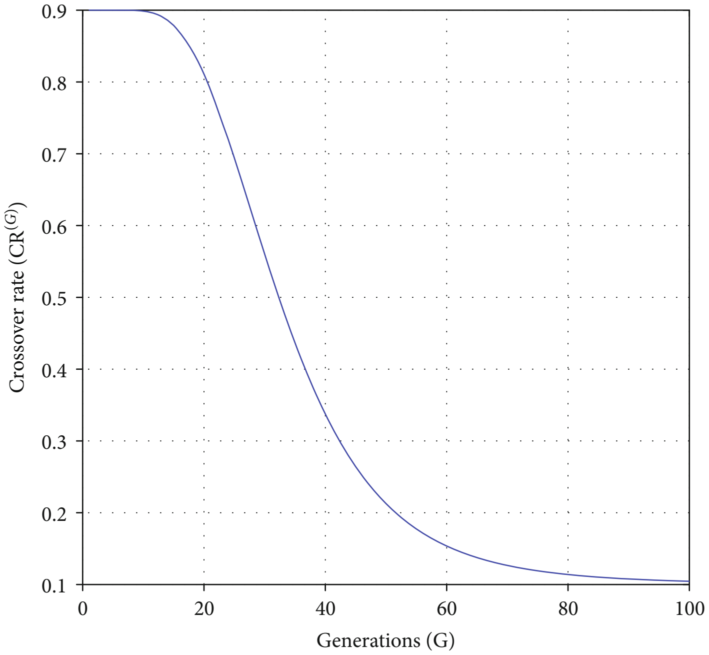

Figure 4: The changes of the proposed adaptive crossover rate $\mathrm{CR}^{(G)}$ during the evolution process.

[25]. In general, the NM method is described for the minimization of an objective function in n-dimensional space. The evolution process of the NM method starts with an initial simplex in the search space, which is a polyhedron with $n+1$ vertices, i.e., $\{\mathbf{x}_i : 1 \le i \le n\}$. The objective function is evaluated at each vertex, and then all vertices are ranked based on their objective function values, i.e., the vertex corresponding to the best objective function value is denoted as $\mathbf{x}_1$ and the vertex corresponding to the worst objective function value is denoted as $\mathbf{x}_{n+1}$. The NM method uses four elementary geometric transformations, called reflection, expansion, contraction, and shrinkage. Using these geometric transformations, the simplex moves through the search space towards the optimal solution. The following steps are executed repeatedly until reaching stopping criteria:

```
if $\delta^{(G)} > 0.5$ then
    if rand > 0.5 then
        $\mathbf{v}_i^{(G)} = \mathbf{x}_{r1} + F_i^{(G)}(\mathbf{x}_{r2} - \mathbf{x}_{r3})$
    else
        $\mathbf{v}_i^{(G)} = \mathbf{x}_{r1} + F_i^{(G)}(\mathbf{x}_{r2} - \mathbf{x}_{r3}) + F_i^{(G)}(\mathbf{x}_{r4} - \mathbf{x}_{r5})$
    end if
else if $\delta^{(G)} > 0.5$ then
    if rand > 0.5 then
        $\mathbf{v}_i^{(G)} = \mathbf{x}_{\text{best}} + F_i^{(G)}(\mathbf{x}_{r1} - \mathbf{x}_{r2})$
    else
        $\mathbf{v}_i^{(G)} = \mathbf{x}_i + F_i^{(G)}(\mathbf{x}_{\text{best}} - \mathbf{x}_i) + F_i^{(G)}(\mathbf{x}_{r1} - \mathbf{x}_{r2})$
    end if
end if
```

Algorithm 1: Pseudocode of the adaptive mutation operator.

Step 1. Initialization. Given an initial vertex $\mathbf{x}_1^{(0)}$, obtained by the ADE algorithm, generate the remaining n vertices $\{\mathbf{x}_{i+1}^{(0)} : 1 \leq i \leq n\}$ according to

$$\mathbf{x}_{i+1}^{(0)} = \mathbf{x}_1^{(0)} + \lambda \mathbf{e}_i, \quad \forall i \in \{1, 2, \cdots, n\}, \tag{34}$$

where $\mathbf{e}_i$ is the unit vector of the ith axis and λ is the initial step size, set as $\lambda = 1$.

Step 2. Sorting. Rank the vertices as follows

$$f\left(\mathbf{x}_1^{(k)}\right) \leq \cdots \leq f\left(\mathbf{x}_{n+1}^{(k)}\right), \tag{35}$$

where k is the current iteration. Here, the objective function value is obtained from the ML objective function in Equation (8), as follows: $f(\mathbf{x}_i^{(k)}) = J_{\text{ML}}(\mathbf{x}_i^{(k)})$.

Step 3. Reflection. Generate the vertex $\mathbf{x}_r^{(k)}$ by reflecting the vertex $\mathbf{x}_{n+1}^{(k)}$, as follows:

$$\mathbf{x}_r^{(k)} = \bar{\mathbf{x}}^{(k)} + \alpha\left(\bar{\mathbf{x}}^{(k)} - \mathbf{x}_{n+1}^{(k)}\right), \tag{36}$$

where $\alpha > 0$ is the reflection coefficient, usually suggested as $\alpha = 1$ and

$$\bar{\mathbf{x}}^{(k)} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i^{(k)}, \tag{37}$$

is the centroid of the n best vertices. If $f(\mathbf{x}_1^{(k)}) \leq f(\mathbf{x}_r^{(k)}) < f(\mathbf{x}_n^{(k)})$, replace $\mathbf{x}_{n+1}^{(k+1)}$ with $\mathbf{x}_r^{(k)}$ and terminate the iteration. If $f(\mathbf{x}_r^{(k)}) < f(\mathbf{x}_1^{(k)})$, continue with the expansion; otherwise, perform contraction.

Step 4. Expansion. Calculate the vertex $\mathbf{x}_e^{(k)}$ in the same direction as $\mathbf{x}_r^{(k)}$, as follows:

$$\mathbf{x}_e^{(k)} = \bar{\mathbf{x}}^{(k)} + \beta\left(\mathbf{x}_r^{(k)} - \bar{\mathbf{x}}^{(k)}\right), \tag{38}$$

where $\beta > 1$ is the expansion coefficient, usually suggested as $\beta = 2$. If $f(\mathbf{x}_e^{(k)}) < f(\mathbf{x}_r^{(k)})$, replace the vertex $\mathbf{x}_{n+1}^{(k+1)}$ with $\mathbf{x}_e^{(k)}$ and terminate the iteration; otherwise, replace $\mathbf{x}_{n+1}^{(k+1)}$ with $\mathbf{x}_r^{(k)}$ and terminate the iteration.

Step 5. Contraction. After reflection, there are two possible contraction cases:

Step 5.1. Outside contraction. If $f(\mathbf{x}_n^{(k)}) \leq f(\mathbf{x}_r^{(k)}) < f(\mathbf{x}_{n+1}^{(k)})$, generate the vertex $\mathbf{x}_c^{(k)}$ as

$$\mathbf{x}_c^{(k)} = \bar{\mathbf{x}}^{(k)} + \gamma\left(\mathbf{x}_r^{(k)} - \bar{\mathbf{x}}^{(k)}\right), \tag{39}$$

where $0 < \gamma < 1$ is the contraction coefficient, usually suggested as $\gamma = 0.5$. If $f(\mathbf{x}_c^{(k)}) \leq f(\mathbf{x}_r^{(k)})$, replace $\mathbf{x}_{n+1}^{(k+1)}$ with $\mathbf{x}_c^{(k)}$ and terminate the iteration; otherwise, perform shrinkage.

Step 5.2. Inside contraction. If $f(\mathbf{x}_r^{(k)}) \geq f(\mathbf{x}_{n+1}^{(k)})$, generate the vertex $\mathbf{x}_{cc}^{(k)}$ as follows:

$$\mathbf{x}_{cc}^{(k)} = \bar{\mathbf{x}}^{(k)} - \gamma\left(\mathbf{x}_r^{(k)} - \bar{\mathbf{x}}^{(k)}\right). \tag{40}$$

If $f(\mathbf{x}_{cc}^{(k)}) < f(\mathbf{x}_{n+1}^{(k)})$, replace $\mathbf{x}_{n+1}^{(k+1)}$ with $\mathbf{x}_{cc}^{(k)}$ and terminate the iteration; otherwise, do shrinkage.

Step 6. Shrinkage. Perform the shrinkage on the vertices $\forall i \in \{2, \cdots, n+1\}$ as follows:

$$\mathbf{x}_i^{(k+1)} = \mathbf{x}_1^{(k)} + \delta\left(\mathbf{x}_i^{(k)} - \mathbf{x}_1^{(k)}\right), \tag{41}$$

where $0 < \delta < 1$ is the shrinkage coefficient, usually suggested as $\delta = 0.5$. The optimization process described in Steps 1–6 is repeated until the termination criteria are reached. The obtained optimal solution represents the estimated position $\hat{\mathbf{x}}$ of the unknown target position $\mathbf{x}$.

8. HADENM Algorithm

In this section, the proposed HADENM algorithm is introduced that hybridizes the ADE algorithm with the local search NM method, in order to efficiently solve the passive target localization problem given in Equation (8). In this way, the pseudocode of the HADENM algorithm is presented in Algorithm 2, for the passive target localization problem.

9. Cramer-Rao Lower Bound

In the passive target localization problem, the CRLB can be used as a benchmark for evaluating the performance of

```
Initialize the parameters: $N_P$, $G_{\max}$, $F_{\max}$, $F_{\min}$, $CR_{\max}$, $CR_{\min}$, $\alpha$, $\beta$, $\gamma$
Generate a uniformly distributed random population
while ADE termination condition is not satisfied do
  for $i = 1$ to $N_P$ do
    Compute the adaptive parameters $F_i^{(G)}$ and $CR^{(G)}$
    Compute $\delta^{(G)}$
    if $\delta^{(G)} > 0.5$ then
      if rand $> 0.5$ then
        $\mathbf{v}_i^{(G)} \longleftarrow \mathbf{x}_{r1} + F_i^{(G)}(\mathbf{x}_{r2} - \mathbf{x}_{r3})$
      else
        $\mathbf{v}_i^{(G)} \longleftarrow \mathbf{x}_{r1} + F_i^{(G)}(\mathbf{x}_{r2} - \mathbf{x}_{r3}) + F_i^{(G)}(\mathbf{x}_{r4} - \mathbf{x}_{r5})$
      end if
    else if $\delta^{(G)} \le 0.5$ then
      if rand $> 0.5$ then
        $\mathbf{v}_i^{(G)} \longleftarrow \mathbf{x}_{\text{best}} + F_i^{(G)}(\mathbf{x}_{r1} - \mathbf{x}_{r2})$
      else
        $\mathbf{v}_i^{(G)} \longleftarrow \mathbf{x}_i + F_i^{(G)}(\mathbf{x}_{\text{best}} - \mathbf{x}_i) + F_i^{(G)}(\mathbf{x}_{r1} - \mathbf{x}_{r2})$
      end if
    end if
    if $\text{rand}_{j,i} \leqslant CR^{(G)} \vee j = j_{\text{rand}}$ then
      $u_{i,j}^{(G)} \longleftarrow v_{i,j}^{(G)}$
    else
      $u_{i,j}^{(G)} \longleftarrow x_{i,j}^{(G)}$
    end if
    if $f(\mathbf{u}_{i,j}^{(G)}) \leqslant f(\mathbf{x}_{i,j}^{(G)})$ then
      $\mathbf{x}_i^{(G+1)} \longleftarrow \mathbf{u}_i^{(G)}$
    else
      $\mathbf{x}_i^{(G+1)} \longleftarrow \mathbf{x}_i^{(G)}$
    end if
  end for
end while
Initialize the NM with the current best solution found by ADE
while NM termination condition is not satisfied do
  Sort the vertices according to $f(\mathbf{x}_1^{(k)}) \le \cdots \le f(\mathbf{x}_{n+1}^{(k)})$
  Perform reflection
  if $f(\mathbf{x}_1^{(k)}) \le f(\mathbf{x}_r^{(k)}) < f(\mathbf{x}_n^{(k)})$ then
    $\mathbf{x}_{n+1}^{(k+1)} \longleftarrow \mathbf{x}_r^{(k)}$
  end if
  Perform expansion
  if $f(\mathbf{x}_e^{(k)}) < f(\mathbf{x}_r^{(k)})$ then
    $\mathbf{x}_{n+1}^{(k+1)} \longleftarrow \mathbf{x}_e^{(k)}$
  else
    $\mathbf{x}_{n+1}^{(k+1)} \longleftarrow \mathbf{x}_r^{(k)}$
  end if
  if $f(\mathbf{x}_n^{(k)}) \le f(\mathbf{x}_r^{(k)}) < f(\mathbf{x}_{n+1}^{(k)})$ then
    Perform outside contraction
    if $f(\mathbf{x}_c^{(k)}) \le f(\mathbf{x}_r^{(k)})$ then
      $\mathbf{x}_{n+1}^{(k+1)} \longleftarrow \mathbf{x}_c^{(k)}$
    else
      Perform Shrinkage
    end if
  else if $f(\mathbf{x}_r^{(k)}) \ge f(\mathbf{x}_{n+1}^{(k)})$ then
    Perform inside contraction
```

Algorithm 2: Continued.

```
            if $f(\mathbf{x}_{cc}^{(k)}) < f(\mathbf{x}_{n+1}^{(k)})$ then
                $\mathbf{x}_{n+1}^{(k+1)} \longleftarrow \mathbf{x}_{cc}^{(k)}$
            else
                Perform Shrinkage
            end if
        end if
    end while
    Display the results
```

Algorithm 2: Pseudocode of the HADENM algorithm.

different algorithms. The CRLB is obtained from the diagonal elements of the inverse of the Fisher information matrix (FIM) [26], denoted by $\mathbf{I}(\mathbf{x})$, which is based on the probability density function $f(\mathbf{r}|\mathbf{x})$, defined in Equation (6), and is represented as

$$\begin{aligned}\mathbf{I}(\mathbf{x}) &= E\left[\left(\frac{\partial \ln\left(f(\mathbf{r}|\mathbf{x})\right)}{\partial \mathbf{x}}\right)\left(\frac{\partial \ln\left(f(\mathbf{r}|\mathbf{x})\right)}{\partial \mathbf{x}}\right)^{T}\right] \\ &= -E\left[\frac{\partial^2 \ln(f(\mathbf{r}|\mathbf{x}))}{\partial \mathbf{x}\partial \mathbf{x}^T}\right].\end{aligned} \tag{42}$$

Based on this, the FIM can be defined as

$$\mathbf{I}(\mathbf{x}) = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{yx} & I_{yy} \end{bmatrix}, \tag{43}$$

where the corresponding elements are given as follows:

$$I_{xx} = \sum_{i=1}^{N} \frac{1}{\sigma_{ni}^2}\left(\frac{x - x_i^r}{\|\mathbf{x} - \mathbf{x}_i^r\|_2} + \frac{x}{\|\mathbf{x}\|_2}\right)^2, \tag{44}$$

$$I_{xy} = I_{yx} = \sum_{i=1}^{N} \frac{1}{\sigma_{ni}^2}\left(\frac{x - x_i^r}{\|\mathbf{x} - \mathbf{x}_i^r\|_2} + \frac{x}{\|\mathbf{x}\|_2}\right)\left(\frac{y - y_i^r}{\|\mathbf{x} - \mathbf{x}_i^r\|_2} + \frac{y}{\|\mathbf{x}\|_2}\right), \tag{45}$$

$$I_{yy} = \sum_{i=1}^{N} \frac{1}{\sigma_{ni}^2}\left(\frac{y - y_i^r}{\|\mathbf{x} - \mathbf{x}_i^r\|_2} + \frac{y}{\|\mathbf{x}\|_2}\right)^2. \tag{46}$$

Hence, the relationship between the variance and CRLB can be determined as

$$E\left[(\hat{\mathbf{x}} - \mathbf{x})(\hat{\mathbf{x}} - \mathbf{x})^T\right] \geq \operatorname{tr}\left\{\mathbf{I}(\mathbf{x})^{-1}\right\} = \operatorname{CRLB}(\mathbf{x}), \tag{47}$$

where $\hat{\mathbf{x}}$ is the estimated value of $\mathbf{x}$.

10. Experimental Study

In this section, experiments are conducted to evaluate the localization performance and to perform the analysis of the benefits of the proposed modifications on the optimization performance of the HADENM algorithm. In this regard, the presentation of the experimental results is divided into two subsections, described below.

10.1. A Parametric Study on HADENM Algorithm. In this subsection, the experiments are carried out to evaluate the performance of the HADENM algorithm by studying the effects of the proposed adaptive scale factor and crossover rate, as well as the proposed adaptive mutation operator on the optimization performance. Furthermore, the effects of the hybridization between the ADE and NM algorithms have been analysed. Then, the experiments are performed in order to verify that the performance is enhanced after combining the previously described improvements.

To evaluate the performance of the proposed HADENM algorithm, the solution error measure $(f(\hat{\mathbf{x}}) - f(\mathbf{x}^*))$ has been employed, where $\hat{\mathbf{x}}$ denotes the best solution of the algorithm obtained in one run and $\mathbf{x}^*$ represents the well-known global optimal solution of the corresponding ML objective function. All experiments for each algorithm have been independently run 30 times and statistical results are provided. From the statistical point of view, the quality of the obtained solutions has been analysed and compared using two nonparametric statistical hypothesis tests such as the Wilcoxon signed-rank test and Friedman test.

Firstly, the Wilcoxon signed-rank test can be used to determine the significant differences between two samples obtained by the algorithms. This statistic test has been applied with a significance level of $\alpha = 0.05$. Here, R^+ denotes the sum of ranks for the problem in which the first algorithm outperformed the second and R^- is the sum of ranks for the problem where the first algorithm performed worse than the second. According to [41], in the Wilcoxon signed-rank test for the null hypothesis, it is assumed that, "*there is no difference between the mean results of the two samples.*" On the contrary, the alternative hypothesis is that, "*there is a difference in the mean results of the two samples.*" In the statistical analysis, the p value is used and compared to the significance level. Thus, the null hypothesis can be rejected when the p value is less than or equal to $\alpha = 0.05$. Based on the obtained results of the statistical test, one of the following three signs (+, −, and ≈) has been assigned for the comparison between any two algorithms. Thus, the plus (+) sign denotes that the first algorithm is significantly better than the second, the minus (−) sign means that the first algorithm is significantly worse than the second, and the approximation (≈) sign denotes that there is no significant difference between two algorithms.

The second test is the Friedman test, which obtains the ranks of all considered algorithms over every tested function, with the aim to find the significant differences in performance between two or more algorithms. In this statistical test, the algorithm with a minimum rank value is considered as the best performing algorithm, while the one with the highest rank value is considered as the worst. According to [41], the null hypothesis for Friedman test states that, "*there is no difference among the performance of all algorithms*," whereas the alternative hypothesis states that, "*there is a difference among the performance of all algorithms*".

In this regard, the presentation of the obtained results of the considered evaluations is divided into four sub-subsections. In the first sub-subsection, the effectiveness of the proposed adaptive scale factor and crossover rate has been evaluated. In the second sub-subsection, the effectiveness of the proposed adaptive mutation operator has been considered. The third sub-subsection considers the hybridization of ADE and NM algorithms and the overall performance improvements. Finally, in the fourth sub-subsection, the statistical results of applying the Friedman test between the considered algorithms have been analysed.

10.1.1. Effectiveness of the Adaptive Scale Factor and Crossover Rate. In this sub-subsection, the experimental studies have been performed to evaluate the effectiveness and benefits of the proposed adaptive scale factor $F_i^{(G)}$ (given in Equation (31)) and adaptive crossover rate $CR^{(G)}$ (given in Equation (32)) on the optimization performance of the HADENM algorithm. Firstly, the effectiveness of the adaptive scale factor is considered, where the performance of the HADENM algorithm is evaluated with two different fixed values of F, as follows:

(1) HADENM$_{F=0.5}$, which has the same operators as HADENM, except that the scale factor is set to a fixed value of $F = 0.5$

(2) HADENM$_{F=0.9}$, which has the same operators as HADENM, except that the scale factor is set to a fixed value of $F = 0.9$

The summary of statistical results of applying the Wilcoxon test between the proposed HADENM and the above two algorithms is presented in Table 1.

From the results in Table 1, it can be seen that the proposed HADENM algorithm is significantly better than HADENM$_{F=0.5}$. On the other hand, in the case of HADENM versus HADENM$_{F=0.9}$, the proposed algorithm has a higher R^+ value than R^-. This shows the effectiveness of the proposed adaptive scale factor $F_i^{(G)}$ given in Equation (31) and the idea of increasing the scale factor during the evolution process. Hence, the main purpose of this comparison is to show that the HADENM algorithm with the adaptive scale factor $F_i^{(G)}$ can achieve better performance than the same algorithm with a fixed value of F.

Table 1: Results of the Wilcoxon test for HADENM, HADENM$_{F=0.5}$, and HADENM$_{F=0.9}$ at a significance level of 0.05.

Algorithm	R^+	R^-	p value	Dec.
HADENM versus HADENM$_{F=0.5}$	346	119	0.0177	+
HADENM versus HADENM$_{F=0.9}$	282	183	0.3207	≈

Secondly, in order to show the efficiency of the proposed adaptive crossover rate $CR^{(G)}$ in Equation (32), the performance of the HADENM algorithm is compared with three different fixed values of CR, as follows:

(1) HADENM$_{CR=0.1}$, which has the same operators as HADENM, except that the crossover rate is set to a fixed value of $CR = 0.1$

(2) HADENM$_{CR=0.5}$,which has the same operators as HADENM, except that the crossover rate is set to a fixed value of $CR = 0.5$

(3) HADENM$_{CR=0.9}$, which has the same operators as HADENM, except that the crossover rate is set to a fixed value of $CR = 0.9$

Table 2 shows the summary of statistical results of applying the Wilcoxon test between the proposed HADENM and the above three algorithms.

From the results in Table 2, it can be seen that the proposed HADENM algorithm has significantly better performance than HADENM$_{\mathrm{CR}=0.5}$ and HADENM$_{\mathrm{CR}=0.9}$. Furthermore, in the case of HADENM versus HADENM$_{\mathrm{CR}=0.1}$, it can be observed that the proposed algorithm provides higher R^+ values than R^- in all considered cases. This shows that the proposed adaptive crossover rate $CR^{(G)}$ plays a vital role in determining the optimal crossover rate value for the considered optimization problem.

10.1.2. Effectiveness of the Proposed Adaptive Mutation Operator. In this sub-subsection, to evaluate the effectiveness and improvement of the proposed adaptive mutation operator on the optimization performance of the HADENM algorithm, the experimental studies have been performed. In this way, two different versions of the HADENM algorithm have been tested and compared against the proposed one, such as

(1) HADENM-1, which has the same operators as HADENM, except that only the explorative mutations of the proposed adaptive mutation operator are applied. Hence, only the mutation operators DE/rand/1 and DE/rand/2 will be chosen randomly with the probability of 0.5

(2) HADENM-2, which has the same operators as HADENM, except that only the exploitative mutations of the proposed adaptive mutation operator are applied. In this case, the mutation operators DE/best/1 and DE/current-to-best/1 will be randomly selected, with the probability of 0.5

Table 2: Results of the Wilcoxon test for HADENM, $\text{HADENM}_{\text{CR}=0.1}$, $\text{HADENM}_{\text{CR}=0.5}$, and $\text{HADENM}_{\text{CR}=0.9}$ at a significance level of 0.05.

Algorithm	R^+	R^-	p value	Dec.
HADENM versus $\text{HADENM}_{\text{CR}=0.1}$	264.5	150	0.7761	≈
HADENM versus $\text{HADENM}_{\text{CR}=0.5}$	253	72	0.0135	+
HADENM versus $\text{HADENM}_{\text{CR}=0.9}$	259	66	0.0079	+

In Table 3, the summary of statistical results of applying the Wilcoxon test between the proposed HADENM, HADENM-1, and HADENM-2 algorithms is shown.

According to the Wilcoxon test, given in Table 3, it can be observed that the proposed HADENM algorithm provides higher R^+ values than R^- in both cases. The obtained results indicate that the proposed adaptive mutation operator can effectively keep the balance between exploration and exploitation abilities, thus improving the overall optimization performance of the HADENM algorithm.

10.1.3. Effectiveness of the Proposed Hybridization. To study the effects of the proposed hybridization between the ADE and NM algorithms, in this sub-subsection, the following experiment has been performed. In this regard, the performance of the HADENM algorithm without using the NM method has been compared to the performance of the proposed HADENM algorithm. Thus, the summary of statistical results of applying the Wilcoxon test between the two abovementioned algorithms is shown in Table 4.

According to the obtained statistical results in Table 4, it can be concluded that the proposed HADENM algorithm has better performance compared to the algorithm without using the NM method. This shows that the NM method can further enhance the quality of the obtained solution, and in this way, it can improve the optimization performance of the HADENM algorithm.

10.1.4. Comparison of the Proposed Improvements. In this sub-subsection, the experimental studies have been performed in order to demonstrate the effectiveness of the proposed HADENM algorithm. As there are more than two algorithms for comparison, the overall performance of the considered algorithms has been compared using the Friedman test. In this regard, Table 5 shows the average ranks according to the Friedman test for the considered algorithms, using different values of the variance of the noise 10 log (σ_{ni}^2), where the proposed HADENM algorithm is selected as the base algorithm. The best ranks are shown in bold, and the second ranks are in italics.

From Table 5, it can be noted that the p values computed through the Friedman test for different values of the variance of the noise 10 log (σ_{ni}^2) are less than 0.05. Therefore, this means that there is a significant difference in performance between the considered algorithms. Furthermore, it can be observed that the proposed HADENM algorithm outperforms other considered algorithms for all values of 10 log (σ_{ni}^2), which demonstrates the effectiveness of the modifications proposed in this paper.

Table 3: Results of the Wilcoxon test for HADENM, HADENM-1, and HADENM-2 at a significance level of 0.05.

Algorithm	R^+	R^-	p value	Dec.
HADENM versus HADENM-1	314.5	105.5	0.1019	≈
HADENM versus HADENM-2	283	182	0.2915	≈

Table 4: Results of the Wilcoxon test for the HADENM algorithm without using the NM method and the proposed HADENM algorithm at a significance level of 0.05.

Algorithm	R^+	R^-	p value	Dec.
HADENM versus ADE	465	0	0.000	+

Table 5: Average ranks computed through the Friedman test for all algorithms across different values of the variance of the noise 10 log (σ_{ni}^2), at the significance level of 0.05.

Algorithm	-40	-20	20	40	Mean ranking	Rank
HADENM	**3.07**	**3.30**	**3.37**	**2.98**	**3.27**	1
HADENM-2	*3.87*	*3.70*	*3.47*	*3.02*	*3.63*	2
HADENM-1	4.00	3.93	4.10	4.67	4.22	3
$\text{HADENM}_{\text{CR}=0.1}$	4.73	4.53	3.80	4.28	4.27	4
$\text{HADENM}_{F=0.9}$	4.03	5.07	4.80	4.68	4.64	5
$\text{HADENM}_{\text{CR}=0.5}$	4.80	4.87	5.33	5.08	4.94	6
$\text{HADENM}_{F=0.5}$	5.57	5.00	5.47	5.35	5.34	7
$\text{HADENM}_{\text{CR}=0.9}$	5.93	5.60	5.67	5.93	5.69	8
Friedman p value	0.000	0.003	0.000	0.000		

10.2. Localization Performance. The simulations are performed for the passive target localization system in the LOS environment, which consists of one transmitter, four receivers located at known positions, and the passive target, which is located at a different position for each simulation scenario. The transmitter is located at $\mathbf{x}_T = [0\,0]^T$ m, while the receiver positions are $\mathbf{x}_1^r = [60\,60]^T$ m, $\mathbf{x}_2^r = [60 - 60]^T$ m, $\mathbf{x}_3^r = [-60 - 60]^T$ m, and $\mathbf{x}_4^r = [-60\,60]^T$ m, respectively. To evaluate the effectiveness of the HADENM algorithm to the measurement error, three scenarios are considered: (i) the position of the target is inside the convex hull formed by the set of receivers at the position $\mathbf{x} = [5\,25]^T$ m, (ii) the target is located outside of the convex hull at the position $\mathbf{x} = [80\,50]^T$ m, and (iii) the position of the target is deployed randomly over an area $80 \times 80\,\text{m}^2$, and its position is different for each simulation run.

Here, the RMSE is employed to evaluate the localization accuracy, which is expressed as

$$\mathrm{RMSE} = \sqrt{\frac{1}{N_m}\sum_{n=1}^{N_m}\left\|\widehat{\mathbf{x}}^{(n)} - \mathbf{x}\right\|_2^2}, \qquad (48)$$

where the estimated value of the true target position is $\widehat{\mathbf{x}}^{(n)}$ and $N_m = 1000$ is the number of simulation runs for a given variance of the noise σ_{ni}^2.

Figure 5 shows the results of the first scenario, where the RMSEs of the considered algorithms versus $p = 10\log(\sigma_{ni}^2)$ are plotted and compared against the CRLB.

From Figure 5, it can be observed that the RMSE performance of the HADENM algorithm is better compared to the DE and the CWLS, and the proposed algorithm attains the CRLB for all the considered ranges of p. Moreover, the DE and the CWLS algorithms can achieve the RMSE performance several dBs above the CRLB, for small measurement noise. However, as the noise level increases, i.e., $p \geq 20$ dB, the CWLS algorithm separates faster from the CRLB accuracy.

The simulation results of the second scenario are presented in Figure 6, where the RMSEs of all the considered algorithms versus measurement noise p are plotted.

According to the simulation results in Figure 6, it can be observed that the HADENM algorithm attains the CRLB accuracy and has a superior localization performance compared to the other considered algorithms. Furthermore, it is concluded that there is a significant degradation in the localization accuracy of the CWLS and DE algorithms compared to the first simulation scenario. It is also noted that, the RMSE of the CWLS algorithm diverges rapidly from the CRLB for large values of the measurement noise, i.e., when $p \geq 5$ dB.

Moreover, the RMSEs of the considered algorithms versus p are plotted in Figure 7 for the third simulation scenario, where the true position of the passive target is randomly generated within the considered area for each simulation run.

As expected, the results of the third simulation scenario, presented in Figure 7, show that the HADENM algorithm attains the CRLB accuracy and provides superior performance over the CWLS and DE algorithms. Therefore, summarizing the results of the three simulation scenarios, presented in Figures 5–7, it can be concluded that the proposed HADENM algorithm has a better localization performance compared to the considered algorithms and successfully attains the CRLB for every considered scenario. Furthermore, it is observed that the proposed algorithm is less sensitive to the changes in the geometric configuration of the transmitter, receivers, and target.

For a better comparison of the simulation results, the cumulative distribution functions (CDFs) of the passive target localization error of the considered algorithms are investigated, with different variances of the measurement noise σ_{ni}^2. The localization error (LE) is defined as the Euclidean distance between the estimated and the true position of the passive target, i.e.,

$$\mathrm{LE} = \left\|\widehat{\mathbf{x}}^{(n)} - \mathbf{x}\right\|_2, \quad \forall n \in \{1, \cdots, N_m\}. \qquad (49)$$

Figure 8 represents the simulation results for the second scenario with the corresponding CDFs in terms of the localization error obtained for each algorithm for different variances of the measurement noise σ_{ni}^2.

From the CDFs of the localization error, depicted in Figure 8, it can be observed that for a fixed CDF percentage, e.g., 90%, under the measurement noise $\sigma_{ni}^2 = 1\,\mathrm{m}^2$, the HADENM algorithm has the lowest localization error of 1.2 m, while the DE and CWLS algorithms have 1.3 m and 1.5 m, respectively. When the value of the measurement noise is increased to $\sigma_{ni}^2 = 10\,\mathrm{m}^2$, the HADENM algorithm has the localization error of 3.8 m, while the DE and the CWLS show localization errors of 3.9 m and 4.2 m, respectively. For the higher value of measurement noise, e.g., $\sigma_{ni}^2 = 20\,\mathrm{m}^2$, the HADENM algorithm has a localization error of 5.6 m, while the DE and CWLS algorithms have 5.8 m and 6.1 m, respectively. Based on the above, it is evident that the HADENM algorithm has a smaller localization error compared to the other considered algorithms.

Finally, the influence of increasing the number of receivers on the localization accuracy of different algorithms is investigated for the first simulation scenario. In this regard, an arrangement of $N^j = 21$ receivers uniformly distributed on a circle of radius $R = 60\sqrt{2}$ m is considered, where the coordinates of the ith receiver are obtained as follows: $\mathbf{x}_i^r = [R\cos\varphi_i \quad R\sin\varphi_i]^T, \quad \forall i \in \{1, 2, \cdots, N^j\}$. In this way, Figure 9 shows the RMSE performances of the considered algorithms versus the number of receivers, when the variance of the measurement noise is $\sigma_{ni}^2 = 1\,\mathrm{m}^2$.

From Figure 9, it is observed that with the increase of the number of receivers from 4 to 15, the RMSE performances of the CWLS, DE, and HADENM algorithms improve significantly. Further increase in the number of receivers does not explicitly enhance the localization accuracy. Thus, the difference between the RMSE values of the considered algorithms is smaller. Therefore, based on the obtained results, it can be concluded that the proposed HADENM algorithm is robust against variations in the topology and provides a superior performance among all the considered algorithms.

10.2.1. Computational Complexity of the Considered Algorithms. In this sub-subsection, the computational complexity of the HADENM algorithm and other considered algorithms is presented, and the analysis of the average computation time is also given for the comparison. It should be noted that in this paper, the computational complexity of the considered algorithms is only analysed in a single generation. It has been previously shown that the computational complexity of the CWLS algorithm is $O(G_{\max}(n+1))$ [8], while the conventional DE algorithm has the complexity of $O(G_{\max}N_P n)$ [42]. During one generation of the proposed HADENM algorithm, all the individuals in the population are sorted according to the objective function value, where the average computational complexity of this process is $O(N_P\log(N_P))$. Then, all individuals in the population

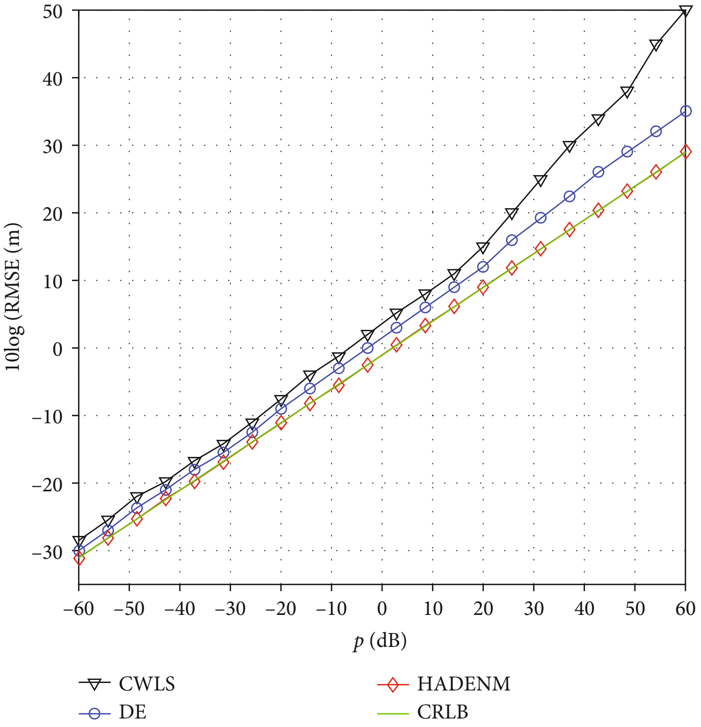

Figure 5: RMSEs versus p for a passive target at $\mathbf{x} = [5\ 25]^T$ m.

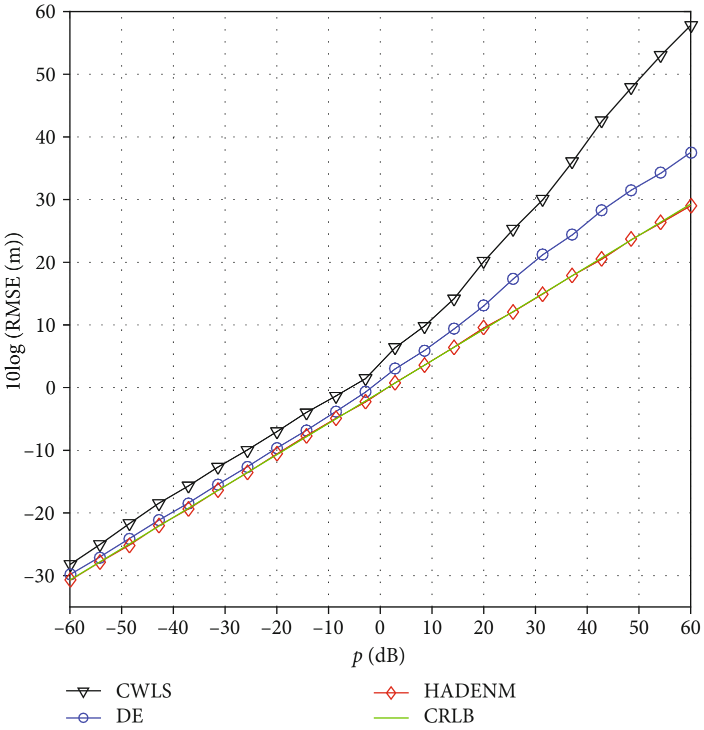

Figure 6: RMSEs versus p for a passive target at $\mathbf{x} = [80\ 50]^T$ m.

go through mutation, crossover, and selection operations, which have the computational complexity approximately equal to $O(N_P(n + Coc + Cos))$ [43]. Here, Coc and Cos denote the costs of crossover and selection operations in the DE algorithm, respectively. Afterwards, the accuracy of the best solution is further improved by the NM algorithm,

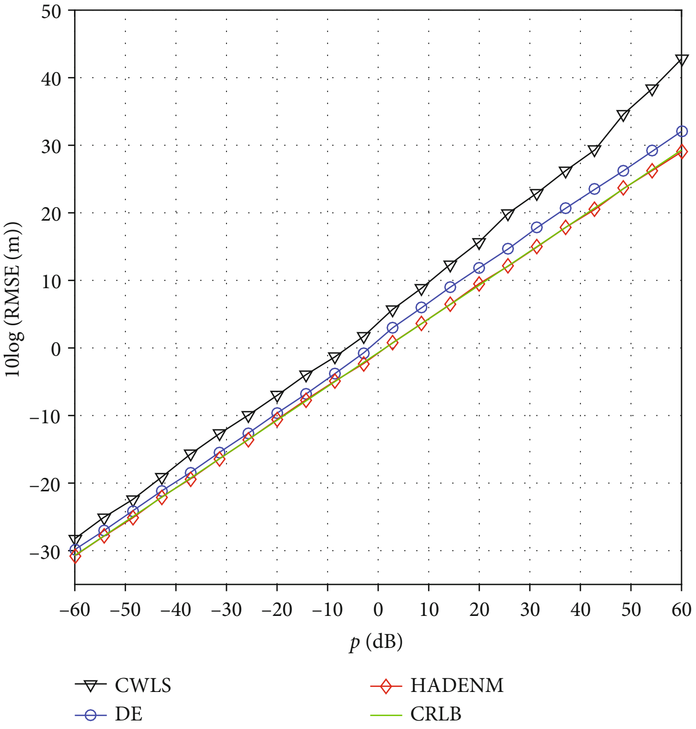

FIGURE 7: RMSEs versus p with a random passive target position.

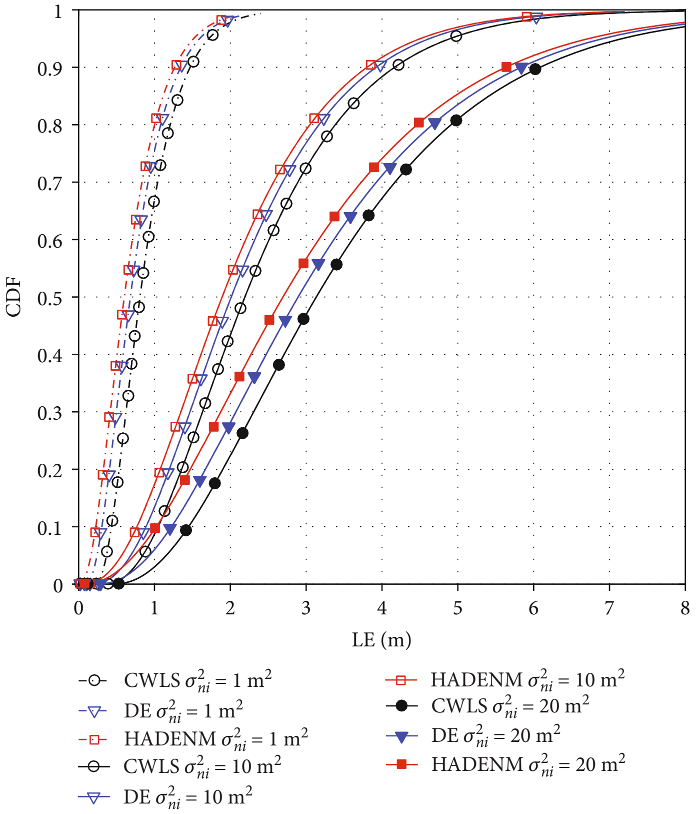

FIGURE 8: CDFs of passive target localization error of the considered algorithms.

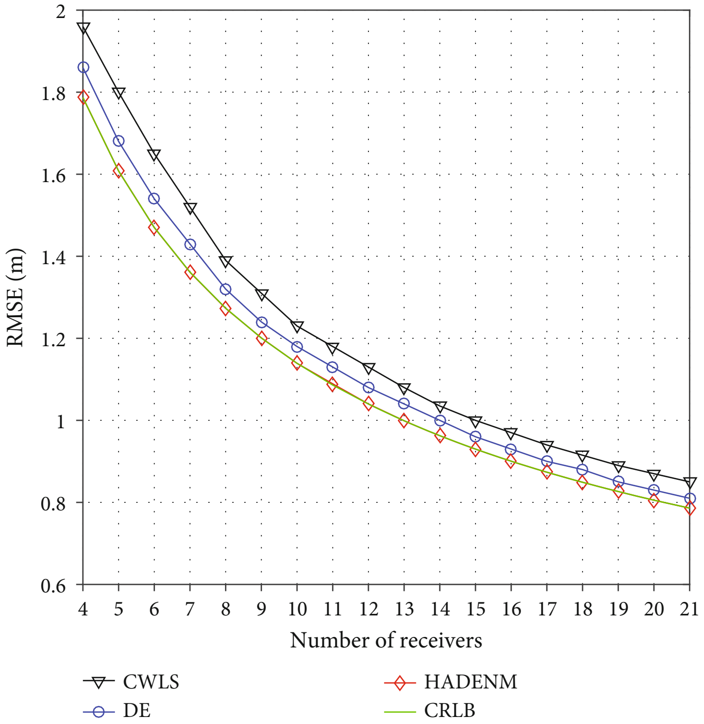

Figure 9: RMSE performances of the considered algorithms versus the number of receivers.

which has the computational complexity $O(n \log n)$ [44]. Based on the above considerations, the computational complexity of the HADENM algorithm can be simplified to $O(G_{\max} N_P n + n \log n)$.

Next, the average computation times in searching for the global optimum have been determined on the same computer with 3.2 GHz CPU and 16 GB of RAM. In this way, based on the considered simulation scenarios, a comparison between the average computation times of the CWLS, DE and HADENM algorithms is shown in Table 6.

Table 6 shows that the CWLS algorithm has the fastest implementation among the considered algorithms, while there is no significant difference between the HADENM and DE algorithms. Based on the results of the analysis of localization accuracy, the results in Table 6 indicate that the proposed HADENM algorithm achieves the best compromise between the localization accuracy and the average computation time in searching for the global optimum.

Table 6: Average computation times of the considered localization algorithms.

	CWLS (ms)	DE (ms)	HADENM (ms)
Scenario 1	0.567	7.1	8.3
Scenario 2	0.466	6.4	7.8
Scenario 3	0.926	8.6	8.7

11. Conclusion

In this paper, the passive target localization problem has been considered, based on TOA measurements, for the system with one transmitter and a set of receivers. In order to solve this nonlinear and nonconvex localization problem even in highly noisy environments, an improved HADENM algorithm, based on the hybridization of the ADE and NM algorithms, has been proposed. An adaptive mutation operator has been developed to provide a good balance between the global exploration and local exploitation abilities. In addition, to further enhance the optimization performance, the control parameters of the ADE are adaptively updated during the evolution process. Furthermore, the exploitation ability is enhanced by the NM method, which improves the accuracy of the best solution previously obtained by the ADE algorithm. To evaluate the benefits of the proposed modifications on the optimization performance, statistical analysis has been conducted.

Based on the comparison results between the HADENM algorithm and its versions, it can be concluded that the modifications proposed in this paper can improve the overall optimization performance. Furthermore, the simulation results show that the proposed HADENM algorithm provides superior localization performance, under different measurement noise conditions, compared to the DE and CWLS algorithms. Moreover, the HADENM algorithm attains the CRLB accuracy and exhibits better robustness against variations in network topology and under high measurement noise. In this way, the HADENM algorithm has the ability to provide both accuracy and robustness compared to the other considered algorithms.

Future research can be focused on further improving the performance of the proposed algorithm and applying it to other complex optimization problems in WSNs.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

The work of M. Simić was supported in part by the Serbian Ministry of Education and Science under Grant (TR32028).

References

[1] J. Sachs, *Handbook of Ultra-Wideband Short-Range Sensing: Theory, Sensors, Applications*, John Wiley & Sons, 2013.

[2] N. Wu, W. Yuan, H. Wang, and J. Kuang, "TOA-based passive localization of multiple targets with inaccurate receivers based on belief propagation on factor graph," *Digital Signal Processing*, vol. 49, pp. 14–23, 2016.

[3] J. Yan, X. Zhang, X. Luo, Y. Wang, C. Chen, and X. Guan, "Asynchronous localization with mobility prediction for underwater acoustic sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 3, pp. 2543–2556, 2018.

[4] J. Xiao, Z. Zhou, Y. Yi, and L. M. Ni, "A survey on wireless indoor localization from the device perspective," *ACM Computing Surveys*, vol. 49, no. 2, pp. 1–31, 2016.

[5] R. Zekavat and R. M. Buehrer, *Handbook of Position Location: Theory, Practice and Advances*, John Wiley & Sons, 2011.

[6] Y. Wang, "Linear least squares localization in sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, no. 1, Article ID 51, 2015.

[7] M. Einemo and H. C. So, "Weighted least squares algorithm for target localization in distributed MIMO radar," *Signal Processing*, vol. 115, pp. 144–150, 2015.

[8] X. Qu and L. Xie, "An efficient convex constrained weighted least squares source localization algorithm based on TDOA measurements," *Signal Processing*, vol. 119, pp. 142–152, 2016.

[9] Z. Li, P.-J. Chung, and B. Mulgrew, "Distributed target localization using quantized received signal strength," *Signal Processing*, vol. 134, pp. 214–223, 2017.

[10] A. W. Mohamed, "An improved differential evolution algorithm with triangular mutation for global numerical optimization," *Computers & Industrial Engineering*, vol. 85, pp. 359–375, 2015.

[11] L. Zhang, L. Liu, X.-S. Yang, and Y. Dai, "A novel hybrid firefly algorithm for global optimization," *PloS One*, vol. 11, no. 9, article e0163230, 2016.

[12] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[13] S. S. Jadon, R. Tiwari, H. Sharma, and J. C. Bansal, "Hybrid artificial bee colony algorithm with differential evolution," *Applied Soft Computing*, vol. 58, pp. 11–24, 2017.

[14] Y. Jiang, M. Liu, T. Chen, and L. Gao, "TDOA passive location based on cuckoo search algorithm," *Journal of Shanghai Jiaotong University (Science)*, vol. 23, no. 3, pp. 368–375, 2018.

[15] G. Wu, X. Shen, H. Li, H. Chen, A. Lin, and P. N. Suganthan, "Ensemble of differential evolution variants," *Information Sciences*, vol. 423, pp. 172–186, 2018.

[16] S. A. El-Quliti, A. H. M. Ragab, R. Abdelaal et al., "A nonlinear goal programming model for university admission capacity planning with modified differential evolution algorithm," *Mathematical Problems in Engineering*, vol. 2015, Article ID 892937, 13 pages, 2015.

[17] A. W. Mohamed and P. N. Suganthan, "Real-parameter unconstrained optimization based on enhanced fitness-adaptive differential evolution algorithm with novel mutation," *Soft Computing*, vol. 22, no. 10, pp. 3215–3235, 2018.

[18] A. W. Mohamed, A. A. Hadi, and K. M. Jambi, "Novel mutation strategy for enhancing SHADE and LSHADE algorithms for global numerical optimization," *Swarm and Evolutionary Computation*, vol. 50, p. 100455, 2019.

[19] S. A. H. ElQuliti and A. W. Mohamed, "A large-scale nonlinear mixed-binary goal programming model to assess candidate locations for solar energy stations: an improved real-binary differential evolution algorithm with a case study," *Journal of Computational and Theoretical Nanoscience*, vol. 13, no. 11, pp. 7909–7921, 2016.

[20] L. Cui, G. Li, Q. Lin, J. Chen, and N. Lu, "Adaptive differential evolution algorithm with novel mutation strategies in multiple sub-populations," *Computers & Operations Research*, vol. 67, pp. 155–173, 2016.

[21] L. Cui, G. Li, Z. Zhu et al., "Adaptive multiple-elites-guided composite differential evolution algorithm with a shift mechanism," *Information Sciences*, vol. 422, pp. 122–143, 2018.

[22] G.-H. Lin, J. Zhang, and Z.-H. Liu, "Hybrid particle swarm optimization with differential evolution for numerical and engineering optimization," *International Journal of Automation and Computing*, vol. 15, no. 1, pp. 103–114, 2018.

[23] J. Zhong and W. Cai, "Differential evolution with sensitivity analysis and the Powell's method for crowd model calibration," *Journal of Computational Science*, vol. 9, pp. 26–32, 2015.

[24] C. Zhang, J. Chen, and B. Xin, "Distributed memetic differential evolution with the synergy of Lamarckian and Baldwinian learning," *Applied Soft Computing*, vol. 13, no. 5, pp. 2947–2959, 2013.

[25] W. Qian, J. Chai, Z. Xu, and Z. Zhang, "Differential evolution algorithm with multiple mutation strategies based on roulette wheel selection," *Applied Intelligence*, vol. 48, no. 10, pp. 3612–3629, 2018.

[26] J. Shen, A. F. Molisch, and J. Salmi, "Accurate passive location estimation using TOA measurements," *IEEE Transactions on Wireless Communications*, vol. 11, no. 6, pp. 2182–2192, 2012.

[27] Y. Miao, H. Wu, and L. Zhang, "The accurate location estimation of sensor node using received signal strength measurements in large-scale farmland," *Journal of Sensors*, vol. 2018, Article ID 2325863, 10 pages, 2018.

[28] S. Xu and K. Doğançay, "Optimal sensor deployment for 3D AOA target localization," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2544–2548, Brisbane, QLD, Australia, April 2015.

[29] S. Tomic, M. Beko, R. Dinis, and P. Montezuma, "Distributed algorithm for target localization in wireless sensor networks

using RSS and AoA measurements," *Pervasive and Mobile Computing*, vol. 37, pp. 63–77, 2017.

[30] L. Song, L. Zhao, and J. Ye, "DV-hop node location algorithm based on GSO in wireless sensor networks," *Journal of Sensors*, vol. 2019, Article ID 2986954, 9 pages, 2019.

[31] S. Subedi and J.-Y. Pyun, "Practical fingerprinting localization for indoor positioning system by using beacons," *Journal of Sensors*, vol. 2017, Article ID 9742170, 16 pages, 2017.

[32] A. Noroozi and M. A. Sebt, "Target localization from bistatic range measurements in multi-transmitter multi-receiver passive radar," *IEEE Signal Processing Letters*, vol. 22, no. 12, pp. 2445–2449, 2015.

[33] Y. Wang, S. Ma, and C. L. P. Chen, "TOA-based passive localization in quasi-synchronous networks," *IEEE Communications Letters*, vol. 18, no. 4, pp. 592–595, 2014.

[34] J. Du, J.-F. Diouris, and Y. Wang, "A RSSI-based parameter tracking strategy for constrained position localization," *EURASIP Journal on Advances in Signal Processing*, vol. 2017, no. 1, Article ID 77, 2017.

[35] G. Wang, Y. Li, and N. Ansari, "A semidefinite relaxation method for source localization using TDOA and FDOA measurements," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 2, pp. 853–862, 2013.

[36] V. R. Kulkarni, V. Desai, and R. V. Kulkarni, "A comparative investigation of deterministic and metaheuristic algorithms for node localization in wireless sensor networks," *Wireless Networks*, vol. 25, no. 5, pp. 2789–2803, 2019.

[37] O. Cakir, I. Kaya, A. Yazgan, O. Cakir, and E. Tugcu, "Emitter location finding using particle swarm optimization," *Radioengineering*, vol. 23, no. 1, pp. 252–258, 2014.

[38] A. A. Hadi, A. W. Mohamed, and K. M. Jambi, "LSHADE-SPA memetic framework for solving large-scale optimization problems," *Complex & Intelligent Systems*, vol. 5, no. 1, pp. 25–40, 2019.

[39] S. Wang, Y. Li, H. Yang, and H. Liu, "Self-adaptive differential evolution algorithm with improved mutation strategy," *Soft Computing*, vol. 22, no. 10, pp. 3433–3447, 2018.

[40] G. Wu, R. Mallipeddi, P. N. Suganthan, R. Wang, and H. Chen, "Differential evolution with multi-population based ensemble of mutation strategies," *Information Sciences*, vol. 329, pp. 329–345, 2016.

[41] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, 2011.

[42] Y. Cai and J. Wang, "Differential evolution with neighborhood and direction information for numerical optimization," *IEEE Transactions on Cybernetics*, vol. 43, no. 6, pp. 2202–2215, 2013.

[43] H. Jia, C. Lang, D. Oliva, W. Song, and X. Peng, "Hybrid grasshopper optimization algorithm and differential evolution for multilevel satellite image segmentation," *Remote Sensing*, vol. 11, no. 9, article 1134, 2019.

[44] Q. H. Zhao, D. Urosevic, N. Mladenovic, and P. Hansen, "A restarted and modified simplex search for unconstrained optimization," *Computers & Operations Research*, vol. 36, no. 12, pp. 3263–3271, 2009.