



Article

A Shrink-Branch-Bound Algorithm for eLoran Pseudorange Positioning Initialization

Kaiqi Liu ^{1,2,3,*}, Jiangbin Yuan ^{1,3}, Wenhe Yan ^{1,3} , Chaozhong Yang ^{1,3} , Wei Guo ^{1,3}, Shifeng Li ^{1,2,3} and Yu Hua ^{1,2,3}

¹ National Time Service Center, Chinese Academy of Sciences, Xi'an 710600, China; yuanjiangbin@ntsc.ac.cn (J.Y.); ywh@ntsc.ac.cn (W.Y.); ycz@ntsc.ac.cn (C.Y.); guowei@ntsc.ac.cn (W.G.); lishifeng@ntsc.ac.cn (S.L.); hy@ntsc.ac.cn (Y.H.)

² University of Chinese Academy of Sciences, Beijing 100049, China

³ Key Laboratory of Precise Positioning and Timing Technology, Chinese Academy of Sciences, Xi'an 710600, China

* Correspondence: liukaiqi@ntsc.ac.cn

Abstract: Currently, eLoran is the ideal backup and supplement for global navigation satellite systems. The time synchronization accuracy between stations in the eLoran system has improved, providing conditions for eLoran pseudorange positioning. The pseudorange positioning of eLoran is a nonlinear least-squares problem and the location of the eLoran transmitting stations may cause the above problem to be non-convex. This makes the conventional pseudorange positioning algorithm strongly depend on the initial value when solving the eLoran pseudorange positioning. We propose a shrink-branch-bound (SBB) algorithm to solve the eLoran pseudorange positioning initialization problem. The algorithm first uses a shrink method to reduce the search space of the position estimator. Then, optimization is performed using a branch and bound algorithm within the shrunk region, where a trust region reflective algorithm is used for the lower bound process. The algorithm can help the receiver to complete the initial positioning without any initial value information. Simulation experiments verify that the algorithm has a success rate of more than 99.5% in solving the initialization problem of eLoran pseudorange positioning, and can be used as an initialization algorithm for pseudorange positioning problems for eLoran or other long-range terrestrial-based radio navigation system.

Keywords: pseudorange positioning; branch and bound; nonlinear least squares; eLoran; trust region reflective algorithm; initialization



Citation: Liu, K.; Yuan, J.; Yan, W.; Yang, C.; Guo, W.; Li, S.; Hua, Y. A Shrink-Branch-Bound Algorithm for eLoran Pseudorange Positioning Initialization. *Remote Sens.* **2022**, *14*, 1781. <https://doi.org/10.3390/rs14081781>

Academic Editors: Yuwei Chen, Changhui Jiang, Qian Meng, Bing Xu, Wang Gao, Panlong Wu, Lianwu Guan and Zeyu Li

Received: 4 March 2022

Accepted: 6 April 2022

Published: 7 April 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Global navigation satellite system (GNSS) provides all-weather, all-day positioning, navigation, and timing (PNT) services in most outdoor environments. However, in cities or canyons, GNSS performance can degrade due to multipath or poor visibility [1–3]. In addition, the high vulnerability of GNSS to interference also seriously affects the security of PNT services [4–6]. Many algorithms have been developed to mitigate the performance degradation of GNSS receivers in dynamic multipath environments [7–10]. However, these algorithms can only improve receiver performance under certain conditions, and it is still difficult for GNSS receivers to work properly in scenarios with fewer visible satellites, such as cities or canyons. Geomagnetic, Wifi, Doppler, and pseudolite-based positioning technologies have been developed for GNSS denial scenarios [11–14], but these technologies can only provide positioning services in small areas, which cannot meet the positioning requirements of large cities or canyon scenes. In recent years, the eLoran system has regained attention due to its unique system performance, which is expected to solve the existing problems of GNSS [15,16]. The eLoran system is a terrestrial-based radio navigation system that transmits navigation information through a pulse signal with

a carrier frequency of 100 kHz. The signal frequency band transmitted by the eLoran system is low and the transmission power is high. Therefore, the eLoran system has the advantages of wide coverage and good anti-interference performance, making it a good backup for GNSS [17–19]. The traditional Loran navigation system uses a hyperbolic positioning method based on the time difference of arrival (TDOA) [20]. The receiver can only use the stations in a single chain for positioning. Therefore, it has the disadvantage of poor geometric dilution of precision (GDOP), limiting its positioning accuracy. In addition, the TDOA observations include delay errors along the two propagation paths, which makes it difficult to measure and remove abnormal propagation delays. This positioning method cannot directly solve the clock deviation between the receiver and the transmitting station. The eLoran positioning method is based on pseudorange measurement and uses a circular positioning method based on the time of arrival (TOA). This method has the following advantages. First, the receiver uses the signals of multiple chains and multiple stations for positioning, which significantly improves the GDOP factor. Second, the receiver can directly complete the clock error calculation. Third, it can be easily integrated with the wireless positioning system, which helps build an integrated world-ground PNT system [21,22]. Due to limited conditions, the eLoran positioning failed to attract attention in the past. With the transformation and upgrading of eLoran stations, the time between stations in different chains has been synchronized to Universal Time Coordinated (UTC) through technologies such as optical fiber, and the time synchronization accuracy reaches the nanosecond level, providing the basis for the use of eLoran positioning technology. In addition, the application of digital technology in eLoran receivers has improved their sensitivity, which allows them to receive signals from multiple chains and stations simultaneously. Owing to this technical background, the Loran positioning method has regained attention in recent years.

Groves briefly introduced the Loran pseudorange positioning method and pointed out that it was processed by analogy with GNSS-related methods [23]. Yan analyzed the feasibility of Loran pseudorange positioning and the influence of additional secondary factor (*ASF*) errors on various errors in pseudorange positioning [24]. Kim used the eLoran pseudorange measurements from multiple chains for positioning and performed real-world testing [25]. Peterson and Fang studied the integrated positioning of eLoran and GNSS and pointed out that eLoran pseudorange positioning is a necessary condition for integrated positioning [22,26]. In the above-mentioned literature, eLoran pseudorange positioning is regarded as a nonlinear least squares problem, and local optimization algorithms such as Newton-Raphson algorithm (NR) are used to solve it. However, the eLoran system is not specifically designed for pseudorange positioning, and the location of the eLoran transmitting station may make the problem non-convex. In addition, the nonlinear term in the eLoran pseudorange function is a complex nonlinear function with trigonometric functions, which may cause an ill-condition problem when using the first-order or second-order derivation information to optimize the objective function. Therefore, for many existing nonlinear least squares algorithms, when the selected initial values are inaccurate, convergence problems to local solutions or erroneous convergence results occur. This initial value dependence affects the ability of the receiver to locate autonomously and causes the receiver to experience localization errors under cold start. At present, there is no literature on the problem of eLoran pseudorange positioning under insufficient initial value information.

This study proposes a shrink-brand-bound (SBB) algorithm to solve the eLoran pseudorange positioning problem. The algorithm first obtains the shrunk region of the estimator through the shrink algorithm. The positioning problem is then solved within this compressed feasible region using a branch-and-bound algorithm, where a trust region reflective algorithm is used for each bound process [27,28]. The SBB algorithm has a global optimization capability and can achieve accurate positioning solutions without initial value information. The algorithm avoids the problem faced by the traditional nonlinear least-squares method by relying on the initial value when solving the Loran pseudor-

ange positioning, which further improves the Loran positioning technology based on pseudorange measurement.

The rest of the paper is organized as follows. In Section 2, first, we describe the eLoran pseudorange measurement method and the error in the pseudorange. Then, we build a mathematical model of the eLoran pseudorange positioning and analyze the shortcomings of the NR algorithm in solving it. The principle of the SBB algorithm and the details of each part of the algorithm are introduced. In Section 3, we evaluate the performance of the SBB algorithm and other nonlinear least squares algorithms in solving the eLoran pseudorange positioning problem without initial value information through simulation experiments. Finally, we present the main conclusions of this paper.

2. Materials and Methods

In this section, we first introduce the pseudorange measurement technology and the error in the pseudorange. Secondly, we construct the mathematical model of eLoran pseudorange positioning and analyze the advantages and disadvantages of the traditional NR algorithm. Finally, we give the principle of the SBB algorithm and the details of each part of the algorithm.

2.1. Principle of eLoran's Pseudorange Measurement and Error Analysis

The eLoran positioning technology based on pseudo-range measurement includes two parts: pseudorange measurement technology and positioning algorithm. This section briefly describes the basic principles of pseudorange measurement technology and the error analysis in pseudorange measurement.

The basic principle of eLoran pseudorange measurement is shown in Figure 1. The receiver obtains the signal propagation delay or time of flight (TOF) by measuring the difference between the signal time of arrival (TOA) and the signal time of transmission (TOT). Usually, a certain characteristic point on the eLoran signal is selected as the TOT, such as the initial point or the zero-crossing point in the third circle. The TOA is obtained through the process of a group repetition period, carrier synchronization, and cycle identification. More details can be found in the references [24,29,30].

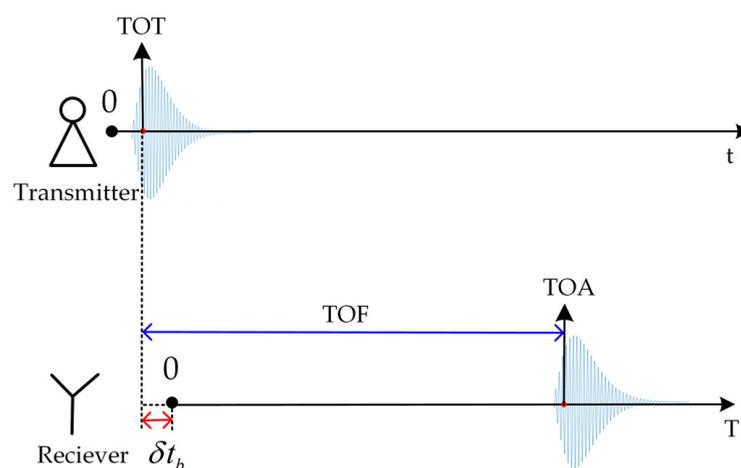


Figure 1. Schematic of eLoran pseudorange measurement principle.

The eLoran signal is mainly propagated by ground waves and its propagation process is affected by terrain, weather, and other conditions. Interference and noise also affect the TOA measurement during the receiver measurement process, so the TOF, which contains various additional time delay items, is not the true distance [31,32], as shown in Equation (1):

$$TOF = TOA - TOT = T_p + \Delta ASF(t) + \delta t_b + t_a + \eta(t), \quad (1)$$

where δt_b is the clock deviation between the receiver and the transmitting station, t_a is the receiver delay, $\eta(t)$ is the delay deviation caused by interference and noise in the *TOF* measurement process, and $\Delta ASF(t)$ is the time-related delay due to the ground wave propagation process time-varying factors such as weather. T_p is the delay term related to the propagation path as in Equation (2)

$$T_p = PF + SF + ASF, \quad (2)$$

where *PF* is the propagation delay of the signal through the atmosphere and is represented by Equation (3)

$$PF = \frac{s \cdot n_s}{c}, \quad (3)$$

where c is the speed of light in vacuum, s is the distance between the signal from the transmitter to the receiver; n_s is the refractive index of the atmosphere, which represents the ratio of the signal propagation speed in the atmosphere lower than the propagation speed in a vacuum. *SF* is the propagation delay of the signal through the entire seawater path, which is mainly related to the conductivity of the propagation path. *ASF* represents the propagation delay of eLoran signal caused by passing through a heterogeneous path of non-full seawater, which is mainly affected by parameters such as distance, surface impedance of the propagation path, and topography. *ASF* is an important factor affecting the positioning accuracy of eLoran, and it is often calibrated by eLoran differential station or *ASF* map [33–36].

In Equation (1), $\eta(t)$ and $\Delta ASF(t)$ are time-related delay items, which are difficult to calibrate. Figure 2 shows the statistical graph of the raw *TOF* value obtained by the receiver over time. The signal in the picture was transmitted from the Pucheng transmitting station (109.5438°E, 34.95043°N) and received in Lintong (109.2221°E, 34.3686°N). The fluctuation of the blue line in Figure 1 represents the *TOF*, which is affected by noise interference and its standard deviation is approximately 9 ns. The red line is the fitted curve of the data shown in blue, representing the fluctuation value with a standard deviation of approximately 10 ns. In order to present these time delays more clearly, we use the Fourier transform to analyze the spectrum of Figure 2a [37], and the obtained spectrum amplitude is shown in Figure 2b. In Figure 2b, we omit the spectrum after 0.001 Hz because its amplitude is too small. Among them, the amplitude at the lowest frequency is about 12 ns, which represents the deviation of the fitted curve in Figure 2a, that is, the delay introduced by $\Delta ASF(t)$. Other amplitudes due to measurement noise or interference are around 6 ns. As regards the delay error caused by measurement noise and interference $\eta(t)$, it is difficult to correct, so we uniformly regard it as noise. The error caused by $\Delta ASF(t)$ is often as high as more than 10 ns, so in high-precision eLoran positioning applications, the *ASF* prediction model is often used for calibration.

The propagation delay error calibration technology is essential for achieving high-precision positioning. There has been considerable research on this aspect [38–40]. Now consider the situation after the delay value is calibrated:

$$TOF_c = \tau + \delta t_b + \eta(t), \quad (4)$$

where TOF_c is the calibrated *TOF*, τ is the time delay value of the signal from the transmitting station to the receiver, δt_b is the clock deviation between the receiver and the transmitting station, $\eta(t)$ is the observation error introduced by the receiver due to time-varying factors such as interference, noise and $\Delta ASF(t)$. The t_a , *SF* and *ASF* in Equation (1) were calibrated. Multiplying both sides by the speed of light is the following pseudorange observation equation:

$$\rho = R_d + \rho_b + \eta, \quad (5)$$

where ρ is the pseudorange observation value of the station received by the receiver, R_d is the distance between the transmitting station and the receiver, ρ_b is the distance error

caused by the clock deviation between the receiver and the transmitting station, and η is the distance error representing all other errors that are difficult to calibrate.

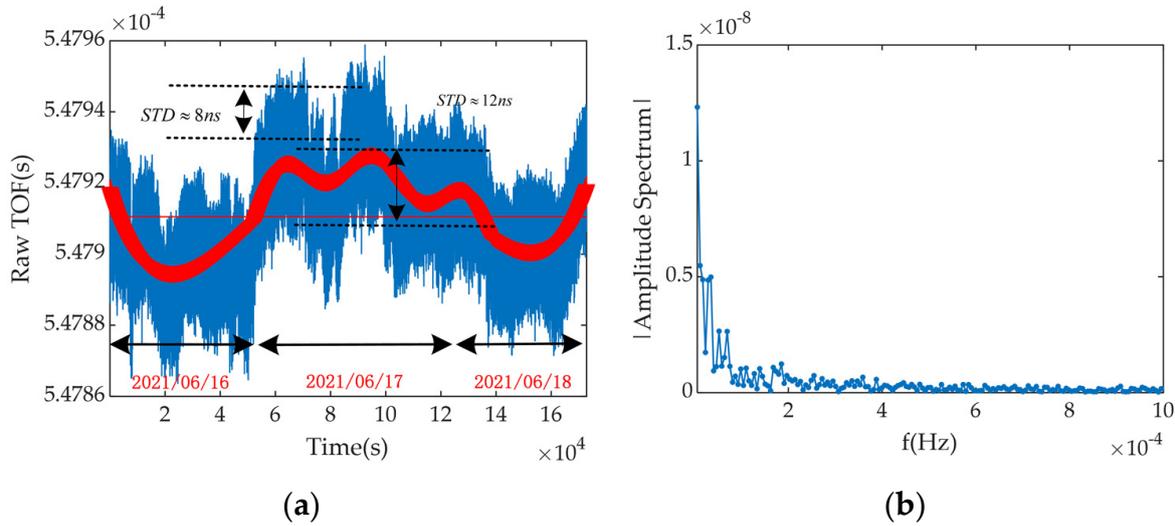


Figure 2. The schematic diagram of measured propagation delay. (a) time delay in time domain (b) the amplitude spectrum of delays in the frequency domain.

It is worth noting that the eLoran signals mainly propagate through ground waves, and the transmitter and receiver are usually not within the line-of-sight range, so R_d cannot be calculated directly using the Euclidean distance formula but needs to be calculated using the great circle distance. The great circle refers to the shortest distance between two points on the surface of a sphere or ellipsoid. The Andoyer–Lambert formula is commonly used in the navigation field to calculate the distance between two points on the earth [41,42]. Suppose the position of the i -th station of eLoran is (λ^i, φ^i) , and the position of the receiver is (λ, φ) . Andoyer–Lambert’s great circle distance formula is:

$$R_d^{(i)} = a\psi^{(i)} + \Delta S^{(i)}, \tag{6}$$

$$\begin{cases} \cos \psi^{(i)} = \sin \varphi^{(i)} \sin \varphi + \cos \varphi^{(i)} \cos \varphi \cos(\lambda - \lambda^{(i)}) \\ \Delta S = a \frac{f}{4} \left[\frac{\sin \psi^{(i)} - \psi^{(i)}}{1 + \cos \psi^{(i)}} (\sin \varphi + \sin \varphi^{(i)})^2 - \frac{\sin \psi^{(i)} + \psi^{(i)}}{1 - \cos \psi^{(i)}} (\sin \varphi - \sin \varphi^{(i)})^2 \right] \end{cases}, \tag{7}$$

Among them, λ^i, φ^i and λ, φ are the longitude and latitude of the transmitting station and the receiver, respectively, and ψ^i is the geocentric angle between the i -th eLoran station and the receiver. f and a are the basic geodetic parameters based on WGS-84; the former is the flattening of the ellipsoid, and the latter is the major axis radius of the reference ellipsoid.

2.2. eLoran Pseudorange Positioning Model and Conventional Positioning Algorithm

The eLoran pseudorange positioning is solving the estimator $\mathbf{x} = [\varphi \ \lambda \ \delta t]^T$. Since the eLoran positioning is a plane positioning system, we only estimate the longitude λ and latitude φ . The principle of eLoran pseudorange positioning is shown in Figure 3. Each circle takes the transmitting station as the center and the calibrated pseudorange observation between point A and each transmitting station as the radius. The circles represent all possible solutions to the pseudorange observation of Equation (5). Since \mathbf{x} contains three unknowns, the pseudorange observation equations of at least three stations are required to determine \mathbf{x} .

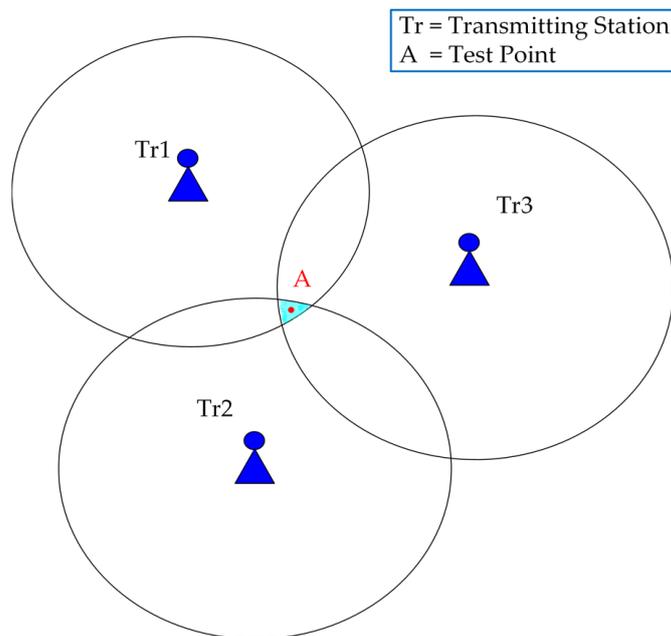


Figure 3. eLoran pseudorange positioning principle.

When we have no less than three pseudorange observation equations, we obtain \mathbf{x} by solving the following equation set:

$$\begin{cases} \rho^{(1)} - R_d^{(1)}(\varphi, \lambda) - \rho_b(\delta t) = 0 \\ \rho^{(2)} - R_d^{(2)}(\varphi, \lambda) - \rho_b(\delta t) = 0 \\ \dots \\ \rho^{(n)} - R_d^{(n)}(\varphi, \lambda) - \rho_b(\delta t) = 0 \end{cases} \quad (8)$$

The superscript of Equation (7) represents the eLoran station number. Owing to the existence of noise in the pseudorange observations, Equation (7) is often transformed into the following least-squares problem:

$$\min\{F(\mathbf{x})\} = \min\left\{ \sum_{i=1}^N \left[\rho^{(i)} - R_d^{(i)}(\varphi, \lambda) - \rho_b(\delta t) \right]^2 \right\}. \quad (9)$$

Equation (9) is the basic mathematical model of eLoran pseudorange positioning. The NR algorithm is widely used to solve the above problems. The algorithm linearizes Equation (9) through Taylor’s formula and transforms it into a linear least-squares problem. The basic process is as follows:

First, we perform Taylor’s first-order expansion of Equation (9) at \mathbf{x}_{k-1} , and obtain:

$$\mathbf{A} \cdot \Delta \mathbf{x} = \mathbf{B}, \quad (10)$$

where

$$\mathbf{H} = \begin{bmatrix} \frac{\partial R_{d,k-1}^1}{\partial \varphi} & \frac{\partial R_{d,k-1}^1}{\partial \lambda} & 1 \\ \frac{\partial R_{d,k-1}^2}{\partial \varphi} & \frac{\partial R_{d,k-1}^2}{\partial \lambda} & 1 \\ \vdots & \vdots & \vdots \\ \frac{\partial R_{d,k-1}^n}{\partial \varphi} & \frac{\partial R_{d,k-1}^n}{\partial \lambda} & 1 \end{bmatrix} \quad (11)$$

$$\Delta \mathbf{x} = \begin{bmatrix} \varphi_k - \varphi_{k-1} \\ \lambda_k - \lambda_{k-1} \\ \delta t_k - \delta t_{k-1} \end{bmatrix}, \tag{12}$$

$$\mathbf{B} = \begin{bmatrix} \rho^1 - \left(R_{d,k-1}^1 + \rho_{b,k-1} \right) \\ \rho^2 - \left(R_{d,k-1}^2 + \rho_{b,k-1} \right) \\ \vdots \\ \rho^n - \left(R_{d,k-1}^n + \rho_{b,k-1} \right) \end{bmatrix}. \tag{13}$$

Then, using the linear least-squares algorithm, the result is:

$$\Delta \mathbf{x} = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{B}. \tag{14}$$

Finally, the state estimator is:

$$\widehat{\mathbf{x}}_k = \mathbf{x}_{k-1} + \Delta \mathbf{x}. \tag{15}$$

The advantage of this method is that it is simple, and if a suitable initial value \mathbf{x}_0 is selected, the convergence speed is fast and the solution is accurate. However, $F(\mathbf{x})$ is affected by the geometry of eLoran stations and may have local minima. Consider a special case, as shown in Figure 4, in which Tr represents the transmitting station, A is the test point, and the four stations are in linear distribution; a common feature as stations are often built along the coastline. It can be seen from the contour line of the function $F(\lambda, \varphi)$ that there is a local minimum value W in $F(\mathbf{x})$. This means that when using local optimization algorithms such as the NR algorithm [43] or the Levenberg–Marquardt (LM) algorithm [44] to solve the above problem, an inappropriate initial point will cause the algorithm to converge to a local minimum. We will confirm this with a simulation in Section 3. In addition, since the great-circle distance function contained in the eLoran pseudorange equation is a nonlinear term with trigonometric functions, which means that the optimization using the first-order and second-order derivation information of the objective function may face the problem of ill-condition, thereby converging to an errorneous result. In view of this, it is necessary to design a global optimization algorithm to satisfy the positioning solution in the case of eLoran receiver cold-start.

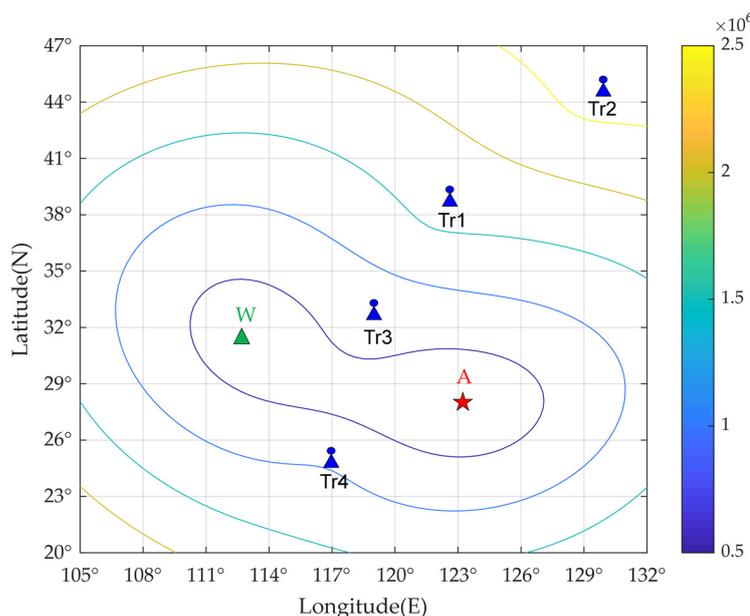


Figure 4. Contour map of $F(\varphi, \lambda)$ when the transmitting stations are linearly distributed.

2.3. The Shrink-Branch-Bound Algorithm

We define the eLoran positioning solution as the following optimization problem:

$$\{\mathbf{x}^* | F(\mathbf{x}^*) = F_{\min}(\mathbf{x}), \mathbf{x} \in D\}, \quad (16)$$

where $F : D \rightarrow R$ is the objective function, and F is defined in Equation (9). D is the feasible region of \mathbf{x} , or search space. λ and φ in \mathbf{x} have the following constraints

$$\mathbf{x} \in D = \begin{cases} -\pi \leq \lambda \leq \pi, \\ -\pi/2 \leq \varphi \leq \pi/2 \end{cases}, \quad (17)$$

The above boundary constraints represent the range of latitude and longitude coordinates of the earth. Since $\rho_b(\delta t)$ and δt have a linear relationship, the selection of the initial value of δt has no effect on the optimization process, so there is no need to consider the range of δt . From now on, we will refer D only to the feasible regions of λ and φ .

The SBB algorithm is a modification of the BB algorithm for the eLoran positioning problem. Before introducing the SBB algorithm, the BB algorithm needs to be described first. To solve the problem P , the BB algorithm first obtains a feasible solution as the optimal solution $\widehat{\mathbf{x}} \in D$ through a certain algorithm, and then iteratively divides the search space D into smaller subsets $D_{s1}, D_{s2}, \dots, D_{sn}$. In each iteration process, when a solution \mathbf{x}_1 with a better objective function value can be found in a subset D_{si} , the current solution is updated to $\widehat{\mathbf{x}} = \mathbf{x}_1$, and the subset is divided into smaller subsets; the above process is repeated. If no solution in the subset is better than $\widehat{\mathbf{x}}$, the subset is pruned. When no subset can be pruned, $\widehat{\mathbf{x}}$ is the optimal value of P , and the iteration stops. The pseudocode for the generic BB algorithm is given in Algorithm 1 [28,45].

Algorithm 1 Generic Branch-and-Bound

1. Set $L = \{D\}$, initial $\mathbf{x}^* = \widehat{\mathbf{x}}$
 2. While $L \neq \emptyset$
 3. Select a subproblem D_s from L to explore
 4. if a solution $\mathbf{x}_1 \in \{\mathbf{x} \in D_s | F(\mathbf{x}) < F(\widehat{\mathbf{x}})\}$ can be found, then $\widehat{\mathbf{x}} = \mathbf{x}_1$
 5. if D_s cannot be pruned:
 6. Partition D_s into $D_{s1}, D_{s2}, \dots, D_{sn}$
 7. Insert $D_{s1}, D_{s2}, \dots, D_{sn}$ into L
 8. Remove D_s from L
 9. Return $\widehat{\mathbf{x}}$
-

The proposed SBB algorithm adds the process of shrinking the feasible region based on the BB algorithm and designs the corresponding branching strategy, bounding method, and pruned strategy according to the eLoran positioning problem. The basic flow chart of the SBB algorithm is shown in Figure 5. We introduce the SBB algorithm from the shrink method and the BB algorithm.

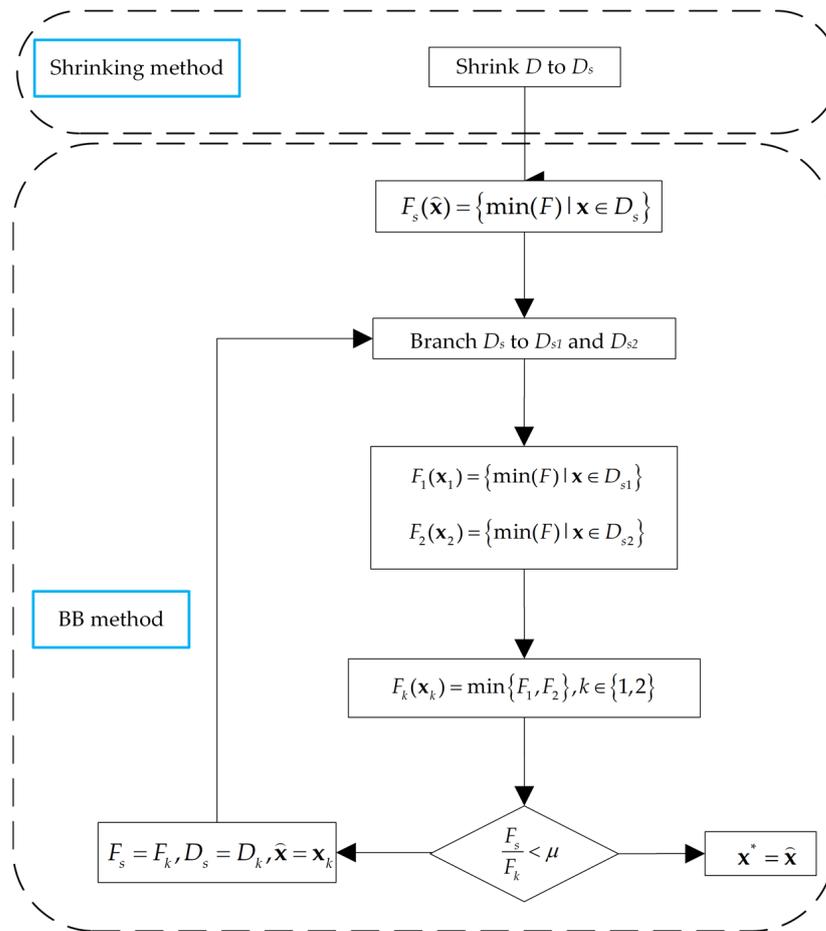


Figure 5. A diagram of the shrink-branch-bound algorithm.

2.3.1. The Shrink Method

From the basic principle of the BB algorithm, the search space D affects the amount of computation of the algorithm. If D can be shrunk, the subsequent BB algorithm can be significantly simplified. The range of D given by Equation (9) is derived from the range of latitude and longitude of the earth. Due to the limited coverage of the eLoran station, we can reduce D according to this feature.

The transmitting power of the eLoran transmitting station is usually fixed, and the eLoran receiver can receive signals from 800 km to 2500 km away from the transmitting station owing to the difference in the propagation path. When the receiver receives signals from multiple stations, it must be within the intersection of the coverage areas of these transmitters. Setting the range of this intersection as D_s , Figure 6 shows the basic schematic for determining D_s . The observable stations are TR1, TR2, TR3, and TR4. The prime vertical arc length between TR2 and TR4 is W , which can be estimated by Equation (18); the meridian arc length between TR1 and TR3 is L . It is estimated by Equation (19).

$$L = R_e(\varphi_{max} - \varphi_{min}), \tag{18}$$

$$W = R_e \cos(\bar{\varphi})(\lambda_{max} - \lambda_{min}), \tag{19}$$

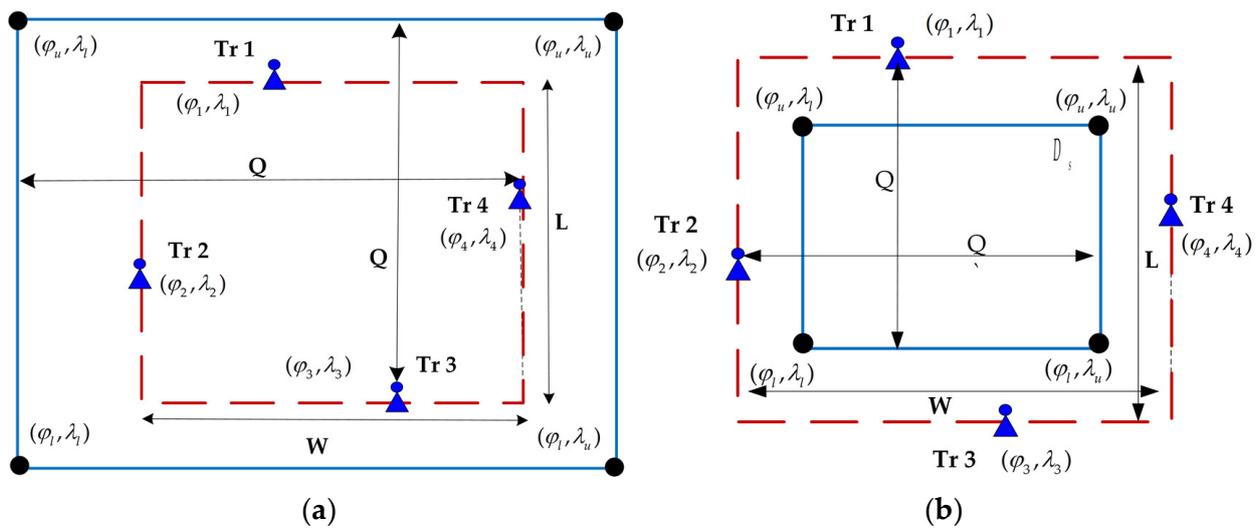


Figure 6. Shrinking diagram, (a) When $Q > W, Q > L$, the range of D_s . (b) When $Q < W, Q < L$, the range of D_s .

Assuming that the maximum working distance between the receiver and the transmitting station is Q , the range in the blue box of Figure 6 is D_s . Figure 6a,b show D_s under different conditions. The value of D_s can be calculated by Equation (20).

$$\begin{cases} \lambda \in [\lambda_l, \lambda_u] = \left[\min(\lambda_i) - \frac{(Q-W)}{R_e \cos(\bar{\varphi})}, \max(\lambda_i) - \frac{(W-Q)}{R_e \cos(\bar{\varphi})} \right] \\ \varphi \in [\varphi_l, \varphi_u] = \left[\min(\varphi_i) - \frac{(Q-L)}{R_e}, \max(\varphi_i) - \frac{(L-Q)}{R_e} \right] \end{cases}, \quad (20)$$

where R_e is the equivalent radius of the earth under the WGS-84 model. $\bar{\varphi}$ is the average latitude of the four stations. In practical applications, the setting of Q does not need to be precise but can be set as the maximum propagation distance according to the receiver performance and actual environment. In addition, Equation (20) is a general equation not limited to the two cases shown in Figure 6a,b. Therefore, once the receiver has identified the station information, Equation (20) can be used to calculate D_s .

2.3.2. The Branch and Bound Method in SBB Algorithm

The proposed branch-and-bound algorithm is as follows: First, a feasible solution \hat{x} of F on D_s is obtained through a shrink algorithm, and \hat{x} is assumed to be the global optimal solution. Then, we divide D_s into D_{s1} and D_{s2} and calculate the lower bounds $F_1(x_1)$ and $F_2(x_2)$ of function F on feasible domains D_{s1} and D_{s2} . We compare F_1 and F_2 and retain the subset D_{si} that has a lower bound F_i , where $i = 1, 2$. Thereafter, we compare the order of F_s and F_i . If the order of F_i is smaller than F_s , we update the solution $\hat{x} = x_1$, and divide D_{si} again and repeat the above steps. If F_s and F_i are of the same order, or the order of F_s is less than F_i , then \hat{x} is the global optimal solution. The pseudocode of the SBB algorithm is shown in Algorithm 2.

Line 3 of the pseudocode is the branch strategy and we adopt the binary branch scheme as shown in Figure 7. The basic division principle is to make a vertical line at the midpoint of the broadest side of D_s to bisect D_s . Because the number of local minima on the F function is small, there is no need to divide D_s too much, and this binary branch strategy can effectively reduce the amount of calculation without losing the accuracy of the algorithm.

Algorithm 2 SBB Algorithm

1. Shrinking D to D_s , using Equation (20)
2. Take the initial value $\mathbf{x}_0 \in D$, use TRR algorithm to calculate $F_s(\hat{\mathbf{x}}) = \{\min(F)|\mathbf{x} \in D_s, \mathbf{x}_0\}$
3. Branch D_s into D_{s1} and D_{s2} .
4. Calculate $F_1 = \{\min(F)|\mathbf{x} \in D_{s1}, \mathbf{x}_0\}$ and $F_2 = \{\min(F)|\mathbf{x} \in D_{s2}, \mathbf{x}_0\}$ and their corresponding solutions \mathbf{x}_1 and \mathbf{x}_2 .
5. $F_k = \min\{F_1, F_2\}$ and F_s , where $k \in \{1, 2\}$,
6. If $\frac{F_s}{F_k} < \mu$,
Then $\mathbf{x}^* = \hat{\mathbf{x}}$, the iteration ends;
7. If $\frac{F_s}{F_k} > \mu$,
Then $F_s = F_k, D_s = D_{sk}, \hat{\mathbf{x}} = \mathbf{x}_k$, and repeat steps 3–5.

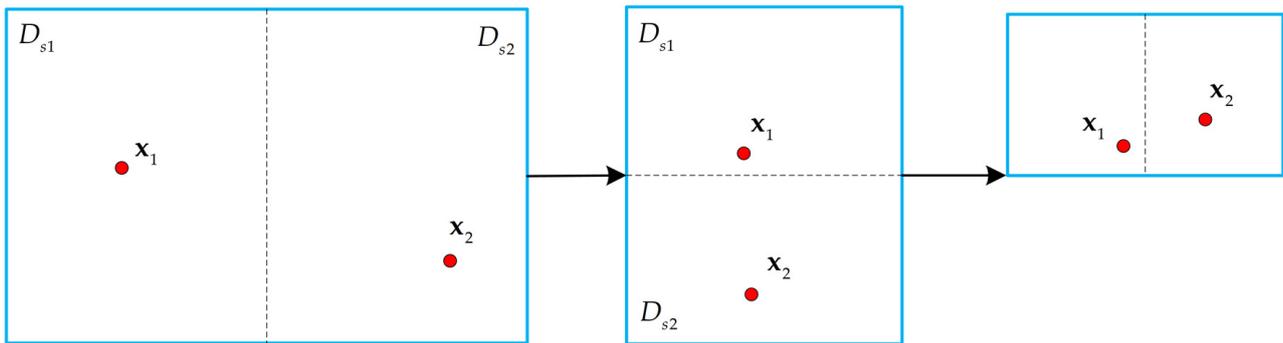


Figure 7. Diagram of the binary branching strategy.

In lines 2 and 4 of the pseudocode, it is necessary to calculate the lower bound of the objective function F in the specified feasible region, that is, to solve the following mathematical equation:

$$\{\mathbf{x}_s | F(\mathbf{x}_s) = F_{min}, \mathbf{x} \in D_s\} \tag{21}$$

D_s is determined by Equation (20). Equation (21) is a nonlinear least-squares problem with box constraints, which can be solved by the trust region reflective (TRR) algorithm. Based on the trust region algorithm, the trust region reflective method transforms the boundary-constrained optimization problem into an unconstrained optimization problem through reflection transformation so that each iteration result satisfies the boundary constraints [27]. The TRR algorithm uses the function $q(s)$ to fully approximate the behavior of the function $F(\mathbf{x})$ in the neighborhood N of \mathbf{x}_k , and find the tentative step s in this neighborhood. The pseudocode of the TRR algorithm is shown in Algorithm 3. In lines 3 and 4 of the pseudocode of Algorithm 3, the trust region model to be solved is as follows:

$$\min \{ q(s) = \frac{1}{2}s^T Hs + s^T g, \quad \|s\| \leq N \}, \tag{22}$$

where g is the gradient of the current $F(x_k)$, H is the Hessian matrix or the approximation of the Hessian matrix of $F(x_k)$, N is the trust region, and $\| \cdot \|$ is the 2-norm. For the solution of Equation (22), please refer to the literature [46,47]. Details of the reflection transformation method in line 5 can be found in the literature [27]. The approximation factor ρ_k of $q(s_k)$ to $F(s_k)$ in line 6 can be given by Equation (23):

$$\rho_k = \frac{F(\mathbf{x}_k) - F(\mathbf{x}_k + s_k)}{F(\mathbf{x}_k) - q(s_k)}, \tag{23}$$

When ρ_k is greater than the set value μ , it means that the current approximation effect of $q(s_k)$ to $F(x_k)$ is good and the update step is $\mathbf{x}_{k+1} = \mathbf{x}_k + N_k$. Otherwise, the trust region radius N_k needs to be adjusted, the trust region sub-problem solved again, and the above process repeated.

Algorithm 3 TRR Algorithm

1. Initial \mathbf{x}_0 , N_0 and μ
 2. While $g(x_k) > \mu$
 3. Build a trust region model $q(s)$
 4. Solve the trust region subproblem, and get s_k
 5. If $s_k \notin D_s$
 6. Perform a reflection transform on s_k
 7. Calculate the approximation ρ_k of $q(s_k)$ to $F(s_k)$ and update s_k or \mathbf{x}_k
 8. Return $\widehat{\mathbf{x}}$
-

The TRR algorithm can make full use of the feature that the BB algorithm divides the feasible region. When the feasible region is divided, the box constraints will continue to shrink, and the probability of the trust region algorithm converging to the global optimal value will continue to increase. Using the TRR algorithm to obtain the lower bound of F under different feasible regions, the following inequalities must be satisfied.

$$\begin{cases} F_1 \leq F_s \\ F_2 \leq F_s \end{cases} \quad (24)$$

where

$$F_s = \{F_{min} | \mathbf{x} \in D_s, \mathbf{x}_0\}, \quad (25)$$

$$F_1 = \{F_{min} | \mathbf{x} \in D_{s1}, \mathbf{x}_0\}, \quad (26)$$

$$F_2 = \{F_{min} | \mathbf{x} \in D_{s2}, \mathbf{x}_0\}, \quad (27)$$

Lines 5 and 6 of Algorithm 2 are the verification phase. We use μ to verify the convergence process, and μ can be a constant less than 5. When $F_s/F_k < \mu$, it means that F_s and F_k are of the same order, and the current iteration value is close to converging to the global optimal value, and the iteration ends. Otherwise, the above branch and bound process needs to be repeated.

2.3.3. Complexity Analysis

The main computational complexity of the proposed SBB algorithm is related to the number of branch iterations N and the convergence accuracy ϵ . In each iteration, the main computational complexity is related to the update of the bounding process of $F(\mathbf{x})$. More specifically, when we set the norm of the gradient of the solution to be $\|\nabla F\| \leq \epsilon$, the upper bounds of the complexity required to solve steps (2) and (4) are $O(\epsilon^{-2})$ and $O(2\epsilon^{-2})$, respectively [48]. Considering the number of branch iterations N , the upper bound of the complexity of the SBB algorithm is $O((2N + 1)\epsilon^{-2})$. The upper bounds of the complexity of the following algorithms are shown in the Table 1.

Table 1. Algorithms Computational Complexity Comparison.

Algorithms	Computational Complexity
NR [49]	$O(kn^2m)$
LM [50]	$O(\epsilon^{-2})$
Dogleg/TTR [48]	$O(\epsilon^{-2})$
SBB	$O((2N + 1)\epsilon^{-2})$

The above table shows the upper bound of the computational complexity of different algorithms. Among them, the LM algorithm, the Dogleg algorithm, and the TTR algorithm are all Cauchy-related algorithms or Newton-like algorithms, and the upper bound of their complexity is $O(\epsilon^{-2})$. The complexity of the NR algorithm is related to the number of iterations and the matrix calculation, where k is the number of iterations required, and m and n represent the dimensions of the estimator and the number of equations, respectively. It can be found that the complexity of the SBB algorithm compared with other algorithms

mainly lies in N . Since we have shrunk D to D_s , this makes the number of branches N usually small, and we will confirm this in simulation experiments.

3. Results

The SBB algorithm is used to solve the initialization problem of eLoran pseudorange positioning. Therefore, the evaluation of the algorithm is mainly from two aspects. First, the algorithm should still be able to solve the position correctly when no initial value is available, which means that given a random initial value, the algorithm should be able to solve the position accurately. Secondly, the computational complexity of the algorithm should be at a reasonable level so that it can be implemented in the receiver. Based on the above evaluation criteria, this section is organized as follows: we first set the simulation parameters according to the actual station distribution. Then, the performance of various algorithms in solving the eLoran pseudorange positioning problem is compared. Finally, the reliability of the SBB algorithm was verified through simulation.

3.1. Simulation Parameter Settings

Assuming that the receiver at point A receives the signals from the four eLoran transmitting stations shown in Table 2, the calibrated pseudorange observations and geodesic distance values between point A and eLoran stations are shown in Table 3, and the atmospheric refractive index n_s is 1.000315. Where the calibrated pseudorange observations ρ are as described by Equation (5), they only include the clock deviation δt and the observation error η caused by time-varying delay factor. We set the clock error δt to be $5 \mu s$ and η follows a normal distribution, that is, $\eta \sim N(0, 50)$.

Table 2. Transmitting station location and coordinates.

Transmitting Station Mark	Position	Longitude (E)	Latitude (N)
M	Rongcheng	122.3228	37.0644
T	Helong	129.1075	42.7199
Y	Xuancheng	118.886	31.0689
Z	Raoping	116.8958	23.7239

Table 3. Distance information from point A to each station.

Test Point (φ^*, λ^*) (N,E)	Transmitting Station	Distance (R_d/m)	PF (μs)	Pseudorange Observations (ρ/m)
A (27, 124)	M	1,128,758	3766.316	1,130,278
	T	1,806,302	6027.074	1,807,799
	Y	672,027	2242.348	673,547
	Z	801,620	2674.758	803,112

3.2. Analysis and Comparison of Simulation Results

Figure 8 shows the location of the transmitter station and receiver on the map. To clearly show the influence of (φ, λ) on $F(\mathbf{x})$, the contour of $F(\varphi, \lambda)$ is shown in Figure 9, where δt is set to a known value. The four black contours in Figure 9 are, respectively, surrounded by the solution sets of the four observation equations. The contour shape shows the non-convexity of $F(\varphi, \lambda)$, which is mainly related to the topology of the transmitting station. Take A as the test point and select the four positions shown in Table 3 as the initial value points. Since δt_0 has no effect on the optimization process, it will always be set to 0 in subsequent simulations.

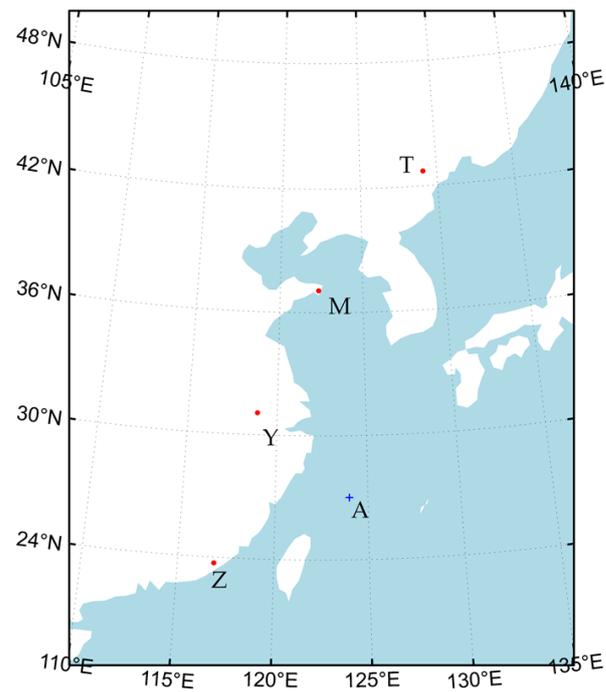


Figure 8. Stations location distribution.

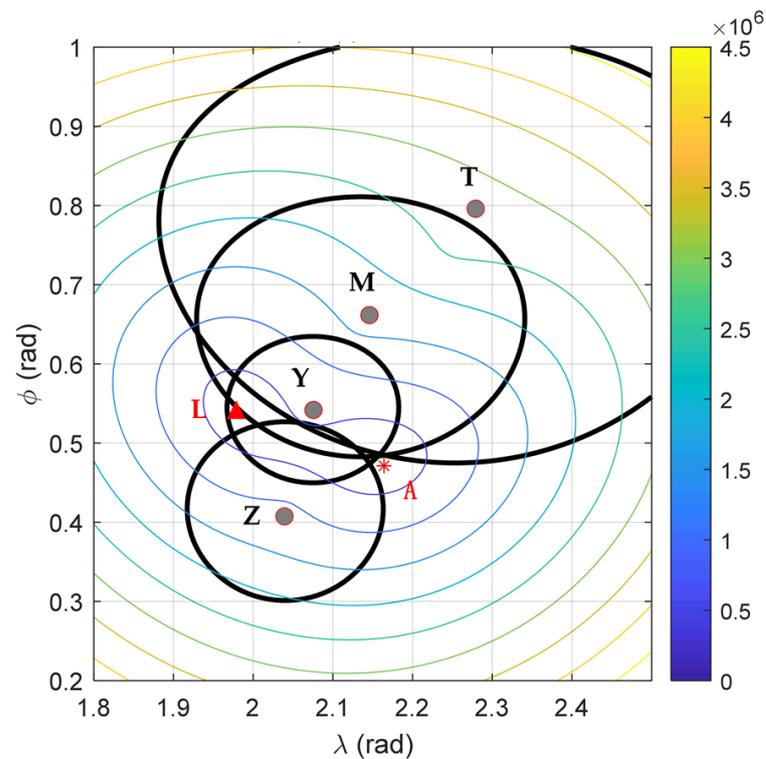


Figure 9. Contour plot of $F(\varphi, \lambda)$.

Since the initialization problem of eLoran has not been studied in the literature, there is a lack of competing algorithms for performance comparison. To this end, we select four commonly used nonlinear least squares methods, namely, the NR algorithm, the Levenberg–Marquardt (LM) algorithm, and the trust region Dogleg algorithm to compare with the SBB algorithm. The NR algorithm is a commonly used algorithm in positioning and is widely used in various pseudorange positioning scenarios. Its advantage is that the calculation is

simple, and if the initial is suitable, it will converge quickly. Currently, only this algorithm is mentioned in the existing papers to solve the eLoran localization problem. The LM algorithm is an algorithm that combines the steepest descent method and Newton’s method, and is currently widely used in nonlinear least squares. It is characterized by considering the stability of the steepest descent method and the fast convergence characteristics of Newton’s method. This algorithm is a benchmark algorithm for solving nonlinear least squares problems based on the derivation algorithm, and is widely used in various scenarios. The LM algorithm can represent a series of scenarios based on the derivation algorithm to demonstrate the problem of solving the eLoran localization problem based on the first-order derivation and the second-order derivation algorithm.

The trust region Dogleg algorithm is representative of another large class of algorithms for solving nonlinear least squares algorithms. It is different from the line search algorithm; the algorithm first sets the step size, and then determines the search direction. The advantage of this algorithm is that it does not require a line search process when solving complex nonlinear least squares problems. Furthermore, even if the condition number of the objective function is poor, it is easy to introduce second-order information of the function. The above three algorithms represent the three most commonly used ideas for solving nonlinear least squares problems. The results are shown in Table 4.

Table 4. Convergence results of conventional algorithms under different initial points.

Point	Initial Points	NR Results	LM Results	Dogleg Results	SBB Results
$(\varphi^*, \lambda^*) (N, E)$	$(\varphi_0, \lambda_0) (N, E)$	$(\varphi, \lambda) (N, E)$			
A (27, 124)	0, 0	23.7162, 148.7832	31.2167, 103.7164	31.2167, 103.7164	27.0001, 124.0001
	28, 125	27.0001, 124.0001	27.0001, 124.0001	27.0001, 124.0001	27.0001, 124.0001
	40.1, 97.4	31.7164, 103.2167	31.21671, 03.7164	31.2167, 103.7164	27.0001, 124.0001
	32, 148.8	30.2167, 10.7162	27.0001, 124.0001	27.0001, 124.0001	27.0001, 124.0001
	28, 100	31.2167, 103.7164	31.2167, 103.7164	31.2167, 103.7164	27.0000, 123.9998
	15, 128	31.2195, 16.7159	26.9991, 124.0011	26.9991, 124.0011	26.9991, 124.0011

The data in red are the incorrect results, and the data in black are the correct results.

In Table 4, the data in red are the incorrect results, and the data in black are the correct results. The results of all algorithms may be incorrect due to the selection of initial values, except for the SBB algorithm. Among them, both the LM and Dogleg algorithms converge to (31.2167, 103.7164), which is the local minimum L shown in Figure 9. In addition, when the initial value point is close to Point A, both the LM and Dogleg algorithms converge correctly; when the initial value point is close to the local minimum point L, all the results of the above two are incorrect. The erroneous results of the NR algorithm may go beyond the feasible region D, mainly because the convergence of the NR algorithm may be out of control due to the lack of line search. Results from the above table verify that we need a global optimization algorithm to solve the eLoran pseudorange positioning problem when the initial value is not available.

We analyze how the SBB algorithm can always converge to the correct result, regardless of the change in the initial value.

Consider the shrink method of the SBB algorithm. Without loss of generality, we set Q in Equation (20) to 3000, and the reduced feasible region D_s is shown as the red box in Figure 10. It can be seen that D_s has been significantly reduced compared to D, which reduces the subsequent computation of the SBB algorithm.

To observe the global optimization performance of the SBB algorithm more clearly, Tables 5 and 6 show the iterative process of branch and bound under some initial value points.

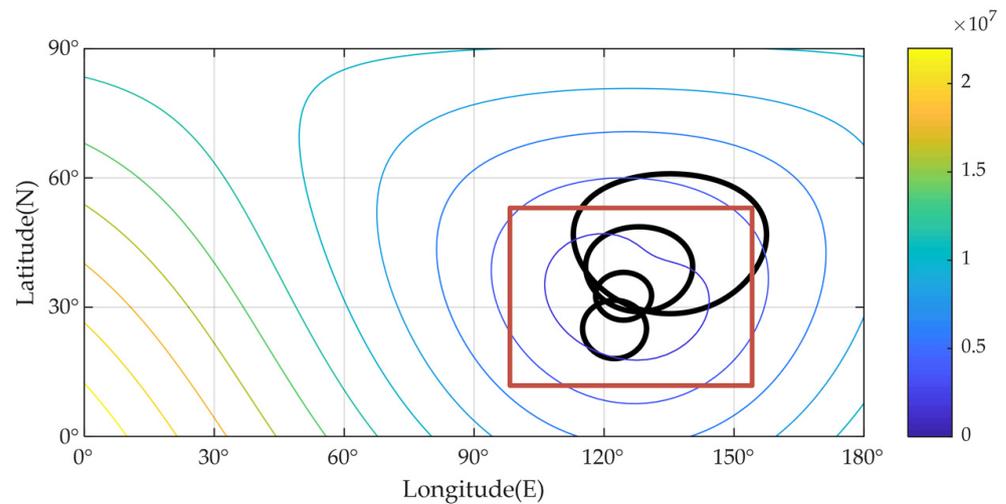


Figure 10. Contour line of F under feasible region D and feasible region D_s .

Table 5. Iterative process with initial value (0, 0, 0).

Feasible Region	F_{min}	$x^* (\varphi, \lambda, \delta t)$ (deg, deg, μs)
D	3.5×10^9	31.2174, 103.7183, -2480
D_s	63.7	27.0001, 124.0001, 5.091
D_{s1}	3.5×10^9	31.2174, 103.7183, -2480
D_{s2}	63.7	27.0001, 124.0001, 5.091

Table 6. Iterative process with initial value (97.4, 40.1, 0).

Feasible Region	F_{min}	$x^* (\varphi, \lambda, \delta t)$ (deg, deg, μs)
D	3.5×10^9	31.2147, 103.7225, -2480
D_s	3.5×10^9	31.2147, 103.7225, -2480
D_{s1}	3.5×10^9	31.2147, 103.7225, -2480
D_{s2}	99.8	26.9998, 124.0009, 4.933
D_{s1}	99.8	31.2147, 103.7225, -2480
D_{s2}	1.5×10^9	26.9998, 124.0009, 4.933

It can be seen from Tables 5 and 6 that, as the feasible region is continuously shrunk and divided, the SBB algorithm gradually converges to close to the global minimum.

To further verify the performance of the SBB algorithm, we designed the following simulation experiments: we randomly selected 1000 locations within D_s as test points and used the above mentioned algorithms to solve for these locations. Note that these locations were chosen to keep the GDOP as consistent as possible to avoid the impact of GDOP on location accuracy. For each algorithm, the initial value was randomly selected in D and D_s . When the positioning error was lower than the set threshold, the solution was successful. The statistical results of the success rate of these algorithms in solving these 1000 positions are shown in Figure 11.

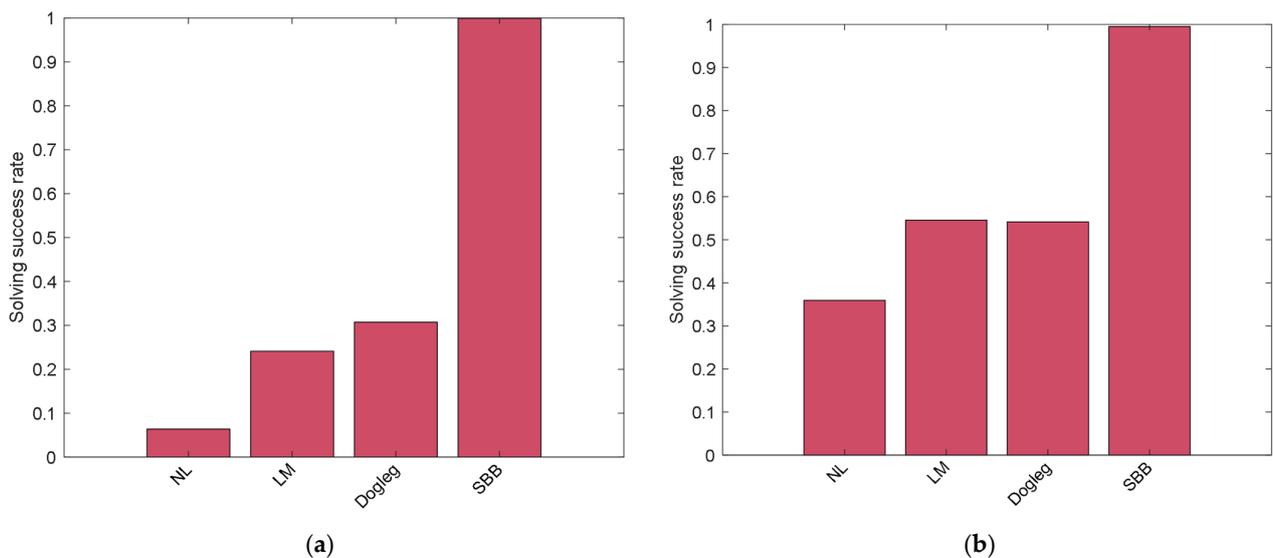


Figure 11. Statistical chart of success rate of different algorithms. (a) $x_0 \in D$; (b) $x_0 \in D_s$.

As shown in Figure 11, the LM and Dogleg algorithms have a success rate of 55% in Figure 11b, while in Figure 11a, the success rates of the two are only 25% and 30%, respectively. This shows that the two algorithms depend strongly on the selection of initial values. The NL algorithm has the lowest success rate, and its solution probabilities are 5% and 35%, respectively, under the two initial value selection schemes. The main reason for the poor performance of the NL algorithm is that it lacks a line search process compared to the LM and Dogleg. The solution success rates of the SBB algorithm under the two initial value selection schemes are 99.9% and 99.5%, respectively, showing good global optimization performance. The possible reason for the failure of the SBB algorithm is that the algorithm will converge to the local minimum value when x_0 is selected very close to the local minimum value. Thus, when x_0 is selected in D , there is a smaller probability of selecting points close to the local minimum. Therefore, the success rate of the algorithm will be improved under $x_0 \in D$ compared to under $x_0 \in D_s$. To avoid choosing a point near the local minimum as the initial value when using the SBB algorithm, we can choose a point far away from all possible solutions as the initial value point, such as $(0, 0)$.

Computational complexity affects the performance of an algorithm. The previous analysis of the complexity of the SBB algorithm showed that the number of branches, N , has an important impact on the complexity of the SBB algorithm. The figure shows the statistical graph of the number of branch iterations, N , required by the SBB algorithm to complete the positioning solution each time in 1000 positioning simulation experiments. Figure 12 shows that the SBB algorithm needs at most two branch iterations to complete the solution, and even only one branch is required in most cases. Comparing Figure 12a,b, it can be found that the probability that the latter requires two branches to solve is 46%, which is much higher than the 24% of the former. This is because when the initial value is randomly selected in D_s , there will be a higher probability of selecting the point close to the local minimum, which makes it converge to the global optimal value after two branches.

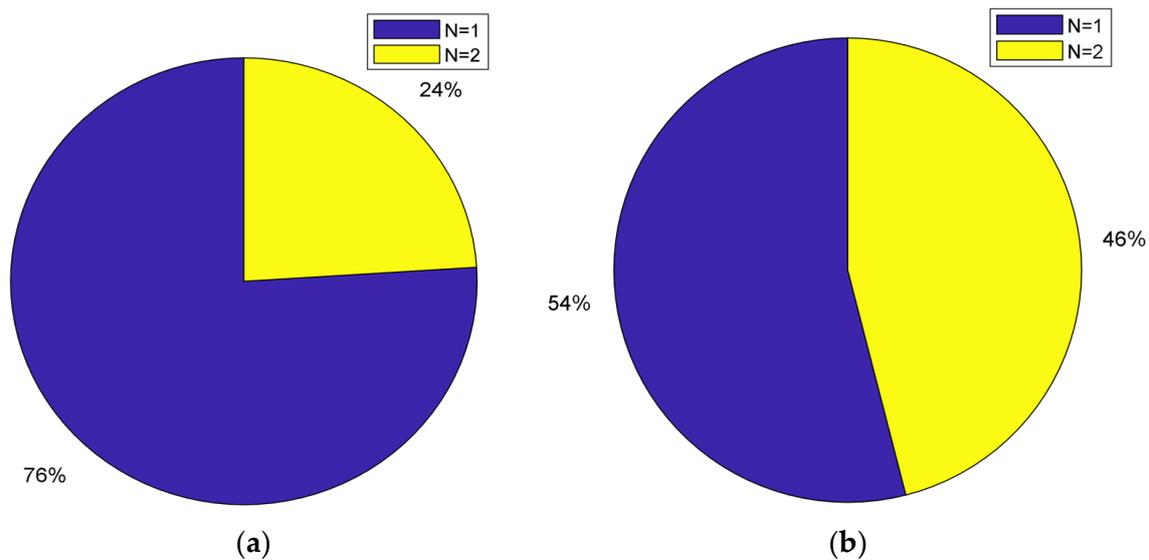


Figure 12. Pie chart of number of branch iterations N . (a) $x_0 \in D$; (b) $x_0 \in D_s$.

4. Discussion

Using pseudorange measurements for positioning in the eLoran system can make full use of the available eLoran stations, thereby expanding the coverage of the eLoran system and improving the positioning accuracy of the system. An important problem with eLoran pseudorange positioning, however, is that the geometric distribution of available eLoran transmitting stations may cause the positioning problem to be non-convex. This makes the existing pseudorange positioning algorithms such as NR algorithms extremely dependent on the selection of initial value. In practical positioning applications, it is difficult for the receiver to obtain reliable initial values in many cases. Therefore, conventional positioning algorithms may converge to wrong solutions due to lack of reliable initial values. At present, there is no literature to study the eLoran pseudorange localization initialization problem.

We transformed the eLoran pseudorange positioning into a nonlinear least squares problem with box constraints and proposed the shrink-branch-bound algorithm (SBB), a global optimization algorithm that can achieve accurate positioning without any initial value. The SBB algorithm first obtains the shrunk region of the estimator through the shrink method. The positioning problem is then solved within this shrunk feasible region using a branch-and-bound algorithm, where a trust region reflective algorithm is used for each bound process. We verified the performance of this method through simulation experiments. The results show that the success rate of the SBB algorithm to solve the position is more than 99.5%, when no initial value is available. However, the success rate of other conventional nonlinear least squares algorithms (such as LM algorithm, Dogleg algorithm) in this case is only around 50%. These results confirm that our proposed SBB algorithm can help the receiver to obtain correct positioning results when no initial value is available.

For the eLoran receiver, both the accuracy of the positioning algorithm and the computational complexity need to be considered. The computational complexity of the SBB algorithm is comparable to traditional Newton-based methods or Cauchy-related methods, which means that it can be implemented in the receiver.

5. Conclusions

eLoran is the ideal backup and supplement to GNSS systems. The improved accuracy of time synchronization between eLoran stations provides conditions for eLoran pseudorange positioning. We proposed a shrink-branch-bound (SBB) algorithm to solve the eLoran pseudorange positioning problem when the receiver has no initial value available. We verified the performance of the SBB algorithm through simulation experiments. The

results show that the success rate of SBB algorithm in converging to the correct result without initial value is over 99.5%, which is more than 40% higher than that of conventional nonlinear least squares algorithms such as LM algorithm and Dogleg algorithm.

The proposed SBB algorithm is expected to make up for the defect that the existing eLoran pseudorange localization algorithm may converge to wrong results when no initial value is available, so it can be used as a cold-start algorithm for eLoran receivers. Therefore, the focus of follow-up research is to combine the SBB algorithm with the existing high-precision positioning algorithms, which is expected to further improve the positioning accuracy and reliability of the eLoran system under high dynamic conditions

Author Contributions: Conceptualization, K.L. and J.Y.; methodology, K.L.; software, K.L. and W.Y.; validation, K.L., J.Y. and W.Y.; formal analysis, S.L. and C.Y.; investigation, K.L.; resources, Y.H.; data curation, W.G.; writing—original draft preparation, K.L.; writing—review and editing, K.L.; visualization, W.Y.; supervision, J.Y.; project administration, S.L.; funding acquisition, C.Y., J.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Chinese Academy of Sciences “Light of West China” Program (Grant No. E017YR1R10) and “Youth Innovation Promotion Association CAS” (GrantNo. 1188000YCZ).

Data Availability Statement: Restrictions apply to the availability of these data. The ownership of data belongs to the National Time Service Center (NTSC). These data can be available from the corresponding author with the permission of NTSC.

Acknowledgments: The authors would like to thank their colleagues for testing of the data provided in this manuscript. We are also very grateful to our reviewers who provided insight and expertise that greatly assisted the research.

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

References

- Hussain, A.; Akhtar, F.; Khand, Z.H.; Rajput, A.; Shaukat, Z. Complexity and Limitations of GNSS Signal Reception in Highly Obstructed Enviroments. *Eng. Technol. Appl. Sci. Res.* **2021**, *11*, 6864–6868. [[CrossRef](#)]
- Lu, D.; Jiang, S.; Cai, B.; Shangguan, W.; Liu, X.; Luan, J. Quantitative analysis of GNSS performance under railway obstruction environment. In Proceedings of the 2018 IEEE/ION Position, Location and Navigation Symposium (PLANS), Monterey, CA, USA, 23–26 April 2018; pp. 1074–1080. [[CrossRef](#)]
- Wang, L.; Groves, P.D.; Ziebart, M.K. Multi-Constellation GNSS Performance Evaluation for Urban Canyons Using Large Virtual Reality City Models. *J. Navig.* **2012**, *65*, 459–476. [[CrossRef](#)]
- Zhao, X.; Zhan, X.-Q.; Liu, X.; Li, S.-J. GNSS Vulnerability Analysis and Assessment. *J. Aeronaut. Astronaut. Aviat.* **2014**, *46*, 11–19. [[CrossRef](#)]
- Dinesh, S. Globle Navigation Satellite System (GNSS) Spoofing: A Review of Growing Risks and Mitigation Steps. *Def. ST Tech. Bull.* **2013**, *6*, 42–61.
- Panagiotis, P.; Jovanovic, A. Protection and Fundamental Vulnerability of GNSS. In Proceedings of the the 2008 IEEE International Workshop on Satellite and Space Communications 2008, Toulouse, France, 1–3 October 2008. [[CrossRef](#)]
- Fascista, A.; Coluccia, A.; Ricci, G. A Pseudo Maximum likelihood approach to position estimation in dynamic multipath environments. *Signal Process.* **2020**, *181*, 107907. [[CrossRef](#)]
- Lesouple, J.; Robert, T.; Sahmoudi, M.; Tourneret, J.-Y.; Vigneau, W. Multipath Mitigation for GNSS Positioning in an Urban Environment Using Sparse Estimation. *IEEE Trans. Intell. Transp. Syst.* **2018**, *20*, 1316–1328. [[CrossRef](#)]
- Cheng, L.; Wang, K.; Ren, M.; Yan, G. Adaptive Filter Approach for Gps Multipath Estimation under Correntropy Criterion in Dynamic Multipath Environment. *IEEE Trans. Signal Process.* **2019**, *67*, 5798–5810. [[CrossRef](#)]
- Cheng, C.; Tourneret, J.-Y. An EM-based multipath interference mitigation in GNSS receivers. *Signal Process.* **2019**, *162*, 141–152. [[CrossRef](#)]
- Ferrigno, L.; Laracca, M.; Milano, F.; Cerro, G.; Bellitti, P.; Serpelloni, M.; Piedrafita, O.C. Magnetic Localization System for Short-Range Positioning: A Ready-to-Use Design Tool. *IEEE Trans. Instrum. Meas.* **2020**, *70*, 1–9. [[CrossRef](#)]
- Wang, T.; Sui, T.; Liu, X.; Yuan, M.; Sun, G.; Gao, Z. WiFi positioning algorithm in tunnel based on Fuzzy C-means clustering and KNN algorithm. In Proceedings of the the 2019 Chinese Automation Congress (CAC), Hangzhou, China, 22–24 November 2019; pp. 567–571. [[CrossRef](#)]
- Halili, R.; BniLam, N.; Yusuf, M.; Tanghe, E.; Joseph, W.; Weyn, M.; Berkvens, R. Vehicle Localization Using Doppler Shift and Time of Arrival Measurements in a Tunnel Environment. *Sensors* **2022**, *22*, 847. [[CrossRef](#)]

14. Sheng, C.; Gan, X.; Yu, B.; Zhang, J. Precise Point Positioning Algorithm for Pseudolite Combined with GNSS in a Constrained Observation Environment. *Sensors* **2020**, *20*, 1120. [[CrossRef](#)] [[PubMed](#)]
15. Griffioen, J.W.; Oonincx, P.J. Suitability of Low-Frequency Navigation Systems for Artillery Positioning in a GNSS Denied Environment. *J. Navig.* **2012**, *66*, 35–48. [[CrossRef](#)]
16. Johnson, G.W.; Swaszek, P.F.; Hartnett, R.J.; Shalaev, R.; Wiggins, M. An Evaluation of Eloran as a Backup to GPS. In Proceedings of the 2007 IEEE Conference on Technologies for Homeland Security, Woburn, MA, USA, 16–17 May 2007. [[CrossRef](#)]
17. Offermans, G.; Bartlett, S.; Schue, C. Providing a Resilient Timing and UTC Service Using eLoran in the United States. *Navigation* **2017**, *64*, 339–349. [[CrossRef](#)]
18. Son, P.-W.; Park, S.G.; Han, Y.; Seo, K. eLoran: Resilient Positioning, Navigation, and Timing Infrastructure in Maritime Areas. *IEEE Access* **2020**, *8*, 193708–193716. [[CrossRef](#)]
19. Narkus-Kramer, M.; Scales, W.; Calle, E. Evaluating Eloran as a Backup for Surveillance and Navigation: A Comparative Cost Analysis. In Proceedings of the the 2009 IEEE/AIAA 28th Digital Avionics Systems Conference, Orlando, FL, USA, 23–29 October 2009. [[CrossRef](#)]
20. Pierce, J.A. An Introduction to Loran. *IEEE Aerosp. Electron. Syst. Mag.* **1990**, *5*, 16–33. [[CrossRef](#)]
21. Peterson, B.; Hartnett, R.; Bruckner, D.; Heatherington, R.; Fiedler, R. Integrated GPS/LORAN: Structures and Issues. *Navigation* **1998**, *45*, 183–193. [[CrossRef](#)]
22. Peterson, B.B.; Lo, S.; Enge, P. Integrating Loran and Gns for Safety of Life Applications. In Proceedings of the Institute of Navigation GNSS Conference, Savannah, GA, USA, 16–19 September 2008.
23. Groves, P.D. *Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems*; Boston Artech House: Norwood, MA, USA, 2013.
24. Yan, B.; Li, Y.; Guo, W.; Hua, Y. High-Accuracy Positioning Based on Pseudo-Ranges: Integrated Difference and Performance Analysis of the Loran System. *Sensors* **2020**, *20*, 4436. [[CrossRef](#)]
25. Kim, Y.; Fang, T.H.; Kim, D.; Seo, K.; Park, S.H. Loran-C Multiple Chain Positioning Using Toa Measurements. *J. Navig. Port Res.* **2019**, *43*, 23–32. [[CrossRef](#)]
26. Fang, T.H.; Kim, Y.; Park, S.G.; Seo, K.; Park, S.H. GPS and eLoran Integrated Navigation for Marine Applications Using Augmented Measurement Equation Based on Range Domain. *Int. J. Control. Autom. Syst.* **2020**, *18*, 2349–2359. [[CrossRef](#)]
27. Coleman, T.F.; Li, Y. A Reflective Newton Method for Minimizing a Quadratic Function Subject to Bounds on Some of the Variables. *SIAM J. Optim.* **1996**, *6*, 1040–1058. [[CrossRef](#)]
28. Morrison, D.R.; Jacobson, S.; Sauppe, J.J.; Sewell, E.C. Branch-and-bound algorithms: A survey of recent advances in searching, branching, and pruning. *Discret. Optim.* **2016**, *19*, 79–102. [[CrossRef](#)]
29. Shapiro, L.D. Time synchronization from Loran-C. *IEEE Spectr.* **1968**, *5*, 46–55. [[CrossRef](#)]
30. Yan, W.; Dong, M.; Li, S.; Yang, C.; Yuan, J.; Hu, Z.; Hua, Y. An eLoran Signal Cycle Identification Method Based on Joint Time–Frequency Domain. *Remote Sens.* **2022**, *14*, 250. [[CrossRef](#)]
31. Johler, J. Propagation of the Low-Frequency Radio Signal. *Proc. IRE* **1962**, *50*, 404–427. [[CrossRef](#)]
32. Samaddar, S.N. The Theory of Loran-C Ground Wave Propagation—A Review. *Navigation* **1979**, *26*, 173–187. [[CrossRef](#)]
33. Yang, S.H.; Lee, C.B.; Lee, Y.K.; Lee, J.K.; Kim, Y.J.; Lee, S.J. Accuracy Improvement Technique for Timing Application of Loran-C Signal. *IEEE Trans. Instrum. Meas.* **2011**, *60*, 2648–2654. [[CrossRef](#)]
34. Li, Y.; Hua, Y.; Yan, B.; Guo, W. Research on the eLoran Differential Timing Method. *Sensors* **2020**, *20*, 6518. [[CrossRef](#)]
35. Son, P.-W.; Rhee, J.H.; Hwang, J.; Seo, J. Universal Kriging for Loran Asf Map Generation. *IEEE Trans. Aerosp. Electron. Syst.* **2018**, *55*, 1828–1842. [[CrossRef](#)]
36. Kim, Y.; Park, S.H.; Fang, T.H.; Seo, K. A Test of a Loran-C Multi-Chain Positioning Method Using Asf Estimates. In Proceedings of the the 2019 European Navigation Conference (ENC), Warsaw, Poland, 9–12 April 2019. [[CrossRef](#)]
37. Felski, A.; Jaskólski, K.; Zwolak, K.; Piskur, P. Analysis of Satellite Compass Error’s Spectrum. *Sensors* **2020**, *20*, 4067. [[CrossRef](#)]
38. Wang, L.-L.; Liang, Z.-C.; Pu, Y.-R.; Xi, X.-L. Method for Loran-C Additional Secondary Factor Correction Based on Neural Network and Transfer Learning. *IEEE Antennas Wirel. Propag. Lett.* **2021**, *21*, 332–336. [[CrossRef](#)]
39. Pu, Y.; Zheng, X.; Wang, D.-D.; Xi, X. Accuracy Improvement Model for Predicting Propagation Delay of Loran-C Signal Over a Long Distance. *IEEE Antennas Wirel. Propag. Lett.* **2021**, *20*, 582–586. [[CrossRef](#)]
40. Li, Y.; Hua, Y.; Yan, B.; Guo, W. Experimental Study on a Modified Method for Propagation Delay of Long Wave Signal. *IEEE Antennas Wirel. Propag. Lett.* **2019**, *18*, 1716–1720. [[CrossRef](#)]
41. Fisikopoulos, V. Geodesic Algorithms: An Experimental Study. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2019**, *42*, 45–47. [[CrossRef](#)]
42. Tseng, W.-K.; Guo, J.-L.; Liu, C.-P. A Comparison of Great Circle, Great Ellipse, and Geodesic Sailing. *J. Mar. Sci. Technol.* **2013**, *21*, 7. [[CrossRef](#)]
43. Kornerup, P.; Muller, J.-M. Choosing starting values for certain Newton–Raphson iterations. *Theor. Comput. Sci.* **2005**, *351*, 101–110. [[CrossRef](#)]
44. Moré, J.J. The Levenberg-Marquardt algorithm: Implementation and theory. In *Numerical Analysis*; Springer: Berlin, Germany, 1978; pp. 105–116.
45. Land, A.H.; Doig, A.G. An Automatic Method of Solving Discrete Programming Problems. *Econometrica* **1960**, *28*, 497. [[CrossRef](#)]

46. Steihaug, T. The Conjugate Gradient Method and Trust Regions in Large Scale Optimization. *SIAM J. Numer. Anal.* **1983**, *20*, 626–637. [[CrossRef](#)]
47. Byrd, R.H.; Schnabel, R.B.; Shultz, G.A. Approximate Solution of the Trust Region Problem by Minimization over Two-Dimensional Subspaces. *Math. Program.* **1988**, *40*, 247–263. [[CrossRef](#)]
48. Gratton, S.; Sartenaer, A.; Toint, P.L. Recursive Trust-Region Methods for Multiscale Nonlinear Optimization. *SIAM J. Optim.* **2008**, *19*, 414–444. [[CrossRef](#)]
49. Battiti, R. First- and Second-Order Methods for Learning: Between Steepest Descent and Newton's Method. *Neural Comput.* **1992**, *4*, 141–166. [[CrossRef](#)]
50. Ueda, K.; Yamashita, N. On a Global Complexity Bound of the Levenberg-Marquardt Method. *J. Optim. Theory Appl.* **2010**, *147*, 443–453. [[CrossRef](#)]