Fast hardware-oriented algorithm for 3D positioning in line-of-sight and single bounced non-line-of-sight environments

Cem Yağlı1,∗, Emre Özen2, Arif Akkeleș3
1Department of Mathematics, Faculty of Arts and Science, Eastern Mediterranean University, Famagusta, North Cyprus Mersin 10, Turkey
2Information Technology, School of Computing and Technologies, Eastern Mediterranean University, Famagusta, North Cyprus Mersin 10, Turkey

Received: 21.08.2020 • Accepted/Published Online: 12.03.2021 • Final Version: 26.07.2021

Abstract: The ability to find the location of a mobile object and track it in two-dimensional (2D) and three-dimensional (3D) space is an indispensable feature of wireless communication systems. In particular, the increase in demand for using self-navigation technology in unmanned vehicles is attracting additional attention to this area of research. The methods and techniques developed for these systems compete in terms of simplicity, performance, and accuracy. However, the majority of solutions focus on only one of these performance metrics and are not applicable to the projected systems. In this study, a new location estimation solution that satisfies all three performance metrics (simplicity, accuracy, and efficiency) is achieved. The developed algorithms can be executed on a mobile object to allow it to learn its position in space. Furthermore, the simplicity of the mathematical operations used, namely binary shift and add operations, makes our algorithm hardware-friendly. Accelerated coordinate rotation digital computer (CORDIC) algorithms are used for the first time with line-of-sight (LOS) and single-bounced-scattering non-line-of-sight (SBS NLOS) signal arriving approaches for mobile object self-positioning. This study also introduces a newly developed ‘vector-breaking’ method (VBM) to estimate the location of a mobile object using the arriving signals from scatters with unknown locations. Test results using the developed algorithm show that the projected level of accuracy has been achieved. Furthermore, the developed algorithm performs better than the solutions currently available in the field.

Key words: 3D positioning, self positioning, line-of-sight, non-line-of-sight, coordinate rotation digital computer, time difference of arrival

1. Introduction

Mobile object (MO) location estimation in 2D and 3D space has become a significant area of research since cellular wireless communication systems and their applications have established a growing place in everyday life. Today, mobile phones have turned into extensions of our hands so much that people have become addicted to mobile phones and cannot function without them. Many studies on MO localisation have already been encouraged and supported by the governments, especially for location determination [9, 10]. In 2001, the US Federal Communications Commission (FCC) declared E911 error margin accuracy requirements of 50 m to 150 m for handset-based and 100 m to 300 m for network-based localisation[9]. In the most recent update (November 2019)[52], the FCC made a vertical (z-axis) accuracy of 3 m compulsory and claimed that the existing accuracy requirements for the horizontal plane (x-axis and y-axis) would also be tightened and refined by 2023. Various
technologies that are vital to public safety utilise location estimation. Electronic bracelets were used first for locating and retrieving lost children and pets. Today, they are used to track down people with coronavirus disease 2019 (COVID-19) and those in contact with them\textsuperscript{[4, 5]}. Localisation also plays a special role in fleet management, in that it allows operators (e.g., police forces, taxi and shuttle transportation managers, directors of emergency vehicles) to track and manage their units in an efficient way \textsuperscript{[6–8]}. Location-specific advertisements and marketing are new trends in the business world. These technologies are used to attract potential customers to products and services that are geographically near them \textsuperscript{[11–16]}. The design, production, and management technologies for unmanned vehicles have been improving at a dizzying speed over the last decade. Some crucial abilities, like self-departure, self-navigation, and self-landing, have been added to the newest generation of drones, allowing them to be operated in every type of condition. The self-navigation ability of a drone requires precise and fast self-positioning techniques that must simultaneously be energy efficient. While the majority of location estimation technologies use satellite global positioning signals (GPS), in dense residential areas and tower blocks, cellular networks’ more powerful signals yield better results \textsuperscript{[1–3]}.

This study aimed to develop a new technique for MO self-positioning using wireless network signals. The developed positioning algorithm will work for both line-of-sight (LOS) and single-bounced-scattering non-line-of-sight (SBS NLOS) approaches. All required calculations were realised via the mobile object instead of the base stations.

Simulation results showed that the proposed technique has good accuracy with better efficiency in terms of computational cost, and therefore reduced energy consumption, relative to its competitors.

In the literature, various solutions to the localisation problem are discussed. All of them are built on a model containing base stations (BSs), whose geographic locations are known, and a MO, whose unknown location is estimated using the signals transmitted between the BSs. Many of these methods use fully synchronous clocks to calculate the distances between the components of the model (i.e., the BSs and the MO) via the time-of-arrival (TOA) technique\textsuperscript{[19, 23, 24, 48, 51]}. Time-difference-of-arrival (TDOA) is the second popular distance calculation technique and can be used even when the MO clock is not synchronised with those of the BSs \textsuperscript{[47, 48, 51]}. Both techniques use total signal transmission time to calculate the distance between the components. Some of the techniques use the direction of signal propagation, determined with antenna arrays. The angle-of-departure (AOD) and angle-of-arrival (AOA) values are used in these location estimation processes. The AOA can be calculated using the measured TDOA between the individual elements of the array \textsuperscript{[48, 49, 51]}. Detecting and tracing flying objects are essential applications of angle tracking radars, especially in airports. These systems measure the angular changes of the received signals to track their targets \textsuperscript{[50]}. How the signals are received also plays a huge role in localisation. LOS is one of the most widely acclaimed approaches and is used in many studies \textsuperscript{[17–21, 42–44]} because of its simplicity and relatively low cost of computation. The LOS approach is built on a geometric model, since, if there are no obstacles between the BSs and the MO, the sent signals may travel directly to their destinations. The TOA technique with the LOS approach examines the intersection of spheres in 3D (or circles in 2D) space, where the centre points are the known coordinates of the BSs, and the radii are the calculated distances from the MO \textsuperscript{[47, 51]}. The common intersection point of all the spheres is taken to be the location of the MO in 3D space. On the other hand, the TDOA LOS technique intersects hyperboloids in 3D (or hyperboles in 2D) space to locate the MO; the hyperboloids are drawn using the calculated values of the time differences of arriving signals \textsuperscript{[27, 47, 51]}. In practice, observed time-difference-of-arrival (OTDOA) and uplink-time-difference-of-arrival (UTDOA) are the two main forms of the TDOA approach. While OTDOA
uses the signals transmitted by the BSs and received by the MO to self-position, UTDOA uses the transmitted signals of the MO, which are received by multiple BSs, to estimate the position of the MO on a BS [51]. Cell identification (CID) and enhanced cell identification (eCID) are the basic positioning methods used by GSM providers. These methods detect the serving cell (BS) identity information and its known effective broadcasting range to estimate the possible location of the MO. Additionally, the enhanced version uses the received signals’ power and quality along with the Rx-Tx (xth received-transmitted) time difference values of the serving cells to improve its accuracy [29, 30, 51]. The received-signal-strength (RSS) method is another location estimation method [23, 51] that uses the differences in signal strength between emitted and received signals. Although LOS is a widely used approach, it encounters difficulties in dense areas and mountainous terrain where direct sight is blocked. In such places, emitted signals may be absorbed by or bounced from scatters (SCs) with unknown locations. The NLOS signal arriving approach aims to estimate the location of an MO under these conditions. Despite its computational complexity, the NLOS approach is more compatible with the multipath nature of the signals, making it more popular in recent studies [22–25, 46]. In the majority of these studies, the location of the MO is calculated on the BSs, and if necessary, this information is sent to the MO, generating an additional networking cost and increasing the time consumed. In only a few study, the MO can determine its location by using the signals received from the BSs [26, 53, 54]. Up to this point, the crucial elements of the methods and models described have been the algorithms and calculation techniques. Similar geometric models and signal arriving approaches have been used in several solutions and differ from each other only in their calculation techniques. Trigonometric, probabilistic, and statistical calculation techniques [27, 28, 33, 34, 46] are frequently used within localisation models to satisfy the requirements of a variety of applications which each have different accuracy and efficiency requirements in terms of speed and cost of calculation. The coordinate rotation digital computer (CORDIC) algorithm [35–37, 45] was developed in 1956 and used as a real-time digital resolver on the navigation computers of B-58 bombers to obtain accurate and fast results. It was also used in the navigational systems of the Apollo program’s Lunar Roving Vehicle. The technical requirements of CORDIC are very low and it can be executed on any kind of central processing unit (CPU) because it uses simple shift-add operations for several computing tasks. Tasks that can be computed via the CORDIC algorithm include trigonometric, hyperbolic, and logarithmic functions, division, square root calculations, real and complex multiplication, solutions of linear systems, and eigenvalue estimations. Thus, CORDIC reduces computational costs and makes solutions applicable to both hardware and software by reducing the processor chip area and yielding energy consumption.

In order to achieve the aims of this study, a unique graphical model was constructed that allows a MO to self-position using both LOS and SBS NLOS approaches. CORDIC was used to speed up the calculations and to reduce both computational cost and energy consumption. There are existing solutions that use CORDIC with the LOS approach [42–44, 47], but in our solution, CORDIC is used for the first time with both LOS and SBS NLOS approaches. This study also developed and used a new algorithm for self-localisation called ‘ACE’, which is compatible with both LOS and SBS NLOS signal arriving approaches. Our model was simulated using MATLAB (Matrix Laboratory) version R2014b. Accuracy and computational cost were used as performance indicators. Our solution presented remarkable accuracy, with a computational cost four times better than its competitor’s [46]. Major contributions of this study include (i) the construction of a unique model solution that is accurate, efficient, and applicable to any MO which must self-position in both LOS and NLOS signal arriving conditions, (ii) the development of a new algorithm that estimates the position of an MO using the arriving signals transmitted by BSs bounced from SCs with unknown locations, and (iii) the novel implementation of
CORDIC with SBS NLOS signal approaching.

2. Model
The method by which received signals are transformed into a form that our algorithms can use is explained in Section 2.1. Section 2.2 presents the components of the problem, their relationships with each other, and the simulation model used for the solution in 3D space.

2.1. Technical model
Our solution requires the cost-effectiveness and speed of LOS systems and the accuracy of NLOS systems as they have been reviewed in the literature. This model has BSs with known positions and SCs and an MO with unknown positions. The clocks of the BSs are assumed to be synchronised with each other but not with the clock of MO; by using the TDOA technique, the total distance travelled by each signal through a BS, to an SC, and then to the MO can be calculated. It is also assumed that the MO and the BSs have antenna arrays, so, by using SAGE-like [38] algorithms, the AOD, AOA, and TDOA values can be measured [39]. These known and calculated elements were used to construct our geometric model of the solution. Signals in real-life broadcasting have a multipath nature, so they can reach their destination in many ways, including directly, scattered with single-bounce, or scattered with many bounces. On every bounce scattering, signals lose their power and start to delay. In this study, only the direct and single-bounce scattered arriving signals were considered. As described in Seow and Tan [40], a two-step proximity detection algorithm was used to distinguish and ignore the rest.

2.2. Geometric model
Our model contains five components illustrated in Figure 1. These components are the MO, two BSs, and two SCs. In order to simplify this model, the techniques described in Fang [41] were used: A set of local, right-handed orthogonal axes was chosen and the nearest BS was carried to the origin to become BS$_1$ (0, 0, 0). One of the axes was used as the station baseline (i.e., the x-axis), and the other axes were placed orthogonal to the x-axis. Then, the whole system and its components were shifted and rotated until the second BS was placed on the x-axis. The remaining components’ coordinates in the simplified model are therefore BS$_2$ (xb$_2$, 0, 0), SC$_1$ (xs$_1$, ys$_1$, zs$_1$), SC$_2$ (xs$_2$, ys$_2$, zs$_2$), and MO (xm, ym, zm). Figure 2 shows the model after this geometric transformation, with transformed points and rotated axes.

The angles of arrival (aoa$_1$, aoa$_2$, aoa$_3$, aoa$_4$), angles of departure (aod$_1$, aod$_2$, aod$_3$, aod$_4$), and the coordinates of BS$_1$ and BS$_2$ are known (see Section 2.1). The total travel time of each signal through the BSs, to the SCs, and then to the MO can be calculated using TDOA; therefore, the total distance travelled by the signals can also be calculated (DL$_1$, DL$_2$). However, the locations of the SCs and the mobile object MO are unknown. Figure 2 presents the elements of the model on the X–Z and X–Y planes.

DL$_1$ is the calculated distance travelled by the signal transmitted from BS$_1$ to the MO. This signal can reach the MO directly, without bouncing any scatter; this is the LOS part of the model. There is also the SBS NLOS part where the signal bounces with a scatter on its route. On the X–Y plane, r$_1$ and r$_2$ are the distances from BS$_1$ to SC$_1$ and from SC$_1$ to MO, respectively; their sum must be equal to DL$_1$. The sum of r$_5$ and r$_6$ on the X–Z plane must also be equal to DL$_1$. Although the magnitudes of r$_1$, r$_2$, r$_3$, r$_4$, r$_5$, r$_6$, r$_7$, and r$_8$ are unknown, the equations can be written with the calculated magnitudes of DL$_1$, and DL$_2$ as follows:

\[
DL_1 = r_1 + r_2 = r_5 + r_6 \quad \text{and} \quad DL_2 = r_3 + r_4 = r_7 + r_8
\]
Figure 1. Components of the model.

Figure 2. Elements of the geometrically transformed model.

To develop an applicable solution, the projections of the components onto the X–Y plane are focused on first. This approach delivers a solution in 2D space. Projecting the components onto the X–Z plane brings the solution to the 3D space. As demonstrated in Figure 3, the SC₁ can be located anywhere on the line |OA| and the MO must be placed on the line |AB|. The first LOS case is when SC₁ and the MO share a location at point A. Since DL₁ is the sum of r₁ and r₂, r₁ should be equal to DL₁ and r₂ should be equal to zero. The second LOS case places the MO on point B, and BS₁ and SC₁ share a location on the origin (O) of |OA|. In this case, r₁ must be zero and r₂ must be equal to DL₁. Other positions on |AB| represent the SBS NLOS cases since they require r₁ > 0 and r₂ > 0. In Figure 3, two BS-SC-MO sets are merged to estimate the
MO position on the X–Y plane. The location of the MO must be at the intersection point of the lines $|AB|$ and $|CE|$. The angles $n$, $s$, and $q$ in Figure 3 and Figure 4 are equal to the angles $\text{aoa}_1$, $\text{aoa}_2$, and $\text{aoa}_3$, respectively, so the coordinates of A, B, C, D, and E can be calculated and the equations for $|AB|$, $|DC|$, and $|EF|$ can be written. MO $(x_m, z_m)$ must be a point on the line $|EF|$, and since $x_m$ is already estimated, the last unknown of the model, $z_m$, can be calculated using the equations for $|EF|$ and $x_m$.

3. Algorithms of estimation

Based on the literature review conducted, this study aimed to develop a solution for the location estimation problem that is software- and hardware-oriented, computationally cost-effective, fast, and accurate. Consequently, we decided to use CORDIC methods within the solution algorithms. Three well-known CORDIC methods serve this purpose: the original CORDIC and two faster variations of it, known as the fixed angle rotation method (FARM) [42, 43] and the dynamic angle rotation method (DARM) [44]. They are cost-effective substitutes for the trigonometric functions. In this part of the work, a set of algorithms we named ACE (Arif-Cem-Emre) were developed and constructed on the geometrically transformed model, and its components are described in Section 2.

3.1. ACE algorithm

The ACE algorithm is composed of eight main steps. The first five are for finding the coordinates of the MO on the X–Y plane in 2D space. The remaining three steps are for extending the 2D solution to the 3D space by including the X–Z pane and its components. The steps of the ACE algorithm are depicted in Figure 3.
5. Two methods are used in ACE to emulate the trigonometric functions: FARM [44] and a vector-breaking method (VBM) were developed in this study by the vector lengthening method [44]. Salamah and Doukhnitch [42] showed that the original CORDIC algorithm has some performance leakages, as it requires numerous sub-iterations and a factor compensation process on every main iteration. FARM uses a wiser algorithm that excludes unnecessary iterations and compensation processes from each rotation. It uses recursive iterations, as its step sizes are determined by a fixed value (angle in radians) $\sigma = \arcsin(2^{-k})$, where $k$ is a parameter to decide the level of accuracy. $M$ is the circular rotation matrix [47]. Here, $u = 2^{-(2k+1)}$, $v = 2^{-k}$, and $s = \pm 1$ is the operators used to decide the direction of rotation. Using basic arithmetic operations and binary shifts instead of trigonometric functions in any rotation or coordinate calculation reduces the computational cost of the process. Therefore, FARM is more useful in applications where there are limited hardware resources.

$$M = \begin{pmatrix} \cos(\sigma) & -\sin(\sigma) \\ \sin(\sigma) & \cos(\sigma) \end{pmatrix} = \begin{pmatrix} 1 & -2^{-k} \\ 2^{-(2k+1)} & 1 - 2^{-k} \end{pmatrix}$$

(2)

The vector coordinates are changing recursively as follows:

$$V(j + 1) = \begin{pmatrix} x_{j+1} \\ y_{j+1} \end{pmatrix} = \begin{pmatrix} x_j - x_j u + s y_j v \\ y_j - y_j u - s x_j v \end{pmatrix}$$

(3)
3.1.1. Fixed angle rotation method

FARM [44] is used to place the head of the related vector on the correct coordinates to emulate the trigonometric functions. It uses equations (2) and (3) to calculate the coordinates of the vector head in 2D space while the vector is rotated at a fixed stepping angle until the total angle of rotation get close to the target angle with an error $\epsilon < 10^{-k}$. The FARM algorithm can be found in Appendix 1. This method is used to position the vectors $\overrightarrow{OA}$, $\overrightarrow{DC}$, $\overrightarrow{OE}$ with lengths $DL_1$, $DL_2$, $DL_1$ and angles $aod_1$, $aod_2$, $aod_3$, respectively, on the graph. It is also used to place the stepping vectors ($\overrightarrow{V_{11}}$, $\overrightarrow{V_{12}}$), which are essential elements of the vector-breaking method, over the model.

3.1.2. Vector breaking method

As discussed in Section 2.2 and illustrated in Figure 3, points A and C represent the possible first cases of LOS MO positions. As shown in Figure 6, if the MO is placed on point A, from (1) it is required that $r_1 = DL_1$, and $r_2 = 0$. When the MO is placed in SBS NLOS case, it requires $r_1 < DL_1$, $r_2 > 0$.
When they are drawn on the same graph, it looks as though the vector $\overrightarrow{OA}$ has been broken into two pieces, $r_1$, and $r_2$, and hence the name ‘vector-breaking method’. The method shortens $r_1$ and lengthens $r_2$ while keeping the magnitude, $DL_1$, constant. For this purpose, the size of $r_1$ is reduced by subtracting the stepper vector $V_{11}$ from $r_1$ on every turn of the iteration. In order to keep $DL_1$ constant, the second stepper vector $V_{12}$ is recursively added to $r_2$ within an inner loop. The stepper vectors $V_{11}$ and $V_{12}$ are horizontally $\mu$ meters lengthened vectors which are then rotated with angles $aod_1$ and $aoa_1$, respectively. $\mu$ is the second parameter, along with $k$, that has a significant effect on the accuracy of the model. The VBM is presented in Figure 6 and the algorithm is described in Appendix 1.

4. Simulation

The graphical model described in Section 2 and the algorithms described in Section 3 were tested for accuracy and computational cost. For this purpose, two applications were developed and executed in MATLAB (MathWorks, Inc., Natick, MA, USA). The first application follows an algorithm (see Figure 7) to generate sample sets of the graphical model components ($BS_1$, $BS_2$, $SC_1$, $SC_2$, and MO) and elements ($DL_1$, $DL_2$, $aod_1$, $aod_2$, $aod_3$, $aod_4$, $aoa_1$, $aoa_2$, $aoa_3$, and $aoa_4$) that are illustrated in Figure 2. They are randomly placed into a 3D space bounded at 500 m on the x-axis, 500 m on the y-axis, and 100 m on the z-axis. The second application was developed using the ACE algorithm explained in Section 3.1 and illustrated in Figure 5. The ACE algorithm was tested at a 95% confidence level. The tests were repeated with different accuracy levels ($k$) in order to comprehend the influence of chosen values of $k$ on the resulting errors and computational costs of the processes. As mentioned in Section 3.1, $k$ is a significant parameter in vector rotation and lengthening, and therefore has an important role in the ACE algorithm. The weights of the operations, shown in Table , were used for calculating the computational cost of each process [45]. For each sample set, the calculated computational cost and the errors in the distance between the estimated and actual coordinates of the MO were recorded to evaluate the accuracy of the ACE algorithm.

**Figure 7.** Sample generation algorithm.
Table. Weights of the operations.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Addition</th>
<th>Subtraction</th>
<th>Shift</th>
<th>Multiplication</th>
<th>Division</th>
<th>Sin</th>
<th>Cos</th>
<th>Tan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>40</td>
<td>40</td>
<td>404</td>
<td>404</td>
<td>1448</td>
</tr>
</tbody>
</table>

4.1. Simulation results

The recorded accuracies and computational costs of the ACE algorithm with different k values are illustrated in Figure 8 and Figure 9. Figure 8 presents the exponentially improving accuracy of the ACE algorithm as k is incrementally increased. Figure 9 shows the expected side effect of increasing k, that is, the exponentially rising computational costs. As explained in Section 1, the FCC’s 2001 E911 accuracy requirements have been accepted as mandatory standards in many studies on location estimation. However, the 2019 declaration of the FCC requires an important revision to the existing localisation techniques in order to fulfil the new standards. Many techniques which satisfied the FCC requirements prior to the update have similar working conditions to ACE in that they (a) use an SBS NLOS signal arriving approach and (b) perform location estimation on the MO and examine the accuracy of their estimations. The root mean square error of the solution by Behailu and Tobias (BTS) [46] is 34.5 m. The solution by Khajehnouri and Sayed (KSS) [22] has a mean error of 25 m at its best performance. The direction of arrival-based localisation algorithm developed by Li and Lu [53] presented estimation errors (400 m × 400 m horizontal scale) of 16 m under its best conditions. The weighted centralized cooperative single-bounced scattering NLOS technique by Liu, Zhu, Jiang, and Huang [54] has an average location estimation error of 12 m.

![Figure 8](image_url)

**Figure 8.** Average absolute error versus accuracy level (k).

In recent studies [55, 56], more accurate localisation techniques have been introduced; however, they all are designed to work on the BSs and therefore are not comparable to our study. As the test results shown in Figure 8 reveal, ACE satisfied the new FCC rules under all conditions of k. One of the best-known features of the CORDIC algorithm is the computational efficiency it provides while imitating trigonometric functions. To prove our solution is a nontrivial contribution to the field, we decided that the BTS would be used as the...
benchmark by which to measure our algorithm since it uses matrix calculations to estimate the position of the MO and is expected to present better computational efficiency than the other techniques referred to above. Since the original study does not include metrics on its computational cost, we considered the weights of the operations given on Table along with the methods and calculations described in its text and calculated its computational cost. We found the cost of BTS to be 43,852, which is four times the cost of ACE under its best condition of $k = 7$. The computational cost comparison between ACE and BTS is demonstrated in Figure 9. The presented efficiency and precision of the ACE algorithm make it a significant option among its alternatives.

5. Conclusion
In this paper, the details of a novel location estimation technique are given. The first contribution is a new algorithm, named ACE, which (1) allows a MO to self-position using the fixed angle rotation method (FARM) of the CORDIC algorithm with both LOS and SBS NLOS signal arriving approaches and (2) is a hardware-oriented solution, as the CORDIC method performs all calculations with basic binary shift and add operations supported by all CPUs since 1956. The second contribution of the study is a new calculation method, VBM, used for estimating the location of an MO using the signals bounced from scatters with unknown locations. CORDIC is used for the first time with SBS NLOS signal arriving approach, and this is the third contribution of the study. The ACE algorithm performed remarkably well relative to the accuracy and computational costs consented to in the field of research. The FARM variation of the CORDIC algorithm was used in our study to speed up and reduce the cost of computations. There is another promising variation of CORDIC, DARM, discussed in the literature. Dynamically changing angle sizes optimise the number of rotations; as a result, DARM can decrease the computational cost relative to FARM. In future studies, ACE can be modified by replacing FARM with DARM. ACE can also be extended to work with multibounced scattering NLOS and LOS arriving signals.

References


Algorithm 1 Methods

Given:

- $DL_1$ is the length of vector $\overrightarrow{OA}$
- $A$ is the edge point of vector $\overrightarrow{OA}$
- $\hat{A}_0$ is the Initial position of $A$
- $\hat{A}_i$ is the $i^{th}$ position of point $A$
- $r_{1i}, r_{2i}$ are broken pieces of vector $\overrightarrow{OA}$ at the $i^{th}$ iteration

Error: $\epsilon = 10^{-k}$

$v = 2^{-k}$

$u = 2^{-(2k+1)}$

$\overrightarrow{V_{11}}$ and $\overrightarrow{V_{12}}$ are stepping vectors generated by FARM

procedure FARM (item target_angle, item vlength, item k)

// FARM: Fixed angle rotation method, vlength: vector length

current_angle $\leftarrow 0$

$x \leftarrow$ vlength

$y \leftarrow 0$

if target_angle $> \pi/2$ then

$sign \leftarrow -1$

target_angle $\leftarrow \pi/2 - target_angle$

end if

while target_angle $- current_angle > \epsilon$ do

old_x $\leftarrow x$

old_y $\leftarrow y$

$x \leftarrow old_x - old_x * u + sign * old_y * v$

$y \leftarrow old_y - old_y * u - sign * old_x * v$

current_angle $\leftarrow current_angle + 2^{-k}$

end while

end procedure

procedure VBM (item t)

// The algorithm is illustrated in Figure 6

VBM: Vector breaking method, t: target point on $\overrightarrow{AJ}$

$A_0' \leftarrow A$

$r_{10} \leftarrow DL_1$

$r_{20} \leftarrow 0$

$i \leftarrow 0$

while $t - A_i' > \epsilon$ do

$r_{1i} \leftarrow r_{1i} - \overrightarrow{V_{11}}$

$i \leftarrow i + 1$

while $DL_1 - (r_{1i} + r_{2i}) > \epsilon$ do

$r_{2j} \leftarrow r_{2j} + \overrightarrow{V_{12}}$

$j \leftarrow j + 1$

end while

end while

end procedure