A triangle decomposition method for the mobility control of mecanum wheel-based robots

Kouame Yann Olivier Akansie[1], Rajashekhar C. Biradar[1], Rajendra Karthik[1], Geetha D. Devanagavi[2]

[1]School of Electronics and Communication Engineering, REVA University, Bangalore, India
[2]School of Computing and Information Technology, REVA University, Bangalore, India

ABSTRACT

Mobile robots are used in a variety of applications including research, education, healthcare, customer service, security and so on. Based upon the application, the robots employ different locomotion systems for their mobility. When it comes to rolling locomotion, the wheels used to provide mobility to robots can be categorized as: tracks, omnidirectional wheels, and unidirectional wheels with a steering system. The ability of omnidirectional wheels to drive machines in small spaces makes them interesting to use. Among the types of omnidirectional wheels, mecanum wheels are widely used due to their inherent benefits. With the right control strategy, robots equipped with mecanum wheels can move freely, in all possible directions. In this study, a triangle decomposition approach is employed for controlling omnidirectional mecanum wheel-based robots. The method consists of breaking down any path into a set of linear motions that can be horizontal, vertical, or oblique. Furthermore, the oblique paths are divided into smaller segments that can be resolved into a horizontal and vertical component in a right-angle triangle. The suggested control method is tested and proved on a simple scenario using Webots simulation software.

Corresponding Author:

Kouame Yann Olivier Akansie
School of Electronics and Communication Engineering, REVA University
Rukmini knowledge park, Kattigenahalli, Yelahanka, Bangalore, 560064, India
Email: akansiekouame@gmail.com

1. INTRODUCTION

Mobile robots have a variety of applications that meet many requirements, such as in medical facilities, hospitality and customer service, agriculture, warehousing and order fulfillment, package delivery, disaster recovery, and emergency response. Mobile robots should be able to move themselves to accomplish the assigned tasks. One of the challenges mobile robots face in performing any task is the toughness of the terrain to be navigated, which calls upon the robot's versatility to overcome these challenges. For that, they rely on locomotion systems which help them move as they want in accomplishing the given tasks [1], [2]. Focusing on rolling locomotion, different types of wheels are employed to provide mobility to robots. Omnidirectional mecanum wheels are often used as they are advantageous and allow robots to be versatiles [3], [4]. However, the type of control method employed for driving mecanum wheeled robots affects their performances. While some control methods offer seamless mobility to mecanum wheel robots with less drwabacks, others provide solutions with some limitations which disqualify them for applications in specific scenarios.

Numerous control methods have been proposed throughout time for path-tracking and motion control of mecanum wheel based mobile robots. Fuzzy proportional integral derivative (PID) control, robust sliding mode, non singular sliding mode, extended state observer-based sliding mode, bio-inspired motion control and energy optimal motion control are some of the well-known strategies suggested by researchers. To enhance the

performance of control systems for mecanum wheel robots, researchers combine the ideas of fuzzy logic with PID control. Fuzzy PID control's fundamental idea is to employ fuzzy logic to dynamically alter the PID controller's settings in response to the operating conditions of the system at hand. For example, paper [5] offers a thorough control scheme for mecanum wheel robots. The proposed scheme combines a fuzzy PID motion control algorithm with the popular and efficient A* search algorithm for path planning. Through this integration, the robot can quickly traverse its surroundings while instantly changing the control settings, resulting in more accurate and efficient motion control. Malayjerdi *et al.* [6] provides a control method which combines PID with self-tuning fuzzy logic to regulate the motion of the mobile robot effectively and precisely. The study provides a unique control strategy using a self-tuning fuzzy logic with PID control, resulting in precise and effective motion control. Chen [7] proposes a backstepping technique supported by a finite-time adaptive interval type-2 fuzzy controller. The technique consists of approximating the dynamics of a mecanum wheel platform using an improved version of the begian-melek-mendel (BMM) algorithm. An advantage of this technique is the availability of parameters required to approximate the dynamics, as they are easily available upon time invariant. Cao *et al.* [8] uses geometric modeling to determine parameters like overturning angle, kinematic slip factor and kinematic error in relation to the stability of the robot. The authors use fuzzy logic to adapt the robot movement through a PID controller and test their method on an eight-mecanum-wheeled robot for trajectory control.

A well-known method for controlling mecanum wheel robots is the robust sliding mode control, improves the performance and stability of mecanum wheel mobile robots, especially in the presence of ambiguities, disturbances, and modeling faults. This technique seeks to maintain the system's states on a particular sliding surface while preserving the system's stability and resistance to disturbances. An example of such method is found in [9], which suggests an approach to handle uncertainties and disturbances in the dynamics of a mecanum wheel mobile robot using a robust adaptive control technique. The suggested control approach makes real-time adjustments to the control settings using adaptive control techniques to improve the robot's performance and stability. Alakshendra and Chiddarwar [10] describes an adaptive robust control approach for the movement control of mecanum wheel robots which are subjected to dynamics uncertainties. The control approach combines robust control principles with adaptive control techniques to manage fluctuating circumstances and disturbances. Saad *et al.* [11] present a robust sliding mode controller (RSMC), which was created especially for omniwheeled mobile robots. The main contribution is the creation of the RSMC, which improves the robot's motion control robustness by maintaining precise trajectory tracking when subjected to external disturbances. Likewise, an omni-mecanum wheeled robot tracking control technique is suggested in [12]. The authors discuss the design of a control system that maintains accurate tracking of planned trajectories despite uncertainties and disruptions as the key contribution.

Another method employed by researchers to control mecanum wheel-based robots is the non-singular terminal sliding mode. This technique is a combination of sliding mode control and terminal sliding mode to produce accurate and quick trajectory tracking while preventing chattering and singularities in the control signals. An example of such an approach is presented in [13], which proposes a control strategy known as nonsingular terminal sliding control with a fuzzy wavelet network. The main add on in this research is the incorporation of fuzzy wavelet networks into the control law design, enabling efficient and adaptable control of the mecanum wheel robot. The strategy tries to eliminate chattering problems frequently associated with sliding mode control while achieving accurate trajectory tracking and reliable motion control. Sun *et al.* [14] suggests a similar approach for accurate trajectory tracking and reliable motion control of mecanum wheeled automated guided vehicles (AGVs). The authors apply the principle of fuzzy logic and nonsingular terminal sliding mode control to increase the AGV's agility, effectiveness, and precision when following predetermined paths in challenging settings. Sun *et al.* [15] provides a similar control scheme for mecanum wheeled omnidirectional mobile robots in response to the trajectory tracking issue and aims to achieve smooth path-following behavior for mecanum wheel robots. The suggested control strategy leads to an accurate path-following even in the presence of external disturbances. Wu and Huang [16] presents a fuzzy wavelet neural network-based adaptive fractional-order non-singular terminal sliding mode control technique for omnidirectional mobile robot manipulators. The authors combine a fuzzy wavelet neural network with a fractional-order control to accomplish an adaptive and accurate control of manipulators placed on omnidirectional mobile robots.

Among the techniques used in motion control and tracking of mecanum wheel robots, state observer-based sliding mode control is a technique which combines the ideas of sliding mode control and state observers to produce accurate and reliable motion control. The aim of this approach is to build a sliding mode controller to control the motion of the system by estimating the robot's states i.e., position, velocity, orientation and so on, using a state observer estimator. Wang *et al.* [17] employs such an approach by integrating slipping and skidding effects in the trajectory tracking control of mecanum wheeled mobile robots. The authors make use of an extended state observer to estimate uncertainties like slippage and skidding, and further use an adaptive sliding mode for motion control in their study to improves the robot's performance in trajectory tracking in the

presence of uncertainties and disturbances. A similar approach is discussed in [18], considering the effects of frictions instead for the design of the sliding mode control strategy. The proposed method offers friction compensation using a state observer, which results in an increased trajectory tracking and control performance. Yuan *et al.* [19] employ the same combination of techniques, using an extended state observer-based sliding control to compensate uncertainties and disturbances thereby enhancing the robot's trajectory tracking capabilities. A fixed-time observer-based adaptive fuzzy tracking control that guarantees transient performance during trajectory tracking is proposed in [20]. The authors combined a fixed-time state observer and an adaptive fuzzy control to improve robustness and trajectory tracking precision. Such a combination ensures transient performance for mobile robots with mecanum wheels. Many more techniques are suggested by researchers to control mecanum wheel robots, considering multiple factors that can interfere in the trajectory tracking and motion control. we can cite bio-inspired motion controls [21], energy optimal motion control [22] and time varying based PID control [23], [24]. The common limitations shared by the suggested methods include model dependencies, complex and time-consuming parameter tuning, computationally intensive algorithms and optimization processes, lack of adaptability to dynamic and evolving environments and finally challenges in practical implementations of the proposed techniques.

This paper proposes a lightweight algorithm that acts as an upper layer in combination with a simple PID controller for the omnidirectionality control of mecanum wheel robots. The proposed algorithm computes real time coordinates corresponding to the intermediate states of the robot from its origin to its destinantion using a triangle decomposition method. This method is applicable to linear motions, as an alternative approach to drive mecanum wheel robots in all possible directions without self-rotation, by breaking down any linear path into two basic linear motions i.e., vertical and horizontal motions. The benefits of this approach include less computational power, easy implementation, less execution time, and low energy consumption.

2. METHOD

The proposed algorithm for omnidirectional control of mecanum wheel robots is an implementation of the triangle decomposition method. The method approach mobility control by converting any path into multiple linear paths, which are further resolved into two basic linear motion i.e., horizontal and vertical motions, avoiding complex computations and reducing the amount of manoeuver needed to drive mecaum wheel robot to a known destination. Each resolved motion is further executed with the help of a simple PID controller once the coordinates and control parameters are computed. The method is presented as follows: i) a case study is carried out for path planning employing linear motions; ii) the triangle decomposition method is explained; and iii) an algorithm is proposed.

2.1. Case study

Let us consider three cases where a system with four mecanum wheels must move from a starting point to a destination while avoiding an obstacle. The study aims to determine the total distance to reach the destination in each case. The result of each case will finally be compared to determine the shortest path to reach the destination. Let us consider a scenario where a vehicle equipped with four mecanum wheels is to be moved from a starting point A to a destination point B, with an obstacle to be avoided as shown in Figure 1.
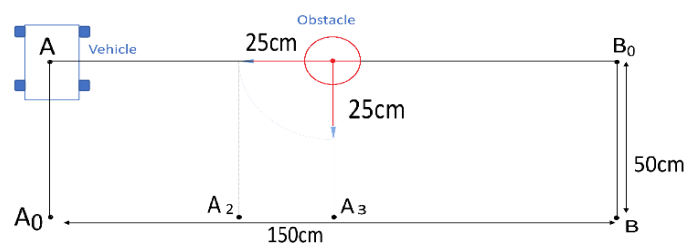

Figure 1. Case study scenario

Assume the clearance from the obstacle to be 'C' and the center of the obstacle to be O with $AB_0 = 150$cm, $AA_0 = 50$cm, $AO = 75$cm, and $C = 25$cm. We first propose different paths to be followed by the system from a combination of movements to avoid the obstacle and reach the destination. We then propose a path beyond conventional control's capability and analyze them all. For analysis, the vehicle and obstacle are represented with a single point. The analysis of each path is done in the upcoming sections.

2.1.1. Case 1: combination of horizontal and vertical movements

Assuming the case where simple horizontal and vertical movements can be used to solve the problem: reach the destination; The vehicle can follow the path shown in Figure 2. The path is made by the combination of a horizontal movement from left to right till the obstacle, a vertical movement going down, considering the clearance to be observed from the obstacle, and finally another horizontal movement from left to right, till the destination. Let AO be the distance from the starting point to the obstacle.
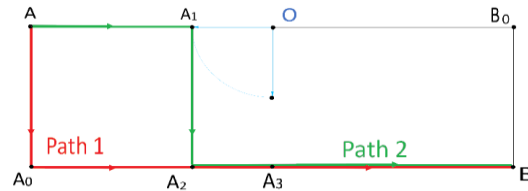
Figure 2. Path generation with combination of horizontal and vertical movements

Considering the clearance to be $C = A_1O$, let AA_1 be the distance from the starting point up to near the obstacle, such that,

$$AO = AA_1 + A_1O$$

AA_1 is therefore representing the first horizontal movement towards the destination. The second movement is represented by the portion A_1A_2 such that $A_1A_2 = 2 \times C$, and the third movement denoted as A_2B represents the second horizontal movement from A_2 till the destination B. Finally, the overall path can be broken down as,

$$AB = AA_1 + A_1A_2 + A_2B \tag{1}$$

Since $A_2B = A_0B - A_0A_2$ and $A_0A_2 = AA_1$; in (1) becomes: $AB = AA_1 + A_1A_2 + A_0B - AA_1$
An alternative to this path consists of only two movements i.e., AA_0 and A_0B such that,

$$AB = AA_0 + A_0B \tag{2}$$

For the theoretical calculations of paths to reach the destination through the combination of segments, practical values for each segment depicted in the study case scenario are given in Table 1.

Table 1. Table of the path's segments lengths

Segment	AA_0	AB_0	AA_1	A_1A_2	A_1O	A_3B	B_0B
Distance (cm)	50	150	50	50	25	75	50

Substituting the values of each segment in (1) gives,

$$AB = AA_1 + A_1A_2 + A_0B - AA_1 = 200 \text{ cm}$$

Substituting the values of each segment in (2) gives,

$$AB = AA_0 + A_0B = 200 \text{ cm}$$

we can see that in both cases, the length of the path remains the same. However, any one can be preferred, considering the sensing range of sensors used to detect the obstacle. In case the sensor is not able to detect the obstacle from so far in a practical scenario, in (1) will be used.

2.1.2. Case 1: combination of horizontal and obliques movements

Assuming the case where simple horizontal and oblique movements can be used to solve the problem: reach the destination, the vehicle can follow the path shown in Figure 3. The combination makes the path of a horizontal movement from left to right till the obstacle, an oblique movement going down, considering the

clearance to be observed from the obstacle, and finally, another horizontal movement from left to right till the destination. Considering the previous case (case 1), from in (1), we have,
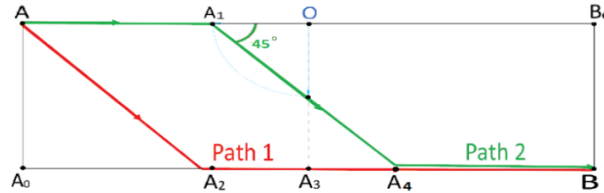
$$AB = AA_1 + A_1A_2 + A_2B$$

Figure 3. Path generation with combination of horizontal and oblique movements

The same equation is applicable in the current case, with slight changes in the segment A_1A_2 which becomes A_1A_4 as shown in Figure 3. From the path highlighted in Figure 3, we get the following,

$$AB = AA_1 + A_1A_4 + A_4B \tag{3}$$

With $A_4B = A_3B - A_2A_3$ and $A_1A_4 = \sqrt{A_1A_2{}^2 + A_2A_4{}^2}$ from Figure 3, in (3) becomes:
Considering the lengths provided for each segment in Table 1, the value of path AB in this case is found to be,

$$AB = AA_1 + \sqrt{A_1A_2{}^2 + A_2A_4{}^2} - (A_3B - A_2A_3) = 170.71 \text{ cm}$$

With reference from case 1, two cases are possible in this scenario as well, implying oblique and horizontal movements. From Figure 3, the equation for AB considering segments AA_2 and A_2B is as follows,

$$AB = AA_2 + A_2B \tag{4}$$

Replacing $AA_2 = A_1A_4$ and $A_2B = A_2A_3 + A_3B$ in (4) Considering the lengths provided for each segment in Table 1, the value of path AB in this case is found to be,

$$AB = \sqrt{A_1A_2{}^2 + A_2A_4{}^2} + A_2A_3 + A_3B = 170.71 \text{ cm}$$

2.1.3. Case 1: combination of obliques and horizontal movements

Assuming the case where simple horizontal and oblique movements can be used to solve the problem: reach the destination, The vehicle can follow the path shown in Figure 4. The path is made by combining an oblique movement going down, considering the clearance to be observed from the obstacle, and a horizontal movement from left to right till the destination. From Figure 4, the length of path 1 in reaching the destination point B is given by,

$$AB = AA_3 + A_3B \tag{5}$$

Given $AA_3 = \sqrt{AA_0{}^2 + A_0A_3{}^2}$, in (5) becomes,

$$AB = \sqrt{AA_0{}^2 + A_0A_3{}^2} + A_3B = 165.14 \text{ cm}$$

With an angle β given by: $\beta = \tan^{-1}\frac{AA_0}{A_0A_3} = 33.69°$
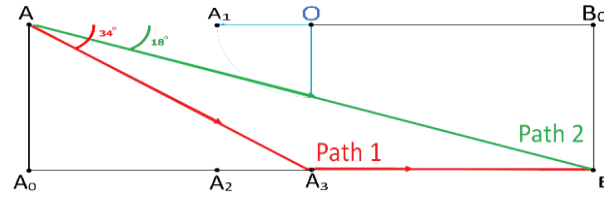
Figure 4. Path generation with combination of oblique and horizontal movements

Considering the direct path 2 for reaching the destination point B as shown in Figure 4, the expression and theoretical value for AB is given by,

$$AB = \sqrt{BB_0^2 + AB_0^2} = 158.11 \text{ cm} \tag{6}$$

With an angle β given by: $\beta = \tan^{-1}\frac{BB_0}{AB_0} = 18.43°$

        Table 2 summarizes the value of the angle of movement β in each scenario for different paths. It can be observed from the table that the shortest path is achieved in the third scenario with path 2, where a single oblique path is used to reach the destination point B. Since conventional mecanum wheel control methods employing simple PID controllers cannot achieve oblique movement at angles different than 45°, they would not be suitable for the third case scenario where variable angles are used to achieve shortest paths [25]–[27]. Consequently, an approach providing such kind of control is suggested to provide better results in terms of path planning for mecanum wheel robots.

Table 2. Comparison of paths length and angle of direction in each case

Parameters cases	Angle (deg)	Path length (cm)
Case 1	0	200
Case 2	45	170.71
Case 3 (path 1)	33.69	165.14
Case 3 (path 2)	18.43	158.11

2.2. Proposed mecanum wheel movements control using triangular decomposition method

        From the previous section, comparing the length of the traveling path in each case revealed that the third scenario, implying a single oblique linear movement at low angle led to the shortest path. Achieving a short path towards a destination reduces the efforts and time required to reach the destination. The principle behind this fact is shown in Figure 5.
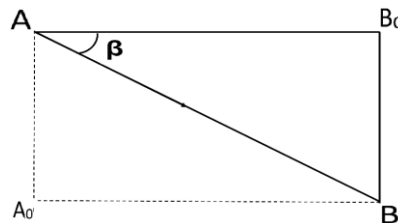

Figure 5. Relationship between hypotenuse of a right angle and adjacent side

Let's consider an oblique movement represented by the portion AB in the triangle AB_0B as shown in the Figure 5. The length of the portion AB with respect to the angle β is given by the expression:

$$AB = \frac{AB_0}{\cos \beta} \tag{7}$$

        In (7) implies that the length of the path AB for reaching the destiantion is inversely proportional to the cosine of the angle of movement β. In other words, the path's length increases with an increasing angle of movement. Hence, driving an mecanum wheel robot at lower angles, using oblique movements leads to short distances to reach the destination. Since conventional methods can drive a mecanum wheel vehicles at 45° only, reaching the destination using the shortest path will not be possible. Therefore, let us propose an approach

intended to drive a robot equipped with four mecanum wheels in all directions, at any angle, provided few parameters like angle, speed, and more.

Let us now consider a scenario where a four-mecanum wheel vehicle must move from point A to point B, such that ABC forms a triangle, right in C, and the direction angle formed with AC and BA is denoted as β. Since conventional PID based methods cannot drive the vehicle using linear motion if β is different 45°, we will divide the whole path into multiple segments that will be represented as small triangles as shown in Figure 6 to emulate a motion at any angle with succession of horizontal and vertical movements using the same method, such that,

$$\begin{cases} AB = AA_1 + A_1A_2 + \cdots + A_{n-2}A_{n-1} + A_{n-1}B \\ AC = AC_1 + C_1C_2 + \cdots + C_{n-2}C_{n-1} + C_{n-1}C \\ CB = C_1A_1 + B_1A_2 + \cdots + B_{n-2}A_{n-1} + B_{n-1}B \end{cases} \quad (8)$$

where 'n' represents the total number of segments in which the portion has been decomposed. As an example, for 'n=4', in (8) becomes,
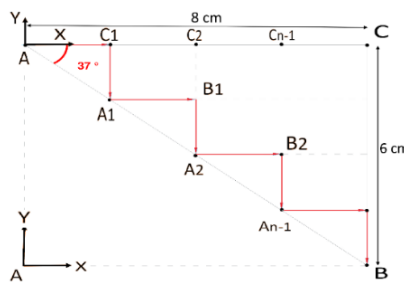

Figure 6. illustration of the triangle decomposition method for any angle linear movement

$$\begin{cases} AB = AA_1 + A_1A_2 + A_2A_3 + A_3B \\ AC = AC_1 + C_1C_2 + C_2C_3 + C_3C \\ CB = C_1A_1 + B_1A_2 + B_2A_3 + B_4B \end{cases}$$

Let 'a', 'b', and 'c', be the lengths of the decomposition segments with respect the portions 'AB', 'BC', and 'AC'. Assuming each portion is divided equally into several segments, in (8) for a total number of segments 'n=4' is given by,

$$\begin{cases} AB = a + a + a + a = 4 \times a \\ AC = c + c + c + c = 4 \times c \\ CB = b + b + b + b = 4 \times b \end{cases}$$

Since the total number of segments 'n=4', a standard equation for each portion with respect to the total number of segments is given by,

$$\begin{cases} AB = n \times a \\ AC = n \times c \\ CB = n \times b \end{cases}$$

The suggested approach aims to achieve any angle linear omnidirectionality by breaking down complex paths into linear paths, with each linear path resolved into two components i.e., horizontal, and vertical path considering an angle of movement for oblique paths. Coming on to single portions of a path, the method achieves the resolution into two components by considering a right-angle triangle, keeping the portion as the hypothenuse of that triangle. The next step is to divide that path into 'n' segments of equal distances and draw a vertical line from each segment's end to the adjacent side of the triangle with respect to the direction angle β to resolve each segment into two components, one with respect to the opposite side and another to the adjacent side. Finally, using two linear movements, a staircase-like pattern can be achieved from the starting point to the destination, thereby simulating an oblique movement ensuring the desired angle of direction is met. By increasing the number of segments for the triangle decomposition, the path looks more like a straight line rather than a staircase, thereby achieving the desired linear movement.

Let us now analyze a single triangle obtained after segmentation in order to find the expressions for coordinates calculation in reaching the destination point. Let us consider the decomposed right triangle AC_1A_1

from Figure 6, Assuming a 2D plan with two axes, X and Y, with point 'A' as the origin for a reference frame (A, X, J). Moving a robot across the segment AA_1 is executed by moving first from 'A' to 'C_1', then from 'C_1' 'A_1'. The expression required to compute the coordinates to achieve the movement considering the two resolved components is given by,

$$\begin{cases} AC_1 = {AA_1}/{\cos\beta} \\ C_1A_1 = {AA_1}/{\sin\beta} \end{cases} \tag{9}$$

Since $AA_1 = a$, $AC_1 = c$ and $C_1A_1 = b$, in (9) becomes: $\begin{cases} c = {a}/{\cos\beta} \\ b = {a}/{\sin\beta} \end{cases}$

Finally, if 'd' is considered to be the length of the path such that $AB = d$, the calculation of the path length considering the coordinates required to reach the destination with respect to the angle of direction is,

$$d = n\sqrt{a^2 + b^2} \tag{10}$$

In conclusion, with 'n' iterations using in (10), we can achieve omnidirectional control at any angle of direction, provided the angle of direction 'β', speed of movement 's', total number of segments or steps 'n', and distance to be moved 'd'. Let 'A' represent the Akansie's parameters that encompasses the four parameters used to achieve an omnidirectional control of mecanum wheel robot using the triangle decomposition method.

2.3. Omnidirectional linear mobility control algorithms using triangle decomposition

Figure 7 depicts the flowchart proposed for any angle omnidirectional control from the proposed framework. First, Akansie's parameters should be provided. Based on the input given, the algorithm understands whether it must compute the remaining parameters from the given coordinates.
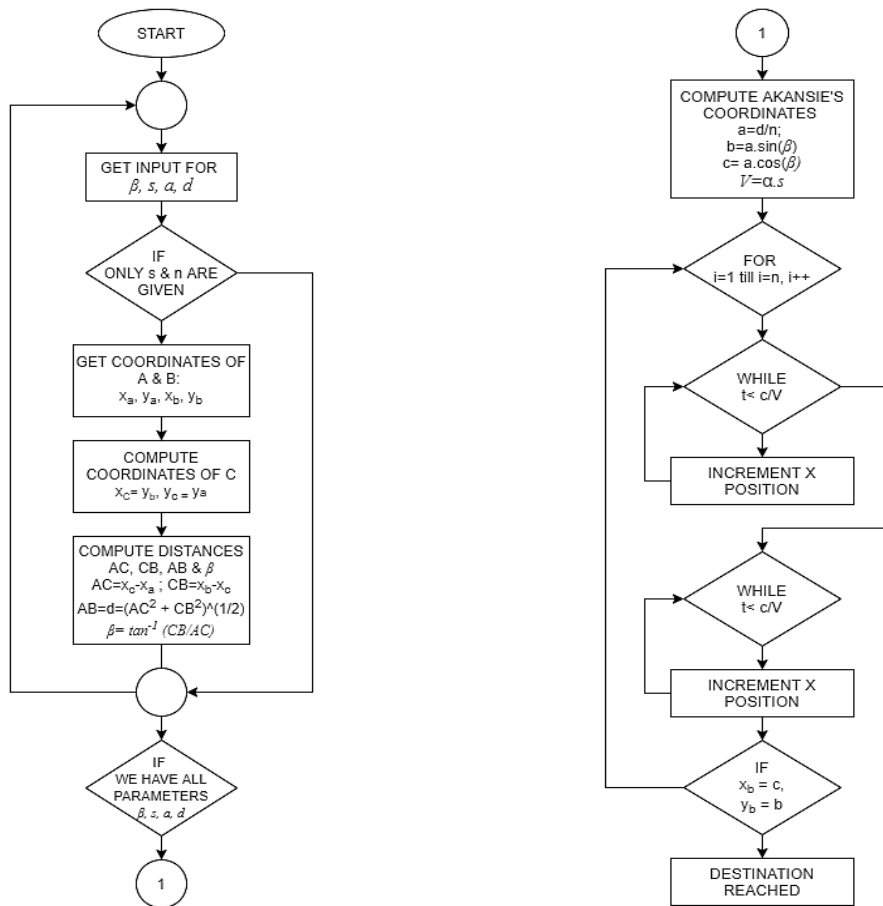

Figure 7. Flowchart for any angle linear control of mecanum wheels based systems

Once that step is achieved, the coordinate for the third point, 'C,' followed by the distances of portions AC, CB, and AB, segment length 'a', and the angle 'β' are calculated sequentially. Finally, Akansie's coordinates are computed from the computed parameters, and the iteration process is started. The Akansie's coordinates are calculated for each iteration, and the vehicle moves as per the calculated coordinate, ensuring they match the displacement made by the vehicle. Once the final destination coordinates match the current coordinates, the process stops. The corresponding algorithm is depicted in Figure 8.

```
Algorithm 1
Require: β, s, n, d
Ensure: n ≠ 0
  if β = 0  and  n = 0  and  s ≠ 0  and  d ≠ 0 then
      xa, xb, ya, yb ← i                                    ▷ i is get
      xc = xb
      yc = ya
      AC = xc − xa
      AB = AC² + CB²
      d = √AB
      β = tan⁻¹(CB/AC)
      if β ≠ 0  and  n ≠ 0  and  s ≠ 0  and  d ≠ 0 then
          print   (Data recorded successfully)
      else
          goto : first   statement
      end if
      a = d/n
      b = a ∗ sin(β)
      c = a ∗ cos(β)
      V = α ∗ s
      i = 1
      loop   i ≤ n
          while t < (c/V) do
              Xposition ← Xposition + 1
          end while
          while t < (b/V) do
              Yposition ← Yposition + 1
          end while
          i ← i + 1
      end loop
      if xb=c and yb=b then
          print(Path Executed)
      end if
  end if
```

Figure 8. Algorithm for any angle linear control of mecanum wheels based systems

3. RESULTS AND DISCUSSIONS
3.1. Simulation scenario

Webots simulation software along with Kuka youbot model are used to simulate a scenario where a four mecanum wheel vehicle is made to follow a linear oblique path at four different angles as shown in Figure 9. Assuming the starting and destination points are given in terms of coordinates as shown in Table 3, the vehicle is made to reach the destination point each time, based on the destination coordinate, by applying the proposed algorithm. The software provides libraries for the implementation of a PID controller required to drive each wheel independently in either direction at desired speed. Once the input is given, the triangle decomposition algorithm drive the wheels through the PID controller till the destination is reached. The vehicle model used is Kuka youbot, readily available in the simulation software. The simulation is done assuming five segments for the segment decomposition, keeping the speed of the wheels to be 25 m/s.
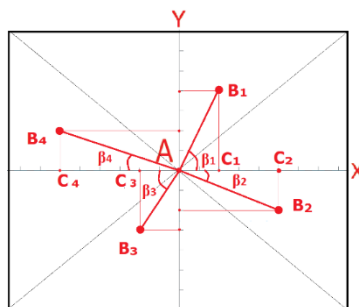

Figure 9. Simulation scenario for testing the proposed method

Table 3. Destination coordinates and theoretical angle value for each scenario

Destination coordinates	B_1	B_2	B_3	B_4
x (cm)	20	50	20	60
y (cm)	40	20	30	20
Angle β {tan-1 (y/x)} (deg)	63.43	21.80	56.30	28.43

3.2. Results and discussions

Figure 10 shows the images resulting from the actual simulation for the second case where the vehicle had to reach destination point B2. Figure 11 shows the theoretical and practical direction angles achieved with the proposed triangular decomposition approach across the linear oblique path depicted by the destination coordinates, compared with angles achieved with a basic PID control method such as traditional robust sliding mode control methods.
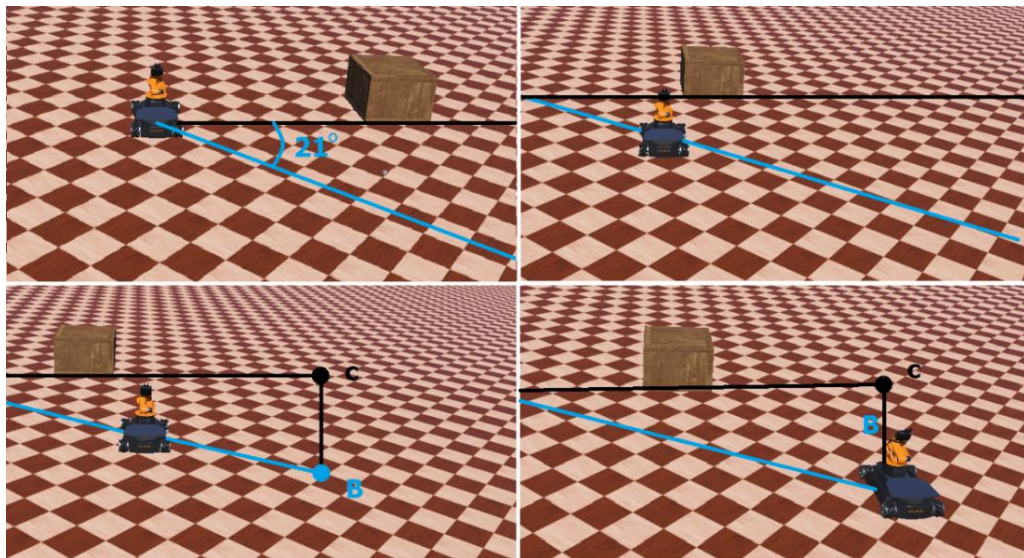
Figure 10. Snapshots of the simulation for any angle linear control of mecanum wheels based systems
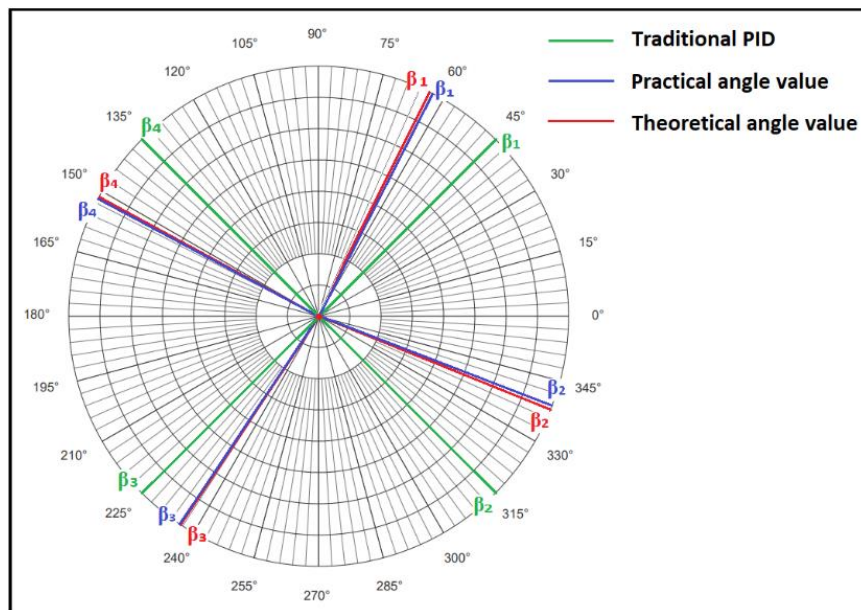
Figure 11. Representation of theoretical and achieved angle values for traditional and triangular decomposition method in a polar coordinate system

The results show the accuracy of the proposed method as there is a minimal error between the theoretical and practical values compared with values obtained with a basic PID control method. Figure 11 shows the theoretical value in each case (red plot), the practical values obtained (blue plot) as per the proposed method, and the values obtained with a traditional PID control method (green plot) in a polar grid. The traditional PID control method at constant speed for each wheel can still follow a linear path with minimal error, but the vehicle must rotate and align with the traveling path. Instead, our method uses the same basic PID at constant speed without a need of rotation and alignment with the linear path to be followed, as the vehicle can still be moved across the linear oblique path, keeping the desired angle for the whole duration of the movement till it reaches the destination. This ability helps the vehicle to move in confined spaces with less maneuver, which is an advantage, considering specific scenarios. It's important to note that the same method can be used along with a path planning algorithm to move mecanum wheel robots however we want, irrespective of the shape of the path, since the proposed approach is not restricted to linear paths only. A qualitative comparison with different approaches is presented in Table 4.

Table 4. A qualitative comparison between triangle decomposition and different approaches

	Triangle decomposition contol	Fuzzy PID control	Robust sliding mode
Model dependency	less dependent on accurate system models	Inaccurate or incomplete models may lead to suboptimal performance	Deviations from the model might affect the performance and robustness of the control
Total number of parameters	Uses only four parameters	Fuzzy PID controllers typically have more parameters to tune	larger number of tuning parameters to ensure robustness
Parameter tuning	well-established and straightforward tuning procedure	Tuning fuzzy PID parameters can be challenging	Tuning additional parameters in robust sliding mode control can be challenging
Execution time	computationally simple with have low execution time	introduce additional computational overhead	introduce additional computational overhead due to the complex control laws
Computational complexity	low computational complexity, suitable for embedded applications	computationally intensive with complex fuzzy rule bases	can be computationally intensive
Environment adaptability	can adapt reasonably well to different environments without significant parameter adjustments	lack adaptability to changing environments	designed to handle uncertainties, disturbances, and external perturbations
Energy consumption	low energy consumption due to their simplicity and lower computational demands	The additional computational complexity result in higher energy consumption	The additional processing requirements result in higher energy consumption

4. CONCLUSION

The paper presented a method to achieve linear omnidirectionality motion of mecanum wheel robots at any angle. Conventional PID based control methods fail to provide a linear omnidirectionality, maintaining a constant speed for each wheel. The mechanical design of mecanm wheels allows a basic control of robots by driving wheels independently either forward or backward. Such control allows a 45 degrees oblique linear motion, which is not optimum to accomplich shortest path in some scenarios. A case study has been carried out to investigate the direction's angle effects on the length of linear paths. As a result, it was found that low direction angles, yet not achievable with basic PID control, lead to shortest paths. To allow basic PID controllers to achieve variable angles motion, an algorithm that uses the inherent capabilities of mecanum wheels to move forward and sideways was proposed to accomplish any linear motion at any angle through the triangle decomposition method algorithm. This method is driven by a framework using Akansie's parameters i.e., the angle of direction 'β', speed of movement 's', total number of segments or steps 'n', and distance to be moved 'd'. The method was tested on Webots simulation software with simple scenarios, where the 'Kuka youbot' mecanum wheel robot model had to follow a linear path at different angles without self rotation. The results obtained from the simulation were a validation of our expectation, with less angle deviation from the theoretical calculations. In comparison to a basic PID motion controller which can provide an oblique motion of 45 degrees only at constant speed for each wheel, our method provides an omnidirectional motion that is not restricted to 45 degrees. Moreover, the proposed algorithm is light weight thereby uses less computational power and resources to control the robot. The triangle decomposition algorithm can be employed in any path planning method to accomplish motions over complex paths' shapes by approximating them as a succession of linear motions, leading to short distance and time to reach any destination.

ACKNOWLEDGEMENTS

REFERENCES

[1] M. B. Alatise and G. P. Hancke, "A Review on Challenges of Autonomous Mobile Robot and Sensor Fusion Methods," *IEEE Access*, vol. 8, pp. 39830–39846, 2020, doi: 10.1109/ACCESS.2020.2975643.

[2] F. Rubio, F. Valero, and C. Llopis-Albert, "A review of mobile robots: Concepts, methods, theoretical framework, and applications," *International Journal of Advanced Robotic Systems*, vol. 16, no. 2, 2019, doi: 10.1177/1729881419839596.

[3] L. Tagliavini, G. Colucci, A. Botta, P. Cavallone, L. Baglieri, and G. Quaglia, "Wheeled Mobile Robots: State of the Art Overview and Kinematic Comparison Among Three Omnidirectional Locomotion Strategies," *Journal of Intelligent and Robotic Systems: Theory and Applications*, vol. 106, no. 3, pp. 1–18, 2022, doi: 10.1007/s10846-022-01745-7.

[4] H. Taheri and C. X. Zhao, "Omnidirectional mobile robots, mechanisms and navigation approaches," *Mechanism and Machine Theory*, vol. 153, 2020, doi: 10.1016/j.mechmachtheory.2020.103958.

[5] R. Wen and M. Tong, "Mecanum wheels with Astar algorithm and fuzzy PID algorithm based on genetic algorithm," in *2017 International Conference on Robotics and Automation Sciences, ICRAS 2017*, 2017, pp. 114–118, doi: 10.1109/ICRAS.2017.8071927.

[6] E. Malayjerdi, H. Kalani, and M. Malayjerdi, "Self-Tuning Fuzzy PID Control of a Four-Mecanum Wheel Omni-directional Mobile Platform," in *26th Iranian Conference on Electrical Engineering, ICEE 2018*, 2018, pp. 816–820, doi: 10.1109/ICEE.2018.8472568.

[7] Y. Chen, "Design of Begian–Melek–Mendel structure-based interval type-2 fuzzy logic systems optimized with backpropagation algorithms," *Transactions of the Institute of Measurement and Control*, vol. 44, no. 15, pp. 3057–3067, 2022, doi: 10.1177/01423312221099700.

[8] G. Cao, X. Zhao, C. Ye, S. Yu, B. Li, and C. Jiang, "Fuzzy adaptive PID control method for multi-mecanum-wheeled mobile robot," *Journal of Mechanical Science and Technology*, vol. 36, no. 4, pp. 2019–2029, 2022, doi: 10.1007/s12206-022-0337-x.

[9] V. Alakshendra and S. S. Chiddarwar, "A robust adaptive control of mecanum wheel mobile robot: Simulation and experimental validation," in *IEEE International Conference on Intelligent Robots and Systems*, 2016, pp. 5606–5611, doi: 10.1109/IROS.2016.7759824.

[10] V. Alakshendra and S. S. Chiddarwar, "Adaptive robust control of Mecanum-wheeled mobile robot with uncertainties," *Nonlinear Dynamics*, vol. 87, no. 4, pp. 2147–2169, 2017, doi: 10.1007/s11071-016-3179-1.

[11] M. Saad, U. Tople, A. Khandare, and Z. Ul Islam, "Robust Sliding Mode Controller (RSMC) for an Omniwheeled Mobile Robot with Uncertainties and External Perturbations," in *Lecture Notes in Mechanical Engineering*, 2021, pp. 465–472, doi: 10.1007/978-981-15-3639-7_55.

[12] Z. Hendzel and M. Kołodziej, "Robust Tracking Control of Omni-Mecanum Wheeled Robot," in *Advances in Intelligent Systems and Computing*, 2021, vol. 1390, pp. 219–229, doi: 10.1007/978-3-030-74893-7_21.

[13] C. C. Tsai and H. L. Wu, "Nonsingular terminal sliding control using fuzzy wavelet networks for Mecanum wheeled omni-directional vehicles," 2010, doi: 10.1109/FUZZY.2010.5584223.

[14] Z. Sun, S. Hu, N. Li, and D. He, "Trajectory-following control of mecanum-wheeled AGV using fuzzy nonsingular terminal sliding mode," in *2020 4th CAA International Conference on Vehicular Control and Intelligence, CVCI 2020*, 2020, pp. 342–347, doi: 10.1109/CVCI51460.2020.9338561.

[15] Z. Sun, H. Xie, J. Zheng, Z. Man, and D. He, "Path-following control of Mecanum-wheels omnidirectional mobile robots using nonsingular terminal sliding mode," *Mechanical Systems and Signal Processing*, vol. 147, 2021, doi: 10.1016/j.ymssp.2020.107128.

[16] X. Wu and Y. Huang, "Adaptive fractional-order non-singular terminal sliding mode control based on fuzzy wavelet neural networks for omnidirectional mobile robot manipulator," *ISA Transactions*, vol. 121, pp. 258–267, 2022, doi: 10.1016/j.isatra.2021.03.035.

[17] G. Wang, C. Zhou, Y. Yu, and X. Liu, "Adaptive sliding mode trajectory tracking control for WMR considering skidding and slipping via extended state observer," *Energies*, vol. 12, no. 17, 2019, doi: 10.3390/en12173305.

[18] C. Ren, X. Li, X. Yang, and S. Ma, "Extended state observer-based sliding mode control of an omnidirectional mobile robot with friction compensation," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 12, pp. 9480–9489, 2019, doi: 10.1109/TIE.2019.2892678.

[19] Z. Yuan, Y. Tian, Y. Yin, S. Wang, J. Liu, and L. Wu, "Trajectory tracking control of a four mecanum wheeled mobile platform: An extended state observer-based sliding mode approach," *IET Control Theory and Applications*, vol. 14, no. 3, pp. 415–426, 2020, doi: 10.1049/iet-cta.2018.6127.

[20] T. Zhao, X. Zou, and S. Dian, "Fixed-time observer-based adaptive fuzzy tracking control for Mecanum-wheel mobile robots with guaranteed transient performance," *Nonlinear Dynamics*, vol. 107, no. 1, pp. 921–937, 2022, doi: 10.1007/s11071-021-06985-0.

[21] C. Kim, J. Suh, and J. H. Han, "Development of a hybrid path planning algorithm and a bio-inspired control for an omni-wheel mobile robot," *Sensors (Switzerland)*, vol. 20, no. 15, pp. 1–22, 2020, doi: 10.3390/s20154258.

[22] L. Zhang, J. Kim, and J. Sun, "Energy modeling and experimental validation of four-wheel mecanum mobile robots for energy-optimal motion control," *Symmetry*, vol. 11, no. 11, 2019, doi: 10.3390/sym11111372.

[23] K. L. T. Trinh and H. T. Nguyen, "Path tracking control for car-like robots by pid controller with time-varying parameters," *VNUHCM Journal of Engineering and Technology*, vol. 5, no. 3, pp. 1642–1650, 2022, doi: 10.32508/stdjet.v5i3.961.

[24] N. H. Thai, T. T. K. Ly, and L. Q. Dzung, "Trajectory tracking control for mecanum wheel mobile robot by time-varying parameter PID controller," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 4, pp. 1902–1910, 2022, doi: 10.11591/eei.v11i4.3712.

[25] H. A. Ibukun, O. W. Ofuzim, and O. J. Aanuoluwa, "Kinematic Analysis of Omnidirectional Mecanum Wheeled Robot," *International Journal of Engineering and Applied Physics*, vol. 3, no. 1, pp. 634–644, 2023.

[26] Z. Sun, S. Hu, D. He, W. Zhu, H. Xie, and J. Zheng, "Trajectory-tracking control of Mecanum-wheeled omnidirectional mobile robots using adaptive integral terminal sliding mode," *Computers and Electrical Engineering*, vol. 96, 2021, doi: 10.1016/j.compeleceng.2021.107500.

[27] M. Alfiyan and R. D. Puriyanto, "Mecanum 4 Omni Wheel Directional Robot Design System Using PID Method," *Journal of Fuzzy Systems and Control*, vol. 1, no. 1, pp. 6–13, 2023, doi: 10.59247/jfsc.v1i1.27.

BIOGRAPHIES OF AUTHORS

Kouame Yann Olivier Akansie obtained the B.E. (2018) degree in Electrical and Electronics Engineering from East Point College of Engineering and Technology, India. He further obtained the M.Tech. (2020) degree in Digital Communications and Networking from REVA University, India. He is currently a Ph.D. scholar at REVA University, India, within the school of Electronics and Communication Engineering. His thesis is about the design and development of autonomous hybrid wheel-legged robot for terrestrial navigation. He can be contacted at email: akansiekouame@gmail.com.

Rajashekhar C. Biradar completed his B.E. (Electronics and Communication Engineering) in 1990, M.E. (Digital Electronics) in 1997 from Karnataka University Dharwad, India, and Ph.D. in 2011 under Visvesvaraya Technological University (VTU), Belgaum, India. He is currently working as professor and director of school of ECE, REVA university, Bangalore, India. To his credit, he has many national/international journal and conference publications. His research areas include multicast routing in mobile ad hoc networks, wireless Internet, group communication in MANETs, agent technology. He is a member of IEEE (USA), member of IETE (MIETE, India), member of ISTE (MISTE, India), member of IE (MIE, India) and member of ACM. He can be contacted at email: rcbiradar@reva.edu.in.

Rajendra Karthik received M. Tech degree from the Visvesvaraya Technological University, India and Ph.D. degree from VIT University, India. His Ph.D. thesis research work was carried out at one of the labs of Center for Nano electronics, Indian Institute of Technology-Bombay, India. He was the Professor at Department of Electronics and Communication and Dean R&D at MLR Institute of Technology, Hyderabad. Earlier, he was working as a faculty member at VIT University, Vellore. He received best researcher award from VIT University for his contribution to Nano dielectrics in 2013 and 2014. Also, he received Best researcher award in 2017, 2018 and 2019 at MLR Institute of Technology. His current area of research includes Fabrication and Modeling of Nano electronic or Optoelectronic material-based devices, Microwave Antennas, Medical Image Processing, and Transformation in Engineering Education. At present, he is guiding 1 PhD research scholar. He has completed 5 Sponsored research projects worth Rs. 44 Lakhs. At present he has 2 ongoing research projects funded by DST, Govt. of India worth Rs. 3 Crores. He has published more than 110 research papers in reputed journals and conferences, 4 Book Chapters and filed 3 Patents. He is one of the co-designers for developing a Nano-size high performance capacitor in 2013 and 2020. He can be contacted at email: karthik.rajendra@reva.edu.in.

Geetha D. Devanagavi received her Ph.D., M.Tech., and B.E. degrees in 2014, 2005, and 1993, respectively. She is currently working as associate Professor in the school of Computing and Information Technology at Reva University, Bangalore, Inida. She has 24 years of teaching experience. Her research interests include wireless sensor networks, network security, and computer networks. She can be contacted at email: dgeetha@reva.edu.in.