Path Generated and Optimized of Mobile Robot using Simulation

Teuku Firsa, Muhammad Tadjuddin, Iskandar, Syahriza
Department of Mechanical Engineering, Syiah Kuala University
Jl. Tgk. Syech Abdurrauf No. 7 Darussalam – Banda Aceh 23111, INDONESIA
Phone/Fax.: +62-651- 7428069, e-mail: teuku.firsa@unsyiah.ac.id

Abstract

There are several ways to guide a wheeled mobile robot on the floor. One way is to mount the mobile robot on a rail, so that the mobile robot can follow the rail and its movement is controlled by a controller. Another method is by using colour mark sensors. The sensors are used to detect the line on the floor thus guiding the robot along it. An autonomous mobile robot uses images from a camera to recognize the obstacle ahead and compute the best path to follow and avoid obstacles. But this type of system is expensive and image processing takes time.

The best method is prepared, in which a predetermined path is derived from simulation. The robot movement is programmed on a simulated bordered floor, and its path is optimized before the program is downloaded to the mobile robot. By using this approach, the time taken for image processing will be reduced.

This paper describes the simulation of a two wheel mobile robot. The kinematics and dynamic model of the robot is simulated as well. The simulation program was developed in MATLAB.

Kata kunci : Mobile robot, Two wheel, Mobile robot simulation

BACKGROUND

An industrial robot can only manipulate objects that it can reach. Most industrial robots are fixed in their place, their workplace is limited by the maximum extension of their linkages. Components are brought to the robot and, taken away by conveyors and other mechanical feed device. To overcome these problems caused by the limited reach of a robot arm, a wheeled robot is preferred. A wheeled mobile robot has greater mobility and flexibility in performing a task.

Some variety of types of wheeled mobile robot, there are one-wheeled vehicles, two-wheeled vehicles (bicycles), Front steering wheels for steering, Rear driving wheels, Front wheels for both driving and steering, Steering mechanism of a wheel loader used in construction and mining industry and two wheel mobile robot using a differential speed steering.

The turning of the differential steering robot can be controlled by providing a different angular velocity to each wheel (changing the speed of each motor). The term differential steering comes from the fact that the turning radius of the robot is a function of the ratios (or differences) of the wheel or tank-tread speeds on both sides of the robot.

In this paper a method is prepared in which a predetermined path is derived from simulation. The robot movement is programmed, simulated and path will be optimized with manipulate the velocity and acceleration before the program is downloaded to the real mobile robot. By using this approach, the time taken for image processing will be reduced.

The kinematics and dynamic model of the robot is simulated. The simulation program was developed in MATLAB. MATLAB is a numerical computing environment with programming language that is used to program the path of the mobile robot. The resulting trajectory is then downloaded to the actual mobile robot. The advantage of this method is that we can verify the motion of mobile robot before actually

This is due to friction between wheel and floor as well as slip and skid of the robot movement. However, this was not taken into consideration since it wasn’t made a deeper analysis.

KINEMATICS AND DYNAMICS OF THE MOBILE ROBOT

Locomotion is the process of causing an autonomous robot or vehicle to move. In order to produce motion, forces must be applied to the vehicle. The study of motion in which these forces are modeled is known as dynamics, whereas kinematics is the study of the mathematics of motion without considering the forces that affect the motion. There are three parameters involved in the kinematics of mobile robot when the robot is moving on a horizontal plane. The Cartesian coordinates give the instantaneous position and the orientation of the mobile robot $^{(x,y)}$as a reference point of the robot fixed on the floor, the heading angle of the robot (θ), and the absolute position is denoted by the function(x, y, θ).

Kinematics

This kinematics of mobile robot discusses the linear velocity, acceleration and the differential steering system included in rounding a corner, the derivation and straight line movement. The dynamics of a mobile robot is also included.

a. Control Input for Mobile Robot

Figure 1 illustrates the robot movement. Each wheel has the same radius of Rw, and the wheels have a distance of l. The control inputs of the robot are: Angular velocity of motor on left wheel: ωl [rad/s] Angular velocity of motor on right wheel: ωr
[rad/s]
ω = Angular velocity of the center of the mobile robot
Rc = the radius of the instantaneous curvature

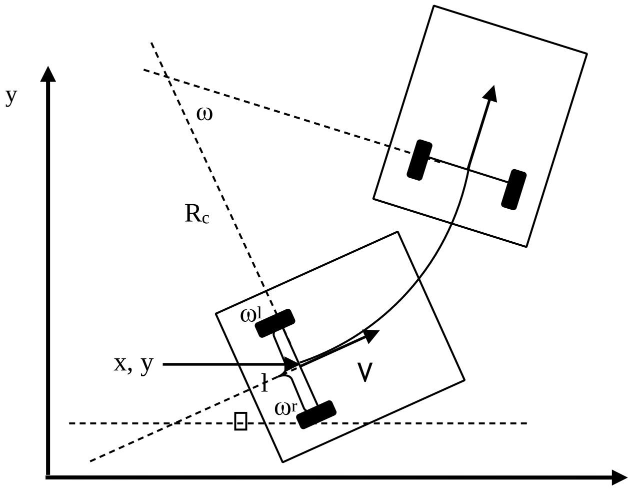


Figure 1 Differential of a steering robot

Whereas, Figure 2 shows the robot is moving pivot which the speed of two wheels are at the same speed but different direction.

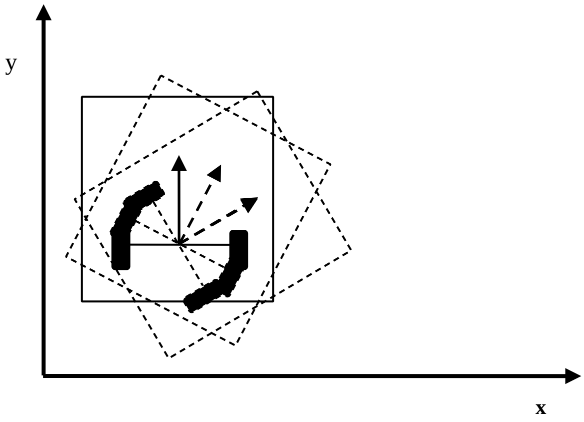


Figure 2 Robot moving pivot

b. Instantaneous Curvature and Linear Motion

An instantaneous curvature occurs when ωr > ωl. Figure 1 shows the instantaneous curvature of such a robot.

The angular velocities of the center of the robot is
:

$$\omega = \frac{R_w}{l\eta}(\omega_r - \omega_l) \qquad [1]$$

The linear velocity of mobile robot is:

$$v = \frac{R_w}{2\eta}(\omega_l + \omega_r) \qquad [2]$$

So that the radius of the instantaneous curvature is given by

$$R_c = \frac{v}{\omega} \qquad [3]$$

From equation [2], the position of the robot is given by

$$\dot{x} = \frac{R_w}{2\eta}(\omega_l + \omega_r)\cos\theta \qquad [4]$$

$$\dot{y} = \frac{R_w}{2\eta}(\omega_l + \omega_r)\sin\theta \qquad [5]$$

Dynamics

The dynamics of a robot is related to acceleration of the loads, masses and inertias. In dynamic, to accelerate a mass, force or torque is required to act on it.

To accelerate a robot, it is necessary to have actuators that are capable of exerting enough vastly forces and torques to move it at the desired acceleration and velocity. Otherwise, the robot may not be moving fast enough, and thus it will lose its positional accuracy. To calculate how strong each actuator must be, it is necessary to determine the dynamic relationships that govern the motions of the robot. These equations are the torque, inertia and angular acceleration relationships. Based on these equations and considering the external loads on the robot, it is possible to calculate the largest loads to which the actuators may be subjected and thereby design the actuators to be able to deliver the necessary torques, [7].

In general, the dynamic equations may be used to find the equations of motion of mechanisms by knowing the forces and torques. The desired accelerations of the robot can be found. These equations are also used to see the effects of different inertial loads on the robot depending on the desired accelerations. The dynamic equations allow the designer to investigate the relationship between different elements of the robot and design its components appropriately.

a. Lagrangian Mechanics Lagrangian mechanics is based on the differentiation of the energy terms with respect to the system's variables and time. The lagrangian method is relatively simple to use. The Lagrangian mechanics is based on the following two generalized equations, one for linear motions, and one for rotational motions [7].

L = K – P [6]
Where:
L = the Lagrangian
K = the kinetic energy
P = the potential energy

The relation between the angular acceleration of each wheel and the torque applied is obtained from equation [1], using the following relation:

$$T = \frac{\partial}{\partial t}\left(\frac{\partial L}{\partial \dot{\theta}}\right) - \frac{\partial L}{\partial \theta} \qquad [7]$$

Where:
T = the summation of all torques in a rotational motion,
θ = system variables

b. Dynamic of Mobile Robot

Figure 3.6 illustrates the dynamic model of wheel robot. Which the moment inertia (Iw)is the rotational analog of mass, it is the inertia of a rigid rotating body with respect to its rotation.

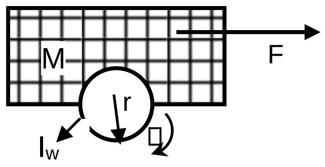


Figure 3 Dynamics model

The kinetic energy of a mass rigid body is given

K = ½ Mv2 [8]

And the potential energy by

P = mgh [9]

I = Moment of Inertia of a wheel of mass and radius

I = ½ m r2 [10]

The dynamic equation using Lagrangian formulation as following derived,

L = K - P

Where,

$$K = \frac{1}{2}MV^2 + 2(\frac{1}{2}I\dot{\omega}^2 \quad)$$

$$L = \frac{1}{2}MV^2 + \frac{1}{2}mr^2\dot{\omega}^2 \qquad [11]$$

Replacing value at Equation [6] on equation [11], the Lagrangian expression is obtained:

$$T = \frac{1}{8}M\ r^2\left(\dot{\omega}^2{}_l + \dot{\omega}^2{}_r\right) + \frac{1}{2}mr^2\dot{\omega}^2 \quad [12]$$

Where,

T = Torque from 0 to max velocity (NM) M = mass of robot (N) m = mass of wheel (N)
r = radius of wheel (m)

P= FV [13]

F = Cr m g
Where,
P = Required Power
F = Required Force, Cr = Coefficient of rubber wheel

c. Angular Kinematics The angular motion of an object over a period of time can also be fully describe by using three mechanical variables, namely angular displacement, angular velocity and angular acceleration.

Angular acceleration can be quantified by considering that the change is angular velocity in a given period of time,

$$\alpha = \frac{\omega_f - \omega_i}{t} \qquad [14]$$

Where,
ω_f = the initial angular velocity of the object
ω_i = the final angular velocity of the object
α = constant angular acceleration of the object
t = time taken for the change in orientation to occur

These equations are valid for all angular motions in which the angular acceleration is constant. It predicts that the angular velocity produced by a constant angular acceleration will be directly proportional to the time for which the angular acceleration is applied [13].

Design and Fabrication of the Chassis

The mobile robot's characteristics, number of wheels, materials, structure etc., are first designed by using Pro-Engineer software (wildfire). It is one of the world's most widely used computer Aided Design (CAD) software. It is a powerful part modeling, easy 2D and 3D geometry creation environment. The mobile robot chassis in this thesis is designed based on the L-shape plate that is available and easily found in the market. Figure 4.1 and 4.2 show the chassis of a mobile robot being assembled. The height of the base plate form from the floor is 460 mm, the length is 404 mm and the width is 395 mm.

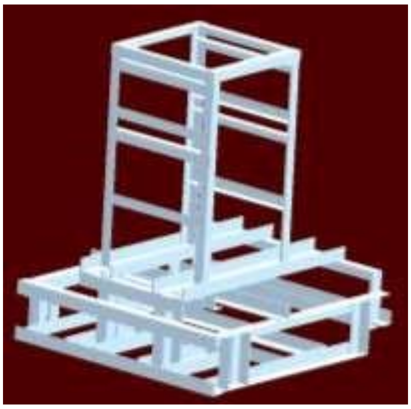


Figure 4 The whole Platform without components

a. Mechanical Design

The mechanical and electrical design of robots is a complex task, involving a variety of differing mechanisms, and a host of conflicting constraints. However, good software cannot overcome poor mechanical design. Thus, potentially weak points in mechanical design must be eliminated. Such weak points include structural deformation, gear backlash, and poor bearing clearance, friction, thermal effects, and poor connection of transducer. The chassis of the robot is manually made; most of the structure is simplified due to the limitation of technology and skill available in the laboratory. Mounting the motor on the gearbox and wheel requires high accuracy of alignment. Misalignment need extra torque to start up and turn the wheel and causes new adjustment of the configuration of the motor. The ratio of gear box 1:15 is chosen because of it could increase torque of the robot, whereas the rubber wheel with diameter 30 mm is selected using the existing component in market and it is assumed type and size are suitable. Figure 5 is the mechanism designs of motor gear box and wheel assembled into the base chassis. This design has been used as a reference to build the actual mobile robot frame.

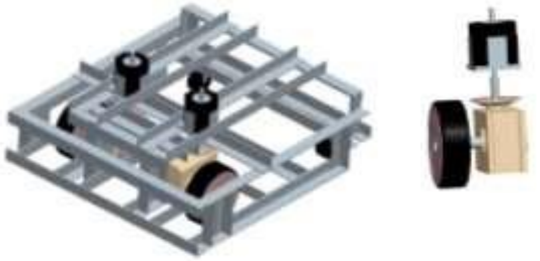


Figure 5 Base chassis assembled with motor, gear box and wheel

The chassis is made from L-shaped aluminum for lightness and strength. Each aluminum bar is riveted together for rigidness.

The design of mobile robot is shown below.

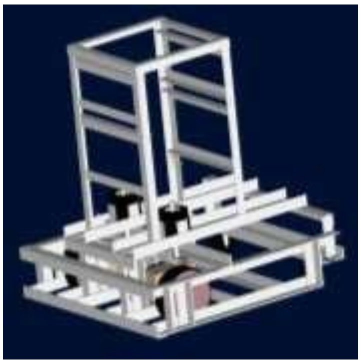


Figure 6 CAD model of the mobile robot

The robot is an integrated mobile robot system with the proximity sensor, and the body of the robot is supported by four wheels of which the two front rubber wheels is driven by a stepper motor of 39 W.

Mobile Robot Modeling

Figure 7 below shows the mobile robot model with the basic parameters used in the system. The body of robot is considered to be square with mass M, with two wheels of radius R_w and a mass m of each. The right wheel rotates at an angular speed of $\omega_l = \dot{\theta}_l$, and the left at $\omega_r = \dot{\theta}_r$. Each wheel is connected to an independent DC motor by using a gear system of the ratio 15 : 1.

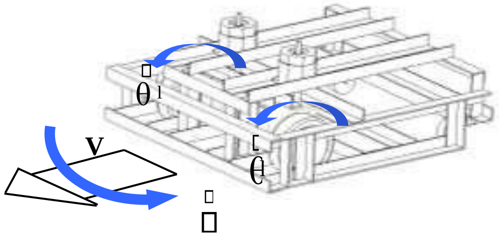


Figure 7 The robot model (showing the main dynamic parameter)

a. Robot Kinematics

The kinematics for the robot relates the state or posture of the machine, with the angular displacement of each wheel. Figure 8 shows the kinematic model of a differential steering robot when it is turning a curvature. In this case, the angular velocity of the right wheel is faster than the left wheel, so that the mobile robot will turn to the left. The mobile robot has a gear box to reduce the speed and increase the torque as shown in Figure 9 In this paper, the ratio of the gears is as follows:

$$\eta = \frac{R_w}{R_m} >> 1$$

Where :

$R_w = 15$, $R_m = 1$

So, $\eta = \frac{15}{1}$

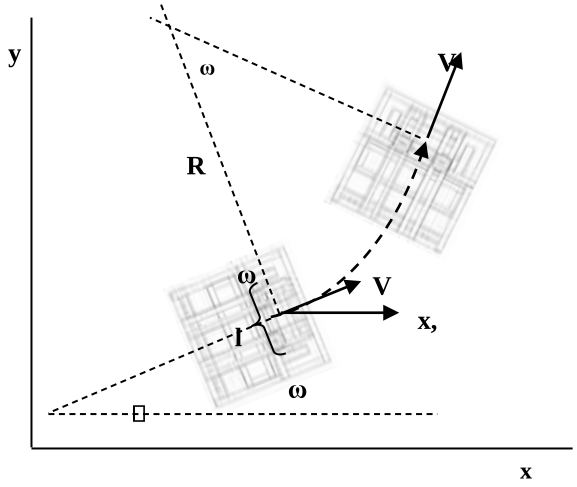


Figure 8 The kinematic model of differential steering robot

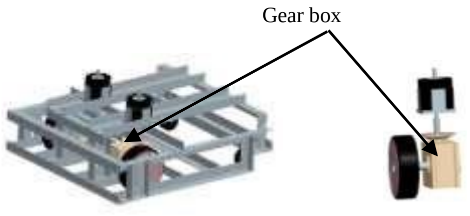


Figure 9 Gear box

The kinematics equations of [15] and [16] show the relation between the angular speed of the wheel and rotational and tangential speed of the robot with the gear ratio is taken into account.

In this example, it is assumed that the angular velocity of the right wheel and the left wheel are different. The angular velocity of the center of the robot is given by:

$$\omega = \frac{R_w}{l\eta}(\omega_r - \omega_l) \qquad [15]$$

Where,

$R_w = 0.05m$

$l = 0.269m$

$\omega_r = 500rpm$ = 52.33 rad/sec (assumed)

$\omega_l = 200rpm$ = 20.933 rad/sec (assumed)

$$\omega = \frac{0.05}{0.269\times 15}(52.33-20.933)$$

$\omega = 0.377$ rad/sec

The linear velocity of robot is:

$$v = \frac{R_w}{2\eta}(\omega_l + \omega_r) \qquad [16]$$

$$v = \frac{0.05}{2\times 15}(20.933+52.33)$$

$v = 0.147$m/s

The velocity at x and y position is as follows:

$$\dot{x} = \frac{R_w}{2\eta}(\omega_l + \omega_r)\cos\theta$$

$$\dot{y} = \frac{R_w}{2\eta}(\omega_l + \omega_r)\sin\theta$$

Where,

θ is as input for initial orientation of the robot (see Figure 8)

b. Robot Dynamics

The dynamic equation of the robot is related to the torque which is applied to the wheels, considering the mass inertia of the different elements in the model. These equations can be deduced by using the Lagrangian formulation, which is based on the calculation of the energy of the system. The total energy of the robot can be calculated as the sum of the kinetic energy of the body and the kinetic energy of each wheel. The potential energy is not used as the robot is considered to move on a single level plane.

Using Equation [13], the torque required is given by

$$T = \frac{\partial}{\partial t}\left(\frac{\partial L}{\partial \dot{\theta}}\right) - \frac{\partial L}{\partial \theta}$$

$$T = \frac{1}{8}M\ r^2\left(\ddot{\theta}_l + \ddot{\theta}_r\right) + \frac{1}{2}mr^2\,\ddot{\theta}$$

Where,

$\ddot{\theta} = \alpha$ = acceleration of wheel

$$\alpha = \frac{\omega_f - \omega_i}{t}$$

Where,

$\ddot{\theta} = \alpha$ = acceleration of wheel

$$\alpha = \frac{\omega_f - \omega_i}{t}$$

Where,

ω_{f_l} = 62.8 rad/sec, ω_{f_r} = 62.8 rad/sec

t = 10 sec

$$\alpha_l = \frac{\omega_{fl} - \omega_{il}}{t}$$

α_l = 6.28 rad/sec^2

α_r = 6.28 rad/sec^2

$$T = \frac{1}{8}M\ r^2(6.28+6.28) + \frac{1}{2}mr^2 6.28$$

Where,

M = 15 kg = 147.15 N
m = 1 Kg = 9.81 N
r = 0.05 m

$$T = \frac{1}{8}147.15\times(0.05)^2(6.28+6.28) + \frac{1}{2}9.81\times(0.05)^2 6.28$$

So, for the robot to move to its speed, the maksimum required torque from 0 to maximum velocity required is:

T = 0.655 Nm

The maximum torque can be reached in the specifications of motor is 1 Nm (See appendix A, SIGMAX M21 SERIES row, type M21NRFA and graphic of torque of stepper motor).

F = C_r m g

Where,

Cr = 0.02
F = 0.02 x 15 x 9.81
F = 2.943

So,

$$v = \frac{0.05}{2\times 15}(62.8+62.8)$$

v □ 0.209m/s (maximum velocity)

P= 2.943 x 0.209

P = 0.616 Watt (power required)

Simulation Program

The actual environment of the mobile robot is assumed available. The simulation program is developed using MATLAB. The program that is written in MATLAB is called M-files. MATLAB Compiler takes M-file as the input and generates redistributable, standalone applications or software components.

The simulation program is used to manipulate the movement and to create the desired trajectory of a mobile robot. In addition, the simulation can also generate program of the real mobile robot. In this term, the dynamic of mobile robot is not simulated.

The development of this simulation is started by constructing a main and sub program. The main program consists of a number of modules that are used to calculate the angular velocity, linear velocity and control the movement of the robot. The sub program is the input of the dimension that is used to create a two wheeled mobile robot model. When a simulation is compiled, the main program will call for a sub-program that contains the physical data of the robot to show the movement of the machine. Figure 10 shows the flowchart of the program.

a. Flowchart

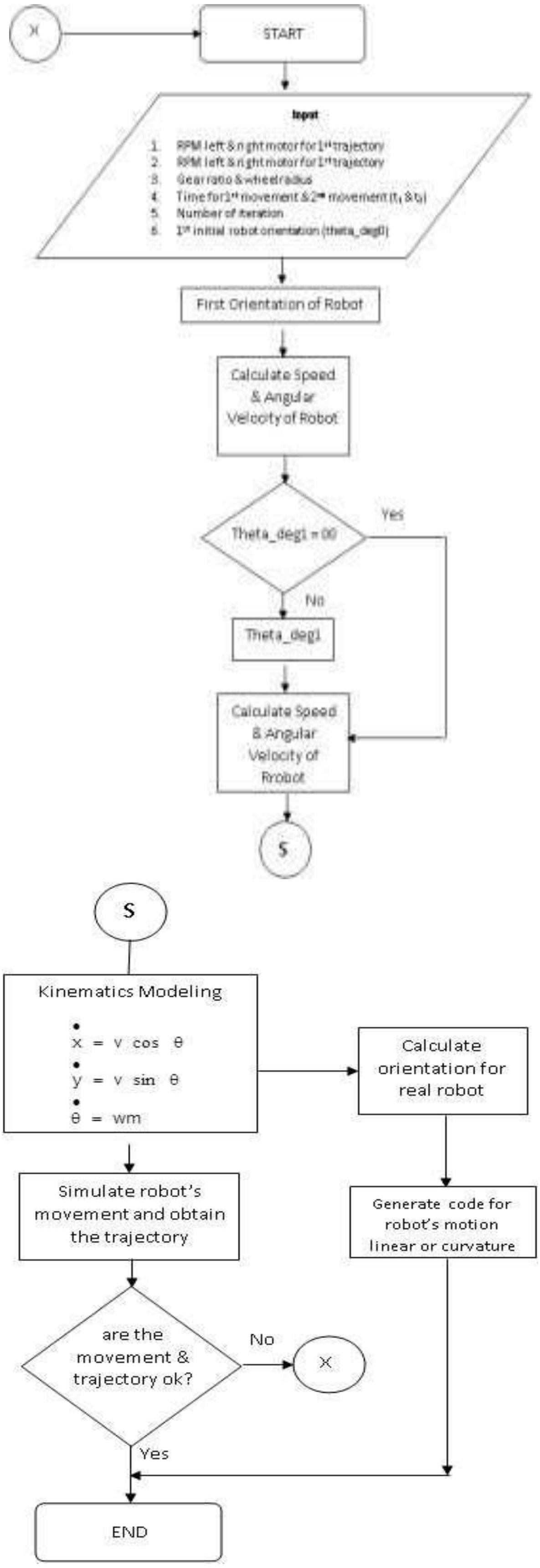

X
START
Input
1. RPM left & right motor for 1st trajectory
2. RPM left & right motor for 1st trajectory
3. Gear ratio & wheel radius
4. Time for 1st movement & 2nd movement (t1 & t2)
5. Number of iteration
6. 1st initial robot orientation (theta_deg0)
First Orientation of Robot
Calculate Speed & Angular Velocity of Robot
Theta_deg1 = 00
Yes
No
Theta_deg1
Calculate Speed & Angular Velocity of Rrobot
S
S
Kinematics Modeling
ẋ = v cos θ
ẏ = v sin θ
θ̇ = wm
Calculate orientation for real robot
Simulate robot's movement and obtain the trajectory
Generate code for robot's motion linear or curvature
are the movement & trajectory ok?
No
X
Yes
END

Figure 10 Simulation Flowchart

From the flow chart the physical parameters of the robot are entered initially followed by the motion values. The orientation of the robot is represented as theta_deg0 and theta_deg1. The robot will decide its first orientation according to theta_deg0 and then the program will calculate the speed and angular velocity. For the second orientation, the robot will decide the rotation from the input value of theta_deg1, whether the value is “00” or value as degree 0 deg to 360 deg. If the value is “00”, this means the second orientation follows the last position of the first move. Again, the program will calculate the speed and angular velocity. The kinematic model for the two wheeled mobile robot is used in this simulation. The velocity at x, y position and orientations is described for the robot’s move.

b. Algorithm

A linear dynamical system in a general form is:

$$\dot{x} = f(x,u)$$

Where x and u are state variables and control inputs, respectively. Based on the form above, the algorithm to simulate the motion of the robot is shown in Figure 6.2.

```
n: Iteration   x(n): State variable
vector u(n): Control input vector
func(x(n), u(n)) : Robot model
xdot(n): Rate of change of state
vector x0: Initial state variable
vector dt: Time step  n = 0;
x(n) = x0;
        xdot(n)=func(x(n),u(n));
        x(n+1)=x(n)+dt*xdot(n);
        n=n+1;
```

Figure 11 Algorithm of simulation

The time taken for the simulation is based on the number of iteration.

c. Input Parameters

The parameter input for the simulation must be in the ISO standard. ➢ Physical parameters of the robot

- Distance between the driven wheels (meter)
- Gear ratio
- Radius of wheel (meter)

An example input parameters are as follows: d (l) = 0.269; % Distance between the rear wheels [m] gratio = 15; % Gear ratio R_w = 0.05; % Radius of wheel [m]

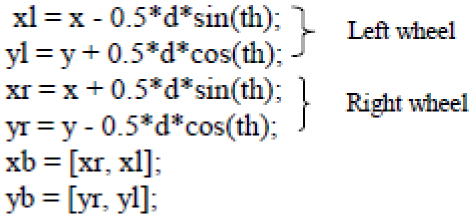
➢ Motion value

- Left motor RPM for first trajectory
- Right motor RPM for first trajectory
- Left motor RPM for second trajectory
- Right motor RPM for second trajectory - Initial orientation of the robot (degree)
- Simulation time (second) - Number of iterations (n)

Robot Modeling

In the sub program, to construct the kinematics model of the mobile robot, it is necessary to decide which type of the wheeled robot will be used, that is either three wheels steering type, the four wheels steering type or the two wheel differential steering type. The simulation program is as described for a two wheel differential steering. The two wheels are driven by separate motors. The path of the two wheeled mobile robot is described by the centre of the mobile robot. x = z(1); y = z(2); th = z(3);

Codes x,y is the position and th (theta) is the orientation. When the robot moves the value of x, y and th and or the position of the left and right wheels will increase or decrease as follows:



```
 xl = x - 0.5*d*sin(th);  }  Left wheel
yl = y + 0.5*d*cos(th);
xr = x + 0.5*d*sin(th);  }  Right wheel
yr = y - 0.5*d*cos(th);
xb = [xr, xl];
yb = [yr, yl];
```

xb and yb are combined from the two correlation (xl,yl) and (xr,yr) respectively and plot as a path of the wheel, see Figure (13).

```
xlwf = xl + R_w*cos(th);
ylwf = yl + R_w*sin(th);
xlwb = xl - R_w*cos(th);
ylwb = yl - R_w*sin(th);
xrwf = xr + R_w*cos(th);
yrwf = yr + R_w*sin(th);
xrwb = xr - R_w*cos(th);
yrwb = yr - R_w*sin(th);
xlw = [xlwf, xlwb];
ylw = [ylwf, ylwb];
xrw = [xrwf, xrwb];
yrw = [yrwf, yrwb];
```

Where,

xlwf = x position of left front wheel

yrwb = y position of right back wheel

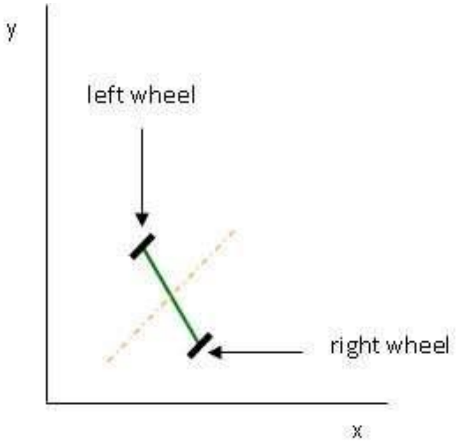


Figure 13 Robot model in MATLAB

The position of "xlwf" is the x position of the wheel on the left front, whereas "yrwb" is the position of the wheel on the right back. So that xlw and ylw are represented as (x,y) position on the left wheel by two points, on the other hand xrw and yrw are represented as (x,y) position on the right wheel by two points. Finally, all the points will be drawn as a line by using the plot statements, such as plot zb, plot zlw and plot zrw to plot the robot base at the main program.

Control Variable

In order to simulate the mobile robot, the following must be made, namely the initial velocity, initial state vector and initial for control vector, respectively as V0, (X0,Y0 and theta_rad0) and U0 = V0. The codes used for initialization are shown as follows. v0 = V(1); [m/s] z0 = [x0, y0, theta_rad0];

u0 = [v0];

The simulation will generate two trajectories of the movement. The total timing for both trajectories will be divided into two, the divided values will be used for each trajectory.

Simulation Results

The user can enter the desired motion. The program will generate the coordinates as shown in Table 1 Moreover, all dimensions of the robot modeling and environment in simulation are similar to the real one.

X coordinate	y coordinate	Orientation (□)	Remark
0.000000	0.000000	0.000000	Goes straight
0.174533	0.000000	0.000000	
0.349066	0.000000	0.000000	
0.523599	0.000000	0.000000	
0.698132	0.000000	0.000000	
0.872665	0.000000	0.000000	
1.047198	0.000000	0.000000	
1.221730	0.000000	0.000000	
1.396263	0.000000	0.000000	
1.570796	0.000000	0.000000	
1.570796	0.000000	0.000000	

1.745329	0.000000	22.304833	Turning curve
1.906803	0.066241	44.609665	
2.031055	0.188811	66.914498	
2.099490	0.349367	89.219331	
2.101868	0.523884	111.524164	
2.037833	0.686246	133.828996	
1.916967	0.812155	156.133829	
1.757358	0.882772	178.438662	
1.582890	0.887527	200.743494	
1.419671	0.825710	223.048327	

Table 1 The mobile robot trajectory as x,y, and□

In Table 1 above, with th = 0.000 deg and y = 0.000 the robot moves in straight line because there is no orientation. At the 11th step, theta begins to increase. This means the robot is turning along a curvature.

The combination of the speed between motor 1 and motor 2 along a curvature with different radius can be seen in Table 6.2.

No	Motor 1 (rpm)	Motor 2 (rpm)	Radius (meter)
1	100	300	0.268
2	150	300	0.4025
3	200	300	0.672
4	100	400	0.222
5	200	400	0.402
6	300	400	0.941
7	100	500	0.199
8	200	500	0.312
9	300	500	0.536
10	400	500	1.210

Table 2 The radius for curve turning of the robot

In Table 2 above, the radius of turning curvature of a robot depend on the different between the speeds of two motors. It can be seen in row number one (motor1 = 100 rpm, motor2 = 300), the radius obtained is 0.268. Base on the graph below, we can conclude that the incremental value of the radius is constant.

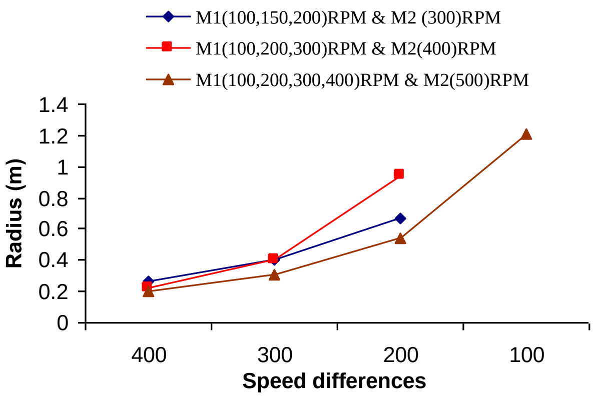


Figure 14 The incremental value of radius

Conclusion

The simulation program can manipulate the movement and to create the desired trajectory of a mobile robot. In addition, the simulation can also generate command for the real mobile robot program.

The linear error is smaller than the curvature error. A few factors affecting the motion of the robot such as friction on the wheel and surface, power lost in transmission, skid and slip was not considered. However, based on the calculations and the strategies in the simulation, the path is almost similar compared to the actual mobile robot path.

Moreover, the program helps the user to start the programming and optimize it to obtain the best trajectory and position of a mobile robot. It is expected that the user can operate the program without problems.

There are two outputs from the simulation. Firstly, x and y coordinates position and theta as the orientation angle.

Reference

1. Groover, M.P., at al., "*Industrial Robotics: Technology, Programming, and*

Application," McGraw-Hill, New York, 1978.

2. Giovanni C. Pettinaro, "*Simulation of SBot Mobile Robots*", Switzerland, 2003.
3. http://en.wikipedia.org/wiki/MATLAB,
04/02/2006
4. MATLAB Help Navigator, "*Introduction; What is MATLAB?*", Version 7.04.365 (R14), 2005.
5. G.W.Lucas: *A Tutorial and Elementary Trajectory Model for the Differential Steering System of Robot Wheel Actuator,* The Rossum Project volunteers.
6. Furukawa T, *Robot Design ; Lecture 3: Kinematic Fundamentals*, School of Mechanical and Manufacturing 7. Saeed B. Niku, "*Introduction to Robotics: Analysis, Systems, Application*", PrenticeHall, New York, 2001.