Motion Planning and Trajectory for Wheeled Mobile Robot

Sheha Ame Mnubi

Tianjin University of Technology and Education (TUTE), School of Mechanical Engineering1310,DaGuNan Rd,Hexi,Tianjin,China,P.C 300222

Abstract: *In general, the problems of robots can be divided into two sub-problems of motion planning and motion control. A motion planning problem is solved where a geometrical model is given. Furthermore, motion planning problems can be fundamentally divided into path, trajectory and task planning problems. These planning have many constraints concerning kinematics and dynamics of the robot as well its environment. This paper introduces path, trajectory and task planning methods for wheeled mobile robots. Wheeled mobile robots are becoming increasingly important in industry as a means of transport, inspection, and operation because of their efficiency and flexibility. The motion of a wheeled mobile robot, in general, be subject to non-holonomic constraints due to the rolling constraints of the wheels, which render a motion perpendicular to the wheels impossible. These non-holonomic constraints give rise to highly nonlinear mathematical models of the mobile robots, and the control problem is not trivial although the full state is measured. Feedback control of non-holonomic mobile robots is, therefore, a challenging problem which combines nonlinear control theory and differential geometry. Path planning in mobile robots must ensure optimality of the path. The optimality achieved may be in path, time, energy consumed etc. Path planning in robots also depends on the environment in which it operates like, static or dynamic, known or unknown etc. Global path planning using A* algorithm and genetic algorithm is investigated in this paper. A known dynamic environment, in which a control station will compute the shortest path and communicate to the mobile robot and the mobile robot will traverse through this path to reach the goal.*

Keywords: Wheeled mobile robot, Path planning, Kinematics Model. Feed forward Compensator.

1. Introduction

Recently, robots have been seen in various fields. In general, robots can be divided into mobile robots and arm robots (robot manipulators). Lately, the mobole robots with manipulators including humanoid robots have been developed improving the performance and flexibility. The main aim of mobile robots is to carry materials,products, tools and others,while the arm of robot manipulators is tp handle them. The motion of a robot is basically constrained by its dynamics, kinematics and environments. Because robots controlled using jionts and wheeled the motion of robots results from the motion of controllable degrees of freedom(DOF). Basically the problems of robots can be divided into two sub-problems of motion of motion planning and motion control. First,the motion planning is solved when a geometric model is given. Next, the motion control for a physical robot is done according to a planned trajectory.This paper focus on the motion planning. A robot receives task from a human operator and performs the task in the workspace including a lot of obstacles such as humans, machining centers, and other robots.

The robot should account the collision avoidance with the obstacles. Furthermore, the robot should generate its motion satisfying the spiral and temporal constraints for performing the task. There exist various method for solving motion planning problems (Donald et al. (1993), Fogel (1995), Goldberg (1989), bertsekas (1999), Ecker and Kupferschmid (1998), Canny (1988), Brooks (1983), Paul (1981), Russell and Narving (1995), Fukuda and Kubota (1999a,b), Fukuda et al. (1997), Lozano-Perez. (1981), Davidor (1991)).The mobile robot path planning problem is typically formulated as follows: given a mobile robot and a description of an environment, plan a path between two specified locations, a start and end point. The path should

be free of collision and satisfies certain optimization criteria (i.e., shortest cost path). According to this definition, path planning problem is categorized as an optimization problem. Researchers classify various methods used to solve the path planning problem based on two factors, (1) the environment type (i.e., static or dynamic), (2) the path planning algorithms (i.e., global or local).The static environment is defined as the environment which doesn't contain any moving objects other than a navigating robot; while the dynamic is the environment which has dynamic moving objects (i.e., human beings, moving machines and other moving robots. The global path planning algorithms requires all terrain to be static and also a complete knowledge about the search environment. On the other hand, local path planning means that path planning is being implemented while the robot is moving; in other words, the algorithm is capable of producing a new path in response to environmental changes.

A wheeled mobile robot is a wheeled vehicle which is capable of autonomous motion. Autonomous mobile robots are a very interesting
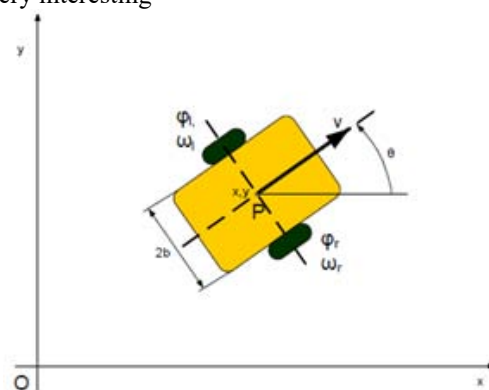

Figure 1: Position of mobile robot in plane

subject both in scientific research and practical applications. First the control strategies using the controller for each motor including the simple PI feedback controller and the feedforward compensator which has the inverse vehicle dynamics are proposed. In this paper the model of the vehicle has two driving wheels (which are attached to both sides of the vehicle) and the angular velocities of the two wheels are independently controlled. The center of the driving wheels is regarded as the gravity center. This model is the simplest and the most suitable for a small-sized and light, battery-driven autonomous vehicle.

2. Modeling of Wheeled Mobile Robots

2.1 Kinematics Constraints

We consider a mechanical system with generalized coordinate's q subject to m kinematics constraints:

$$A(q)\dot{q} = 0 \tag{1}$$

Where, $A \in R^{m \times n}$ is a full rank matrix. A large class of mechanical systems, such a wheeled vehicle and mobile robots involve kinematics constraints. In the literature these kinematics constraints can generally be classified as non-holonomic or holonomic. A mobile robot involving two actuator wheels is considered as a system subject to non-holonomic constraints.

2.2 Kinematics Model

Let's consider the kinematics model for an autonomous vehicle. The position of the mobile robot in the plane is shown in Figure1.

The inertial-based frame (Oxy) is fixed in the plane of motion and the moving frame is attached to the mobile robot. In this paper we will assume that the mobile robots are rigid cart equipped, with non-deformable conventional wheels, and they are moving on a non-deformable horizontal plane. During the motion: the contact between the wheel and the horizontal plane is reduced to a single point, the wheels are fixed, the plane of each wheel remains vertical, the wheel rotates about its horizontal axle and the orientation of the horizontal axle with respect to the cart can be fixed. The contact between the wheel of the mobile robots and the non-deformable horizontal plane supposes both the conditions of pure rolling and non-slipping during the motion. This means that the velocity of the contact point between each wheel and the horizontal plane is equal to zero. For low rolling velocities this is a reasonable wheel moving model. The center of the fixedwheel is a fixed point of the cart and b is the distance of the center of the wheel from P.

The rotation angle of the wheel about its horizontal axle is denoted by $\varphi(t)$ and the radius of the wheel by R. Hence, the position of the wheel is characterized by two constants: b and R and its motion by a time-varying angle:. The configuration of the mobile robot can be described by five generalized coordinates such as:

$$q = [x, y, \varphi_r, \varphi_l]^T \tag{2}$$

Where: x and y are the two coordinates of the origin P of the moving frame (the geometric center of the mobile robot), θ is the orientation angle of the mobile robot (of the moving frame), $\varphi_r(t)$ – the rotation angle of the right driving wheel, $\varphi_l(t)$ – the rotation angle of the left driving wheel. The vehicle velocity v can be found in equation (3):

$$V = R\frac{(\omega_r + \omega_l)}{2} \tag{3}$$

Where:

$\omega_r = \frac{d\varphi_r}{dt}$ – Angular velocity of the right wheel and

$\omega_l = \frac{d\varphi_r}{dt}$ – Angular velocity of the left wheel

The position and the orientation of the mobile vehicle are determined by a set of differential equations (4-6) in the following form:

$$\dot{x} = R\cos\theta\frac{(\omega_r + \omega_l)}{2} \tag{4}$$

$$\dot{y} = R\cos\theta \tag{5}$$

$$\dot{\theta} = R\frac{(\omega_r - \omega_l)}{2b} \tag{6}$$

Here,

$$\dot{x} = v\cos\theta \tag{7}$$
$$\dot{y} = v\sin\theta \tag{8}$$

Then the matrix form is:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} V \\ \dot{\theta} \end{bmatrix} \tag{9}$$

Finally, the kinematics model of the vehicle velocity v and the angular velocity $\dot{\theta}$ can be represented by the matrix as follows

$$\begin{bmatrix} V \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \dfrac{R}{2} & \dfrac{R}{2} \\ \dfrac{R}{2b} & \dfrac{-R}{2b} \end{bmatrix} \begin{bmatrix} \omega_r \\ \omega_l \end{bmatrix} \tag{10}$$

3. Path Planning for Mobile Robots

The path planning problem is one of the most fundamental problems in robotics systems. In general, a physical motion planning problem is transformed into geometrical path planning problem. Accordingly, a robot is represented a point in an appropriate search space, because the motion of a particle mass is easy to describe. Therefore, a path can be represented as a route from a point to another on the space. The path planning problems are close related with the collision avoidance problems. The motion planning problems for performing given task can be fundamentally divided into path planning problems, trajectory planning problems and task planning problems (figure 2).

**Figure 2:** Task, trajectory and path planning for robotic motion

In this case, we now consider the mobile robot motion as a non-holonomic mechanical system, where three kinematics constraints exist:

$$\dot{x}\,sin\theta - \dot{y}\,cos\theta = 0$$
$$\dot{x}\,sin\theta + \dot{y}\,cos\theta = R\omega_r - b\dot{\theta} \qquad (11)$$
$$\dot{x}\,sin\theta + \dot{y}\,cos\theta = R\omega_l + b\dot{\theta}$$

The constraints can be written in the form (1), where matrix $A \in R^{m \times n}$ (m=3, n=5) can be described as:

$$A = \begin{bmatrix} sin\,\theta & -cos\,\theta \\ cos\,\theta & sin\,\theta \\ cos\,\theta & sin\,\theta \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 \\ b & -R & 0 \\ -b & 0 & -R \end{bmatrix} \qquad (12)$$

In this article the angular velocities of the two wheels of the mobile robot are independently controlled actually, the path planning problem is to find a path connecting some points for avoiding the collision with the obstacles in a workspace. A general approach for the path planning is shown as follows. First, two-dimensional map around the robot including the starting and target points is built. An obstacle in the space is represented as an approximated polygonal objects. The size of objects is larger than the real size of the objects. If the size of a robot is added to that of a polygonal objects, a robot can be represented as a point in the workspace map, not a polygon, because the size of every polygonal object include the size of the robot (Figure 3). This method can simplify the workspace map with any accuracy. However, the workspace map is still continuous. In order to reduce the search space size, the workspace map can be transformed into various types of search spaces from the visual point of view. A search space is built by cell decomposition methods, skeletonization (roadmap) methods, and/or articial potential field methods (Fogel (1995), Goldberg (1989), Bertsekas (1999), Esker and Kupferchmid (1998), Canny (1988)). In the cell decomposition methods, a two dimensional workspace is basically divided into a finite number of cells. The skeletonization methods transforms the workspace into the set of vertices and path that enable a graph search. In the artificial potential field method, a robot moves based on attractive force from the target point and repulsive force from the obstacles in the workspace. Next, we must consider search algorithms for optimizing paths in the built map. Generally, the path planning can be classified into combinational optimization in the discrete search space and numerical optimization in the continuous space. The detail of optimization algorithms is explain later. In the following, we discuss how to build a map (search space) and how to solve the path planning problems.
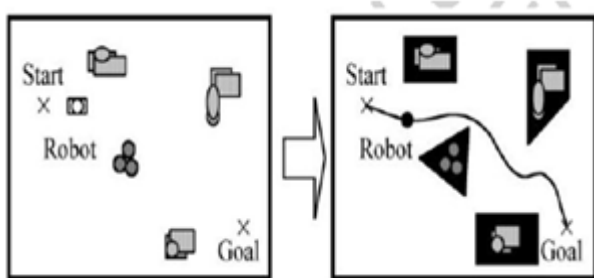
(a)Real workspace (b) Polygonal workspace
Figure 3: Polygonal workspace approximated from a real environment

4. Path Planning Algorithm

4.1 General Description

Our algorithm contains two general aspects: global static obstacle-avoidance path planning, and local dynamic collision-avoidance navigation. We adopt the A* algorithm to obtain the static optimum path and use trajectory prediction to achieve real-time collision-avoidance. We assume that there are only static obstacles at the initial time and use the A* algorithm to obtain a global optimum path from the initial position to the target position. Then, the robot measures the distance L between the center of the dynamic obstacle and itself when moving along the optimum path S. A threshold M, decided by (1) the velocity of the robot and (2) the sizes of the robot and the obstacle, is set up for distance L. The robot predicts the future position of the dynamic obstacle if L is smaller than M. Local navigation can be accomplished according to the result of the prediction and the value of L. The final optimum path can be obtained by repeating this process for each single planning cycle until the robot attains the target position.

4.2 Global Optimum Path

The A* algorithm is well known as one of the best-first search searching methods among the numerous global path-planning algorithms. The A* algorithm is actually an enlightening searching method, and its instinctive nature is "enlightening," which avoids blindness during the path–planning process. Hence, the A* algorithm is one of the popular path-planning algorithms. The D* algorithm is an improvement on the A* algorithm. It can continually re-plan the path using the A* algorithm in a dynamic environment. The evaluation function, $f(n)$, of the A* algorithm is given by the following expression:

$f(n) = g(n) + h(n)$ where $g(n)$ is the real cost for the node n to move from its initial position to the target position, and $h(n)$ is the predicted cost for the node n to move from its initial position to the target position along the optimum path. We plan the global optimum path using the grid method, and the cost between two nodes in a grid map is usually measured by their Euclidean distance. Thus, $h(n)$ can be represented as:

$$h(n) = \sqrt{(G.x - n.x)^2 + (G.y - n.y)^2}$$

The global optimum path derived from the A* algorithm reflects the trend and is treated as a reference for local navigation. In other words, the local navigation is adjusted to the global optimum path. The steps of the A* algorithm are as follows:

Step 1: Build OPEN and CLOSE lists. Initial them with NULL and insert the initial position R of the robot into the OPEN list.
Step 2: If OPEN = NULL, go to Step 5. If not, pick out the least node N in $f(n)$ and go to Step 4 if N = G, or else intend node N to eight-connected grids and put N into CLOSE list.
Step 3: Update the OPEN and CLOSE lists according to the values of $h(n)$ and $f(n)$ of the eight connected grids. Go to Step 2.

Step 4: Find the target position G and return the available path, or else go to Step 5.
Step 5: Fail to accomplish planning and return no available path.

4.3 Collision-avoidance during Local

Navigation
A mobile robot continually measures the distance L between a dynamic obstacle and itself when moving along the global optimum path derived from the A* algorithm and predicts the trajectory of the dynamic obstacle when L is not bigger than M. In this study, the robot measures the motion angle θ according to the positions of the obstacle in two contiguous planning cycles and defines an influential area that extends 4 *rad* around the motion angle. In this area, the motion of the robot is affected by the dynamic obstacle, and the effect of the dynamic obstacle on the mobile robot is measured by the Effect Factor (EF). The EF decreases in steps along the predicted moving direction Φ of the robot to reflect the trend of the impact caused by the dynamic obstacle on the robot. The initial value and the step-length of the EF are determined by the availability and the optimum trajectory of the path. If there is no position change of the obstacle in two contiguous planning cycles, θ does not exist; thus, EF only includes the initial value. Here, Φ can be given by the expression as:

$$\Phi = \frac{\pi}{4} \times k(\theta)$$

$$k(\theta) = m + 1, -2m < 7, m \in Z, \theta$$
$$\in [\frac{(2m+1)\pi}{8}, \frac{(2m+3)\pi}{8}]$$

The prediction results and the immediate value of L are the parameters needed for the local navigation algorithm if local navigation is required. If so, we use rolling-path planning to build a local rolling window for the robot's current position. This rolling window intersects the known global optimum path at a sub-target position. The sub-target position here is the final target in the current rolling window. Let the initial value of the sub-target grid be one; to build a dynamic-obstacle map, obtain the value of any other grid from the enlightening function $g(x, y)$ dx dy1 and initialize the value of the dynamic obstacle's grid, as shown in Figures 1 and 2. Next, plan the path using collision-avoidance between the robot's current position and the sub-target position in accordance with the values of the grids in the local rolling window.

5. Feed Forward Compensator

In this section a control strategy of wheeled mobile robots using feed forward compensator is proposed (Figure 4). The controller for each motor includes the simple PI feedback controller and the feedforward compensator which has the inverse vehicle dynamics. The inputs to the feed forward compensator are both sides of the desired wheel angular velocity (ω_r and ω_l – angular velocity of the right and left wheel).
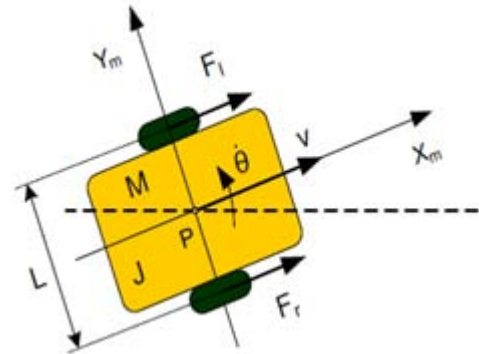

Figure 4: Parameter of a vehicle in motion

The vehicle motion equations are:

$$M\frac{dv}{dt} = F_r + F_l \qquad (13)$$

$$J\frac{d\dot{\theta}}{dt} = L(F_r + F_l) \qquad (14)$$

Where the parameters of the vehicle motion are:
F_r – Driven force at the right wheel,
F_l – Driven force at the left wheel,
M – Mass of the vehicle,
J – Inertia of the vehicle.
The following equations can formulate the torque (the vehicle dynamics) at each side of the driving wheel:

$$\tau_r = A\dot{\omega}_r + B\dot{\omega}_l + C\dot{\omega}_r \qquad (15)$$
$$\tau_l = D\dot{\omega}_l + E\dot{\omega}_r + F\dot{\omega}_l \qquad (16)$$

Where: A, B, C, D, E and F are the parameters of the motion of the vehicle. The proposed controller is as follows:

$$\tau_r = k_p(\omega_{rd} - \omega_r) + k_i\omega_{rd}\int_0^t(-\omega_r)dt + A\dot{\omega}_{rd} + B\dot{\omega}_{ld} + C\omega_{rd} \qquad (17)$$

$$\tau_l = k_p(\omega_{ld} - \omega_l) + k_i\int_0^t(\omega_{ld} - \omega_{rl})dt + A\dot{\omega}_{ld} + B\dot{\omega}_{rd} + C\omega_{ld} \qquad (18)$$

Where: k_p – is the proportional gain, k_i – is the integral gain, ω_{rd} – is the desired value of the angular velocity of the right wheel, ω_{ld} - is the desired value of the angular velocity of the left wheel. The structure of the proposed controller is presented in Figure 5:
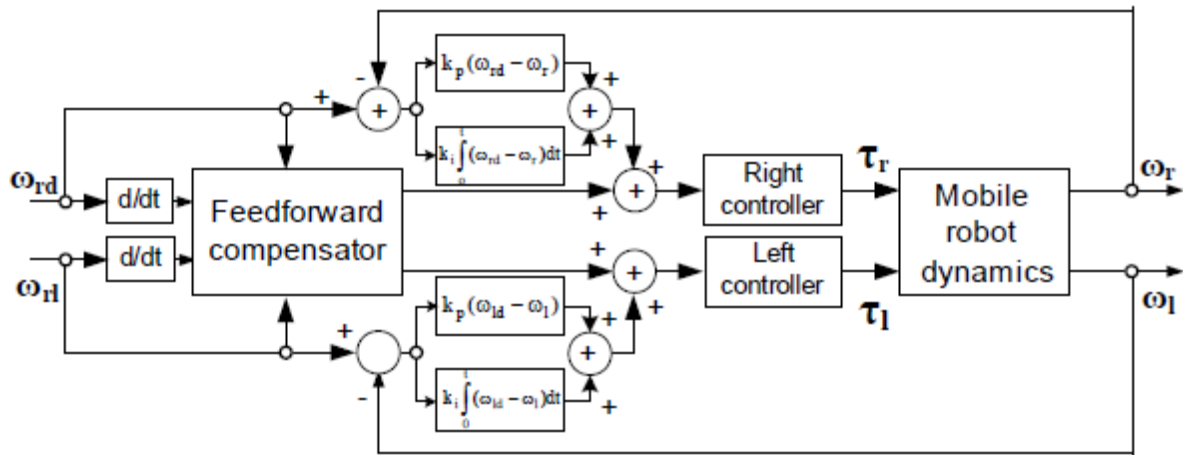
Paper ID: NOV152938
1067

Figure 5: Block diagram of the proposed controller using feedforward compensator

6. Conclusions

The article deals with the modeling and control strategies of the motion of wheeled mobile robots. The model of the vehicle has two driving wheels and the angular velocities of the two wheels are independently controlled. The vehicle kinematics model and the control strategies using a feedforward compensator are analyzed. When the vehicle is moving towards the target and the sensors detect an obstacle, an avoiding strategy is necessary.

References

[1] Craig J. J (2004), Introduction to Robotics: Mechanics and Control, Prentice Hall; 3rded.

[2] B.Donaid., P.xavier.J.Canny.,andJ.Reif. (1993), Kinodynamic Motion Planning, Joumalof Association for Computing Mechinery, Vol.40, No 5, pp.1048-1066.

[3] Delling,D.; Sanders,P.;Schultes, D.; Wagner,D. (2009)."Engineering Route planningalgorithms". Algorithmics of Large and Complex Networks: Design, Analysis, and Simulation. Springer. pp. 117–139. Doi:10.1007/978-3-642-02094-0_7.

[4] GyulaMester: Modeling the Control Strategies of Wheeled Mobile Robots, Proceedings of the KandóKonferencia 2006, pp. 1-4, Budapest, Hungary, 2006.

[5] Jian Wanga, XiangyangZhub, Masahiro Oyac, Chun-Yi Sud: Robust Motion Tracking Control of Partially Non-holonomic Mechanical Systems, Robotics and Autonomous Systems, Vol. 35, pp. 332-341, 2006

[6] S.Iida, S.Yuta: "Feedforward Current Control Method using 2-dimensional Table for DC Motor Software Servo Systeni",in Proc. of IEEE Int. Conf. of IECON'88, pp.466-471, Oct.1988.

[7] Alexander, J.C. and Maddocks, J.H.: On the Kinematics of Wheeled Mobile Robots.

[8] The International Journal of Robotics Research 8(5), 15-27, 1989,

[9] D'Andréa-Novel, B.; Campion, G. and Bastin, G.: Control of non-holonomic wheeled mobile robots by state feedback linearization.

[10] Y.Davidor. (1991), A genetic Algorithm Applied to Robot Trajectory Generation, in Handbook of Genetic Algorithms, Van Nostrand, pp.144-165.

[11] R.A. Books. (1983), Planning Collision Free Motion for Pick and Place Operation, Robotics Research, MIT Press, pp. 5-38.