Line following of a mobile robot

May 18, 2004

1 In brief....

The project is about controlling a differential steering mobile robot so that it follows a specified track. Steering is achieved by setting different angular velocities on the wheels. The track is comprised of straight lines and circular curves. It is possible to take advantage of this knowledge switching the control algorithm when the robot enters a curve. A simulink block and a M-file that sets suitable parameters is provided with the robot.

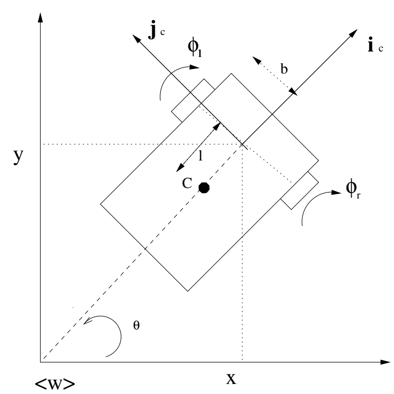

Figure 1: Schematic representation of the nonholonomic vehicle.

2 System Dynamics

The modeling of a two-wheeled nonholonomic vehicle is a problem deeply analyzed by a several papers in the past (see, as an example, [1, 2]). In this context, a few basic concepts will be recalled and some notational conventions, useful throughout the papers, will be established.

The mobile robot is modeled as a two wheels nonholonomic vehicle, endowed with an external sensor (e.g. a camera) able to determine the Cartesian relative position of a moving target. Since it is assumed that the wheels can not slide on the ground, in the direction perpendicular to the actual motion, in order to derive the system dynamics, it is necessary to take into account these non-integrable kinematic constraints. The analytical dynamics of the vehicle is determined using the principle of virtual displacements and the d'Alembert-Lagrange equations [3]:

$$\sum_{k=1}^{n}\left(\frac{d}{dt}\frac{\partial T}{\partial \dot{q}_k} - \frac{\partial T}{\partial q_k} - Q_k\right)\delta q_k = 0 \quad , \tag{1}$$

where n is the dimension of the configuration space, $q_k,\ k = 1,\dots,n$ are the generalized coordinates of the configuration space, T is the kinetic energy of the system, and Q_k are the generalized forces. In particular, the state of the mechanical system evolves in a configuration space which is five dimensional (see fig.1), i.e. there are five generalized coordinates ($n = 5$): the point of intersection $[x,\ y]^{\mathrm{T}}$ (expressed in the inertial frame $< w >$) of the symmetry axis of the carriage with the axle on which the wheels are mounted; the orientation angle θ between the symmetry axis of the carriage and a fixed axis of the frame $< w >$; and the angles of rotation of the left-hand wheel ϕ_l and of the right-hand one ϕ_r. The vehicle physical parameters are denoted as follows: C_o is the center of the mass of the carriage, m_0 is the mass of the carriage, m_1 and r are respectively the mass and radius of each wheel, k_o is the radius of gyration of the carriage about the vertical passing through $[x,\ y]^{\mathrm{T}}$, l is the distance between C_o and the point $[x,\ y]^{\mathrm{T}}$, A is the moment of inertia of the wheel about a diameter, and C is the axial moment of inertia including the actuator inertia. The absence of sliding in the direction of translational motion for both the wheels, and in the lateral direction lead to three kinematic non-integrable constraints

$$\begin{aligned}
\dot{x} &= r\,\frac{\dot{\phi}_l + \dot{\phi}_r}{2}\cos\theta \\
\dot{y} &= r\,\frac{\dot{\phi}_l + \dot{\phi}_r}{2}\sin\theta \\
\dot{\theta} &= \frac{r}{2\,b}(\dot{\phi}_r - \dot{\phi}_l) \quad ,
\end{aligned} \tag{2}$$

hence the system has 2 degrees of freedom, i.e. there are only two independent virtual displacements. Defined $E_x = \frac{d}{dt}\frac{\partial T}{\partial \dot{x}} - \frac{\partial T}{\partial x} - Q_x$, the similar expressions of E_y, E_θ, E_{ϕ_r}, E_{ϕ_l}, by directly eliminating the dependent virtual displacements as functions of the independent variations: $\delta x = \frac{r\cos\theta}{2}(\delta\phi_l + \delta\phi_r)$, $\delta y = \frac{r\sin\theta}{2}(\delta\phi_l + \delta\phi_r)$, and $\delta\theta = \frac{r}{2b}(\delta\phi_r - \delta\phi_l)$, and since $Q_x = Q_y = Q_\theta = 0$, being $Q_{\phi_r} = \tau_r$, and $Q_{\phi_l} = \tau_l$ the torques applied, respectively, to the right and left wheel, the system dynamics results

$$\begin{aligned}
m_{11}\ddot{\phi}_r + m_{12}\ddot{\phi}_l + c_1(\dot{\phi}_l^2 - \dot{\phi}_r\,\dot{\phi}_l) &= \tau_r \\
m_{21}\ddot{\phi}_r + m_{22}\ddot{\phi}_l + c_2(\dot{\phi}_r^2 - \dot{\phi}_r\,\dot{\phi}_l) &= \tau_l \quad ,
\end{aligned} \tag{3}$$

the inertia matrix $M = \{m_{ij}\},\ i,j = 1,2$, has been defined as $m_{11} = m_{22} = \frac{mr^2}{4} + \frac{Jr^2}{4b^2} + C$, $m_{12} = m_{21} = \frac{mr^2}{4} - \frac{Jr^2}{4b^2}$, where $m = m_0 + 2m_1$, and $J = \frac{1}{2}m_0k_0^2 +$

$m_1 b^2 + A$. The quantities $c_1 = c_2 = \frac{m_0 l r^3}{4b^2}$, which are constant coefficients, are related to the geometry and mass of the system. We also assumed that the torques applied to the wheels are generated by permanent-magnet D.C. motors (one for each wheel), able to provide torques proportional to the actual current in the armature circuit. In the rest of the paper we will address either the case of current-driven motors, able to generate a torque proportional to the command by an analog control loop, or the case of voltage driven motors. In the latter case, the considered motor equations for the right wheel are:

$$\begin{array}{rcl} \tau_r & = & k_p\, I_r \\ V_r & = & R_a I_r + L_a\, \dot{I}_r + k_p\, \dot{\phi}_r \quad , \end{array} \tag{4}$$

where I_r is the armature current, k_p is the constant gain between the torque and the current, V_r is the armature voltage, and R_a, L_a are the circuit parameters. We assume that the equation relative to the left wheel has the same parameters (k_p, R_a, L_a). When dealing with voltage driven motors, the above linear equation will be included in the equations of the whole system and accounted for in the control law synthesis.

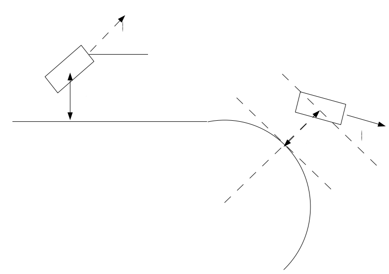

Figure 2: Schematic representation of the nonholonomic vehicle.

3 Sensors

Sensors deployed on the vehicle collect the following data:

- left and right angular velocities; these data are collected by an optical encoder and the minimum sampling period is 1ms
- distance y from the line (see Figure 2); this distance can be either positive or negative meaning that the the vehicle is located in the internal or in the external part of the track. This information is collected by a sensor whose minimum sampling period is 20ms. The sensor has a maximum range of $1m$ beyond which the line is not measurable.
- inclination with respect to the direction of the path (see Figure 2); also this information is collected by a sensor whose minimum sampling period

is 20ms. Also for this information the sensor has a limited range of $1m$ beyond which it looses sight of the track.

- Curve indication: upon entrance and exits from curve an event-based sensor notifies that the vehicle is on clockwise or on a counter-clockwise curve.

4 Actuators

Actuators are the armature tensions on the left and right motors. There is saturation on the motors, maximum tension is 50 volts.

5 Control goals

5.1 Mandatory goals

The minimum required goal is to make the vehicle follow the road with a minumum constant velocity of $1m/s$. The vehicle has to always staty within the range of the path. Higher velocities are appreciated provided that the range requirement is always complied with.

5.2 Optional goals

- When the controller is notified the "enter-curve" or "exit-curve" event it can change the vehicle speed and or swithch to a different control algorithm.

- The sensors that measures the inclination of the vehicle can suffer faults. In this case it is possible to switch to a degraded algorithm that reconstruct the inclination using an observer but the speed is reduced to $0.5m/s$.

6 Structure of the software and implementation

All the files needed are in directory "shark/demos/rtw". The directory includes many files, most of which have been automatically generated by Real-Time workshop and are needed for simulating the robot.

The most important files are:

- **rtw.c** This file includes the main() function, the system initialization, the simulation tasks. You should not modify this file.

- **control.c** This file is the one that should include the control algorithm. Currently it includes only one task that performs a very simple kinematic control.

- **percorso.c** This file includes the definition of the path. You should not modify this file.

- **carrello0_grt_rtw/carrello0.c** This file is the initialization file for the Simulink simulation. At the beginiing of the file there is a `#define DURATA` that is used to set the lenght of the simulation in seconds.

6.1 Units of measure

The position is in meters, angles are in radiants, wr and wl are in rad/seconds, tension is in volts.

6.2 Sign conventions

Distance `d` is from the center of the robot to the path. It is positive if the path is to the left of the robot, it is negative if the path is to the right of the robot.

Angle `th` is the angle between the robot direction axis and the path tangent. It is positive if the robot must turn left to get in line with the path; it is negative if the robot must turn right to get in line with the path.

References

[1] P.F. Muir and C.P. Neuman, "Kinematic modeling of wheeled mobile robots", *Journal Robotic Systems*, vol.4, no.2, pp. 281-329, 1987.

[2] G. Campion, G. Bastin and B. D'Andrea-Novel, "Structural properties and classification of kinematic and dynamic models of wheeled mobile robots", *IEEE Trans. on Robotics and Automation*, vol 12, no.1, 1996.

[3] J.I. Neimark and N.A. Fufaev, *Dynamics of Nonholonomic Systems*, American Math. Society, Providence, Rhode Island, 1972.