2008 IEEE International Conference on
Robotics and Automation
Pasadena, CA, USA, May 19-23, 2008

Practical Construction and Position Control of a Modular Actuated Holonomic Wheeled Mobile Robot

Ahmed El-Shenawy Achim Wagner Essam Badreddin
Automation Laboratory
University of Heidelberg
Germany D-68131
ashenawy@rumms.uni-mannheim.de

Abstract— **The presented work is related to previous development of the holonomic wheeled mobile robot C3P. This paper focuses on the platform implementation and the kinematics/dynamics solutions used for its position control structure. The platform prototype is proposed in detailed description concerning its construction and configuration. A controller based on feed forwarding the inverse dynamics torques with the inverse kinematics to overcome the platform singularities is proposed. The based controllers practical experiments results illustrate the position controller performance and its efficiency.**

I. Introduction

The **W**heeled **M**obile **R**obots (WMRs) research area has a high speed of development and implementation through the last decades [1], [2], [3] and [4]. The development of the WMR is inspired by many objectives; the main objective is developing a wheeled mobile robot with the best motion behavior. Such target released the ideas and creativity of the researchers to over come the mobility problems without taking in consideration the complexity and hardware construction of the platforms and their actuation. The WMR illustrated in this paper is influenced by the main objective of ”Developing a holonomic wheeled mobile robot, which is simple in construction and efficient in performance”.

A holonomic configuration implies that the numbers of robot velocity DOF (Degrees Of Freedom) are equal to the number of position coordinates. The WMR normally drives on a planner surface with three position co-ordinates: X, Y and rotational angle around Z which is θ_z. Therefore, the holonomic WMR is the robot that can drive in three degrees of freedom (3DOF) [5].

WMR are among the most complex engineering systems to be designed [6][7] and the hardware simplicity affects the WMR mobility characteristics. As a result many problems may be found, for example singularities and actuator conflicts are the two main problems facing different configurations of holonomic mobile robots. Such problems affects the kinematic and dynamic properties, which are challenging from the theoretical and practical point-of-view.

This paper is an extension for previous work, which concentrated on the theoretical and methodological part of developing the holonomic mobile robot C3P (**C**aster **3** wheeled **P**latform). The C3P theoretical problem was found and analyzed previously in [8]. The robot dynamic model was developed for the simulation process based on Euler Lagrange method in [9]. The inverse dynamics problem was proposed and compared to the inverse kinematics problem in [10].

In this paper the practical implementation and configuration is described in section II. In section III the singularity problem is formulated with an overview about the theoretical kinematics and dynamics modeling. A combination between the inverse kinematics and inverse dynamics solution is proposed to develop a based velocity and position controller for the C3P in IV. The results of practical experiments illustrating the behavior of the proposed based controller are shown in section IV-B. The proposed combined controller shows its efficiency on the practical implemented prototype.

II. Hardware Configuration

The C3P platform is realized in the Automation Laboratory. The practical platform used in this paper is the first prototype of the C3P configuration shown in Fig.1.

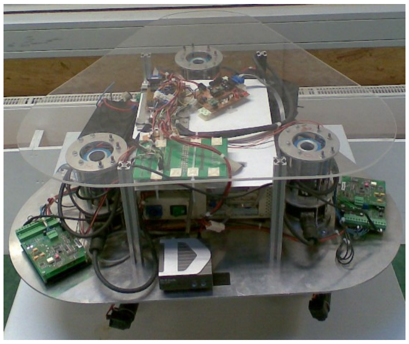

Fig. 1. The C3P prototype

The platform has three caster wheel units, each unit contains four main components; i) Brush-less DC motor for the wheel angular velocity actuation, ii) incremental encoders for sensing the angular velocity of the wheels, ii) absolute encoders for measuring the wheel steering angle, and iv) slip rings for signal transfer between the motors and the control cards. The caster wheel unit has two levels; the upper level (attached to the platform) and the lower level (attached to the wheel). The lower level is shown in Fig.2 with the DC-Servomotors, which are mounted on the driven axis of each wheel along with the incremental encoders.

The wiring of each motor and each encoder is connected to a slip-ring mounted on the upper level of the caster

978-1-4244-1647-9/08/$25.00 ©2008 IEEE.

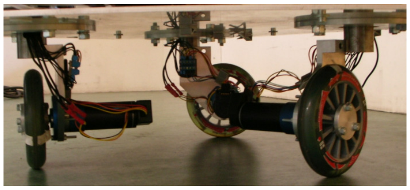

Fig. 2. The lower level of the caster wheels units

wheel unit Fig.1. The slip ring has 12 rings with 16 Amper, and below the slip-ring the absolute encoder is mounted. The motors are controlled by a velocity/torque control cards mounted on the platform body. The controller is 4-Quadrant PWM Servo Amplifier, 24 V power supply with rated speed 40,000 r/min. The sensed and control signals are transferred to an industrial computer with 1 GHz speed running under QNX operating system.

A. Geometric Configuration

The C3P WMR is a holonomic mobile robot, which is previously discussed in [11]. The C3P has three caster wheels attached to a triangular shaped platform with smooth rounded edges as shown in Fig.3. Each caster wheel is attached to each hip of the platform. The platform origin coordinates are located at its geometric center, and the wheels are located with distance h from the origin and $\alpha_1 = 30^o$, $\alpha_2 = 150^o$, and $\alpha_3 = 270^o$ shifting angles,

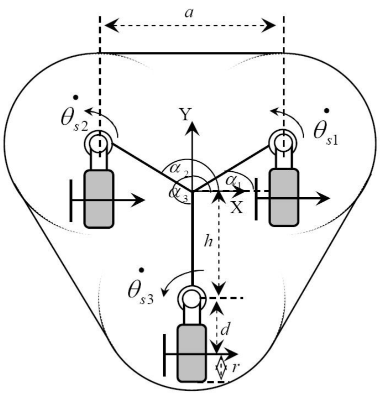

Fig. 3. C3P platform configuration

where

X, Y, ϕ : WMR translation and rotation displacement.
θ_{s_i} : the steering angle for wheel i.
r, d : the wheel radius and the caster wheel offset.

The conventional Caster wheel has three DOF due to the wheel angular velocity $\dot{\theta}_{x_i}$, the contact point angular velocity $\dot{\theta}_{c_i}$ and the steering angular velocity $\dot{\theta}_{s_i}$. The C3P has modular actuation by only the angular wheel velocities ($\dot{\theta}_{x_i}$), the steering angular velocities ($\dot{\theta}_{s_i}$) are not actuated.

III. Problem Formulation and Modeling

In order to analyze the mobility and to control the robot platform , first the system is modeled on the kinematic level. Furthermore, calculating the velocity and acceleration variables are important in the dynamics modeling procedures. In order to analyze and derive the C3P mathematical models, some variables are assigned, such as the following: the robot position vector $p = [X\ Y\ \phi\ ]^T$, the wheel angles vector $q_x = [\theta_{x_1}\ \theta_{x_2}\ \theta_{x_3}]^T$, the steering angles vector $q_s = [\theta_{s_1}\ \theta_{s_2}\ \theta_{s_3}]^T$, and the contact angles vector $q_c = [\theta_{c_1}\ \theta_{c_2}\ \theta_{c_3}]^T$. By differentiating the robot and wheel vectors with respect to time, the robot and wheel velocities are

$$\dot{p} = \frac{dp}{dt},\ \dot{q}_x = \frac{dq_x}{dt},\ \dot{q}_s = \frac{dq_s}{dt},\ \dot{q}_c = \frac{dq_c}{dt}. \tag{1}$$

A. Kinematics Modeling

From the generalized inverse kinematic solution described in [12], the wheel angular velocity inverse kinematic solution is

$$\dot{q_x} = J_{in_x}\dot{p}$$

$$J_{in_x} = \tfrac{1}{r}\begin{bmatrix} -S(\theta_{s_1}) & C(\theta_{s_1}) & h\,C(\alpha_1 - \theta_{s_1}) \\ -S(\theta_{s_2}) & C(\theta_{s_2}) & h\,C(\alpha_2 - \theta_{s_2}) \\ -S(\theta_{s_3}) & C(\theta_{s_3}) & h\,C(\alpha_3 - \theta_{s_3}) \end{bmatrix} \tag{2}$$

while the steering angular actuation is

$$\dot{q_s} = J_{in_s}\dot{p}$$

$$J_{in_s} = \tfrac{-1}{d}\begin{bmatrix} C(\theta_{s_1}) & S(\theta_{s_1}) & -h\,S(\alpha_1 - \theta_{s_1}) + d \\ C(\theta_{s_2}) & S(\theta_{s_2}) & -h\,S(\alpha_2 - \theta_{s_2}) + d \\ C(\theta_{s_3}) & S(\theta_{s_3}) & -h\,S(\alpha_3 - \theta_{s_3}) + d \end{bmatrix} \tag{3}$$

and the contact angular velocity inverse solution is

$$\dot{q_c} = J_{in_c}\dot{p}$$

$$J_{in_c} = \tfrac{-1}{d}\begin{bmatrix} -S(\theta_{s_1}) & C(\theta_{s_1}) & -h\,C(\alpha_1 - \theta_{s_1}) \\ -S(\theta_{s_2}) & C(\theta_{s_2}) & -h\,C(\alpha_2 - \theta_{s_2}) \\ -S(\theta_{s_3}) & C(\theta_{s_3}) & -h\,C(\alpha_3 - \theta_{s_3}) \end{bmatrix} \tag{4}$$

where "C" stands for "cos" and "S" stands for "sin". The solution (2) shows singularities for some steering angles configurations. The singularity appears only when the steering angles are equal. For example, when the steering angles are -90^o, the robot velocity $\dot{Y}$ is not actuated (Fig. 4-a), and when they are 0^o the velocity $\dot{X}$ is not actuated (Fig. 4-c). The steering configuration in Fig. 4-b gives singular determent for the matrix J_{in_x} with -45^o steering angles although all the robot DOFs are actuated.

Obviously, the direction of $[-1\ 1\ 0]^T$ is not actuated, which result the conclusion; if all steering angles yield the same value, then the robot is not actuated in the direction parallel to the wheel axes. Fig.4-d represents a non-singular steering wheels configuration condition.

The forward sensed kinematics used in [13] shows that the sensed variables are sufficient for robust sensing and slippage detection through the following equation

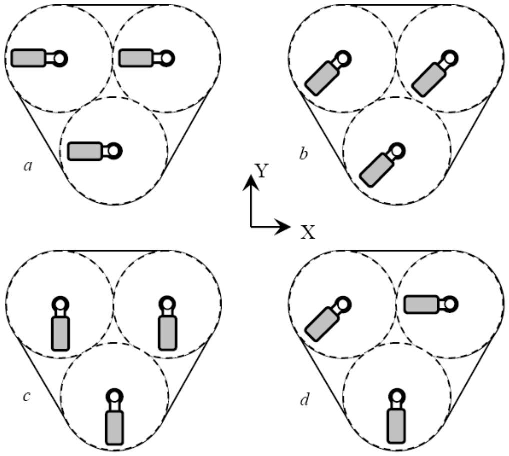

Fig. 4. Different steering angles configurations

$$\dot{p} = J_{f_x}\dot{q}_x + J_{f_s}\dot{q}_s, \tag{5}$$

where J_{f_x} and J_{f_s} are the sensed forward solutions for the wheel angular and steering angular velocities respectively. The derivative of equation (5), yields the robot accelerations,

$$\ddot{p} = J_{f_x}\ddot{q}_x + J_{f_s}\ddot{q}_s + g(q_s, \dot{q}_x, \dot{q}_s), \tag{6}$$

from equations (2), (3) and the inversion method proposed in [12] the inverse actuated kinematic accelerations is

$$\begin{bmatrix} \ddot{q}_x \\ \ddot{q}_s \end{bmatrix} = \begin{bmatrix} J_{in_x} \\ J_{in_s} \end{bmatrix} \ddot{p} - g_{cs}(q_s, \dot{p}) \tag{7}$$

B. Dynamics Modeling

The inverse dynamic solution proposed was used previously in the velocity control loop to overcome the singularity with a simpler velocity controller and better performance [10]. The inverse dynamic equation depends on the platform constraints, which are described in the forward kinematic solution. They are combined using Lagrangian formulation [14] and the dynamic torque equation

$$\tau = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\right) - \frac{\partial L}{\partial q}, \tag{8}$$

to obtain the described wheel torques equation

$$\tau_{x_a} = \begin{bmatrix} M_{x_a} & M_{s_a} \end{bmatrix} \begin{bmatrix} \ddot{q}_x \\ \ddot{q}_s \end{bmatrix} + G_{sx_a}(\dot{q}_x, \dot{q}_s, q_s) \tag{9}$$

where τ_{x_a} is the vector of actuated torques. The matrix M_{x_a} is the inverse dynamic solution for actuating the wheels torques τ_{x_a}, while the matrix M_{s_a} is the inverse dynamic solution for actuating the steering angular acceleration $\ddot{q}_s$ using the wheel torques τ_{x_a}. The inverse dynamics solution is a relation between the desired robot velocities and accelerations ($\dot{q}$, $\ddot{q}$) as an input and the actuated applied torques of the wheels (τ_{x_a}) as an output. However, the dynamic torque equation (9) is a function of $\ddot{q}_x$, $\ddot{q}_s$, $\dot{q}_x$, and $\dot{q}_s$. By using the velocity and acceleration inverse kinematic solutions(2, 3, 4 and 7), the desired torque equation is found and the actuation characteristics of the steering angular velocities and accelerations are included in the inverse dynamic solution as well. As a result the actuated torques equation will have the robot velocities $\dot{p}$ and accelerations $\ddot{p}$ as input variables

$$\tau_{x_a} = M_x(q_s)\ddot{p} + G_{xi}(q_s, \dot{p}) \tag{10}$$

$$M_x = \begin{bmatrix} M_{x_a} & M_{s_a} \end{bmatrix} \begin{bmatrix} J_{in_x} \\ J_{in_s} \end{bmatrix}, \tag{11}$$

$$G_{xi}(q_s, \dot{p}) = G_{sx_a}(q_s, \dot{p}) - \begin{bmatrix} M_{x_a} & M_{s_a} \end{bmatrix} g_{cs}(q_s, \dot{p}) \tag{12}$$

IV. Dynamics and Kinematics Based Position controller

The C3P motion control consists of two main cascaded controllers; velocity controller and position controller. In this section, the inverse kinematic and the inverse dynamics solutions are used in the velocity and position controllers structure as shown in Fig.5.

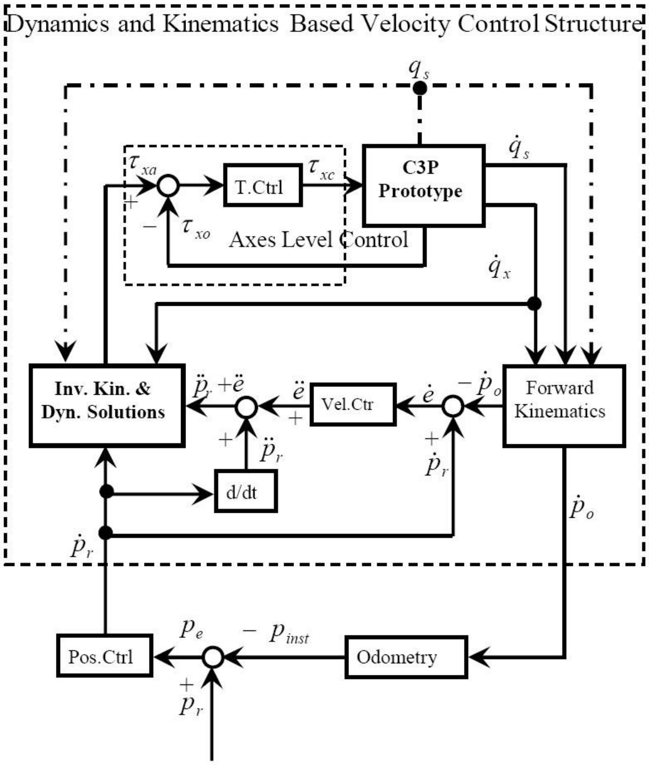

Fig. 5. Different steering angles configurations

The Axes level control loop is applied on the torque control of each wheel (T.Ctrl). The controller is already implemented on the DC Driver card as a regular proportional controller. The velocity controller (Vel.Ctrl.) is used to calculate the required error in acceleration ($\ddot{e}$), which is added to the reference robot acceleration $\ddot{p}_r$ (Fig.5). The reference robot velocities $\dot{p}_r$ and accelerations $\ddot{p}_r + \ddot{e}$ are used in the inverse dynamic solution to deliver the actuated wheels torques τ_{xa}. The position controller is used to calculate the reference velocity vector $\dot{p}_r$ needed to drive from initial co-ordinates p_i to goal co-ordinates p_g.

In the practical implementation the usage of the inverse kinematic solution for the singularity problem may not be effective. Therefore, the inverse kinematics solution is used as the steady state solution and the inverse dynamics solution is used to feed-forward the kinematics in case of singularities. The combined inverse solution is shown in Fig (6).

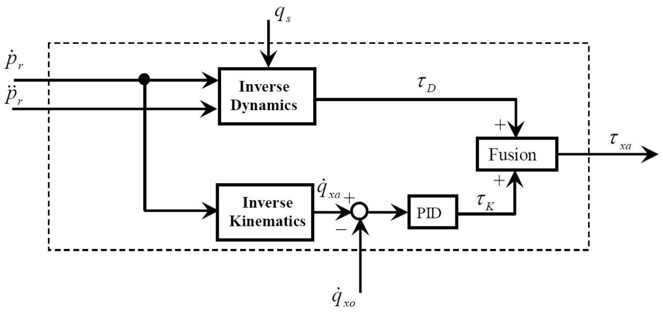

Fig. 6. Dynamics and Kinematics fusion block

The " Dynamics and Kinematics fusion " block has one main function, fusing between the inverse kinematics (1)and the inverse dynamics solution (2). In the case of using the inverse dynamic solution separately the wheel torques found in the vector τ_D are not symmetrically accumulated in the torque controller signal due to the friction and slippage problem. These unsymmetrical accumulations results in unstable behavior of the C3P, therefor the inverse kinematic solution is used to stabilize the behavior and over come the instabilities (Fig. 6). The fusion criterion depends mainly on the effect of each solution, for example the inverse dynamic solution main function is solving the singularity problem, therefore it is needed in the transient phase in case of a singular condition. The inverse kinematic solution is sufficient in the steady state phase in case of non-singular condition. As a result, the fusion equation is a function of the singularity indicator Ψ as described below

$$\tau_{xa} = (1 - cos(|\Psi|))^{\lambda_1} \tau_D + cos(|\Psi|)^{\lambda_2} \tau_K, \qquad (13)$$

where τ_D is the wheel torque resulted from the inverse dynamic solution and τ_K is the torque resulting from the wheel velocity axes control. The fusion parameters λ_1 and λ_2 are tuned manually according to the operator desgin.

The singularity indicator $|\Psi|$ detects the robot singularity conditions. The angle $\Psi = \delta - \beta$, where $\delta = \arctan(-\dot{Y}_a, \dot{X}_a)$ and $\beta = \arctan(-\dot{Y}_r, \dot{X}_r)$.

$\dot{X}_a, \dot{Y}_a$: The measured robot Linear velocities in X and Y directions

$\dot{X}_r, \dot{Y}_r$: The reference robot Linear velocities in X and Y directions

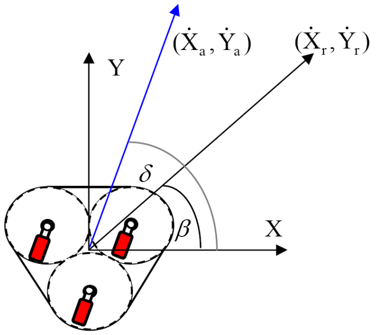

Fig. 7. The geometric representation for obtaining Ψ

The indicator $|\Psi|$ works as a sensor for the C3P singularity condition. The condition of $|\Psi| = 90^o$ indicates for pure un-actuation in the desired linear velocity direction $(\dot{X}, \dot{Y})$. And the condition of $|\Psi| = 0^o$ or $|\Psi| = 180^o$ stands for full ability of motion in the desired direction.

A. Position Controller

The position controller has 3 main controlled variables, $\dot{X}$, $\dot{Y}$and $\dot{\Phi}$. The position control problem can be stated as follows. Given position goal co-ordinates vector $p_g = \begin{bmatrix} X_g & Y_g & \Phi_g \end{bmatrix}^T$ and initial position reference co-ordinates vector $p_i = \begin{bmatrix} X_i & Y_i & \Phi_i \end{bmatrix}^T$.

The robot is assumed to drive in straight line between the initial and the goal points Therefore, there is another variable should be controller, which is δ_{er}. The variables $\delta_{er} = \vartheta - \delta_g$, where $\vartheta = \arctan(\frac{-Y_g}{X_g})$ and $\delta_g = \arctan(\frac{-Y_e}{X_e})$ (Fig.8)

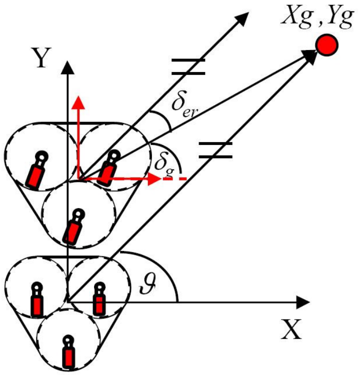

Fig. 8. Robot Position representation

Where $p_e = p_g - p_i = \begin{bmatrix} X_e & Y_e & \Phi_e \end{bmatrix}^T$ is the displacement error vector in the 3DOFs . From the previous section, the velocity controller succeeded in driving the robot in the floor co-ordinates but aiming for wrong goal point with distance error. The angle δ_{er} is the difference angle between the direction of the robot and the error vector with reference to the mass point of the robot at every instant.

The controller deliver robot reference velocities value to the velocity controller. Such is exponentially a function of the robot displacement error as the following

$$\dot{X} = sign(X_e)K_x(1 - e^{(-\mu_x \|X_e\|)}), \tag{14}$$

$$\dot{Y} = sign(Y_e)K_y(1 - e^{(-\mu_y \|Y_e\|)}), \tag{15}$$

and the rotation angular signal will be the same

$$\dot{\Phi} = K_\Phi \Phi_e + K_{er}\delta_{er} \tag{16}$$

The parameters μ_x and μ_y are tuned according to the required smoothness of the robot velocity signal while the robot reaches the goal point.

B. Practical output

Before demonstrating the practical results, all the robot and controller parameters are set for the used experiments as shown in Table I.

TABLE I
THE C3P PARAMETERS

C3P Parameters	Value	Units	Ctrl. Par.	Value
h	0.343	m	K_x,K_y	0.5, 0
d	0.04	m	K_Φ,K_{er}	0.2, -0.3
r	0.04	m	μ_x,μ_y	1.2, 1.2
M_p (P_l mass)	30	Kg	λ_1,λ_2	0, 0
I_p (P_l inertia)	3.51	$Kg\ m^2$	θ_{s_i}	0^o

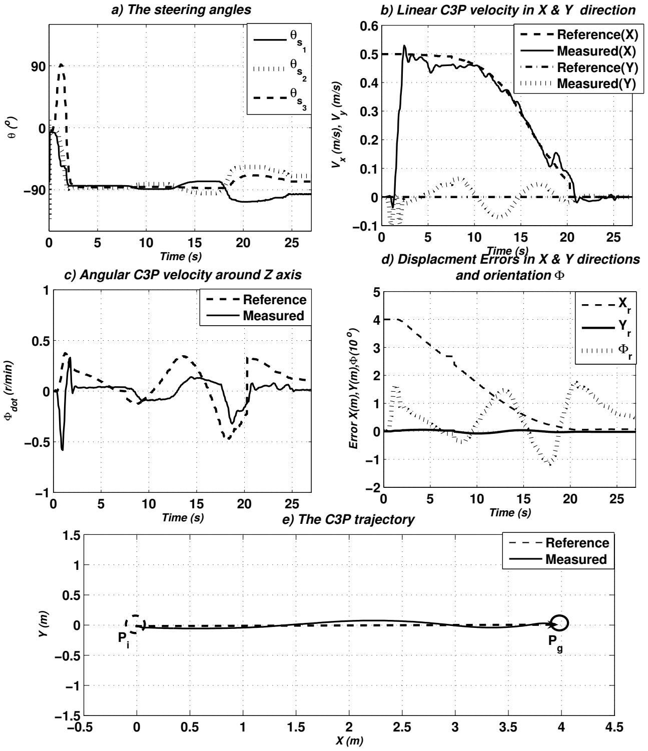

Fig. 9. The position controller behavior for driving from from $p_i = [0\ 0\ 0]^T$ to $p_g = [4m\ 0m\ 0^o]^T$

The first experiment for the position controller implementation is to drive from initial point $p_i = [0\ 0\ 0]^T$ to goal point co-ordinate of $p_g = [4m\ 0m\ 0^o]^T$. The initial steering angles are all around value of 0^o which is an singular starting condition and the controller parameters are shown in Table I. The robot results are shown in Fig. 9, where the steering angles took around 2 seconds to adjust their values to drive in the desired direction (X-axis) (Fig. 9 a).

The same interval of time took the robot velocities to reach their desired reference values, which are generated by position controller as shown in Fig.9b& c. The goal point has a zero error area, since the practical prototype is hard to reach the exact goal point. This zero area is a circle with radius of $5cm$, when the mass center point of the robot reaches this zone, the robot velocities will be zero. From Fig.9b it is clear that goal zone is reached after 20 seconds. The robot trajectory is shown in Fig.9e, where the robot drives around the reference trajectory with error of $\pm 4cm$ till it reaches the goal zone.

As it was mentioned before, the behavior of the C3P depends on its initial variables such as the initial steering angles values, the initial angular velocities and the initial steering angular velocities. This can be shown using a triangle shape as shown in Fig (10e) The experiment is divided into three stages to show the C3P performance in a homing application. Stage number one is to drive from $P_1 = [0m\ 0m\ 0^o]^T$ with initial steering angles or zero degrees to $P_2 = [0m\ 3.5m\ 0^o]^T$. Stage two is driving from P_2 to $P_3 = [3m\ 3.5m\ 0^o]^T$, then the final stage is to drive back to the starting point P_1. The experiment illustrates the performance of the position controller in 3DOFs.

From the steering angles values shown in Fig. 10a, the direction of the C3P can be easily observed, where they are zero during the first stage (driving in Y direction) then the value switch to -90^0 (driving in X direction) then they had the value around -225^o which represents driving in (-X,-Y) direction. But the values of the steering angles are sensed with respect to the C3P frame of co-ordinates. The response of the linear velocities V_x or $\dot{X}$ and V_y or $\dot{Y}$ (Fig. 10-b & c) are sensed with respect to the floor co-ordinates. They show that the output velocities follow the reference signal with delay at the second 17.5 till the second 19 because of the steering angle switching. Oscillations are noticed as well due to the floor friction, steering torques and slippage. These three factors exist in the practical implementation, while the are absence in the inverse solutions and simulations. That is the reason they are common to appear in practical experiments. The C3P trajectory shown in Fig. 10-e illustrates that the robot achieves its goal in the three stages. First it reached point P_2 with ending displacement error of 2 cm, and stage three with error of 5 cm, finally it reached the starting point again with error of 20 cm. The robot drove for about 11 meter with relative error of 1.8%, which is

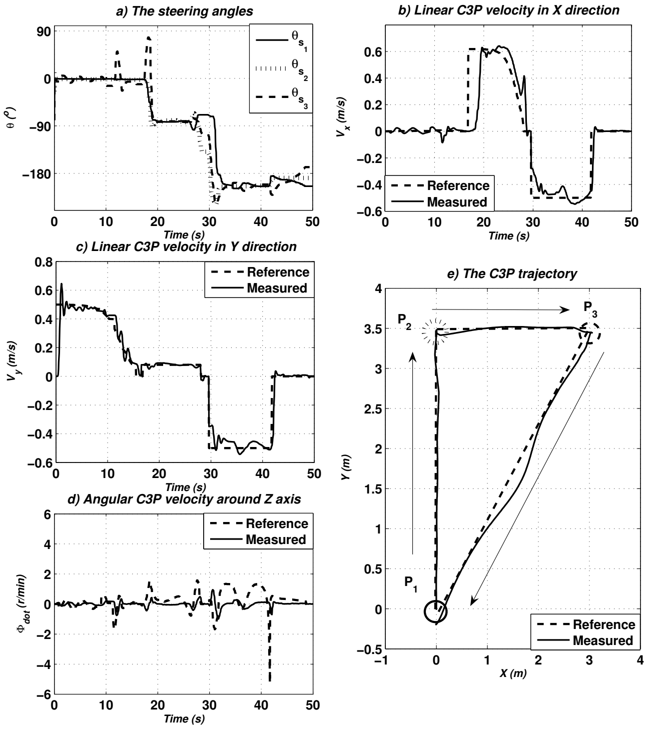

Fig. 10. C3P behavior for driving in triangle shape

acceptable with the absence of friction, slippage and steering torque modeling.

Fig.11 shows the robot results for driving in double circle. This experiments shows the quality of the robot velocity controller for curvature motion.

For smooth behavior of the used prototype platform, the robot will drive in the Y direction with respect to the robot frame of co-ordinates. Therefore the robot frame velocities are $\dot{p}_{C3P} = [0\ 0.3(m/s)\ 0.2(r/min)]^T$ and floor frame velocities are $\dot{p}_{Floor} = [0.3\cos(\Phi)(m/s)\ 0.3\sin(\Phi)(m/s)\ 0.2(r/min)]^T$. Such input robot velocities will drive the robot in a complete circle, when the C3P finishes the circle the $\dot{\Phi}$ will be $-0.2(r/min)$ to drive in the other circle as shown in Fig.11). Such step change shows the controller response in the rotational velocity output.

V. Conclusion

This paper presents the practical implementation of the holonomic wheeled mobile robot C3P. The C3P has the problem of singularity, which is the un-ability of driving in any direction parallel to the wheels driving axes when all the the steering angles have the same value. This paper illustrates the fusion between the inverse kinematics and inverse dynamics solution to be used in the velocity and position control loops. The based controller showed its feasibility in the velocity control loop and to reach the goal co-ordinates of the position controller.

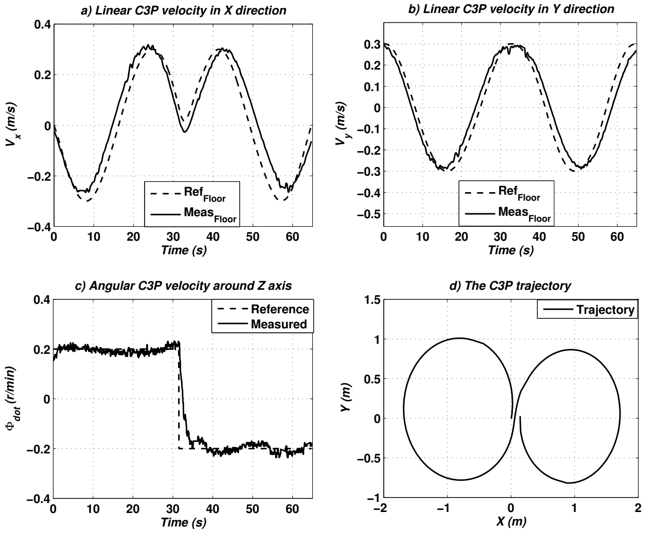

Fig. 11. C3P results for driving in ∞ shape

References

[1] M. Jung, J. Jang and J. Kim, "Development of RoboSot Category Robot Soccer Team," *Dept. of Electrical Engineering and Computer Science, KAIST, Taejon-shi*, pp.305-701.

[2] M. Souma, A. Alhaj and E. L. Hall, "Designing and simulation a motion Controller for a Wheeled Mobile Robot Autonomous Navigation." *Proc. of SPIE Intelligent Robots and Computer Vision XXI: Algorithms, Techniques, and Active Vision*,Vol. 6006. 2005.

[3] L. Huang, Y. S. Lim, David Li and E. L. Teoh, "Desgin and Analysis of a four-wheel Omnidirectional Mobile Robot." *2nd international conference on Autonomous Robots and Agents*,New Zealand. 2004.

[4] Essam Badreddin, "A Hybrid Control Structure for a Robot Soccer Player." *Proceedings of world Automation Congress*,Maui, 2000.

[5] X. Yun, and N. Sarkar, "Unified formulation of robotic systems with holonomic and nonholonomic constrains," *IEEE Trans. on robotics and automation*, Vol.12, 1998, pp.640-650.

[6] J. Sitte and P. Winzer, "Mastering complexity in robot design." *Proceedings of international conference on intelligent robots and systems*,Japan, 2004.

[7] P. Xu, "Mechatronics Design of a Mecanum Wheeled Mobile Robot." *Cutting Edge Robotics*,Germany,pp:784 2005.

[8] A. El-Shenawy,A. Wagner, and E. Badreddin . "Solving The Singularity Problem for a Holonomic Mobile," *4th IFAC-Symposium on Mechatronic Systems (MECHATRONICS 2006)* , Germany, 2006.

[9] A. El-Shenawy,A. Wagner, and E. Badreddin . "Dynamic Model of a Holonomic Mobile Robot with Actuated Caster Wheels." *The 9th International Conference on Control, Automation, Robotics and Vision, ICARCV* , Singapore, 2006.

[10] A. El-Shenawy,A. Wagner, and E. Badreddin . "Inverse Dynamic Solution for Holonomic Wheeled Mobile Robot with Modular Actuation." *The European Control Conference, ECC07*, Kos, Greece, 2007.

[11] Y. Peng, and E. Badreddin. "Analyses and Simulation of the Kinematics of A Wheeled Mobile Robot Platform with Three Castor Wheels." *Internal Report Automation Lab, Computer Engineering University of Mannheim*, 2000.

[12] P. Muir. "Modeling and control of wheeled mobile robots," *PhD dissertation Carnegie mellon university* (1987).

[13] A. El-Shenawy,A. Wagner, and E. Badreddin . "Controlling a Holonomic Mobile Robot With Kinematics Singularities." *The 6th World Congress on Intelligent Control and Automation*, China, 2006.

[14] J. Naudet and D. Lefeber, "Recursive Algorithm Based on Canonical Momenta For Forward dynamics of Multibody Systems," *Proceedings of IDETC/CIE 2005*, 2005.