Proceedings of the 2006 IEEE International Conference on Robotics and Automation
Orlando, Florida - May 2006

Motion Control and Sensing Strategy for a Two-axle Compliant Framed Wheeled Modular Mobile Robot

Xiaorui Zhu, *Student Member, IEEE*, Roy Merrell and Mark A. Minor, *Member, IEEE*

***Abstract*— A novel motion control and sensing strategy for a two-axle Compliant Framed wheeled Modular Mobile Robot (CFMMR) is studied in this paper. This type of wheeled mobile robot uses rigid axles coupled by compliant frame modules to provide both full suspension and enhanced steering capability without additional hardware. The proposed control and sensing system is developed by combining a curvature-based kinematic controller, a robust dynamic motion controller and a sensor fusion algorithm incorporating relative position sensors. Experimental results verify improved motion control of posture regulation.**

Index Terms: Motion control and sensing, Posture regulation, CFMMR.

I. Introduction

A cooperative motion control strategy unifying a time-invariant curvature-based kinematic motion control algorithm, a distributed robust dynamic motion control algorithm and a sensor fusion algorithm incorporating relative position sensors is developed, and applied to a final target, a two-axle CFMMR (Compliant Framed wheeled Modular Mobile Robot), Fig. 1. The CFMMR has uniqueness in modularity and simple structure to provide suspension and highly controllable steering capability without any additional hardware. Despite these attributes, the CFMMR provides new challenges in motion planning, dynamic motion control and sensor instrumentation. First the compliant frames provide path curvature constraints in kinematics of the robot [1]. Secondly, the compliant frames encounter large deflections (i.e., post-buckling) and these nonlinear compliance effects are considerable in dynamics of the robot [2]. Thirdly, the compliant coupling also makes sensor instrumentation complex [3].

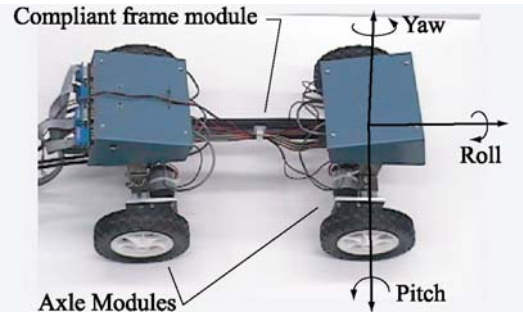


Fig. 1. Two-axle CFMMR experimental configuration.

In recent years, much attention has been paid to cooperative motion control of multiple mobile robots [4-8]. Some of them only focus on motion planning and coordination issues or only consider motion planning and sensor architecture ignoring dynamic motion control [4, 6, 8]. Others only focus on dynamic motion control and coordinated force control without considering motion planning and sensor issues [5, 7, 9]. But none of them consider the combination of motion planning, dynamic motion control and complex sensor fusion. Therefore in this paper a unified motion control strategy is proposed to solve multiple issues for the two-axle CFMMR, Fig. 2. A curvature-based *kinematic motion controller* should be realized to deal with bounded path curvature and specify individual axle motion such that the CFMMR executes the desired net motion. These individual axle motions then provide real-time reference inputs to a *dynamic motion controller*, which deals with nonlinear interaction forces, through a cascade connection. A *sensor fusion algorithm* incorporating the relative position sensors is then realized to avoid aggressive interaction forces and compensate the drift of traditional local sensing techniques and inaccuracy of global localization techniques. Hence, the CFMMR control system consists of three parts; (1) *kinematic motion control*, (2) *dynamic motion control* and (3) *sensor fusion algorithms*. Furthermore, this unified motion control strategy can be extended to any cooperative wheeled mobile robot.

The main contributions of this paper involve a cooperative motion control and sensing structure and a new sensor fusion algorithm incorporating the relative position sensors to deal with multiple sensor instrumentation. The proposed motion control and sensing structure unifies a kinematic motion controller, a dynamic motion controller and a sensor algorithm, which are independently developed to resolve their own kinematics, dynamics and sensor fusion issues.

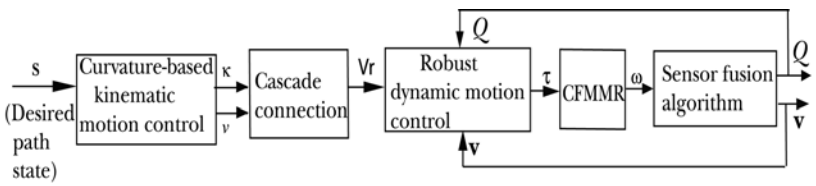


Fig. 2. The motion control diagram on CFMMR

0-7803-9505-0/06/$20.00 ©2006 IEEE

The structure of the paper follows. An overall motion control and sensing strategy is discussed for a two-axle CFMMR in Section II. A *kinematic motion controller*, a cascade connection and a *dynamic motion controller* are presented in Section III-V. The control laws are based on the results of [10]. A *sensor fusion algorithm* is proposed in Section VI. The performance of the unified motion control and sensing strategy on a two-axle CFMMR is evaluated in Section VII. Concluding remarks and future work are described in Section VIII.

II. BACKGROUND

A. Generic Modeling Structure

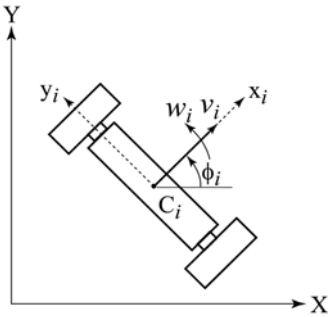


Fig. 3. The i^{th} axle module.

Consider the i^{th} axle of a two-axle cooperative modular unicycle-type wheeled robot, Fig. 3. Let us define a fixed global reference frame $G(X,Y)$ and moving frames $f_i(x_i, y_i)$ attached to the points C_i at the midpoint of the i^{th} axle, where $i = 1,2$, Fig. 4. At any instant, the i^{th} axle module is rotating about the IC (Instantaneous Center), such that the IC's projections onto the x_i axes define point C_i at the midpoint of each axle. A module configuration vector $q_i = [x_i \quad y_i \quad \phi_i]$ is then attached to this point for each axle. Then we have the dynamic model as:

$$\begin{aligned} &\mathbf{M}_i(q_i)\ddot{q}_i + \mathbf{V}_i(q_i,\dot{q}_i)\dot{q}_i + \mathbf{F}_i(\dot{q}_i) + \mathbf{G}_i(q_i) + \tau_{d,i} \\ &+\mathbf{F}_{K,i}(q_i, q_{i\pm1}) = \mathbf{E}_i(q_i)\tau_i - \mathbf{A}_i^T(q_i)\lambda_i \end{aligned} \quad (1)$$

where $\mathbf{M}_i(q_i) \in R^{3\times3}$ is a symmetric, positive definite inertia matrix for the i^{th} axle module. $\mathbf{V}_i(q_i,\dot{q}_i) \in R^{3\times3}$ is the centripetal and Coriolis forces, $\mathbf{F}_i(\dot{q}_i) \in R^{3\times1}$ denotes the friction, $\mathbf{G}_i(q_i) \in R^{3\times1}$ is the gravitational vector, $\tau_{d,i}$ denotes bounded unknown disturbances including unstructured un-modeled dynamics, $\mathbf{E}_i(q_i) \in R^{3\times2}$ is the input transformation matrix, $\tau_i \in R^{2\times1}$ is the input torques, and $\lambda_i \in R^{1\times1}$ is the vector of nonholonomic constraint forces. $\mathbf{A}_i(q_i) \in R^{1\times3}$ is the global matrix associated with the nonholonomic constraints. $\mathbf{F}_{K,i}(q_i, q_{i\pm1}) \in R^{3\times1}$ represents the compliant frame forces which is related to the geometric constraints due to the flexible beam.

According to the nonholonomic and modular interaction constraints, we have the kinematic model,

$$\mathbf{A}_i(q_i)\dot{q}_i = \mathbf{0} \quad (2)$$

$$K(q_i, q_{i\pm1}) = 0 \quad (3)$$

where (2) is the nonholonomic constraint for traditional unicycle-type robots and (3) represents the additional kinematic (geometric) constraints.

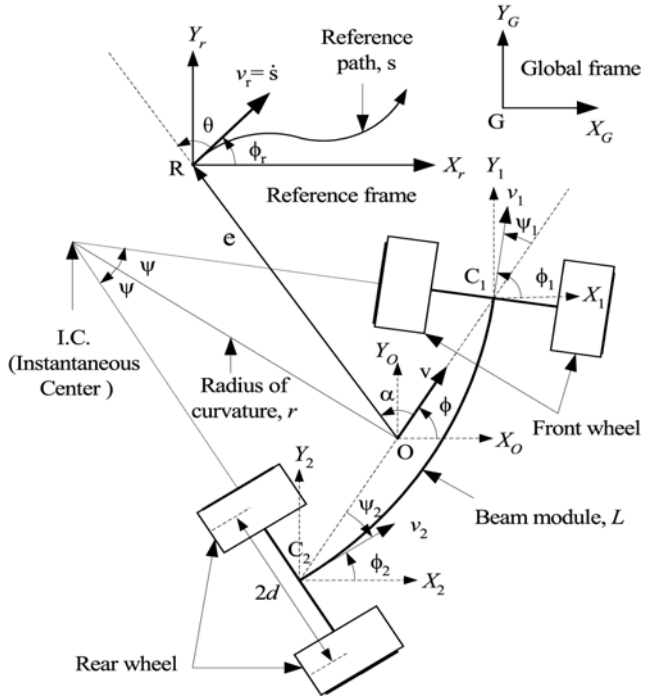


Fig. 4. The two-axle CFMMR.

B. Motion Control and Sensing Strategy

Given the two-axle CFMMR, the control objective is to solve multiple navigation problems using a general approach. These include posture regulation, path following and trajectory tracking.

Let us analyze the target system first. Compared to the traditional unicycle-type wheeled mobile robot, the kinematics of the CFMMR has geometric constraints in addition to nonholonomic constraints. The dynamics also has complex compliant frame forces. These forces are quite dependent on the ability of the measurement system to predict relative axle posture, and thus the data fusion and instrumentation systems must be modified to improve relative position sensing.

This proposed motion control structure, Fig. 2, has several benefits for the mentioned system. First, the complicated control problem is divided into three simpler control issues. Secondly, each controller or algorithm is designed independently. This feature specially benefits large complex systems where team work may be mandatory and parallel task completion will improve efficiency. Thirdly, by independently changing each controller or algorithm, the performance of the whole system can easily be improved and adapted to different target tasks and environments.

To evaluate our proposed strategy, in the following sections, each component is derived for posture regulation, path following and trajectory tracking on the two-axle CFMMR.

III. Centralized Kinematic Motion Control

A kinematic motion controller was proposed in [10], which can be applied for posture regulation, path following and trajectory tracking with bounded curvature.

A two-axle CFMMR can be efficiently steered via a reduced equivalent kinematic model [1], which is derived from front and rear axle relative steering angles $\psi_1=-\psi_2$, Fig. 4. Thus, the net position and orientation of the robot is described by an equivalent posture attached to point O located at the center of $\overline{C_1C_2}$.

Using the circular path manifold and the Lyapunov techniques, the curvature-based time invariant control law was developed as,

$$v=\frac{\left\{\begin{array}{l} k_1 e\sqrt{\zeta-\cos 2\theta}\,\tanh(e-r\sqrt{2}\sqrt{\zeta-\cos 2\theta}) \\ +v_r e\cos\theta\sqrt{\zeta-\cos 2\theta}+v_r r\sqrt{2}\sin 2\theta\left(\sin\theta+\kappa_r e\right)\end{array}\right\}}{e\sqrt{\zeta-\cos 2\theta}+r\sqrt{2}\sin 2\theta\sin\alpha} \quad (4)$$

$$\kappa=\frac{\dot{\phi}}{v}=\frac{k_2\tanh(\theta+\alpha)+\frac{2}{e}\left(v\sin\alpha-v_r\sin\theta\right)-v_r\kappa_r}{v}\,.$$

where r is the radius of a circular *path manifold*. $\zeta=1+\varepsilon$ and ε is a sufficiently small perturbation. ϕ is the heading angle of the coordinate frame R. The variable v represents the velocity of the coordinate frame O moving in a relative to the global frame G. The subscript r denotes the reference frame. That is, v_r and ϕ_r are the reference velocity and the reference heading- angle, respectively.

IV. Cascade connection

The cascade connection was developed in [10] to generate the reference velocities $\mathbf{v}_r$ of each axle based on the simplified kinematics to feed into the robust *dynamic motion controller*, Fig. 2. The linear and angular velocities of each axle can be described from the linear velocity v and path curvature κ of the center posture, O. By the assumption of pure bending, where $\psi = \psi_1 = -\psi_2$, ψ may be solved numerically using the expression for the path radius and curvature of point O,

$$\kappa=\frac{1}{r}=\frac{2\psi}{L\cos\psi}\,, \quad (5)$$

where L is the frame length. Then the linear and angular velocities of each axle, v_i, and ω_i are,

$$\begin{aligned} v_i &= \frac{v}{\cos\psi}+\frac{(-1)^i}{6}L\psi\dot{\psi} \quad \begin{cases} i=1 \text{ for front axle} \\ i=2 \text{ for rear axle}\end{cases} \\ \omega_i &= \dot{\phi}+(-1)^{i-1}\dot{\psi}\,. \end{aligned} \quad (6)$$

where d is the half length of an axle and r_w is the wheel radius.

V. Dynamic Motion Control

A distributed robust dynamic motion controller was proposed in [10], which can compensate the uncertainties caused by the complex interaction forces and the other dynamic disturbances.

Using the regular backstepping form and the Lyapunov technique, the distributed robust control law was developed as [10],

$$\tau_i=-(\mathbf{S}_i^{\,T}\mathbf{E}_i)^{-1}K_i\mathbf{e}_{c,i}\left\|\xi_i\right\|^2\,. \quad (7)$$

Here

$$\xi_i^{\,T}=\left\{\left\|\mathbf{v}_{c,i}\right\|,\left\|\mathbf{v}_i\right\|,\left\|\mathbf{v}_{r,i}\right\|,\left\|\dot{\mathbf{v}}_{r,i}\right\|,\left\|\mathbf{v}_i\right\|\left\|\mathbf{v}_i\right\|,1,\left\|\mathbf{F}_{K,i}(q_i,q_{i\pm1})\right\|\right\} \quad (8)$$

$$\mathbf{v}_{c,i}=\begin{bmatrix} v_{r,i}\cos e_{\phi,i}+k_{X,i}e_{X,i} \\ \omega_{r,i}+k_{Y,i}v_{r,i}e_{Y,i}+k_{\phi,i}v_{r,i}\sin e_{\phi,i}\end{bmatrix} \quad (9)$$

$$e_i=\begin{bmatrix} e_{X,i} & e_{Y,i} & e_{\phi,i}\end{bmatrix}^T=\mathbf{R}_{\phi,i}\left(q_{r,i}-q_i\right) \quad (10)$$

where ξ_i is a known, positive definite vector. $K_i=\begin{bmatrix} K_{1,i} & 0 \\ 0 & K_{2,i}\end{bmatrix}$ is the matrix control gain and $K_{1,i}$, $K_{2,i}$ are positive constants. The matrix $k_i=\begin{bmatrix} k_{X,i} & k_{Y,i} & k_{\phi,i}\end{bmatrix}$ is composed of positive constants and $q_{r,i}$ is the reference vector for the i^{th} axle. The tracking error was proven globally uniformly bounded and the bounds become arbitrarily small by increasing the control gain K_i. On the other hand, the larger control gains increase the control effort and then may cause saturation or chattering.

Note that the reference velocity vector is included in the control input since the motion controller could provide time varying reference velocities. The compliant frame force $\mathbf{F}_{K,i}(q_i,q_{i\pm1})$ is also taken into consideration by the controller. As the configuration and environment become more complicated, however, the controller without the compliant frame force compensation may be desirable (i.e., $\mathbf{F}_{K,i}(q_i,q_{i\pm1})$ could be considered as a disturbance).

VI. Sensor Fusion Algorithm

A. Relative Position Sensor

The unique aspect of the sensor fusion algorithm is the Relative Position Sensor (RPS) used to improve the positional tracking between the front and rear axles. The flexible frame allows the robot to control its shape and move over or around many obstacles. Vital to this advanced control is the accurate knowledge of the relative position and orientation of the axles. Even the simplest of navigational maneuvers will result in antagonistic forces exerted between the axles if the error in relative position and orientation between the axles is too large. Therefore, it is of importance that the posture estimation drift between the two axles be eliminated. The RPS accomplishes this feat with a package small and light enough to sit on the robot without limiting its full range of motion or impeding its travel in any way.

The following assumptions are used in defining the theoretical basis for the RPS. First, small angle

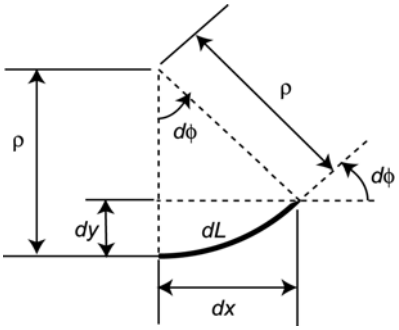


Fig. 5. Curvature integration diagram.

approximations cannot be used because the beam will be subjected to extreme bending conditions. Secondly, the robot will operate in a planar space. This means that the RPS will only need to detect the relative x, y, and ϕ of one axle to the other. Along with this assumption is the caveat that the ultimate goal is a robot that can operate in rough terrain. Thirdly, the sensor will consist of a series of strain bridges placed at known locations along the length of the beam.

The starting point for the derivation of the RPS algorithm is the strain at discrete locations along the beam, ε_1, ε_2, ..., ε_n. A smooth function is then derived based upon the strain gauge data that will interpolate the strain along the length of the beam. Due to their relative ease of calculation, the polynomial interpolation was chosen. For this interpolation it is desired to determine the coefficients of a polynomial that relates strain (ε) to location along the beam (x) where the polynomial exactly intersects the strain reported by each strain gauge at the location of each strain gauge. This polynomial will have the form of equation (11), where n is equal to the number of discrete locations along the beam at which the strain is known.

$$\varepsilon(x) = a_1 + a_2 x + \ldots + a_{n-1} x^{n-2} + a_n x^{n-1} \qquad (11)$$

The coefficients in (11) can be determined by solving the system of linear equations in (12) where l_1, … l_n are the discrete strain gauge locations along the beam.

$$\begin{bmatrix} 1 & l_1 & \cdots & l_1^{n-2} & l_1^{n-1} \\ 1 & l_2 & \cdots & l_2^{n-2} & l_2^{n-1} \\ 1 & l_3 & \cdots & l_3^{n-2} & l_3^{n-1} \\ \vdots & \vdots & & \vdots & \vdots \\ 1 & l_n & \cdots & l_n^{n-2} & l_n^{n-1} \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \vdots \\ \varepsilon_n \end{bmatrix} \qquad (12)$$

To calculate the posture of one axle relative to the other, the relationship between strain and curvature (κ) and the radius of curvature (ρ) in (13) is used, where y is the inflection point for the strain internal to the beam.

$$\frac{1}{\rho(x)} = \kappa(x) = \frac{\varepsilon(x)}{y(x)} \qquad (13)$$

A small segment of the beam (dL), Fig. 5, is examined assuming the segment dL is chosen sufficiently small so that the change in curvature over its length is negligible. Therefore, the dx, dy and $d\phi$ are described as:

$$\begin{aligned} d\phi &= dL / \rho \\ dx &= \rho \sin(d\phi) \\ dy &= \rho(1 - \cos(d\phi)) \end{aligned} \qquad (14)$$

Then the equation (14) is applied to determine dx, dy, and $d\phi$ for each segment. Once this is done, the $d\phi$ from each segment is summed to calculate the total angular difference between the endpoint and root of the beam. This application assumes that each segment of the beam starts at an angular orientation of 0°. However, each segment starts with an angular orientation equal to the sum of $d\phi$'s from all previous segments. In order to calculate the endpoint x and y relative to the beam's root, each vector (dx, dy) from each segment must be rotated into alignment with the previous segment. Equation (15) shows how this is accomplished where the vector (dx_r, dy_r) is the rotated position vector and ϕ_p is the sum of each dϕ from all of the previous segments.

$$\begin{bmatrix} dx_r \\ dy_r \end{bmatrix} = \begin{bmatrix} \cos(-\phi_p) & \sin(-\phi_p) \\ -\sin(-\phi_p) & \cos(-\phi_p) \end{bmatrix} \begin{bmatrix} dx \\ dy \end{bmatrix} \qquad (15)$$

After this rotation, the dx_r and dy_r from each segment can be summed to calculate the total change in x and y over the length of the beam. Note that the small rigid sections at the endpoints of the beam should also be included, where $d\phi$ is always zero. After this process of piecewise integration is completed, the result is the calculation of the relative pose (x, y, and ϕ) of the center point of one axle with respect to the other.

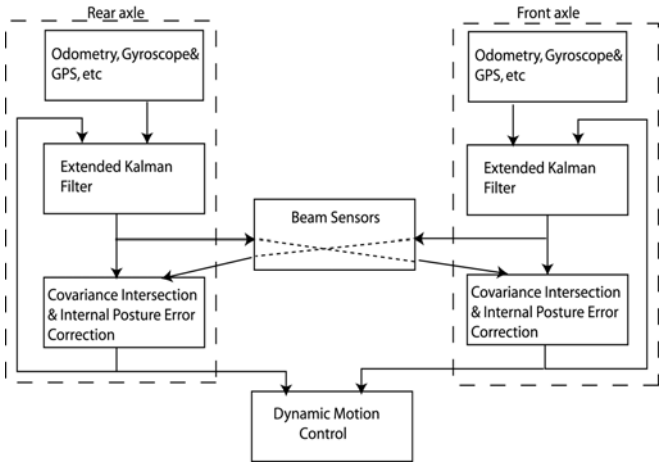


Fig. 6. Sensor fusion algorithm block diagram

B. Axle Level Data Fusion

Another unique aspect of this sensor fusion algorithm is the combination of data fusion methods to incorporate the data from the RPS, and other sensors that the robot may use, Fig. 6. This research uses an Extended Kalman Filter (EKF) to fuse data sources on an individual axle module, such as wheel encoders, axle gyroscopes, accelerometers, GPS, etc. When multiple axle modules are joined with RPS instrumented flexible frame members, a second tier of data fusion is necessary to facilitate data transfer from the RPS to the adjoining axle modules and between the joined axle modules themselves. The Covariance Intersection (CI) filter strategy is used for the second tier data fusion module. The large degree of unknown correlation between data sources as relative posture information is passed back and forth between axle modules and the RPS makes this module of the

data fusion algorithm particularly challenging. In order for the EKF to be consistent in its state estimation it is required that the correlation between data sources be known. The CI filter assumes complete correlation and is therefore consistent in its estimation in the face of uncertain correlations [11]. All of this must be accomplished in real time without impeding the kinematic and dynamic motion controllers [10]. A third component of the data fusion structure is the incorporation of Borenstein's Internal Posture Error Correction (IPEC) algorithm [12]. The IPEC algorithm has been augmented with the ability to propagate the variance information required by the two data fusion modules and inserted into the upper tier of the data fusion architecture. This combination of data fusion techniques is a new approach that is required by the modular nature of the robot and the unique sensing capability of the RPS.

In the algorithm proposed here, the purpose of the EKF is to fuse the data from the axle level position sensors with the state predictions of the system model in order to provide optimal estimation of the true system states. The EKF requires a state model including the uncertainties inherent to the model, and a measurement model including the noise inherent to the sensors as [13]:

$$\begin{aligned} x_k &= a\left(x_{k-1}, u_{k-1}, w_{k-1}, T\right) \text{ where } w_{k-1} \sim N\left(0, q_{k-1}\right) \\ z_k &= h\left(x_k, v_k, T\right) \text{ where: } v_k \sim N\left(0, r_k\right) \end{aligned} \tag{16}$$

where $N(i, j)$ notation means that u and v are normal distributions with a mean of i and a variance of j. The u and v are assumed uncorrelated.

The EKF has two primary components, the time update and the measurement update. The time update is to predict the state and covariance values at the current time step from the state estimate at the previous time step as:

$$\begin{aligned} \hat{x}_k^- &= a\left(\hat{x}_{k-1}, 0, 0, T\right) \\ P_k^- &= AP_{k-1}A^T + Q \end{aligned} \tag{17}$$

The measurement update involves calculating a Kalman gain K, then obtaining the final state estimate and finally getting the final covariance estimate for the time step as:

$$\begin{aligned} K_k &= P_k^- H^T (HP_k^- H^T + R)^{-1} \\ \hat{x}_k &= \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \\ P_k &= (I - K_k H)P_k^- \end{aligned} \tag{18}$$

In (16)-(18) x is the states of the system. The ^ means it is an estimate of the true states. The $^-$ indicates that it is a prediction of the states at the current time step. The subscript $_k$ indicates the current time step and $_{k\text{-}1}$ indicates the previous time step. The function a is the nonlinear model of the system. The u is the input to the system, and T is the time step. The P and Q are the covariance matrices associated with the system states and the uncertainties in the state model. The A is the linearized equation of a. The H matrix is a mapping of the states to the measurements and R is the covariance matrix associated with the measurements z_k. In this implementation the 7 axle states are shown as:

$$x = \begin{bmatrix} x & y & \phi & \mathbf{v} & \dot{\phi} & \dot{\mathbf{v}} & \ddot{\phi} \end{bmatrix}^T \tag{19}$$

where x, y, and ϕ are the axle's global position and orientation. The $\mathbf{v}$ and $\dot{\phi}$ are the translational and rotational velocity of the axle. The two remaining states are the translational and rotational acceleration of the axle.

VII. Performance Evaluation

A. Methods and Procedures

The cooperative motion control strategy for the two-axle CFMMR was simulated in Matlab® and Simulink® to adjust the control gains for better performance but the results are not shown here. Experiments were conducted on a two-module CFMMR experimental platform, Fig. 1, at the University of Utah.

The robot is controlled via tether by a dSpace™ 1103 DSP board and an external power supply. Each wheel is actuated by a DC motor where the real-time position of each wheel is detected by an encoder. Odometry and five relative position sensors are used in the sensing system. Two 7.2v RC car batteries are mounted on the rear axle to provide the power supply for the amplifying circuit board of the sensing system. Second order filters were initially used after the encoder signals to filter noise. But, they caused marginally stability due to phase lag. Hence encoder signals are directly used to provide the position and velocity input to the sensor fusion algorithm. The experiments were operated on a smooth, flat, high traction carpet surface for posture regulation to prove the feasibility of the proposed control and sensing strategy and evaluate the preliminary performance of the robot.

B. Results and Discussion

For all the experiments, the initial condition of the middle point O is $\begin{bmatrix} x & y & \phi \end{bmatrix} = \begin{bmatrix} -1.342m & -1.342m & 0^o \end{bmatrix}$. Fig. 7 shows the robot paths during the posture regulation where the white lines represent the string grids and the black lines are from the odometry results. The solid black line represents the odometry position of the middle point O and the dashed black lines represent the odometry positions of the wheels. Fig. 8 shows the posture errors according to odometry. Table 1 shows the actual measurements of the final posture error data and respective standard deviations for the control structure with RPS and without RPS. Note that there is additional hardware (batteries and amplifying circuit boards) on the rear axle for the control structure with RPS. Hence, this structure has significant disturbance by the mass difference between the front and rear axles than the control structure without RPS.

Fig. 7 predicts that the two-axle CFMMR achieves the posture regulation well by both odometry data and the actual video snapshots. According to odometry, Fig. 8 predicts the final position error e is 0.015m (1.5cm); the final orientation errors θ and α are 0.035rad (2.0°) and 0.18rad (10.3°)

respectively. Table 1 predicts the robot posture regulation is improved for the system with RPS even though in the case of the larger mass disturbance. The position error e is decreased by 10%. The orientation errors θ and α are decreased by 58% and 52% respectively. Therefore, the motion control and sensing strategy proposed in this paper is efficient to achieve the desired locomotion of the two-axle CFMMR.

VIII. Conclusions

This paper introduces a novel motion control and sensing strategy combining a kinematic motion controller, a dynamic motion controller and a sensor algorithm with the Relative Position Sensors for a two-axle Wheeled Compliant Framed Modular Mobile Robots. Experimental results demonstrate the efficiency and robustness of the proposed technique. This motion control and sensing strategy is generally applicable to other cooperative modular unicycle-typed wheeled mobile robots. Future work will focus on the behavior of the two-axle CFMMR in more complicated terrain and motion control of more than two CFMMR modules.

References

[1] B. W. Albiston and M. A. Minor, "Curvature based point stabilization for compliant framed wheeled modular mobile robots," in *IEEE ICRA*, Sep 14-19 2003, Taipei, Taiwan, pp. 83-89 2003.

[2] X. Zhu, M. A. Minor, and S. Park, "Distributed robust control of compliant framed wheeled modular mobile robots," *ASME J. of Dyn. Syst., Meas., and Contr., In press*, 2006.

[3] R. Merrell and M. A. Minor, "Internal posture sensing for a flexible frame modular mobile robot," in *IEEE ICRA*, Sep 14-19 2003, Taipei, Taiwan, pp. 452-457 2003.

[4] M. Kumar and D. P. Garg, "Sensor-based estimation and control of forces and moments in multiple cooperative robots," *Transactions of the ASME. Journal of Dynamic Systems, Measurement and Control*, vol. 126, pp. 276-83, 2004.

[5] Y. Hirata, Y. Kume, T. Sawada, Z.-D. Wang, and K. Kosuge, "Handling of an object by multiple mobile manipulators in coordination based on caster-like dynamics," in 2004 IEEE ICRA, Apr 26-May 1 2004, New Orleans, LA, United States, pp. 807-812 2004.

[6] C. P. Tang, R. Bhatt, and V. Krovi, "Decentralized kinematic control of payload transport by a system of mobile manipulators," in 2004 IEEE ICRA, Apr 26-May 1 2004, New Orleans, LA, United States, pp. 2462-2467 2004.

[7] A. Rodriguez-Angeles and H. Nijmeijer, "Mutual synchronization of robots via estimated state feedback: a cooperative approach," *IEEE Transactions on Control Systems Technology*, vol. 12, pp. 542-54, 2004.

[8] A. G. O. Mutambara and H. E. Durrant-Whyte, "Estimation and control for a modular wheeled mobile robot," *IEEE Transactions on Control Systems Technology*, vol. 8, pp. 35-46, 2000.

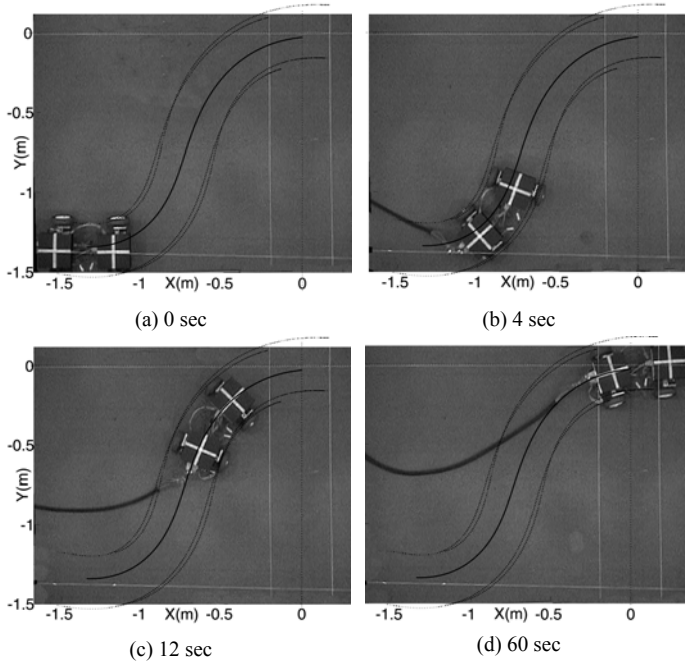
[9] K. Kosuge and M. Sato, "Transportation of a single object by multiple decentralized-controlled nonholonomic mobile robots," in 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'99): Human and Environment Friendly Robots whith High Intelligence and Emotional Quotients', Oct 17-Oct 21 1999, Kyongju, South Korea, pp. 1681-1686 1999.

[10] X. Zhu, Y. Kim, and M. A. Minor, "Cooperative distributed robust control of modular mobile robots with bounded curvature and velocity," in 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Monterey, California 2005.

[11] S. J. Julier and J. K. Uhlmann, "A non-divergent estimation algorithm in the presence of unknown correlations," in Proceedings of 16th American CONTROL Conference, 4-6 June 1997, Albuquerque, NM, USA, pp. 2369-73 1997.

[12] J. Borenstein, "Internal correction of dead-reckoning errors with a dual-drive compliant linkage mobile robot," *Journal of Robotic Systems*, vol. 12, pp. 257-273, 1995.

[13] T. D. Larsen, K. L. Hansen, N. A. Andersen, and O. Ravn, "Design of Kalman filters for mobile robots; evaluation of the kinematic and odometric approach," in Proceedings of the 1999 IEEE International Conference on Control Applications, 22-27 Aug. 1999, Kohala Coast, HI, USA, pp. 1021-6 1999.



(a) 0 sec (b) 4 sec

(c) 12 sec (d) 60 sec

Fig. 7. Robot paths during posture regulation on carpet.

Table 1. Experimental posture error.

	$e \pm \sigma e$ *(cm)*	$\theta \pm \sigma\theta$ *(deg)*	$\alpha \pm \sigma\alpha$ *(deg)*
With RPS	9.5 ± 2.1	-13.3 ± 14.2	-21.6 ± 14.9
Without RPS	10.6 ± 3.4	-31.7 ± 9.4	-45.4 ± 12.2

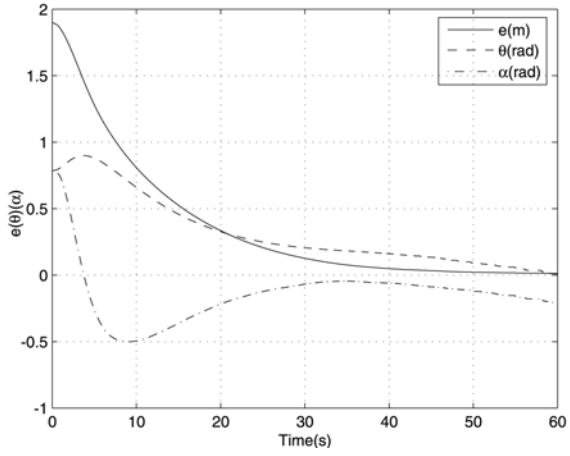


Fig. 8. Posture errors according to odometry.