

Force Guided Docking Control of an Omnidirectional Holonomic Vehicle and its Application to Wheelchairs

by

Stephen A. Mascaro

B.S., Mechanical Engineering
Clarkson University, 1995

B.A., Physics
Houghton College, 1995

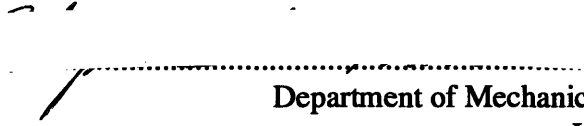
Submitted to the Department of Mechanical Engineering
in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Mechanical Engineering

at the
Massachusetts Institute of Technology

February 1997

© 1997 Massachusetts Institute of Technology
All rights reserved

Signature of Author ...


.....
Department of Mechanical Engineering
January 9, 1997

Certified by

.....
Haruhiko H. Asada
Professor of Mechanical Engineering
Advisor

Accepted by

.....
Ain A. Sonin
Chairman, Department Committee on Graduate Students

MASSACHUSETTS INSTITUTE
OF TECHNOLOGY

APR 16 1997

LIBRARIES

Eng.

Force Guided Docking Control of an Omnidirectional Holonomic Vehicle and its Application to Wheelchairs

by

Stephen A. Mascaro

Submitted to the Department of Mechanical Engineering
on January 9, 1997 in partial fulfillment of the
requirements for the Degree of Master of Science in
Mechanical Engineering

ABSTRACT

A new hybrid wheelchair/bed system for assisting the bedridden has been developed and tested. The powered wheelchair docks with a stationary bed and reconfigures to a chair to provide full mobility assistance by eliminating the need to transfer the bedridden person from bed to chair. Furthermore, the chair can dock directly and automatically with a toilet to eliminate the need to change seating. The chair is driven by an omnidirectional holonomic vehicle with ball wheel mechanisms. The omnidirectional and precise dead reckoning characteristics of the vehicle provide the chair with superior maneuverability and ability to navigate in closely confined environments. The holonomic nature of the vehicle adds to this the ability to dock easily and precisely against a fixture using force guided control.

A clear set of functionalities were defined for the wheelchair. The omnidirectional vehicle was mechanically redesigned to ensure stability without sacrificing performance. A new control system was developed, allowing the chair to be driven both manually and automatically. An instrumented bumper system was designed and installed on the vehicle. A force-guided docking control method was developed and tested, consisting of a combination of impedance control based on force feedback from the bumpers and active compliant control through servo stiffness. The vehicle was successfully docked to the bed despite a small clearance ratio as well as large lateral and angular initial errors.

Thesis Supervisor: Haruhiko H. Asada

Title: Professor of Mechanical Engineering

ACKNOWLEDGMENTS

I wish to thank my parents for their unfailing love, support, and instruction. Thanks also to Professor Asada for his guidance and ideas, and to Joe for his many contributions to this project. Thanks to all my friends in Christ for their daily encouragements.

Above all, I thank God my Creator, who has given me the talents and strength to achieve.

II Timothy 3:14-15

“But as for you, continue in what you have learned and have become convinced of, because you know those from whom you learned it, and how from infancy you have known the holy Scriptures, which are able to make you wise for salvation through faith in Christ Jesus.”

TABLE OF CONTENTS

1. INTRODUCTION	6
1.1 MOBILITY ASSISTANCE	6
1.2 FORCE GUIDED DOCKING.....	7
1.3 OVERVIEW	8
2. THE RHOMBUS BED/CHAIR SYSTEM.....	10
2.1 THE RHOMBUS CONCEPT.....	10
2.2 DEVELOPMENT OF THE FIRST PROTOTYPE.....	14
3. THE OMNIDIRECTIONAL HOLONOMIC VEHICLE.....	18
3.1 THE BALL WHEELED VEHICLE	18
3.2 DOCKING WITH A HOLONOMIC VEHICLE	22
4. THE FOUR WHEELED VEHICLE	24
4.1 DESIGN.....	24
4.1.2 <i>Design of Footprint/Wheel Configuration</i>	25
4.1.3 <i>Design of Suspension System</i>	27
4.2 KINEMATICS	30
4.3 CONTROL.....	36
4.3.1 <i>Hardware</i>	36
4.3.2 <i>Manual Control</i>	36
4.3.3 <i>Automatic Control</i>	38
4.4.4 <i>Compliance Control</i>	39
5. FORCE GUIDED DOCKING OF THE VEHICLE.....	43
5.1 TASK DEFINITION	43
5.2 BUMPER DESIGN	45
5.3 DOCKING CONTROL	49
5.4 IMPLEMENTATION.....	56
6. CONCLUSIONS.....	60
6.1 SUMMARY OF PERFORMANCE/FUNCTIONALITY	60
6.2 FUTURE DESIGN OF BUMPER SYSTEM	60
APPENDIX A. CAD DRAWINGS.....	64
APPENDIX B. KINEMATIC DERIVATIONS.....	65
APPENDIX C. CONTROL HARDWARE.....	69
APPENDIX D. C PROGRAMS	70
D.1 JOYSTICK CONTROL PROGRAM.....	70
D.2 DOCKING CONTROL PROGRAM.....	75
REFERENCES	84

FIGURES

2.1.1 a-b	Bed Mode/Reconfiguration	10
2.1.1 c	Chair Mode.....	11
2.1.1 d-e	Toilet Docking.....	11
2.1.2	4 DOF Reconfigurable Chair.....	13
2.2.1	RHOMBUS Prototype.....	14
2.2.2	Bed Mode.....	15
2.2.3	Intermediate Reconfiguration Stage.....	16
2.2.4	Chair Mode.....	16
3.1.1	Ball Wheel Mechanism.....	19
3.1.2	Three Wheeled Configuration.....	20
3.1.3	Original Ball-Wheeled Vehicle.....	21
4.1.1	Four Wheel Configuration.....	26
4.1.2	Suspension System.....	29
4.1.3	Ball Wheeled Vehicle - Bottom View	30
4.1.4	Ball Wheel Mechanism with Suspension	30
4.2.1	Four Wheeled Configuration	31
4.2.2	Vehicle Model	34
4.2.3	Bond Graph Model of Vehicle	35
4.3.1	Block Diagram of Manual Control	37
4.3.2	Block Diagram of Automatic Control	38
5.1.1	Misalignment	44
5.1.2	Instrumented Bumper	44
5.2.1	Schematic of Vehicle/Bumper System	46
5.2.2	Determining Location of Contact Forces	47
5.2.3	Prototype Bumper System	49
5.3.1	Simplified Model for Lateral Dynamics	50
5.3.2	Bond Graph Model of Vehicle/Bumper System	51
5.3.3	Stiffness Control	52
5.3.4	Near Complete Insertion	54
5.3.5	Hybrid Stiffness/Compliant Control	54
5.4.1	Bumper Prototype	56
5.4.2	Results of Stiffness Control Alone	58
5.4.3	Results of Variable Stiffness Control	58
5.4.4	Results of Hybrid Stiffness/Compliance Control	59
A.1	Four Wheeled Vehicle Assembly Drawing	64
B.1	Ball Wheel Geometry	65
B.2	Four Wheeled Configuration	66
B.3	Vehicle Geometry	67
C.1	Schematic of Control Hardware	69

1. INTRODUCTION

1.1 MOBILITY ASSISTANCE

The rapid increase in elderly population is a critical problem faced by today's society. By the early twenty-first century, more than 20 percent of the population will be 65 years old or higher in many of the developed countries. Effective technologies for elder care are badly needed to cope with this social crisis predicted in the near future. The traditional elderly care at hospitals and nursing homes is costly and limited in capacity. Home-based elder care provided by various home health agencies is significantly lower in costs than institution based care, but the current practice is heavily dependent upon labor-intensive services, which may not be available in the future.

This thesis describes a new technology for home-based elderly care, focusing on mobility assistance for bedridden persons. Caring for the bedridden at home is one of the hardest tasks faced in elder care. More than 80 percent of the elderly currently residing in nursing homes or hospitals are there primarily because they could no longer be treated at home after having been left permanently bedridden. Transferring the bedridden from a bed to a wheelchair is an extremely laborious, physical job, which average people are unable to perform without the use of special equipment. A variety of equipment for lifting the bedridden has been developed and deployed at both hospitals and homes, such as the Hoyer Lift, Trans-Aid, and Ambulift [1]. Most of the hoists, however, are awkward,

uncomfortable, and often even frightening for the elderly. During the transfer, the body is airborne and tends to swing and rotate, causing great embarrassment for the patient.

Instead of trying to improve such existing technologies, we will take a totally different approach to the care of bedridden persons. We will completely eliminate the need for transferring the patient between the bed and the wheelchair by devising a hybrid wheelchair/bed system that serves both as a wheelchair and as a bed. When used as a wheelchair, the patient can take various sitting positions and move around freely within a house. When used as a bed, the patient can lie flat in a commodious space. Moreover, the wheelchair can be docked to a toilet directly and automatically so that the bedridden can use a toilet without changing the seating. This will allow the bedridden to depend less upon caregivers and thereby maintain dignity in their living.

1.2 FORCE GUIDED DOCKING

One of the unique features of this wheelchair is its method of locomotion. Specifically, the wheelchair is mounted upon and driven by a ball-wheeled omnidirectional holonomic vehicle. The omnidirectional and precise dead reckoning characteristics of the vehicle provide the chair with superior maneuverability and ability to navigate in closely confined environments. The holonomic nature of the vehicle adds to this the ability to dock easily and precisely against a fixture using force guided control.

The area of force guided docking or force guided assembly processes, has been a topic of considerable interest for some years. A review of the quasi-static assembly process is

given by Whitney [2]. Rigid part insertion can be aided by supporting the part with springs or other compliant mechanical elements. By modeling the spring and contact forces and the geometry of the peg in hole, the life cycle of the insertion process can be accurately predicted. Studies show that jamming and wedging can be avoided by locating the compliance center near the front of the peg, and by choosing appropriate stiffnesses/compliances. Devices called *remote center compliances* (RCC's) accomplish this passively [4]. Other methods seek to use some form of active stiffness control. Salisbury introduced a method of simulating part compliance through softening the servo position gains [5][6]. An alternative is to use sensors to measure the actual forces and adjust the trajectory of the part accordingly. Examples of this are *stiffness control* [3] and *impedance control* [3][7], the latter being a generalization of the former.

Which of these control methods is best depends upon the specific task which is to be performed and the mechanical system which will be used to perform it. Sometimes, different control methods may be better suited for different stages of the same task. In the case of the omnidirectional holonomic vehicle, we will show how a combination of stiffness control methods can be used to successfully and robustly dock the vehicle into a rigid hole with small clearance.

1.3 OVERVIEW

In Chapter 2 of this thesis, a detailed description of the wheelchair/bed concept is given and early prototypes are shown. Chapter 3 discusses the nature of the original ball

wheel mechanism developed at MIT and describes the unique characteristics and advantages of the ball-wheeled omnidirectional vehicle. A new four-ball-wheeled vehicle was designed to be used with the wheelchair. The design and control of this vehicle is detailed in Chapter 4. Chapter 5 presents a breakdown of the docking task to be performed with the vehicle and describes the force guided control method which was used to successfully dock the vehicle with a bed fixture. Finally, Chapter 6 concludes with a summary of the results of the design and experimentation. Important contributions and areas which require further study are noted.

2. THE *RHOMBUS* BED/CHAIR SYSTEM

2.1 THE RHOMBUS CONCEPT

Figure 2.1.1 shows the schematic of the RHOMBUS system, i.e. a reconfigurable, holonomic, omnidirectional mobile bed with unified seating. The system, consisting of a reclining wheel chair, a U-shaped bed, and a special toilet, allows a bed ridden person to move to a chair as well as to a toilet without assistance. Namely, the bedridden person does not have to change seating when getting in and out of the bed and going to toilet. The wheelchair can be detached from the bed for the transport of the bedridden person, and docked to the bed for sleep. The transition can be made while the patient is lying in the bed. The chair is reconfigurable so that it can be a flat bed or a cushioned seat with a reclining back and a footrest. The wheelchair is narrow enough to go through residential doors and maneuver freely within a crowded room, while the bed is wide enough to prevent the patient from falling out and roomy enough to provide comfort.

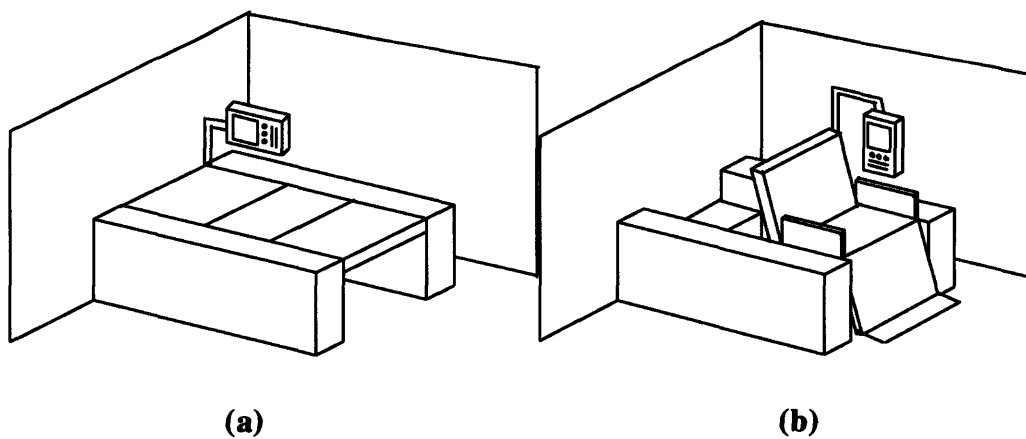


Figure 2.1.1 a-b: Bed Mode/Reconfiguration

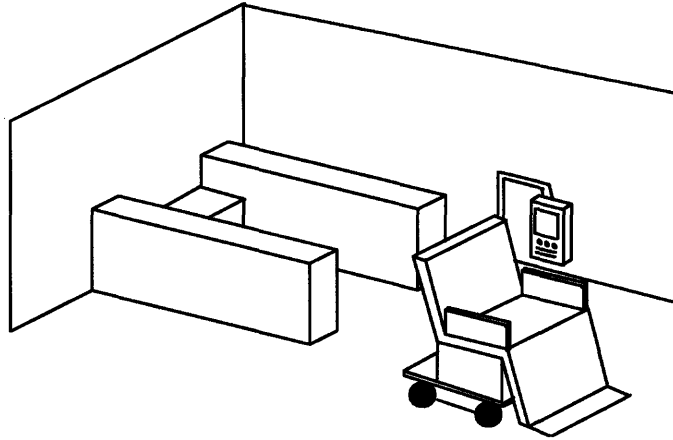


Figure 2.1.1 c: Chair Mode

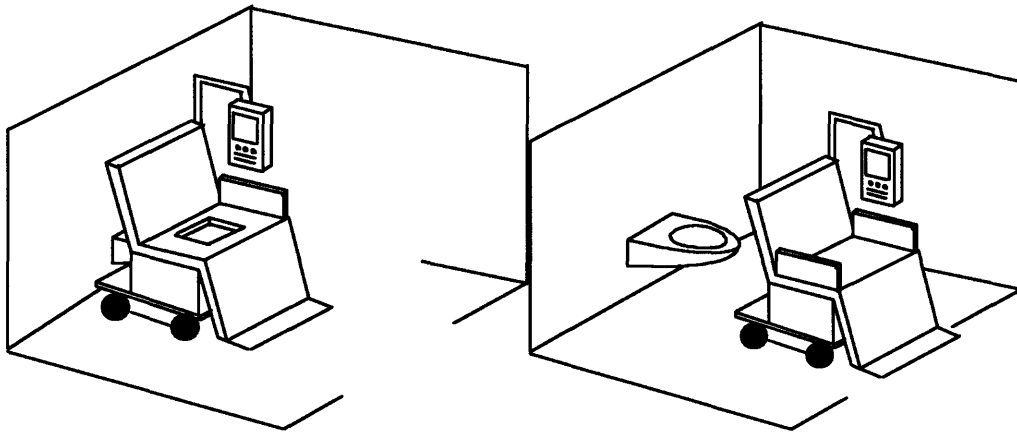


Figure 2.1.1d-e: Toilet Docking

To detach the wheelchair from the bed portion, the back of the chair is first raised from the position shown in Figure 2.1.1a, the whole body of the chair is then slid off from the end of the bed portion as the foot rest is folded down and the arm rests are raised, as shown in Figure 2.1.1b. The chair continues to slide from the bed until completely undocked as shown in Figure 2.1.1c. To move the chair back to the bed configuration, the procedure is simply reversed.

When the bedridden person wants to use a toilet, the wheelchair moves into a bathroom and is docked directly to the toilet body, as shown in Figures 2.1.1c-d. The back of the wheelchair has an open space so that the toilet body can move right beneath the seat of the wheelchair. The toilet is a wall-mounted type that comes out of a bathroom wall clearing the floor. The toilet would be equipped with a shower and dryer for automatic cleaning, while the seat of the wheelchair has a small window which would be engaged with the toilet bowl in such a way as to maintain a clean and sanitary seal. The window would be opened and closed automatically to allow use of the toilet. After the toilet has been used, the wheelchair is undocked from the toilet and can go back to the bedroom or anywhere in the house. Again no transfer of the bedridden person between the wheelchair and toilet is needed.

To dock the wheelchair to the bed and toilet and mate it with them, the wheelchair must be positioned precisely against the fixture. To perform this docking or undocking operation in crowded bedroom and bathroom, the wheelchair must be highly maneuverable and capable of omni-directional motion. To this end we use a holonomic omni-directional vehicle that can move sideways as well as forward and backward and turn left and right with as small as zero radius. The vehicle has three degrees of freedom and moves in an arbitrary direction from an arbitrary configuration. This holonomic mobility allows for accurate docking and tight mating with fixtures, which could not otherwise be performed by traditional nonholonomic vehicles.



Figure 2.1.2: 4 DOF Reconfigurable Chair

The hybrid chair/bed mounted on the vehicle must have the degrees of freedom to recline and raise the back as well as to fold and extend the foot rest all the way from the up-right position to the completely flat position. In addition, the height and slope of the seat must be adjustable depending on the patient's body dimensions. The desired height and slope for the bed configuration may be different from those for the chair configuration, hence the seat height and slope must be adjustable. To meet these requirements, we will use a reconfigurable chair/bed with four active degrees of freedom which has been designed at MIT [8]. A prototype is shown in Figure 2.1.2. The seat and the back can be raised gradually in order to push the back of the infirm person up to an

almost up-right position, should the person wish to stand up. Likewise, the seat and back of the chair can be lowered slowly so that the infirm person can sit down with a minimum of effort. Moreover, by extending the stroke of each degree of freedom, the chair can be used for assisting the elderly in standing and sitting.

2.2 DEVELOPMENT OF THE FIRST PROTOTYPE

Figure 2.2.1 shows the overall view of the prototype RHOMBUS system developed at MIT. The system consisting of a vehicle, chair, bed and toilet was designed for use in standard residential homes with minor remodeling for elderly residents. The vehicle can go through standard doors of 70 cm in width, and can turn within a small space of 120 cm in diameter. The vehicle can go over a step lower than 10 mm and go up a ramp way of up to 10 degrees of gradient. It can be used for wooden, linoleum, and hard carpet floors.

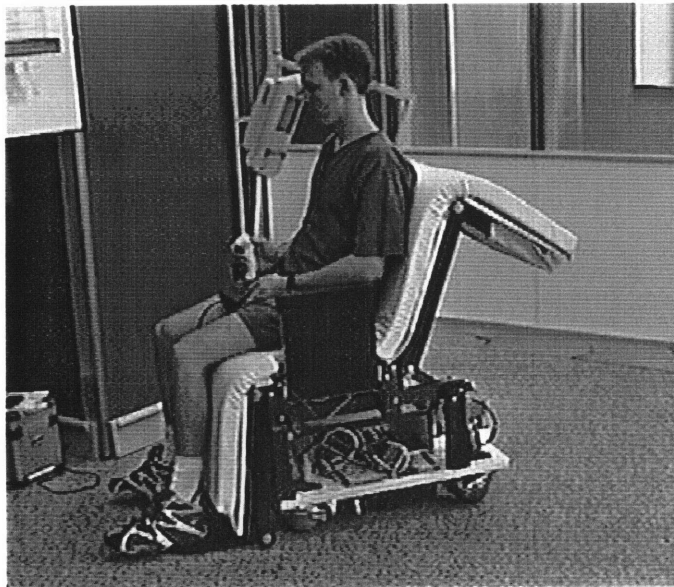


Figure 2.2.1: RHOMBUS Prototype

The vehicle shown in Figure 2.2.1 has four ball wheels driven by independent motors, each generating a traction force in the direction 45 degrees from the chassis centerline. The vehicle chassis is supported by four independent suspensions for improved ride comfort and floor grip. The maximum payload is 150 kg, including the reconfigurable chair and other equipment of up to 50 kg.

Figures 2.2.2 through 2.2.4 show the undocking process of the vehicle from the bed. A patient is lying on his back in the bed, while the reconfigurable chair is in the bed mode (Figure 2.2.2). The vehicle is guided out of the bed (Figure 2.2.3) as the back of the chair and armrests are raised, and the footrest is lowered. By the time the vehicle has undocked from the bed portion, reconfiguration has been completed, and the vehicle is in the chair mode (Figure 2.2.4).

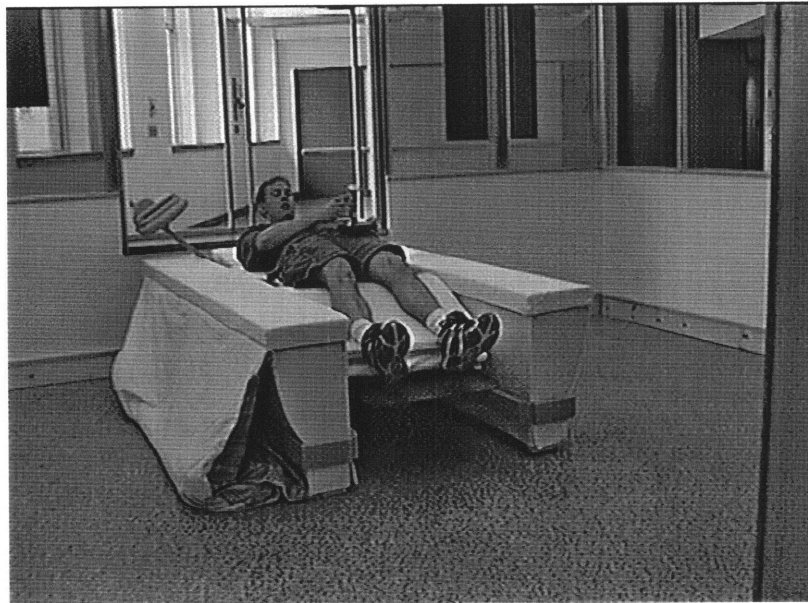


Figure 2.2.2: Bed Mode

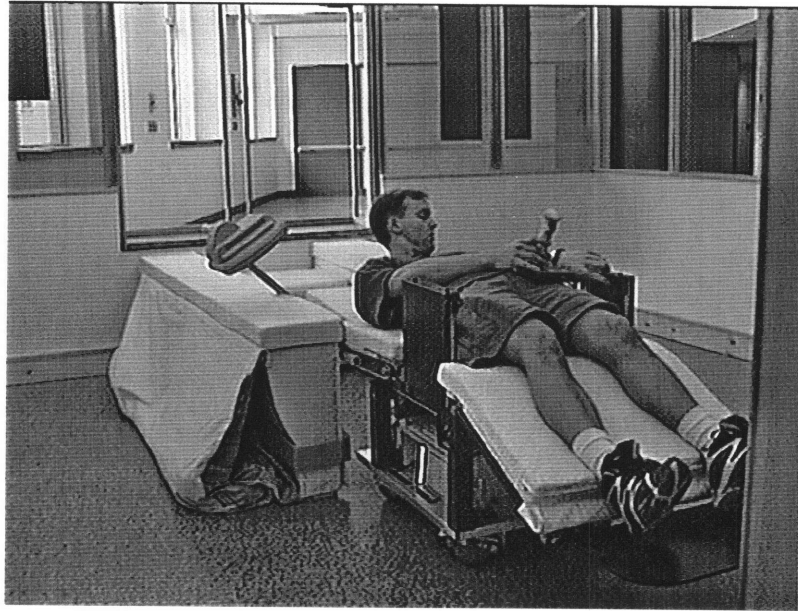


Figure 2.2.3: Intermediate Reconfiguration Stage

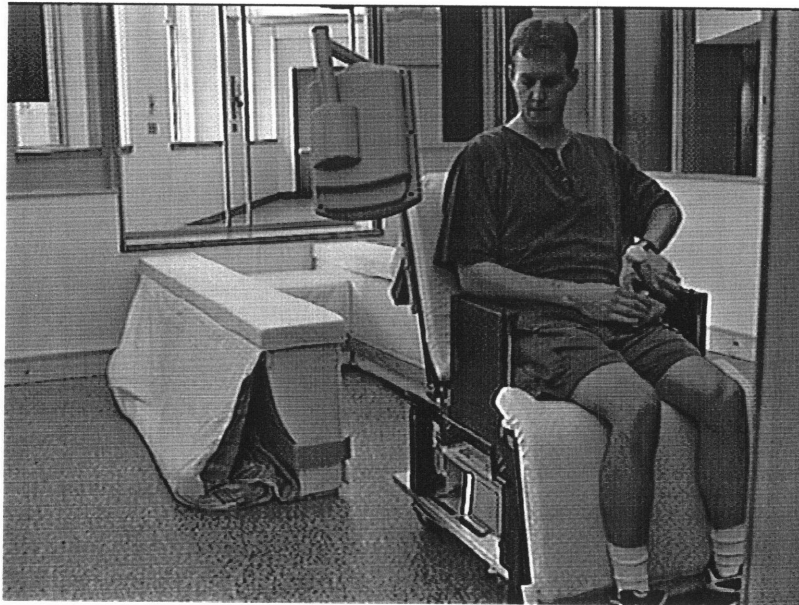


Figure 2.2.4: Chair Mode

The docking and undocking are performed automatically based on the patient's command. During the process, however, the patient is allowed to stop or reverse the process at any time. For safety, the process is supervised by a caregiver at a distal telenursing center. As shown in Figure 2.2.1, the vehicle is equipped with a teleconferencing console consisting of a camera, microphone, and monitor, all connected to a home computer through a two-way, wireless communication line. The caregiver at the telenursing center can be accessed from the chair/bed through the ISDN line connecting the home computer to the telenursing center.

3. THE OMNIDIRECTIONAL HOLONOMIC VEHICLE

3.1 THE BALL WHEELED VEHICLE

Holonomic omni-directional vehicles have been developed by different groups in the last several years, e.g. the OmniTrack [9], the VUTON track mechanism [10], and the orthogonal wheel by [11], to name just a few. The vehicle mechanism which will be used for the RHOMBUS system is based on the ball wheel mechanism by West and Asada [12]. The ball wheel mechanism has no singularity in its entire configuration space, and allows for precision dead reckoning and smooth motion.

Figure 3.1.1 shows the ball wheel with a special ring mechanism. The ball is held by the roller ring at a great circle together with a set of chassis mounted rollers arranged on another great circle. The roller ring is rotated by the servo motor to drive the ball wheel. Since the ring roller is inclined, a traction force is created between the ball wheel and the floor. The vehicle has at least three ball wheels, each generating a traction force in a different direction. The resultant force acting on the vehicle is given simply by the vectorial sum of the traction forces. Varying the combination of the traction forces creates an arbitrary force and moment driving the vehicle. It should be noted that, to move the vehicle in an arbitrary direction, each ball wheel must not be over-constrained. In this ball wheel mechanism, each ball wheel is held by the ring mechanism in such a way that the ball can rotate freely along the chassis mounted rollers. Namely, the whole body

of the ring is actively rotated by the servo motor, but the small rollers arranged on the ring allow the ball to rotate freely in the direction perpendicular to the active axis. Therefore the three ball wheels do not interfere with each other.

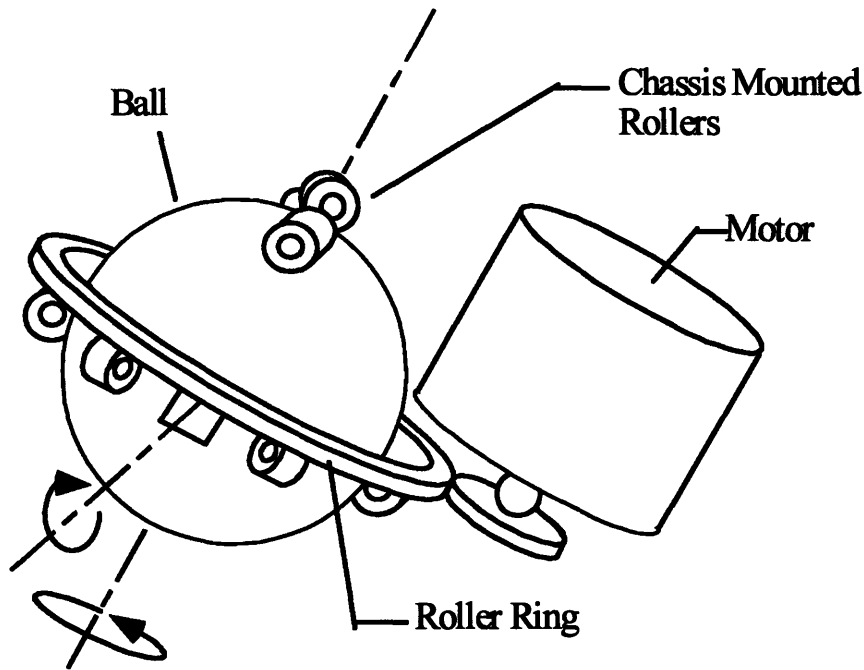


Figure 3.1.1: Ball Wheel Mechanism

Figure 3.1.2 shows the configuration of the original three wheeled vehicle by West and Asada [12]. The active directions of the three ball wheels are oriented at intervals of 120 degrees with respect to each other. The vehicle can move in an arbitrary direction with an arbitrary rotational velocity at an arbitrary position and orientation. The kinematic relationship between motor velocities and vehicle velocities is given by equation (1).

$$\begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix} = \frac{\eta}{R \sin \phi} \mathbf{R}_Z(\Theta) \begin{bmatrix} 0 & 1 & L_2 \\ -\frac{\sqrt{3}}{2} & -\frac{1}{2} & L_1 \\ \frac{\sqrt{3}}{2} & -\frac{1}{2} & L_1 \end{bmatrix} \begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{\Theta} \end{pmatrix} \quad (1)$$

where ω are the motor velocities, η is the gear ratio, R is the ball radius, ϕ is the wheel inclination, and $(\dot{X}, \dot{Y}, \dot{\Theta})^T$ are the velocities of the vehicle measured at point O . $\mathbf{R}_Z^T(\Theta)$ is a rotational transformation for arbitrary Θ . There is no singular point in this mechanism, hence it is omni-directional and holonomic. Moreover, this ball wheel vehicle allows for smooth motion with no shimmy and jerk, all of which are desirable in wheelchairs used for transporting the elderly.

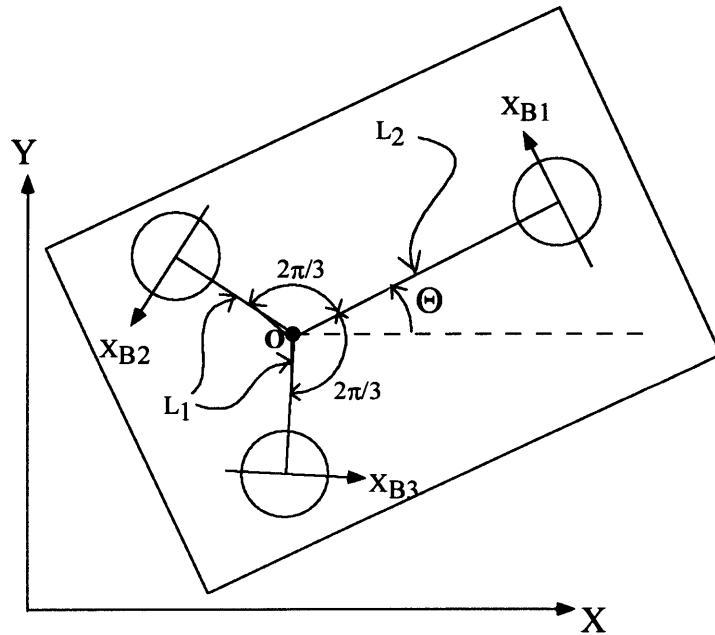


Figure 3.1.2: Three Wheeled Configuration

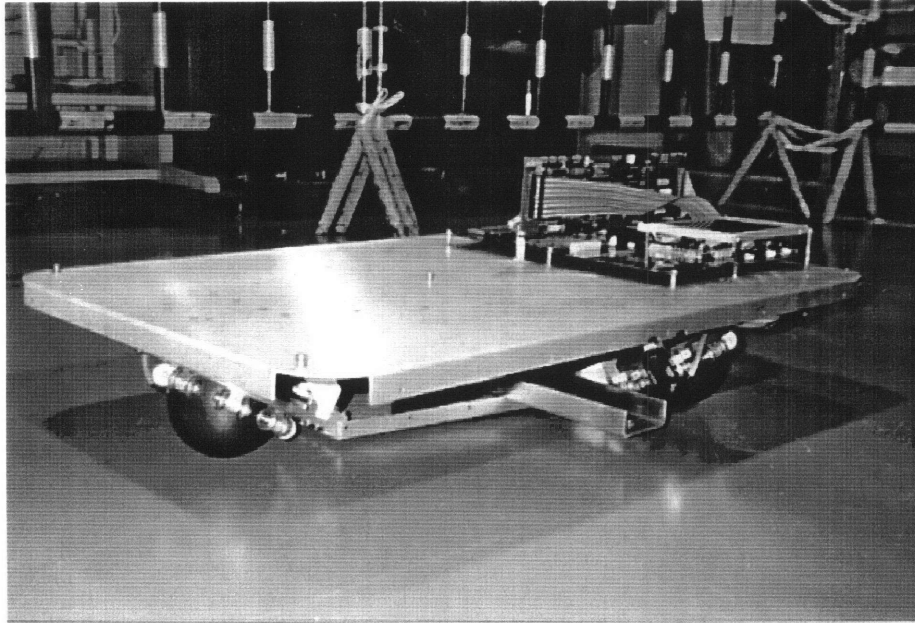


Figure 3.1.3: Original Ball-Wheeled Vehicle

The original vehicle however, shown in Figure 3.1.3, was limited in payload and stability. It has a payload of only 40 kg, driven by three wheels with no suspension. The triangular foot print formed by the three wheels was so small that static balance could hardly be maintained during the reconfiguration operation. For the hybrid chair/bed application, the payload must be increased to carry both a patient and a powered chair, and the foot print must be enlarged, since the mass centroid of the chair and the patient may shift significantly during the reconfiguration operation. To overcome these difficulties, a four-wheeled vehicle with augmented stability and a large payload has been designed. The enlarged foot print encloses the entire range of the mass centroid position during the reconfiguration operation.

3.2 DOCKING WITH A HOLONOMIC VEHICLE

Central to the concept of the RHOMBUS system is that the wheelchair is docked to fixtures, i.e. a bed and toilet. The critical functionality required for the vehicle carrying the chair is that the vehicle must be positioned against and mated with a bed, a toilet and other fixtures in the house environment. In this regard, the holonomic omnidirectional vehicle has three significant advantages over its traditional counterpart, non-holonomic vehicles:

- To drive a traditional, non-holonomic vehicle to a desired position and orientation, a complex nonlinear control entailing the generation of trajectories and switching between forward and backward motions is necessary. In other words, the vehicle cannot be positioned at a desired position and orientation by simply feeding back the error between the vehicle and the target positions [13]. The holonomic omnidirectional vehicle, however, does not incur such a complex problem: direct feedback of position errors drives the vehicle to the target location.
- Docking is not merely a positioning task, but entails the mating with a fixture. During the mating process, the vehicle contacts with the fixture and is constrained by the contact with the fixture. The traditional non-holonomic vehicle is unable to conform to the geometric constraints. Since the non-holonomic vehicle has only two-degrees of freedom, it can move only in one direction when contacting with the fixture at one point. When it contacts at two points, the vehicle totally loses all the degrees of freedom. Therefore, the vehicle gets stuck at the configuration where the two point contact occurs. The insertion can no longer proceed without violating the geometric

constraint. This problem occurs almost always unless the vehicle is perfectly aligned with the fixture. The holonomic omnidirectional vehicle has three degrees of freedom with no singular point in its entire configuration space. Therefore, no kinematic deadlock occurs.

- Since the holonomic vehicle is capable of generating both forces and displacements in arbitrary direction, the vehicle can be programmed to behave with arbitrary stiffnesses in each of its three degrees of freedom about any arbitrary center of compliance. Compliance has been shown to be effective in rigid peg-in-hole insertion or docking tasks [2]. Being able to locate the compliance center at arbitrary position is important because the probability of jamming and wedging can be minimized by locating the compliance center near the front of the peg (in our case the vehicle). Ordinary vehicles cannot exhibit compliance in either lateral or angular directions.

4. THE FOUR WHEELED VEHICLE

4.1 DESIGN

4.1.1 Problem Definition

As noted in Chapter 3, the three-wheeled omnidirectional holonomic vehicle is limited in several respects. Particularly, its small triangular footprint makes it extremely unstable and prone to tipping over when carrying heavy payloads with high center of gravity and variable location, therefore rendering it quite unsafe for use as with a wheelchair. Another concern is the low ground clearance of the three-wheeled vehicle, and its lack of a suspension system, making it ill equipped for navigation of potentially unsmooth terrain and uncomfortable for a human occupant. A final concern is the fact that with only three wheels, there is no redundancy in traction. Although the encoder information can be used to monitor traction and detect when a wheel slips, there is nothing it can do about it. In other words, if one of the wheels should lose traction by coming in contact with a local low friction spot on the floor, the vehicle may not be able to continue moving in the desired direction.

For these reasons, it is necessary to construct a four wheeled omnidirectional holonomic vehicle with augmented stability, comfort, and terrain handling. Specific design goals are given as follows:

- The footprint of the four-wheeled vehicle should be as large as possible in all directions but small enough to navigate through typical residential doorways, bathrooms, etc.
- The vehicle must have a suspension system to provide ride comfort, as well as ensure that all four wheels maintain traction with the floor.
- The ground clearance of the vehicle must be increased to allow navigation over inclines up to at least 5 degrees and bumps up to at least 1 cm. Note: In the case of crossing bumps in the floor, the ultimate limitation will be the ability of the ball itself to climb over the bump. This can be improved by using a bigger ball, but this would require a complete redesign of the wheel itself, and is not the focus at this time.

4.1.2 Design of Footprint/Wheel Configuration

In order to provide sufficient stability without jeopardizing navigation, it was determined that the vehicle chassis should be 24 (61 cm) inches wide by 30 (76 cm) inches long. The length is greater than the width, because the center of gravity shifts lengthwise when the chair is reconfigured to a bed. Eventually, the possibility of using the 4-DOF chair to compensate for the shift in center of gravity can be investigated, in which case the length can be shortened to around 24 inches also.

The four wheels are placed at the corners of the chassis so as to maximize the footprint for the given chassis. The result is a rectangular footprint which is 22.26 in (56.54 cm) long by 16.26 in (41.31 cm) wide. The active directions of the wheels are

oriented at 45 degree angles with respect to the chassis rectangle, so as to equalize the distribution of mobility in all directions to the greatest extent. Figure 4.1.1 shows a schematic of the configuration for the four wheeled vehicle prototype.

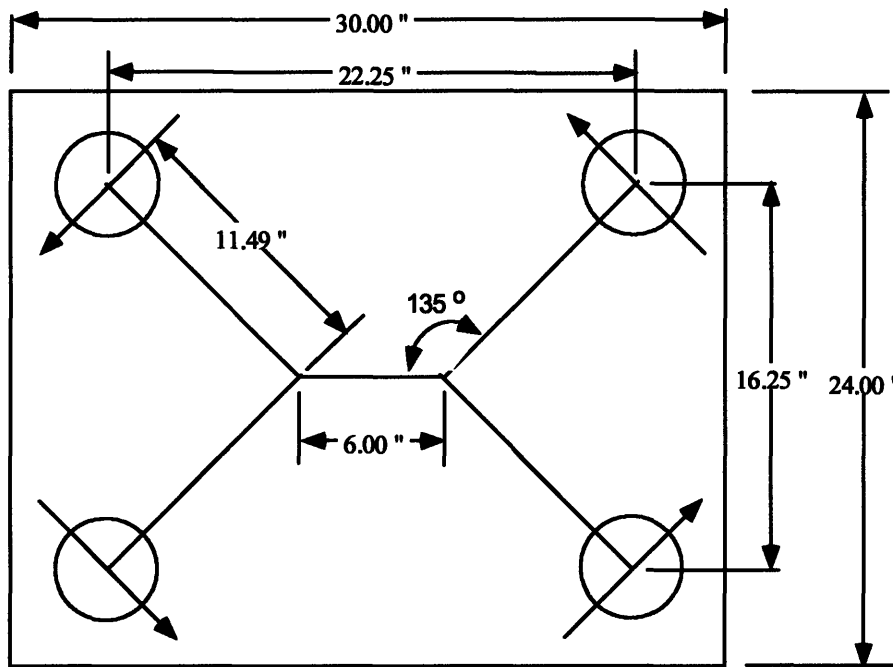


Figure 4.1.1 Four Wheel Configuration

A final decision which must be made in regards to wheel configuration is whether or not to have four active wheels. Three actively driven ball wheels are necessary and sufficient for omnidirectional holonomic motion. Four active wheels would act as an overconstraint, unless the “fourth” wheel is perfectly synchronized with the motion of the vehicle determined by the other three. This could cause potential control difficulty. Certainly we could ensure stability by having three active wheels and one passive wheel. However, especially with large payloads, friction in the wheel mechanisms is a primary concern. A completely passive ball wheel would exert a large drag on the vehicle and the

asymmetry of traction forces would tend to cause the vehicle to veer from a desired trajectory. An ordinary castor wheel with low friction could be used in place of a passive ball-wheel, however such castor wheels have a singularity and the vehicle would no longer be perfectly holonomic. Furthermore, redundancy in traction may be desired in order to better deal with situations where one of the wheels encounters a slippery spot or otherwise loses traction. Therefore, we will design the vehicle with four active wheels and deal with any overconstraint problems via control.

4.1.3 Design of Suspension System

The goal in designing a suspension system for the four-wheeled vehicle is to provide compliance for each of the wheels in the vertical direction, so that traction can be maintained on all four wheels and bumps will be absorbed. Since some compliance can also be designed into the chair itself to filter out jerky motions from the human occupant, we will assume that maintaining traction is the more important function of this suspension system. The most important constraint is that motion of each wheel must only be allowed in the vertical direction. If the wheel significantly deflects horizontally (parallel to the floor) or rotates in any way, this will change the kinematics of the vehicle and dead reckoning will no longer be accurate. This is a serious concern since the large payload of the vehicle can potentially generate high forces and moments on the wheels in a variety of directions. In any case, suspension stiffness must be relatively high. Tilting of the chassis can cause the wheel to change inclination with respect to the floor. Small variations in this inclination will cause quite large changes in kinematics.

Firstly, horizontal forces and rotational moments can be minimized by transmitting the weight from the chassis to each wheel through a point directly above the center of each ball. Secondly, horizontal and rotational motions can be prevented by providing a secondary horizontal four-bar-linkage support between each wheel and the chassis. As long as the four-bar links are long enough, and vertical stroke of the wheel suspension is small enough, the four-bar-link will approximate purely vertical motion.

An alternative to the four-bar-linkage might have been to use linear bearings on top of each wheel to restrict motion to purely vertical. However, a high payload could put large transverse stresses on the bearings, and any cantilever bending of the slides would cause a horizontal deflection of the wheel and a change in inclination. The four-bar-linkage design, on the other hand, is particularly well suited for resisting changes in wheel inclination, since rotational stiffness is provided by axial stiffness of the links, which can be easily made quite high.

Figure 4.1.2 shows a CAD drawing of the final design of the suspension system for the prototype four-wheeled vehicle. As shown, the weight of the payload is transmitted through a point directly above the center of the ball by an air cylinder shock absorber, which provides both stiffness and damping. A four-bar-linkage attaches the side of the wheel to a mount underneath the chassis. The chassis itself is reinforced with several ribs and a belly plate connects the bottoms of all four mounts. Figures 4.1.3 and 4.1.4 show pictures of the fabricated prototype with suspension system.

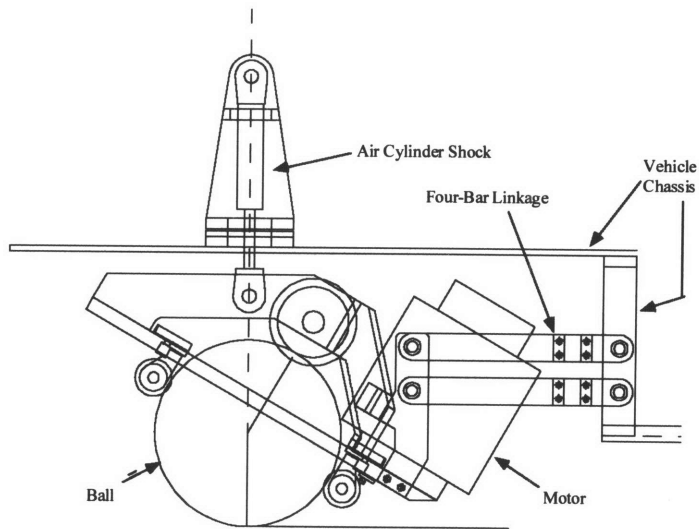


Figure 4.1.2: Suspension System

Appendix A shows a complete CAD drawing of the layout of the four-wheeled vehicle.

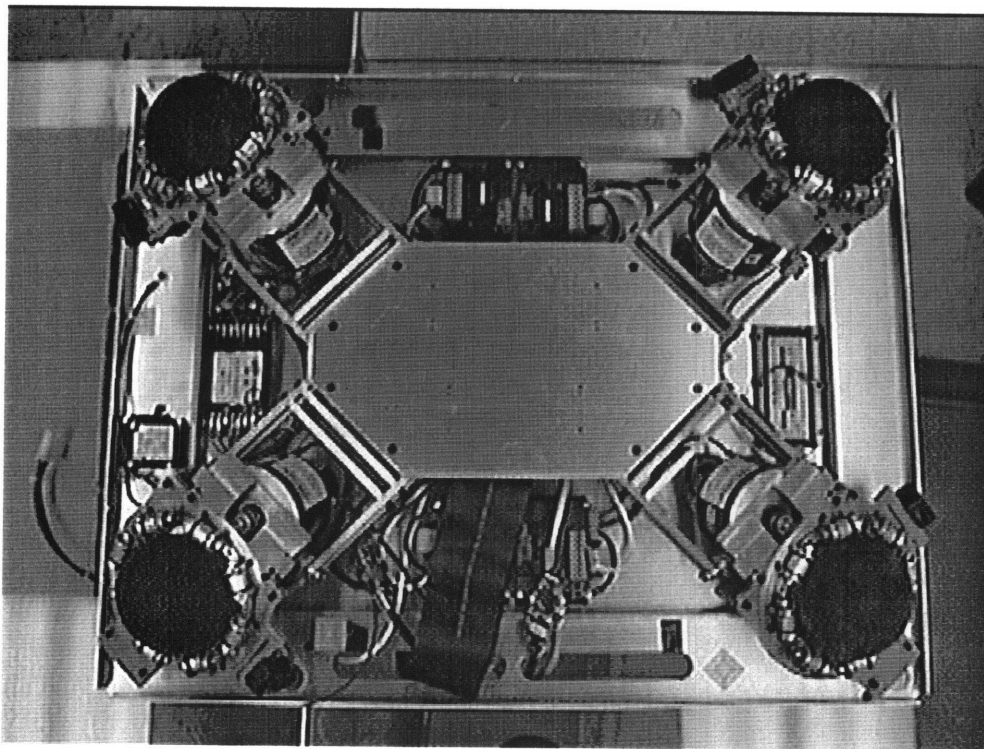


Figure 4.1.3: Ball Wheeled Vehicle - Bottom View

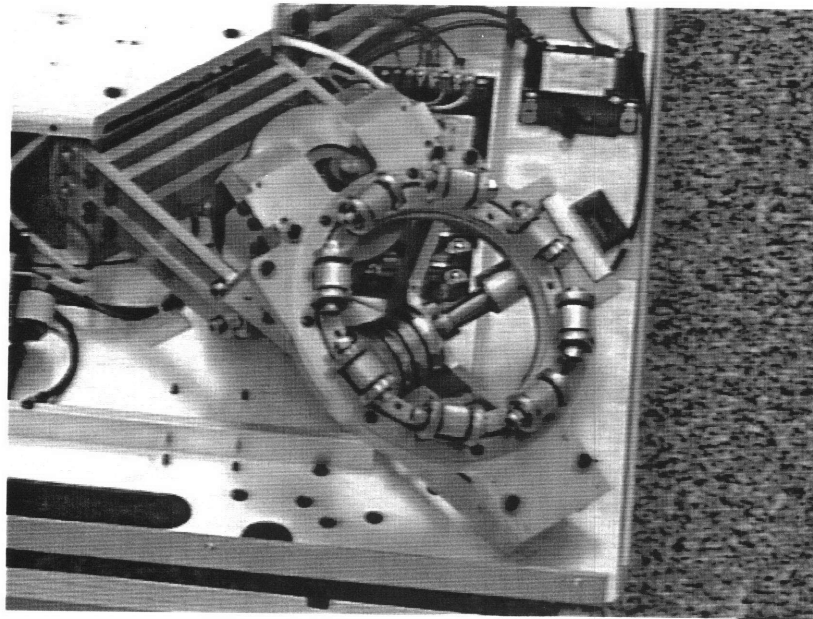


Figure 4.1.4 Ball Wheel Mechanism with Suspension

4.2 KINEMATICS

In chapter 3, the kinematics for the three-wheeled omnidirectional vehicle were given by eq. (1). As with the three-wheeled vehicle, it is a simple matter to write the inverse kinematics for the four-wheeled vehicle, expressing wheel motor velocities in terms of vehicle velocities. Figure 4.2.1 shows the vehicle configuration at arbitrary vehicle coordinates $(X, Y, \Theta)^T$, measured. The active contact velocities of the balls are shown by

$$(\dot{X}_{B1}, \dot{X}_{B2}, \dot{X}_{B3}, \dot{X}_{B4})^T .$$

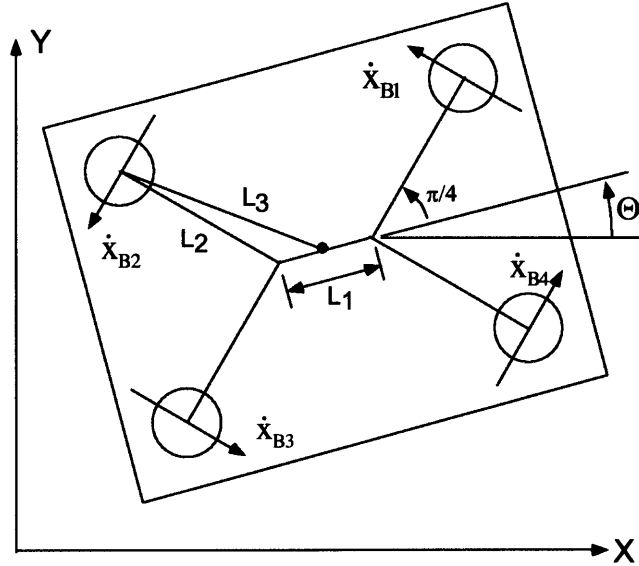


Figure 4.2.1: Four Wheeled Configuration

A complete derivation of the inverse kinematics is demonstrated in Appendix B. The results are summed up by equations (2)-(4).

$$\begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{pmatrix} = \frac{\eta}{R \sin \phi} \mathbf{J}^{-1} \mathbf{R}_Z^T(\Theta) \begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{\Theta} \end{pmatrix} \quad (2)$$

where ω are the motor velocities, η is the gear ratio, R is the ball radius, and ϕ is the wheel inclination, and $(\dot{X}, \dot{Y}, \dot{\Theta})^T$ are the velocities of the vehicle at its geometric center. The inverse Jacobian \mathbf{J}^{-1} , is given by equation (). $\mathbf{R}_Z^T(\Theta)$ is a rotational transformation for arbitrary Θ .

$$\mathbf{J}^{-1} = \begin{bmatrix} -\sin(\pi/4) & \cos(\pi/4) & \alpha L_3 \\ -\sin(3\pi/4) & \cos(3\pi/4) & \alpha L_3 \\ -\sin(-3\pi/4) & \cos(-3\pi/4) & \alpha L_3 \\ -\sin(-\pi/4) & \cos(-\pi/4) & \alpha L_3 \end{bmatrix} = \begin{bmatrix} -\sqrt{2}/2 & \sqrt{2}/2 & \alpha L_3 \\ -\sqrt{2}/2 & -\sqrt{2}/2 & \alpha L_3 \\ \sqrt{2}/2 & -\sqrt{2}/2 & \alpha L_3 \\ \sqrt{2}/2 & \sqrt{2}/2 & \alpha L_3 \end{bmatrix} \quad (3)$$

$$\text{where } \alpha = \frac{L_2^2 + L_3^2 - \frac{1}{4}L_1^2}{2L_2L_3} \quad (4)$$

As stated in the previous section, the four active wheels pose an overconstraint. This is seen in the 4x3 size of the inverse Jacobian. We cannot directly invert this matrix to obtain the Jacobian matrix, which transforms from wheel velocities to vehicle velocities. However, there are alternative methods to synthesizing a Jacobian. The simplest solution is to define the Jacobian as the minimum norm pseudoinverse of \mathbf{J}^{-1} . Specifically, the Jacobian will be the left pseudoinverse \mathbf{J}^{LM} , which is obtained by equation (5).

$$\mathbf{J}^{LM} = (\mathbf{J}^{-T}\mathbf{J}^{-1})^{-1}\mathbf{J}^{-T} \quad (5)$$

By performing the algebra, we get:

$$\begin{aligned} \mathbf{J}^{LM} &= \begin{bmatrix} -\frac{1}{2}\sin(\pi/4) & -\frac{1}{2}\sin(3\pi/4) & -\frac{1}{2}\sin(-3\pi/4) & -\frac{1}{2}\sin(-\pi/4) \\ \frac{1}{2}\cos(\pi/4) & \frac{1}{2}\cos(3\pi/4) & \frac{1}{2}\cos(-3\pi/4) & \frac{1}{2}\cos(-\pi/4) \\ \frac{1}{4\alpha L_3} & \frac{1}{4\alpha L_3} & \frac{1}{4\alpha L_3} & \frac{1}{4\alpha L_3} \end{bmatrix} \\ &= \begin{bmatrix} -\sqrt{2}/4 & -\sqrt{2}/4 & \sqrt{2}/4 & \sqrt{2}/4 \\ \sqrt{2}/4 & -\sqrt{2}/4 & -\sqrt{2}/4 & \sqrt{2}/4 \\ \frac{1}{4\alpha L_3} & \frac{1}{4\alpha L_3} & \frac{1}{4\alpha L_3} & \frac{1}{4\alpha L_3} \end{bmatrix} \end{aligned} \quad (6)$$

The forward kinematics are then given by:

$$\begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{\Theta} \end{pmatrix} = \frac{R \sin \phi}{\eta} \mathbf{R}_Z(\Theta) \mathbf{J}^{LM} \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{pmatrix} \quad (7)$$

There is a second way of synthesizing a Jacobian, where the physical significance is more obvious. We can actually simplify the inverse kinematic equation (7) such that we only use three out of the four wheel velocities to calculate $(\dot{X}, \dot{Y}, \dot{\Theta})^T$. The inverse Jacobian then becomes a 3x3 matrix. We can then take the inverse of this to get a 3x3 Jacobian. There are four combinations of three wheels which we can choose, and four corresponding 3x3 submatrices of the inverse Jacobian, which can be used to calculate four corresponding 3x3 Jacobians. Each of these four “sub-Jacobians” can be used to calculate $(\dot{X}, \dot{Y}, \dot{\Theta})^T$. If the vehicle behaves perfectly and no wheel slipping or other unmodelled kinematic effects occur, then any of these “sub-Jacobians” will do. However, in the case of wheel slipping or other errors, we can calculate $(\dot{X}, \dot{Y}, \dot{\Theta})^T$ more accurately by using some combination of the four “sub-Jacobians.” It turns out that if we simply average all four, we end up with the same 3x4 pseudo-Jacobian that we derived above in eq. (6). If we wish to improve the accuracy of the forward kinematic even further, we could choose an algorithm which compares the results from each of the four “sub-Jacobians” and gives less weight to the ones which appear to have more error.

In order to gain more insight into the nature of the overconstraint on the system, we can model the vehicle kinematics and dynamics using a bond graph approach. Figure 4.2.2 shows a model of the vehicle including the new suspension system. The greatest

compliance in the suspension which will affect the kinematics is the compliance due to bending of the four-bar-linkages. As shown, each suspension linkage is subject to bending in the direction of active velocities of the wheels.

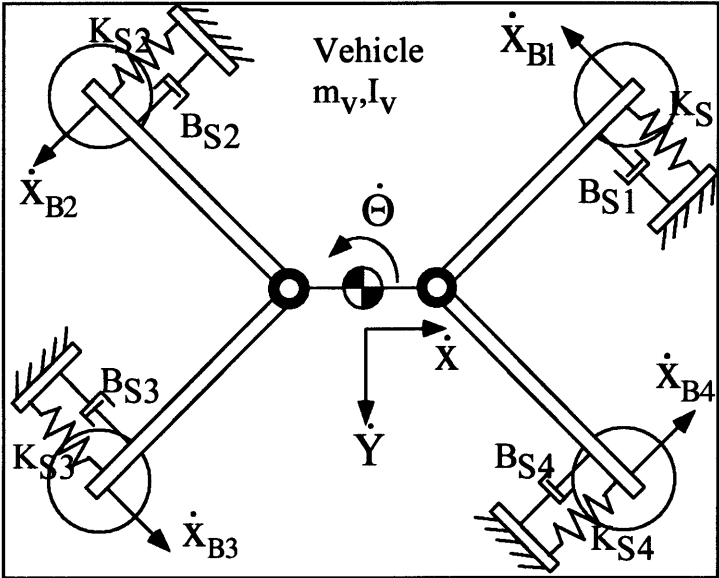


Figure 4.2.2: Vehicle Model

Figure 4.2.3 shows the bond graph associated with Figure 4.2.2. The four wheel subsystems are shown in the shaded blocks on the left-hand side of the figure, and the X, Y, Θ subsystems are shown in the shaded blocks on the right-hand side. The bonds between left and right illustrate the kinematic relationships between wheel forces/velocities and vehicle forces/velocities. In this model, it is assumed that a high gain position feedback loop is being used to precisely control the velocities of the wheels, such that the wheels can be modeled as flow sources. As the model stands, with suspension compliance included, the causalities were placed without difficulty. However, if the suspension compliance were removed, all four wheels would impose flow on the zero

junctions at the left of the kinematic bonds. This means that at each of these four zero junctions, one of the kinematic bonds must impose effort. This requires a total of four kinematic bonds imposing effort on the zero-junction side. These same four bonds would then have to impose flow on the one-junction side. However, each one-junction can only have one bond imposing flow. Therefore, we see that the causalities cannot work in this case. In other words, unless we include some method of relieving the overconstraint (such as suspension compliance), we cannot properly model the vehicle dynamics.

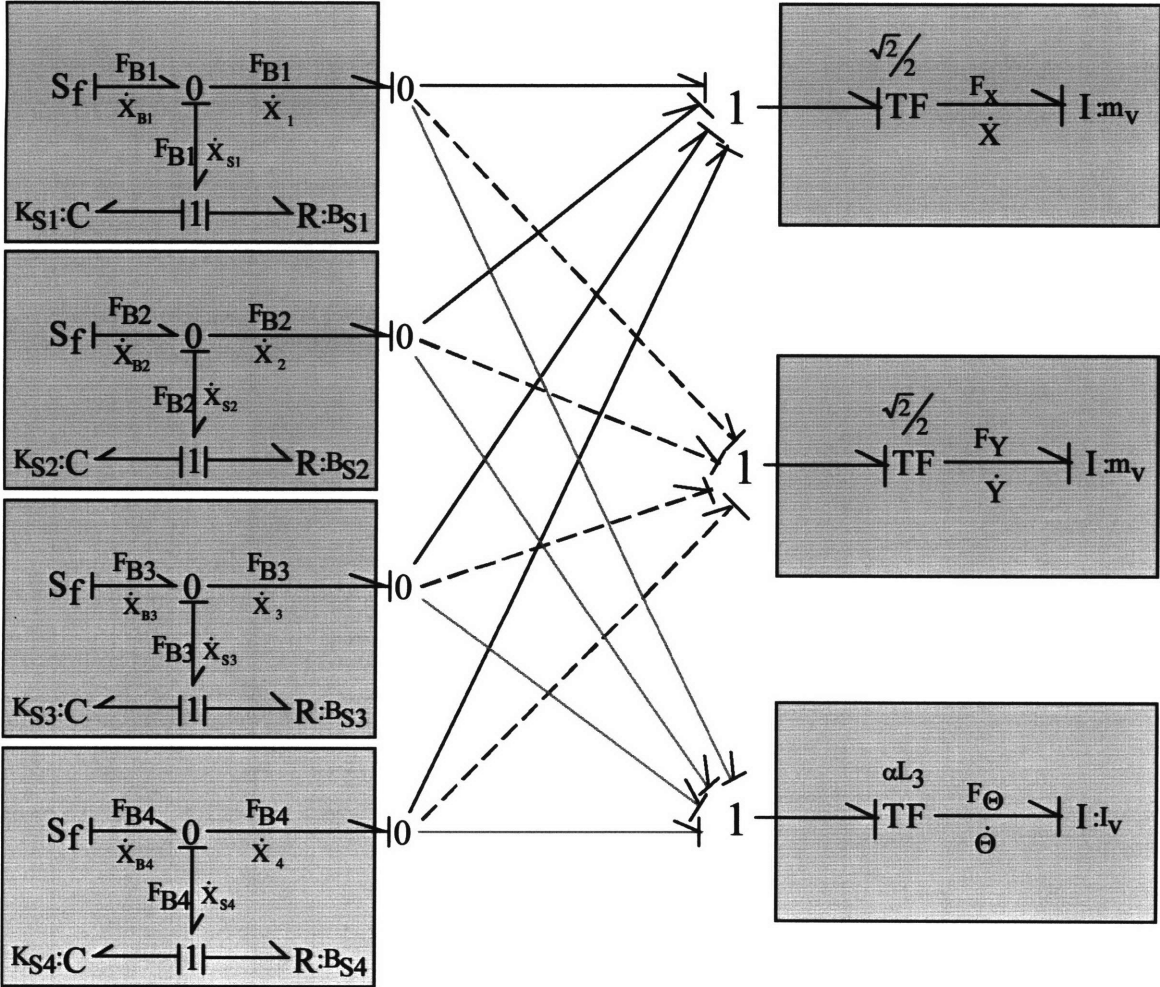


Figure 4.2.3 Bond Graph Model of Vehicle

4.3 CONTROL

4.3.1 Hardware

The vehicle control system is implemented on a portable pentium PC with ISA bus to D/A and encoder interface boards. The actuators are rare earth magnet brush DC motors driven by PWM amplifiers. Wireless communication is provided for by a wireless ethernet card. An optical analog joystick is used for manual drive. A/D and Digital IO are also available for sensor input. The entire system is powered by a 24 Volt, 31 Amp-Hour rechargeable battery pack. Batteries are sealed lead-acid gel cell type. A complete schematic of control hardware is shown in Appendix C.

4.3.2 Manual Control

Figure 4.3.1 shows a block diagram of the simple control system which was used to manually drive the vehicle. Input from the joystick is first translated into desired vehicle velocities $(\dot{X}, \dot{Y}, \dot{\Theta})^T$, which are digitally filtered before passing on. The joystick used has two perpendicular axes of analog control. The velocity of the vehicle in the X and Y directions are proportional to the displacement of the stick along these axes. Velocities are zero below a marginal threshold. A pair of buttons on the stick are used to control rotational motion. The vehicle rotates counterclockwise when the left button is depressed, and clockwise when the right is depressed. In future versions, a joystick with a third axis of analog rotation will be used. Rotational motion can then be controlled more smoothly

by twisting of the wrist. A throttle next to the stick is used to set maximum velocity and thus proportionality of velocity to stick displacement. $(\dot{X}, \dot{Y}, \dot{\Theta})^T$.

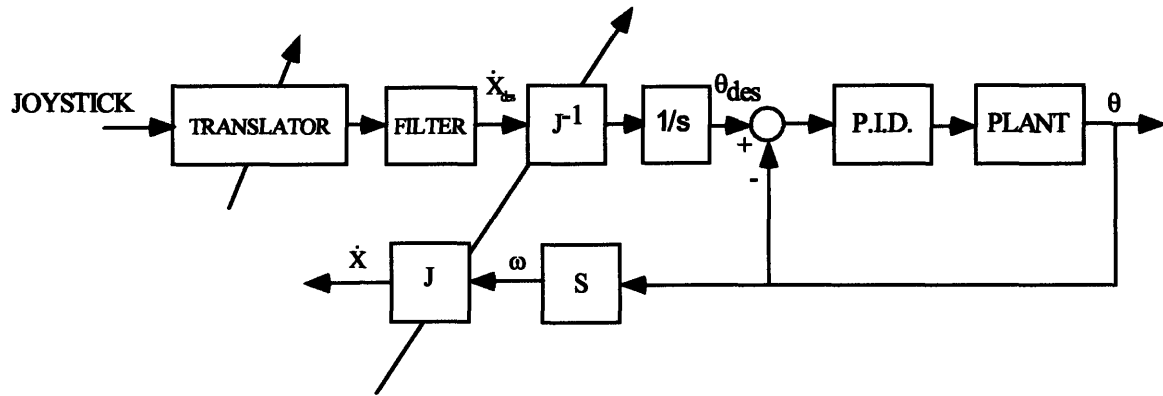


Figure 4.3.1: Block Diagram of Manual Control

A PID loop is used to control the position/velocity of each wheel based on the desired vehicle position. The forward Jacobian can be used to keep track of the actual vehicle position for dead reckoning purposes. Although we could feedback the updated position to improve tracking, manual control does not require that amount of accuracy. As long as the gains are high enough, the vehicle will not deviate enough to affect driving. A program written in C language for manual control of the vehicle is included in Appendix D.

Rotational and linear motions can of course occur simultaneously. As added features, the vehicle can be programmed to turn with any specific radius by setting the magnitude of the rotational velocity according the linear velocity. The vehicle can also be programmed to rotate about some other axis than its geometric center by simply premultiplying the

inverse Jacobian by a simple translational operator. This brings up the issue of *human-centered-control*, which addresses how one might regulate the motions of the vehicle in order to maximize the comfort of the rider or the ease of driving. This idea will be discussed briefly in Chapter 6.

4.3.3 Automatic Control

With a few modifications to the control scheme described above, the vehicle can be automatically guided without low level input from the rider. Figure 4.3.2 illustrates the control system. Based on a knowledge of starting points, end points, and via points, a smooth trajectory is generated for the vehicle to follow. Since the vehicle is completely omnidirectional and holonomic, there are no kinematic restrictions on choosing a trajectory. The way in which the trajectory is chosen is another issue which can be addressed through *human-centered-control*: specifically, how do we generate a trajectory which is most comfortable for the rider.

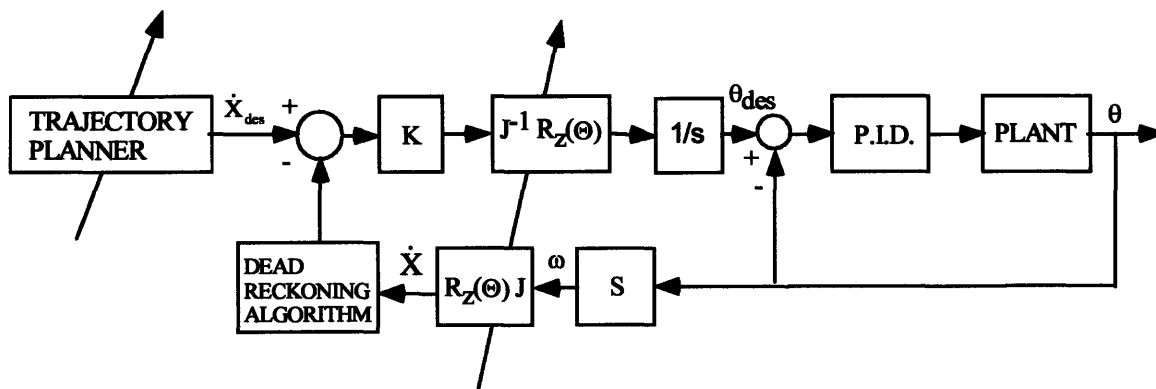


Figure 4.3.2: Block Diagram of Automatic Control

Although there are many ways in which the vehicle can interact with the environment and receive information concerning about its position, the core control is based on the vehicle's dead reckoning. As in manual control, each wheel is still controlled by a PID loop. However, we now make use of the Jacobian to update and feedback the vehicle position using a dead reckoning algorithm as in West [12]. This improves the tracking performance of the vehicle and minimizes the accumulation of error in the trajectory. As in West [12], we could also include a slip detection algorithm in series with a traction control algorithm to update the trajectory planner. As shown by the block diagram, the rotational operator must now be included with the Jacobian to account for varying Θ . As before, we can also premultiply the Jacobian by a translational operator to locate the center of rotation at any desired point.

The automatic control as described was successfully implemented on the vehicle and used to perform simple docking maneuvers which require precise positioning without rigid mating. The vehicle was able to travel across a 10 foot room with a wandering trajectory and successfully park itself over a wall mounted toilet with a few inches of clearance between toilet and chair.

4.3.4 Compliance Control

In order for the vehicle to perform rigid docking maneuvers, *active compliance control* can be used. Part insertion tasks have been performed with *passive compliance control*, where the compliance of the part comes from actual mechanical springs from

which it is supported [4]. Here we seek to program the vehicle to behave with certain compliances by softening the position gains in the PID feedback loop. This method was originally suggested by Salisbury [5]. Although this method can also be referred to as *stiffness control*, we will refer to it as throughout this work as *compliance control* in order to distinguish it from other types of stiffness control.¹

The key to compliance control is the relation between joint stiffness and end-effector stiffness, or in our case, the relation between wheel stiffness and vehicle stiffness. Vehicle stiffness is defined by K_v , where

$$\mathbf{F} = \mathbf{K}_v \delta \mathbf{X} \quad (8)$$

$\mathbf{F} = (F_x, F_y, \Gamma_\theta)^T$ are the forces exerted by the vehicle, and $\delta \mathbf{X} = (\delta X, \delta Y, \delta \theta)$ are the vehicle displacements. Recalling equation (7), we have

$$\delta \mathbf{X} = \mathbf{R}_z(\Theta) \mathbf{J} \delta \theta \quad (9)$$

If we want to locate the compliance center at arbitrary position with respect to the geometric center of the vehicle, we must include a translational operator \mathbf{D} .

$$\delta \mathbf{X} = \mathbf{R}_z(\Theta) \mathbf{D} \mathbf{J} \delta \theta \quad (10)$$

Combining (8) and (10) gives

$$\mathbf{F} = \mathbf{K}_v \mathbf{R}_z(\Theta) \mathbf{D} \mathbf{J} \delta \theta \quad (11)$$

Static wheel torques are related to vehicle forces by the transpose of the Jacobian.

$$\boldsymbol{\tau} = (\mathbf{R}_z(\Theta) \mathbf{D} \mathbf{J})^T \mathbf{F} = \mathbf{J}^T \mathbf{D}^T \mathbf{R}_z^T(\Theta) \mathbf{F} \quad (12)$$

¹ Whitney describes *stiffness control* as a subset of *impedance control*, where the stiffness is derived by controlling position based on feedback of force measurements [3]. *Stiffness control* by this definition, will be discussed in Chapter 5.

Combining (11) and (12) yields

$$\boldsymbol{\tau} = \mathbf{J}^T \mathbf{D}^T \mathbf{R}_z^T(\Theta) \mathbf{K}_v \mathbf{R}_z(\Theta) \mathbf{D} \mathbf{J} \delta \theta \quad (13)$$

We can therefore define the wheel stiffness as

$$\mathbf{K}_w = \mathbf{J}^T \mathbf{D}^T \mathbf{R}_z^T(\Theta) \mathbf{K}_v \mathbf{R}_z(\Theta) \mathbf{D} \mathbf{J} \quad (14)$$

So in order to implement compliant control for the vehicle, we first choose a set of desired stiffnesses for the vehicle in the floor coordinates.

$$\mathbf{K}_v = \begin{bmatrix} K_x & 0 \\ 0 & K_y & 0 \\ 0 & 0 & K_\theta \end{bmatrix} \quad (15)$$

Equation (14) is then used to calculate the wheel stiffnesses \mathbf{K}_w necessary to produce these vehicle stiffnesses. Finally, we write the proportional control law for the four wheels as:

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix} = \mathbf{K}_w \begin{bmatrix} \tilde{\theta}_1 \\ \tilde{\theta}_2 \\ \tilde{\theta}_3 \\ \tilde{\theta}_4 \end{bmatrix} = \mathbf{K}_w \begin{bmatrix} \theta_{1DES} - \theta_1 \\ \theta_{2DES} - \theta_2 \\ \theta_{3DES} - \theta_3 \\ \theta_{4DES} - \theta_4 \end{bmatrix} \quad (16)$$

where \mathbf{K}_w is the position gain matrix. Derivative and integral terms as well as friction compensation can be easily added to the control law in a normal fashion. This PID control law is used directly in the control system shown by Figure 4.3.2.

By implementing this compliant control scheme, the vehicle will follow a desired trajectory while complying to applied forces and moments. The center of compliance is determined by the transform \mathbf{D} , and the stiffnesses are specified by \mathbf{K}_v . This compliant

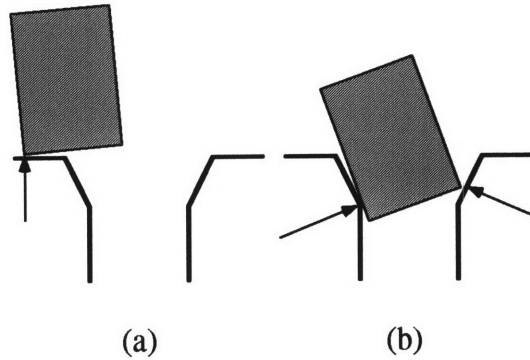
control method is very convenient, because it enables the vehicle to exhibit compliance without any special mechanical flexibility or added sensors. Furthermore, the stiffnesses and the center of compliance can be changed on the fly via software. However, the performance of this control method is limited by the high friction in the wheel mechanisms or poor backdriveability of the vehicle, which may cause the vehicle to be insensitive to small forces and moments.

5. FORCE GUIDED DOCKING OF THE VEHICLE

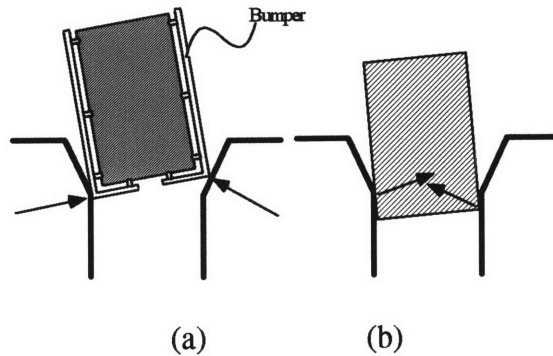
5.1 TASK DEFINITION

The overall goal for functionality of the vehicle in regards to docking is that it be able to successfully, robustly, and safely dock itself automatically with both the bed and toilet, and perhaps with other stations in the home. At this stage, the focus will primarily be on docking with the bed. However, by developing a system which is capable of docking smoothly with very small tolerances under a variety of initial alignments, the prototype system will be easily adaptable to a variety of docking tasks. In designing the mechanics and control of the docking process, there are a few unique issues which need special consideration:

- It is desirable to maximize ride comfort for the human who is sitting in the chair, especially during docking maneuvers. It is therefore important to minimize the jerk and impact forces felt by the human during the docking operation.
- The vehicle when fully loaded, is limited in backdriveability. Therefore compliance control alone without force feedback cannot guarantee smooth docking with low impact forces.
- Since the vehicle must travel long distances over imperfect terrain to reach the bed portion, the initial misalignment with the bed is much larger than that of robotic assembly. As shown in Figure 5.1.1, the vehicle must be docked despite large lateral error (a) and orientation error (b).



(a) (b)
Figure 5.1.1: **Misalignment**



(a) (b)
Figure 5.1.2: **Instrumented Bumper**

To satisfy these restrictions and requirements, we will use an instrumented bumper. The bumper will consist of a rail or system of rails about the perimeter of the vehicle, as shown in Figure 5.1.2. The bumpers will be designed with a specific set of compliances, either discrete or distributed, and will be equipped with pressure sensors measuring contact forces with the bed portion. This bumper system as shown, has at least four distinct advantages:

- As suggested by its name, the bumper alleviates impacts and jerk during the docking process. The bumper can be designed with a sufficiently large stroke and small stiffness such that the vehicle can react slowly and smoothly to contact forces.
- The compliance of the bumper is much greater than the compliance due to part deformation in a typical peg insertion task. Therefore, wedging can be tolerated

without causing physical damage, and docking can proceed under wedging conditions.

Whitney [2] defines the smallest angle at which wedging occurs by

$$\theta_w = c/\mu \quad (17)$$

where μ is the coefficient of friction and c is the clearance ratio. When c is very small, i.e. the peg width and hole width are nearly equal, the bumper compliance can effectively allow the clearance to increase by factors as large as 10, and therefore allow the minimum wedging angle to increase by the same factor.

- The embedded sensors in the bumper allow the vehicle to be docked using active force feedback. The bumpers are sensitive enough to measure forces which the vehicle could not otherwise respond to, i.e. forces which would be too small to overcome the friction of the wheel mechanisms and backdrive the vehicle.
- Finally, unlike traditional peg insertion using a wrist force sensor [14], in which the resultant force and moment of contact forces are measured, the instrumented bumper detects individual contact forces. For example, the two contact forces in Figure 5.1.2 are measured separately rather than measuring their vectorial sum. Therefore, various contact configurations (states) can be clearly distinguished, and the vehicle can be guided correctly despite a large misalignment. In particular, this separate contact force measurement allows direct estimation of the wedging force, i.e. the force acting between two opposing contact points (Figure 5.2.1b), which is not measurable in the traditional peg insertion.

5.2 BUMPER DESIGN

With the docking task defined, a simple prototype bumper system was designed. The first step was to choose a mechanical configuration for the bumpers. The goal at this stage was to choose the simplest configuration that would guarantee the following requirements:

1. The bumper mechanism must have at least two separate components such that contact forces on either side can be measured separately.
2. The bumper stiffness should be at least an order of magnitude less than the suspension stiffness in order to avoid problems associated with non-collocated sensor feedback.

Figure 5.2.1 models the vehicle/bumper system with suspension stiffness K_s and bumper stiffness K_b .

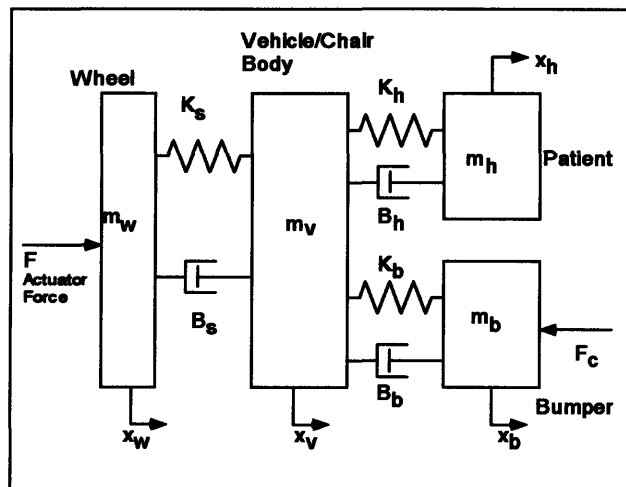


Figure 5.2.1: Schematic of Vehicle/Bumper System

3. The bumpers should also be compliant enough and have a long enough stroke such that impact forces are not felt and decelerations can be kept below a few cm/sec^2 .

Figure 5.2.1 shows how contact force F_c is transmitted through bumper stiffness K_b and cushion stiffness K_h .

4. Sensors must be chosen and placed such that both magnitudes and locations of contact forces can be determined, so that both the net lateral force and moment can be calculated.

There are also a few points to be noted concerning simplification of the design:

1. While it is important to make direct measurements of the magnitude of the force, the location of the force applied can be determined by a knowledge of the contact state of the docking process. This can be accomplished by using a small number of simple contact sensors at critical points on the bumper. For example, Figure 5.2.2 shows the bumper system with two contact sensors at the front corners of the bumpers. If sensor 2 reports contact, then we know the force f_2 is applied at the front corner of the vehicle while f_1 is applied at a distance L from the front of the vehicle. L is directly determined from the insertion depth, which is known by keeping track of the history of the docking process.

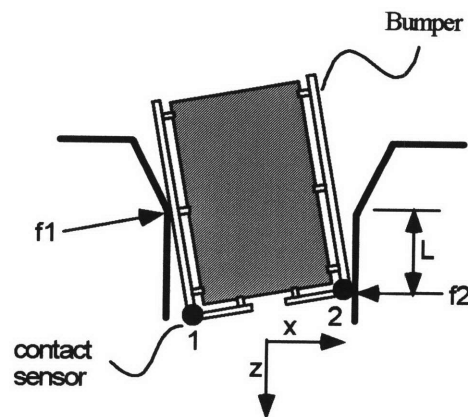


Figure 5.2.2: Determining Location of Contact Forces

2. The insertion force is not needed for direct feedback control, so only a rough estimate is necessary in order to supervise the docking procedure. If insertion forces get too high, or contact is detected along the surface outside of the hole, motion in the z-direction can be halted.
3. The most critical time in the life cycle of the docking process is when the vehicle is at a shallow insertion depth. The highest impact forces will occur during chamfer contact, and wedging is most likely to occur when the vehicle is barely inserted. Therefore, it is more important that the bumpers be sensitive towards the front of the vehicle than towards the back.

Based on these design requirements and considerations, a simple prototype bumper system was developed. A schematic is shown in Figure 5.2.3. The design consists of dual single-degree-of-freedom bumpers on either side of the vehicle. Each bumper is spring loaded at the front and pin-jointed at the rear. A precision linear potentiometer is placed near the springs and is used to measure the bumper displacement, which will be used to calculate bumper force based on spring stiffness. A roller/limit switch is imbedded in the front end to minimize friction while detecting contact.

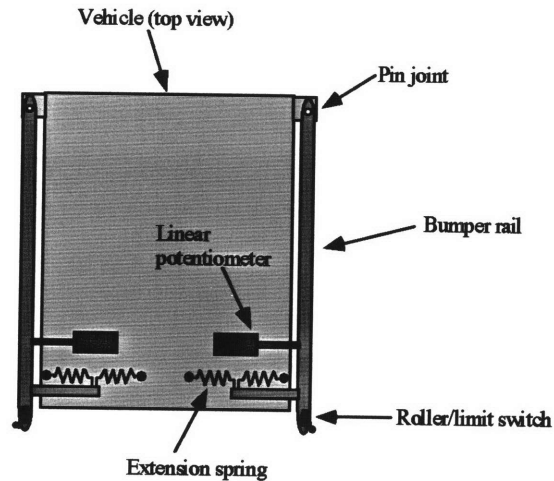


Figure 5.2.3: Prototype Bumper Design

A few important features of this design should be noted:

- The number of mechanical degrees of freedom have been kept to a minimum of two.
- The pin joints at the rear allow the rail to bear loads in the insertion direction without putting transverse loading on the displacement sensors or springs.
- The stroke of the bumper is approximately 1 degree of rotation, or approximately 1/2 inch at the point where the displacement sensor is attached. Under these conditions, the motion of the sensor at this point is sufficiently close to linear.
- The compliance of the bumper increases from rear to front. There is actually a singularity at the point where the bumper is pinned. The significance of this will be addressed in the next section.

5.3 DOCKING CONTROL

5.3.1 Bumper Model

In order to successfully dock the vehicle, we have decided to use a control scheme based on force feedback from the instrumented bumpers. The simplest way to accomplish this is to use stiffness control, which is a specific case of impedance control. First, however, it is important to derive an appropriate model for the system we have designed. To simplify matters, we will first model the dynamics of the vehicle in the lateral direction, assuming dynamics in the insertion direction and rotational direction are independent from this. Furthermore, we will assume that all four wheels act as a single effort source driving the vehicle in the lateral direction. The suspension system stiffness can be neglected, compared with the bumper stiffness. Figure 5.3.1 shows the simplified model.

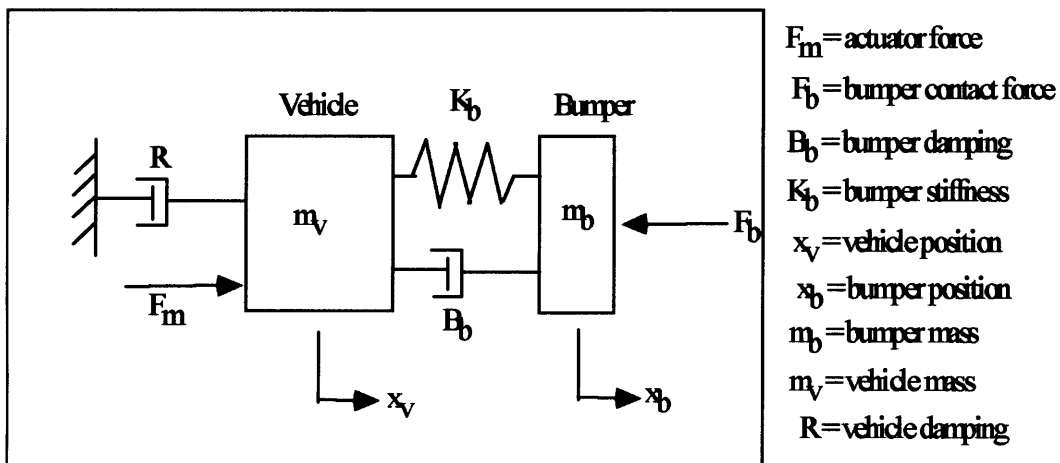


Figure 5.3.1: Simplified Model for Lateral Dynamics

The system can be further described using a bond graph model as shown in Figure 5.3.2.

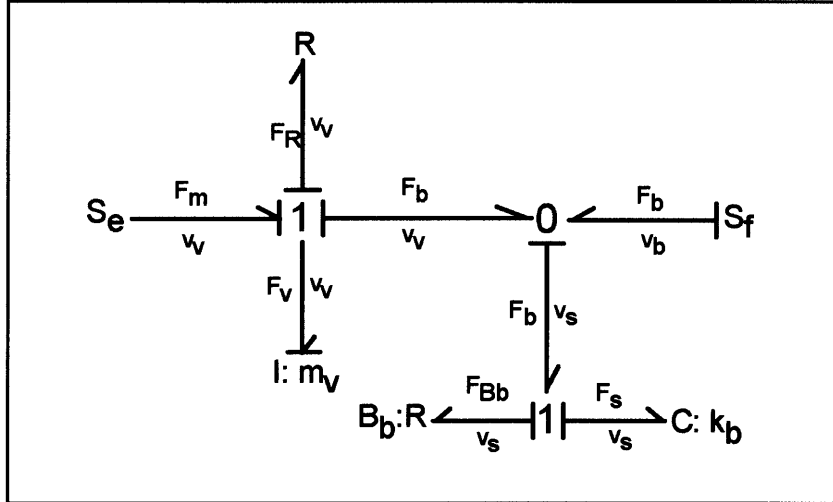


Figure 5.3.2: Bond Graph Model of Vehicle/Bumper System

The inputs to the systems are the actuator force F_m (effort source) and the bumper velocity v_b (flow source). The open loop system is of order two, and assuming B_b is negligible, state equations can be written as:

$$\begin{pmatrix} \dot{F}_b \\ \dot{v}_v \end{pmatrix} = \begin{bmatrix} 0 & K_b \\ -\frac{1}{m_v} & -\frac{R}{m_v} \end{bmatrix} \begin{pmatrix} F_b \\ v_v \end{pmatrix} + \begin{bmatrix} -K_b & 0 \\ 0 & \frac{1}{m} \end{bmatrix} \begin{pmatrix} v_b \\ F_m \end{pmatrix} \quad (18)$$

The open loop eigenvalues are given by:

$$\lambda = -\frac{R}{2m_v} \pm \frac{1}{2} \sqrt{\left(\frac{R}{m_v}\right)^2 - 4\frac{K_b}{m_v}} \quad (19)$$

They are as expected, universally stable.

5.3.2 Stiffness Control

The next step is to close the control loop around the actuator force input using the stiffness control. As defined by Whitney [3], stiffness control works by controlling the position of the vehicle in response to forces imposed by the environment. To elaborate,

we measure the force on the vehicle resulting from contact with the environment. In our case, that force will be proportional to the displacement of the bumper. We then use this force to calculate a desired vehicle deflection based on our specified vehicle stiffness. The desired vehicle deflection is then fed into the position control loop. Figure 5.3.3 shows a control block diagram of this scheme.

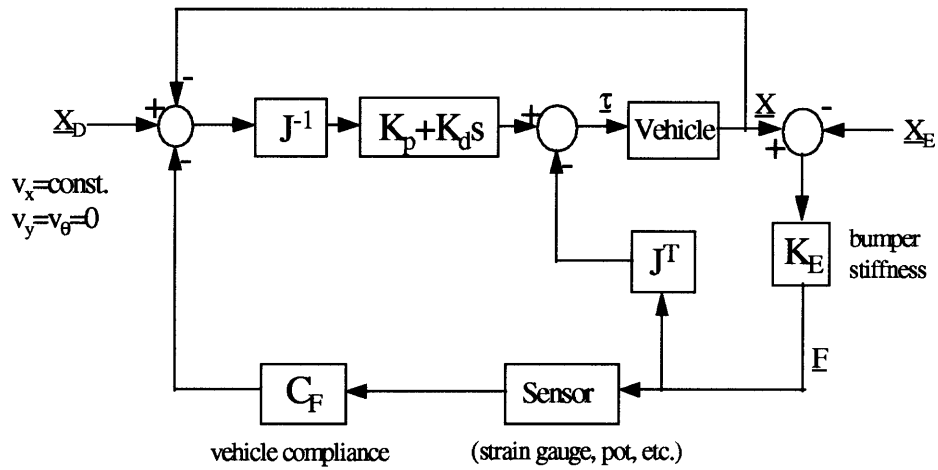


Figure 5.3.3 Stiffness Control

Applying this method to our system, we can define a position control law where:

$$F_m = K_p (x_{des} - x_v) \quad (20)$$

The desired vehicle position x is determined by the stiffness control law:

$$x_{des} = \frac{F_B}{K_v} \quad (21)$$

where K_v is the desired stiffness with which the vehicle reacts to the bumper contact force.

Combining these equations with the open loop equation results in a third order system with new closed loop state equations:

$$\begin{pmatrix} \dot{F}_b \\ \dot{v}_v \\ \dot{x}_v \end{pmatrix} = \begin{bmatrix} 0 & K_b & 0 \\ \frac{1}{m} \left(\frac{K_p}{K_v} - 1 \right) & -\frac{R}{m_v} & -\frac{K_p}{m_v} \\ 0 & 1 & 0 \end{bmatrix} \begin{pmatrix} F_b \\ v_v \\ x_v \end{pmatrix} + \begin{pmatrix} -K_b \\ 0 \\ 0 \end{pmatrix} v_b \quad (22)$$

The algebraic expressions for the eigenvalues in this case are not as helpful, but it can be shown that increasing either the position gain or bumper stiffness beyond a certain point will drive the system unstable. Likewise, decreasing the vehicle stiffness beyond a certain point will also cause instability. Choosing reasonable values for these three parameters is therefore an important selection for the control design.

5.3.3 Hybrid Stiffness/Compliant Control

We are now ready to address the issue of the singularity due to the mechanical design of the bumper. Obviously at this point, it becomes impossible to control the vehicle via force feedback from the bumpers. However, even as we approach this point, the stiffness of the bumpers increases. Beyond a certain point, this increase in bumper stiffness will drive the control loop unstable. Therefore, we should be concerned with the entire area around the singular point.

Although this may seem to be a problem at first, there is a simple solution - we can decrease the position feedback gain in this region of instability. What this essentially does is return us to the compliant control scheme mentioned earlier. By the time the instability region is reached, the vehicle has almost completely inserted itself, impact forces are at a minimum, and precise force feedback is no longer necessary.

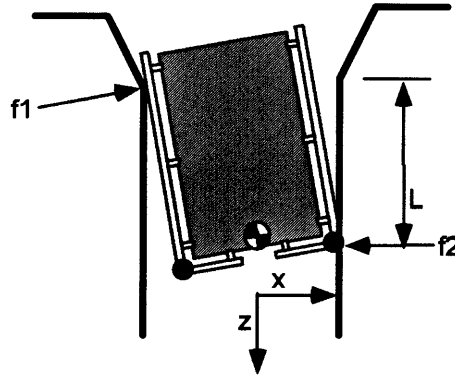


Figure 5.3.4: Near Complete Insertion

Referring to Figure 5.3.4, we can see that force f_1 is acting at a large distance from the vehicle compliance center, which we actively locate at the tip of the vehicle via the compliance control. Therefore, the force necessary to overcome wheel friction and cause rotational motion in this case is not unreasonably large. We can continue to use force feedback from f_2 to provide lateral stiffness, since this force is not acting near the singularity. Therefore, we can use a hybrid combination of stiffness control and compliant control in the latter region of the docking cycle.

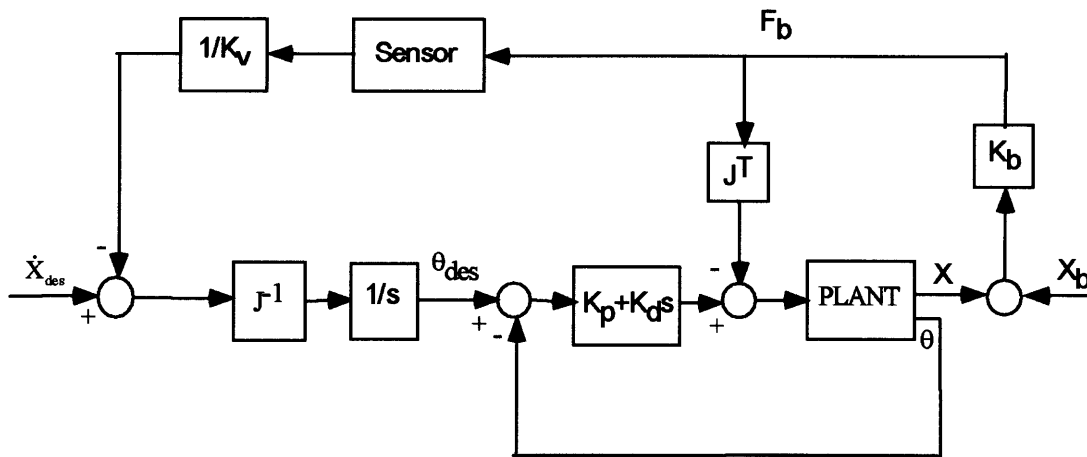


Figure 5.3.5 Hybrid Stiffness/Compliant Control

Figure 5.3.5 shows a schematic of the compliant control scheme. As before, an inner PD loop is used to control the position of the wheels. Now, the potentiometers measure the bumper forces F_b , which are used to calculate the desired deflection of the vehicle based on the specified vehicle stiffnesses K_v . The desired vehicle deflections are used to modify the nominal trajectory which is input to the position control loop. So far, this describes stiffness control alone. However, we are free to change the servo position gains. By choosing these gains according to equations (14) and (15), based on desired vehicle compliances, we implement the compliant control as described in Chapter 4.

The docking strategy is therefore performed as follows:

1. During initial approach, information from contact sensors and displacement sensors are used to determine the state of the vehicle, and guide vehicle to chamfer, if it is not already there.
2. During early-middle stages of insertion, stiffness control is used to control lateral and angular displacements while insertion velocity is constant. Sensor measurements are also used to monitor wedging forces and act appropriately.
3. Beyond a certain insertion depth L , the position gains are altered in such a way as to drastically increase the rotational compliance of the vehicle, while maintaining stiffness control in the lateral direction as well as constant velocity in the insertion direction.
4. During all stages of the docking, the compliance center is located at the front of the vehicle to minimize the chances of jamming and wedging.

Appendix D includes a program in C for performing bed docking using the hybrid stiffness/compliant control scheme.

5.4 IMPLEMENTATION

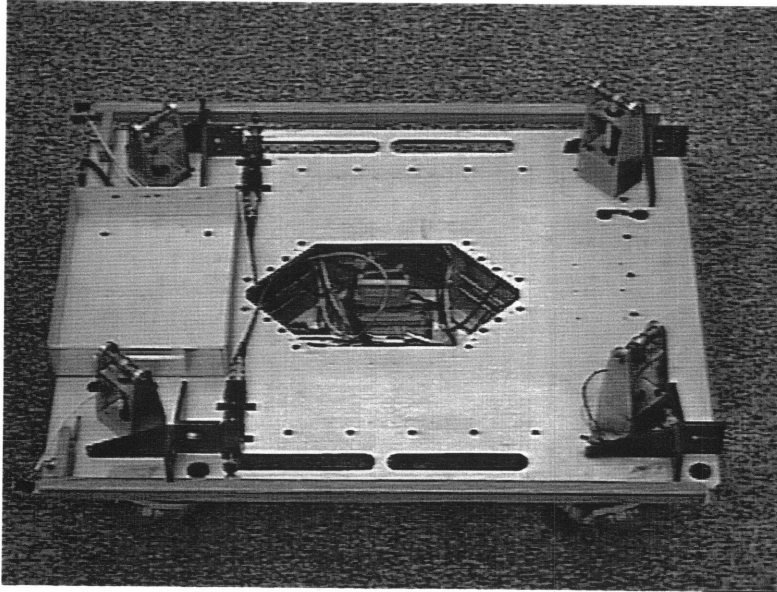


Figure 5.4.1: Bumper Prototype

A bumper prototype was constructed according to the design outlined above, and control was implemented as described. Figure 5.4.1 shows a picture of the actual prototype system. The docking process was tested experimentally for a variety of initial conditions. Figure 5.4.2 shows a plot of angular displacement vs. insertion depth using stiffness control alone. As predicted, the system goes unstable when insertion depth increases beyond a certain point. A more direct approach to avoiding this might have been to increase vehicle stiffness as insertion depth increases. Figure 5.4.3 shows a plot of this. This does get rid of instability, but causes the wheels to slip on the floor. This is obvious

from the plot since the vehicle believes it has returned to its original angle based on encoder feedback, when it really has not. In any case, wheel slippage is inevitable as the singularity point is approached.

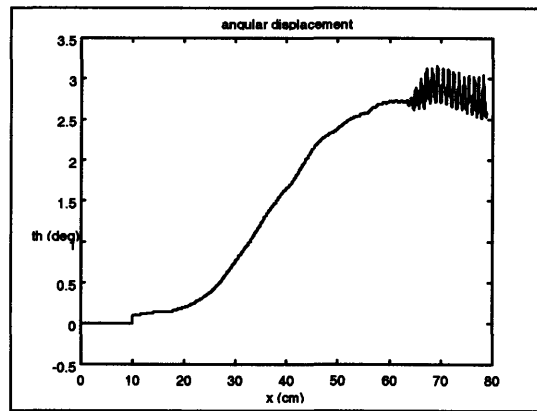


Figure 5.4.2: Results of Stiffness Control Alone

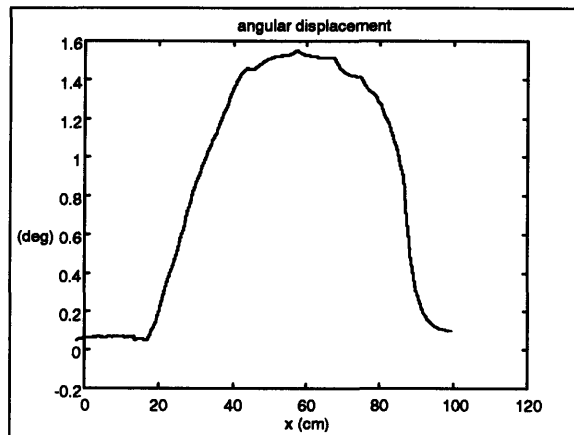


Figure 5.4.3: Results of Variable Stiffness Control

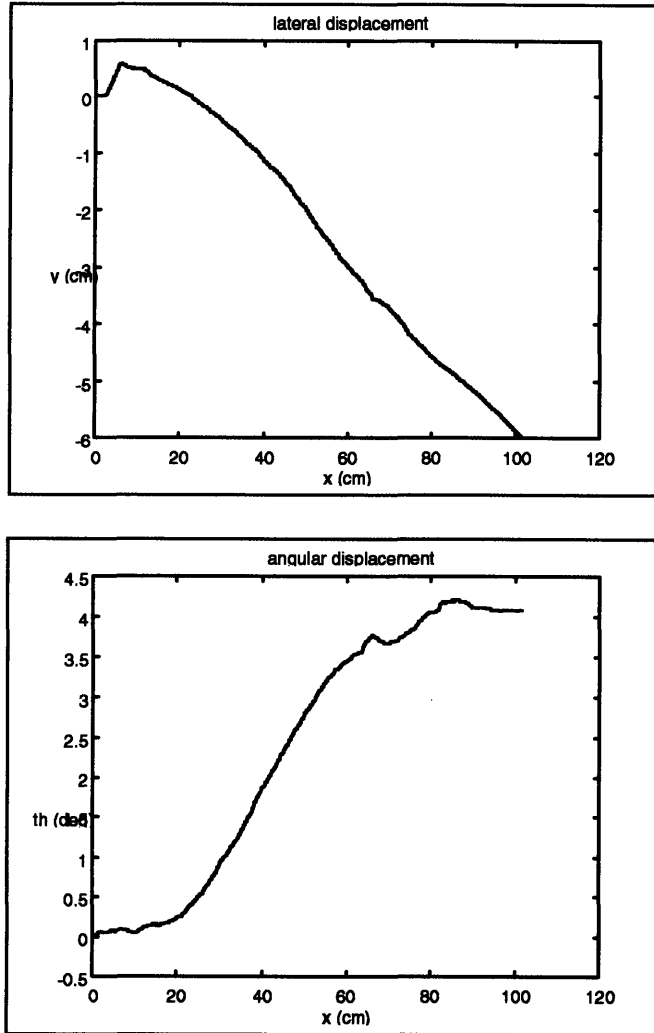


Figure 5.4.4: Hybrid Stiffness/Compliance Control

Finally, figure 5.4.4 shows a plot of both lateral and angular displacements vs. insertion depth for the hybrid stiffness/compliance control scheme. The displacements in these plots are the displacements as seen by the vehicle. Therefore, the angular displacement converges to a constant value, while the lateral displacement converges to a straight line with slope equal to the angular displacement. For this particular experiment, the initial angular error was ~ 4 degrees and the initial lateral error was ~ 2 cm. The clearance was ~ 0.1 mm for a hole width of 26 cm, resulting in a clearance ratio of $\sim 0.5\%$.

The peak in the lateral deflection plot shows the onset of two point contact at a very early stage in docking, a condition under which wedging would ordinarily occur [12]. In fact, for a coefficient of friction $\mu=0.1$, and $c=0.005$, equation (17) predicts a wedging angle of ~ 2 degrees, which is less than the initial angular error. However, the compliance of the dual bumpers serves to relieve wedging and allow docking to proceed.

The small fluctuation in angular displacement at insertion depth around 65 cm indicates the point at which the servo position gains are adjusted to drastically increase the rotational servo compliance of the vehicle, according to the compliant control method. The plot of lateral displacement barely shows any effects of this, since the lateral displacement continues to be governed by the stiffness control. As shown, the vehicle was inserted smoothly despite large tolerancing errors of the bed and chair as well as a small clearance ratio. The impacts due to contacts between the vehicle and the bed rail were almost intangible as well.

6. CONCLUSIONS

6.1 SUMMARY OF ACHIEVEMENTS AND PERFORMANCE

The two primary goals of this thesis project were to:

1. Create a new hybrid bed/wheelchair system for aiding the bedridden, by completely eliminating the need for bed to chair transfer.
2. Show that by using an omnidirectional holonomic vehicle to drive the wheelchair, we can successfully dock the chair with a bed, toilet, and other fixtures using force guided control.

While by no means have either of these goals been brought to final completion, both goals have been met with initial success. The concepts in these goals have been sufficiently developed and tested to justify their feasibility and warrant continued work.

In summary, the need for bed to chair transfer has been eliminated by designing a wheelchair which is capable of docking with a bed fixture, and thus becoming the main portion of the bed. The wheelchair is also capable of docking with a toilet and allowing the bedridden person to use the toilet without transferring from the chair. A new four-ball-wheeled omnidirectional vehicle has been designed and built for specific use as the drive platform for the wheelchair. Safety and comfort objectives were met while ensuring navigational performance of the vehicle. A prototype instrumented bumper system was designed and implemented on the vehicle in order to measure contact forces during the

docking process. Finally, a hybrid stiffness/compliant control method for force guided docking of the vehicle with a bed was successfully developed and tested.

Experimental testing of the bed docking shows that by using dual mechanically compliant bumpers, we can tolerate initial alignment errors between bed and wheelchair which would ordinarily result in wedging. By using a combination of stiffness control and servo compliance control, we can actively locate the compliance center of the vehicle at its front and avoid the occurrence of both wedging and jamming. The vehicle was docked safely and smoothly with a wide range of initial tolerancing errors and small clearance ratio. Due to the relatively low bumper compliance and the responsive nature of the stiffness control, contact forces and jerks were imperceptible to the rider.

Modeling of the stiffness controlled system and experimental results both indicate that the critical step in designing the bumper and controlling the docking process is the choice of bumper and vehicle stiffnesses. A very significant tradeoff exists between docking performance and stability of the feedback control. For low vehicle stiffness, the vehicle complies easier with the hole, but tends towards oscillatory behavior and sometimes instability. For higher vehicle stiffness, the vehicle remains stable, but becomes less responsive to force stimuli.

Most innovatively, it was found that a breakdown of the docking task shows that different phases of the docking process suggest different requirements in terms of the mechanical system and control of the vehicle. Unlike a traditional peg insertion task,

where the mechanical nature of the peg itself is often pre-determined, we have the opportunity to design the vehicle and bumper system with docking in mind. Furthermore, by instrumenting the vehicle and bumpers at the points of contact, we can obtain information about the docking process, which would not be available for an ordinary peg-in-hole task, and hopefully come to a greater understanding of the docking task. According to Whitney [12], the future of force control lies in the understanding of tasks more than in the development of more elaborate feedback methods.

6.2 RECOMMENDATIONS AND FUTURE AREAS OF WORK

Having stressed the importance of task understanding, this is where the most effort needs to be placed in the future of this project. One need is to study further the utility of the direct measurement of the contact forces and direct estimation of wedging forces via the dual bumper concept. Another need is to study the importance of being able to recognize a wide variety of contact states before and during the docking process, such as the work done in this area by McCarragher and Asada [14]. Particularly in docking processes involving humans, where safety is of utmost concern, it becomes extremely important to have a complete knowledge of the state of the docking process, and how to proceed at any given time.

More work also needs to be done in developing the RHOMBUS system in general. The toilet docking task needs to be more clearly defined, and additional functionalities of the chair should be investigated. Navigational control is also open to much development.

Central to this issue is the idea of *Human-Centered-Control*: how we can design the human-machine interface and how we can control the chair in order to maximize the comfort of the occupant. Particularly with the use of the omnidirectional holonomic vehicle, the chair is capable of much more than ordinary motions. The question is how to best make use of this advantage in both navigation and docking. These are issues which have not yet been fully explored. In conclusion, the largely unstructured nature of this project coupled with the vast amount of measurements which can be made and the extreme need for safety leave plenty of room for further developments.

APPENDIX A. CAD DRAWINGS

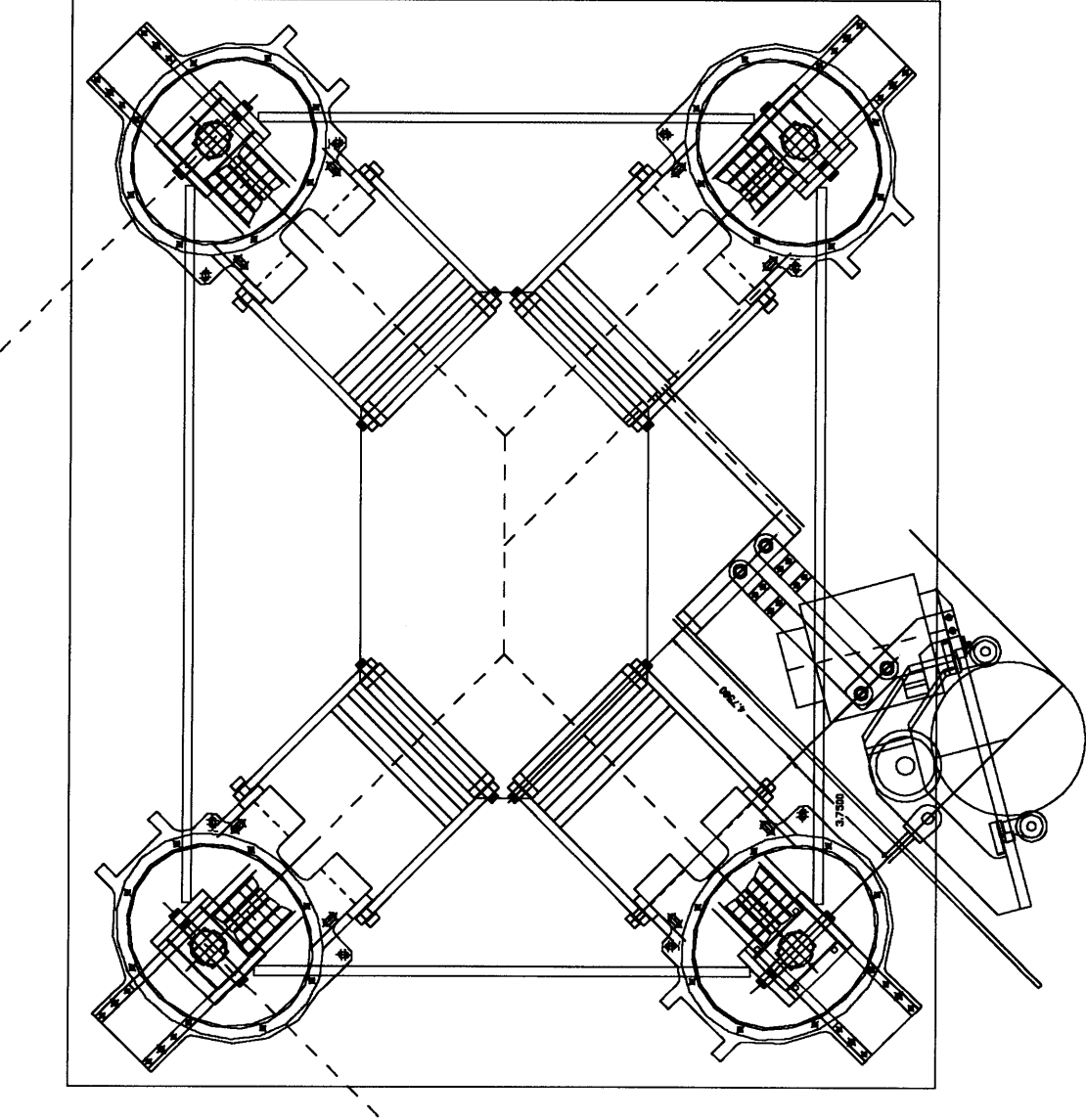


Figure A.1 Four Wheeled Vehicle Assembly Drawing

APPENDIX B. KINEMATIC DERIVATIONS

Figure B.1 shows the geometry of the ball wheel mechanism. The roller ring encircling the ball is inclined at angle ϕ as shown. For our design, $\phi = 30$ degrees. The ring is geared to the motor pinion by transmission ratio η , which can be adjusted anywhere from 24/192 to 48/192.

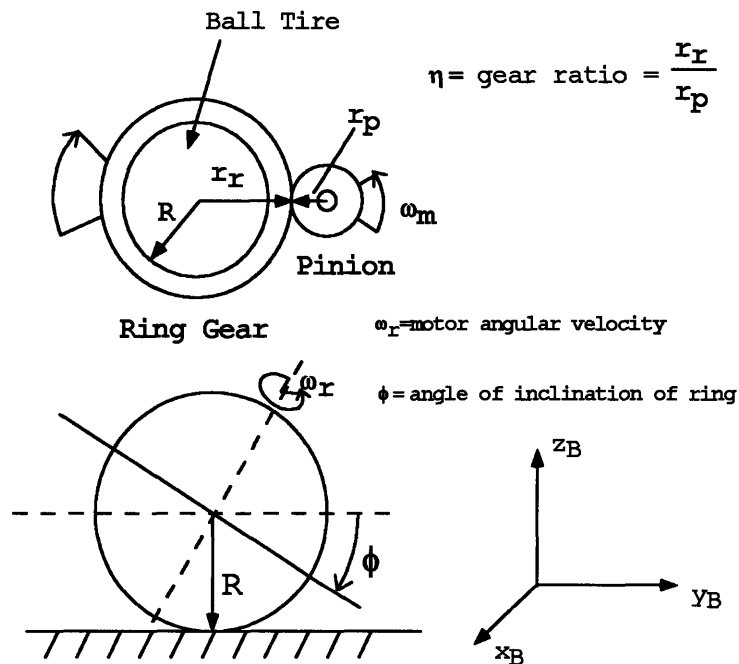


Figure B.1: Ball Wheel Geometry

Using Figure B.1, we can write the kinematics of the wheel as:

$$\omega_m = \frac{\eta \dot{x}_B}{R \sin \phi} \quad (23)$$

where x_B is the component of linear motion of the ball which induces rotation of the ring and motor. Figure B.2 shows the configuration of the four-wheeled vehicle with vectors representing the active directions of rolling for each wheel.

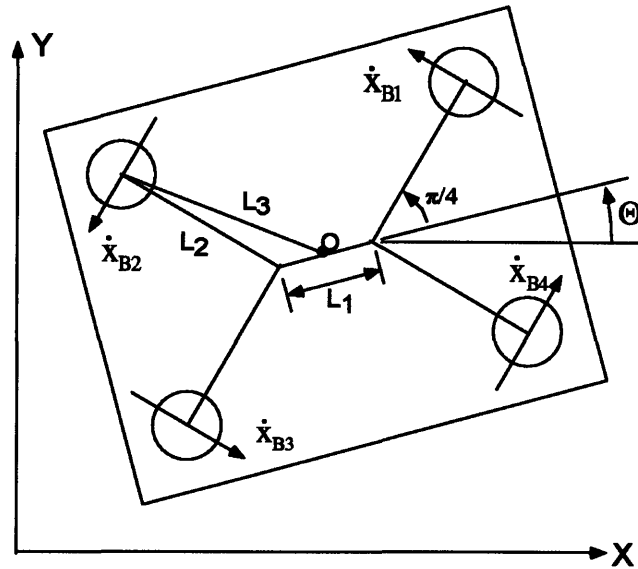


Figure B.2: Four Wheeled Configuration

The geometry can be completely described by the two lengths L_1 and L_2 , and the angles of the active directions of rolling for the wheels ($\pi/4, 3\pi/4, -3\pi/4, -\pi/4$) with respect to L_1 , which are of course symmetric. We can simplify the kinematics by defining two geometric parameters L_3 and $\alpha = \cos\psi$, as shown by Figure B.3. L_3 is the distance from center of ball to center of vehicle, and ψ is the angle between the active direction of rolling and the direction of rolling induced by rotation of the vehicle about its center O .

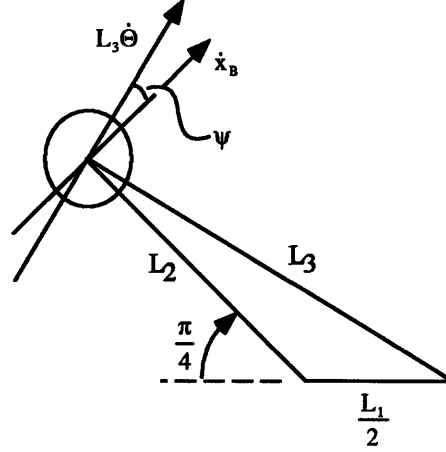


Figure B.3: Vehicle Geometry

Using the Law of Cosines, we can solve for the new parameters:

$$L_3 = \sqrt{L_2^2 + \frac{1}{4}L_1^2 + L_1L_2 \frac{\sqrt{2}}{2}} \quad (24)$$

$$\alpha = \cos \psi = \frac{L_2^2 + L_3^2 - \frac{1}{4}L_1^2}{2L_2L_3} \quad (25)$$

The complete inverse kinematics are thus given by:

$$\begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{pmatrix} = \frac{\eta}{R \sin \phi} \mathbf{J}^{-1} \mathbf{R}_Z^T(\Theta) \begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{\Theta} \end{pmatrix} \quad (26)$$

where ω are the motor velocities, η is the gear ratio, R is the ball radius, and ϕ is the wheel inclination, and $(\dot{X}, \dot{Y}, \dot{\Theta})^T$ are the velocities of the vehicle at its geometric center.

The inverse Jacobian \mathbf{J}^{-1} , is given by equation (). $\mathbf{R}_Z^T(\Theta)$ is a rotational transformation for arbitrary Θ .

$$\mathbf{J}^{-1} = \begin{bmatrix} -\sin(\pi/4) & \cos(\pi/4) & \alpha L_3 \\ -\sin(3\pi/4) & \cos(3\pi/4) & \alpha L_3 \\ -\sin(-3\pi/4) & \cos(-3\pi/4) & \alpha L_3 \\ -\sin(-\pi/4) & \cos(-\pi/4) & \alpha L_3 \end{bmatrix} = \begin{bmatrix} -\sqrt{2}/2 & \sqrt{2}/2 & \alpha L_3 \\ -\sqrt{2}/2 & -\sqrt{2}/2 & \alpha L_3 \\ \sqrt{2}/2 & -\sqrt{2}/2 & \alpha L_3 \\ \sqrt{2}/2 & \sqrt{2}/2 & \alpha L_3 \end{bmatrix} \quad (27)$$

$$\text{where } \alpha = \frac{L_2^2 + L_3^2 - \frac{1}{4}L_1^2}{2L_2L_3} \quad (28)$$

Or we can compactly write:

$$\mathbf{J}^{-1} \mathbf{R}_Z^T(\Theta) = \begin{bmatrix} -\sin(\Theta + \pi/4) & \cos(\Theta + \pi/4) & \alpha L_3 \\ -\sin(\Theta + 3\pi/4) & \cos(\Theta + 3\pi/4) & \alpha L_3 \\ -\sin(\Theta - 3\pi/4) & \cos(\Theta - 3\pi/4) & \alpha L_3 \\ -\sin(\Theta - \pi/4) & \cos(\Theta - \pi/4) & \alpha L_3 \end{bmatrix} \quad (29)$$

If we wish to specify vehicle velocities with respect to some point other than the geometric center of the vehicle, then we must post-multiply the Jacobian by a translational operator \mathbf{D} .

$$\begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{pmatrix} = \frac{\eta}{R \sin \phi} \mathbf{J}^{-1} \mathbf{D} \mathbf{R}_Z^T(\Theta) \begin{pmatrix} \dot{X} \\ \dot{Y} \\ \dot{\Theta} \end{pmatrix} \quad (30)$$

APPENDIX C. CONTROL HARDWARE

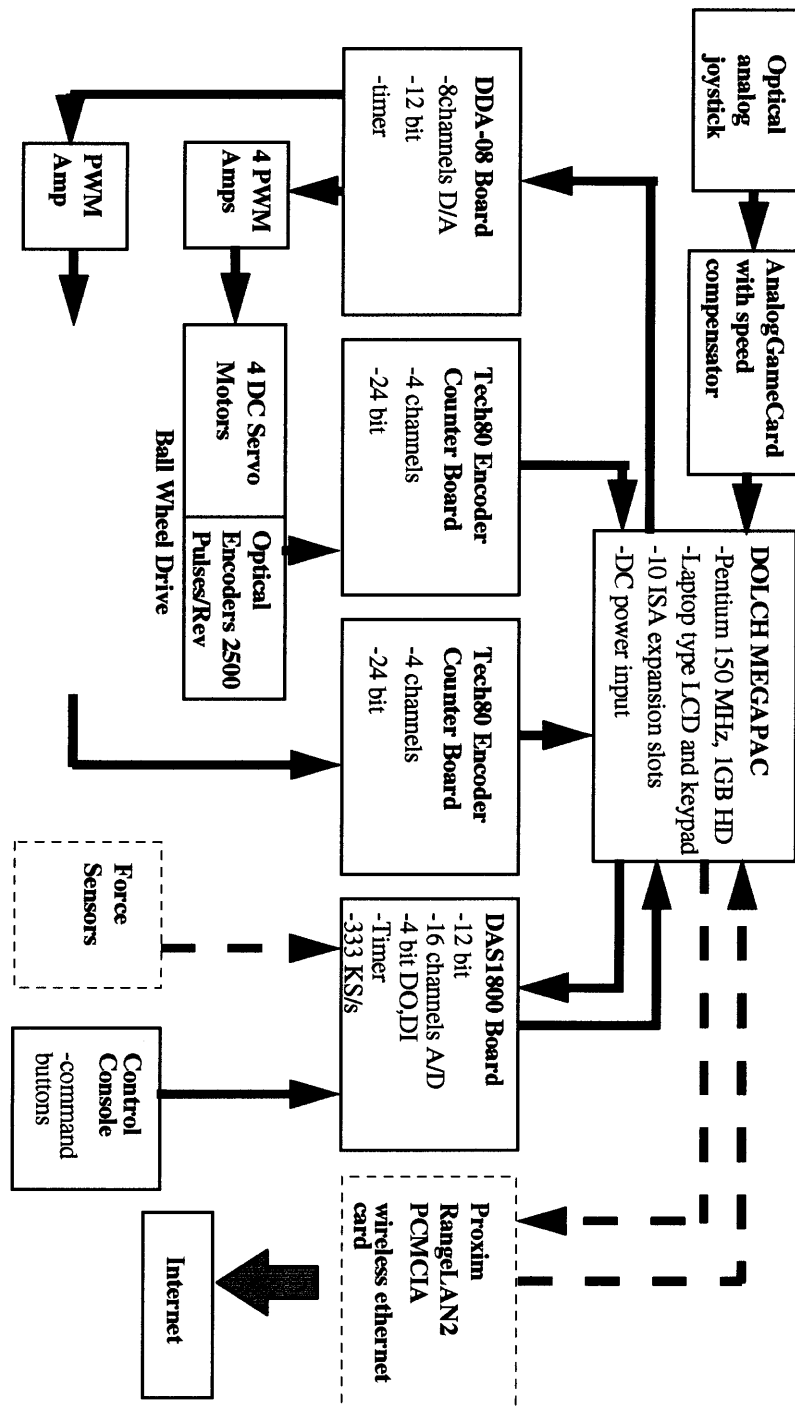


Figure C.1 Schematic of Control Hardware

APPENDIX D. C PROGRAMS

D.1 JOYSTICK CONTROL PROGRAM

```
/******  
/* 4weel2.c */  
/* */  
/* four channel, PD control using DAS1801ST, DDA08,TE 5312B; */  
/* calculations are tuned to kinematics of wheel from */  
/* omnidirectional vehicle; program will drive vehicle using */  
/* filtered input from the joystick to generate a constant */  
/* velocity trajectory */  
/* -gains are set in program */  
/* -this program modified from 4weeljoy.c to switch analog */  
/* output duties from DAS1800 to DDA08 */  
/* -all board commands are register level */  
/* */  
/* Stephen Mascaro */  
/* 10/27/96 */  
/* */  
/******  
  
#include <stdio.h>  
#include <stdlib.h>  
#include <conio.h>  
#include <string.h>  
#include <math.h>  
#include <graph.h>  
  
#include <te5312.h>  
  
#define GLOBAL -1  
#define CntrPrst 8388608L  
#define PI 3.141592654  
  
int nErr, num, i, j, c, DAvalue[8], DALow[8], DAhigh[8], quit=1, creset[4];  
float Ts=0.005; /*Sampling Period*/  
float theta[5], tau[5], theta_d[5], thetaprev[5], kp[5], kd[5];  
float thconvert, vel[5], vbar[5], vdes[5], rmax, vmax;  
float vx, vy, vth;  
const float L1=13.7855, phi = 8.51869*PI/180.;  
float alpha = 0.1; /*weighting factor for velocity calc*/  
float beta = 0.98; /*weighting factor to filter velocity input*/  
unsigned short CntrBoardAddr=0x330, DASBoardAddr=0x300, DDABoardAddr=0x310;  
long lCnt[4], ClkTck;  
int MSB,LSB,status,N,q=0,p=0,s[1000];  
  
main()  
{  
    set_up();  
    control();  
    /* dataout(); */  
}
```

```

}

set_up()
{
    _clearscreen(_GCLEARSCREEN);

    /*Set Clock Rate*/
    outp(DASBoardAddr+0X4,0); /*disable counters*/
    outp(DASBoardAddr+0X5,128); /*enable terminal count detection*/
    outp(DASBoardAddr+0X7,0); /*clear terminal count*/

    ClkTck = 5000000L*Ts;
    N = (int)(sqrt(ClkTck));
    if (N<2) N=2;
    if (N>65535) N=65535;
    MSB = (int) (N/256);
    LSB = N-256*MSB;
    outp(DASBoardAddr+0XF,116); /*load counters*/
    outp(DASBoardAddr+0XD,LSB);
    outp(DASBoardAddr+0XD,MSB);
    outp(DASBoardAddr+0XF,180);
    outp(DASBoardAddr+0XE,LSB);
    outp(DASBoardAddr+0XE,MSB);

    /*Initialize Analog Output*/
    outp(DDABoardAddr+0x6,1); /*enable auto incrementing of channels*/
    outp(DDABoardAddr+0x4,8);

    outp(DDABoardAddr+0x2,0); /*initialize DA values*/
    for (i=0;i<8;i++)
    {
        outp(DDABoardAddr,255);
        outp(DDABoardAddr+0x1,7);
    }

    outp(DDABoardAddr+0x6,1); /*enable DA*/
    outp(DDABoardAddr+0x4,24);

    outp(DDABoardAddr+0x6,2); /*select software clock*/
    outp(DDABoardAddr+0x4,3);

    /*Initialization of Encoder Counter Board*/
    InitSw(); /*Initialize Encoder Driver Software*/
    InitBoard(CntrBoardAddr); /*Initialize Encoder Board*/
    LoadCntr(GLOBAL,CntrPrst); /*Load counter with preset value*/

    /*Initialization of control parameters*/

    for (j=0;j!=4;j++)
    {
        creset[j]=0;
        vbar[j] = 0;
        thetaprev[j] = 0;
    }
}

```

```

    theta_d[j] = 0;
    vdes[j] = 0;
}

/*conversion ratio from encoder count to angular position of wheel*/
/*10000 counts per rev of motor, gear ratio = 24/192*/
thconvert = 2.*PI*24./(10000.*192.);

for (j=0;j!=4;j++)
{
    kp[j]=5;
    kd[j]=1;
}

vmax=20.0;
rmax=1.0;
}

control()
{
    outp(DASBoardAddr+0X4,4); /*enable counters (start clock)*/

    do    /*begin control loop*/
    {

        /*check clock status*/
        /*when a new clock cycle is detected, sampling begins*/

        do
        {
            status=inp(DASBoardAddr+0X7);
        } while (status==0);
        outp(DASBoardAddr+0X7,0); /*clear terminal count*/

        for (j=0;j!=4;j++)
            ICnt[j] = ReadCntr(j); /*Read encoder count*/

        for (j=0;j!=4;j++)
        {
            theta[j] = -(float)(CntrPrst-ICnt[j]-creset[j]*8000000L)*thconvert;

            /*Calculate a weighted velocity to reduce noise*/
            vel[j] = (theta[j] - thetaprev[j])/Ts;
            vbar[j] = vel[j]*(1-alpha) + alpha*vbar[j];

            tau[j] = kp[j]*(theta_d[j]-theta[j])+kd[j]*(vdes[j]-vbar[j]); /*calculate torque*/

            Dvalue[j] = (int)(tau[j]*4096/20+2047);
            if (Dvalue[j] < 0) Dvalue[j] = 0;
            if (Dvalue[j] > 4095) Dvalue[j] = 4095;

```



```

    DAhigh[j] = DValue[j]/256;
    DAlow[j]=DValue[j]-DAhigh[j]*256;
}

/*Send torque command to motor*/

outp(DDABoardAddr+0x2,0);
for (j=0;j!=4;j++)
{
    outp(DDABoardAddr,DAlow[j]);
    outp(DDABoardAddr+0x1,DAhigh[j]);
}
outp(DDABoardAddr+0x6,0); /*initiate update*/
outp(DDABoardAddr+0x4,2);

for (j=0;j!=4;j++)
{
    thetaprev[j] = theta[j];
    if (lCnt[j] > 16388608L)
    {
        LoadCntr(j,lCnt[j] - 8000000L);
        creset[j]++;
    }
    if (lCnt[j] < 388608L)
    {
        LoadCntr(j,lCnt[j] + 8000000L);
        creset[j]--;
    }
}

get_theta_d();

} while (!kbhit() && quit);

/*Stop motor after end of samples*/
outp(DDABoardAddr+0x6,1);
outp(DDABoardAddr+0x4,0); /*disable DA*/

/*Stop clock*/
outp(DASBoardAddr+0X4,0);

}

get_theta_d()
{
    short unsigned int axismask, joy;
    int k, joycount[4], cnt;
    float throttle, throtmax = 53, joymax=100;

    for (k=0;k!=4;k++)
    {
        if (k!=2)
        {

```

```

    cnt=0;
    axismask=pow(2,k);
    outp(0X201,0);
    do
    {
        cnt++;
        if (cnt > 120)
        {
            cnt= 3;
            break;
        }
    } while (axismask & inp(0X201));
    joycount[k] = cnt-3;
    _settextposition(20+k,20);
    printf("%10d",joycount[k]);
}
}

throttle = (joycount[3])/throtmax;
vy = (1 - 2*joycount[0]/joymax)*throttle*vmax;
vx = (1 - 2*joycount[1]/joymax)*throttle*vmax;
if ( (fabs(joycount[0]-joymax/2)) <= 15) vy = 0.;
if ( (fabs(joycount[1]-joymax/2)) <= 15) vx = 0.;
joy = inp(0X201);
if (!(joy & 32))
    vth = throttle*rmax;
else if (!(joy & 64))
    vth = -throttle*rmax;
else
    vth = 0.;
if (!(joy & 16))
    quit = 0;

vdes[0] = vdes[0]*beta - (1-beta)*(-vx*sqrt(2.)/2. + vy*sqrt(2.)/2. + L1*cos(phi)*vth);
vdes[1] = vdes[1]*beta - (1-beta)*(-vx*sqrt(2.)/2. - vy*sqrt(2.)/2. + L1*cos(phi)*vth);
vdes[2] = vdes[2]*beta - (1-beta)*(vx*sqrt(2.)/2. - vy*sqrt(2.)/2. + L1*cos(phi)*vth);
vdes[3] = vdes[3]*beta - (1-beta)*(vx*sqrt(2.)/2. + vy*sqrt(2.)/2. + L1*cos(phi)*vth);

for (j=0;j!=4;j++)
    theta_d[j] = theta_d[j] + vdes[j]*Ts;

return;
}

/* dataout() */
/* { */
/* char string[20]; */
/* FILE *out_file, *fopen(); */
/* Save data in a file */
/* printf("\n\nName of a file to save data: "); */
/* scanf("%s", string); */
/* out_file=fopen(string,"w"); */

```

```

/* printf("Data format in a file %s\n",string); */
/* printf(" 1st Column : 2nd Column : 3rd Column : 4th Column\n"); */
/* printf("Displacement[rad]:Velocity[rad/s]:Filtered Velocity[rad/s]:Torque [Nm]\n"); */

/* for(j=0; j<DATANUM; j++) /*
/* { /*
/* fprintf(out_file, "%lf\t%lf\t%lf\t%lf\n",thetadata[j],veldata[j],vbardata[j],taudata[j]); */
/* } /*
/* fclose(out_file); */
/*} */

```

D.2 DOCKING CONTROL PROGRAM

```

/*****
/* comply.c */
/* four channel, PD control using DAS1801ST, DDA08,TE 5312B; */
/* calculations are tuned to kinematics of wheel from */
/* omnidirectional vehicle; */
/* -program will use force feedback from dual bumpers to dock */
/* vehicle with bed */
/* -joystick initiation and cutoff */
/* -gains are set in program */
/* -this program modified from forcdock.c */
/* -all DAS/DDA board commands are register level */
/* Stephen Mascaro */
/* 1/20/97 */
*****/

#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#include <string.h>
#include <math.h>
#include <graph.h>

#include <te5312.h>

#define GLOBAL -1
#define CntrPrst 8388608L
#define PI 3.141592654
#define DATANUM 5000

int i=0, j, k,DAvalue[8], DAlow[8], DAhigh[8], quit=1, creset[4];
int MSB,LSB,status,N,p=0,q=0,s[1000], ADVvalue[3], DIvalue;

```

```

unsigned short CntrBoardAddr=0x330, DASBoardAddr=0x300, DDABoardAddr=0x310;
long ICnt[4], ClkTck;

const float Ts=0.005; /*Sampling Period*/
const float L1 =13.7855, phi = 8.51869*PI/180., R = 2.125;
float Ls = 28.75, Lp = 21.5, Lc = -31., rmax=0.0, vmax=2.0;
const float Ks = 3.22, Kvy = 1.0 , Kvth = 1500., accel=20;
const float alpha = 0.1; /*weighting factor for velocity calc*/
const float beta = 0.98; /*weighting factor to filter velocity input*/
const float gamma = 0.995; /*weighting factor for A/D filter*/

float theta[5], tau[5], theta_d[5], thetaprev[5], kd[5];
float thconvert, bumpcon, d, vel[5], vbar[5], vdes[5];
float vx=0., vy=0., vth=0., ydes, thdes, vdx=0., vdy=0., vdt=0.;
float x = 0.0, y =0.0, th =0.0 ,t = 0.0;
float T1,T2,T3, Fb, Mb, Lf1, Lf2, xpot1, xpot2, Lf1a, Lf2a;
float Volts[3], Vref[3];
float a,b1, b2, b3, b4, Kw[5][5], Kd[5][5], Kx=75., Ky=75., Kth=10000.;
float xdata[DATANUM],ydata[DATANUM],thdata[DATANUM];
float Fbdata[DATANUM],Mbdata[DATANUM],tdata[DATANUM];

char string[20];

main()
{
    set_up();
    trajplan();
    control();
    dataout();
}

set_up()
{
    _clearscreen(_GCLEARSCREEN);

/*Set Clock Rate*/
    outp(DASBoardAddr+0X4,0); /*disable counters*/
    outp(DASBoardAddr+0X5,128); /*enable terminal count detection*/
    outp(DASBoardAddr+0X7,0); /*clear terminal count*/

    ClkTck = 5000000L*Ts;
    N = (int)(sqrt(ClkTck));
    if (N<2) N=2;
    if (N>65535) N=65535;
    MSB = (int) (N/256);
    LSB = N-256*MSB;
    outp(DASBoardAddr+0XF,116); /*load counters*/
    outp(DASBoardAddr+0XD,LSB);
    outp(DASBoardAddr+0XD,MSB);
    outp(DASBoardAddr+0XF,180);
    outp(DASBoardAddr+0XE,LSB);
    outp(DASBoardAddr+0XE,MSB);

/*Initialize Analog Output*/

```

```

outp(DDABoardAddr+0x6,1); /*enable auto incrementing of channels*/
outp(DDABoardAddr+0x4,8);

outp(DDABoardAddr+0x2,0); /*initialize DA values*/
for (i=0;i<8;i++)
{
    outp(DDABoardAddr,255);
    outp(DDABoardAddr+0x1,7);
}

outp(DDABoardAddr+0x6,1); /*enable DA*/
outp(DDABoardAddr+0x4,24);

outp(DDABoardAddr+0x6,2); /*select software clock*/
outp(DDABoardAddr+0x4,3);

/*Initialize Analog Input*/
outp(DASBoardAddr+0x2,0x1); /*set data select register to point to QRAM*/
outp(DASBoardAddr+0xA,0x2); /*initialize QRAM to scan channels 0-2*/
outpw(DASBoardAddr,0x0); /*set chan 0 to gain of 1*/
outpw(DASBoardAddr,0x1); /*set chan 1 to gain of 1*/
outpw(DASBoardAddr,0x2); /*set chan 2 to gain of 1*/
outp(DASBoardAddr+0xA,0x2); /*reinitialize QRAM to starting address*/

outp(DASBoardAddr+0x6,0xC8); /*set A/D operating modes*/
outp(DASBoardAddr+0x4,0x1); /*enable A/D FIFO*/
outp(DASBoardAddr+0x2,0x0); /*set data select register to point to A/D FIFO*/

outp(DASBoardAddr+0x7,0x80); /*enable A/D conversions*/

/*Initialization of Encoder Counter Board*/
InitSw(); /*Initialize Encoder Driver Software*/
InitBoard(CntrBoardAddr); /*Initialize Encoder Board*/
LoadCntr(GLOBAL,CntrPrst); /*Load counter with preset value*/

/*Initialization of control parameters*/

for (j=0;j!=4;j++)
{
    creset[j]=0;
    vbar[j] = 0;
    thetaprev[j] = 0;
    theta_d[j] = 0;
    vdes[j] = 0;
}

/*conversion ratio from encoder count to angular position of wheel*/
/*10000 counts per rev of motor, gear ratio = 24/192*/
thconvert = 2.*PI*24./(10000.*192.);

/*bumper conversion constant*/
bumpcon = 2*Ks* $L_s$ * $L_s$ / $L_p$ ;

a =  $L_1$ *cos(phi);

```

```

d = 15 + Lc; /*distance between vehicle centers of symmetry & compliance*/

printf("*** Force Guided Docking of 4 wheeled vehicle ***\n\n");
printf("\n\nName of a file to save data: ");
scanf("%s", string);

b1 = (sqrt(2.)/4.+d/(4.*a))*(sqrt(2.)/4.+d/(4.*a));
b2 = 1./(16.*a*a);
b3 = (-1./8.+d*d*b2);
b4 = (-sqrt(2.)/4.+d/(4.*a))*(-sqrt(2.)/4.+d/(4.*a));

get_gains();

/*Find unperturbed bumper readings to use as reference levels*/
for (j=0; j!=3; j++)
{
    outpw(DASBoardAddr,0); /*initiate an A/D conversion*/
    while(!(64 & inp(DASBoardAddr+0x7)));
    ADValue[j] = inpw(DASBoardAddr);
    Vref[j] = (float)(ADValue[j]*5./4096.);
    Volts[j]=Vref[j];
}

}

control()
{
    while (inp(0x201) & 128); /*joystick initiation - thumb button*/
    outp(DASBoardAddr+0X4,5); /*enable counters (start clock) and preserve FIFO enable*/

    do /*begin control loop*/
    {

        /*check clock status*/
        /*when a new clock cycle is detected, sampling begins*/
        do
        {
            status=inp(DASBoardAddr+0X7);
        } while (status== 0x80);

        outp(DASBoardAddr+0X7,0x80); /*clear terminal count and preserve A/D enable*/

        for (j=0;j!=4;j++)
            ICnt[j] = ReadCntr(j); /*Read encoder count*/

        for (j=0; j!=3; j++) /*get bumper readings*/
        {
            outpw(DASBoardAddr,0); /*initiate an A/D conversion*/
            while(!(64 & inp(DASBoardAddr+0x7)));
            ADValue[j] = inpw(DASBoardAddr);
            Volts[j] = gamma*Volts[j]+(1-gamma)*((float)(ADValue[j]*5./4096.));
        }
    }
}

```

```

Dlvalue=inp(DASBoardAddr+0x3); /*get limit switch readings*/

for (j=0;j!=4;j++)
{
theta[j] = -(float)(CntrPrst-lCnt[j]-creset[j]*8000000L)*thconvert;

/*Calculate a weighted velocity to reduce noise*/
vel[j] = (theta[j] - thetaprev[j])/Ts;
vbar[j] = vel[j]*(1-alpha) + alpha*vbar[j];

tau[j] = 0; /*calculate torque*/
for (k=0;k!=4;k++)
tau[j] = tau[j] + Kw[j][k]*(theta_d[k]-theta[k]); /* + Kd[j][k]*(vdes[k]-vbar[k]);*/
tau[j] = tau[j] + kd[j]*(vdes[j]-vbar[j]);

/*calculate D/A outputs*/
DAvalue[j] = (int)(tau[j]*4096/20+2047);
if (DAvalue[j] < 0) DAvalue[j] = 0;
if (DAvalue[j] > 4095) DAvalue[j] = 4095;
DAhigh[j] = DAvalue[j]/256;
DALow[j]=DAvalue[j]-DAhigh[j]*256;
}

/*Send torque command to motor*/
outp(DDABoardAddr+0x2,0);
for (j=0;j!=4;j++)
{
outp(DDABoardAddr,DALow[j]);
outp(DDABoardAddr+0x1,DAhigh[j]);
}
outp(DDABoardAddr+0x6,0); /*initiate update*/
outp(DDABoardAddr+0x4,2);

/*keep track of counter resets*/
for (j=0;j!=4;j++)
{
thetaprev[j] = theta[j];
if (lCnt[j] > 16388608L)
{
LoadCntr(j,lCnt[j] - 8000000L);
creset[j]++;
}
if (lCnt[j] < 388608L)
{
LoadCntr(j,lCnt[j] + 8000000L);
creset[j]--;
}
}

get_theta_d();
get_gains();

if (!(inp(0x201) & 16)) quit = 0; /*joystick cutoff-trigger*/

```

```

    i++;

} while (!kbhit() && quit);

/*Stop motor after end of samples*/
    outp(DDABoardAddr+0x6,1);
    outp(DDABoardAddr+0x4,0); /*disable DA*/

/*disable A/D*/
    outp(DASBoardAddr+0x7,0);
/*Stop clock*/
    outp(DASBoardAddr+0x4,0);

}

get_theta_d()
{
    if (t<T1)
        vdx = -accel*t;
    else if (t<T2)
        vdx = -vmax;
    else if (t<T3)
        vdx = -vmax+accel*(t-T2);
    else
    {
        vdx = 0;
        quit=0;
    }

    xpot1 = (Volts[1]-Vref[1])/Vref[0];
    xpot2 = (Volts[2]-Vref[2])/Vref[0];

    if (Dival & 1)
        {Lf1a = 33. + x;
        Lf1 = 10.;}
    else
        Lf1 = 33.;
    if (Lf1a < 5.)
        Lf1a=5;

    if (Dival & 2)
        { Lf2a = 31. + x;
        Lf2 = 10.;}
    else
        Lf2 = 31.;
    if (Lf2a<5.)
        Lf2a = 5.;

    Fb = bumpcon*(xpot2/31. - xpot1/31.); /*synthetic forces*/

```



```

Mb = -bumpcon*((1+Lc/Lf2a)*xpot2 - (1+Lc/Lf1a)*xpot1);

ydes = Fb/Kvy;
thdes = Mb/Kvth;

Fb = bumpcon*(xpot2/Lf2a - xpot1/Lf1a); /*actual forces*/
Mb = -bumpcon*((1+Lc/Lf2a)*xpot2 - (1+Lc/Lf1a)*xpot1);

vdy = (ydes-y)/Ts;
vdth = (thdes-th)/Ts;

vdes[0] = -sin(th+PI/4)*vdx + cos(th+PI/4)*vdy + (a-d*cos(PI/4))*vdth;
vdes[1] = -sin(th+3*PI/4)*vdx + cos(th+3*PI/4)*vdy + (a-d*cos(3*PI/4))*vdth;
vdes[2] = -sin(th-3*PI/4)*vdx + cos(th-3*PI/4)*vdy + (a-d*cos(-3*PI/4))*vdth;
vdes[3] = -sin(th-PI/4)*vdx + cos(th-PI/4)*vdy + (a-d*cos(-PI/4))*vdth;

for (j=0;j!=4;j++)
{
vdes[j] = -vdes[j]*2.0/R;
theta_d[j] = theta_d[j] + vdes[j]*Ts;
}

upd_glob_pos();

tdata[i]=t;
Fbdata[i]=Fb;
Mpdata[i]=Mb;

th = th + vdth*Ts;
x = x + vdx*Ts;
y = y + vdy*Ts;
t = t + Ts;

return;
}

trajplan()
{
float D1=40, Tblend, Dblend;

Tblend = vmax/accel;
Dblend = 0.5*accel*Tblend*Tblend;

T1 = Tblend;
T2 = T1 + (D1-2*Dblend)/vmax;
T3 = T2 + Tblend;
}

get_gains()
{

```

```

if ( x < -25.)
    Kth = 0;

Kw[0][0] = (Kx/8. + Ky*b1 + Kth*b2)*R*R/4.;
Kw[0][1] = (Kx/8. + Ky*b3 + Kth*b2)*R*R/4.;
Kw[0][2] = (-Kx/8. + Ky*b3 + Kth*b2)*R*R/4.;
Kw[0][3] = (-Kx/8. + Ky*b1 + Kth*b2)*R*R/4.;
Kw[1][0] = Kw[0][1];
Kw[1][1] = (Kx/8. + Ky*b4 + Kth*b2)*R*R/4.;
Kw[1][2] = (-Kx/8. + Ky*b4 + Kth*b2)*R*R/4.;
Kw[1][3] = (-Kx/8. + Ky*b3 + Kth*b2)*R*R/4.;
Kw[2][0] = Kw[0][2];
Kw[2][1] = Kw[1][2];
Kw[2][2] = (Kx/8. + Ky*b4 + Kth*b2)*R*R/4.;
Kw[2][3] = (Kx/8. + Ky*b3 + Kth*b2)*R*R/4.;
Kw[3][0] = Kw[0][3];
Kw[3][1] = Kw[1][3];
Kw[3][2] = Kw[2][3];
Kw[3][3] = (Kx/8. + Ky*b1 + Kth*b2)*R*R/4.;

for (j=0;j!=4;j++) /*set PD gains*/
{
    kd[j] = .5;
}

}

upd_glob_pos() /*use forward kinematics to update global position*/
{
    float vx1,vx2,vx3,vx4,vy1,vy2,vy3,vy4;

    vth = (vel[0]+vel[1]+vel[2]+vel[3])/(4.*a);
    thdata[i] = thdata[i-1] - vth*Ts/2.*R/2.;
    vx1 = (-sin(thdata[i]+PI/4.)/2.-d*sin(thdata[i])/(4.*a))*vel[0];
    vx2 = (-sin(thdata[i]+3.*PI/4.)/2.-d*sin(thdata[i])/(4.*a))*vel[1];
    vx3 = (-sin(thdata[i]-3.*PI/4.)/2.-d*sin(thdata[i])/(4.*a))*vel[2];
    vx4 = (-sin(thdata[i]-PI/4.)/2.-d*sin(thdata[i])/(4.*a))*vel[3];
    vx = vx1+vx2+vx3+vx4;
    vy1 = (cos(thdata[i]+PI/4.)/2.+d*cos(thdata[i])/(4.*a))*vel[0];
    vy2 = (cos(thdata[i]+3.*PI/4.)/2.+d*cos(thdata[i])/(4.*a))*vel[1];
    vy3 = (cos(thdata[i]-3.*PI/4.)/2.+d*cos(thdata[i])/(4.*a))*vel[2];
    vy4 = (cos(thdata[i]-PI/4.)/2.+d*cos(thdata[i])/(4.*a))*vel[3];
    vy = vy1+vy2+vy3+vy4;

    xdata[i] = xdata[i-1] - vx*Ts*R/2.;
    ydata[i] = ydata[i-1] - vy*Ts*R/2.;
    thdata[i] = thdata[i] - vth*Ts/2.*R/2.;

}

dataout()

```

```

{
    FILE *out_file, *fopen();
    /*Save data in a file*/
    out_file=fopen(string,"w");
    printf("Data format in a file %s\n",string);
    printf("1st Column:2nd Column:3rd Column:4th Column:5th column:6th column\n");
    printf(" t(sec)  x(in)  y(in)  th (rad)  Fb(lb)  Mb(lb-in) ");
    for(j=0; j!=i; j++)
    {
        fprintf(out_file,
"%lf\t%lf\t%lf\t%lf\t%lf\t%lf\n",tdata[j],xdata[j],ydata[j],thdata[j],Fbdata[j],Mbdata[j]);
    }
    fclose(out_file);
}

```

REFERENCES

- [1] Khatchadourian, Rahman, Harwin, "A Technique for Robotic Assisted Transfer for People with Motor Difficulties," *IEEE Int. Conference on Intelligent Robotics and Systems*, 1994.
- [2] Whitney, D.E., "Quasi-Static Assembly of Compliantly Supported Rigid Parts", *ASME Journal of Dynamic Systems, Measurement and Control*, March 1982.
- [3] Whitney, D.E., "Historical Perspective and State of the Art in Robot Force Control," *Int. Journal of Robotics Research*, 6(1), 1997.
- [4] Drake, S.K., and Simunovic, S.N. "The Use of Compliance in a Robot Assembly System," *Preprints, IFAC Symp. Information and Contr. Problems in Manufacturing Technology*, 1977.
- [5] Salisbury, J.K., "Active Stiffness Control of a Manipulator in Cartesian Coordinates," *Proc. 19th IEEE Conf. on Decision and Control*, Dec 1980.
- [6] Salisbury, J.K., and Craig, J.J., "Articulated Hands: Force Control and Kinematic Issues," *Int. Journal of Robotics Research*, 1(1):4-17, 1982.
- [7] Hogan, N., "Control of Mechanical Impedance of Prosthetic Arms," *Proc. of 1980 JACC*.
- [8] Spano, J., "Design and Control of a Reconfigurable Bed/Chair System with Body Pressure Sensing," S.M. thesis, Massachusetts Institute of Technology, 1997.
- [9] West, M., Asada H., "Design of a Holonomic Omnidirectional Vehicle," *IEEE Int. Conference on Robotics and Automation*, May 1992.

- [10] Hirose S., Amano S., "The VUTON: High Payload Efficiency Holonomic Omnidirectional Vehicle," *6th Int. Symposium on Robotics Research*, October 1993.
- [11] Killough S.M., Pin F.G., "Design of an Omnidirectional and Holonomic Wheeled Platform Prototype," *IEEE Int. Conference on Robotics and Automation*, May 1992.
- [12] West, M., Asada H., "Design and Control of Ball Wheel Omnidirectional Vehicles," *IEEE Int. Conference on Robotics and Automation*, May 1995.
- [13] Walsh, G., Tilbury, D., Sastry, S., Murray, R., Laumond, J.P., "Stabilization of Trajectories for Systems with Nonholonomic Constraints," *IEEE Int. Conference on Robotics and Automation*, May 1992.
- [14] McCarragher, B.J., Asada H., "The Discrete Event Control of Robotic Assembly Tasks," *ASME Journal of Dynamic Systems, Measurement, and Control*, Sept. 1995.