Kinematic Modeling of Wheeled Mobile Robots

Patrick F. Muir and Charles P. Neuman

CMU-RI-TR-86-12

Department of Electrical and Computer Engineering
The Robotics Institute
Carnegie-Mellon University
Pittsburgh, Pennsylvania 15213

June 1986

This research has been supported by the Office of Naval Research under Contract N00014-81-K-0503 and the Department of Electrical and Computer Engineering, Carnegie-Mellon University.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (*When Data Entered*)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: CMU-RI-TR-86-12

2. GOVT ACCESSION NO.:

3. RECIPIENT'S CATALOG NUMBER:

4. TITLE (*and Subtitle*): KINEMATIC MODELING OF WHEELED MOBILE ROBOTS

5. TYPE OF REPORT & PERIOD COVERED: Technical Report, 5/84 - 5/86

6. PERFORMING ORG. REPORT NUMBER:

7. AUTHOR (*s*): Patrick F. Muir and Charles P. Neuman

8. CONTRACT OR GRANT NUMBER (*s*): N00014-81-K-0503

9. PERFORMING ORGANIZATION NAME AND ADDRESS: The Robotics Institute, Mobile Robot Lab; and The Department of Electrical and Computer Engineering, Carnegie-Meloon University, Schenley Park, Pittsburgh, PA 15213

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: Work Unit NR 610-001

11. CONTROLLING OFFICE NAME AND ADDRESS: Office of Naval Research, Dr. Alan R. Meyrowitz, Code 433, 800 N. Quincy St., Arlington, VA 22217

12. REPORT DATE: June 1986

13. NUMBER OF PAGES: 126

14. MONITORING AGENCY NAME & ADDRESS (*if different form Controlling Office*):

15. SECURITY CLASS. (*of this report*): Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE: None

16. DISTRIBUTION STATEMENT (*of this Report*): Unlimited

17. DISTIBUTION STATEMENT (*of the abstract entered in Block 20; if different from Report*): Unlimited

18. SUPPLEMENTARY NOTES:

19. KEY WORDS (*Continue on reverse side if necessary and identify by block number*): Mobile Robot, Kinematics, Wheeled Robot, Forward Solution, Inverse Solution, Mobility Characteristics, Actuation Characteristics, Sensing Characteristics, Robot Design, Dead Reckoning, Kinematic Control, Slip Detection, Uranus

20. ABSTRACT (*Continue on reverse side if necessary and identify by block number*): We formulate the kinematic equations-of-motion of wheeled mobile robots incorporating conventional, omnidirectional, and ball wheels. While our approach parallels the kinematic modeling of stationary manipulators, we extend the methodology to accomodate such special characteristics of wheeled mobile robots as multiple closed-link chains, higher-pair contact points between a wheel and a surface, and unactuated and unsensed wheel degrees-of-freedom. We survey existing wheeled mobile robots to motivate our development. To communicate the kinematic features of wheeled mobile robots, we introduce a diagrammatic convention and nomenclature. We apply the Sheth-Uicker convention to assign coordinate axes and develop

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-014-6601

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20.) a matrix coordinate transformation algebra to derive the equations-of-motion. A wheel Jacobian matrix is formulated to relate the motions of each wheel to the motions of the robot. We combine the individual wheel equations to form the composite robot equation-of-motion. We calculate the sensed forward and actuated inverse solutions and interpret the conditions which guarantee their existance. We interpret the properties of the composite robot equation to characterize the mobility of a wheeled mobile robot according to the mobility characterization tree. Similary, we apply actuation and sensing characterization trees to delineate the robot motions producible by the wheel actuators and discernable by the wheel sensors, respectively. We apply our kinematic model to design, kinematics-based control, dead-reckoning, and wheel slip detection. To illustrate the development, we formulate and interpret the kinematic equations-of-motion of six prototype wheeled mobile robots.

Table of Contents

1. Introduction 1
2. Survey of Kinematic Configurations 5
3. Wheel Types 11
4. Kinematic Modeling 14
   4.1. Introduction 14
   4.2. Definitions And Assumptions 14
   4.3. Coordinate System Assignments 16
      4.3.1. Sheth-Uicker Convention 16
      4.3.2. WMR Coordinate Systems 17
      4.3.3. Instantaneously Coincident Coordinate Systems 19
   4.4. Transformation Matrices 22
   4.5. Matrix Coordinate Transformation Algebra 29
   4.6. Position Kinematics 31
   4.7. Velocity Kinematics 33
      4.7.1. Introduction 33
      4.7.2. Point Velocities 34
      4.7.3. Wheel Jacobian Matrix 35
      4.7.4. Tranforming Robot Velocities 37
   4.8. Acceleration Kinematics 39
   4.9. Summary 40
5. The Composite Robot Equation 42
   5.1. Introduction 42
   5.2. Formulation of the Composite Robot Equation 43
   5.3. Solution of **Ax = By** 43
   5.4. Robot Mobility Characteristics 46
   5.5. Actuated Inverse Solution 52
   5.6. Robot Actuation Characteristics 54
   5.7. Sensed Forward Solution 57
   5.8. Robot Sensing Characteristics 59
   5.9. Conclusions 62
6. Applications 64
   6.1. Introduction 64
   6.2. Design 64
   6.3. Dead Reckoning 66
   6.4. Kinematics-Based Feedback Control 68

6.5. Wheel Slip Detection 70
6.6. Summary 71
7. Examples 73
7.1. Introduction 73
7.2. Unimation Robot 73
7.3. Newt 77
7.4. Uranus 81
7.5. Neptune 87
7.6. Rover 89
7.7. Stanford Cart 92
7.8. Conclusions 95
8. Conclusions 97
9. Continuing Research 102
10. References 104
A1. Appendix 1: A Nomenclature and Symbolic Representation of WMRs 110
A1.1. Introduction 110
A1.2. Symbolic Representation Rules 111
A1.3. Nomenclature Rules 112
A1.4. Examples 114
A2. Appendix 2: Symbol Tables 116
A3. Appendix 3: Wheel Jacobian Matrices 118
A3.1. Introduction 118
A3.2. Conventional Non-Steered Wheel 118
A3.3. Conventional Steered Wheel 119
A3.4. Omnidirectional Wheel 120
A3.5. Ball Wheel 120
A4. Appendix 4: Actuated Inverse Solution Matrix Calculations 122
A5. Appendix 5: Sensed Forward Solution Matrix Calculations 125

List of Figures

Figure 2.1 Kinematic Representations of Shakey, Newt, and Topo 5
Figure 2.2 Kinematic Representations of Terregator and Gemini 6
Figure 2.3 Kinematic Representations of Neptune and Pluto 6
Figure 2.4 Kinematic Representations of the Stanford Cart, the JPL Rover, and Kludge 7
Figure 2.5 Kinematic Representations of Hybrid Spider Drive and Hybrid Locomotion Vehicle .. 8
Figure 2.6 Kinematic Representations of Uranus and the Unimation Robot 9
Figure 3.1 Conventional, Omnidirectional, and Ball Wheels 11
Figure 3.2 Wheel Equations-of-Motion 12
Figure 3.3 Omnidirectional Wheel (Rollers at 45°) 13
Figure 4.3.1 Planar Pair Model of a Wheel 17
Figure 4.3.2 Placement of Coordinate Systems on a WMR 18
Figure 4.3.3 Ball in Motion Before Instantaneous Coincidence 19
Figure 4.3.4 Ball in Motion at Instantaneous Coincidence 19
Figure 4.3.5 Coordinate System $\dot{R}$ in Motion 20
Figure 4.6.1 Transformation Graph of a WMR 32
Figure 4.6.2 Point Mass in the Steering Link 33
Figure 5.3.1 The Solution Tree for the Vector $\mathbf{x}$ in (5.3.1) 46
Figure 5.4.1 The Solution Tree for the Robot Velocity Vector $\dot{\mathbf{p}}$ 47
Figure 5.4.2 The Mobility Characterization Tree 49
Figure 5.6.1 The Actuation Chacterization Tree 56
Figure 5.8.1 The Sensing Characterization Tree 61
Figure 6.4.1 Kinematics-Based WMR Control System 69
Figure 7.2.1 Coordinate System Assignments for the Unimation Robot 74
Figure 7.3.1 Coordinate System Assignments for Newt 77
Figure 7.4.1 Coordinate System Assignments for Uranus 82
Figure 7.4.2 Uranus with an Inadequate Actuation Structure 84
Figure 7.4.3 Converting Uranus into a Robust Actuation Structure 85
Figure 7.5.1 Coordinate System Assignemnts for Neptune 87
Figure 7.6.1 Coordinate System Assignments for Rover 91
Figure 7.7.1 Coordinate System Assignments for the Stanford Cart 92

List of Tables

Table 4.3.1 Coordinate System Assignments 21
Table 4.4.1 Scalar Rotational and Translational Displacements 25
Table 4.4.2 Transformation Matrices of the WMR Model 27
Table 4.4.3 Transformation Matrix Time-Derivatives 28
Table 4.4.4 Transformation Matrix Second Time-Derivatives 28
Table 4.5.1 Matrix Coordinate Transformation Algebra Axioms 29
Table 4.5.2 Matrix Coordinate Transformation Algebra Corollaries 30
Table 6.2.1 Design Criteria for an Omnidirectional (3 DOF) WMR 65
Table 6.4.1 Kinematics-Based WMR Control Algorithm 69

List of Named Equations

(4.7.15) Non-Redundant Wheel Criterion 37
(5.2.2) Composite Robot Equation 43
(5.4.1) Soluble Motion Criterion 50
(5.4.2) Three DOF Motion Criterion 51
(5.4.3) Kinematic Motion Constraints 51
(5.4.4) Number of WMR DOFs 52
(5.5.5) Actuated Inverse Solution 53
(5.6.4) Adequate Actuation Criterion 55
(5.6.5) Actuator Coupling Criterion 57
(5.6.6) Robust Actuation Criterion 57
(5.7.5) Sensed Forward Solution 58
(5.8.4) Adequate Sensing Criterion 60
(5.8.5) Robust Sensing Criterion 60
(5.8.6) Wheel Slip Criterion 62
(6.3.4) Dead Reckoning Update Calculation 67
(6.5.2) Detection of Wheel Slip 71
(A3.2.2) Conventional Non-Steered Wheel Jacobian Matrix 119
(A3.3.2) Conventional Steered Wheel Jacobian Matrix 119
(A3.4.2) Omnidirectional Wheel Jacobian Matrix 120
(A3.5.2) Ball Wheel Jacobian Matrix 121

Abstract

We formulate the *kinematic equations-of-motion* of wheeled mobile robots incorporating *conventional*, *omnidirectional*, and *ball wheels*. While our approach parallels the kinematic modeling of stationary manipulators, we extend the methodology to accommodate such special characteristics of wheeled mobile robots as *multiple closed-link chains*, *higher-pair contact points* between a wheel and a surface, and *unactuated* and *unsensed wheel degrees-of-freedom*. We survey existing wheeled mobile robots to motivate our development. To communicate the kinematic features of wheeled mobile robots, we introduce a diagrammatic convention and nomenclature. We apply the *Sheth-Uicker* convention to assign coordinate axes and develop a *matrix coordinate transformation algebra* to derive the equations-of-motion. A *wheel Jacobian matrix* is formulated to relate the motions of each wheel to the motions of the robot. We combine the individual wheel equations to form the *composite robot equation-of-motion*. We calculate the *sensed forward* and *actuated inverse* solutions and interpret the conditions which guarantee their existence. We interpret the properties of the composite robot equation to characterize the mobility of a wheeled mobile robot according to the *mobility characterization tree*. Similarly, we apply *actuation* and *sensing characterization trees* to delineate the robot motions producible by the wheel actuators and discernable by the wheel sensors, respectively. We apply our kinematic model to *design*, *kinematics-based control*, *dead-reckoning* and *wheel slip detection*. To illustrate the development, we formulate and interpret the kinematic equations-of-motion of six prototype wheeled mobile robots.

1. Introduction

Over the past twenty years, as robotics has become a scientific discipline, research and development have concentrated on *stationary* robotic manipulators[12, 43], primarily because of their industrial applications. Less effort has been directed to mobile robots. Although legged[58] and treaded[37] locomotion has been studied, the overwhelming majority of the mobile robots which have been built and evaluated utilize wheels for locomotion. Wheeled mobile robots (WMRs) are more energy efficient than legged or treaded robots on hard, smooth surfaces[6,7]; and will potentially be the first mobile robots to find widespread application in industry, because of the hard, smooth plant floors in existing industrial environments. Wheeled transport vehicles, which automatically follow paths marked by reflective tape, paint, or buried wire, have already found application[20]. WMRs find application in space and undersea exploration, nuclear and explosives handling, warehousing, security, agricultural machinery, military, education, mobility for the disabled and personal robots.

The wheeled mobile robot literature documents investigations which have concentrated on the application of mobile platforms to perform intelligent tasks [52], rather than on the development of methodologies for analyzing, designing, and controlling the mobility subsystem. Improved mechanical designs and mobility control systems will enable the application of WMRs to tasks were there are no marked paths and to autonomous mobile robot operation. A kinematic methodology is the first step towards achieving these goals.

Even though the methodologies for modeling and controlling stationary manipulators are applicable to WMRs, there are inherent differences which cannot be addressed with these methodologies. Examples include:

1.) WMRs contain multiple *closed-link chains*[53]; whereas stationary manipulators form closed-link chains only when in contact with stationary objects.

2.) The contact between a wheel and a planar surface is a *higher-pair*; whereas stationary manipulators contain only lower-pair joints[3,62,63].

3.) Only some of the degrees-of-freedom (DOFs) of a wheel on a WMR are actuated; whereas all of the DOFs of each joint of a stationary manipulator are actuated.

4.) Only some of the DOFs of a wheel on a WMR have position or velocity sensors; whereas all of the DOFs of each joint of a stationary manipulator have both position and velocity sensors.

Wheeled mobile robot control requires a methodology for modeling, analysis and design which

parallels the technology of stationary manipulators.

Our objective is thus to model the *kinematics* of WMRs. Kinematics is the study of the geometry of motion. In the context of WMRs, we are interested in determining the motion of the robot from the geometry of the constraints imposed by the motion of the wheels. Our kinematic analysis is based upon the assignment of coordinate axes within the robot and its environment, and the application of (4×4) matrices to express transformations between coordinate systems. Each step is defined precisely to lay a solid foundation for the dynamic modeling and feedback control of WMRs. Dynamic models may then be applied to design dynamics-based controllers and simulators. A kinematic methodology may also be applied to design WMRs which satisfy such mobility characteristics as three DOFs (i.e., two translations and a rotation in the plane).

Our kinematic analysis of WMRs parallels the development of kinematics for stationary manipulators. A standard method for modeling the kinematics of stationary robotic manipulators begins by applying the Denavit-Hartenberg convention[18] to assign coordinate axes to each of the robot joints. *Successive* coordinate systems on the robot are related by (4×4) homogeneous transformation **A**-matrices. The **A**-matrices are specified completely by four characteristic parameters (two displacements and two rotations) between consecutive coordinate systems. Each **A**-matrix describes both the shape and size of a robot link, and the translation (for a prismatic joint) or rotation (for a rotational joint) of the associated joint. We assign coordinate axes to the steering links and wheels of a WMR, and apply the Sheth-Uicker convention[61] to define transformation matrices. The Sheth-Uicker convention separates the *constant* shape and size parameters from the *variable* wheel joint parameters, and simplifies the matrix formulation. The Sheth-Uicker convention allows us to model the *higher-pair* relationship between each wheel on a WMR and the floor.

The position and orientation in base coordinates of the end-effector of a stationary manipulator is found by cascading the **A**-matrices *from* the base link *to* the end-effector[56]. Velocity and acceleration relationships are found by differentiating the matrix positions[19]. Velocities of the individual joints are related to the velocities of the end-effector by the manipulator Jacobian matrix[54] in the *forward solution*. The inverse Jacobian matrix is applied in the *inverse solution* to calculate the velocities of the joint variables from the velocities of the end-effector. We develop the wheel Jacobian matrix to relate the velocities of each wheel on a WMR to the robot body velocities. Since WMRs are multiple closed-link chains, the forward and inverse solutions are obtained by solving simultaneously the kinematic equations-of-motion of all of the wheels.

In this paper, we advance the kinematic modeling of WMRs, from the motivation of the kinematic methodology through its development and applications. In Section 2, we survey kinematic configurations (i.e., the relative arrangements and types of wheels) of existing WMRs. These proto-

types illuminate the complexity of the kinematic problem. In Section 3, we describe the three wheels (conventional, omnidirectional and ball wheels) utilized in all existing and foreseeable WMRs.

In Section 4, we develop our approach for modeling the kinematics of WMRs. Coordinate systems are assigned to prescribed positions on the the robot. We introduce transformation matrices to characterize the translations and rotations between coordinate systems. We develop a matrix coordinate transformation algebra to calculate the position, velocity, and acceleration relationships between coordinate systems. We apply the axioms and corollaries of this algebra to transform positions, velocities, and accelerations which are specified in one coordinate frame to another coordinate frame, and develop the wheel Jacobian matrix to relate the motions of a wheel to the motions of the robot. In Section 4.9, we outline our kinematic methodology for WMRs.

In Section 5, we form the composite robot equation-of-motion by adjoining the equations-of-motion of all of the wheels. We then solve the composite robot equation. Specifically, we calculate the actuated wheel velocities in terms of the robot velocities (the actuated inverse solution), and the robot velocities in terms of the sensed wheel velocities (the sensed forward solution). We characterize a WMR by interpreting the properties of the composite robot equation. We present a mobility characterization tree which specifies tests to be conducted on the composite robot equation and displays the mobility characteristics of the WMR. We also calculate the number of degrees-of-freedom of a WMR. The ability of the actuators to produce robot motion is determined by the actuation characterization tree. Similarly, the sensing structure is specified by the sensing characterization tree.

In Section 6, we apply our kinematic modeling methodology to the design, dead-reckoning, kinematics-based control, and wheel slip detection for WMRs. Just as we apply the mobility characterization tree to delineate the mobility of a WMR, we may design a WMR to satisfy desired mobility characteristics by proper choice of wheel type and placement. We calculate the current robot position (i.e., dead-reckoning) by summing the robot velocities in real-time. We introduce a kinematics-based WMR feedback control system in which the actuated inverse and sensed forward solutions are integral components. Our development of the sensing characterization tree illuminates a method of detecting the onset of wheel slip. We present our slip detection method and describe the proper positioning of the wheel sensors for implementation. We are continuing our study of WMRs by applying our kinematic model to formulate dynamic models of WMRs.

In Section 7, we apply our kinematic modeling methodology to six prototype WMRs. We present the kinematic description, coordinate system assignments, transformation matrices, wheel Jacobian matrices, mobility characteristics and the sensed forward and actuated inverse solutions for each. From our experience with these prototype examples, we draw practical conclusions about

the applicability of three DOFs vs two DOFs and the utilization of redundant steered-conventional wheels.

We summarize (in Section 8) our kinematic methodology and its implications, and outline (in Section 9) our plans for continued research in dynamic modeling and feedback control. In Appendix 2, we compile our symbols.

2. Survey of Kinematic Configurations

In this section, we survey the kinematic configurations of existing WMRs. We are interested in determining the types of wheels utilized and the relative placement of the wheels on WMRs. Documentation of WMRs is scattered throughout the robotics, artificial intelligence, control engineering, scientific, industrial, popular and hobbiest literature[8,16,23,38,60]. We examine documented WMRs to understand the requirements of a kinematic methodology for this class of mobile robots. We then generalize the kinematic model of these exemplary robots and define (in Section 4) a WMR which specifies the range of mobile robots to which our methodology applies. Our survey also provides a set of prototype WMRs for evaluating our kinematic methodology.

In Appendix 1, we introduce a nomenclature and a pictorial representation for describing the kinematic structure of WMRs. The diagramming conventions provide a convenient tool for describing and comparing kinematic structures of WMRs. We apply these rules to develop symbolic diagrams and kinematic names for the WMRs presented in this survey and refer to these representations as we describe each WMR.

The most common kinematic arrangement of mobile robots documented in the literature has two diametrically opposed wheels (i.e., two parallel conventional wheels, one on each side of the robot). These robots also possess one or two castors for stability. Among the most widely known examples are: Shakey[52], Newt[32] (in Figure 2.1), Jason[64], Hilare[24], Yamabiko[40,35], ROBART II[22], and RB5X[44]. By mounting the two driven wheels at an acute angle to the floor in their Topo[27] robot (in Figure 2.1), the Androbot Company stabilized the robot without the use of castors.

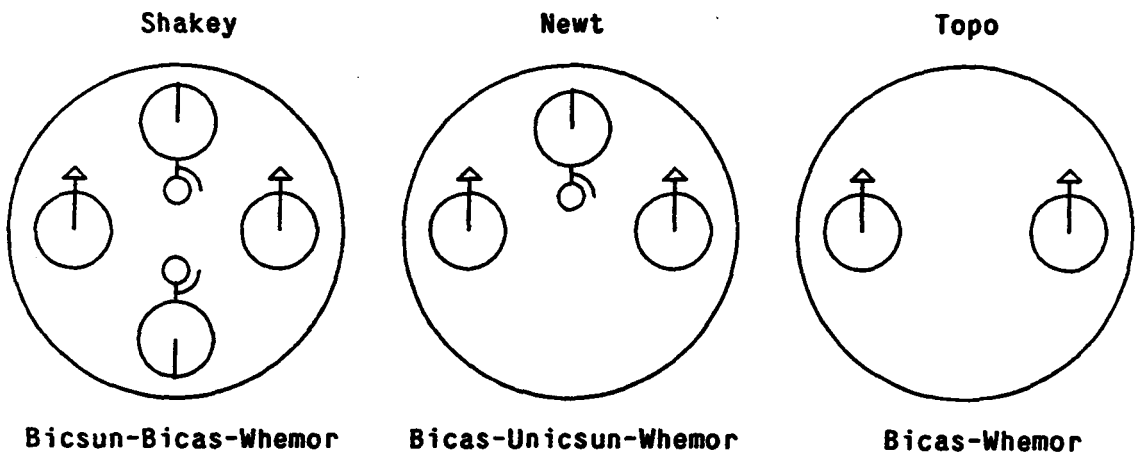


Figure 2.1

Kinematic Representations of Shakey, Newt, and Topo

Mobile robots which possess multiple non-steered, driven wheels whose axes are non-colinear must rely on wheel slip if the robot is to navigate turns. Such is the case with the RDS Prowler[59] and the Terregator[66] (in Figure 2.2), both of which use six parallel, non-steered, conventional wheels, three on each side. Similarly, Gemini[28] (in Figure 2.2) utilizes two synchronously driven wheels on each side.

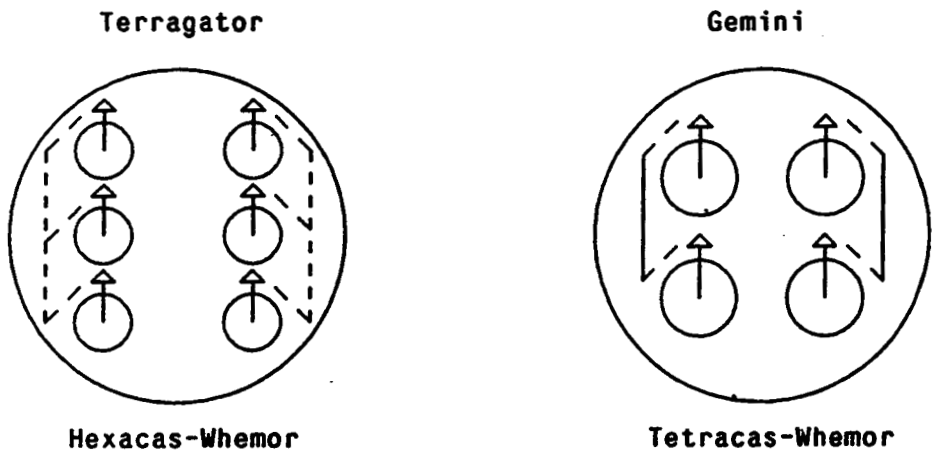


Figure 2.2

Kinematic Representations of Terregator and Gemini

The mechanically more complex, steered and driven conventional wheel is utilized on Neptune[57] (in Figure 2.3), Hero-1[26] and Avatar[4]. These three robots have a tricycle wheel arrangement; the front wheel is steered and driven, while the two rear wheels are at a fixed parallel orientation and are undriven.

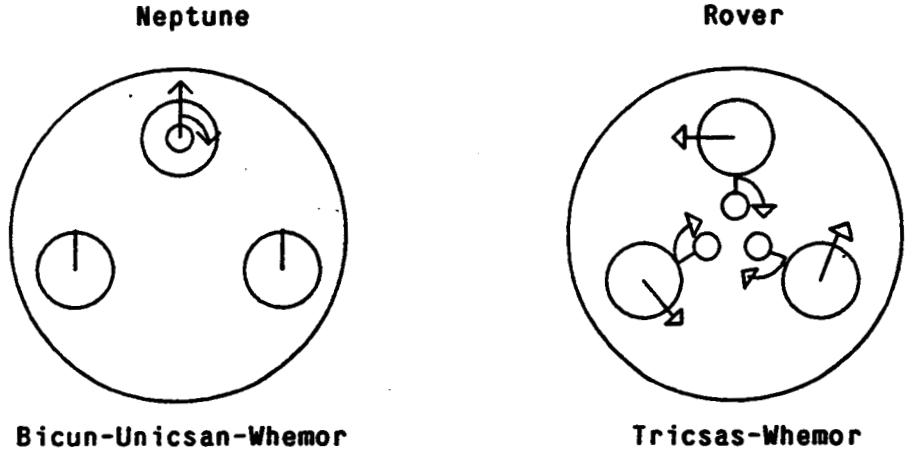


Figure 2.3

Kinematic Representations of Neptune and Pluto

The CMU Rover[48] (in Figure 2.3), also known as Pluto, has three steered and driven wheels. The Stanford Cart[46] (in Figure 2.4) has two steered, undriven wheels in the front and two fixed, driven wheels in the rear. The two front wheels are coupled by an Ackerman steering linkage.[1] Both the front and back wheels of the JPL Rover[41] (in Figure 2.4) are coupled by Ackerman steering linkages, and all four wheels are driven independently. Kludge[30] (in Figure 2.4) is an example of a robot with complex functional dependencies between the wheels. This robot has three conventional wheels that are both steered and driven. A chain and gear arrangement is used to equalize all drive velocities and steering angles (Synchro-Drive). To complicate further the arrangement, each wheel is mounted on an actuated link which can be pivoted towards or away from the center of the robot for stability. Kludge's successor K2A[30] embodies the synchro-drive mechanism using concentric shafts instead of chains and does not have any actuated links. The Denning Sentry robot[70] also utilizes a three-wheel synchronous drive and steer system.

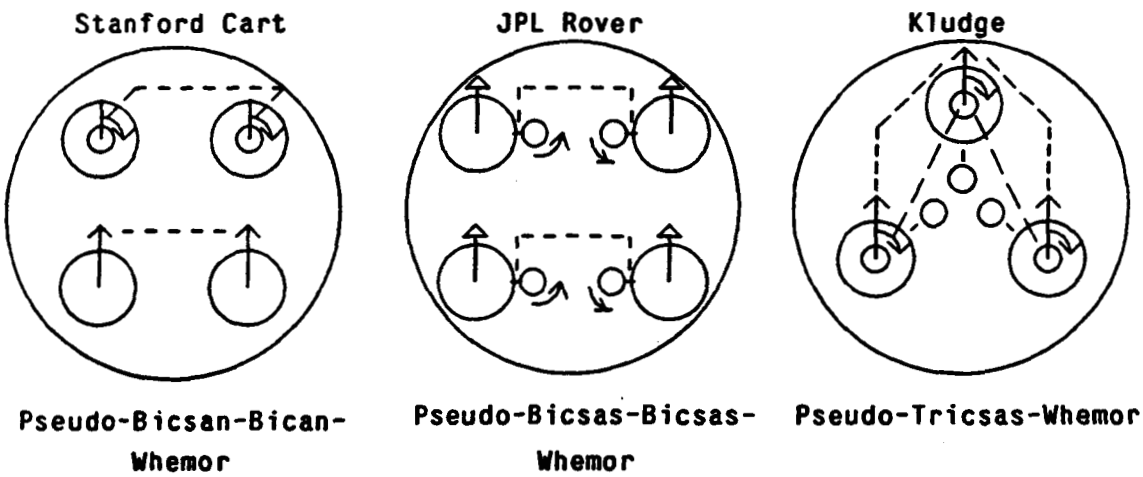


Figure 2.4

Kinematic Representations of the Stanford Cart, the JPL Rover, and Kludge

The hybrid spider drive[29] (in Figure 2.5) utilizes four conventional wheels, two on either side of the robot, each of which is mounted at the end of a three DOF leg linkage. The hybrid locomotion vehicle[34] (in Figure 2.5) utilizes six steered and driven conventional wheels, each at the end of an actuated vertical leg.

[1] An Ackerman steering linkage[45] approximatly ensures the correct wheel angles to avoid wheel slip.

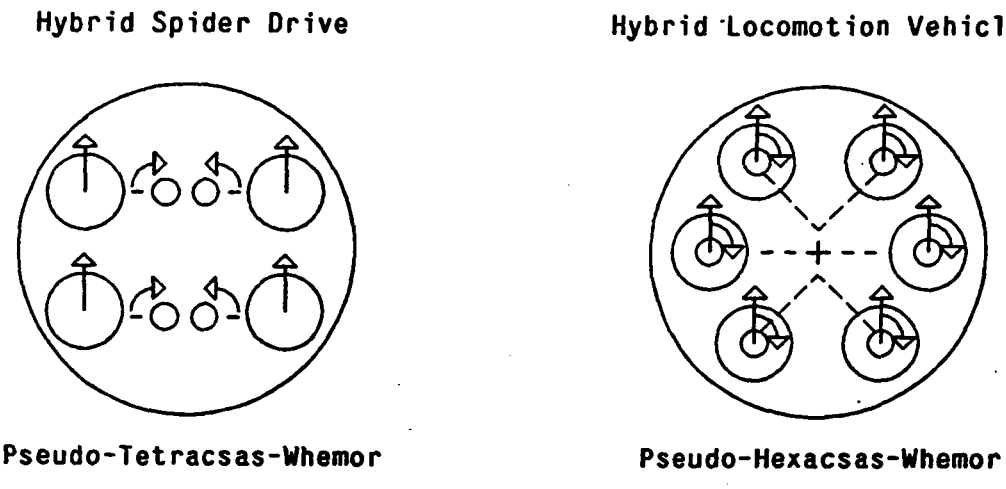


Figure 2.5

Kinematic Representations of the Hybrid Spider Drive and the Hybrid Locomotion Vehicle

Equally obscure is the triangle wheel step climber[67], which possesses four sets of three wheels mounted at the vertices of equilateral triangles. When a wheel encounters a step, the triangle pivots about its center and the robot reaches the top of the step by rolling on a different set of wheels.

The recent application of omnidirectional wheels (in Section 3) has led to novel mobile kinematic configurations. Omnidirectional wheels have been used for powered wheelchairs (e.g., Omni drive[29] and Wheelon[2]) and ambulatory drive platforms [69]. The later orients the omnidirectional wheels at an acute angle to the floor for stability. Uranus[49] (in Figure 2.6) has a rectangular wheel base with four omnidirectional wheels having rollers at 45° angles. The Unimation robot[14] (in Figure 2.6) and Fetal1[38] have triangular wheel bases and three omnidirectional wheels with 90° rollers.

Omnidirectional treads[10, 11] operate as omnidirectional wheels with the rollers mounted upon tank-like treads. A ball wheel (in Section 3) is the most maneuverable wheel allowing three DOF motion[47, 13, 39]. The first design of Jason[64] incorporated three ball wheel castors which were later replaced by a single conventional castor. We are unaware of any other documented applications of ball wheels on WMRs.

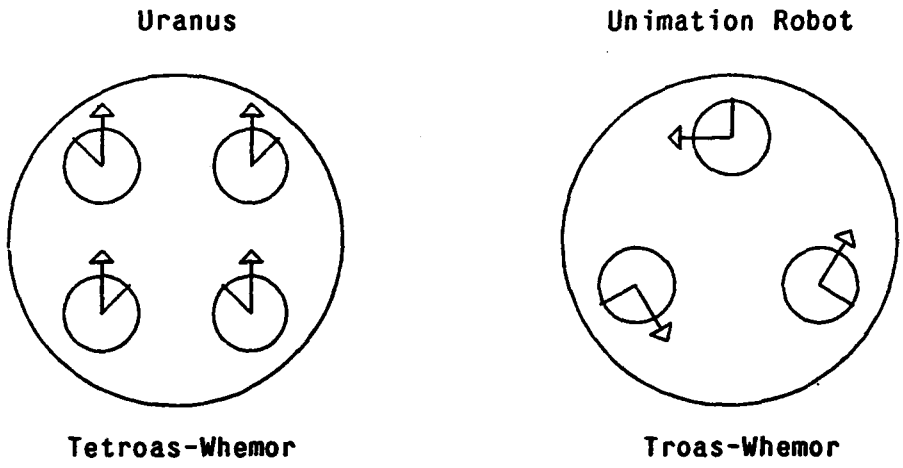


Figure 2.6

Kinematic Representations of Uranus and the Unimation Robot

Because of the variability in the numbers and types of wheels and actuating mechanisms, formulating a kinematics methodology for WMRs requires analytically complex robot models. Since the preponderance of existing and foreseeable WMRs have simpler kinematic configurations then those on the periphery of WMRs (e.g., the hybrid spider drive), applying a general-purpose and universal approach to model the kinematics of practical WMRs would be unduly cumbersome. To reduce substantially the complexity of the kinematic model and associated calculations, we limit our analysis to WMRs with zero or one steering links per wheel. The robots which do not satisfy this constraint (e.g., hybrid spider drive, hybrid locomotion vehicle, and Kludge) can be modeled by extending our analytical approach on a case-by-case basis.

From this survey, we specify the requirements of a kinematic model of WMRs. A WMR model must allow any number of wheels. The wheels can be mounted at any position and orientation with respect to the robot body provided that each touches the surface of travel. This constraint includes the ability to mount wheels at acute angles to the surface. The WMR can incorporate any combination of conventional, omnidirectional or ball wheels. Even though each wheel can be mounted at the end of an articulated linkage, we will deal with zero or one steering link per wheel. Finally, there may be coupling between wheels (e.g., two wheels may steer together as on the Stanford Cart). With these observations, we define a WMR in Section 4 to develop a methodology for kinematic modeling. In Section 3, we detail the operation of the three basic wheel types.

3. Wheel Types

Three wheel types are used in WMR designs: conventional, omnidirectional, and ball wheels. In addition, conventional wheels are often mounted on a steering link to provide an additional DOF. Schematic views of the three wheels are shown in Figure 3.1. The DOFs of each wheel are indicated by the arrows in Figure 3.2. The kinematic relationships between the angular velocity of the wheel and its linear velocity along the surface of travel are also compiled in the figure.

The *conventional* wheel having two DOFs is the simplest to construct. It allows travel along a surface in the direction of the wheel orientation, and rotation about the point-of-contact between the wheel and the floor. We note that the rotational DOF is slippage, since the point-of-contact is not stationary with respect to the floor surface[1]. Even though we define the rotational slip as a DOF, we do not consider slip transverse to the wheel orientation a DOF, because the magnitude of force required for the transverse motion is much larger than that for rotational slip. The conventional wheel is by far the most widely used wheel; automobiles, roller skates and bicycles utilize this wheel.

The *omnidirectional* wheel has three DOFs. One DOF is in the direction of the wheel orientation. The second DOF is provided by motion of rollers mounted around the periphery of the main wheel. In principle, the roller axles can be mounted at any nonzero angle η with respect to the wheel orientation. The omnidirectional wheels in Figures 3.1 and 3.3 have roller axle angles of 90° [9,11,25], and 45°[36], respectively. The third DOF is rotational slip about the point-of-contact. It is possible, but not common, to actuate the rollers of an omnidirectional wheel[29] with a complex driving arrangement. When sketching WMRs having omnidirectional wheels, the rollers on the underside of the wheel (i.e., those touching the surface of travel) are drawn and not the rollers which are actually visable from a top view, to facilitate kinematic analysis.

The most maneuverable wheel is a *ball* which possesses three DOFs without slip. Schemes have been devised for actuating and sensing ball wheels[47], but we are unaware of any existing implementations. An omnidirectional wheel which is steered about its point-of-contact is kinematically equivalent to a ball wheel, and may be a practical design alternative.

[1] Two bodies are in rolling contact if the points-of-contact of the two bodies are stationary relative to each other[63].

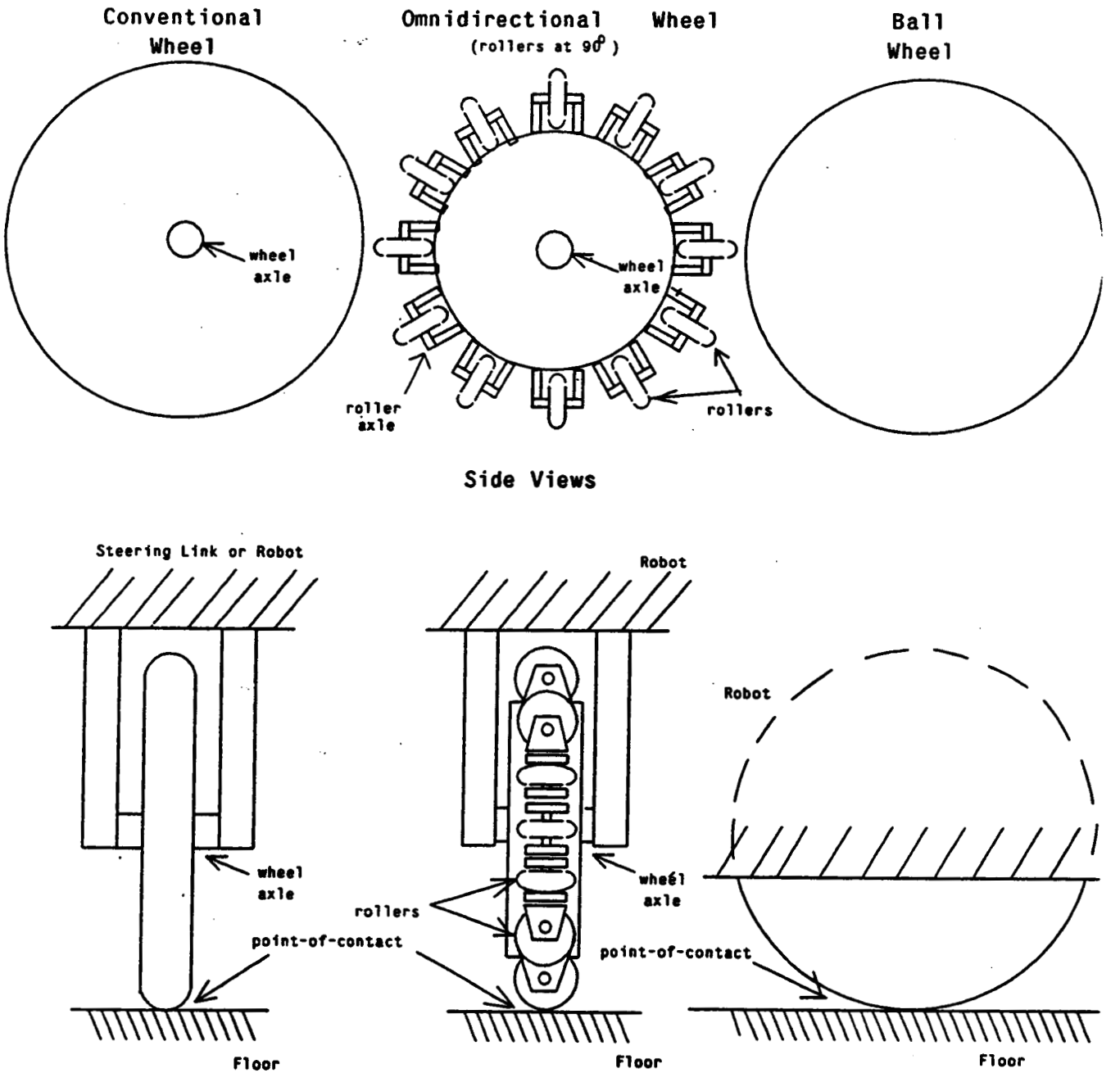
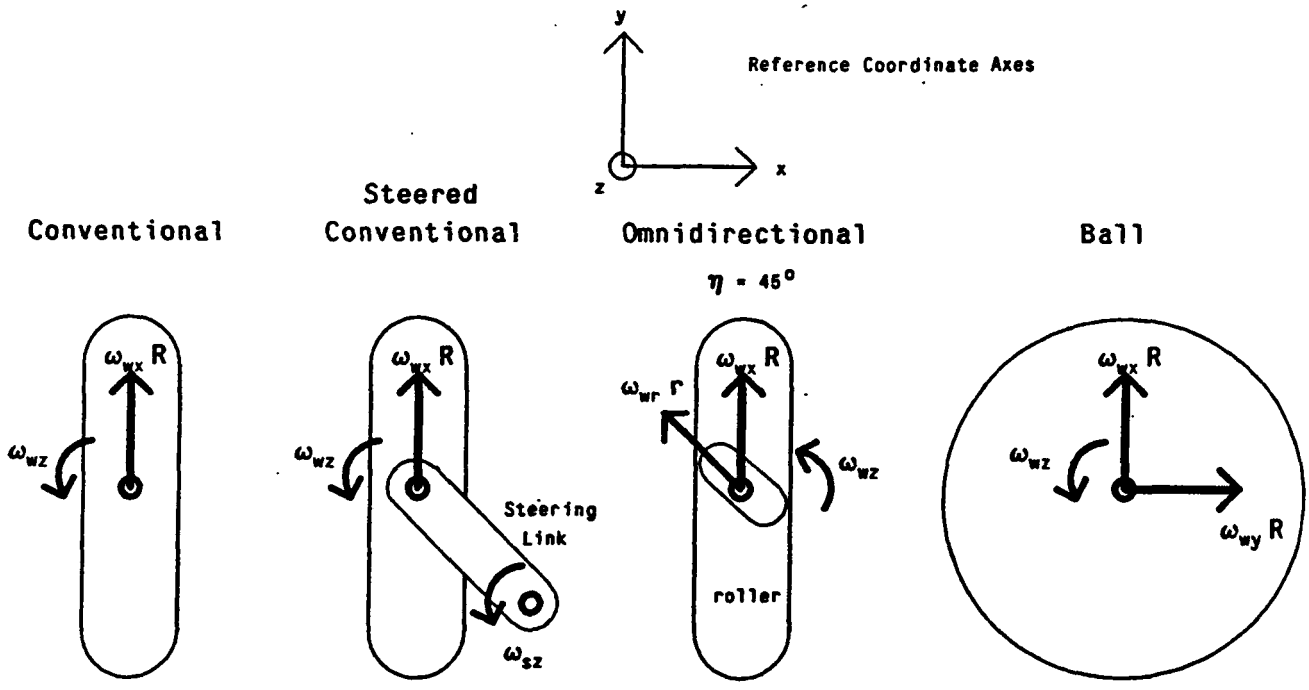


Figure 3.1

Conventional, Omnidirectional, and Ball Wheels



Wheel DOFs

Kinematic Relationships		
Conventional and Steered Conventional	Omnidirectional	Ball
$V_y = \omega_{wx} R$	$V_y = \omega_{wx} R - \omega_{wr} r \cos(\eta)$	$V_y = \omega_{wx} R$
$V_x = 0$	$V_x = \omega_{wr} r \sin(\eta)$	$V_x = \omega_{wy} R$
$\omega_z = \omega_{wz}$	$\omega_z = \omega_{wz}$	$\omega_z = \omega_{wz}$

Legend

V_x and V_y	=	x and y components of the linear velocity of the wheel at the point-of-contact
ω_z	=	z component of the angular velocity of the wheel at the point-of-contact
ω_{wr}	=	angular velocities of the roller about their axles
ω_{wx}, ω_{wy} and ω_{wz}	=	x, y, and z angular velocities of the wheel about its center
ω_{sz}	=	angular velocity of the steering link about the steering axis
η	=	angle of the roller axles with respect to the wheel orientation
R and r	=	radii of a wheel and a roller

Figure 3.2

Wheel Equations-of-Motion

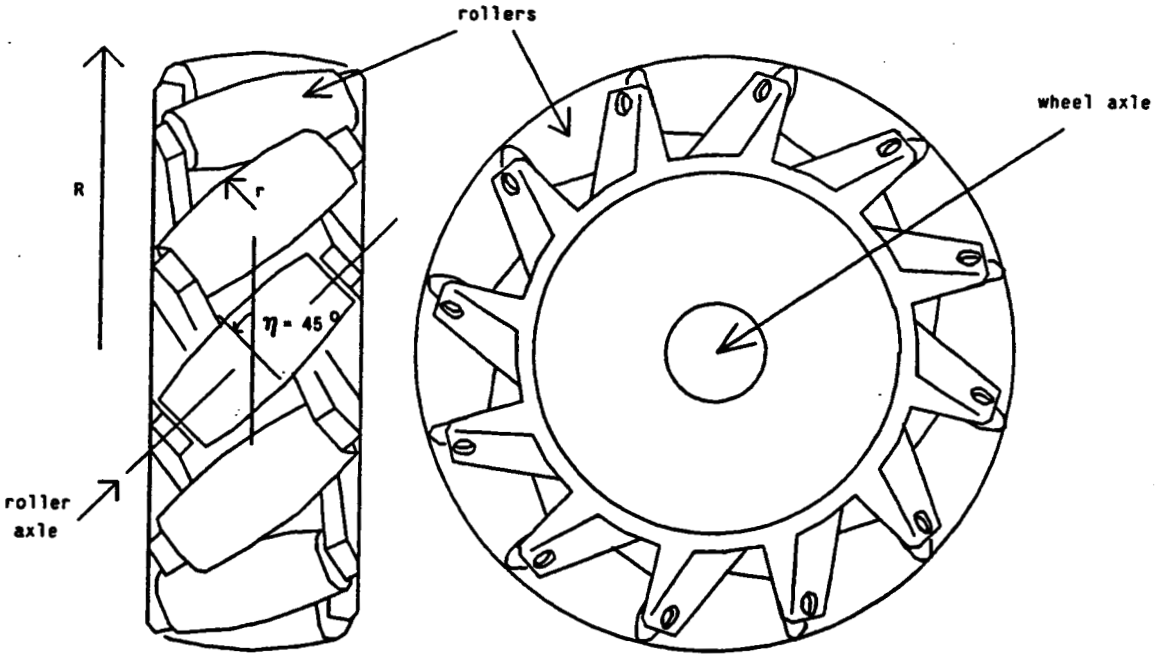


Figure 3.3

Omnidirectional Wheel (Rollers at 45°)

4. Kinematic Modeling

4.1 Introduction

In this section, we apply and extend standard robotic nomenclature and methodology[54] to model the *kinematics* of WMRs. The novel aspects are our treatment of the higher-pair joint between each wheel and the floor, and the development of a transformation matrix algebra.

We begin (in Section 4.2) by defining a WMR and enumerating our modeling assumptions to constrain the class of mobile robots to which our modeling methodology applies. To include all existing and foreseeable WMRs, we would have to generalize our methodology and thereby complicate the modeling of the overwhelming majority of WMRs. In Section 4.3, we assign coordinate systems to the robot body, wheels and steering links to facilitate kinematic modeling. It is essential to define *instantaneously coincident coordinate systems* to model the higher-pair joints at the point of contact between each wheel and the floor. In Section 4.4, we assign homogeneous (4×4) transformation matrices to relate coordinate systems. We present (in Section 4.5) a matrix coordinate transformation algebra to formulate the equations-of-motion of a WMR. All kinematics are derived by straightforward application of the axioms and corollaries of the transformation algebra. Position kinematics are treated in Section 4.6. We demonstrate that transforming the coordinates of a point between coordinate systems is equivalent to finding a path in a transformation graph. Then, in Section 4.7, we formulate the velocity kinematics. The relationships between the wheel velocities and the robot velocities are linear. We thus develop a wheel Jacobian matrix to calculate the vector of robot velocities from the vector of wheel velocities. Finally, in Section 4.8, we apply our matrix coordinate transformation algebra to acceleration kinematics.

To summarize the development, we enumerate in Section 4.9 our kinematic modeling procedure. In Section 5, we combine the equations-of-motion of all of the wheels to form the composite robot equation. We then proceed to solve the composite robot equation and interpret the solutions.

4.2 Definitions And Assumptions

The Robot Institute of America defines a robot as "*A programmable, multifunction manipulator designed to move material, parts, tools, or specialized devices through variable programmed motions for the performance of a variety of tasks*"[29]. Our survey of kinematic configurations in Section 2 anticipates the definition of a WMR. Kinematic models of WMRs are inherently different from those of stationary robotic manipulators and legged or treaded mobile robots. We thus introduce an operational definition of a WMR to specify the range of robots to which the kinematic methodology presented in this paper applies.

Wheeled Mobile Robot - A robot capable of locomotion on a surface solely through the actuation of wheel assemblies mounted on the robot and in contact with the surface. A wheel assembly is a device which provides or allows relative motion between its mount and a surface on which it is intended to have a *single* point of *rolling* contact.

Each wheel (conventional, omnidirectional or ball wheel) and all links between the robot body and the wheel constitute a wheel assembly. With the exception of the omnidirectional treaded vehicle, the hybrid spider drive (when walking), the hybrid locomotion vehicle (when climbing) and the triangle wheel step climber (when climbing steps), the mobile robots reviewed in Section 2 satisfy our definition of a WMR.

We introduce the following practical assumptions to make the modeling problem tractable.

Design Assumptions

1.) The WMR does not contain flexible parts.

2.) There is zero or one steering link per wheel.

3.) All steering axes are perpendicular to the surface.

Operational Assumptions

4.) The WMR moves on a planar surface.

5.) The translational friction at the point of contact between a wheel and the surface is large enough so that no translational slip may occur.

6.) The rotational friction at the point of contact between a wheel and the surface is small enough so that rotational slip may occur.

We discuss our assumptions in turn. Assumption 1 states that the dynamics of such WMR components as flexible suspension mechanisms and tires are negligible. We make this assumption to apply rigid body mechanics to kinematic modeling. We recognize that flexible structures may play a significant role in the kinematic analysis of WMRs. A dynamic analysis to determine the changes in kinematic structure due to forces/torques acting on flexible components is required to model these components. Such an analysis is appropriate for WMRs even though it has not conventionally been addressed for stationary open-link manipulators because WMRs are inherently *closed-link* mechanisms. Flexible components, that allow compliance in the multiple closed-link chains of a WMR, lead to a consistent kinematic model. Without compliant structures, there

cannot be a consistent kinematic model for WMRs in the presence of surface irregularities, inexact component dimensions and inexact control actuation[50]. A simultaneous kinematic and dynamic analysis of WMRs is thus a natural continuation of our research.

We introduce Assumptions 2 and 3 to reduce the range of WMRs that our methodology must address, by limiting the complexity of our kinematic model. WMRs which have more than one link per wheel can be analyzed by our methodology if only one steering link is allowed to move. We require that all non-steering links must be stationary, as if they are extensions of the robot body or wheel mounts. By constraining the steering links to be perpendicular to the surface of travel in Assumption 3, we reduce all motions to a plane. We thus constrain all component motions to a rotation about the normal to the surface, and two translations in a plane parallel to the surface.

Assumption 4 neglects irregularities in the actual surface on which a WMR travels. Even though this assumption restricts the range of practical applications, environments which do not satisfy this assumption (e.g., rough, bumpy or rocky surfaces) do not lend themselves to energy efficient wheeled vehicle travel[7].

Assumption 5 ensures the applicability of the theoretical kinematic properties of a wheel in rolling contact[5, 62] for the two translational degrees-of-freedom. This assumption is realistic for dry surfaces as demonstrated by the success of braking mechanisms on automobiles. Automobiles also illustrate the practicality of Assumption 6. The wheels must rotate (i.e., slip) about their points-of-contact to navigate a turn. Since WMRs also rely on rotational wheel slip, we include Assumption 6.

4.3 Coordinate System Assignments

4.3.1 Sheth-Uicker Convention

Coordinate system assignment is the first step in the kinematic modeling of a stationary manipulator[54]. Lower-pair mechanisms[1] (such as revolute and prismatic joints) function with two surfaces in relative motion. In contrast, the wheels of a WMR are higher-pairs which function ideally by point contact. Because the A-Matrices which model manipulators depend upon the relative position and orientation of two successive joints, the Denavit-Hartenberg convention[18] leads to ambiguous assignments of coordinate transformation matrices in multiple closed-link chains[61] which are inherent in WMRs. The ambiguity arises in deciding the joint ordering when there are more than two joints on a single link.

[1] Lower-pair mechanisms are pairs of components whose relative motions are constrained by a common surface contact; whereas higher-pairs are constrained by point or line contact[5].

We apply the Sheth-Uicker convention[61] to assign coordinate systems and model each wheel as a planar pair at the point of contact. This convention allows the modeling of the higher-pair wheel motion and eliminates ambiguities in coordinate transformation matrices. The planar pair allows three DOFs as shown in Figure 4.3.1 : X and Y translation, and rotation about the point-of-contact. The Sheth-Uicker convention is ideal for modeling ball wheels; the angular velocities of the wheel are converted directly into translational velocities along the surface. The planar pair motions must be constrained to include wheels which do not allow three DOFs. For example, the coordinate system assigned at the point-of-contact of a conventional wheel is aligned with the y-axis parallel to the wheel. The wheel model is completed by constraining the x-component of the wheel velocity to zero to satisfy Assumption 5 (in Section 4.2) and avoid translational slip.

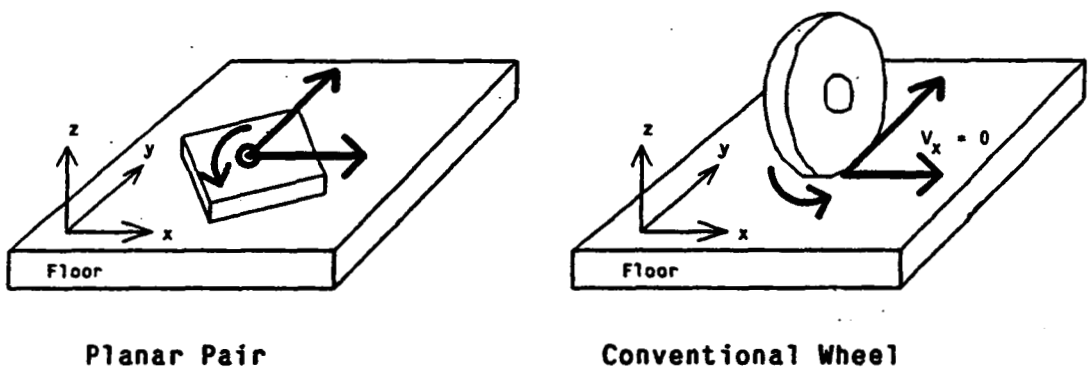


Figure 4.3.1

Planar Pair Model of a Wheel

4.3.2 WMR Coordinate Systems

We assign coordinate systems at both ends of each *link* of the WMR. The links of the closed-link chain of a WMR are the floor, the robot body and the steering links. The *joints* are: a revolute pair at each steering axis, a planar pair to model each wheel, and a planar pair to model the robot body. When the joint variables are zero, the coordinate systems of the two links which share the joint coincide. We summarize our approach to the modeling of a WMR having N wheels with the coordinate system assignments defined in Table 4.3.1 . Placement of the coordinate systems is illustrated in Figure 4.3.2 for the pictorial view of a WMR. For a WMR with N wheels, we assign $3N + 1$ coordinate systems to the robot and one stationary reference frame. There are also $N + 1$ instantaneously coincident coordinate systems (described in Section 4.3.3) which need not be assigned explicitly.

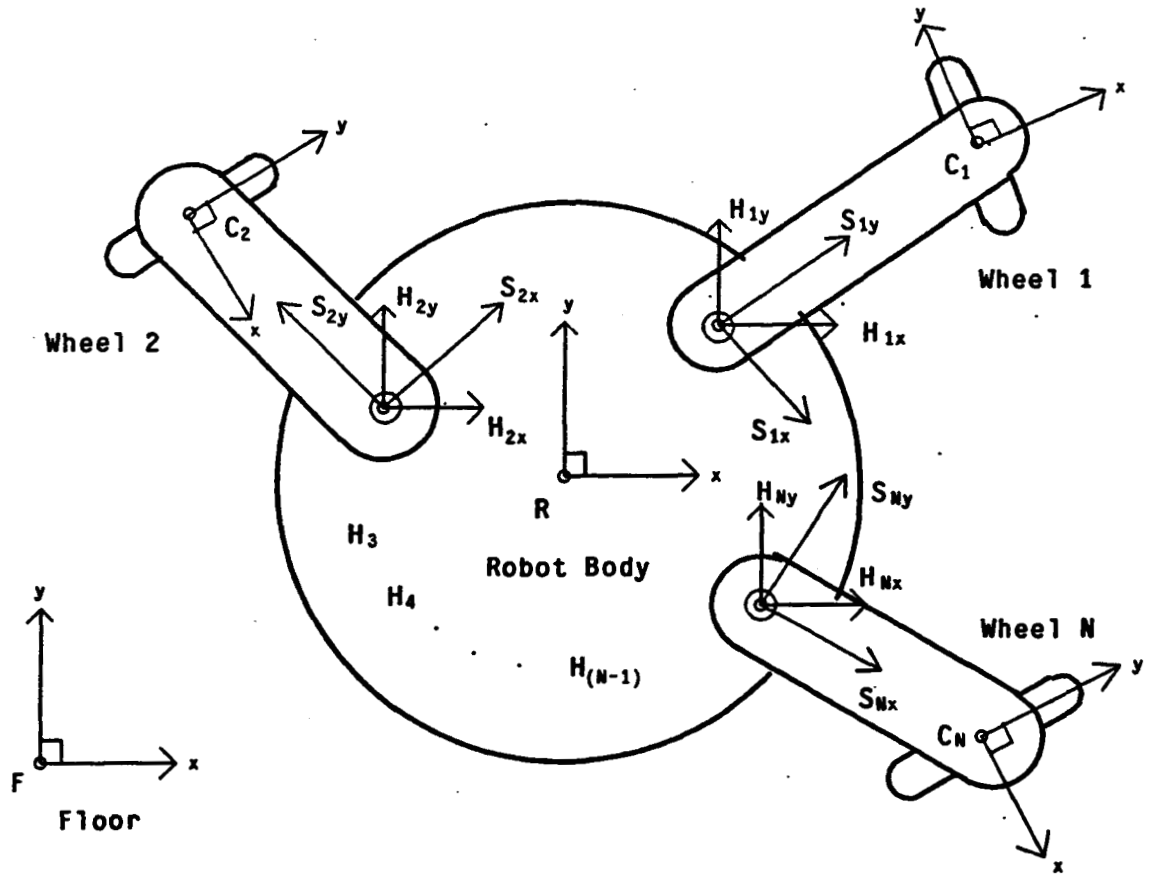


Figure 4.3.2

Placement of Coordinate Systems on a WMR

The *floor* coordinate system F is stationary relative to the surface of travel and serves as the reference coordinate frame for robot motions. The *robot* coordinate system R is assigned to the robot body so that the position of the WMR is the displacement from the floor coordinate system to the robot coordinate system. The *hip* coordinate system H_i is assigned at the point on the robot body which intersects the steering axis of wheel i. The *steering* coordinate system S_i is assigned at the same point along the steering axis of wheel i, but is fixed relative to the steering link. We assign a *contact point* coordinate system C_i at the point-of-contact between each wheel and the floor.

Coordinate system assignments are not unique. There is freedom to assign the coordinate systems at positions and orientations which lead to convenient structures of the kinematic model. For example, all of the hip coordinate systems may be assigned parallel to the robot coordinate system resulting in sparse robot-hip transformation matrices and thus simplifying the model. Alternatively, the x-axes of the hip coordinate systems can be aligned with the zero position of the

steering joint position encoders so that the hip-steering transformation is expressed in terms of the actual steering angle.

4.3.3 Instantaneously Coincident Coordinate Systems

To introduce the concept of instantaneously coincident coordinate systems, we consider the one-dimensional example of a ball rolling in a straight line on a flat surface. The position of the ball is depicted by the point r in Figure 4.3.3.

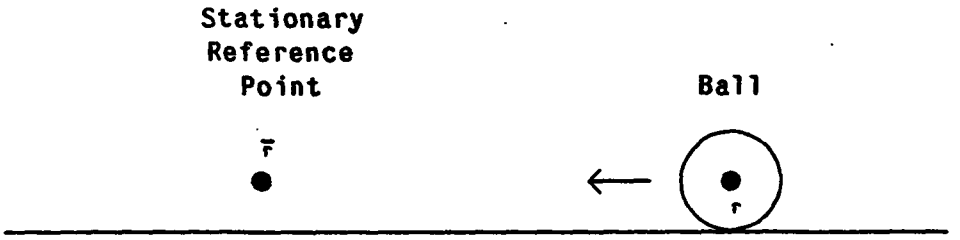


Figure 4.3.3

Ball in Motion Before Instantaneous Coincidence

The ball is moving right to left with velocity v_r and acceleration a_r. The stationary reference point $\bar{r}$ lies in the path of the moving ball. At the *instant* the ball (point r) and the reference (point $\bar{r}$) *coincide* in Figure 4.3.4, we observe that: (1) The position of the ball relative to the reference point ${}^{\bar{r}}p_r$ is zero: and (2) The velocity ${}^{\bar{r}}v_r$ and acceleration ${}^{\bar{r}}a_r$ of the ball relative to the reference point are non-zero. We call the point $\bar{r}$ an *instantaneously coincident* reference point for the moving ball at the instant shown in Figure 4.3.4.

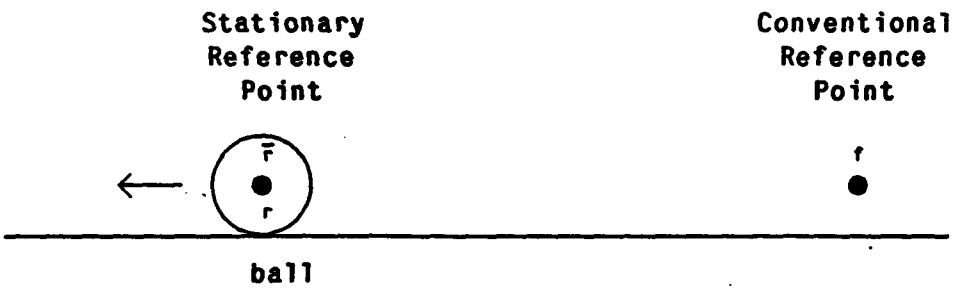


Figure 4.3.4

Ball in Motion at Instantaneous Coincidence

We continuously assign an instantaneously coincident reference point $\bar{r}$ during the motion of the ball to generalize our observations for all time t. The position of the ball relative to its instantaneously coincident reference point is zero (i.e., ${}^{\bar{r}}p_r(t) = 0$), and the velocity and acceleration

of the ball relative to its instantaneously coincident reference point are non-zero (i.e., ${}^{\bar{r}}v_r(t) \neq 0$ and ${}^{\bar{r}}a_r(t) \neq 0$). In the framework of instantaneously coincident reference points, we emphasize that we *cannot* differentiate the position (velocity) equation-of-motion to obtain the velocity (acccleration) equation-of-motion.

The stationary reference point f in Figure 4.3.4 is a conventional reference point whose position is fixed. Since both reference points f and $\bar{r}$ are stationary, the velocity (acceleration) of the ball relative to the point f is equal to the velocity (acceleration) of the ball relative to the point $\bar{r}$ in this one-dimensional example. Consequently, it is not advantagous to introduce instantaneously coincident references in the one-dimensional example. The practical need for instantaneously coincident coordinate systems arises in the multi-dimensional example as depicted in Figure 4.3.5.

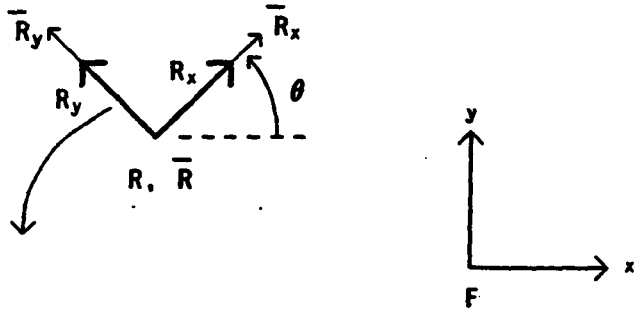


Figure 4.3.5

Coordinate System R in Motion

The coordinate system R is moving in three-dimensions: X, Y, and θ. The coordinate systems $\bar{R}$ and F are stationary; $\bar{R}$ is an instantaneously coincident coordinate system and F is a conventional reference coordinate system. We make the analogous observations. The position of the moving coordinate system relative to its instantaneously coincident coordinate system is zero (i.e., ${}^{\bar{R}}\mathbf{p}_R = 0$). The position of the moving coordinate system relative the conventional reference coordinate system is non-zero (i.e., ${}^{F}\mathbf{p}_R \neq 0$). The non-zero velocity ${}^{\bar{R}}\mathbf{v}_R$ (acceleration ${}^{\bar{R}}\mathbf{a}_R$) of the moving coordinate system relative to the instantaneously coincident coordinate system is not equal to the velocity ${}^{F}\mathbf{v}_R$ (acceleration ${}^{F}\mathbf{a}_R$) of the moving coordinate system relative to the conventional reference coordinate system. The velocity (acceleration) of the moving coordinate system relative to the conventional reference coordinate system F depends upon the position and orientation of the moving coordinate system *relative* to the reference coordinate system. *The motivation for assigning instantaneously coincident coordinate systems is that the velocities (accelerations) of*

a multi-dimensional moving coordinate system can be computed or specified independently of the position of the moving coordinate system. The instantaneously coincident coordinate system is a conceptual tool which enables us to calculate the velocities and accelerations of a moving coordinate system relative to its instantaneous current position and orientation.

Table 4.3.1: Coordinate System Assignments

F *Floor* : Stationary reference coordinate system with the z-axis orthogonal to the surface of travel.

R *Robot* : Coordinate system which moves with the WMR body, with the z-axis orthogonal to the surface of travel.

H_i *Hip* (for $i = 1, ..., N$) : Coordinate system which moves with the WMR body, with the z-axis coincident with the axis of steering joint i if there is one; coincident with the contact point coordinate system C_i if there is no steering joint.

S_i *Steering* (for $i = 1, ..., N$) : Coordinate system which moves with steering link i, with the z-axis coincident with the z-axis of H_i, and the origin coincident with the origin of H_i.

C_i *Contact Point* (for $i = 1, ..., N$) : Coordinate system which moves with steering link i, with the origin at the point-of-contact between the wheel and the surface; the y-axis is parallel to the wheel (if the wheel has a preferred orientation; if not, the y-axis is assigned arbitrarily) and the x-y plane is tangent to the surface.

$\overline{R}$ *Instantaneously Coincident Robot* : Coordinate system coincident with the R coordinate system and stationary relative to the F coordinate system.

$\overline{C_i}$ *Instantaneously Coincident Contact Point* (for $i = 1, ..., N$) : Coordinate system coincident with the C_i coordinate system and stationary relative to the F coordinate system.

For stationary serial link manipulators, all joints are one-dimensional lower-pairs: prismatic joints allow Z motion and revolute joints allow θ motion. In contrast, WMRs have three-dimensional higher-pair wheel-to-floor and robot-to-floor joints allowing simultaneous X, Y and θ motions. We assign an *instantaneously coincident robot* coordinate system $\overline{R}$ at the same position and orientation in space as the robot coordinate system R. In Table 4.3.1, we define the instantaneously coincident robot coordinate system to be stationary relative to the floor coordinate system F. By design, the position and orientation of the robot coordinate system R and the instantaneously coincident robot coordinate system $\overline{R}$ are identical, but (in general) the relative velocities and accelerations between the two coordinate systems are non-zero. When the robot coordinate system moves relative to the floor coordinate system, we assign a different instantaneously coincident coordinate system for each time instant. The instantaneously coincident robot coordinate system facilitates the specification of robot velocities (accelerations) independently of the robot position. Similarly, the *instantaneously coincident contact point* coordinate system $\overline{C}_i$ (in Table 4.3.1) coincides with the contact point coordinate system C_i and is stationary relative to the floor coordinate system. Since the position of the wheel contact point is not sensed, we require the instantaneously coincident contact point coordinate system to specify wheel velocities and accelerations.

4.4 Transformation Matrices

Homogeneous (4×4) transformation matrices are defined to express the relative positions and orientations of coordinate systems[54]. The homogeneous transformation matrix ${}^{A}\mathbf{\Pi}_{B}$ transforms the coordinates of the point ${}^{B}\mathbf{r}$ in coordinate frame B to its corresponding coordinates ${}^{A}\mathbf{r}$ in the coordinate frame A:

$$ {}^{A}\mathbf{r} = {}^{A}\mathbf{\Pi}_{B}\ {}^{B}\mathbf{r}\ . \tag{4.4.1} $$

We adopt the following notation. Scalar quantities are denoted by lower case letters (e.g., w). Vectors are denoted by lower case boldface letters (e.g., $\mathbf{r}$). Matrices are denoted by upper case boldface letters (e.g., $\mathbf{\Pi}$). Pre-superscripts denote reference coordinate systems. For example, ${}^{A}\mathbf{r}$ is the vector $\mathbf{r}$ in the A coordinate frame. The pre-superscript may be omitted if the coordinate frame is transparent from the context. Post-subscripts are used to denote coordinate systems or components of a vector or matrix. For example, the transformation matrix ${}^{A}\mathbf{\Pi}_{B}$ defines the position and orientation of coordinate system B relative to coordinate frame A; and r_x is the x-component of the vector $\mathbf{r}$.

Vectors denoting points in space, such as ${}^{A}\mathbf{r}$ in (4.4.1), consist of three cartesian coordinates

and a scale factor as the fourth element:

$$ {}^{A}\mathbf{r} = \begin{pmatrix} {}^{A}r_x \\ {}^{A}r_y \\ {}^{A}r_z \\ 1 \end{pmatrix} . \tag{4.4.2} $$

We always use a scale factor of unity. Transformation matrices contain the (3×3) rotational matrix $(\mathbf{n} \quad \mathbf{o} \quad \mathbf{a})$, and the (3×1) translational vector $\mathbf{p}$[54]:

$$ {}^{A}\Pi_B = \begin{pmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{pmatrix} . \tag{4.4.3} $$

The three vector components $\mathbf{n}$, $\mathbf{o}$, and $\mathbf{a}$ of the rotational matrix in (4.4.3) express the orientation of the x, y, and z axes, respectively, of the B coordinate system relative to the A coordinate system and are thus orthonormal. The three components p_x, p_y, and p_z of the translational vector $\mathbf{p}$ express the displacement of the origin of the B coordinate system relative to the origin of the A coordinate system along the x, y, and z axes of the A coordinate system, respectively.

The aforementioned properties of a transformation matrix guarantee that its inverse always has the special form:

$$ {}^{A}\Pi_B^{-1} = \begin{pmatrix} n_x & n_y & n_z & -(\mathbf{p} \cdot \mathbf{n}) \\ o_x & o_y & o_z & -(\mathbf{p} \cdot \mathbf{o}) \\ a_x & a_y & a_z & -(\mathbf{p} \cdot \mathbf{a}) \\ 0 & 0 & 0 & 1 \end{pmatrix} . \tag{4.4.4} $$

Before we define the transformation matrices between the coordinate systems of our WMR model, we compile in Table 4.4.1 our nomenclature for rotational and translational displacements, velocities and accelerations.

In general, any two coordinate systems A and B in our WMR model are located at non-zero x, y and z-coordinates relative to each other. The transformation matrix must therefore contain the translations ${}^{A}d_{Bx}$, ${}^{A}d_{By}$ and ${}^{A}d_{Bz}$. We have assigned all coordinate systems with the z-axes perpendicular to the surface of travel, so that all rotations between coordinate systems are about the z-axis. A transformation matrix in our WMR model thus embodies a rotation ${}^{A}\theta_B$ about the z-axis of coordinate system A and the translations ${}^{A}d_{Bx}$, ${}^{A}d_{By}$ and ${}^{A}d_{Bz}$ along the respective

coordinate axes:

$$ {}^{A}\Pi_{B} = \begin{pmatrix} \cos {}^{A}\theta_{B} & -\sin {}^{A}\theta_{B} & 0 & {}^{A}d_{Bx} \\ \sin {}^{A}\theta_{B} & \cos {}^{A}\theta_{B} & 0 & {}^{A}d_{By} \\ 0 & 0 & 1 & {}^{A}d_{Bz} \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{4.4.5} $$

For zero rotational and translational displacements, the coordinate transformation matrix in (4.4.5) reduces to the identity matrix.

In Section 4.6, we apply the inverse of the transformation matrix in (4.4.5) to calculate position kinematics. By applying the inverse in (4.4.4) to the transformation matrix in (4.4.5), we obtain

$$ {}^{A}\Pi_{B}^{-1} = \begin{pmatrix} \cos {}^{A}\theta_{B} & \sin {}^{A}\theta_{B} & 0 & -{}^{A}d_{Bx}\cos {}^{A}\theta_{B} - {}^{A}d_{By}\sin {}^{A}\theta_{B} \\ -\sin {}^{A}\theta_{B} & \cos {}^{A}\theta_{B} & 0 & {}^{A}d_{Bx}\sin {}^{A}\theta_{B} - {}^{A}d_{By}\cos {}^{A}\theta_{B} \\ 0 & 0 & 1 & -{}^{A}d_{Bz} \\ 0 & 0 & 0 & 1 \end{pmatrix} . \tag{4.4.6} $$

In Section 4.7, we differentiate the transformation matrix in (4.4.5) componentwise to calculate robot velocities:

$$ {}^{A}\dot{\Pi}_{B} = \begin{pmatrix} -{}^{A}\omega_{B}\sin {}^{A}\theta_{B} & -{}^{A}\omega_{B}\cos {}^{A}\theta_{B} & 0 & {}^{A}v_{Bx} \\ {}^{A}\omega_{B}\cos {}^{A}\theta_{B} & -{}^{A}\omega_{B}\sin {}^{A}\theta_{B} & 0 & {}^{A}v_{By} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} , \tag{4.4.7} $$

and in Section 4.8, we differentiate the transformation matrix in (4.4.7) componentwise to calculate robot accelerations:

$$ {}^{A}\ddot{\Pi}_{B} = \begin{pmatrix} -{}^{A}\alpha_{B}\sin {}^{A}\theta_{B} - {}^{A}\omega_{B}^{2}\cos {}^{A}\theta_{B} & -{}^{A}\alpha_{B}\cos {}^{A}\theta_{B} + {}^{A}\omega_{B}^{2}\sin {}^{A}\theta_{B} & 0 & {}^{A}a_{Bx} \\ {}^{A}\alpha_{B}\cos {}^{A}\theta_{B} - {}^{A}\omega_{B}^{2}\sin {}^{A}\theta_{B} & -{}^{A}\alpha_{B}\sin {}^{A}\theta_{B} - {}^{A}\omega_{B}^{2}\cos {}^{A}\theta_{B} & 0 & {}^{A}a_{By} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} . \tag{4.4.8} $$

Table 4.4.1

Scalar Rotational and Translational Displacements

$^{A}\theta_{B}$: The rotational displacement about the z-axis of the A coordinate system between the x-axis of the A coordinate system and the x-axis of the B coordinate system (counterclockwise by convention).

$^{A}d_{Bj}$: (for $j \in [x, y, z]$) : The translational displacement along the j-axis of the A coordinate system between the origin of the A coordinate system and the origin of the B coordinate system.

Scalar Rotational and Translational Velocities

$^{A}\omega_{B}$: The rotational velocity $^{A}\dot{\theta}_{B}$ about the z-axis of the A coordinate system between the x-axis of the A coordinate system and the x-axis of the B coordinate system.

$^{A}v_{Bj}$: (for $j \in [x, y]$) : The translational velocity $^{A}\dot{d}_{Bj}$ along the j-axis of the A coordinate system between the origin of the A coordinate system and the origin of the B coordinate system. Since all motion is in the x-y plane, the z-component $^{A}\dot{d}_{Bz}$ of the translational velocity is zero.

Scalar Rotational and Translational Accelerations

$^{A}\alpha_{B}$: The rotational acceleration $^{A}\ddot{\theta}_{B} = {}^{A}\dot{\omega}_{B}$ about the z-axis of the A coordinate system between the x-axis of the A coordinate system and the x-axis of the B coordinate system.

$^{A}a_{Bj}$: (for $j \in [x, y]$) : The translational acceleration $^{A}\ddot{d}_{Bj} = {}^{B}\dot{v}_{A}$ along the j-axis of the A coordinate system between the origin of the A coordinate system and the origin of the B coordinate system. Since all motion is parallel to the x-y plane, the z-component $^{A}\ddot{d}_{Bz}$ of the translational acceleration is zero.

The assignment of coordinate systems results in two types of transformation matrices between coordinate systems: *constant* and *variable*. The transformation matrix between coordinate systems fixed at two different positions on the same link is constant. Transformation matrices relating the position and orientation of coordinate systems on different links include joint variables and thus are variable. Constant and variable transformation matrices are denoted by ${}^{A}\mathbf{T}_{B}$ and ${}^{A}\mathbf{\Phi}_{B}$, respectively[61]. In Table 4.4.2, we compile the transformation matrices in our WMR model. The constant transformation matrices are the floor-$\bar{r}$obot transformation (${}^{F}\mathbf{T}_{\overline{R}}$), the robot-hip transformation (${}^{R}\mathbf{T}_{H_i}$), the steering-contact transformation (${}^{S_i}\mathbf{T}_{C_i}$) and the floor-$\bar{c}$ontact transformation (${}^{F}\mathbf{T}_{\overline{C_i}}$). Since the instantaneously coincident coordinate systems $\bar{R}$ and $\bar{C}_i$ are stationary relative to the floor coordinate system, all transformation matrices between the floor coordinate system and the instantaneously coincident coordinate systems are constant. The variable transformation matrices are the $\bar{r}$obot-robot transformation (${}^{\overline{R}}\mathbf{\Phi}_{R}$), the hip-steering transformation (${}^{H_i}\mathbf{\Phi}_{S_i}$) and the $\bar{c}$ontact-contact transformation (${}^{\overline{C}_i}\mathbf{\Phi}_{C_i}$). The transformation matrix from a coordinate system to its instantaneously coincident counterpart (or visa-versa) is variable because there is relative motion. We compile the first and second time-derivatives of the variable transformation matrices in Tables 4.4.3 and 4.4.4, respectively. The matrix derivatives involving instantaneously coincident coordinate systems (i.e., ${}^{\bar{R}}\dot{\mathbf{\Phi}}_{R}$, ${}^{\bar{C}_i}\dot{\mathbf{\Phi}}_{C_i}$, ${}^{\bar{R}}\ddot{\mathbf{\Phi}}_{R}$, and ${}^{\bar{C}_i}\ddot{\mathbf{\Phi}}_{C_i}$) are formed by differentiating and simplifying the elements of the transformation matrices ${}^{\bar{R}}\mathbf{\Phi}_{R}$ and ${}^{\bar{C}_i}\mathbf{\Phi}_{C_i}$, respectively, by substituting ${}^{\bar{R}}\theta_{R} = 0$ and ${}^{\bar{C}_i}\theta_{C_i} = 0$. Because of the simplifying substitutions, the second time-derivative of a transformation matrix involving an instantaneously coincident coordinate system cannot be obtained by differentiating the first time-derivative. Time-derivatives of instantaneously coincident coordinate systems are calculated in Section 4.5 by applying matrix coordinate transformation algebra. The time-derivatives of constant transformation matrices are zero.

For wheels which do *not* have steering links, the hip and steering coordinate systems are assigned to coincide with the contact point coordinate system, so that the hip-steering and steering-contact transformation matrices reduce to identity matrices and thereby simplify the ensuing kinematic modeling.

Table 4.4.2 : Transformation Matrices of the WMR Model

Floor – $\overline{R}$obot Transformation :

$$ {}^{F}\mathbf{T}_{\overline{R}} = \begin{pmatrix} \cos {}^{F}\theta_{\overline{R}} & -\sin {}^{F}\theta_{\overline{R}} & 0 & {}^{F}d_{\overline{R}x} \\ \sin {}^{F}\theta_{\overline{R}} & \cos {}^{F}\theta_{\overline{R}} & 0 & {}^{F}d_{\overline{R}y} \\ 0 & 0 & 1 & {}^{F}d_{\overline{R}z} \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$\bar{R}$obot – Robot Transformation :

$$ {}^{\bar{R}}\Phi_{R} = \begin{pmatrix} \cos {}^{\bar{R}}\theta_{R} & -\sin {}^{\bar{R}}\theta_{R} & 0 & {}^{\bar{R}}d_{R} \\ \sin {}^{\bar{R}}\theta_{R} & \cos {}^{\bar{R}}\theta_{R} & 0 & {}^{\bar{R}}d_{R} \\ 0 & 0 & 1 & {}^{\bar{R}}d_{R} \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

Robot – Hip Transformation :

$$ {}^{R}\mathbf{T}_{H_i} = \begin{pmatrix} \cos {}^{R}\theta_{H_i} & -\sin {}^{R}\theta_{H_i} & 0 & {}^{R}d_{H_ix} \\ \sin {}^{R}\theta_{H_i} & \cos {}^{R}\theta_{H_i} & 0 & {}^{R}d_{H_iy} \\ 0 & 0 & 1 & {}^{R}d_{H_iz} \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

Hip – Steering Transformation :

$$ {}^{H_i}\Phi_{S_i} = \begin{pmatrix} \cos {}^{H_i}\theta_{S_i} & -\sin {}^{H_i}\theta_{S_i} & 0 & 0 \\ \sin {}^{H_i}\theta_{S_i} & \cos {}^{H_i}\theta_{S_i} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

Steering – Contact Transformation :

$$ {}^{S_i}\mathbf{T}_{C_i} = \begin{pmatrix} \cos {}^{S_i}\theta_{C_i} & -\sin {}^{S_i}\theta_{C_i} & 0 & {}^{S_i}d_{C_ix} \\ \sin {}^{S_i}\theta_{C_i} & \cos {}^{S_i}\theta_{C_i} & 0 & {}^{S_i}d_{C_iy} \\ 0 & 0 & 1 & {}^{S_i}d_{C_iz} \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$\bar{C}$ontact – Contact Transformation :

$$ {}^{\bar{C}_i}\Phi_{C_i} = \begin{pmatrix} \cos {}^{\bar{C}_i}\theta_{C_i} & -\sin {}^{\bar{C}_i}\theta_{C_i} & 0 & {}^{\bar{C}_i}d_{C_i} \\ \sin {}^{\bar{C}_i}\theta_{C_i} & \cos {}^{\bar{C}_i}\theta_{C_i} & 0 & {}^{\bar{C}_i}d_{C_i} \\ 0 & 0 & 1 & {}^{\bar{C}_i}d_{C_i} \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

Floor – $\overline{C}$ontact Transformation :

$$ {}^{F}\mathbf{T}_{\overline{C}_i} = \begin{pmatrix} \cos {}^{F}\theta_{\overline{C}_i} & -\sin {}^{F}\theta_{\overline{C}_i} & 0 & {}^{F}d_{\overline{C}_ix} \\ \sin {}^{F}\theta_{\overline{C}_i} & \cos {}^{F}\theta_{\overline{C}_i} & 0 & {}^{F}d_{\overline{C}_iy} \\ 0 & 0 & 1 & {}^{F}d_{\overline{C}_iz} \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

Table 4.4.3 : Transformation Matrix Time-Derivatives

$\bar{R}obot - Robot$:

$$\dot{\bar{R}}\dot{\bar{\Phi}}_R = \begin{pmatrix} 0 & -\bar{R}\omega_R & 0 & \bar{R}v_{Rx} \\ \bar{R}\omega_R & 0 & 0 & \bar{R}v_{Ry} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$Hip - Steering$:

$$\dot{}^{H_i}\dot{\bar{\Phi}}_{S_i} = \begin{pmatrix} -{}^{H_i}\omega_{S_i} \sin {}^{H_i}\theta_{S_i} & -{}^{H_i}\omega_{S_i} \cos {}^{H_i}\theta_{S_i} & 0 & 0 \\ {}^{H_i}\omega_{S_i} \cos {}^{H_i}\theta_{S_i} & -{}^{H_i}\omega_{S_i} \sin {}^{H_i}\theta_{S_i} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$\bar{C}ontact - Contact$:

$$\dot{\bar{C}}_i\dot{\bar{\Phi}}_{C_i} = \begin{pmatrix} 0 & -\bar{C}_i\omega_{C_i} & 0 & \bar{C}_iv_{C_ix} \\ \bar{C}_i\omega_{C_i} & 0 & 0 & \bar{C}_iv_{C_iy} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Table 4.4.4 : Transformation Matrix Second Time-Derivatives

$\bar{R}obot - Robot$:

$$\ddot{\bar{R}}\ddot{\bar{\Phi}}_R = \begin{pmatrix} -\bar{R}\omega_R^2 & -\bar{R}\alpha_R & 0 & \bar{R}a_{Rx} \\ \bar{R}\alpha_R & -\bar{R}\omega_R^2 & 0 & \bar{R}a_{Ry} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$Hip - Steering$:

$$\ddot{}^{H_i}\ddot{\bar{\Phi}}_{S_i} = \begin{pmatrix} -{}^{H_i}\alpha_{S_i} \sin {}^{H_i}\theta_{S_i} - {}^{H_i}\omega_{S_i}^2 \cos {}^{H_i}\theta_{S_i} & -{}^{H_i}\alpha_{S_i} \cos {}^{H_i}\theta_{S_i} + {}^{H_i}\omega_{S_i}^2 \sin {}^{H_i}\theta_{S_i} & 0 & 0 \\ {}^{H_i}\alpha_{S_i} \cos {}^{H_i}\theta_{S_i} - {}^{H_i}\omega_{S_i}^2 \sin {}^{H_i}\theta_{S_i} & -{}^{H_i}\alpha_{S_i} \sin {}^{H_i}\theta_{S_i} - {}^{H_i}\omega_{S_i}^2 \cos {}^{H_i}\theta_{S_i} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$\bar{C}ontact - Contact$:

$$\ddot{\bar{C}}_i\ddot{\bar{\Phi}}_{C_i} = \begin{pmatrix} -\bar{C}_i\omega_{C_i}^2 & -\bar{C}_i\alpha_{C_i} & 0 & \bar{C}_ia_{C_ix} \\ \bar{C}_i\alpha_{C_i} & -\bar{C}_i\omega_{C_i}^2 & 0 & \bar{C}_ia_{C_iy} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

4.5 Matrix Coordinate Transformation Algebra

The kinematics of stationary manipulators are modeled by exploiting the properties of transformation matrices[19]. We formalize the manipulation of transformation matrices in the presense of instantaneously coincident coordinate systems by defining a *matrix coordinate transformation algebra*. The algebra consists of a set of operands and a set of operations which may be applied to the operands. The operands of matrix coordinate transformation algebra are transformation matrices and their first and second time-derivatives (in Section 4.4). The operations are listed in Table 4.5.1 as seven axioms. In the table, A, B, and X are coordinate systems and Π denotes either a constant T transformation matrix or a variable Φ transformation matrix. Matrix coordinate transformation algebra allows the direct calculation of the relative positions, velocities and accelerations of robot coordinate systems (including instantaneously coincident coordinate systems).

Table 4.5.1 : Matrix Coordinate Transformation Algebra Axioms

Identity :	${}^{A}\Pi_{B} = \mathbf{I} \quad for\ B = A\ or\ B = \overline{A}$
Cascade :	${}^{A}\Pi_{B} = {}^{A}\Pi_{X}\ {}^{X}\Pi_{B}$
Inversion :	${}^{A}\Pi_{B} = {}^{B}\Pi_{A}^{-1}$
Zero − Velocity :	${}^{A}\dot{\Pi}_{B} = \mathbf{0} \quad for\ B = A\ or\ \Pi = \mathbf{T}$
Velocity :	${}^{A}\dot{\Pi}_{B} = {}^{A}\dot{\Pi}_{X}\ {}^{X}\Pi_{B} + {}^{A}\Pi_{X}\ {}^{X}\dot{\Pi}_{B}$
Zero − Acceleration :	${}^{A}\ddot{\Pi}_{B} = \mathbf{0} \quad for\ B = A\ or\ \Pi = \mathbf{T}$
Acceleration :	${}^{A}\ddot{\Pi}_{B} = {}^{A}\ddot{\Pi}_{X}\ {}^{X}\Pi_{B} + 2\ {}^{A}\dot{\Pi}_{X}\ {}^{X}\dot{\Pi}_{B} + {}^{A}\Pi_{X}\ {}^{X}\ddot{\Pi}_{B}$

The *identity axiom* is self-evident since neither rotations nor translations are required to transform from a coordinate system to itself or to its instantaneously coincident coordinate system. The *cascade axiom* specifies the order in which transformation matrices are multiplied: the coordinate transformation matrix from the reference system to the destination is the cascade of two coordinate transformation matrices, the first from the reference system to an intermediate coordinate system, and the second from the intermediate coordinate system to the destination. The *inversion axiom* states that the coordinate transformation matrix from a reference coordinate system to a destination coordination system is the inverse of the coordinate transformation matrix from the destination coordinate system to the reference coordinate system.

Just as the multiplication of transformation matrices is specified by the cascade axiom, time-differentiation of transformation matrices is specified by the four velocity and acceleration axioms. Specifically, we cannot differentiate both sides of a matrix transformation equation. For example, if we were to differentiate both sides of the equation ${}^{A}\Pi_{\overline{A}} = \mathbf{I}$, we would obtain the incorrect result that ${}^{A}\dot{\Pi}_{\overline{A}} = \mathbf{0}$ since the velocities between a coordinate system and its instantaneously coincident counterpart are (in general) non-zero. The *zero-velocity* axiom states that the relative velocities between a coordinate system A and itself ($B = A$) or another coordinate system assigned to the same link ($\Pi = \mathbf{T}$) are zero. This is because two coordinate systems assigned to the same link are stationary relative to the link and each other. Similarly, the *zero-acceleration* axiom states that the relative accelerations between a coordinate system A and itself ($B = A$) or another coordinate system assigned to the same link ($\Pi = \mathbf{T}$) are zero. The *velocity* axiom specifies how the time-derivative of a transformation matrix may be expressed in terms of the two cascaded transformation matrices and their time-derivatives. Finally, the *acceleration* axiom specifies how the second time-derivative of a transformation matrix may be expressed in terms of the two cascaded transformation matrices and their first and second time-derivatives.

The matrix coordinate transformation axioms in Table 4.5.1 lead to the *corollaries* in Table 4.5.2 which we apply to the kinematic modeling of WMRs.

Table 4.5.2 : Matrix Coordinate Transformation Algebra Corollaries

$$
\begin{aligned}
\textit{Instantaneous Coincidence}: \quad & {}^{A}\dot{\Pi}_{B} = {}^{A}\dot{\Pi}_{\overline{B}} = {}^{\overline{A}}\dot{\Pi}_{B} = {}^{A}\Pi_{B} \\
\textit{Cascade Position}: \quad & {}^{A}\Pi_{Z} = {}^{A}\Pi_{B}\,{}^{B}\Pi_{C}\,{}^{C}\Pi_{D}\,\ldots\,{}^{Y}\Pi_{Z} \\
\textit{Cascade Velocity}: \quad & {}^{A}\dot{\Pi}_{Z} = {}^{A}\dot{\Pi}_{B}\,{}^{B}\Pi_{Z} + {}^{A}\Pi_{B}\,{}^{B}\dot{\Pi}_{C}\,{}^{C}\Pi_{Z} + \ldots + {}^{A}\Pi_{Y}\,{}^{Y}\dot{\Pi}_{Z} \\
\textit{Cascade Acceleration}: \quad & {}^{A}\ddot{\Pi}_{Z} = {}^{A}\ddot{\Pi}_{B}\,{}^{B}\Pi_{Z} + {}^{A}\Pi_{B}\,{}^{B}\ddot{\Pi}_{C}\,{}^{C}\Pi_{Z} + \ldots + {}^{A}\Pi_{Y}\,{}^{Y}\ddot{\Pi}_{Z} \\
& \quad + 2\,{}^{A}\dot{\Pi}_{B}[{}^{B}\dot{\Pi}_{C}\,{}^{C}\Pi_{Z} + {}^{B}\Pi_{C}\,{}^{C}\dot{\Pi}_{D}\,{}^{D}\Pi_{Z} + \ldots + {}^{B}\Pi_{Y}\,{}^{Y}\dot{\Pi}_{Z}] \\
& \quad + 2\,{}^{A}\Pi_{B}\,{}^{B}\dot{\Pi}_{C}[{}^{C}\dot{\Pi}_{D}\,{}^{D}\Pi_{Z} + \ldots + {}^{C}\Pi_{Y}\,{}^{Y}\dot{\Pi}_{Z}] \\
& \quad + \ldots + 2\,{}^{A}\Pi_{X}\,{}^{X}\dot{\Pi}_{Y}\,{}^{Y}\dot{\Pi}_{Z}
\end{aligned}
$$

We develop the *instantaneous coincidence corollary* by applying the identity and cascade axioms. The instantaneous coincidence corollary simplifies transformation matrix expressions by eliminating the instananeously coincident coordinate systems. The *cascade position corollary* calculates the transformation matrix from a reference coordinate system to a destination coordinate

system which may be kinematically separated from the reference system by a number of cascaded intermediate coordinate systems. The cascade position corollary, which is derived by repeated applications of the cascade axiom, is the foundation of position kinematics (in Section 4.6). The *cascade velocity corollary* is derived by repeated applications of the velocity axiom and the cascade axiom. The *cascade acceleration corollary* is derived by repeated applications of the cascade, velocity and acceleration axioms. In Sections 4.7 and 4.8, we apply the cascade velocity and cascade acceleration corollaries to relate linear and angular velocities and accelerations between coordinate systems. Throughout Section 4.7, we apply the axioms and corollaries of the matrix coordinate transformation algebra to derive the wheel Jacobian matrix.

4.6 Position Kinematics

We apply the transformation matrices (in Section 4.4) and the matrix coordinate transformation algebra (in Section 4.5) to calculate position kinematics. The practical position relationships in WMR control require the calculation of the position of a point (e.g., $\mathbf{r}$) relative to one coordinate system (e.g., A) from the position of the point relative to another coordinate system (e.g., Z). For example, we calculate the position of the point mass relative to the floor coordinate system from the position of the point mass in a steering link relative to the steering coordinate system.

We transform position vectors by applying the transformation matrix in (4.4.1):

$$^{A}\mathbf{r} = {}^{A}\Pi_{Z}\ {}^{Z}\mathbf{r}\ . \tag{4.6.1}$$

When the transformation matrix ${}^{A}\Pi_{Z}$ is not known directly, we apply the cascade position corollary to calculate ${}^{A}\Pi_{Z}$ from known transformation matrices:

$$^{A}\Pi_{Z} = {}^{A}\Pi_{B}\ {}^{B}\Pi_{C}\ {}^{C}\Pi_{D}\ \ldots\ {}^{Y}\Pi_{Z}\ . \tag{4.6.2}$$

We apply transformation graphs to determine whether there is a complete set of known transformation matrices which can be cascaded to create the desired ${}^{A}\Pi_{Z}$. In Figure 4.6.1, we display a transformation graph of a WMR with one steering link per wheel.

The origin of each coordinate system is represented by a dot, and transformations between coordinate systems are depicted by directed arrows. The transformation in the direction opposing an arrow is calculated by applying the inversion axiom. Finding a cascade of transformations to calculate a desired transformation matrix (e.g. ${}^{F}\Pi_{S_1}$) is thus equivalent to finding a path from the reference coordinate system of the desired transformation (F) to the destination coordinate system (S_1). The matrices to be cascaded are listed by traversing the path in order. Each transformation in the path which is traversed from the tail to the head of an arrow is listed as the matrix itself, while transformations traversed from the head to the tail are listed as the inverse of the matrix.

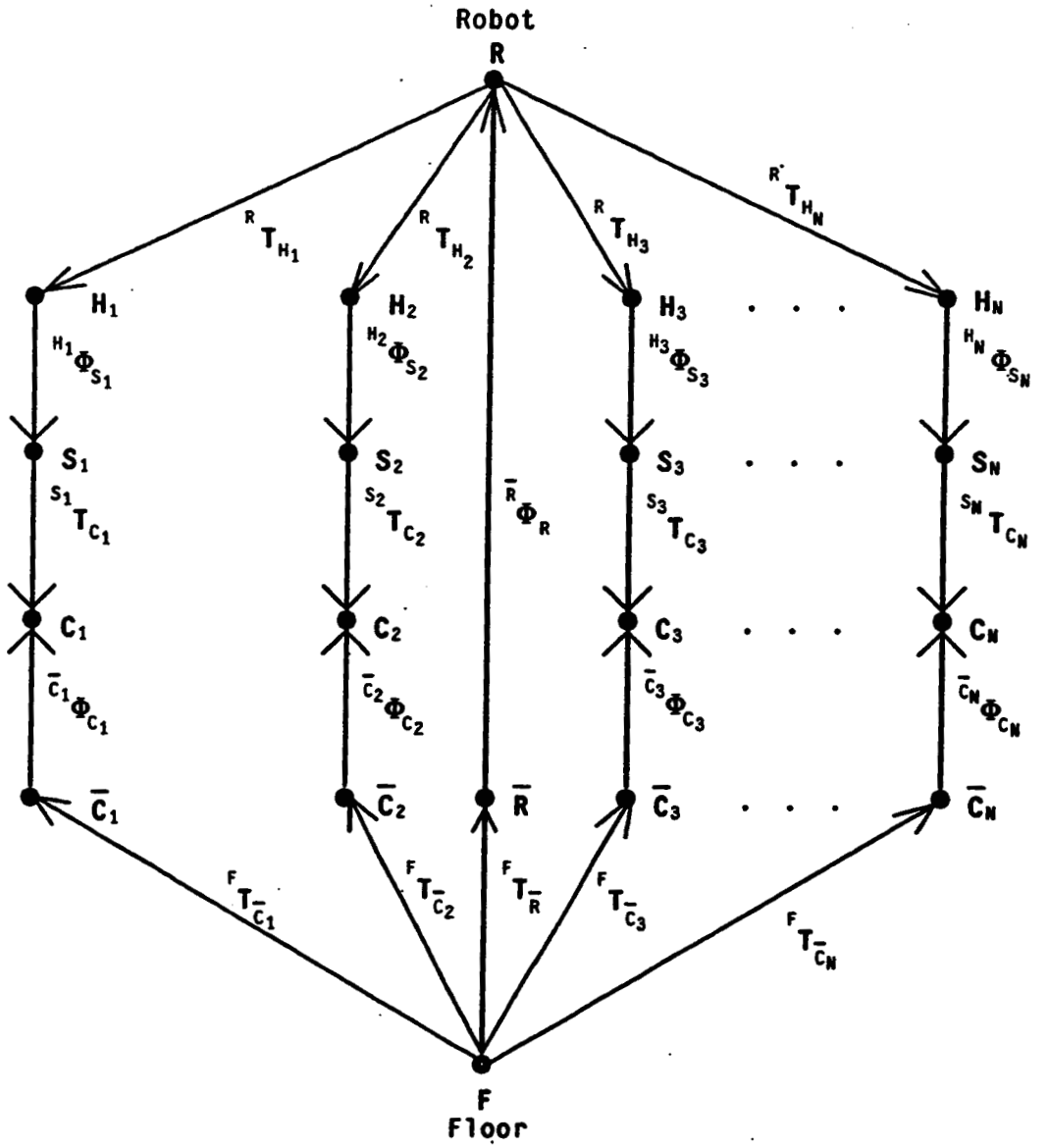


Figure 4.6.1

Transformation Graph of a WMR

For example, the point mass in Figure 4.6.2 located at position $\mathbf{r}$ relative to the steering coordinate system S_1 is transformed to its position relative to the floor coordinate system F according to:

$$^{F}\mathbf{r} = {}^{F}\Pi_{S_1}\, {}^{S_1}\mathbf{r}\,, \tag{4.6.3}$$

where

$$^{F}\Pi_{S_1} = {}^{F}\mathbf{T}_{\bar{R}}\, {}^{\bar{R}}\Phi_{R}\, {}^{R}\mathbf{T}_{H_1}\, {}^{H_1}\Phi_{S_1}\,. \tag{4.6.4}$$

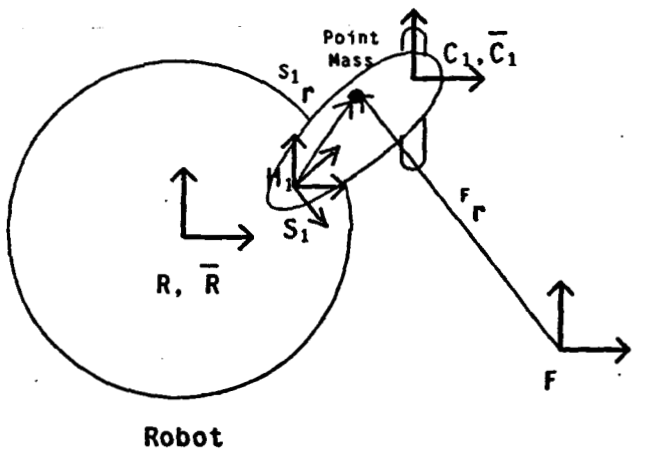


Figure 4.6.2

Point Mass in the Steering Link

In this example, the reference coordinate system is the floor coordinate system F and the destination coordinate system is the steering coordinate system S_1. There are multiple paths between any two coordinate systems in Figure 4.6.1 because WMRs are closed-link structures. In practice, the number of feasible paths is reduced because some of the transformation matrices are unknown. For example, we may seek to calculate the desired transformation matrix in (4.6.4) as:

$$^F\Pi_{S_1} = {}^F\Phi_{\overline{C}_1}\ {}^{\overline{C}_1}\Phi_{C_1}\ {}^{C_1}\mathbf{T}_{S_1} \tag{4.6.5}$$

but the transformation matrix from the floor to the wheel contact point ${}^F\mathbf{T}_{\overline{C}_1}$ is typically unknown.

4.7 Velocity Kinematics

4.7.1 Introduction

We relate the velocities of the WMR by applying the matrix coordinate transformation algebra axioms and the cascade velocity corollary. In Section 4.7.2, we calculate the velocity of a point (e.g., $\mathbf{r}$) relative to a coordinate system (e.g., A), when the position of the point is fixed relative to another moving coordinate system (e.g., Z). This solution is applicable to the dynamic modeling of WMRs (in Section 9) for computing the velocity of a differential mass element on the WMR relative to the floor coordinate system. In Section 4.7.3, we apply this same methodology to calculate the velocities of the robot relative to the instantaneously coincident robot coordinate system when the velocities of a wheel[2] are sensed. We introduce the wheel Jacobian matrix to calculate the robot velocity vector from the wheel velocity vector. We also calculate (in Section 4.7.4) the robot

[2] The wheel velocities are the steering velocity ω_{sz}, the wheel velocity about its axle ω_{wx}, the rotational slip velocity ω_{wz}, the roller velocities ω_{wr} (for omnidirectional wheels) and the rotational velocity ω_{wy} (for ball wheels).

velocity vector relative to the floor coordinate system, when the robot velocity vector is sensed relative to the instantaneously coincident robot coordinate system. In Section 6.3, we apply these calculations to dead reckoning[3] for WMR control.

4.7.2 Point Velocities

We differentiate the point transformation in (4.6.1) with respect to time to compute the velocity of the point $\mathbf{r}$ in the A coordinate system:

$$ {}^{A}\dot{\mathbf{r}} = {}^{A}\dot{\mathbf{\Pi}}_{Z}\ {}^{Z}\mathbf{r} \,. \tag{4.7.1} $$

When the matrix ${}^{A}\dot{\mathbf{\Pi}}_{Z}$ is not known directly, we apply the cascade velocity corollary to calculate ${}^{A}\dot{\mathbf{\Pi}}_{Z}$ from known transformation matrices and known transformation matrix time-derivatives according to:

$$ {}^{A}\dot{\mathbf{\Pi}}_{Z} = {}^{A}\dot{\mathbf{\Pi}}_{B}\ {}^{B}\mathbf{\Pi}_{Z} + {}^{A}\mathbf{\Pi}_{B}\ {}^{B}\dot{\mathbf{\Pi}}_{C}\ {}^{C}\mathbf{\Pi}_{Z} + \ldots + {}^{A}\mathbf{\Pi}_{Y}\ {}^{Y}\dot{\mathbf{\Pi}}_{Z} \tag{4.7.2} $$

For example, equation (4.6.3) relates the position $\mathbf{r}$ of a point mass in the steering coordinate system S_1 to its position in the floor coordinate system F. We calculate the velocity of the point $\mathbf{r}$ relative to the floor coordinate system by differentiating (4.6.3):

$$ {}^{F}\dot{\mathbf{r}} = {}^{F}\dot{\mathbf{\Pi}}_{S_1}\ {}^{S_1}\mathbf{r} \,. \tag{4.7.3} $$

Since the vector ${}^{S_1}\mathbf{r}$ is constant, its time-derivative is zero. We apply the cascade velocity corollary and the WMR transformation graph to obtain an expression for the unknown transformation matrix derivative in (4.7.3):

$$ {}^{F}\dot{\mathbf{\Pi}}_{S_1} = {}^{F}\dot{\mathbf{T}}_{\overline{R}}\ {}^{\overline{R}}\mathbf{\Pi}_{S_1} + {}^{F}\mathbf{T}_{\overline{R}}\ {}^{\overline{R}}\dot{\mathbf{\Phi}}_{R}\ {}^{R}\mathbf{\Pi}_{S_1} + {}^{F}\mathbf{\Pi}_{R}\ {}^{R}\dot{\mathbf{T}}_{H_1}\ {}^{H_1}\mathbf{\Phi}_{S_1} + {}^{F}\mathbf{\Pi}_{H_1}\ {}^{H_1}\dot{\mathbf{\Phi}}_{S_1} \,. \tag{4.7.4} $$

We simplify (4.7.4) to require only known transformation matrices and known transformation matrix derivatives.

$$ \begin{aligned} {}^{F}\dot{\mathbf{\Pi}}_{S_1} &= {}^{F}\mathbf{T}_{\overline{R}}\ {}^{\overline{R}}\dot{\mathbf{\Phi}}_{R}\ {}^{R}\mathbf{\Pi}_{S_1} + {}^{F}\mathbf{\Pi}_{H_1}\ {}^{H_1}\dot{\mathbf{\Phi}}_{S_1} && \textit{Zero-Velocity Axiom} \\ &= {}^{F}\mathbf{T}_{\overline{R}}\ {}^{\overline{R}}\dot{\mathbf{\Phi}}_{R}\ {}^{R}\mathbf{T}_{H_1}\ {}^{H_1}\mathbf{\Phi}_{S_1} + {}^{F}\mathbf{T}_{\overline{R}}\ {}^{\overline{R}}\mathbf{\Phi}_{R}\ {}^{R}\mathbf{T}_{H_1}\ {}^{H_1}\dot{\mathbf{\Phi}}_{S_1} && \textit{Cascade Corollary} \\ &= {}^{F}\mathbf{T}_{\overline{R}}\ {}^{\overline{R}}\dot{\mathbf{\Phi}}_{R}\ {}^{R}\mathbf{T}_{H_1}\ {}^{H_1}\mathbf{\Phi}_{S_1} + {}^{F}\mathbf{T}_{\overline{R}}\ {}^{R}\mathbf{T}_{H_1}\ {}^{H_1}\dot{\mathbf{\Phi}}_{S_1} && \textit{Identity Axiom} \\ &= {}^{F}\mathbf{\Pi}_{R}\ {}^{\overline{R}}\dot{\mathbf{\Phi}}_{R}\ {}^{R}\mathbf{T}_{H_1}\ {}^{H_1}\mathbf{\Phi}_{S_1} + {}^{F}\mathbf{\Pi}_{R}\ {}^{R}\mathbf{T}_{H_1}\ {}^{H_1}\dot{\mathbf{\Phi}}_{S_1} && \textit{Instananteous Coincidence} \end{aligned} \tag{4.7.5} $$

[3] Dead reckoning is the real-time calculation of the WMR position in floor coordinates from wheel sensor measurements.

In (4.7.5), the robot velocity (in ${}^{\bar{R}}\dot{\Phi}_R$) is calculated in the sensed forward solution (in Section 5.7), the steering position (in ${}^{H_i}\Phi_{S_i}$) and velocity (in ${}^{H_i}\dot{\Phi}_{S_i}$) are sensed, the robot position (in ${}^{F}\Pi_R$) is calculated by dead reckoning (in Section 6.3), and the robot-to-hip transformation (${}^{R}\mathbf{T}_{H_i}$) is specified by design. The right-hand side of (4.7.5) is thus known. We then substitute (4.7.5) into (4.7.3) to calculate the velocity of the point mass **r** relative to the floor coordinate system.

4.7.3 Wheel Jacobian Matrix

We formulate the equations-of-motion to model the velocities of the robot in terms of the velocities of a wheel. We begin our development by applying the cascade velocity corollary to write the matrix equation (4.7.6) with the unknown dependent variables (i.e., robot velocities, ${}^{\overline{R}}\dot{\Phi}_R$) on the left-hand side, and the independent variables (i.e., the wheel i velocities, ${}^{H_i}\dot{\Phi}_{S_i}$ and ${}^{\overline{C}_i}\dot{\Phi}_{C_i}$) on the right-hand side:

$$\begin{aligned} {}^{\overline{R}}\dot{\Pi}_R &= {}^{F}\mathbf{T}_{\overline{R}}^{-1}\,{}^{F}\mathbf{T}_{\overline{C}_i}\,{}^{\overline{C}_i}\dot{\Phi}_{C_i}\,{}^{S_i}\mathbf{T}_{C_i}^{-1}\,{}^{H_i}\Phi_{S_i}^{-1}\,{}^{R}\mathbf{T}_{H_i}^{-1} \\ &\quad + {}^{F}\mathbf{T}_{\overline{R}}^{-1}\,{}^{F}\mathbf{T}_{\overline{C}_i}\,{}^{\overline{C}_i}\Phi_{C_i}\,{}^{S_i}\mathbf{T}_{C_i}^{-1}\,{}^{H_i}\dot{\Phi}_{S_i}^{-1}\,{}^{R}\mathbf{T}_{H_i}^{-1}\,. \end{aligned} \tag{4.7.6}$$

The transformation graph of Figure 4.6.1 is utilized to determine the order in which to cascade the transformation matrices; the inversion axiom is applied when an arrow in the transformation graph is traversed from head-to-tail and the zero-velocity axiom is applied to eliminate the matrices which multiply the derivatives of constant **T** matrices. Since the position of the wheel contact point relative to the floor is typically unknown, we apply the cascade position corollary to write an alternative expression for the floor-contact transformation matrix:

$${}^{F}\mathbf{T}_{\overline{C}_i} = {}^{F}\mathbf{T}_{\overline{R}}\,{}^{\overline{R}}\Phi_R\,{}^{R}\mathbf{T}_{H_i}\,{}^{H_i}\Phi_{S_i}\,{}^{S_i}\mathbf{T}_{C_i}\,{}^{\overline{C}_i}\Phi_{C_i}^{-1}\,. \tag{4.7.7}$$

We substitute (4.7.7) into (4.7.6) to obtain:

$$\begin{aligned} {}^{\overline{R}}\dot{\Pi}_R &= {}^{\overline{R}}\Phi_R\,{}^{R}\mathbf{T}_{H_i}\,{}^{H_i}\Phi_{S_i}\,{}^{S_i}\mathbf{T}_{C_i}\,{}^{\overline{C}_i}\Phi_{C_i}^{-1}\,{}^{\overline{C}_i}\dot{\Phi}_{C_i}\,{}^{S_i}\mathbf{T}_{C_i}^{-1}\,{}^{H_i}\Phi_{S_i}^{-1}\,{}^{R}\mathbf{T}_{H_i}^{-1} \\ &\quad + {}^{\overline{R}}\Phi_R\,{}^{R}\mathbf{T}_{H_i}\,{}^{H_i}\Phi_{S_i}\,{}^{H_i}\dot{\Phi}_{S_i}^{-1}\,{}^{R}\mathbf{T}_{H_i}^{-1}\,. \end{aligned} \tag{4.7.8}$$

We apply the identity axiom to simplify (4.7.8).

$$\begin{aligned} {}^{\overline{R}}\dot{\Pi}_R &= {}^{R}\mathbf{T}_{H_i}\,{}^{H_i}\Phi_{S_i}\,{}^{S_i}\mathbf{T}_{C_i}\,{}^{\overline{C}_i}\dot{\Phi}_{C_i}\,{}^{S_i}\mathbf{T}_{C_i}^{-1}\,{}^{H_i}\Phi_{S_i}^{-1}\,{}^{R}\mathbf{T}_{H_i}^{-1} \\ &\quad + {}^{R}\mathbf{T}_{H_i}\,{}^{H_i}\Phi_{S_i}\,{}^{H_i}\dot{\Phi}_{S_i}^{-1}\,{}^{R}\mathbf{T}_{H_i}^{-1} \end{aligned} \tag{4.7.9}$$

We next apply Tables 4.4.2 and 4.4.3 to write the transformation matrices and the transformation matrix derivatives and multiply the result to obtain:

$$\begin{pmatrix} 0 & -{}^{\overline{R}}\omega_R & 0 & {}^{\overline{R}}v_{Rx} \\ {}^{\overline{R}}\omega_R & 0 & 0 & {}^{\overline{R}}v_{Ry} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$= \begin{pmatrix} 0 & -{}^{\overline{C}_i}\omega_{C_i} & 0 & {}^{\overline{C}_i}\omega_{C_i}{}^{R}d_{C_iy} + {}^{\overline{C}_i}v_{C_ix}\cos{}^{R}\theta_{C_i} - {}^{\overline{C}_i}v_{C_iy}\sin{}^{R}\theta_{C_i} \\ {}^{\overline{C}_i}\omega_{C_i} & 0 & 0 & -{}^{\overline{C}_i}\omega_{C_i}{}^{R}d_{C_ix} + {}^{\overline{C}_i}v_{C_ix}\sin{}^{R}\theta_{C_i} + {}^{\overline{C}_i}v_{C_iy}\cos{}^{R}\theta_{C_i} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$+ \begin{pmatrix} 0 & {}^{H_i}\omega_{S_i} & 0 & -{}^{H_i}\omega_{S_i}{}^{R}d_{H_iy} \\ -{}^{H_i}\omega_{S_i} & 0 & 0 & {}^{H_i}\omega_{S_i}{}^{R}d_{H_ix} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{4.7.10}$$

To simplify the notation in (4.7.10), we have made the following substitutions:

$$\begin{aligned} {}^{R}\theta_{H_i} + {}^{H_i}\theta_{S_i} + {}^{S_i}\theta_{C_i} &= {}^{R}\theta_{C_i} \\ {}^{S_i}d_{C_ix}\cos({}^{R}\theta_{H_i} + {}^{H_i}\theta_{S_i}) - {}^{S_i}d_{C_iy}\sin({}^{R}\theta_{H_i} + {}^{H_i}\theta_{S_i}) + {}^{R}d_{H_ix} &= {}^{R}d_{C_ix} \\ {}^{S_i}d_{C_ix}\sin({}^{R}\theta_{H_i} + {}^{H_i}\theta_{S_i}) + {}^{S_i}d_{C_iy}\cos({}^{R}\theta_{H_i} + {}^{H_i}\theta_{S_i}) + {}^{R}d_{H_iy} &= {}^{R}d_{C_iy} \end{aligned} \tag{4.7.11}$$

Upon equating the elements in (4.7.10), we obtain the robot velocities:

$${}^{\overline{R}}\dot{\mathbf{p}}_R = \begin{pmatrix} {}^{\overline{R}}v_{Rx} \\ {}^{\overline{R}}v_{Ry} \\ {}^{\overline{R}}\omega_R \end{pmatrix} = \begin{pmatrix} \cos{}^{R}\theta_{C_i} & -\sin{}^{R}\theta_{C_i} & {}^{R}d_{C_iy} & -{}^{R}d_{H_iy} \\ \sin{}^{R}\theta_{C_i} & \cos{}^{R}\theta_{C_i} & -{}^{R}d_{C_ix} & {}^{R}d_{H_ix} \\ 0 & 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} {}^{\overline{C}_i}v_{C_ix} \\ {}^{\overline{C}_i}v_{C_iy} \\ {}^{\overline{C}_i}\omega_{C_i} \\ {}^{H_i}\omega_{S_i} \end{pmatrix} = \hat{\mathbf{J}}_i\,\dot{\hat{\mathbf{q}}}_i\ , \tag{4.7.12}$$

where $i = 1 \ldots N$ is the wheel index, ${}^{\overline{R}}\dot{\mathbf{p}}_R$ is the vector of robot velocities in the robot frame, $\hat{\mathbf{J}}_i$ is the pseudo-Jacobian matrix for wheel i, and $\dot{\hat{\mathbf{q}}}_i$ is the pseudo-velocity vector for wheel i. We define the number of wheel variables of wheel i to be w_i. The physical velocity vector $\dot{\mathbf{q}}_i$ of typical wheels does not contain the four component velocities in (4.7.12). Typical wheels posses fewer than four wheel variables and thus fewer than four elements in the velocity vector. Furthermore, since all physical wheel motions are rotations about physical wheel axes, the wheel velocity vector contains

the angular velocities of the wheels rather than the linear velocities of the point of contact along the surface of travel. We relate the (4×1) pseudo-velocity vector to the $(w_i \times 1)$ physical velocity vector $\dot{\mathbf{q}}_i$ by the $(4 \times w_i)$ wheel matrix $\mathbf{W}_i$:

$$\dot{\tilde{\mathbf{q}}}_i \;=\; \mathbf{W}_i\, \dot{\mathbf{q}}_i \;. \tag{4.7.13}$$

We substitute (4.7.13) into (4.7.12) to calculate the robot velocities from the wheel velocity vector:

$$^{\overline{R}}\dot{\mathbf{p}}_R \;=\; \hat{\mathbf{J}}_i\, \mathbf{W}_i\, \dot{\mathbf{q}}_i \;=\; \mathbf{J}_i\, \dot{\mathbf{q}}_i \;. \tag{4.7.14}$$

The product $\mathbf{J}_i \;=\; (\hat{\mathbf{J}}_i\, \mathbf{W}_i)$ is the $(3 \times w_i)$ wheel Jacobian matrix of wheel i. The rank of the wheel Jacobian matrix indicates the number of DOFs of the wheel. A wheel having fewer DOFs than wheel variables is *redundant*. The Jacobian matrix of a redundant wheel has dependent columns. We thus formulate the following computational method to determine whether a wheel is non-redundant:

Non-Redundant Wheel Criterion

$$det[\mathbf{J}_i^T \mathbf{J}_i] \neq 0 \tag{4.7.15}$$

Only three different wheels have been utilized in the WMR designs of Section 2: non-steered conventional wheels, steered conventional wheels and omnidirectional wheels. The wheel Jacobian matrices for these wheels and the ball wheel are detailed in Appendix 3. We utilize (4.7.14) in Section 5 to develop the inverse and forward solutions. In Section 6, we apply the matrices in Appendix 3 to calculate the inverse and forward solutions of specific WMRs.

4.7.4 Transforming Robot Velocities

We equate the components in matrix equation (4.7.2) to compute the translational $^{A}v_{Zx}$, and $^{A}v_{Zy}$ and rotational $^{A}\omega_Z$ velocities[4] of the coordinate system Z relative to coordinate system A. We apply this methodology to the practical problem of transforming velocities of the robot from robot coordinates $\overline{R}$ to floor coordinates F. We assume that the floor-robot transformation matrix $^{F}\mathbf{T}_R$ (i.e., the position and orientation of the robot relative to the floor) and the matrix $^{\overline{R}}\dot{\Phi}_R$ (i.e., the velocities of the robot relative to its current position and orientation) are known. The velocities

[4] There are no translational velocities along the z-axis or angular velocities about the x and y-axes because of our coordinate system assignments.

to be calculated (i.e, the velocities of the robot relative to the floor) are the components of the matrix ${}^{F}\dot{\Pi}_{R}$. We apply the cascade velocity corollary (in Section 4.5) and the WMR transformation graph (in Section 4.6) to write the matrix equation

$$ {}^{F}\dot{\Pi}_{R} = {}^{F}\dot{\mathbf{T}}_{\overline{R}}\, {}^{\overline{R}}\Phi_{R} + {}^{F}\mathbf{T}_{\overline{R}}\, {}^{\overline{R}}\dot{\Phi}_{R} \tag{4.7.15} $$

in terms of known matrices. To simplify (4.7.15), we apply the zero-velocity axiom and the instantaneous coincidence corollary:

$$ {}^{F}\dot{\Pi}_{R} = {}^{F}\mathbf{T}_{R}\, {}^{\overline{R}}\dot{\Phi}_{R} \tag{4.7.16} $$

We expand each matrix into scalar components: the matrix derivative ${}^{F}\dot{\Pi}_{R}$ according to (4.4.7), the transformation matrix ${}^{F}\mathbf{T}_{R}$ according to (4.4.5), and the transformation matrix derivative ${}^{\overline{R}}\dot{\Phi}_{R}$ according to Table 4.4.3. Upon multiplying, we obtain:

$$ \begin{pmatrix} -{}^{F}\omega_{R}\sin{}^{F}\theta_{R} & -{}^{F}\omega_{R}\cos{}^{F}\theta_{R} & 0 & {}^{F}v_{Rx} \\ {}^{F}\omega_{R}\cos{}^{F}\theta_{R} & -{}^{F}\omega_{R}\sin{}^{F}\theta_{R} & 0 & {}^{F}v_{Ry} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} $$

$$ = \begin{pmatrix} -{}^{\overline{R}}\omega_{R}\sin{}^{F}\theta_{R} & -{}^{\overline{R}}\omega_{R}\cos{}^{F}\theta_{R} & 0 & {}^{\overline{R}}v_{Rx}\cos{}^{F}\theta_{R} - {}^{\overline{R}}v_{Ry}\sin{}^{F}\theta_{R} \\ {}^{\overline{R}}\omega_{R}\cos{}^{F}\theta_{R} & -{}^{\overline{R}}\omega_{R}\sin{}^{F}\theta_{R} & 0 & {}^{\overline{R}}v_{Rx}\sin{}^{F}\theta_{R} + {}^{\overline{R}}v_{Ry}\cos{}^{F}\theta_{R} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} . \tag{4.7.17} $$

We obtain the angular velocity of the robot ${}^{F}\omega_{R}$ from elements (1,1) and (2,1) and read the translational velocities ${}^{F}v_{Rx}$ and ${}^{F}v_{Ry}$ directly from elements (1,4) and (2,4) of (4.7.17), respectively. We find that:

$$ {}^{F}\dot{\mathbf{p}}_{R} = \begin{pmatrix} {}^{F}v_{Rx} \\ {}^{F}v_{Ry} \\ {}^{F}\omega_{R} \end{pmatrix} = \begin{pmatrix} \cos{}^{F}\theta_{R} & -\sin{}^{F}\theta_{R} & 0 \\ \sin{}^{F}\theta_{R} & \cos{}^{F}\theta_{R} & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} {}^{\overline{R}}v_{Rx} \\ {}^{\overline{R}}v_{Ry} \\ {}^{\overline{R}}\omega_{R} \end{pmatrix} = \mathbf{V}\, {}^{\overline{R}}\dot{\mathbf{p}}_{R} \tag{4.7.18} $$

In (4.7.18), we observe that the angular velocity of the robot is equal in both coordinate frames; whereas the translational velocities in the floor coordinate frame are dependent upon the robot orientation. The matrix $\mathbf{V}$ is the (3×3) *motion matrix* which depends upon the robot position ${}^{F}\mathbf{p}_{R}$. In Section 6.3, we apply the motion matrix to dead-reckoning for WMRs.

4.8 Acceleration Kinematics

We calculate the accelerations of the WMR by applying the cascade acceleration corollary. Since the development paralliels that of the velocity kinematics in Section 4.7, we omit the computational details and concentrate on interpreting the results. We cannot formulate the acceleration equations-of-motion by differentiating the results of Section 4.7, because differentiation of both sides of a transformation matrix equation is not an allowable operation in our matrix coordinate transformation algebra. This is in contrast to the acceleration kinematics of mechanisms containing only lower-pairs (e.g., stationary manipulators) which are formulated by differentiating velocity kinematics.

The acceleration of the point r fixed relative to the moving coordinate system Z is transformed to the A coordinate frame according to:

$$ {}^{A}\ddot{\mathbf{r}} = {}^{A}\ddot{\mathbf{\Pi}}_{Z}\, {}^{Z}\mathbf{r} \,. \tag{4.8.1} $$

We apply the cascade acceleration corollary to calculate the second time-derivative of the transformation matrix ${}^{A}\ddot{\mathbf{\Pi}}_{Z}$.

By applying the cascade acceleration corollary, the component accelerations of the robot (${}^{\overline{R}}a_{Rx}$, ${}^{\overline{R}}a_{Ry}$ and ${}^{\overline{R}}\alpha_{R}$) are related to the wheel accelerations (${}^{H_i}\alpha_{S_i}$, ${}^{\overline{C_i}}a_{C_ix}$, ${}^{\overline{C_i}}a_{C_iy}$, and ${}^{\overline{C_i}}a_{C_ix}$) as the cascade velocity corollary, in Section 4.7.3, relates the robot velocities to the wheel velocities. In the notation of (4.7.11), the robot accelerations are:

$$ \begin{pmatrix} {}^{\overline{R}}a_{Rx} \\ {}^{\overline{R}}a_{Ry} \\ {}^{\overline{R}}\alpha_{R} \end{pmatrix} = \begin{pmatrix} \cos{}^{R}\theta_{C_i} & -\sin{}^{R}\theta_{C_i} & {}^{R}d_{C_iy} & -{}^{R}d_{H_iy} \\ \sin{}^{R}\theta_{C_i} & \cos{}^{R}\theta_{C_i} & -{}^{R}d_{C_ix} & {}^{R}d_{H_ix} \\ 0 & 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} {}^{\overline{C_i}}a_{C_ix} \\ {}^{\overline{C_i}}a_{C_iy} \\ {}^{\overline{C_i}}\alpha_{C_i} \\ {}^{H_i}\alpha_{S_i} \end{pmatrix} $$

$$ + \begin{pmatrix} {}^{R}d_{C_ix} & {}^{R}d_{H_ix} & {}^{R}d_{H_ix} \\ {}^{R}d_{C_iy} & {}^{R}d_{H_iy} & {}^{R}d_{H_iy} \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} {}^{\overline{C_i}}\omega_{C_i}^{2} \\ -2\,{}^{\overline{C_i}}\omega_{C_i}\,{}^{H_i}\omega_{S_i} \\ {}^{H_i}\omega_{S_i}^{2} \end{pmatrix} \tag{4.8.2} $$

The robot accelerations in (4.8.2) are composed of three components: the *self*-accelerations (${}^{\overline{C_i}}a_{C_ix}$, ${}^{\overline{C_i}}a_{C_iy}$, ${}^{\overline{C_i}}\alpha_{C_i}$ and ${}^{H_i}\alpha_{S_i}$); the *centripetal* accelerations (${}^{\overline{C_i}}\omega_{C_i}^{2}$ and ${}^{H_i}\omega_{S_i}^{2}$) having squared velocities; and the *Coriolis* accelerations (${}^{\overline{C_i}}\omega_{C_i}\,{}^{H_i}\omega_{S_i}$) having products of different velocities.

Transforming robot accelerations from robot coordinates to floor coordinates is analogous to transforming robot velocities (in Section 4.7.4). We find that the robot accelerations are transformed from the robot to the floor coordinate frame by the motion matrix V that transforms the

velocities in (4.7.18):

$$ {}^{F}\ddot{\mathbf{p}}_R = \begin{pmatrix} {}^{F}a_{Rx} \\ {}^{F}a_{Ry} \\ {}^{F}\alpha_R \end{pmatrix} = \begin{pmatrix} \cos{}^{F}\theta_R & -\sin{}^{F}\theta_R & 0 \\ \sin{}^{F}\theta_R & \cos{}^{F}\theta_R & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \overline{R}a_{Rx} \\ \overline{R}a_{Ry} \\ \overline{R}\alpha_R \end{pmatrix} = \mathbf{V}\, \ddot{\overline{R}}\mathbf{p}_R \,. \qquad (4.8.3) $$

The acceleration equations-of-motion are not solved in practice because accurate acceleration measurements are difficult to obtain.

4.9 Summary

We have formulated a systematic procedure for modeling the position, velocity and acceleration kinematics of a WMR. In this section, we outline a step-by-step enumeration of the methodology to facilitate engineering applications.

1.) **Make a sketch of the WMR.** Show the relative positioning of the wheels and the steering links. The sketch need not be to scale. A top and a side view are typically sufficient.

2.) **Assign the coordinate systems.** The robot, hip, steering, contact point and floor coordinate systems are assigned according to the conventions introduced in Table 4.3.1.

3.) **Develop the (4×4) coordinate transformation matrices.** The robot-hip, hip-steering, and steering-contact transformation matrices are written according to Table 4.4.1.

4.) **Formulate the position equations-of-motion.** The relative positions and orientations of two coordinate systems are determined by applying the cascade position corollary. The transformation graph of Figure 4.6.1 is utilized to determine the order in which to cascade the matrices.

5.) **Formulate the velocity equations-of-motion.** The equations relating velocities are formulated by applying the cascade velocity corollary. The wheel Jacobian matrix, which relates wheel velocities to robot velocities, may be written directly by substituting components of the transformation matricies into the symbolic wheel Jacobian matricies compiled in Appendix 3.

6.) **Formulate the acceleration equations-of-motion.** The equations relating accelerations are formulated by applying the cascade acceleration corollary.

The *non-redundant wheel criterion* in (4.7.15) is a test on the Jacobian matrix to determine whether a wheel has as many DOFs as wheel variables. We apply this criterion in Section 5 to reveal

disadvantages of redundant wheels. A kinematic model; i.e., the position, velocity and acceleration equations-of-motion, may be applied to the dynamic modeling, design and control of a WMR. In these applications, the equations-of-motion are solved to compute unknown variables from constant and sensed variables. In Section 5, we compute the inverse and forward solutions by utilizing the wheel Jacobian matrix (introduced in Section 4.7.3) as the foundation.

5. The Composite Robot Equation

5.1 Introduction

We combine the kinematic equations-of-motion of all of the wheels on a WMR to form the composite robot equation. We then investigate solutions of the composite robot equation and their properties and implications for WMR locomotion. Our investigation illuminates WMR mobility (in Section 5.4), actuation (in Sections 5.5 and 5.6) and sensing (in Sections 5.7 and 5.8).

In Section 5.2, we formulate the *composite robot equation* and in Section 5.3 we discuss the conditions for its solution. We apply the results of Section 5.3 to develop a *mobility characterization tree* in Section 5.4 which allows us to interpret the solubility conditions in terms of the mobility characteristics of the WMR. The mobility characterization tree indicates whether the mobility structure is *determined*, *overdetermined* or *undetermined*, and associates specific mobility characteristics with each possibility. For example, we may apply the mobility characterization tree to determine whether a WMR allows three DOF motion, and if it does not, the tree indicates the motion constraints.

We proceed to solve the composite robot equation by addressing two classical kinematic modeling problems: the actuated inverse solution (in Section 5.5) and the sensed forward solution (in Section 5.7). The *actuated inverse solution* computes the actuated wheel velocities from the robot velocities. For WMR control, we solve only for the velocities of the actuated wheel variables. The solution for *all* of the wheel velocities is a special case which may be obtained by assuming that all of the wheel variables are actuated.

The actuated inverse solution does not guarantee that the specified robot velocities will be attained when the actuated wheel variables are driven to the calculated velocities. We investigate the possible robot motions when the actuated wheel variables attain the velocities computed by the actuated inverse solution in Section 5.6. We develop an *actuation characterization tree*, analogous to the mobility characterization tree, which allows us to determine the actuation structure (determined, overdetermined or undetermined) of a WMR. The actuation characterization tree is applicable for WMR design to avoid overdetermined actuation (which may cause actuator conflict) and undetermined actuation (which allows the WMR uncontrollable DOFs). From our analysis, we are able to determine whether the actuated wheel variables are sufficient for producing all of the motions allowed by the mobility structure.

The *sensed forward solution* in Section 5.7 computes the robot velocities from the sensed wheel velocities and positions. Since a WMR consists of closed kinematic chains, it is not required to sense all of the wheel positions and velocities, and in practice, it is difficult to do so.

In Section 5.8, we develop a *sensing characterization tree* which allows us to determine the character (undetermined, determined or overdetermined) of the WMR sensing. We thus are able to determine whether the sensed wheel variables are sufficient for discerning all of the motions allowed by the mobility structure. Finally, in Section 5.9, we summarize our development.

5.2 Formulation of the Composite Robot Equation

In Section 4.7.3, we developed the wheel Jacobian matrix $\mathbf{J}_i$ by applying velocity kinematics to compute the robot velocity vector $\dot{\mathbf{p}}$ from the wheel velocity vector $\dot{\mathbf{q}}_i$:

$$\dot{\mathbf{p}} = \mathbf{J}_i \dot{\mathbf{q}}_i \qquad for\ i = 1, ..., N\ , \tag{5.2.1}$$

where i is the wheel index, N is the total number of wheels, $\dot{\mathbf{p}}$ is the vector of robot velocities, $\mathbf{J}_i$ is the $(3 \times w_i)$ Jacobian matrix for wheel i, w_i is the number of variables for wheel i, and $\dot{\mathbf{q}}_i$ is the $(w_i \times 1)$ vector of wheel velocities.

The $3N$ wheel equations in (5.2.1) must be solved simultaneously to characterize the WMR motion. We combine the wheel equations to form the *composite robot equation*:

$$\begin{pmatrix} \mathbf{I}_1 \\ \mathbf{I}_2 \\ \vdots \\ \mathbf{I}_N \end{pmatrix} \dot{\mathbf{p}} = \begin{pmatrix} \mathbf{J}_1 & 0 & \dots & 0 \\ 0 & \mathbf{J}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \mathbf{J}_N \end{pmatrix} \begin{pmatrix} \dot{\mathbf{q}}_1 \\ \dot{\mathbf{q}}_2 \\ \vdots \\ \dot{\mathbf{q}}_N \end{pmatrix} \tag{5.2.2}$$

or

$$\mathbf{A}_o \dot{\mathbf{p}} = \mathbf{B}_o \dot{\mathbf{q}} \tag{5.2.3}$$

where the $\mathbf{I}_i$, $for\ i = 1, \ldots, N$, are (3×3) identity matrices, $\mathbf{A}_0$ is a $(3N \times 3)$ matrix, $\mathbf{B}_0$ is a $(3N \times w)$ block diagonal matrix, $w = w_1 + w_2 + \ldots + w_N$ is the total number of wheel variables and $\dot{\mathbf{q}}$ is the composite wheel velocity vector.

Having formulated the matrix equation in (5.2.3) to model the robot motion, we proceed to investigate the solution for the robot velocity vector $\dot{\mathbf{p}}$ in Section 5.3 and its implications for WMR locomotion in Section 5.4.

5.3 Solution of $\mathbf{Ax} = \mathbf{By}$

We characterize WMR mobility (in Section 5.4), actuation (in Section 5.6) and sensing (in Section 5.8) by examining the properties of the solutions of the composite robot equation in (5.2.3). We extend the standard criteria[15] for the systems of linear algebraic equations $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{A}$

is an $(m \times n)$ matrix, $\mathbf{x}$ is a $(n \times 1)$ vector and $\mathbf{b}$ is a $(m \times 1)$ vector, to the solution of the systems of linear algebraic equations

$$\mathbf{A}\mathbf{x} = \mathbf{B}\mathbf{y} \ , \tag{5.3.1}$$

where $\mathbf{B}$ is an $(m \times p)$ matrix and $\mathbf{y}$ is a $(p \times 1)$ vector. Since the composite robot equation (5.2.3) has the form of (5.3.1), solutions of (5.3.1) are directly applicable to the solution of the composite robot equation.

We apply the method of least-squares[15] to compute the vector $\mathbf{x}$ for *overdetermined* (i.e., having fewer variables than independent equations) and *determined* (i.e., having the same number of variables as independent equations) systems of linear algebraic equations:

$$\mathbf{x} = (\mathbf{A}^T \mathbf{A})^{-1}\mathbf{A}^T \mathbf{B}\,\mathbf{y} \ . \tag{5.3.2}$$

The necessary condition for applying the least-squares solution in (5.3.2) is that $\text{rank}(\mathbf{A}) = n$. There is no unique solution for *undetermined* systems (i.e., systems having fewer independent equations than independent variables).

The residual error of the least-squares method is:

$$\mathbf{A}\mathbf{x} - \mathbf{B}\mathbf{y} = [\mathbf{A}(\mathbf{A}^T \mathbf{A})^{-1}\mathbf{A}^T - \mathbf{I}]\,\mathbf{B}\,\mathbf{y} = \Delta(\mathbf{A})\,\mathbf{B}\,\mathbf{y} \ . \tag{5.3.3}$$

We define the *Delta* matrix function $\Delta(\bullet)$ for expository convienience as:

$$\Delta(\mathbf{U}) = \begin{cases} -\mathbf{I} & \textit{for } \mathbf{U} = \textit{null} \\ \mathbf{U}(\mathbf{U}^T \mathbf{U})^{-1}\mathbf{U}^T - \mathbf{I} & \textit{Otherwise} \end{cases} \tag{5.3.4}$$

where the argument $\mathbf{U}$ is a $(c \times d)$ matrix of rank d.

To characterize WMR motion, we must determine whether the least-squares error in (5.3.3) is zero for *all* $\mathbf{y}$. To do so, we may apply either of the following equivalent tests:

$$\Delta(\mathbf{A})\,\mathbf{B} = \mathbf{0} \tag{5.3.5}$$

or

$$\textit{rank}[\mathbf{A};\mathbf{B}] = \textit{rank}[\mathbf{A}] \ . \tag{5.3.6}$$

If either test (5.3.5) or (5.3.6) is satisfied, the least-squares error is zero for all $\mathbf{y}$. The first test in (5.3.5) is apparent from the expression for the least-squares error in (5.3.3). The second test in (5.3.6) states that if the columns of the matrix $\mathbf{B}$ lie in the vector space spanned by the columns

of the matrix **A**, then the vector **By** must also lie in the vector space spanned by the columns of **A** for all **y**. The vector **By** can then be expressed as a linear combination of the columns of **A** by proper choice (via the least-squares solution) of **x**. Similarly, we may determine whether the least-squares error is zero for a *specific* **y** by applying either of the following two equivalent tests:

$$\Delta(\mathbf{A})\,\mathbf{By} = \mathbf{0}\,, \tag{5.3.7}$$

or

$$rank[\mathbf{A};\mathbf{By}] = rank[\mathbf{A}]\,. \tag{5.3.8}$$

We depict in Figure 5.3.1 a tree illustrating the nature of all possible solutions for the vector **x** of the system of linear algebraic equations in (5.3.1). The tree branches (directed arrows) indicate tests on the matrices **A**, **B** and **y** and are numbered for future reference. The leaves (boxes) indicate the corresponding properties of the solution.

As depicted in Figure 5.3.1, the system of linear algebraic equations in (5.3.1) may be determined, overdetermined or undetermined. The top branches, (0) and (1), determine whether the least-squares solution is applicable by testing the rank of the matrix **A**. If the rank of **A** is n (branch (0)). the least-squares solution is applicable and there is a unique solution for some **y**. If the rank of **A** is less than n (branch (1)), the least-squares solution is not applicable indicating that the system is *undetermined* and there is no unique solution for any **y**. An undetermined system has more unknowns than independent equations.

A *determined* system is one in which the number of independent equations (less than or equal to m) equals the number of unknowns (n). The least-squares error is zero for all **y** and thus tests (5.3.3) and (5.3.4) apply at branch (00).

An *overdetermined* system is one in which the number of independent equations is greater than the number of unknowns. The least-squares error of an overdetermined system is thus non-zero for some **y** (branch (01)). Tests (5.3.7) and (5.3.8) are applied at branch (010) to determine whether the least-squares error is zero for a specific **y**. If so, the system is consistent and there is a unique solution. If the least-squares error is non-zero for a specific **y** (branch (011)), the system is inconsistent and there is no exact solution.

In Section 5.4, we apply the solution tree in Figure 5.3.1 to the composite robot equation in (5.2.3) and discuss the implications for WMR mobility characterization.

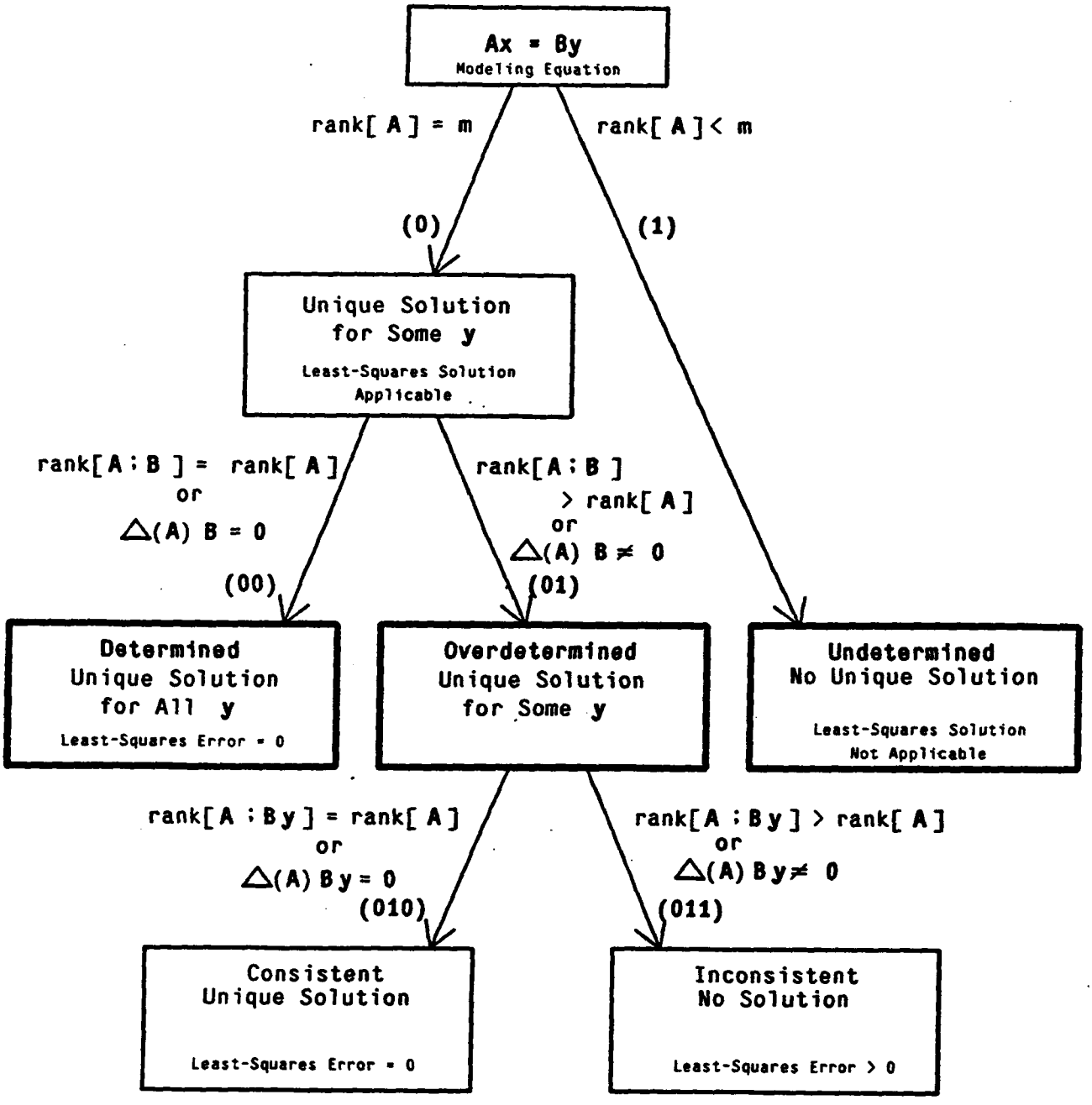


Figure 5.3.1

The Solution Tree for the Vector x in (5.3.1)

5.4 Robot Mobility Characteristics

The composite robot equation in (5.2.3) has the form of the system of linear algebraic equations in Figure 5.3.1, in which $\mathbf{A}_o$, $\mathbf{B}_o$, $\dot{\mathbf{p}}$, and $\dot{\mathbf{q}}$ play the roles of $\mathbf{A}$, $\mathbf{B}$, $\mathbf{x}$ and $\mathbf{y}$, respectively. Since the robot velocity vector $\dot{\mathbf{p}}$ plays the role of the dependent variable, we investigate the conditions

under which the forward solution may be computed. In Figure 5.4.1, we apply the solution tree in Figure 5.3.1 to the composite robot equation in (5.2.3).

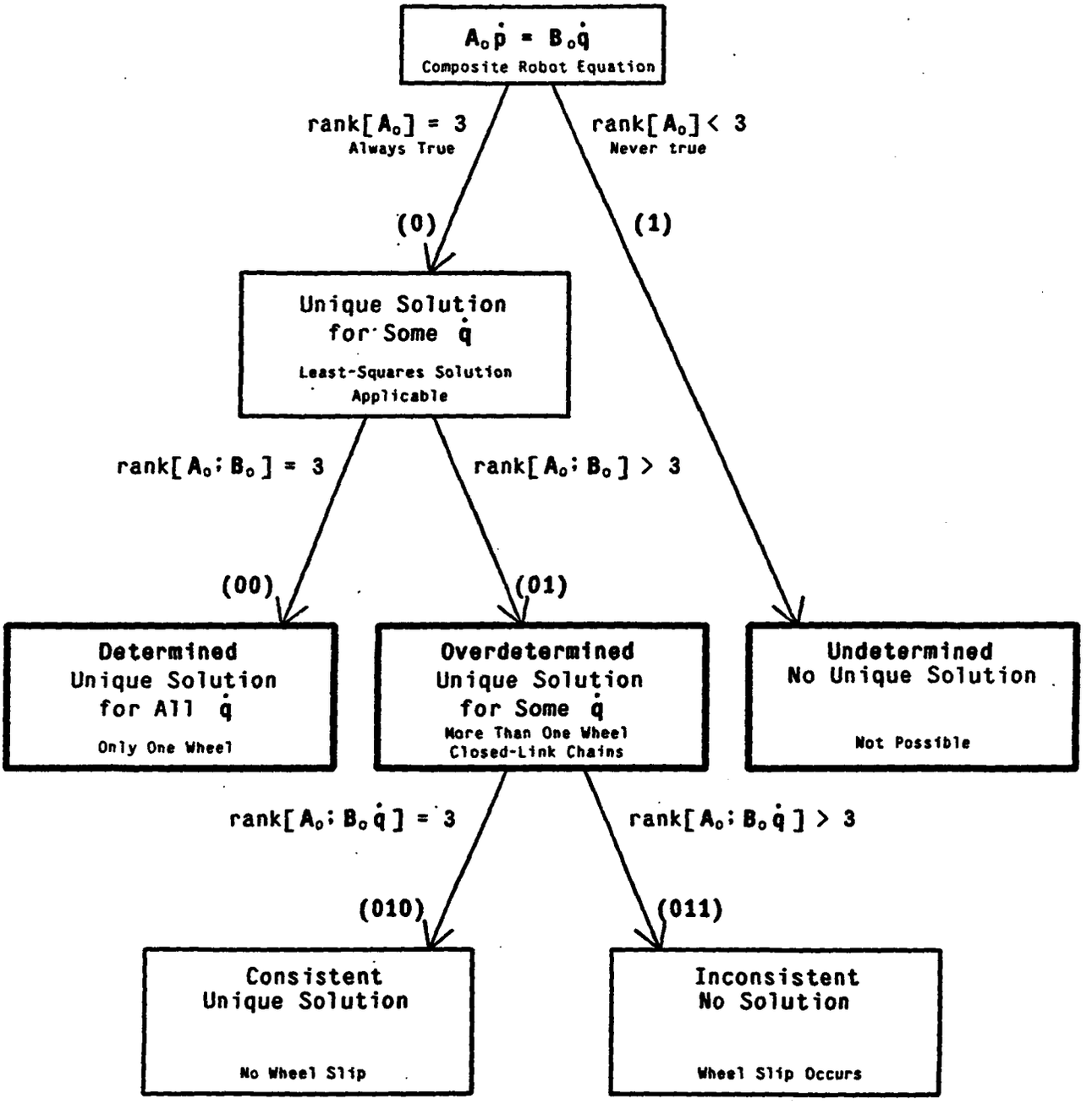


Figure 5.4.1

The Solution Tree for the Robot Velocity Vector $\dot{p}$

By inspection of (5.2.2), we observe that the rank of the $(3N \times 3)$ matrix $\mathbf{A}_o$ is 3 and thus branch (0) always applies. Since branch (1) does not apply, the solution cannot be undetermined; and hence the robot motion is completely specified by the motion of the wheels. From the structure of the matrices $\mathbf{A}_o$ and $\mathbf{B}_o$ in (5.2.2), we observe that the rank of the augmented matrix $[\mathbf{A}_o, \mathbf{B}_o]$ is greater than 3 when there is more than one wheel. A WMR with one wheel is determined (branch (00)), and a WMR with more than one wheel is overdetermined (branch (01)). The overdetermined nature of WMRs having more than one wheel is a consequence of the closed-link kinematic structure of a WMR. As indicated in Figure 5.4.1, the composite robot equation in (5.2.3) will be consistent (and have a solution at branch (010)) or inconsistent (and have no solution at branch (011)) depending upon the wheel velocity vector $\dot{\mathbf{q}}$. Our no-slip assumption (in Section 4.2) ensures that the motions of the wheels and the robot are consistent and that there is thus an exact solution.

We depict in Figure 5.4.1 the solution of the robot velocity vector $\dot{\mathbf{p}}$ from the complete wheel velocity vector $\dot{\mathbf{q}}$. In practice, the wheel velocity vector must be measured by sensors. It is difficult to sense some of the wheel velocities, such as the rotational wheel slip. Since a WMR with more than one wheel has closed-link chains, it is not necessary to sense all of the wheel velocities to calculate the robot velocity because many of the sensor motions are dependent. In Sections 5.7 and 5.8, we investigate the solution of the robot velocity vector from the sensed wheel velocities.

Although the nature of the forward solution of the composite robot equation provides us with little physical insight, we gain significant understanding of WMR motion by investigating the nature of the inverse solution. For WMR control it is not necessary to compute all of the wheel variables in the inverse solution since they are not all actuated. Because of the closed-link chains, moreover not all of the wheel variables must be actuated. In Section 5.5, we compute the actuated inverse solution for the actuated wheel variables. In the remainder of this section, we focus on the *complete* inverse solution to gain physical insight into WMR mobility characteristics.

We investigate the inverse solution by interchanging the roles of the right and left-hand sides of the composite robot equation in (5.2.3) and applying the solution tree in Figure 5.3.1. Thereby, $\mathbf{B}_o$, $\mathbf{A}_o$, $\dot{\mathbf{q}}$ and $\dot{\mathbf{p}}$ in (5.2.3) play the roles of $\mathbf{A}$, $\mathbf{B}$, $\mathbf{x}$ and $\mathbf{y}$ in (5.3.1), respectively. The solution tree for the inverse solution, subsequently referred to as the *mobility characterization tree*, is depicted in Figure 5.4.2 . The branch tests indicated within curly brackets "$\{\bullet\}$" are simplified tests which apply if there are no couplings between wheels.

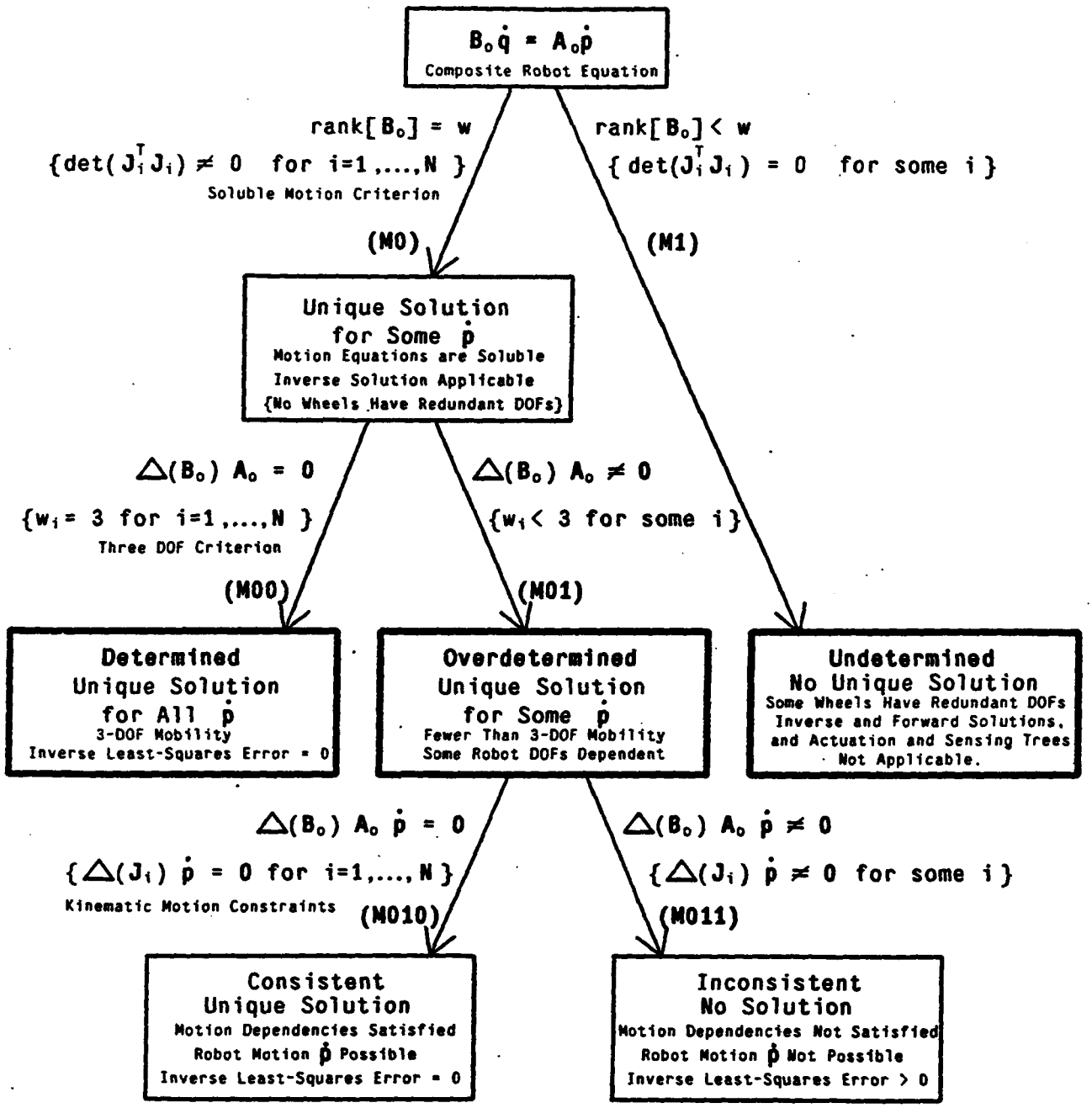


Figure 5.4.2

The Mobility Characterization Tree

The inverse solution can be determined, undetermined or overdetermined depending upon the kinematics (i.e., $\mathbf{B}_o$ and $\mathbf{A}_o$). The top branches test the rank of the $(3N \times w)$ matrix $\mathbf{B}_0$ against the total number of wheel variables w. Since the rank of $\mathbf{B}_0$ is the sum of the ranks of all of the wheel Jacobian matrices when there are no wheel couplings, we test the rank of each wheel

Jacobian matrix $\mathbf{J}_i$ against the number of wheel variables w_i for all wheels $i = 1, \ldots, N$. The rank of the $(3 \times w_i)$ wheel Jacobian matrix $\mathbf{J}_i$ is w_i if the determinant of the matrix $[\mathbf{J}_i^T \mathbf{J}_i]$ is non-zero as indicated by the non-redundant wheel criterion in (4.7.15). We refer to branch test (M0) as the *soluble motion criterion* because it determines whether the composite robot equation can be solved.

Soluble Motion Criterion

$$rank[\mathbf{B}_0] = w \tag{5.4.1}$$

Soluble Motion Criterion With No Wheel Couplings

$$det[\mathbf{J}_i^T \mathbf{J}_i] \neq 0 \quad for \quad i = 1, \ldots, N$$

If the determinant of the matrix $[\mathbf{J}_i^T \mathbf{J}_i]$ is zero, the associated wheel is redundant. *A WMR having redundant wheels and no wheel couplings is undetermined.* We cannot compute the inverse solution for a WMR with redundant wheels. Since the inverse solution is utilized in WMR control (in Section 6.4), we suggest that undetermined mobility structures (i.e., redundant wheels) be avoided.

WMRs without redundant wheels allow some robot motions since there is a unique solution to the system of linear algebraic equations in (5.2.3) for some $\dot{\mathbf{p}}$. Branches (M00) and (M01) test the rank of the augmented matrix $[\mathbf{B}_0; \mathbf{A}_0]$ against the rank of $\mathbf{B}_0$. From their structure in (5.2.2), the ranks of these two matrices are equal when all of the wheel Jacobian matrices are (3×3) and rank 3 (i.e., all of the wheels are non-redundant and possess three DOFs). The mobility structure of a WMR is therefore determined if the test at branch (M00) succeeds. A determined structure has a unique solution for all $\dot{\mathbf{p}}$; i.e., for any desired three dimensional robot velocity vector $\dot{\mathbf{p}}$ there is a wheel velocity vector $\dot{\mathbf{q}}$ which is consistent with the motion. We thus conclude: *The kinematic design of a WMR allows three DOF motion if and only if all of the wheels possess three DOFs.* This requirement is expressed computationally in the *three DOF motion criterion* in (5.4.2).

Three DOF Motion Criterion

$$rank[\mathbf{B}_0] = w \quad and \quad \Delta(\mathbf{B}_0)\,\mathbf{A}_0 = \mathbf{0} \tag{5.4.2}$$

Three DOF Motion Criterion With No Wheel Couplings

$$det[\mathbf{J}_i^T \mathbf{J}_i] \neq 0 \quad and \quad w_i = 3 \quad for \quad i = 1, \ldots, N$$

If branch (M0) succeeds and the WMR does not possess three DOFs, the solution is overdetermined (branch (M01)). The robot does not allow some motions because some of the robot DOFs are dependent. For example, a WMR with a non-steered conventional wheel which satisfies branch (M0) must have an overdetermined mobility structure because no motions perpendicular to the wheel orientation may occur without slip. Branches (M010) and (M011) indicate the possible robot motions $\dot{\mathbf{p}}$ without slip. If the least-squares error is zero, the solution is consistent, and the motion may occur. We thus determine the kinematic constraints on the robot motion by equating the least-squares error to zero in (5.4.3). By examining the structure of the error in (5.4.3), we find an equivalent computationally simpler test in (5.4.3) when there are no couplings between wheels.

Kinematic Motion Contraints

$$\Delta(\mathbf{B}_0)\,\mathbf{A}_0\dot{\mathbf{p}} = \mathbf{0} \tag{5.4.3}$$

Kinematic Motion Constraints With No Wheel Couplings

$$\Delta(\mathbf{J}_i)\,\dot{\mathbf{p}} = 0 \quad for \quad i = 1, \ldots, N$$

We may thus determine the kinematic motion constraints for a WMR without redundant wheels or wheel couplings by considering each wheel independently.

The augmented matrix $[\Delta(\mathbf{B}_0)\ \mathbf{A}_0]$ indicates whether the WMR possesses three DOFs at branch (M00) or fewer than three DOFs at branch (M01). When there are fewer than three DOFs,

the number of independent columns of the matrix $[\Delta(\mathbf{B}_0)\ \mathbf{A}_0]$ specifies the number of dependent robot DOFs. The number of DOFs of a WMR having no redundant wheels is:

Number of WMR DOFs

$$DOFs = 3 - rank[\Delta(\mathbf{B}_0)\ \mathbf{A}_0]\ . \tag{5.4.4}$$

The test at branch (M0) determines whether the *complete* inverse solution for *all* of the wheel variables can be calculated by the least-squares solution. In Section 5.5, we apply the least-squares solution to calculate the *actuated* inverse solution for the actuated wheel variables. Although the actuated inverse solution may exist for some robot velocities $\dot{\mathbf{p}}$ for which the complete inverse solution does not, it is not practical to apply such an actuated inverse solution because the desired robot velocities are constrained by the unactuated wheel variables. We thus utilize the soluble motion criterion in (5.4.1) to indicate when the actuated inverse solution in Section 5.5 is practically applicable.

5.5 Actuated Inverse Solution

We calculate the actuated inverse solution by solving for the actuated wheel velocities in (5.2.3). Because of the closed-link chains in WMRs, we need not actuate all of the wheel variables. To separate the actuated and unactuated wheel variables, we partition the wheel equation in (5.2.1) into two components:

$$\dot{\mathbf{p}} = \mathbf{J}_{ia}\dot{\mathbf{q}}_{ia} + \mathbf{J}_{iu}\dot{\mathbf{q}}_{iu}\ . \tag{5.5.1}$$

The "a" subscripts denote the actuated components and the "u" subscripts denote the unactuated components. We let a_i denote the number of actuated variables, and u_i denote the number of unactuated variables for wheel i (i.e., $a_i + u_i = w_i$). We define the total number of actuated wheel variables to be $a = a_1 + a_2 + \ldots + a_N$ and the total number of unactuated wheel variables to be $u = u_1 + u_2 + \ldots + u_N$. We combine the partitioned wheel equations in (5.5.1) to rewrite the

composite robot equation in (5.2.2) as

$$\begin{pmatrix} \mathbf{I}_1 \\ \mathbf{I}_2 \\ \vdots \\ \mathbf{I}_N \end{pmatrix} \dot{\mathbf{p}} = \begin{pmatrix} \mathbf{J}_{1a} & 0 & \dots & 0 & \mathbf{J}_{1u} & 0 & \dots & 0 \\ 0 & \mathbf{J}_{2a} & \ddots & \vdots & 0 & \mathbf{J}_{2u} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \mathbf{J}_{Na} & 0 & \dots & 0 & \mathbf{J}_{Nu} \end{pmatrix} \begin{pmatrix} \dot{\mathbf{q}}_{1a} \\ \dot{\mathbf{q}}_{2a} \\ \vdots \\ \dot{\mathbf{q}}_{Na} \\ \dot{\mathbf{q}}_{1u} \\ \dot{\mathbf{q}}_{2u} \\ \vdots \\ \dot{\mathbf{q}}_{Nu} \end{pmatrix} , \qquad (5.5.2)$$

or

$$\mathbf{A}_0 \dot{\mathbf{p}} = \mathbf{B}_{0p} \dot{\mathbf{q}}_p \ . \qquad (5.5.3)$$

The $(3N \times w)$ matrix $\mathbf{B}_{0p}$ and the $(w \times 1)$ vector $\dot{\mathbf{q}}_p$ are the *partitioned* counterparts of the matrix $\mathbf{B}_0$ and the vector $\dot{\mathbf{q}}$ in (5.2.2). The soluble motion criterion in (5.4.1) indicates under what conditions the least-squares solution may be practically applied to compute the inverse solution (i.e., $rank[\mathbf{B}_0] = w$). We henceforth assume that the least-squares solution is applicable and that all matrix inverses encountered in its application are computable. We apply the least-squares solution in (5.3.2) to calculate the vector of wheel variables from the robot velocity vector:

$$\dot{\mathbf{q}}_p = (\mathbf{B}_{0p}^T \mathbf{B}_{0p})^{-1} \mathbf{B}_{0p}^T \mathbf{A}_0 \dot{\mathbf{p}} \quad . \qquad (5.5.4)$$

In Appendix 4, we compute the vector of *actuated* wheel velocities $\dot{\mathbf{q}}_a = [\dot{\mathbf{q}}_{ia}^T \ \dots \ \dot{\mathbf{q}}_{Na}^T]^T$ in (5.5.2) as:

Actuated Inverse Solution

$$\dot{\mathbf{q}}_a = \begin{pmatrix} [\mathbf{J}_{1a}^T \Delta(\mathbf{J}_{1u}) \mathbf{J}_{1a}]^{-1} \mathbf{J}_{1a}^T \Delta(\mathbf{J}_{1u}) \\ [\mathbf{J}_{2a}^T \Delta(\mathbf{J}_{2u}) \mathbf{J}_{2a}]^{-1} \mathbf{J}_{2a}^T \Delta(\mathbf{J}_{2u}) \\ \vdots \\ [\mathbf{J}_{Na}^T \Delta(\mathbf{J}_{Nu}) \mathbf{J}_{Na}]^{-1} \mathbf{J}_{Na}^T \Delta(\mathbf{J}_{Nu}) \end{pmatrix} \dot{\mathbf{p}} = \mathbf{J}_a \dot{\mathbf{p}} \ . \qquad (5.5.5)$$

Each $(a_i \times 3)$ block row of the matrix on the right-hand side of (5.5.5), corresponding to the actuated velocities $\dot{\mathbf{q}}_{ia}$, involves only the Jacobian matrix of wheel i. The inverse solution for each

wheel is thus independent of the kinematic equations of all of the $(N-1)$ other wheels. When wheel i is non-redundant with three DOFs and all three wheel variables are actuated, each block row of (5.5.5) simplifies to

$$\dot{\mathbf{q}}_{ia} = (\mathbf{J}_i^{-1})\dot{\mathbf{p}} \ . \tag{5.5.6}$$

We may therefore assume that all of the wheel variables of all of the non-redundant wheels having three DOFs are actuated, apply the inverse Jacobian matrix in (5.5.6) to calculate the wheel velocities, and extract the actuated velocities for robot control. This approach requires approximately one-tenth of the arithmetic operations required for the direct application of (5.5.5).

5.6 Robot Actuation Characteristics

A WMR control engineering application of the actuated inverse solution (in Section 5.5) is to command the velocities of the actuated wheel variables to their calculated values. We investigate the characteristics of the robot motion when the actuated wheel velocities attain the values computed by the actuated inverse solution. We relate the robot velocity vector to the actuated wheel velocities by eliminating the unactuated wheel velocities from the composite robot equation in (5.2.2). Under the no-slip assumption, the unactuated wheel velocities will be consistent and *comply* to the robot motion. We compute the unactuated wheel velocities from the robot velocities in the actuated inverse solution in (5.5.5) by interchanging the roles of the actuated ("a" subscripts) and unactuated ("u" subscripts) variables:

$$\dot{\mathbf{q}}_u = \begin{pmatrix} [\mathbf{J}_{1u}^T \mathbf{\Delta}(\mathbf{J}_{1a})\mathbf{J}_{1u}]^{-1}\mathbf{J}_{1u}^T \mathbf{\Delta}(\mathbf{J}_{1a}) \\ [\mathbf{J}_{2u}^T \mathbf{\Delta}(\mathbf{J}_{2a})\mathbf{J}_{2u}]^{-1}\mathbf{J}_{2u}^T \mathbf{\Delta}(\mathbf{J}_{2a}) \\ \vdots \\ [\mathbf{J}_{Nu}^T \mathbf{\Delta}(\mathbf{J}_{Na})\mathbf{J}_{Nu}]^{-1}\mathbf{J}_{Nu}^T \mathbf{\Delta}(\mathbf{J}_{Na}) \end{pmatrix} \dot{\mathbf{p}} \ . \tag{5.6.1}$$

The conditions guaranteeing the computability of the unactuated and actuated inverse solutions are identical and are indicated in the soluble motion criterion in (5.4.1) . We substitute (5.6.1) into the partitioned composite robot equation in (5.5.2) to obtain:

$$\begin{pmatrix} \mathbf{I} - \mathbf{J}_{1u}[\mathbf{J}_{1u}^T \mathbf{\Delta}(\mathbf{J}_{1a})\mathbf{J}_{1u}]^{-1}\mathbf{J}_{1u}^T \mathbf{\Delta}(\mathbf{J}_{1a}) \\ \mathbf{I} - \mathbf{J}_{2u}[\mathbf{J}_{2u}^T \mathbf{\Delta}(\mathbf{J}_{2a})\mathbf{J}_{2u}]^{-1}\mathbf{J}_{2u}^T \mathbf{\Delta}(\mathbf{J}_{2a}) \\ \vdots \\ \mathbf{I} - \mathbf{J}_{Nu}[\mathbf{J}_{Nu}^T \mathbf{\Delta}(\mathbf{J}_{Na})\mathbf{J}_{Nu}]^{-1}\mathbf{J}_{Nu}^T \mathbf{\Delta}(\mathbf{J}_{Na}) \end{pmatrix} \dot{\mathbf{p}} = \begin{pmatrix} \mathbf{J}_{1a} & 0 & \dots & 0 \\ 0 & \mathbf{J}_{2a} & \ddots & 0 \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \mathbf{J}_{Na} \end{pmatrix} \dot{\mathbf{q}}_a \ , \tag{5.6.2}$$

or

$$\mathbf{A}_a \dot{\mathbf{p}} = \mathbf{B}_a \dot{\mathbf{q}}_a \ . \tag{5.6.3}$$

The *robot actuation equation* in (5.6.3) has the form of (5.3.1) with $\mathbf{A}_a$, $\mathbf{B}_a$, $\dot{\mathbf{p}}$, and $\dot{\mathbf{q}}_a$ playing the roles of $\mathbf{A}$, $\mathbf{B}$, $\mathbf{x}$, and $\mathbf{y}$, respectively. We apply the solution tree in Figure 5.3.1 to (5.6.3) and obtain the *actuation characterization tree* in Figure 5.6.1.

The actuation characterization tree, in analogy with the mobility characterization tree, indicates the properties of the actuation structure of a WMR. The branch tests are developed from the solution tree in Figure 5.3.1. We concentrate on the implications of the solutions.

The system of linear algebraic equations in (5.6.3) representing the actuation structure of the WMR may be determined, undetermined or overdetermined. If branch (A1) succeeds, the actuation structure is undetermined and there is no unique solution for the robot motion $\dot{\mathbf{p}}$. Since we cannot calculate the robot motion, it is unpredictable, and some robot DOFs are uncontrollable. We suggest that undetermined actuation structures be avoided.

If branch (A0) succeeds, we are assured that all robot DOFs are actuated. Specifically, all robot motions allowed by the mobility structure can be produced by the actuators. Consequently, we refer to branch test (A0) as the *adequate actuation criterion*:

Adequate Actuation Criterion

$$det(\mathbf{A}_a^T \mathbf{A}_a) \neq 0 \tag{5.6.4}$$

If the actuation structure is overdetermined (branch (A01)), some of the actuator motions are dependent. If the dependent actuator motions are consistent (at branch (A010)) robot motion is produced, otherwise (at branch (A011)) wheel slip occurs. Any mechanical couplings between actuated wheel variables must satisfy the actuator dependencies to allow robot motion; we therefore refer to branch test (A010) as the *actuator coupling criterion*:

Actuator Coupling Criterion

$$\Delta(\mathbf{A}_a)\, \mathbf{B}_a\, \dot{\mathbf{q}}_a = \mathbf{0} \tag{5.6.5}$$

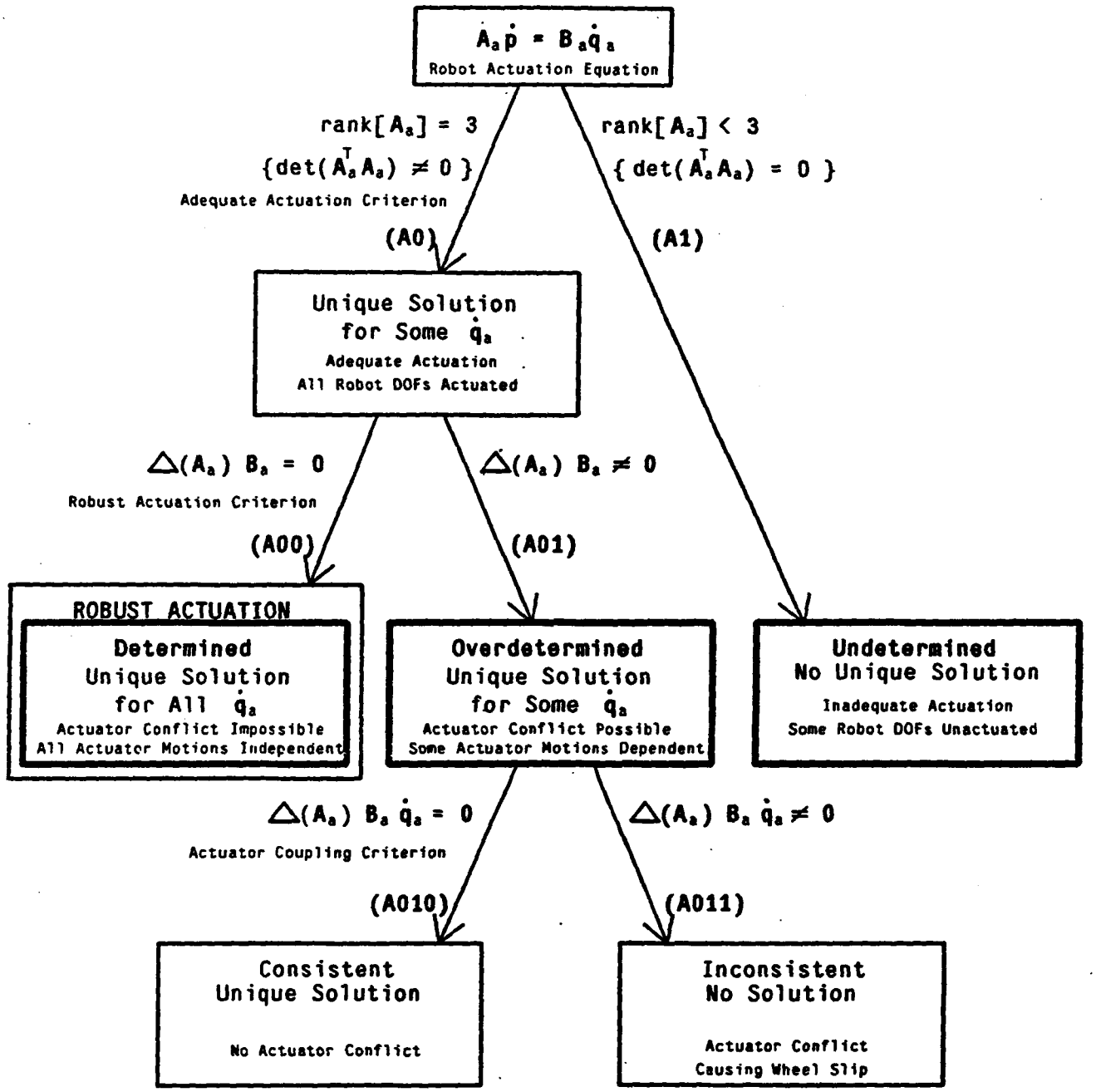


Figure 5.6.1

The Actuation Characterization Tree

If the dependent actuator motions are not consistent (branch (A011)), wheel slip must occur because the least-squares error is non-zero. Since a control system cannot guarantee zero actuator tracking errors, the actuated wheel velocities may deviate from the values computed by the actuated inverse solution. In the presence of these tracking errors, the actuator coupling criterion is

not satisfied and the system of linear algebraic equations in (5.6.3) becomes inconsistent with no solution. We refer to this situation as *actuator conflict* because the forces and torques produced by the inconsistent actuator motions generate stress forces and torques within the WMR structure causing wheel slip instead of generating robot motion. A determined actuation structure (when branch (A00) succeeds) is *robust* in the sense that actuator conflict cannot occur in the presense of actuator tracking errors. The actuator motions are independent and all possible actuated wheel velocity vectors map into unique robot velocity vectors. Branch test (A00) is thus referred to as the *robust actuation criterion*:

Robust Actuation Criterion

$$\Delta(\mathbf{A}_a)\,\mathbf{B}_a = \mathbf{0} \tag{5.6.6}$$

Because of actuator conflict, we suggest that overdetermined actuation structures be avoided. We recommend actuator arrangements leading to a robust (determined) actuation structure. In Sections 5.7 and 5.8, we turn our attention to the sensed forward solution and relate the sensed wheel variables to the robot motion.

5.7 Sensed Forward Solution

The sensed forward solution calculates the robot velocity vector $\dot{\mathbf{p}}$ in (5.2.3) from the sensed wheel positions and velocities $\dot{\mathbf{q}}_s$ and $\dot{\mathbf{q}}_s$. The development of the sensed forward solution parallels the actuated inverse solution in Section 5.5. The first step is to separate the sensed and not-sensed wheel velocities and write (5.2.1) as:

$$\dot{\mathbf{p}} = \mathbf{J}_{is}\dot{\mathbf{q}}_{is} + \mathbf{J}_{in}\dot{\mathbf{q}}_{in}\ . \tag{5.7.1}$$

The subscripts "s" and "n" denote the sensed and not-sensed quantities, respectively. The numbers of sensed and not-sensed variables of wheel i are s_i and n_i, respectively (i.e., $s_i + n_i = w_i$). We assume that both the position and velocity of a sensed wheel variable are available. We combine the wheel equations in (5.7.1) for $i = 1, \ldots, N$ to form the partitioned robot sensing equation, with all of the unknown robot and wheel positions and velocities on the left-hand side:

$$\begin{pmatrix} \mathbf{I}_1 & -\mathbf{J}_{1n} & 0 & \ldots & 0 \\ \mathbf{I}_2 & 0 & -\mathbf{J}_{2n} & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ \mathbf{I}_N & 0 & \ldots & 0 & -\mathbf{J}_{Nn} \end{pmatrix} \begin{pmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{q}}_{1n} \\ \dot{\mathbf{q}}_{2n} \\ \vdots \\ \dot{\mathbf{q}}_{Nn} \end{pmatrix} = \begin{pmatrix} \mathbf{J}_{1s} & 0 & \ldots & 0 \\ 0 & \mathbf{J}_{2s} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & \mathbf{J}_{Ns} \end{pmatrix} \begin{pmatrix} \dot{\mathbf{q}}_{1s} \\ \dot{\mathbf{q}}_{2s} \\ \vdots \\ \dot{\mathbf{q}}_{Ns} \end{pmatrix} \tag{5.7.2}$$

or

$$\mathbf{A}_n\dot{\mathbf{p}}_n = \mathbf{B}_s\dot{\mathbf{q}}_s \ . \tag{5.7.3}$$

We define the total number of sensed wheel velocities to be $s = s_1 + \ldots + s_N$ and the total number of not-sensed wheel variables to be $n = n_1 + \ldots + n_N$. Thereby, $\mathbf{A}_n$ is $(3N \times [3+n])$, $\dot{\mathbf{p}}_n$ is $([3+n] \times 1)$, $\mathbf{B}_s$ is $(3N \times s)$ and $\dot{\mathbf{q}}_s$ is $(s \times 1)$. We apply the least-squares solution in (5.3.2) to calculate the vector of robot and not-sensed wheel velocities $\dot{\mathbf{p}}_n$ from the sensed wheel velocity vector $\dot{\mathbf{q}}_s$:

$$\dot{\mathbf{p}}_n = (\mathbf{A}_n^T\mathbf{A}_n)^{-1}\mathbf{A}_n^T\mathbf{B}_s\dot{\mathbf{q}}_s \ . \tag{5.7.4}$$

In Section 5.8, we develop the adequate sensing criterion in (5.8.4) which indicates the conditions under which the sensed forward solution in (5.7.5) is applicable. In the remainder of this section, we assume that the sensed forward solution applies and that all matrix inverses, such as $(\mathbf{A}_n^T\mathbf{A}_n)^{-1}$ in (5.7.4), are computable.

In contrast to the actuated inverse solution, the least-squares forward solution need not produce a zero error because of sensor noise and wheel slippage. In the presense of these error sources, we cannot calculate the exact velocity of the robot. Our least-squares solution does provide an optimal solution by minimizing the sum of the squared errors in the velocity components. Our least-squares forward solution may thus be applied practically to dead-reckoning for a WMR in the presense of sensor noise and wheel slippage.

In Appendix 5, we solve (5.7.4) for the robot velocities $\dot{\mathbf{p}}$. We find that

Sensed Forward Solution

$$\dot{\mathbf{p}} = [\Delta(\mathbf{J}_{1n}) + \Delta(\mathbf{J}_{2n}) + \ldots + \Delta(\mathbf{J}_{Nn})]^{-1}[\Delta(\mathbf{J}_{1n})\mathbf{J}_{1s} \quad \Delta(\mathbf{J}_{2n})\mathbf{J}_{2s} \quad \ldots \quad \Delta(\mathbf{J}_{Nn})\mathbf{J}_{Ns}]\dot{\mathbf{q}}_s$$

or

$$\dot{\mathbf{p}} = \mathbf{J}_s\dot{\mathbf{q}}_s \ . \tag{5.7.5}$$

A wheel without sensed variables does not contribute any columns $\Delta(\mathbf{J}_{in})\mathbf{J}_{is}$ to (5.7.5). Furthermore, if three independent wheel variables are not sensed, the matrix $\Delta(\mathbf{J}_{in})$ is zero. We may thus eliminate the kinematic equations-of-motion of any wheel which has three not-sensed DOFs

in the calculation of the sensed forward solution. We note that the Jacobian matrix of a steered wheel depends upon the steering angle. Therefore, if any wheel variables of a steered wheel are sensed, the steering angle must also be sensed so that $\mathbf{J}_{in}$ and $\mathbf{J}_{is}$ are computable. Since the matrix $[\Delta(\mathbf{J}_{1n}) + \Delta(\mathbf{J}_{2n}) + \ldots + \Delta(\mathbf{J}_{Nn})]$ is (3×3), solving the system of linear algebraic equations in (5.7.5) for the robot velocities $\dot{\mathbf{p}}$ is *not* a computational burden.

5.8 Robot Sensing Characteristics

The relationship between the sensed wheel variables and the robot motion is the dual of the relationship between the actuated wheel variables and the robot motion. Our development thus parallels the discussion in Section 5.6 on actuation characteristics. We begin by rewriting the composite robot equation in (5.2.2) to relate the robot velocity vector to the sensed wheel velocity vector. We express the not-sensed wheel velocities in terms of the robot velocities by applying the actuated inverse solution in (5.5.5) with the not-sensed ("n" subscripts) and sensed ("s" subscripts) wheel velocities playing the roles of the actuated ("a" subscripts) and unactuated ("u" subscripts) wheel velocities, respectively:

$$\dot{\mathbf{q}}_n = \begin{pmatrix} [\mathbf{J}_{1n}^T \Delta(\mathbf{J}_{1s}) \mathbf{J}_{1n}]^{-1} \mathbf{J}_{1n}^T \Delta(\mathbf{J}_{1s}) \\ [\mathbf{J}_{2n}^T \Delta(\mathbf{J}_{2s}) \mathbf{J}_{2n}]^{-1} \mathbf{J}_{2n}^T \Delta(\mathbf{J}_{2s}) \\ \vdots \\ [\mathbf{J}_{Nn}^T \Delta(\mathbf{J}_{Nn}) \mathbf{J}_{Nn}]^{-1} \mathbf{J}_{Nn}^T \Delta(\mathbf{J}_{Nn}) \end{pmatrix} \dot{\mathbf{p}} \tag{5.8.1}$$

The inverse solution is applicable for any WMR satisfying the soluble motion criterion in (5.4.1). We partition the sensed and not-sensed wheel velocities in the composite robot equation in (5.2.2) and substitute (5.8.1) for the not-sensed wheel velocities to obtain:

$$\begin{pmatrix} \mathbf{I} - \mathbf{J}_{1n}[\mathbf{J}_{1n}^T \Delta(\mathbf{J}_{1s}) \mathbf{J}_{1n}]^{-1} \mathbf{J}_{1n}^T \Delta(\mathbf{J}_{1s}) \\ \mathbf{I} - \mathbf{J}_{2n}[\mathbf{J}_{2n}^T \Delta(\mathbf{J}_{2s}) \mathbf{J}_{2n}]^{-1} \mathbf{J}_{2n}^T \Delta(\mathbf{J}_{2s}) \\ \vdots \\ \mathbf{I} - \mathbf{J}_{Nn}[\mathbf{J}_{Nn}^T \Delta(\mathbf{J}_{Nn}) \mathbf{J}_{Nn}]^{-1} \mathbf{J}_{Nn}^T \Delta(\mathbf{J}_{Nn}) \end{pmatrix} \dot{\mathbf{p}} = \begin{pmatrix} \mathbf{J}_{1s} & 0 & \ldots & 0 \\ 0 & \mathbf{J}_{2s} & \ddots & 0 \\ \vdots & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & \mathbf{J}_{Ns} \end{pmatrix} \dot{\mathbf{q}}_s \,, \tag{5.8.2}$$

or

$$\mathbf{A}_s \dot{\mathbf{p}} = \mathbf{B}_s \dot{\mathbf{q}}_s \,. \tag{5.8.3}$$

The *robot sensing equation* in (5.8.3) has the form of (5.3.1) with $\mathbf{A}_s$, $\mathbf{B}_s$, $\dot{\mathbf{p}}$, and $\dot{\mathbf{q}}_s$ playing the roles of $\mathbf{A}$, $\mathbf{B}$, $\mathbf{x}$, and $\mathbf{y}$, respectively. We apply the solution tree of Figure 5.3.1 to the robot sensing equation in (5.8.3) to obtain the *sensing characterization tree* in Figure 5.8.1.

The solution of the robot velocity $\dot{\mathbf{p}}$ from the sensed wheel velocities $\dot{\mathbf{q}}_s$ may be determined, undetermined or overdetermined, depending on the matrices $\mathbf{A}_s$ and $\mathbf{B}_s$. In parallel with WMR actuation, undetermined systems are undesirable because one or more DOFs of the robot motion cannot be discerned from the sensed wheel velocities. Both determined and overdetermined sensing structures allow a unique solution for consistent sensor motions $\dot{\mathbf{q}}_s$. Branch (S0) thus provides the *adequate sensing criteria* in (5.8.4) which specifies whether all WMR motions allowed by the mobility structure are discernable through sensor measurements:

Adequate Sensing Criterion

$$det(\mathbf{A}_s^T \mathbf{A}_s) \neq 0 \tag{5.8.4}$$

The adequate sensing criterion also specifies the conditions under which the sensed forward solution in (5.7.5) is applicable.

Determined sensing structures provide sufficient information for discerning the robot motion. Overdetermined sensing structures become inconsistent in the presence of sensor noise, which is analogous to the impact of actuator tracking errors on overdetermined actuation structures. Our forward solution in (5.7.5) anticipates the overdetermined nature of the sensor measurements and provides the least-squares solution. In the case of actuation, an overdetermined actuator structure causes undesirable actuator conflict. In contrast, redundant (and even inconsistent) information is desirable for the least-squares solution of the robot velocity from sensed wheel velocities. Redundant information in the least-squares solution reduces the effects of sensor noise on the solution of the robot velocity. Overdetermined sensing structures are thereby robust and branch test (S01) is referred to as the *robust sensing criterion*:

Robust Sensing Criterion

$$\Delta(\mathbf{A}_s)\, \mathbf{B}_s \neq \mathbf{0} \tag{5.8.5}$$

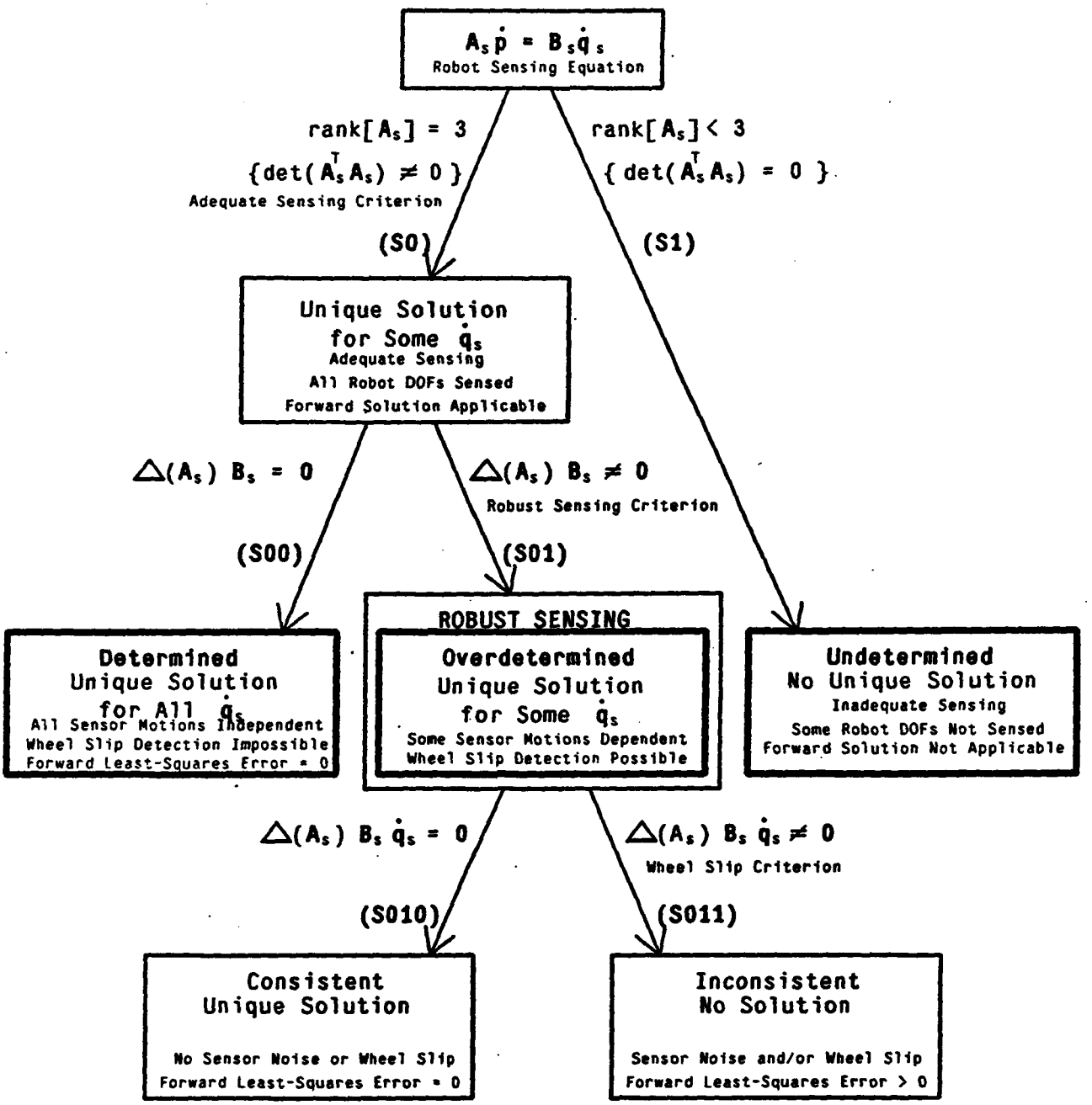


Figure 5.8.1

The Sensing Characterization Tree

We thus recommend that the wheels and wheel sensors be arranged so that the robust sensing criterion is satisfied. When the sensing structure is overdetermined, the least-squared error is zero (at branch (S010)) if there is no wheel slip or sensor noise and non-zero (at branch (S011)) when wheel slip occurs. We therefore denote branch test (S011) as the *wheel slip criterion*:

Wheel Slip Criterion

$$\Delta(\mathbf{A}_s)\,\mathbf{B}_s\dot{\mathbf{q}}_s \neq \mathbf{0} \tag{5.8.6}$$

In Section 6.5, we detect wheel slip by applying the fact that the system of linear algebraic equations in (5.8.3) of a robust sensing structure becomes inconsistent in the presence of wheel slip.

5.9 Conclusions

We have combind the equations-of-motion of each wheel on a WMR to formulate and solve the composite robot equation. The actuated inverse solution in (5.5.5) computes the actuated wheel velocities from the robot velocity vector and is applicable when the *soluble motion criterion* in (5.4.1) is satisfied. We have shown that the actuated inverse solution is calculated independently for each wheel on a WMR. For wheels which possess three DOFs, the actuated inverse solution is calculated directly by applying the inverse wheel Jacobian matrix. The actuated velocities are then extracted for robot control applications.

The sensed forward solution in (5.7.5) is the least-squares solution of the robot velocities in terms of the sensed wheel velocities and is applicable when the *adequate sensing criterion* in (5.8.4) is satisfied. The least-squares forward solution, which minimizes the sum of the squared errors in the velocity components, is the optimal solution of the robot velocities in the presense of sensor noise and wheel slippage. We have found that the sensed forward solution may be simplified by eliminating the equations-of-motion of wheels having three not-sensed DOFs because they do not affect the solution. If any variables of a steered wheel are sensed, the steering angle must also be sensed.

We have discussed the nature of solutions of the composite robot equation and their implications for robot mobility (in Section 5.4), actuation (in Section 5.6) and sensing (in Section 5.8). We have developed the *mobility characterization tree* in Figure 5.4.2 to characterize the motion properties of a WMR. The implications of the mobility characterization tree are summerized by the following insights. If the *soluble motion criterion* in (5.4.1) is satisfied, the actuated inverse solution, actuation and sensing trees, and the WMR DOF calculation in (5.4.4) are applicable. The *three DOF motion criterion* in (5.4.2) indicates whether the WMR kinematic structure allows three DOF motion. If the kinematic structure does not allow three DOF motion, the *kinematic motion constraints* are computed according to (5.4.3). The *number of WMR DOFs* are calculated

from (5.4.4).

The implications of the *actuation characterization tree* in Figure 5.6.1 are summerized by three criteria. The *adequate actuation criterion* in (5.6.4) indicates whether the number and placement of the actuators is adequate for producing all motions allowed by the mobility structure. If the adequate actuation criterion is not satisfied, some robot DOFs are uncontrollable. The *robust actuation criterion* in (5.6.6) determines whether the actuation structure is robust; i.e., actuator conflict cannot occur in the presense of actuator tracking errors. If the actuation structure is adequate but not robust, some actuator motions are dependent. The *actuator coupling criterion* in (5.6.5) calculates these actuator dependencies which must be satisfied to avoid actuator conflict and forced wheel slip.

The *sensing characterization tree* in Figure 5.8.1 indicates properties of the sensing structure of a WMR. The *adequate sensing criterion* in (5.8.4) indicates whether the number and placement of the wheel sensors is adequate for discerning all robot motions allowed by the mobility structure. The *robust sensing criterion* in (5.8.5) indicates whether the sensing structure is such that the calculation of the robot position from wheel sensor measurements is minimally sensative to wheel slip and sensor noise. The *wheel slip criterion* in (5.8.6) provides a computational algorithm for detecting wheel slip in robust sensing structures.

In Section 6, we address the question of three versus two DOFs, the design of WMRs to satisfy kinematic mobility characteristics, and control engineering applications of WMR kinematics. Then, in Section 7, we apply the kinematic modeling of Section 4 and the actuated inverse and sensed forward solutions to prototype WMRs.

6. Applications

6.1 Introduction

WMR kinematics play fundamental roles in design, dynamic modeling, and control. In this section, we illustrate four practical applications of our kinematic methodology: design, dead reckoning, kinematic feedback control and wheel slip detection. We are continuing our study of WMRs by applying our kinematic methodology to the dynamic modeling of WMRs (in Section 9). In Section 6.2, we apply the composite robot equation-of-motion in Section 5 to the *design* of WMRs. We explain how WMRs can be designed to satisfy such desirable mobility characteristics as two and three DOFs, and the ability to actuate and sense the DOFs. *Dead-reckoning* is presented in Section 6.3; the robot velocity calculated from wheel sensor measurements is integrated to calculate the robot position in real-time. We highlight a *kinematics-based WMR control system* (in Section 6.4) by applying the actuated inverse solution in the feedforward path and dead reckoning in the feedback path to reduce the error between the actual robot position and the desired robot position. Knowledge of the robot dynamics will improve control system performance. We apply the kinematic equations-of-motion to *detect wheel slip* in Section 6.5. When a WMR detects the onset of wheel slip, the current robot position is corrected by utilizing slower absolute locating methods (such as computer vision) before continuing motion. The feedback control system can thus track desired trajectories more accurately by continually ensuring an accurate estimate of robot position. Finally, in Section 6.6, we summarize the four applications.

6.2 Design

Just as studying the composite robot equation enables the determination of such mobility characteristics as the number of DOFs, we may design a WMR to possess desirable mobility characteristics. Desirable mobility characteristics which are determinable from an analysis of the composite robot equation are two or three DOFs, and the ability to actuate and sense the motion robustly. By *robust* we mean that the robot motion is insensitive to actuator tracking errors and that the calculation of the robot position from sensor measurements is insensitive to sensor noise and wheel slippage. Designing a WMR to satisfy the desired mobility, actuation and sensing characteristics before construction facilitates the subsequent control system design.

A general-purpose WMR has the ability to move along an X-Y path with an orientation trajectory θ. The WMR thus is capable of controlled motion in the three dimensions x, y, and θ at all times, or equivalently possesses three DOFs. This mobility characteristic is sometimes referred to as omnidirectionality[1]. For a WMR to operate successfully with three DOFs, it must embody the important characteristics tabulated in Table 6.2.1 and discussed below. First, it must

allow three DOF motion. A WMR which possesses three DOFs satisfies the three DOF motion criterion in (5.4.1). An omnidirectional WMR design must thus consist of ball, omnidirectional or non-redundant conventional wheels to allow three DOF motion. A castered backrest used by mechanics for working underneath automobiles has this characteristic.

Table 6.2.1: Design Criteria for an Omnidirectional (3 DOF) WMR

Three DOF Motion :	$det[\mathbf{J}_i^T \mathbf{J}_i] \neq 0$ *and* $w_i = 3$ *for* $i = 1, ..., N$
Adequate Actuation :	$det[\mathbf{A}_a^T \mathbf{A}_a] \neq 0$
Robust Actuation :	$a = 3$
Adequate Sensing :	$det[\mathbf{A}_s^T \mathbf{A}_s] \neq 0$
Robust Sensing :	$s > 3$

Second, all three of the robot DOFs must be actuated to *produce motion in three DOFs.* The placement of wheels and actuators in the WMR design must be chosen to satisfy the adequate actuation criterion in (5.6.4). We require that the actuator structure satisfy the robust actuation criterion in (5.6.6) to avoid actuator conflict. The robust actuation criterion states that there be exactly three actuated wheel variables for the special case of three DOF motion. If there are more than three actuators, their motions must be dependent because robot motion occurs in three dimensions. If there are fewer than three actuators, some robot motions are not actuated and thus not controllable. The design should thus include only three actuators to ensure robust control.

The Unimation robot (in Section 7.2) has three actuated omnidirectional wheels (Troas-whemor) and is an example of a WMR having a robust actuation structure. Uranus (in Section 7.4) has four actuated omnidirectional wheels (Tetroas-whemor) and is not robust because the actuator motions are dependent. In Section 7.4.5, we examine an alternate design of Uranus having a robust actuation structure. Our study of Uranus provides a technique for redesigning adequate actuation structures to be robust.

The third requirement for an omnidirectional WMR is that a control system (e.g., the kinematic feedback control system in Section 6.4) communicates signals to the actuators so that the WMR follows a specified (x, y, θ) trajectory. An omnidirectional WMR which calculates its present position from wheel shaft encoder measurements and controls the actuators to reduce the error

between the desired robot position and the actual robot position possesses this characteristic. To calculate the robot position from wheel shaft encoder measurements, the wheel sensors must be positioned so that the *robot motion may be discerned in three DOFs.* To discern any robot motion, the sensing structure must satisfy the adequate sensing criterion in (5.8.4). We require a robust sensing arrangement (i.e., the WMR design should include more than three wheel sensors) to allow robust calculation of the robot position from wheel sensor measurements.

A WMR which does not allow three DOF motion has *singularities* in its workspace. At a singularity, the WMR cannot attain motion along one or more dimensions (i.e., x, y, or θ). We may determine the kinematic motion constraints of a WMR allowing fewer than three DOFs by computing (5.4.3). Once a WMR design possesses the desired mobility characteristics, we apply the actuation and sensing criteria in Sections 5.6 and 5.8 to verify that the actuation and sensing structures are adequate or robust.

A WMR with two DOFs allows locomotion along any $X-Y$ path and thus has wide applicability for parts and materials transport. Topo[27], Newt (in Section 8.3), and Shakey[52] each possess two DOFs utilizing two diametrically opposed conventional drive wheels. These bicas-polycsun-whemors also have 0,1, and 2 casters, respectively, for stability. We show in Section 7.3 that a design utilizing two diametrically opposed drive wheels is appealing because of its mechanical and modeling simplicity. Because of the practical advantages of two diametrically opposed drive wheels, we recommend the application of bicas-polycsun structures for all tasks requiring fewer than three DOFs. This guideline simplifies the design process for the majority of parts and materials transport applications.

6.3 Dead Reckoning

Dead reckoning is the real-time calculation of the WMR position from wheel sensor measurements. The current robot position is utilized by closed-loop robot control systems, performance monitoring processes and high-level robot planning processes. The least-squares sensed forward solution in (5.7.5) is the exact solution for the robot velocities under the no-slip assumption, if the wheel sensing structure is adequate. The adequate sensing criterion is a prerequisite for implementing three dimensional dead reckoning. To determine the robot position in real-time, the robot velocity is integrated over each sampling period. Since the dead reckoning calculation is erroneous when wheel slip occurs, an alternate method of determining the robot position (e.g., computer vision) must be applied to correct the position calculation before dead reckoning is continued. In Section 6.5, we propose a method to detect the onset of wheel slip.

The integration begins when the robot is at rest or has a sensed initial velocity ${}^F\dot{\mathbf{p}}_R(0)$. The

initial robot position ${}^{F}\mathbf{p}_R(0)$ is either specified or sensed. We assume that the robot motion is adequately modeled by piecewise constant accelerations[1] since the robot is being actuated by constant force/torque generators in each sampling period (the same sampling period as the dead reckoning process). The robot velocity ${}^{\bar{R}}\dot{\mathbf{p}}_R$ in the sampling period from time $t = (n-1)T$ to time $t = nT$ is

$$ {}^{\bar{R}}\dot{\mathbf{p}}_R(t) \;=\; {}^{\bar{R}}\dot{\mathbf{p}}_R[(n-1)T] \;+\; \frac{{}^{\bar{R}}\dot{\mathbf{p}}_R(nT) \;-\; {}^{\bar{R}}\dot{\mathbf{p}}_R[(n-1)T]}{T}\,(t - [(n-1)T])\;, \qquad (6.3.1) $$

where the robot velocity ${}^{\bar{R}}\dot{\mathbf{p}}_R(nT)$ at each sampling instant is calculated by the sensed forward solution in (5.7.5). We transform the robot velocity to the floor coordinate system by applying the velocity transformation in (4.7.18):

$$ {}^{F}\dot{\mathbf{p}}_R(t) \;=\; \mathbf{V}[(n-1)T]\;{}^{\bar{R}}\dot{\mathbf{p}}_R(t)\;. \qquad (6.3.2) $$

We use the angular position of the robot at the sampling instant $t = (n-1)T$ to calculate the motion matrix $\mathbf{V}[(n-1)T]$ since the current angular robot position at time t is unknown. We calculate the robot position at the current sampling instant $t = nT$ by integrating the velocity over the sampling period and adding the result to the robot position at sampling instant $t = (n-1)T$:

$$ {}^{F}\mathbf{p}_R(nT) \;=\; {}^{F}\mathbf{p}_R[(n-1)T] \;+\; \int_{(n-1)T}^{nT} {}^{F}\dot{\mathbf{p}}_R(t)dt\;. \qquad (6.3.3) $$

By subtituting (6.3.1) and (6.3.2) into the integral in (6.3.3), we express the present robot position in terms of the position at the last sampling instant and the robot velocity at the present and last sampling instants:

Dead Reckoning Update Calculation

$$ {}^{F}\mathbf{p}_R(nT) \;=\; {}^{F}\mathbf{p}_R[(n-1)T] \;+\; \frac{T}{2}\,\mathbf{V}[(n-1)T]\,\left\{{}^{\bar{R}}\dot{\mathbf{p}}_R[(n-1)T] \;+\; {}^{\bar{R}}\dot{\mathbf{p}}_R(nT)\right\} \qquad (6.3.4) $$

The computational load for dead reckoning is thus the calculation of the sensed forward solution in (5.7.5).

[1] We apply this assumption as an example. For a specific WMR, it may be necessary to utilise higher-order models of the velocity trajectory.

6.4 Kinematics-Based Feedback Control

The documented WMR control systems are kinematically based[33, 17]; i.e., they do not incorporate a dynamic model of the robot motion. A reference robot trajectory is provided by an independent process (the trajectory planner) and the task of the control system is to produce signals to the wheel actuators so that the WMR tracks the reference trajectory. This is accomplished by wheel level or robot level control (in analogy with joint space or cartesian space control of manipulators [12, 68]).

For *wheel level control*, the reference robot trajectory is applied to generate trajectories for each wheel actuator by calculating the actuated inverse solution. Each wheel actuator is then servoed independently to its calculated trajectory. Each wheel controller may utilize wheel sensors for feedback and a dynamic model of the wheel operating independently, but does not compensate for coupling forces between wheels[50].

Robot level control which utilizes feedback at the robot level is more desirable than wheel level control. A kinematics-based robot level control system is diagramed in Figure 6.4.1. Directed arrows indicate the flow of information. The number of scalar variables represented by each arrow is indicated within the body of the arrow. The computer control algorithm to be executed at each sampling instant T is enumerated in Table 6.4.1 and the sequence of steps is indicated in Figure 6.4.1. At time nT, we sense the wheel variables $\mathbf{q}_s(nT)$ and $\dot{\mathbf{q}}_s(nT)$ and the desired robot position vector ${}^F\mathbf{p}_d(nT)$ in Step 1 of Table 6.4.1 . The $(s \times 1)$ sensor gain vector k_s scales the sensor signals. In Step 2, we apply the sensed forward solution in (5.7.5) to compute the robot velocity ${}^R\dot{\mathbf{p}}_R(nT)$. We apply the dead reckoning update in (6.3.4) in Step 3 to compute the robot position ${}^F\mathbf{p}_R(nT)$. We compare the reference robot position ${}^F\mathbf{p}_d(nT)$ with the actual robot position ${}^F\mathbf{p}_R(nT)$ (in Step 4) to calculate the robot position error ${}^F\mathbf{e}_R(nT)$. The position error is multiplied by the (3×3) feedforward gain vector $\mathbf{k}_f$ and is then transformed to the robot coordinate frame by applying the inverse motion matrix $\mathbf{V}^{-1}(nT)$ in Step 5. Under the assumption that the robot tracking error remains small, the robot position error ${}^R\mathbf{e}_R$ is treated as the differential displacement ${}^R\delta\mathbf{p}_R$. This robot differential displacement is transformed into actuator displacements $\delta\mathbf{q}_a$ (as velocities are transformed) by applying the actuated inverse solution in Step 6:

$$\delta\mathbf{q}_a = \mathbf{J}_a \, {}^R\delta\mathbf{p}_R \, . \tag{6.4.1}$$

In Step 6, we also multiply the computed actuator reference velocities $\dot{\mathbf{q}}_a$ by the $(a \times 1)$ actuator gain vector $\mathbf{k}_a$. The actuator gain vector is the ratio of the actuator set-points to the steady-state actuator velocities under nominal operating conditions and must be determined empirically. The (3×1) feedforward gain $\mathbf{k}_f$ is also adjusted experimentally to provide a fast robot tracking response without excessive robot overshoot or oscillations about the reference trajectory. In Step

7, the resulting actuator set-points are then communicated to the actuator hardware.

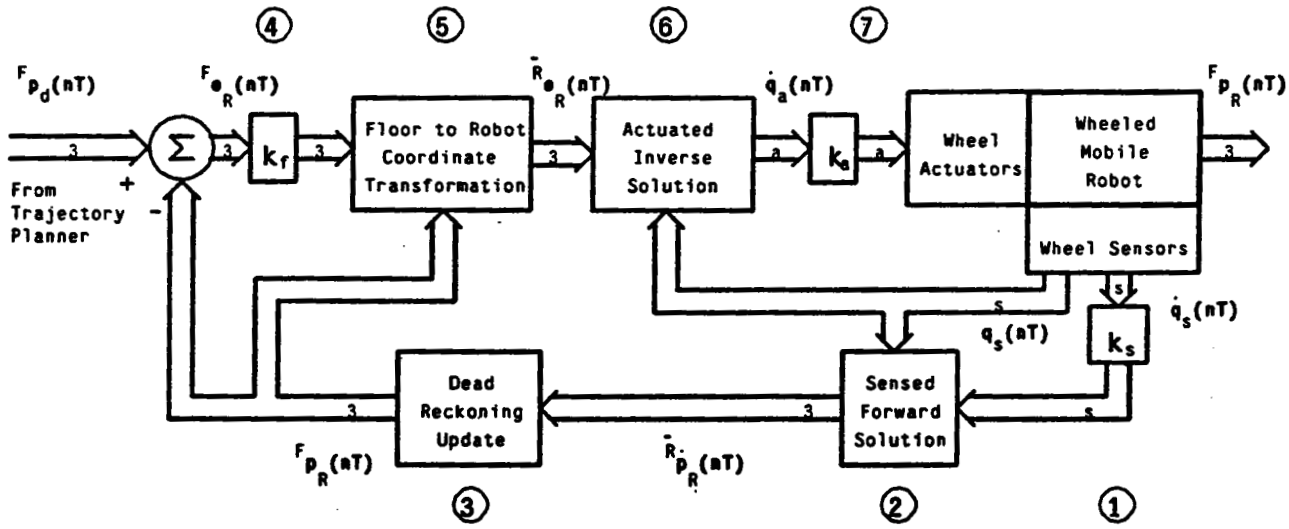


Figure 6.4.1

Kinematics-Based WMR Control System

Table 6.4.1: Kinematics-Based WMR Control Algorithm

1.) Sample $\mathbf{q}_a(nT)$, $\dot{\mathbf{q}}_a(nT)$ and $^{F}\mathbf{p}_d(nT)$

2.) Compute and Store $^{\bar{R}}\dot{\mathbf{p}}_R(nT) = k_a\mathbf{J}_a\dot{\mathbf{q}}_a(nT)$

3.) Compute and Store $^{F}\mathbf{p}_R(nT) = {}^{F}\mathbf{p}_R[(n-1)T] + \frac{T}{2}\mathbf{V}[(n-1)T]\{^{\bar{R}}\dot{\mathbf{p}}_R[(n-1)T] + {}^{\bar{R}}\dot{\mathbf{p}}_R(nT)\}$

4.) Compute $^{F}\mathbf{e}_R(nT) = {}^{F}\mathbf{p}_d(nT) - {}^{F}\mathbf{p}_R(nT)$

5.) Compute $^{\bar{R}}\mathbf{e}_R(nT) = \mathbf{k}_f\mathbf{V}^{-1}(nT)^{F}\mathbf{e}_R(nT)$

6.) Compute $\dot{\mathbf{q}}_a(nT) = \mathbf{k}_a\mathbf{J}_a{}^{\bar{R}}\mathbf{e}_R(nT)$

7.) Communicate the Computed Set-Points $\dot{\mathbf{q}}_a(nT)$ to the Actuators

---

Over the past twenty years, manipulator control systems have improved progressively; from independent joint-space control[55], to kinematics-based cartesian-space control[68], to dynamics-based cartesian-space feedback control[42], to robust dynamics-based feedback control[65] and adaptive control algorithms[21]. We anticipate that future WMR control systems will also incorporate kinematic and dynamic models. Present WMR control system designs are independent wheel level controllers. Future WMR control systems will improve performance once a kinematic methodology (such as our present paper) and dynamic models (outlined in Section 9) become available.

6.5 Wheel Slip Detection

In Section 5.7, we computed the WMR velocity vector from the wheel sensor measurements (i.e., the sensed forward solution), and in Section 5.8 we discussed the characteristics of the solution. We can discern all WMR motions if the adequate sensing criterion is satisfied. If the sensing structure is adequate but not robust, the equations-of-motion will be consistent irrespective of the presense of wheel slip and the error in the least-squares forward solution will be zero. In contrast, for a robust sensing structure (i.e., a sensing structure satisfying the robust sensing criterion), the kinematic equations-of-motion are inconsistent in the presence of wheel slip. The error in the least-squares forward solution is then greater than zero. *We therefore propose to detect the occurrence of wheel slippage for a WMR having a robust sensing structure by calculating the error in the least-squares solution.* In the improbable case that all wheels on a WMR slip simultaneously in such a manner that the equations-of-motion remain consistent, our method will fail to detect the wheel slip.

In practice, sensor noise can also cause the kinematic equations-of-motion to become inconsistent, but we expect that the least-squares error due to sensor noise will be small in comparison with the error caused by wheel slippage. Instead of testing the least-squares error against zero, we propose to compare it with an error threshold e_t set by the worst case sensor noise error. If the least-squares error in the forward solution exceeds the threshold, we conclude that wheel slip has occurred. When a WMR detects that wheel slip has occurred, it should resort to absolute methods of determining its position (e.g., computer vision, ultrasonic ranging sensors, and laser range finders) before continuing the dead-reckoning calculations. Since current locating methods are computationally slow relative to the robot motion, the WMR should halt motion until its dead reckoning calculations are updated by the absolute locating method.

Calculation of the sensed forward solution in (5.7.5) is the first step in determining the least-squares error. The calculated robot velocity vector ${}^{R}\dot{\mathbf{p}}_{R}$ is substituted for the actual robot velocity vector in the robot sensing equation (5.8.3). The least-squares error vector $\mathbf{e}$ is calculated by

subtracting the right-hand side of (5.8.3) from the left-hand side:

$$\mathbf{e} \;=\; \mathbf{A}_s\, {}^{\bar{R}}\dot{\mathbf{p}}_R \;-\; \mathbf{B}_s\, \dot{\mathbf{q}}_s \;. \tag{6.5.1}$$

We calculate and compare the norm of the least-squares error $[\mathbf{e}^T\mathbf{e}]$ with the scalar threshold e_t^2. If the norm of the least-squares error exceeds the threshold, we conclude that wheel slip has occurred:

Detection of Wheel Slip

$$\textit{If} \qquad \mathbf{e}^T\mathbf{e} \;>\; e_t^2 \;, \qquad \textit{wheel slip has occurred} \;. \tag{6.5.2}$$

We note that (6.5.2) is, in principle, equivalent to the wheel slip criterion in (5.8.6) and has the added advantage that the sensed forward solution in (5.7.5) is computed as an intermediate result. The sensed forward solution may then be applied to dead-reckoning and WMR control.

6.6 Summary

We have applied our kinematic methodology to the design, dead reckoning, kinematics-based feedback control and wheel slip detection for WMRs. By proper choice of the wheel type and placement, and the actuator and sensor placement, we may design two and three DOF WMRs. Specifically, we must satisfy the criteria in Table 6.2.1 to achieve a robust omnidirectional WMR design. For two DOFs, a WMR design having two diametrically opposed drive wheels, bicas-polycsun-whemor (e.g., as on the WMRs Newt, Shakey, and Topo), has both mechanical and modeling advantages over other designs. Dead reckoning is the real-time integration of the robot velocity to obtain the robot position. The robot velocity is first calculated by applying the sensed forward solution. We integrate the robot velocity by the update algorithm in (6.3.4) which is a linear function of the robot position and velocity. Current WMR control systems incorporate wheel level algorithms. We have introduced a kinematics-based robot level algorithm which relies on dead reckoning for feedback, and the actuated inverse solution to calculate actuator inputs as feedforward control signals. Future WMR control systems will exhibit enhanced performance by incorporating dynamic models and absolute position feedback. As our final application, we have proposed to detect wheel slippage in robust sensing structures by calculating the least-squares error in the sensed forward solution. If the error exceeds a threshold which can be attributed to wheel sensor noise, we conclude that wheel slip has occurred. By detecting the onset of wheel slippage,

and correcting the calculated robot position with an absolute locating device, the WMR will follow planned trajectories more accurately.

We are also applying our kinematic methodology to the dynamic modeling of WMRs (in Section 9). By analogy with manipulator dynamic modeling, our kinematic methodology will serve as the foundation upon which to formulate the dynamic models. In contrast to manipulator dynamics, we must resolve the special problems of closed-link chains and higher-pair joints.

We note that the composite robot equation in (5.2.2) and the actuated inverse and sensed forward solutions in (5.5.5) and (5.7.5) are essential components of these applications. In Section 7, we apply our kinematic methodology to specific WMRs. For each WMR, we calculate the actuated inverse and sensed forward solutions, where applicable, and characterize their mobility, actuation and sensing structures.

7. Examples

7.1 Introduction

We illustrate the kinematic modeling of six WMRs: the Unimation robot, Newt, Uranus, Neptune, Pluto, and the Stanford cart. For each WMR, we provide four *kinematic descriptions*: a written description, a top and side view sketch, the symbolic diagram and the kinematic name. We *assign the coordinate systems* to create the *coordinate transformation matrices*. We then form the *wheel Jacobian matrices* by substituting elements of the coordinate transformation matrices into the symbolic wheel Jacobian matrices in Appendix 3. We determine the nature of the mobility, actuation and sensing structures to gain insight into the *mobility characteristics* of the WMR. We compute the actuated wheel velocities from the robot velocity vector (i.e., *actuated inverse solution*) and the least-squares robot velocity vector from the sensed wheel velocities and positions (i.e., *sensed forward solution*) when the mobility analysis indicates that these solutions are applicable. We complete each example with *remarks* on its kinematic structure and its suitability for particular tasks.

7.2 Unimation Robot

7.2.1 Kinematic Description

The Unimation robot[14] illustrated in Figure 7.2.1 utilizes three symmetrically positioned omnidirectional wheels with rollers at 90°. A motor actuates each wheel and the velocity of each wheel is measured by shaft encoders. The rollers are neither actuated nor sensed. The coordinate system assignments and pertinent robot dimensions are shown in the figure.

7.2.2 Coordinate Transformation Matrices

We write the coordinate transformation matrices in Table 4.4.2 from Figure 7.2.1:

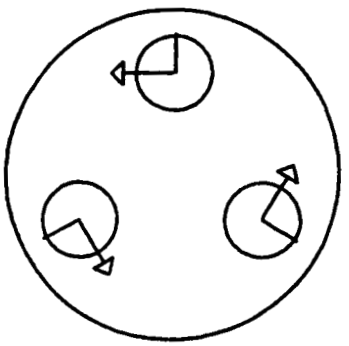
$$ {}^{R}\mathbf{T}_{H_1} = \begin{pmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & l_a \\ 0 & 0 & 1 & -l_b \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_1}\mathbf{\Phi}_{S_1} = {}^{S_1}\mathbf{T}_{C_1} = \mathbf{I} $$

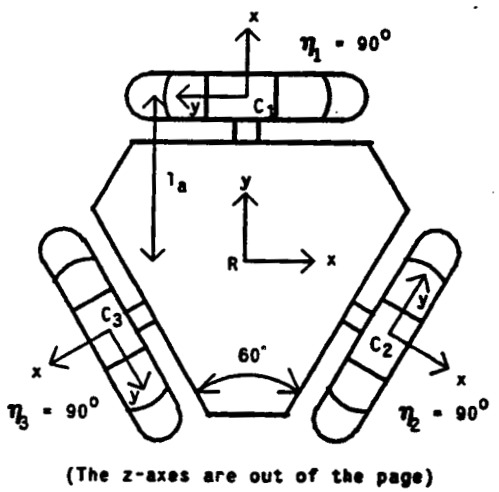
$$ {}^{R}\mathbf{T}_{H_2} = \begin{pmatrix} \sqrt{3}/2 & 1/2 & 0 & \sqrt{3}l_a/2 \\ -1/2 & \sqrt{3}/2 & 0 & -l_a/2 \\ 0 & 0 & 1 & -l_b \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_2}\mathbf{\Phi}_{S_2} = {}^{S_2}\mathbf{T}_{C_2} = \mathbf{I} $$

$$ {}^{R}\mathbf{T}_{H_3} = \begin{pmatrix} -\sqrt{3}/2 & 1/2 & 0 & -\sqrt{3}l_a/2 \\ -1/2 & -\sqrt{3}/2 & 0 & -l_a/2 \\ 0 & 0 & 1 & -l_b \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_3}\Phi_{S_3} = {}^{S_3}\mathbf{T}_{C_3} = \mathbf{I}\,. $$

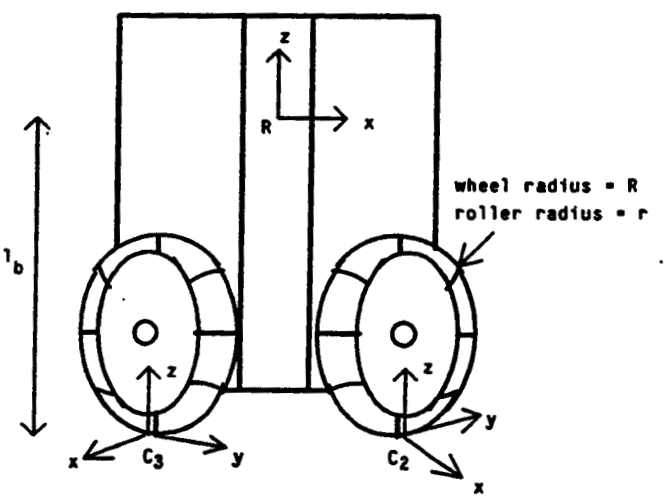
Unimation Robot

(Troas-whemor)





Top View



Side View

Figure 7.2.1

Coordinate System Assignments for the Unimation Robot

7.2.3 Wheel Jacobian Matrices

We substitute the elements of the transformation matrices, the wheel and roller radii, and the roller angles into the symbolic Jacobian matrix for omnidirectional wheels in (A3.4.2) to write the matrix wheel equations:

$$\dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} -R & 0 & l_a \\ 0 & r & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_1 x} \\ \omega_{w_1 r} \\ \omega_{w_1 z} \end{pmatrix} = \mathbf{J}_1 \dot{\mathbf{q}}_1 \tag{7.2.1}$$

$$\dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} R/2 & \sqrt{3}r/2 & -l_a/2 \\ \sqrt{3}R/2 & -r/2 & -\sqrt{3}l_a/2 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_2 z} \\ \omega_{w_2 r} \\ \omega_{w_2 z} \end{pmatrix} = \mathbf{J}_2 \dot{\mathbf{q}}_2 \tag{7.2.2}$$

$$\dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} R/2 & -\sqrt{3}r/2 & -l_a/2 \\ -\sqrt{3}R/2 & -r/2 & \sqrt{3}l_a/2 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_3 x} \\ \omega_{w_3 r} \\ \omega_{w_3 z} \end{pmatrix} = \mathbf{J}_3 \dot{\mathbf{q}}_3 \tag{7.2.3}$$

7.2.4 Mobility Characteristics

To characterize the robot mobility, we note that the soluble motion criterion is satisfied. Therefore, none of the wheels has redundant DOFs and the actuated inverse solution is applicable. Since the three DOF motion criterion is also satisfied, the Unimation robot allows 3-DOF motion.

We calculate the adequate actuation criterion $det[\mathbf{A}_a^T \mathbf{A}_a] = 27l_a^2/4$ as the first step in characterizing the actuation structure. Since the determinant is nonzero, all robot motions are producable by the motions of the actuators. The value of $\Delta(\mathbf{A}_a)\ \mathbf{B}_a$ is zero which indicates that the robust actuation criterion is also satisfied. The actuator motions are independent and no actuator conflict can occur. Since the adequate sensing criterion is satisfied but the robust sensing criterion is not, the sensing structure is adequate but not robust. Although the sensing structure allows three DOFs to be discerned by applying the sensed forward solution, wheel slip cannot be detected by the method of Section 6.5.

7.2.5 Actuated Inverse Solution

Since the soluble motion criterion is satisfied, the actuated inverse solution is computable. The actuated inverse solution in (5.5.5) applies directly:

$$\begin{pmatrix} \dot{\mathbf{q}}_{1a} \\ \dot{\mathbf{q}}_{2a} \\ \dot{\mathbf{q}}_{3a} \end{pmatrix} = \begin{pmatrix} [\mathbf{J}_{1a}^T \Delta(\mathbf{J}_{1u}) \mathbf{J}_{1a}]^{-1} \mathbf{J}_{1a}^T \Delta(\mathbf{J}_{1u}) \\ [\mathbf{J}_{2a}^T \Delta(\mathbf{J}_{2u}) \mathbf{J}_{2a}]^{-1} \mathbf{J}_{2a}^T \Delta(\mathbf{J}_{2u}) \\ [\mathbf{J}_{3a}^T \Delta(\mathbf{J}_{3u}) \mathbf{J}_{3a}]^{-1} \mathbf{J}_{3a}^T \Delta(\mathbf{J}_{3u}) \end{pmatrix} \dot{\mathbf{p}} \ ,$$

resulting in

$$\begin{pmatrix} \omega_{w_1 x} \\ \omega_{w_2 x} \\ \omega_{w_3 x} \end{pmatrix} = 1/R \begin{pmatrix} -1 & 0 & l_a \\ 1/2 & \sqrt{3}/2 & l_a \\ 1/2 & -\sqrt{3}/2 & l_a \end{pmatrix} \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} . \tag{7.2.4}$$

7.2.6 Sensed Forward Solution

Since the adequate sensing criterion is satisfied, the sensed forward solution is computable. We apply the least-squares sensed forward solution in (5.7.5):

$$\dot{\mathbf{p}} = [\Delta(\mathbf{J}_{1n}) + \Delta(\mathbf{J}_{2n}) + \Delta(\mathbf{J}_{3n})]^{-1} [\Delta(\mathbf{J}_{1n})\mathbf{J}_{1s} \quad \Delta(\mathbf{J}_{2n})\mathbf{J}_{2s} \quad \Delta(\mathbf{J}_{3n})\mathbf{J}_{3s}] \begin{pmatrix} \dot{\mathbf{q}}_{1s} \\ \dot{\mathbf{q}}_{2s} \\ \dot{\mathbf{q}}_{3s} \end{pmatrix}$$

and obtain

$$\begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = R \begin{pmatrix} -2/3 & 1/3 & 1/3 \\ 0 & 1/\sqrt{3} & -1/\sqrt{3} \\ 1/(3l_a) & 1/(3l_a) & 1/(3l_a) \end{pmatrix} \begin{pmatrix} \omega_{w_1 x} \\ \omega_{w_2 x} \\ \omega_{w_3 x} \end{pmatrix} . \tag{7.2.5}$$

7.2.7 Remarks

The Unimation robot is a general-purpose three DOF WMR. It allows three DOF motion, has adequate actuation to produce three DOF motion, and has adequate sensing to discern three DOF motion. The actuated inverse and sensed forward solutions are computable in real-time, enabling accurate closed-loop control. The low ground clearance, which only allows locomotion on smooth, level surfaces is a disadvantage of the design. The mechanical complexity of the omnidirectional wheels increases the cost and difficulty of fabrication. It is difficult to construct perfectly round omnidirectional wheels when the rollers are at 90° because of the discontinuities between rollers. An improved wheel design allowing circular omnidirectional wheel profiles has been implemented for Uranus (in Section 7.4). We have noted that the sensing structure does not allow wheel slip detection by the method of Section 6.5. Although the wheel variables which are not-sensed are difficult to instrument, an additional instrumented caster can be added to the design to provide practical robust sensing and wheel slip detection.

Three DOF locomotion is not necessary for parts and materials transport. A transport WMR may operate with two DOFs. The three DOF locomotion is advantageous when utilized with an onboard manipulator. The mobility of the WMR enhances and extends the workspace of the manipulator. Consequently, a manipulator having fewer than six DOFs mounted on the WMR

has an unlimited workspace and can accomplish the tasks of a stationary manipulator having six DOFs.

7.3 Newt

7.3.1 Kinematic Description

Newt[32] is a WMR having two diametrically opposed drive wheels and a free-wheeling castor, as shown in Figure 7.3.1. Both drive wheels are actuated and sensed, while the castor is neither actuated nor sensed.

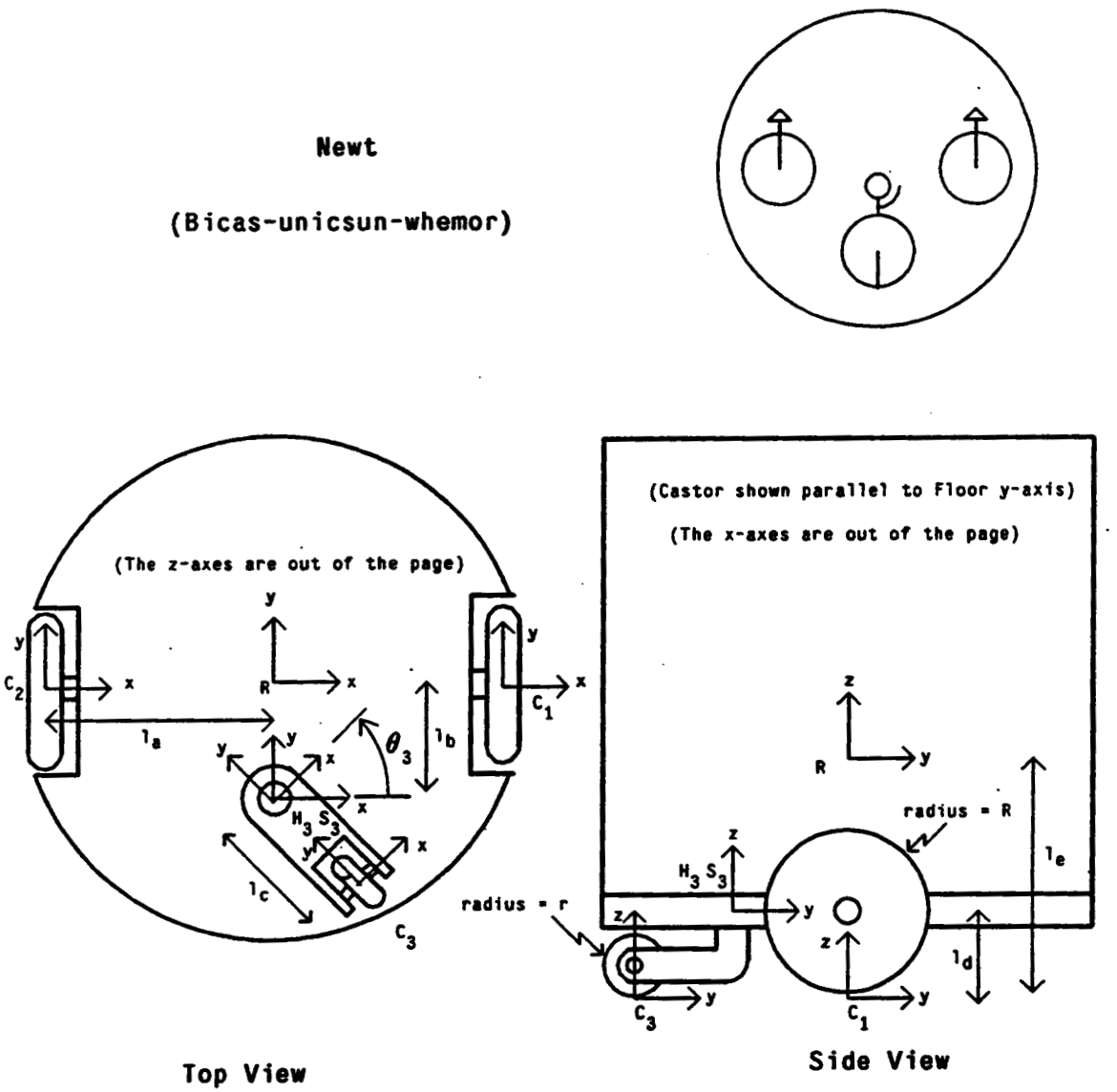


Figure 7.3.1

Coordinate System Assignments for Newt

7.3.2 Coordinate Transformation Matrices

The coordinate transformation matrices for Newt are:

$$ {}^{R}\mathbf{T}_{H_1} = \begin{pmatrix} 1 & 0 & 0 & l_a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -l_e \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_1}\Phi_{S_1} = {}^{S_1}\mathbf{T}_{C_1} = \mathbf{I} $$

$$ {}^{R}\mathbf{T}_{H_2} = \begin{pmatrix} 1 & 0 & 0 & -l_a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -l_e \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_2}\Phi_{S_2} = {}^{S_2}\mathbf{T}_{C_2} = \mathbf{I} $$

$$ {}^{R}\mathbf{T}_{H_3} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -l_b \\ 0 & 0 & 1 & -(l_e - l_d) \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_3}\Phi_{S_3} = \begin{pmatrix} \cos\theta_{S_3} & -\sin\theta_{S_3} & 0 & 0 \\ \sin\theta_{S_3} & \cos\theta_{S_3} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ {}^{S_3}\mathbf{T}_{C_3} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -l_c \\ 0 & 0 & 1 & -l_d \\ 0 & 0 & 0 & 1 \end{pmatrix} . $$

7.3.3 Wheel Jacobian Matrices

The radii of wheels one and two are identical: $R_1 = R_2 = R$, and the radius of wheel three is $R_3 = r$. By applying the Jacobian matrix for non-steered conventional wheels in (A3.2.2), we write the matrix equations for drive wheels one and two:

$$ \dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ R & -l_a \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_1 x} \\ \omega_{w_1 z} \end{pmatrix} = \mathbf{J_1}\dot{\mathbf{q}}_1 \qquad (7.3.1) $$

$$ \dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ R & l_a \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_2 x} \\ \omega_{w_2 z} \end{pmatrix} = \mathbf{J_2}\dot{\mathbf{q}}_2 \qquad (7.3.2) . $$

Similarly, by applying the Jacobian matrix for a steered conventional wheel in (A3.3.2), we

write the matrix equation for wheel three:

$$\dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_{Rz} \end{pmatrix} = \begin{pmatrix} -R\sin\theta_{S_3} & -l_c\cos\theta_{S_3} - l_b & l_b \\ R\cos\theta_{S_3} & -l_c\sin\theta_{S_3} & 0 \\ 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} \omega_{w_3x} \\ \omega_{w_3z} \\ \omega_{S_3} \end{pmatrix} = \mathbf{J_3}\ \dot{\mathbf{q}}_3 \ . \tag{7.3.3}$$

7.3.4 Mobility Characteristics

The soluble motion criterion is satisfied, indicating that the actuated inverse solution is applicable and none of the wheels is redundant. Since $w_i = 2$ for wheels one and two, the three DOF motion criterion is not satisfied. The robot has fewer than three DOFs; i.e., some robot DOFs are dependent. The matrix product $[\mathbf{\Delta}(\mathbf{B_0})\ \mathbf{A_0}]$ has rank one, and according to the expression for the number of WMR DOFs in (5.4.1), Newt has two DOFs. The kinematic motion constraints for wheels one and two simplify to $v_{R\dot{x}} = 0$. Wheel three imposes no constraints on the robot motion. The WMR thus allows independent motion in two DOFs: Y and θ.

We determine the actuation structure by first calculating the adequate actuation criterion $det[\mathbf{A}_a^T\mathbf{A}_a] = 8l_a^2$. This indicates that all robot DOFs are actuated (i.e., all robot motions in the Y and θ directions may be produced by the actuators). We find further that the robust actuation criterion $\mathbf{\Delta}(\mathbf{A}_a)\ \mathbf{B}_a = \mathbf{0}$ is satisfied. All actuator motions are independent, providing robust two DOF actuation. The sensing structure is adequate but not robust because the sensed wheel variables and the actuated ones are identical. Even though the sensing structure is not robust, the sensed forward solution is applicable.

7.3.5 Actuated Inverse Solution

Although the actuated inverse solution is applicable, only robot motions for which the translational velocity v_{Rx} is zero are possible. This means that the actuated inverse solution will be the exact solution if the X-component of the robot velocity is chosen to be zero. If the X-component of the robot velocity is non-zero, the actuated inverse solution will be computable, but it will be erroneous. The result in this case will be the optimal set of actuated wheel velocities which minimizes the least-squares error between the desired robot velocity and the resulting robot velocity. We apply the actuated inverse solution in (5.5.5):

$$\begin{pmatrix} \dot{\mathbf{q}}_{1a} \\ \dot{\mathbf{q}}_{2a} \end{pmatrix} = \begin{pmatrix} [\mathbf{J}_{1a}^T\mathbf{\Delta}(\mathbf{J}_{1u})\mathbf{J}_{1a}]^{-1}\mathbf{J}_{1a}^T\mathbf{\Delta}(\mathbf{J}_{1u}) \\ [\mathbf{J}_{2a}^T\mathbf{\Delta}(\mathbf{J}_{2u})\mathbf{J}_{2a}]^{-1}\mathbf{J}_{2a}^T\mathbf{\Delta}(\mathbf{J}_{2u}) \end{pmatrix} \dot{\mathbf{p}}$$

and obtain

$$\begin{pmatrix} \omega_{w_1 x} \\ \omega_{w_2 x} \end{pmatrix} = \frac{1}{R} \begin{pmatrix} 0 & 1 & l_a \\ 0 & 1 & -l_a \end{pmatrix} \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} . \tag{7.3.4}$$

7.3.6 Sensed Forward Solution

Since the sensing structure is adequate, the sensed forward solution in (5.7.5) is applicable:

$$\dot{\mathbf{p}} = [\Delta(\mathbf{J}_{1n}) + \Delta(\mathbf{J}_{2n}) + \Delta(\mathbf{J}_{3n}]^{-1} \quad [\Delta(\mathbf{J}_{1n})\mathbf{J}_{1s} \quad \Delta(\mathbf{J}_{2n})\mathbf{J}_{2s}] \begin{pmatrix} \dot{\mathbf{q}}_{1s} \\ \dot{\mathbf{q}}_{2s} \end{pmatrix} ,$$

and hence

$$\begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = R/(2l_a) \begin{pmatrix} 0 & 0 \\ l_a & l_a \\ 1 & -1 \end{pmatrix} \begin{pmatrix} \omega_{w_1 x} \\ \omega_{w_2 x} \end{pmatrix} . \tag{7.3.5}$$

The X-component of the robot velocity is zero independent of the sensor measurements. The Y-component of the robot velocity is proportional to the sum of the wheel velocities, and the θ-component is proportional to the difference of the wheel velocities.

7.3.7 Remarks

Newt is a general-purpose robot for tasks requiring only two-dimensional motion. Any path in a plane can be traced by a WMR possessing two DOFs. Since the vast majority of existing WMRs are applied for transporting parts, materials, and tools from one point to another along a path, Newt has wide applicability. The simple mechanical design is advantageous over omnidirectional designs because it requires fewer parts and has reduced cost. A robust sensing structure may be obtained by sensing the wheel and steering velocities of the castor. An important feature of this design is that the dead-reckoning integration calculations for the angular position of the robot are not required. If no wheel slip occurs, the angular robot position can be calculated at any time nT according to

$$^F\theta_R(nT) = \frac{R}{l_a} [\theta_{w_1 x}(nT) - \theta_{w_2 x}(nT)] + {}^F\theta_R(0) . \tag{7.3.6}$$

The computational errors due to finite precision limits and sensor noise do not accumulate in the calculation of $^F\theta_R(nT)$ as they would if the dead reckoning integration in (6.3.4) were required.

From our analysis, we conclude that Newt has two DOFs in the Y and θ directions. If the robot coordinate system is assigned at any point along the robot Y-axis except zero, the two DOFs will be X and Y. If the robot coordinate system is rotated 90°, the two DOFs will be X and θ. Finally, if the robot coordinate system is assigned to an arbitrary position not on the X or Y axes, the two DOFs cannot be specified by two of the three components X, Y, and θ. We conclude that the number of DOFs of a robot is independent of the assignment of coordinate axes, but the allowable directions of motion depend upon the placement of the robot coordinate system.

7.4 Uranus

7.4.1 Kinematic Description

Uranus[49] has the kinematic structure of the Wheelon wheelchair [2]: four omnidirectional wheels with rollers at 45° angles to the wheels. The coordinate system assignments and robot dimensions are shown in Figure 7.4.1.

7.4.2 Coordinate Transformation Matrices

Since there are no steering links, the coordinate transformation matrices for Uranus are:

$$ {}^{R}\mathbf{T}_{H_1} = \begin{pmatrix} 1 & 0 & 0 & l_a \\ 0 & 1 & 0 & l_b \\ 0 & 0 & 1 & -l_c \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_1}\Phi_{S_1} = {}^{S_1}\mathbf{T}_{C_1} = \mathbf{I} $$

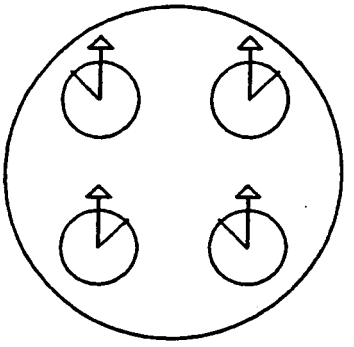
$$ {}^{R}\mathbf{T}_{H_2} = \begin{pmatrix} 1 & 0 & 0 & -l_a \\ 0 & 1 & 0 & l_b \\ 0 & 0 & 1 & -l_c \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_2}\Phi_{S_2} = {}^{S_2}\mathbf{T}_{C_2} = \mathbf{I} $$

$$ {}^{R}\mathbf{T}_{H_3} = \begin{pmatrix} 1 & 0 & 0 & -l_a \\ 0 & 1 & 0 & -l_b \\ 0 & 0 & 1 & -l_c \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_3}\Phi_{S_3} = {}^{S_3}\mathbf{T}_{C_3} = \mathbf{I} $$

$$ {}^{R}\mathbf{T}_{H_4} = \begin{pmatrix} 1 & 0 & 0 & l_a \\ 0 & 1 & 0 & -l_b \\ 0 & 0 & 1 & -l_c \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_4}\Phi_{S_4} = {}^{S_4}\mathbf{T}_{C_4} = \mathbf{I}. $$

Uranus

(Tetroas-whemor)



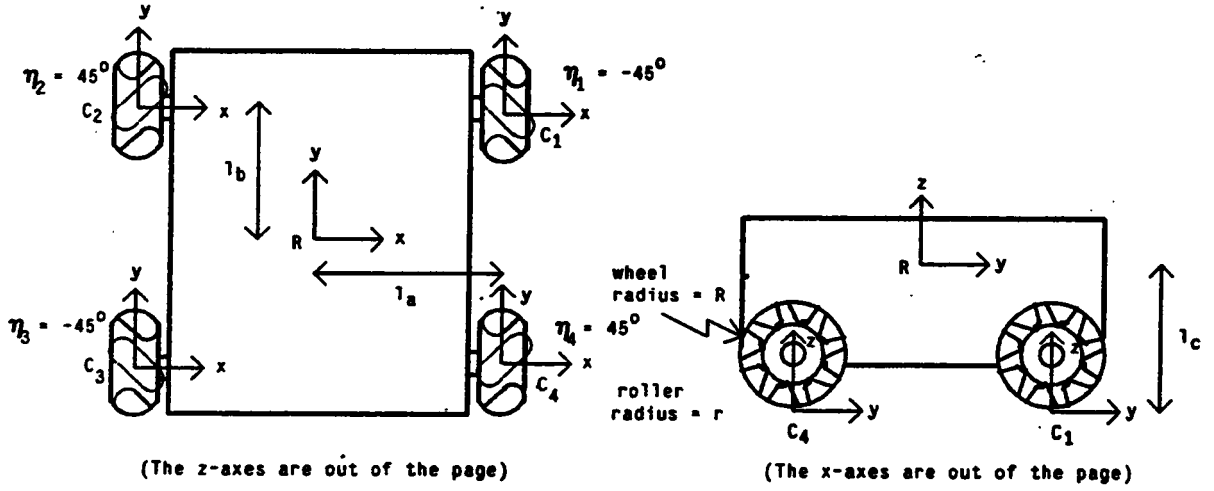


Figure 7.4.1

Coordinate System Assignments for Uranus

7.4.3 Wheel Jacobian Matrices

The radius assignments are $R_1 = R_2 = R_3 = R_4 = R$, and $r_1 = r_2 = r_3 = r_4 = r$, and the roller angles are $\eta_1 = \eta_3 = -45°$, and $\eta_2 = \eta_4 = 45°$. The Jacobian matrix for omnidirectional wheels in (A3.4.2) allows us to write the equation-of-motion for each wheel:

$$\dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} 0 & -r\sqrt{2}/2 & l_b \\ R & -r\sqrt{2}/2 & -l_a \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_1 x} \\ \omega_{w_1 r} \\ \omega_{w_1 z} \end{pmatrix} = \mathbf{J}_1 \dot{\mathbf{q}}_1 \qquad (7.4.1)$$

$$\dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} 0 & r\sqrt{2}/2 & l_b \\ R & -r\sqrt{2}/2 & l_a \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_2 x} \\ \omega_{w_2 r} \\ \omega_{w_2 z} \end{pmatrix} = \mathbf{J_2}\dot{\mathbf{q}}_2 \tag{7.4.2}$$

$$\dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} 0 & -r\sqrt{2}/2 & -l_b \\ R & -r\sqrt{2}/2 & l_a \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_3 x} \\ \omega_{w_3 r} \\ \omega_{w_3 z} \end{pmatrix} = \mathbf{J_3}\dot{\mathbf{q}}_3 \tag{7.4.3}$$

$$\dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} 0 & r\sqrt{2}/2 & -l_b \\ R & -r\sqrt{2}/2 & -l_a \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_4 x} \\ \omega_{w_4 r} \\ \omega_{w_4 z} \end{pmatrix} = \mathbf{J_4}\dot{\mathbf{q}}_4 \tag{7.4.4}$$

7.4.4 Mobility Characteristics

Since the soluble motion criterion is satisfied, the actuated inverse solution is applicable and none of the wheels has redundant DOFs. Furthermore, the three DOF criterion is satisfied and the motion structure is capable of three DOF motion.

The adequate actuation criterion yields: $det[\mathbf{A}_a^T\mathbf{A}_a] = 64(l_a + l_b)^2$. The actuators are thus able to provide motion in all three DOFs. We find that the robust actuation criterion is not satisfied. The actuation structure is thus not robust and actuator conflict may occur. The sensed and actuated wheel variables are identical so that the sensing structure is robust which allows the detection of wheel slip by the method of Section 6.5. The sensed forward solution is therefore applicable.

7.4.5 Alternative Designs

Uranus is a convenient WMR with which to develop an understanding of the differences between inadequate, adequate and robust actuation (sensing) structures, and the need for a kinematic analysis in the design of a WMR. We have shown that Uranus has an adequate but not a robust actuation structure which provides motion in all three DOFs, but allows actuator conflict. In Figure 7.4.2, we consider a slightly different WMR design.

The WMR in Figure 7.4.2 is identical to Uranus except the the wheels on the right and left hand sides of the WMR have been interchanged and the distances l_a and l_b are equal. The wheels are actuated (sensed) as with Uranus. Upon modeling this WMR and characterizing its actuation (sensing) structure, we find that it is *inadequate* (i.e., $det[\mathbf{A}_a^T\mathbf{A}_a] = \mathbf{0}$) . The problem is that the angular rotation of the WMR is not constrained by the motions of the actuators (sensors). We observe in Figure 7.4.2 that the robot can be spun about its center even if the wheel actuators are locked to one position because the rollers are free to turn.

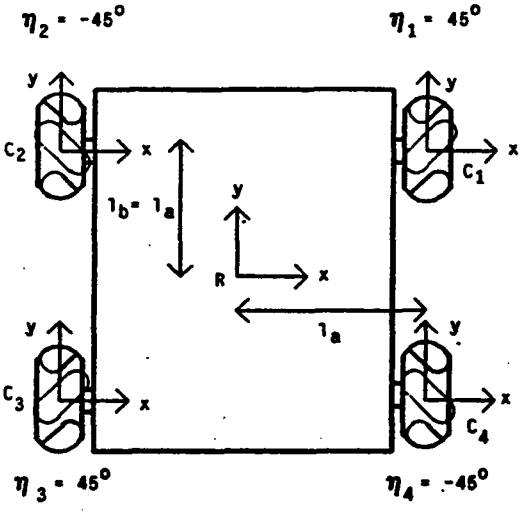


Figure 7.4.2

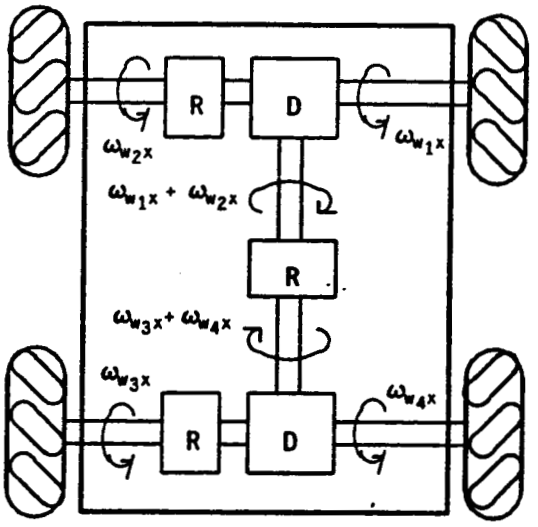
Uranus with an Inadequate Actuation Structure

We realize that the non-robust nature of Uranus' actuation structure allows actuator conflict. We now imagine how Uranus might be altered to avoid actuator conflict. Since we are interested in a practical symmetric alternative, we eliminate the possibility of simply removing one of the actuators. We must ensure that the actuator coupling criterion in (5.6.5) is satisfied. The rank one actuator coupling criterion for Uranus reduces to the scalar equation:

$$\omega_{w_1 x} + \omega_{w_2 x} - \omega_{w_3 x} - \omega_{w_4 x} = 0 . \qquad (7.4.5)$$

Only three of the four actuator motions are independent. Our solution in Figure 7.4.3a is to constrain mechanically the wheel motions with gearing between wheels to ensure that the dependencies in (7.4.5) and thus the actuator coupling criterion is satisfied.

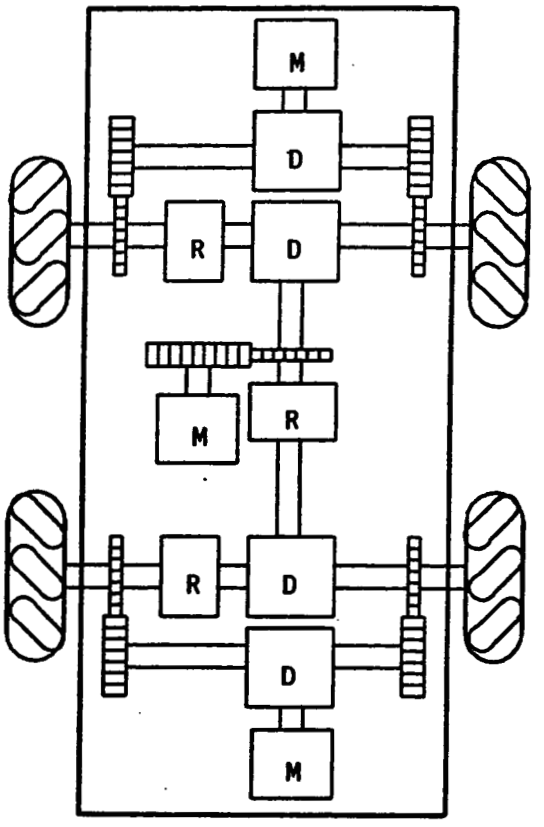
We utilize differential gearing and reversing gearing. A differential gearbox is designed so that the output shaft rotates at a rate equal to the difference of the two input shafts. A reversing gearbox is designed so that the output shaft rotates at a rate equal and opposite to the input shaft. In Figure 7.4.3b, we add three symmetrically placed motors for actuation. The actuation structure of 7.3.3b is robust. We write the composite robot equation-of-motion in terms of the ***motor shaft rotations*** (instead of the wheel axle rotations), and apply the robust actuation criterion to verify the design. Even though the complexity of this gearing may prohibit practical implementation, the procedure may be applied to the design of any WMR.



Uranus with Gearing to Insure Actuator Dependencies

D = Differential gearing
R = Reversing gearing
M = Motor

Figure 7.4.3a



Uranus with Determined Actuation

Figure 7.4.3b

Converting Uranus into a Robust Actuation Structure

7.4.6 Actuated Inverse Solution

Since the mobility structure of Uranus allows three DOFs, the actuated inverse solution in (5.5.5) is exact for all robot motions. The actuated inverse solution is:

$$\begin{pmatrix} \omega_{w_1x} \\ \omega_{w_2x} \\ \omega_{w_3x} \\ \omega_{w_4x} \end{pmatrix} = \frac{1}{R} \begin{pmatrix} -1 & 1 & l_a + l_b \\ 1 & 1 & -l_a - l_b \\ -1 & 1 & -l_a - l_b \\ 1 & 1 & l_a + l_b \end{pmatrix} \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} . \tag{7.4.6}$$

The actuated inverse solution in (7.4.6) may be obtained by assuming that all wheel variables are actuated, applying the inverse solution in (5.5.6) and extracting only the actuated wheel variables.

This alternate approach is less computationally intensive because the inverse solution for each wheel simplifies to inverting each of the Jacobian matrices.

7.4.7 Sensed Forward Solution

We apply the least-squares sensed forward solution in (5.7.5) to obtain:

$$\begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \frac{R}{4(l_a + l_b)} \begin{pmatrix} -(l_a + l_b) & (l_a + l_b) & -(l_a + l_b) & (l_a + l_b) \\ (l_a + l_b) & (l_a + l_b) & (l_a + l_b) & (l_a + l_b) \\ 1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_1 x} \\ \omega_{w_2 x} \\ \omega_{w_3 x} \\ \omega_{w_4 x} \end{pmatrix} \tag{7.4.7}$$

7.4.8 Remarks

Uranus is a general-purpose three DOF WMR, with the kinematic capabilities of the Unimation robot. The actuation structure is adequate and the sensing structure is robust as compared with Unimation's robust actuation and adequate sensing. Uranus has more ground clearance because of the arrangement of the wheels. Also, the wheel profiles are exact circles because the rollers are at 45° angles avoiding the discontinuity of wheels with 90° rollers. To utilize practically the three DOF capabilities of this robot, we envision the simultaneous operation of an onboard manipulator.

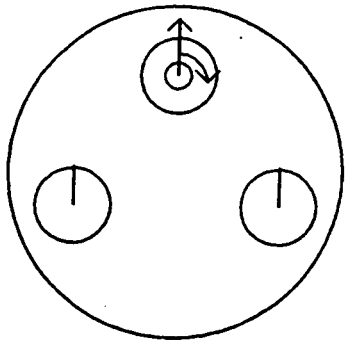
7.5 Neptune

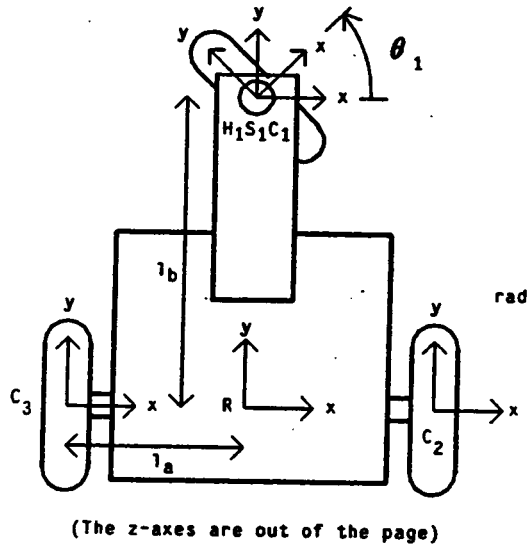
7.5.1 Kinematic Description

Neptune has a tricycle-like kinematic structure as depicted in Figure 7.5.1. The front wheel is steered about its center, and both the steering and the wheel rotations are actuated. The two fixed-orientation wheels are neither actuated nor sensed.

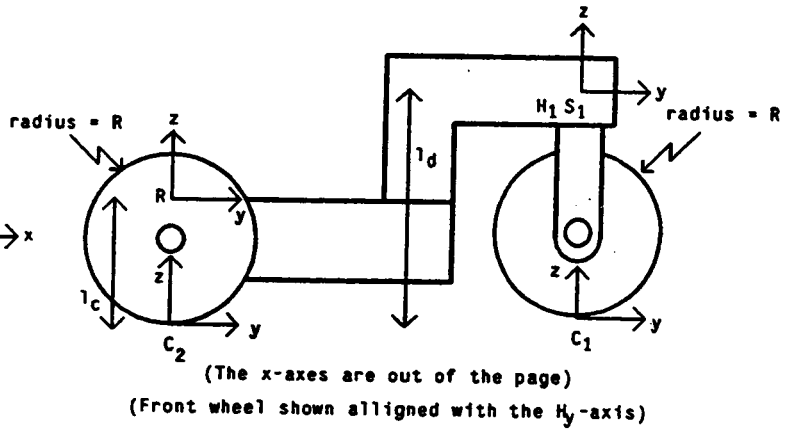
Neptune

(Bicun-unicsan-whemor)





Top View



Side View

Figure 7.5.1

Coordinate System Assignments for Neptune

7.5.2 Coordinate Transformation Matrices

The coordinate transformation matrices are:

$$ {}^{R}\mathbf{T}_{H_1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & l_b \\ 0 & 0 & 1 & l_d - l_c \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ {}^{H_1}\Phi_{S_1} = \begin{pmatrix} \cos\theta_{S_1} & -\sin\theta_{S_1} & 0 & 0 \\ \sin\theta_{S_1} & \cos\theta_{S_1} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ {}^{S_1}\mathbf{T}_{C_1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -l_d \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ {}^{R}\mathbf{T}_{H_2} = \begin{pmatrix} 1 & 0 & 0 & l_a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -l_c \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_2}\Phi_{S_2} = {}^{S_2}\mathbf{T}_{C_2} = \mathbf{I} $$

$$ {}^{R}\mathbf{T}_{H_3} = \begin{pmatrix} 1 & 0 & 0 & -l_a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -l_c \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_3}\Phi_{S_3} = {}^{S_3}\mathbf{T}_{C_3} = \mathbf{I} $$

7.5.3 Wheel Jacobian Matrices

The wheel radius assignments are $R_1 = R_2 = R_3 = R$. We use the Jacobian matrix for a steered conventional wheel in (A3.3.2) to write the equation for wheel one:

$$ \dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} -R\sin\theta_{S_1} & l_b & -l_b \\ R\cos\theta_{S_1} & 0 & 0 \\ 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} \omega_{w_1 x} \\ \omega_{w_1 z} \\ \omega_{S_1} \end{pmatrix} = \mathbf{J}_1 \dot{\mathbf{q}}_1 \tag{7.5.1} $$

The matrix equations for wheels two and three are specified by (A3.2.2):

$$ \dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ R & -l_a \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_2 x} \\ \omega_{w_2 z} \end{pmatrix} = \mathbf{J}_2 \dot{\mathbf{q}}_2 \tag{7.5.2} $$

$$ \dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ R & l_a \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_3 x} \\ \omega_{w_3 z} \end{pmatrix} = \mathbf{J}_3 \dot{\mathbf{q}}_3 \tag{7.5.3} $$

7.5.4 Mobility Characteristics

The soluble motion criterion is not satisfied because wheel one is redundant. Columns two and three of the Jacobian matrix are linearly dependent and thus the associated wheel variables (the steering velocity $\omega_{s_1 z}$ and the wheel rotational slip velocity $\omega_{w_1 z}$) are redundant. The actuated inverse solution is not applicable for Neptune. We cannot determine the actuation and sensing

structures because the foundations of the actuation and sensing characterization trees, the robot actuation and sensing equations in (5.6.3) and (5.8.3), utilize the inverse solution. Furthermore, we cannot determine the number of DOFs by applying (5.4.4) because the matrix $\Delta(\mathbf{B}_0)$ is not computable.

7.5.5 Remarks

Neptune was constructed to provide a mobile platform for vision research and for that purpose the design is sufficient. From a control engineer's point-of-view, the design is undesirable because the actuated inverse and sensed forward solutions cannot be calculated. The redundant wheel disallows these calculations. We suggest two practical design alternatives which allow the mobility and computational simplicity of Newt but require few changes to Neptune. First, wheel one can be made non-redundant by offsetting its center from the steering axis. Secondly, the front wheel can be offset as in the first alternative, and the steering and drive motors can be moved from wheel one to drive wheels two and three producing a structure kinematically identical to Newt.

7.6 Rover

7.6.1 Kinematic Description

As illustrated in Figure 7.6.1, the Rover consists of three conventional steered wheels symmetrically arranged about the center of the robot body. The steering and drive of each wheel is actuated and sensed. Actuator conflict producing shaky robot motion[50], encountered while developing a controller for Rover, fostered our modeling of WMRs.

7.6.2 Coordinate Transformation Matrices

To simplify the coordinate transformation matrices, we have assigned all hip coordinate systems parallel to the robot coordinate system and all steering coordinate systems parallel to their respective contact point coordinate systems:

$$ {}^{R}\mathbf{T}_{H_1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & l_a \\ 0 & 0 & 1 & l_d - l_c \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_1}\mathbf{\Phi}_{S_1} = \begin{pmatrix} \cos\theta_{S_1} & -\sin\theta_{S_1} & 0 & 0 \\ \sin\theta_{S_1} & \cos\theta_{S_1} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ {}^{R}\mathbf{T}_{H_2} = \begin{pmatrix} 1 & 0 & 0 & -\sqrt{3}l_a/2 \\ 0 & 1 & 0 & -l_a/2 \\ 0 & 0 & 1 & l_d - l_c \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_2}\mathbf{\Phi}_{S_2} = \begin{pmatrix} \cos\theta_{S_2} & -\sin\theta_{S_2} & 0 & 0 \\ \sin\theta_{S_2} & \cos\theta_{S_2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ {}^{R}\mathbf{T}_{H_3} = \begin{pmatrix} 1 & 0 & 0 & \sqrt{3}l_a/2 \\ 0 & 1 & 0 & -l_a/2 \\ 0 & 0 & 1 & l_d - l_c \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_3}\mathbf{\Phi}_{S_3} = \begin{pmatrix} \cos\theta_{S_3} & -\sin\theta_{S_3} & 0 & 0 \\ \sin\theta_{S_3} & \cos\theta_{S_3} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ {}^{S_1}\mathbf{T}_{C_1} = {}^{S_2}\mathbf{T}_{C_2} = {}^{S_3}\mathbf{T}_{C_3} = \begin{pmatrix} 1 & 0 & 0 & -l_b \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -l_d \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

7.6.3 Wheel Jacobian Matrices

The radius assignments are $R_1 = R_2 = R_3 = R$. The wheel equations are written by applying the Jacobian matrix for steered conventional wheels in (A3.3.2):

$$ \dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} -R\sin\theta_{S_1} & l_a - l_b\sin\theta_{S_1} & -l_a \\ R\cos\theta_{S_1} & l_b\cos\theta_{S_1} & 0 \\ 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} \omega_{w_1x} \\ \omega_{w_1z} \\ \omega_{S_1} \end{pmatrix} \tag{7.6.1} $$

$$ \dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} -R\sin\theta_{S_2} & -l_b\sin\theta_{S_2} - l_a/2 & l_a/2 \\ R\cos\theta_{S_2} & l_b\cos\theta_{S_2} + \sqrt{3}l_a/2 & -\sqrt{3}l_a/2 \\ 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} \omega_{w_2x} \\ \omega_{w_2z} \\ \omega_{S_2} \end{pmatrix} \tag{7.6.2} $$

$$ \dot{\mathbf{p}} = \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} -R\sin\theta_{S_3} & -l_b\sin\theta_{S_3} - l_a/2 & l_a/2 \\ R\cos\theta_{S_3} & l_b\cos\theta_{S_3} + \sqrt{3}l_a/2 & \sqrt{3}l_a/2 \\ 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} \omega_{w_3x} \\ \omega_{w_3z} \\ \omega_{S_3} \end{pmatrix} \tag{7.6.3} $$

7.6.4 Mobility Characteristics

The soluble motion criterion is not satisfied because the wheels are redundant. Consequently, the inverse solution is not applicable, the actuation and sensing structures cannot be determined and the sensed forward solution cannot be calculated. A dynamic force analysis is required to compute the wheel and robot motions since we cannot determine when wheel rotational slip will occur by kinematic calculations alone. Likewise, the number of DOFs cannot be determined from (5.4.4).

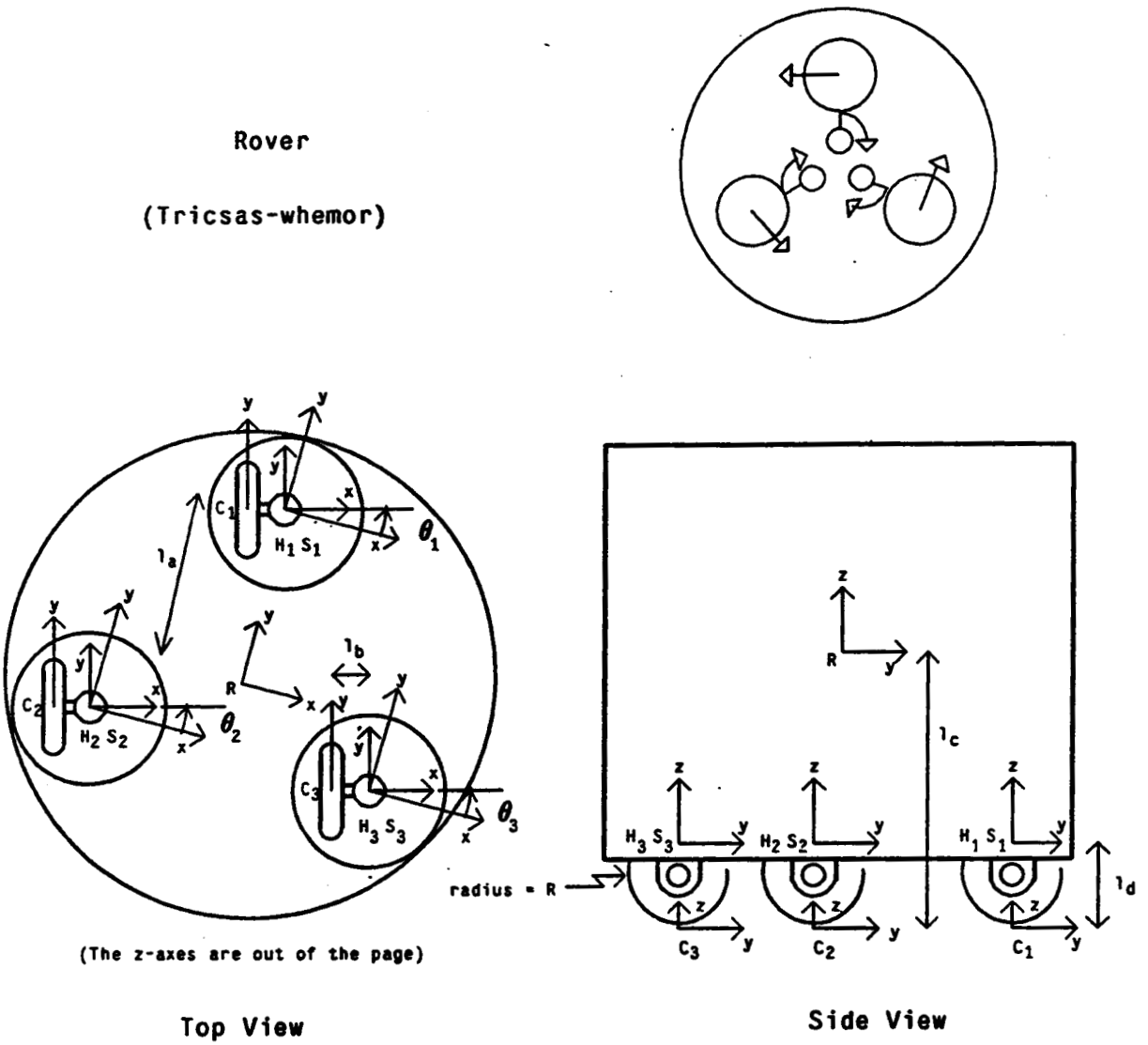


Figure 7.6.1

Coordinate System Assignments for Rover

7.6.5 Remarks

We conclude from this example that kinematic modeling of a WMR must be addressed in the design stage. Rover can be redesigned to operate as an omnidirectional WMR by constructing the steering links so that the wheels are non-redundant. Since there are six actuators, the redesigned actuation structure will not be robust and will allow actuator conflict. The Denning Sentry robot[70] replicates the kinematic structure of Rover, with the exception that all three wheels are mechanically steered and driven in unison. The Denning WMR avoids actuator conflict by utilizing only two actuators and mechanically coupling the wheel motions, but in so doing it sacrifices omnidirectionality.

7.7 Stanford Cart

7.7.1 Kinematic Description

The Stanford Cart has the kinematic structure of an automobile, two front wheels with coupled steering angles and two parallel non-steered back wheels, as shown in Figure 7.7.1. The rotations of wheels three and four and the coupled steering for wheels one and two are actuated.

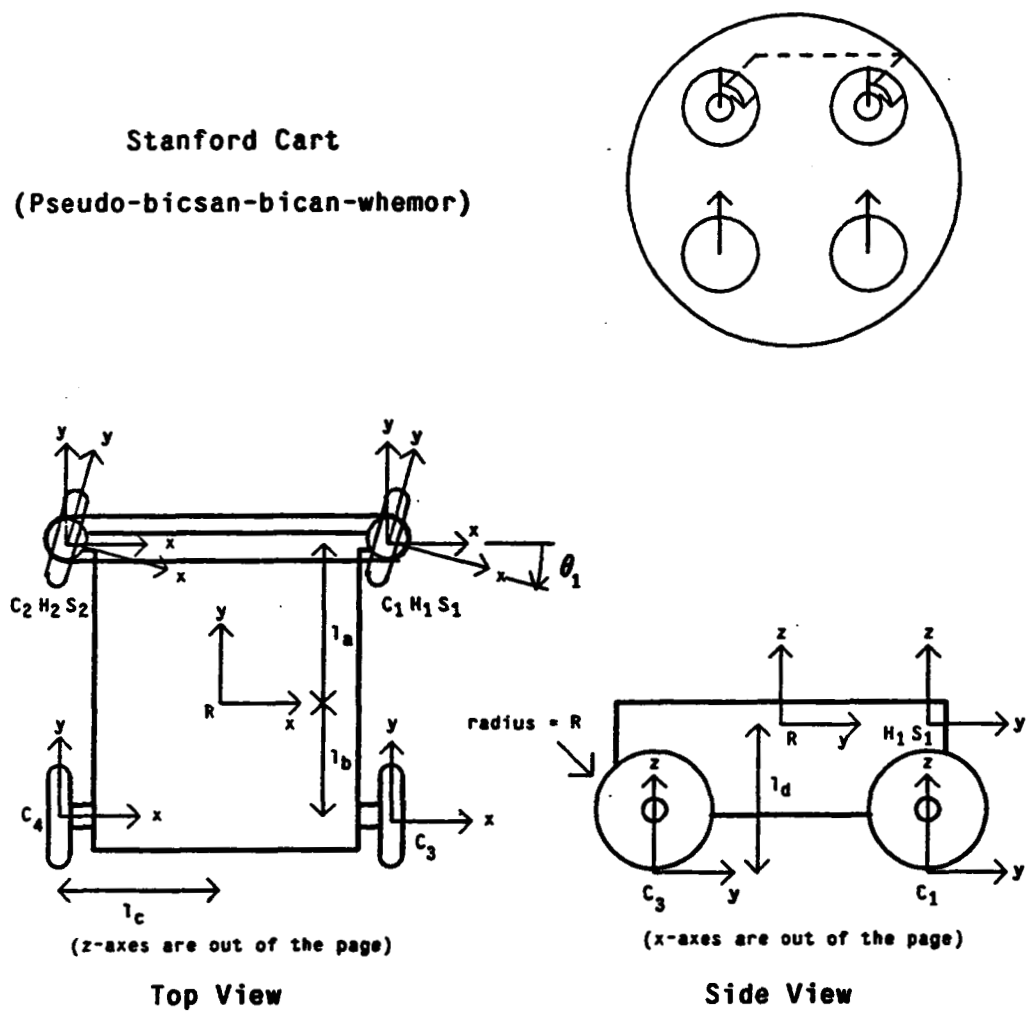


Figure 7.7.1

Coordinate System Assignements for the Stanford Cart

7.7.2 Coordinate Transformation Matrices

The coordinate systems assigned in Figure 7.7.1 lead to the following coordinate transformation matrices:

$$ {}^{R}\mathbf{T}_{H_1} = \begin{pmatrix} 1 & 0 & 0 & l_c \\ 0 & 1 & 0 & l_a \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_1}\mathbf{\Phi}_{S_1} = \begin{pmatrix} \cos\theta_{S_1} & -\sin\theta_{S_1} & 0 & 0 \\ \sin\theta_{S_1} & \cos\theta_{S_1} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ {}^{R}\mathbf{T}_{H_2} = \begin{pmatrix} 1 & 0 & 0 & -l_c \\ 0 & 1 & 0 & l_a \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_2}\mathbf{\Phi}_{S_2} = \begin{pmatrix} \cos\theta_{S_2} & -\sin\theta_{S_2} & 0 & 0 \\ \sin\theta_{S_2} & \cos\theta_{S_2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ {}^{S_1}\mathbf{T}_{C_1} = {}^{S_2}\mathbf{T}_{C_2} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -l_d \\ 0 & 0 & 0 & 1 \end{pmatrix} $$

$$ {}^{R}\mathbf{T}_{H_3} = \begin{pmatrix} 1 & 0 & 0 & l_c \\ 0 & 1 & 0 & -l_b \\ 0 & 0 & 1 & -l_d \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_3}\mathbf{\Phi}_{S_3} = {}^{S_3}\mathbf{T}_{C_3} = \mathbf{I} $$

$$ {}^{R}\mathbf{T}_{H_4} = \begin{pmatrix} 1 & 0 & 0 & -l_c \\ 0 & 1 & 0 & -l_b \\ 0 & 0 & 1 & -l_d \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad {}^{H_4}\mathbf{\Phi}_{S_4} = {}^{S_4}\mathbf{T}_{C_4} = \mathbf{I} $$

7.7.3 Wheel Jacobian Matrices

The equations-of-motion for wheels one and two are written by applying the Jacobian matrix for steered conventional wheels in (A3.3.2), and for wheels three and four by applying the Jacobian matrix for non-steered conventional wheels in (A3.2.2):

$$ \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} -R\sin\theta_{S_1} & l_a & -l_a \\ R\cos\theta_{S_1} & -l_c & l_c \\ 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} \omega_{w_1 x} \\ \omega_{w_1 z} \\ \omega_{S_1} \end{pmatrix} \tag{7.7.1} $$

$$ \begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} -R\sin\theta_{S_2} & l_a & -l_a \\ R\cos\theta_{S_2} & l_c & -l_c \\ 0 & 1 & -1 \end{pmatrix} \begin{pmatrix} \omega_{w_2 x} \\ \omega_{w_2 z} \\ \omega_{S_2} \end{pmatrix} \tag{7.7.2} $$

$$\begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} 0 & -l_b \\ R & -l_c \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_3 x} \\ \omega_{w_3 z} \end{pmatrix} \tag{7.7.3}$$

$$\begin{pmatrix} v_{Rx} \\ v_{Ry} \\ \omega_R \end{pmatrix} = \begin{pmatrix} 0 & -l_b \\ R & l_c \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_4 x} \\ \omega_{w_4 z} \end{pmatrix} \tag{7.7.4}$$

7.7.4 Mobility Characteristics

We assume[1] that the steering angles are equal; i.e., $\theta_{S_2} = \theta_{S_1} = \theta_S$, and consequently $\omega_{S_2} = \omega_{S_1} = \omega_S$. We substitute these equalities into the wheel Jacobian matrices in (7.7.1) and (7.7.2) to form the composite robot equation in (5.2.2):

$$\begin{pmatrix} -R\sin\theta_S & l_a & -l_a & 0 & 0 & 0 & 0 & 0 & 0 \\ R\cos\theta_S & -l_c & l_c & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -l_a & -R\sin\theta_S & l_a & 0 & 0 & 0 & 0 \\ 0 & 0 & -l_c & R\cos\theta_S & l_c & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -l_b & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & R & -l_c & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -l_b \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & R & l_c \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_1 x} \\ \omega_{w_1 z} \\ \omega_S \\ \omega_{w_2 x} \\ \omega_{w_2 z} \\ \omega_{w_3 x} \\ \omega_{w_3 z} \\ \omega_{w_4 x} \\ \omega_{w_4 z} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \dot{\mathbf{p}} . \tag{7.7.5}$$

Because of the coupling between wheels one and two, the applicable soluble motion criterion test is $rank[\mathbf{B}_0] = w$. We observe in (7.7.5) that the rank of the (12×9) matrix $\mathbf{B}_o$ is eight, but there are nine wheel variables (i.e., $w = 9$). Accordingly, the mobility structure of the Stanford cart is not soluble and the inverse and forward solutions are not applicable.

7.7.5 Remarks

The Stanford Cart is kinematically similar to an automobile. Even though automobiles operate satisfactorily for transportation, we cannot satisfactorily model the motion of the Stanford cart using only kinematic characteristics. We conclude that a dynamic analysis is required to model its motion.

[1] The Stanford Cart had an Ackerman steering linkage[45] between the two front wheels. The Ackerman linkage approximatly ensures the actuator coupling criterion by providing the correct wheel angles to avoid wheel slip.

7.8 Conclusions

The six examples presented in this section demonstrate that our kinematic modeling methodology in Section 4 and the solutions in Section 5 establish the foundation for developing and solving the kinematic equations-of-motion of a WMR. Furthermore, we illustrate that writing the equations-of-motion for complex kinematic structures, such as Rover, is not practical without a systematic framework. The examples show that formulating the equations-of-motion for a WMR is a straightforward procedure which does not require insight into the operation of the robot.

We note that the actuated inverse and sensed forward solutions are applicable to WMRs which satisfy the soluble motion criterion (the Unimation robot, Newt and Uranus). The WMRs which have redundant wheels (Neptune, Rover, and the Stanford Cart) do not satisfy the soluble motion criterion and the actuated inverse and sensed forward solutions are not applicable. Without these calculations, the control of WMRs having redundant wheels is inferior. We conclude that kinematic modeling of a WMR must be undertaken in the design stage (Section 6.2). Since kinematic modeling is critical for WMR control, the design of the wheels and the positioning of the wheels, actuators and sensors must ensure that all of the modeling calculations are computationally feasible.

These six examples exhibit noteworthy features. If the wheel variables which are actuated and the wheel variables which are sensed are identical, than either the actuation or the sensing structure can be robust, but not both. For example, the actuation structure of the Unimation robot is robust and the sensing structure is not; whereas, the sensing structure of Uranus is robust but the actuation structure is not. Since we desire both robust actuation and robust sensing, we should not limit our WMR designs by sensing only the wheel variables that are actuated[2]. When wheel level feedback control is implemented, the actuated wheel variables must be sensed to provide local feedback. For the preferred robot level control, we provide robust sensing and actuation. By sensing and actuating different wheel variables, we also reduce the mechanical complexity of the hardware. We note further that wheel slip is more likely to occur with actuated wheel variables than unactuated ones because the actuated variables are force/torque sources. Thus the effects of wheel slip on the calculation of the robot position from wheel sensor measurements are reduced by sensing unactuated wheel variables.

The only WMRs which allow three DOFs motion are the ones which consist exclusively of wheels with three DOFs (the Unimation Robot and Uranus). A WMR having non-steered conventional or redundant conventional wheels may be mechanically easier to construct but cannot allow three DOFs motion. We suggest that three DOF motion can be practically utilized when the WMR

[2] If brushless motors are utilised as actuators, each actuated wheel variable must be sensed to enable electronic commutation.

has an onboard manipulator. The mobility of the WMR extends the workspace of the manipulator. When the WMR is for transportation of parts, materials or tools from place to place, only two DOFs are necessary. The mechanically simplest design to provide two DOFs is two diametrically opposed non-steered conventional wheels, as on Newt. Drive motors may coupled directly to the wheel axles. One or two additional castors are needed for stability. This design also embodies simple and easily calculated sensed forward and actuated inverse solutions.

The application of our methodology to exemplary WMRs completes our study of WMR kinematics. In Section 8, we summarize our development and provide concluding remarks.

8. Conclusions

We have developed and illustrated a methodology for the kinematic modeling of WMRs. We have found that the established kinematic modeling methodology for stationary manipulators is not applicable to WMRs because of the higher pair wheel-to-floor joints, the multiple closed-link chains formed by multiple wheels, and the unactuated and unsensed wheel variables. Our development spans the kinematic analysis of WMRs, including:

- A survey of existing WMRs (in Section 2);
- Modeling of ball, omnidirectional, and conventional wheels (in Section 3);
- Assignment of coordinate systems (in Section 4.3) ;
- Formulation of the transformation matrices (in Section 4.4);
- Formulation of the kinematic equations-of-motion (in Sections 4.6, 4.7, and 4.8);
- Solutions of the kinematic equations-of-motion (in Section 5);
- Characteraization of WMR mobility (in Section 5);
- Applications to design, control, dead-reckoning, and slip detection (in Section 6);
- Kinematic modeling of six examplary WMRs (in Section 7); and
- Naming and diagramming of WMR kinematic structures (in Appendix 1).

In this concluding section, we summarize our development and highlight the significant results and implications.

We begin modeling a WMR by sketching the mechanical structure. We assign one *robot* coordinate system, and a *hip*, *steering*, and *contact* coordinate system for each wheel (in Section 4.3). We apply the Sheth-Uicker convention to coordinate system assignment and transformation matrix calculation because it allows the modeling of the higher-pair wheel contact-point motion and provides unambiguous transformation matrix labeling for the multiple closed-link chains formed by the wheels.

We model each wheel (conventional, steered-conventional, omnidirectional or ball wheel) as a planar-pair which allows three DOFs: X-translation, Y-translation, and θ-rotation. A conventional wheel attains Y-translational motion by rolling contact. The translation in the X direction and the θ rotation about the point-of-contact occur when the wheel slips. We model the rotational slip as a wheel DOF because relatively small forces are required; furthermore, the majority of all WMRs rely on this DOF. We do not consider the X-translational wheel slip a DOF because relatively large forces are necessary. Omnidirectional wheels also rely on rotational wheel slip but ball wheels do not.

By inspection of the sketch, we write the robot-to-hip, hip-to-steering and steering-to-contact

transformation matrices for each wheel in the format of Table 4.4.2. Under the assumption of no wheel slip, the wheel rotations define the motion of the wheel contact-point coordinate system with respect to a stationary coordinate system at the same position and orientation on the floor. The coordinate system fixed with respect to the floor is important because we reference the velocities of the wheel contact-point to this *instantaneously coincident coordinate system.* The rotational velocity of a wheel about its axle is thus proportional to the translational velocity of the contact point coordinate system with respect to the instantaneously coincident wheel contact-point coordinate system. Similarly, there is an instantaneously coincident robot coordinate system to reference the velocities of the robot coordinate system. We assign instantaneously coincident coordinate systems because of the higher-pair wheel contact points.

For each wheel we develop a Jacobian matrix (in Section 4.7.3) to specify the robot velocities (in the instantaneously coincident robot coordinate system: ${}^{\hat{R}}v_{Rx}$, ${}^{\hat{R}}v_{Ry}$, ${}^{\hat{R}}\omega_R$) as linear combinations of the wheel velocities (e.g., the steering velocity, the rotational velocity about the wheel axle, the rotational slip velocity, and the roller velocities for omnidirectional wheels). We write the Jacobian matrix for a wheel by substituting elements of the coordinate transformation matrices, wheel and roller radii and roller orientation angles into the symbolic Jacobian matrices of Appendix 3. For a steered wheel, the Jacobian matrix depends explicitly on the steering angle.

Our study has illuminated the following important wheel properties. A $(3 \times w_i)$ Jacobian matrix $\mathbf{J}_i$ is associated with a wheel having w_i wheel variables. If the Jacobian matrix has rank w_i, it satisfies the non-redundant wheel criterion in (4.7.15), the wheel has w_i DOFs and all wheel variables are independent. If the rank of the Jacobian matrix is less than w_i, the wheel is redundant having fewer than w_i DOFs, and some of the wheel variables are dependent. Specifically, any conventional wheel which is steered about an axis that intersects the wheel contact-point, or is oriented perpendicularly to the line from the steering axis to the contact-point, is redundant. We have noted disadvantages of redundant wheels (without wheel couplings). The actuated inverse and sensed forward solutions do not apply. We cannot characterize the actuation and sensing structure of WMRs with redundant wheels because the actuation and sensing characterization trees are developed by applying the actuated inverse solution. We also cannot determine the number of DOFs of a WMR with redundant wheels (and no wheel couplings) because the DOFs calculation in (5.4.4) is not computable. Since the actuated inverse solution is not applicable, we cannot control such a WMR by calculating the actuator velocities from the desired robot velocities. Steering the WMR by calculating the steering angle of a redundant wheel is an ad-hoc approach since a steering angle cannot be controlled instantaneously. We point-out that some existing WMRs having redundant steered-conventional wheels (e.g., Neptune and the Stanford Cart) are controlled in this manner with some success. Since our survey and examples show that WMRs have been designed

with redundant wheels, we infer that the implications of redundant wheels were not previously well-understood.

We form the *composite robot equation* (in Section 5.2) by adjoining the equations-of-motion of all of the wheels. Linear positional couplings between wheel variables (e.g., steering angles or wheel axle angles) can be incorporated into the model by making the appropriate substitutions in the composite robot equation, as demonstrated in Section 7.7.4 for the Stanford cart. We solve the composite robot equation and interpret properties of the solutions to illuminate the mobility characteristics of the robot.

The composite robot equation may have zero, one, or an infinite number of solutions corresponding to three WMR mobility characterizations: overdetermined, determined, and undetermined, respectively. The *mobility characterization tree* (in Figure 5.4.2) allows us to determine the mobility characteristics of a WMR by indicating tests to be conducted on the composite robot equation. The implications of the mobility characterization tree are summerized by the following. If the *soluble motion criterion* in (5.4.1) is satisfied, the actuated inverse solution, actuation and sensing trees and the WMR DOF calculation in (5.4.4) are applicable. The *three DOF motion criterion* in (5.4.2) indicates whether the WMR kinematic structure allows three DOF motion. If the kinematic structure does not allow three DOF motion, the *kinematic motion constraints* are computed in (5.4.3). The *number of WMR DOFs* are calculated from (5.4.4).

It is both impractical and unnecessary to actuate and sense every wheel variable on a WMR because of the multiple-closed link chains. A subset of the wheel variables is thus actuated, and a subset (not necessarily the same subset) is sensed. Even though a specific WMR may allow three DOF motion, we must be sure that the wheel actuators can actuate all three DOFs, and that the sensors can discern three DOFs. We apply the *actuation and sensing characterization trees* (in Figures 5.6.1 and 5.8.1, respectively) to provide the answers. The implications of the actuation characterization tree are summarized by the following three criteria. The *adeqauate actuation criterion* in (5.6.4) indicates whether the number and placement of the actuators is adequate for producing all motions allowed by the mobility structure. If the adequate actuation criterion is not satisfied, some robot DOFs are uncontrollable. The *robust actuation criterion* in (5.6.6) determines whether the actuation structure is robust; i.e., actuator conflict cannot occur in the presense of actuator tracking errors. If the actuation structure is adequate but not robust, some actuator motions are dependent. The *actuator coupling criterion* in (5.6.5) indicates the actuator dependencies which must be satisfied to avoid actuator conflict and forced wheel slip. The implications of the sensing characterization tree are summerized by the following three criteria. The *adequate sensing criterion* in (5.8.4) indicates whether the number and placement of the wheel sensors is adequate for discerning all robot motions allowed by the mobility structure. The *robust*

sensing criterion in (5.8.5) indicates whether the sensing structure is robust; i.e., wheel slip and sensor noise produce minimal effects on the calculation of the robot position from wheel sensor measurements. The *wheel slip criterion* in (5.8.6) provides a computational method of detecting wheel slip in robust sensing structures.

We calculate two solutions of the composite robot equation: the actuated inverse and sensed forward solutions. In the *actuated inverse solution* in (5.5.5), we calculate the actuated wheel velocities from the desired robot velocities. The actuated inverse solution is applicable for WMRs satisfying the soluble motion criterion. In the *sensed forward solution* in (5.7.5), we calculate the robot velocities from the sensed wheel velocities. The adequate sensing criterion indicates whether the forward solution is applicable for a specific WMR. The composite robot equation in (5.2.2) need not be formed, if there are no wheel couplings, because the actuated inverse and sensed forward solutions and the mobility, actuation, and sensing characterization trees are expressed in terms of the wheel Jacobian matrices. The computations required for the actuated inverse and sensed forward solutions are additions, multiplications and the solution of (at most) three linear algebraic equations.

We apply our kinematic methodology to the design, kinematics-based feedback control, dead-reckoning and wheel slip detection of WMRs. Our kinematic methodology provides valuable insights into these areas. Just as the mobility characterization tree allows us to determine the motion characteristics of an existing WMR, we may utilize the tree to design WMRs to possess such desired characteristics as two or three DOFs. We may design a WMR with any specified workspace (i.e., set of allowable motions) by proper choice and placement of the wheels. We have listed the design criteria for a robust omnidirectional WMR in Tables 6.2.1 . We model two three-DOF WMRs as examples: the Unimation robot (troas-whemor in Section 7.2) and Uranus (tetroas-whemor in Section 7.4). We suggest that three DOF WMRs are applicable for use with an on-board manipulator. The mobility of the base extends the workspace of the manipulator. The majority of practical applications (i.e., parts, tools, and materials transport) require only two DOFs. We conclude that a WMR having two diametrically opposed driven wheels (bicas-polycsun-whemor) is ideal for this application because of the simplicity of its mechanical design and kinematic model. The actuation characterization tree may be applied to design a WMR to have a robust actuation structure, thus avoiding actuator conflict, as shown for Uranus in Section 7.4.4. Similarly, the sensing characterization tree may be applied to design a WMR with a robust sensing structure to minimize the adverse effects of wheel slip on the calculation of the WMR position. We have noted that the set of actuated wheel variables and sensed wheel variables cannot coincide if both robust actuation and robust sensing are desired.

The few WMR control systems which have been documented are wheel level control sys-

tems[17,33], without using a dynamic model of the WMR. The documented designs are tailored to the specific WMR being controlled. We detailed a kinematics-based robot level control system (in Section 6.4) for WMRs for which the sensed forward and actuated inverse solutions are applicable. Dead reckoning is the real time calculation of the robot position from wheel sensor measurements. We develop a dead reckoning update calculation in Section 6.4 by integrating the robot velocity computed by the sensed forward solution.

We have uncovered three methods of dealing with wheel slip: design the actuation structure to avoid slip, design the sensing structure to detect slip, and minimize the errors in the calculated robot position due to slip. We model (in Section 3) rotational wheel slip for both conventional and omnidirectional wheels because many WMR designs rely on this DOF. We wish to avoid, detect or minimize the adverse effects of the unmodeled translational wheel slip. One approach to eliminating wheel slip is to actuate all of the wheels, such as with the four-wheel drive on an automobile. Since this can lead to actuator conflict, we must design wheel couplings to ensure that the actuator coupling criterion is satisfied, as with Uranus (in Section 7.4.4). This solution does not guarantee zero wheel slip, but if slip does occur, all wheels must slip in unison which is unlikely. We have noted that a robust sensing structure allows us to detect wheel slip. We thus design the sensing structure to satisfy the robust sensing criterion and wheel slip is detected by the method of Section 6.5. In this way, we are able to detect the onset of wheel slip and notify the robot processor that an absolute method of robot positioning (e.g., robot vision) should be applied before continuing. This method will also fail in the unlikely case that all wheels slip in unison. The least-squares sensed forward solution (in Section 5.7) is less sensitive to wheel slippage if the sensing structure is designed to be robust. If wheel slip does occur, and no absolute positioning method is available, the adverse effects can be reduced by applying the least-squares sensed forward solution.

Even though our study is tailored to WMRs, our methodology may be applied to the kinematic modeling of other mechanisms, such as legged or treaded vehicles. The analysis of mechanisms having higher-pair joints, multiple closed-link chains or unactuated and unsensed joint variables may benefit from our methodology. In particular, our matrix coordinate transformation algebra (in Section 4.5) may be applied to the transformation matrices expressing the relationships between lower and high-pair joints. Our WMR diagramming and naming conventions (in Appendix 1) may be extended to legged mobile robots (LMRs) and treaded mobile robots (TMRs).

In Section 9, we discuss our continuing research. We are extending our study of WMRs to include the ***dynamic modeling of WMRs.***

9. Continuing Research

Kinematic modeling of WMRs is the first step towards designing feedback control systems. We are continuing our study by applying our kinematic model to formulate the dynamic equations-of-motion of WMRs. In analogy with the past thirty-year study of stationary manipulators, we realize that our kinematic methodology is the foundation for the dynamic modeling of WMRs. As with stationary manipulators, our coordinate system assignments are reference systems for defining the masses and inertias of the robot components. The forces/torques produced by actuators and by motions of the robot components may be transformed from one coordinate system to another by applying our coordinate transformation matrices. Our kinematic calculations of positions, velocities and accelerations can be applied to calculate the dynamic forces and torques produced by the motion of the robot components. For example, the recursive Newton-Euler manipulator dynamics formulation[31] applies kinematics to propagate positions, velocities and accelerations from the robot base to the end-effector. The forces/torques are then calculated from the end-effector to the base.

We are applying, to the extent practicable, existing dynâmic formulations of stationary manipulators[31] to WMR dynamics modeling. We are extending the existing formulations to accommodate the special characteristics of WMRs, such as multiple closed-link chains, higher-pair wheel-to-floor joints and unactuated and unsensed wheel DOFs. Once the kinematic and dynamic models are completed, we will focus on WMR control. Our research is paralleled by the physical construction of Uranus (in Section 7.4). When we establish the foundation for WMR control, we will implement our designs on Uranus to verify the development and evaluate its performance.

We have provided an extensive methodology for kinematic modeling of WMRs, and we conclude by pointing out practical extensions to our work. We have developed the actuated inverse and sensed forward velocity solutions (i.e., the solutions for the actuated wheel velocities from the robot velocities and the robot velocities from the sensed wheel velocities). We are utilizing pulse-width modulation to control the actuators of Uranus. The actuators can be modeled by linear transfer functions from pulse-width to motor velocity[51]; the pulse-width acts as the velocity reference signal and the actuated inverse velocity solution can be applied to calculate these reference velocities. When motor control is accomplished by controlling the motor current, as is the case with many stationary manipulators, the motor torque and current are proportional. Since the motor current acts as an acceleration reference signal, the *actuated inverse acceleration solution* is required. Since there are no commercially-available rotational accelerometers, we utilize available rotational position and velocity sensors for wheel feedback. The sensed forward velocity solution is thus appropriate for computing the robot velocities for feedback control and dead reckoning. When accurate rotational accelerometers are developed, the *sensed forward acceleration solution* will be

applied.

We have advocated the application of kinematic modeling to the design of WMRs for subsequent feedback control. Since present designs are based upon experience with non-robotic mechanisms (e.g., automobiles and tricycles) and ad-hoc methods, we expect that kinematic modeling prior to construction will improve future WMR designs. In Section 6.2, we addressed the design of WMRs. *A systematic procedure for designing WMRs to obtain specified mobility characteristics* is thus a promising area for research.

Stationary manipulators are open-link chains for most operations. When the end-effecter comes in contact with an object (e.g., when picking-up an object and placing a peg in a hole), the structure becomes a closed-link chain and actuator conflict may occur. Compliance has been introduced in the operation and construction of stationary manipulators to reduce actuator conflict. Similarly, *introducing compliance in either the mechanical design or control system of a WMR to eliminate actuator conflict* in overdetermined actuation structures has practical applications.

10. References

[1] R. Adams, "Patent Probe: Omnidirectional Vehicle," *Robotics Age*, Vol. 6, No. 2, February 1984, pp. 21-22.

[2] Alvema Rehab, "Wheelon - the New Movement," (advertisement), P.O. Box 17017,S-16117 Bromma, Sweden.

[3] J. Angeles, *Spatial Kinematic Chains, Analysis, Synthesis, Optimization*, Springer-Verlag, Berlin, 1982.

[4] C. Balmer Jr., "Avatar: A Home Built Robot," *Robotics Age*, Vol. 4, No. 1, January/February 1982, pp. 20-27.

[5] J. S. Beggs, *Advanced Mechanism*, The Macmillan Company, New York NY, 1966.

[6] M.G. Bekker, *Off-The-Road Locomotion*, The University of Michigan Press, Ann Arbor MI, 1960.

[7] M. G. Bekker, *Introduction to Terrain-Vehicle Systems*, The University of Michigan Press, Ann Arbor MI, 1969.

[8] T. E. Bell, "Robots in the Home: Promises, Promises," *IEEE Spectrum*, Vol. 22, No. 5, May 1985, pp. 51-55.

[9] J. F. Blumrich, "Omnidirectional Wheel," U.S. Patent No. 3,789,947 , 1974.

[10] G. Bonmartini, "Rolling Device for Vehicles of Every Kind," U.S. Patent No. 2,751,259 , 1956.

[11] H. M. Bradbury, "Omni-Directional Transport Device," U.S. Patent No. 4,223,753 , 1980.

[12] M. Brady, et al. (editors), *Robot Motion, Planning and Control*, The MIT Press, Cambridge MA, 1982.

[13] E. B. Brown, et. al., "Gimbaled Conveyor Balls," U.S. Patent No. 4,018,322 , 1977.

[14] B. Carlisle, "Omni-Directional Mobile Robot," *Developments in Robotics*, IFS Publishing Ltd., Kempston, Bedfordshire, England, 1983.

[15] P. M. Cohn, "Linear Equations," Library of Mathematics, Routledge and Kegan Paul, London, 1958.

[16] P. Croix, *Robots*, Crown Publishers, One Park Avenue, New York NY, 10016, 1985.

[17] D. J. Daniel, "Analysis, Design, and Implementation of Microprocessor Control for a Mobile Platform," Master's Project Report, Department of Electrical and Computer Engineering, Carnegie-Mellon University, Pittsburgh PA, 15213, August 1984.

[18] J. Denavit and R. S. Hartenberg, "A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices," *Journal of Applied Mechanics*, Vol. 77, No. 2, June 1955, pp. 215-221.

[19] J. Denavit, R. S. Hartenberg, R. Razi, and J. J. Uicker, Jr., "Velocity, Acceleration, and Static-Force Analysis of Spatial Linkages," *Journal of Applied Mechanics*, Vol. 87, No. 4, December 1965, pp. 903-910.

[20] J. F. Derry, "Roving Robots," *Robotics Age*, Vol. 4, No. 5, September/October 1982, pp. 18-23.

[21] S. Dubowsky and D. T. DesForges, "The Application of Model-Referenced Adaptive Control to Robotic Manipulators," *Journal of Dynamic Systems, Measurement, and Control*, Vol. 101, No. 3, September 1979, pp.193-200.

[22] H.R. Everett, "A Second-Generation Autonomous Sentry Robot," *Robotics Age*, Vol. 7, No. 4, April 1985, pp. 29-32.

[23] M. A. Fischetti, "Robots Do the Dirty Work," *IEEE Spectrum*, Vol. 22, No. 4, April 1985, pp. 65-72.

[24] G. Giralt, R. Sobek, and R. Chatila, "A Multi-Level Planning and Navigation System for a Mobile Robot; A First Approach to Hilare," Proceedings of the IJCAI, Tokyo, Japan, August, 1979, pp. 335-338.

[25] J. Grabowiecki, "Vehicle-Wheel," U.S. Patent No. 1,305,535 , 1919.

[26] C. Helmers, "Ein Heldenleben, (Or, A Hero's Life, With Apologies to R. Strauss)," *Robotics Age*, Vol. 5, No. 2, March/April 1983, pp. 7-16.

[27] C. Helmers, "The Wandering Roboteers," *Robotics Age*, Vol. 5, No. 3, July/August 1983, pp. 6-10.

[28] C. Helmers, "Photo Essay: A First Glimpse at Gemini," *Robotics Age*, Vol. 7, No. 2, February 1985, pp. 12-13.

[29] J. M. Holland, *Basic Robotics Concepts*, Howard W. Sams & Co., Indianapolis IN, 1983, pp. 107-170.

[30] J. M. Holland, "Rethinking Robot Mobility," *Robotics Age*, Vol. 7, No. 1, January 1985, pp. 26-30.

[31] J. M. Hollerbach, "A Recursive Lagrangian Formulation of Manipulator Dynamics and a Comparative Study of Dynamics Formulation Complexity," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-10, No. 11, November 1980, pp. 730-736.

[32] R. Hollis, "Newt: A Mobile, Cognitive Robot," *Byte*, Vol. 2, No. 6, June 1977, pp. 30-45.

[33] T. Hongo, et al., "An Automatic Guidance System of a Self-Controlled Vehicle: The Command System and the Control Algorithm," IEEE Proceedings of the IECON, SanFrancisco, CA, November 1985, pp.18-22.

[34] Y. Ichikawa, N. Ozaki and K. Sadakane, "A Hybrid Locomotion Vehicle for Nuclear Power Plants," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-13, No. 6, November/December 1983, pp. 1089-1093.

[35] J. Iijima, et al., "Elementary Functions of a Self-Contained Robot 'Yamabiko 3.1'," Proceedings of the 11th Isir, 1981, pp. 211-218.

[36] B. E. Ilon, "Wheels for a Course Stable Selfpropelling Vehicle Movable in any Desired Direction on the Ground or Some Other Base," U.S. Patent No. 3,876,255 , 1975.

[37] T. Iwamoto, H. Yamamoto, and K. Honma, "Transformable Crawler Mechanism with Adaptability to Terrain Variations," International Conference on Advanced Robotics, Tokyo Japan, September, 1983.

[38] R. Johnson, "Part of the Beginning," *Robotics Age*, Vol. 6, No. 8, August 1984, pp. 35-37.

[39] C. Y. Jones, "Global Wheel", U.S. Patent No. 2,448,222 , 1948.

[40] Y. Kanayama, et al., "A Mobile Robot with Sonic Sensors and its Understanding of a Simple World," Institute of Information Sciences and Electronics, University of Tsukuba, Japan, Techical Report No. ISE-TR-81-22, February 1981.

[41] R. A. Lewis and A. K. Bejczy, "Planning Considerations for a Roving Robot with Arm," Proceedings of the Third IJCAI, Stanford CA, August 1973, pp.308-316.

[42] J. Y. S. Luh, M. W. Walker, and R. P. C. Paul, "Resolved-Acceleration Control of Mechanical Manipulators," *IEEE Transactions on Automatic Control*, Vol. AC-25, No. 3, June 1980, pp. 468-474.

[43] J. Y. S. Luh, "An Anatomy of Industrial Robots and Their Controls", *IEEE Transactions on Automatic Control*, Vol. AC-28, No. 2, February 1983, pp.133-153.

[44] T. Marrs, *The Personal Robot Book*, Tab Books Inc., Blue Ridge Summit PA, 17214, 1985.

[45] *McGraw-Hill Encyclopedia of Science & Technology*, 5th Edition, McGraw-Hill Book Company, New York, Vol. 1, p. 56, 1982.

[46] H. P. Moravec, "Obstacle Avoidance and Navigation in the Real World by a Seeing Robot Rover", PhD Thesis, Department of Computer Science, Stanford University, 1980.

[47] H. P. Moravec, Personal Communication, Carnegie-Mellon University, Pittsburgh PA, January, 1982.

[48] H. P. Moravec, "The Stanford Cart and the CMU Rover," *Proceedings of the IEEE*, Vol. 71, No. 7, July 1983. pp. 872-884.

[49] H. P. Moravec (editor), "Autonomous Mobile Robots Annual Report - 1985," Robotics Institute Technical Report No. CMU-RI-MRL 86-1, Carnegie-Mellon University, Pittsburgh PA, January 1986.

[50] P. F. Muir, "Digital Servo Controller Design For Brushless DC Motors," Master's Project Report, Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh PA, 15213, April 1984.

[51] P. F. Muir and C. P. Neuman, "Pulsewidth Modulation Control of Brushless DC Motors for Robotic Applications," *IEEE Transactions on Industrial Electronics*, Vol. IE-32, No. 3, August 1985, pp. 222-229.

[52] N. J. Nilsson, "Shakey the Robot," Technical Note 323, Artificial Intelligence Center, Computer Science and Technology Division, SRI International, Menlo Park CA, April, 1984.

[53] D. E. Orin and S. Y. Oh, "Control of Force Distribution in Robotic Mechanisms Containing

Closed Kinematic Chains," *Journal of Dynamic Systems, Measurement, and Control*, Vol. 103, No. 2, June 1981, pp. 134-141.

[54] R. P. Paul, *Robot Manipulators: Mathematics, Programming and Control*, The MIT Press, Cambridge MA, 1981.

[55] R. Paul, J. Luh, et al., "Advanced Industrial Robot Control Systems," Technical Report No. TR-EE79-35, School of Electrical Engineering, Purdue University, West Lafayette IN, 47907, July 1979.

[56] D. L. Pieper, "The Kinematics of Manipulators under Computer Control," Stanford Artificial Intelligence Report, Memo No. AI-72, Computer Science Department, Stanford University, Stanford CA, October 1968.

[57] G. Podnar, K. Dowling and M. Blackwell, "A Functional Vehicle for Autonomous Mobile Robot Research", Robotics Institute Technical Report No. CMU-RI-TR-84-28, Carnegie-Mellon University, Pittsburgh PA, April, 1984.

[58] M. Raibert, et. al., "Dynamically Stable Legged Locomotion," Robotics Institute Technical Report No. CMU-RI-TR-83-20, Carnegie-Mellon University, Pittsburgh, PA., December, 1983.

[59] M. Rogers, "Birth of the Killer Robots," *Newsweek*, June 25, 1984, p. 51.

[60] B. J. Schachter, et al., "Robot Vehicles: A Survey and Proposed Test-Bed Facility," DARPA, Proceedings: Image Understanding Workshop, 1983, pp. 163-174.

[61] P. N. Sheth and J. J. Uicker, Jr., "A Generalized Symbolic Notation for Mechanisms", *Journal of Engineering for Industry*, Series B, Vol. 93, No. 70-Mech-19, February 1971, pp. 102-112.

[62] J. E. Shigley, *Kinematic Analysis of Mechanisms*, McGraw-Hill, New York, 1969.

[63] J. E. Shigley and J. J. Uicker, Jr., *Theory of Machines and Mechanisms*, McGraw-Hill Book Company, New York, 1980.

[64] M. H. Smith and L.S. Coles, "Design of a Low Cost, General Purpose Robot," Proceedings of the Third IJCAI, Stanford CA, August, 1973, pp. 324-335.

[65] V. D. Tourassis and C. P. Neuman, "Robust Nonlinear Feedback Control for Robotic Manipulators," *IEE Proceedings - D: Control Theory and Applications*, Special Issue on Robotics, Vol. 132, No. 4, July 1985, pp. 134-143.

[66] R. Wallace, et. al., "First Results in Robot Road-Following," Proceedings of the IJCAI, Los Angeles CA, August, 1985, pp. 1089-1095.

[67] L. E. Whitaker, "Stair Climbing Device," U.S. Patent No. 3,058,754 , 1962.

[68] D. E. Whitney, "Resolved Motion Rate Control of Manipulators and Human Protheses," *IEEE Transactions on Man-Machine Systems*, Vol. MMS-10, No. 2, June 1969, pp.47-53.

[69] J. P Wier and R. A. Garrett, "Ambulator Drive Mechanism," U.S. Patent No. 4,258,815 , 1981.

[70] E. Wilson, "Denning Mobile Robotics: Robots Guard the Pen," *High Technology*, Vol. 5, No. 6, June 1985.

1. Appendix 1: A Nomenclature and Symbolic Representation of WMRs

1.1 Introduction

In this appendix, we introduce a nomenclature and a symbolic representation for describing the essential kinematic structure of WMRs. We define *essential kinematic information* as the minimal information required to solve symbolically the kinematic equations-of-motion. For example, the presense of a steering link is considered essential kinematic information because an equation which relates the velocity of a steered wheel to the velocity of the robot body must depend upon the steering angle. In contrast, the distance between two wheels is not essential kinematic information because knowing the numerical value of the distance does not help to formulate the symbolic equations-of-motion. The nomenclature provides a convenient literal and verbal representation of the essential kinematic information. The symbolic representation displays pictorially the essential kinematic relations between the robot body, wheels and steering links using mnemonic symbols. Our desire to compare the kinematic characteristics of WMRs of differing structures has led to these representations. Without simple, straightforward and informative descriptions of the kinematic structure of a WMR, comparisons between robots become confusing and awkward. The conventional pictorial representations are mechanical drawings in which characteristics unessential for kinematic analysis complicate understanding. Similarly, the conventional literal descriptions of WMR kinematics are through lengthy verbal explanations. Our symbolic and literal representations of WMRs characterize the essential kinematic structure of a WMR through simple diagrams or names.

Our symbolic (naming) representation has been devised to be easily drawn (written or spoken) and interpreted, while providing the following information:

- The number of wheels;
- The type of each wheel;
- The steered wheels;
- The relative positioning of the wheels;
- The actuated DOFs of each wheel; and
- The sensed DOFs of each wheel.

Our symbolic representation can be augmented to include functional dependencies between wheels and define the distances and angles between components (although these characteristics are not considered essential kinematic information). Although functional dependencies are needed for symbolic solutions, it is difficult to incorporate arbitrary functional relations into our representations. Our definition of essential kinematic information is chosen because our ultimate objective is the control of WMRs; consequently, information required for the forward and inverse kinematic

calculations is directly applicable to WMR control. For this reason, we specify the DOFs of each wheel which are actuated and sensed. The motion of an unactuated (non-sensed) DOF may constrain the motion of the robot, whereas the motion of an actuated (sensed) DOF may be calculated symbolically from the motion of the robot body. **Understanding these representations can most easily be accomplished by scanning the rules delineated in Sections A1.2 and A1.3 and then following the examples in Section A1.4.** The reader can then refer back to the rules for a more detailed understanding.

1.2 Symbolic Representation Rules

The rules for generating and interpreting WMR diagrams follow.

1.) A WMR is depicted by a large circle.

2.) Each wheel appears as a small circle within the WMR circle.

3.) Each steering axis is portrayed as circle smaller than the associated wheel; a steering link is drawn as a line segment from the steering axis to the respective wheel. If the steering axis intersects the center of the respective wheel, it is depicted as a small circle within and concentric to the wheel circle, and a steering link is not required.

4.) The relative positions of the wheel circles (for non-steered wheels) and steering axes (for steered wheels) correspond to the relative positions of the wheels and steering axes on the robot.

5.) The DOFs of a wheel are indicated by line segments and arcs within the wheel circle drawn in the directions of the translational and rotational DOFs, respectively. The rotational slip DOF of a wheel is implied and no arc is drawn. A conventional wheel has one radial line segment in the direction of travel from the wheel center to the wheel circle. Similarly, an omnidirectional wheel has two radial line segments, and a ball wheel has two radial line segments and an arc (one quarter of a circle) drawn within the wheel circle.

6.) The actuated DOFs of each wheel are drawn with an arrowhead appended to the line indicating the DOF.

7.) The sensed DOFs of each wheel are drawn with a "T" appended to the line indicting the DOF. A DOF, which is both actuated and sensed, is indicated by a closed arrow (i.e., the combination of a "T" and an arrow).

*8.) (Optional) Functional dependencies between DOFs within or between wheels may be indicated by dashed lines. Dashed lines may also be used to indicate that a component of a WMR cannot be described adequately by our representation.

1.3 Nomenclature Rules

Our nomenclature expresses the identical information as the symbolic representation in Section A1.2. For compactness, we limit the amount of positional, actuation and sensing information in the name of the WMR. The rules for creating and interpreting WMR names follow.

1.) The name of the kinematic structure of a wheeled mobile robot ends with the suffix *-whemor*. This suffix may be omitted when it is understood that the name is of a WMR.

2.) *Sets* of one or more wheels of the same functional type are indicated by syllables separated by hyphens.

3.) Two or more wheels of a WMR are of the same functional type if they are of the same basic type (i.e., conventional, omnidirectional, or ball); are all steered or all not-steered; are all actuated and sensed similarly; and are all placed symmetrically with respect to either the center of the robot, a line through the robot center (the major axis), or a line perpendicular to the major axis (the minor axis).

4.) The syllables are ordered from the beginning to the end of the name according to the following precedence characteristics which are listed from the most to the least important:

Symmetry with respect to the robot center;
Symmetry with respect to the major axis;
Symmetry with respect to the minor axis;
Number of wheels;
Steered wheels;
Ball wheels;
Omnidirectional wheels;
Conventional wheels;
Actuated wheels; and
Sensed wheels.

For example, all wheel sets which are symmetric with respect to the robot center appear first; and if there is more than one wheel set which is symmetric with respect to the robot center, the set having the largest number of wheels (if there is not a tie) is listed first in the name.

5.) Each syllable representing a set of wheels consists of:

i.) One of the prefixes "*uni*", "*bi*", "*tri*", "*tetra*", "*penta*", "*hexa*", "*hepta*", "*octa*", "*ennea*", "*deca*", or "*poly*" to indicate the number of wheels in the set;

ii.) Followed by one of the letters "*c*", "*o*", "*b*", or "*w*" to indicate that they are either conventional, omnidirectional, ball or an unspecified type of wheel. For an omnidirectional wheel, the final vowel of the prefix is dropped before adding "*o*" to make the name pronounceable;

iii.) Followed by "*s*", if the wheels are steered;

iv.) Followed by either an "*a*" or "*u*" to indicate that the wheels are actuated or unactuated, respectively. A wheel having more than one DOF and/or a steering axis is considered actuated if the steering angle or any of the DOFs is actuated;

v.) Followed by either an "*s*" or "*n*" to indicate that the wheels are sensed or not-sensed, respectively. A wheel having more than one DOF and/or a steering axis is considered sensed if the steering angle or any of the DOFs is sensed.

6.) A kinematic structure of a WMR, which cannot be named adequately according to these rules, is named by prefixing the name which most closely indicates the structure with *pseudo-*.

A class of kinematic structures which may consist of a large number of specific instances of WMRs is specified by the *poly* prefix. For example, a *polycas-whemor* refers to the class of WMRs which have only conventional non-steered wheels arranged symmetrically with respect to the robot center or its major axis. Similarly a class of WMRs which has a number of wheels whose type is not specified is called *polywas-whemor*. Also, if the actuation and sensing characteristics are not important for the discussion, the actuation and sensing labels may be omitted, as in *polyw-whemor*. Admittedly, our nomenclature has disadvantages. Names created by these rules may not be easily pronounceable. There is not a one-to-one relationship between WMRs and the names created by our nomenclature. There are examples of WMRs which have several legal names (e.g., wheel sets can always be divided into multiple sets, each having fewer wheels). Furthermore, it is not

always possible to determine the symmetry of a WMR from its name (e.g., a *hexac-whemor* may be symmetric with respect to the robot center or the major axis). These disadvantages are the result of our attempt to assign compact names. Most ambiguities in the nomenclature can be eliminated by assuming the practical alternative. For example, a *tric-whemor* must be symmetric with respect to the robot center and not the major axis, because it would be more practical to name the latter a *bic-unic-whemor*.

1.4 Examples

In Section 2, we illustrate the kinematic diagram and name of fourteen WMRs. The predominant WMR kinematic structure documented in the literature has two parallel conventional wheels, one on each side of the robot (thus, the syllable *bicas*). These robots also possess one or two castors for stability. Among the most widely known examples are Shakey[52] and Newt[32] (in Figure 2.1). Shakey has two free-wheeling casters for stability (*bicsun*), whereas Newt utilizes only one (*unicsun*). By mounting the two driven wheels at an acute angle to the floor in their Topo[27] robot (in Figure 2.1), the Androbot Company stabilized the robot without the use of castors. Even though the acute angle of the wheels cannot be represented in either the symbolic representation or the name, we can infer that the wheels must be angled for stability by assuming the most practical realization. Mobile robots which possess multiple non-steered, driven wheels whose axes are non-colinear must rely on wheel slip if the robot is to navigate turns. Such is the case with the Terregator[66] (in Figure 2.2) which uses six parallel, non-steered, conventional wheels, three on either side (*hexacas*). The mechanically more complex, steered and driven conventional wheel is utilized on Neptune[57] (in Figure 2.3), which has a tricycle wheel arrangement; the front wheel is steered and driven (*unicsas*), while the two rear wheels are at a fixed parallel orientation and are undriven (*bicun*). The CMU Rover[48] (in Figure 2.3) has three steered and driven wheels (*tricsas*).

The Stanford Cart[46] (in Figure 2.4) has two steered, undriven wheels in the front (*bicsan*) and two fixed, driven wheels in the back (*bican*). The two steered wheels are coupled so as to be oriented in the same direction, thus the *pseudo* prefix. The JPL Rover[41] (in Figure 2.4) is similar to the Stanford Cart except that both the front and back wheel pairs have coupled steering *pseudo-bicsan-bicsan-whemor*. Kludge[29] (in Figure 2.4) has complex functional dependencies between the wheels. This robot has three conventional wheels that are both steered and driven, as on the CMU Rover. In addition, a chain and gear arrangement is used to equalize all drive velocities and steering angles. To complicate further the arrangement, each wheel is mounted on an actuated link which can be pivoted towards or away from the center of the robot to adjust its stability properties (*pseudo-tricsas*). Dashed lines are used in the symbolic representation of Kludge to indicate the functional dependencies between steering angles and wheel actuation, and the inability to represent the pivoted link. The hybrid spider drive[29] (in Figure 2.5) utilizes four conventional wheels, two

on either side of the robot, each of which is mounted at the end of a three DOF leg linkage (*pseudo-tetracsas*). The hybrid locomotion vehicle[34] (in Figure 2.5) utilizes six steered and driven conventional wheels, each at the end of an actuated vertical leg (*pseudo-hexacsas*). Uranus[49] (in Figure 2.6) utilizes four omnidirectional wheels positioned at the corners of a rectangle (*tetroas*). The Unimation Robot[14] (in Figure 2.6) possesses three DOFs using only three actuators and three omnidirectional wheels (troas). The most maneuverable wheel is a ball which is actuated so as to possess three DOFs[47] (*unibas*).

We note that our representations can be extended to other classes of mobile robots. For example, Legged Mobile Robots (LMRs) can be denoted by the suffix *lemor*, and Treaded Mobile Robots (TMRs) may be denoted by the suffix *tremor*.

2. Appendix 2: Symbol Tables

Scalars		
Scalar	Page	Definition
ω_{wx}	12	angular velocity of the wheel about the x-axis through its center
ω_{wy}	12	angular velocity of the wheel about the y-axis through its center
ω_{wz}	12	angular velocity of the wheel about the z-axis through its center
ω_{sz}	12	angular veocity of the steering link about its axis
ω_{wr}	12	angular velocity of rollers about their axes
v_x	12	linear velocity along the x-axis
v_y	12	linear velocity along the y-axis
ω_z	12	angular velocity about the z-axis
R	12	wheel radius
r	12	roller radius
η	12	roller angle
N	17	number of wheels
i	36	wheel index
w_i	36	number of wheel variables of wheel i
a_i	52	number of actuated wheel variables of wheel i
u_i	52	number of unactuated wheel variables of wheel i
s_i	57	number of sensed wheel variables of wheel i
n_i	57	number of not-sensed wheel variables of wheel i
w	43	total number of wheel variables
a	52	total number of actuated wheel variables
u	52	total number of unactuated wheel variables
s	58	total number of sensed wheel variables
n	58	total number of not-sensed wheel variables
t	67	continuous time variable
T	67	sampling period
n	67	discrete time index

Vectors			
Vector	Page	Dimension	Definition
$\mathbf{p}$	36	(3×1)	robot position vector
$\mathbf{p}_d$	68	(3×1)	desired robot position vector
$\mathbf{p}_n$	58	$(\{3+n\} \times 1)$	combind robot and not-sensed wheel position vector
$\mathbf{q}$	43	$(w \times 1)$	wheel position vector
$\mathbf{q}_a$	53	$(a \times 1)$	actuated wheel position vector
$\mathbf{q}_u$	54	$(u \times 1)$	unactuated wheel position vector
$\mathbf{q}_s$	57	$(s \times 1)$	sensed wheel position vector
$\mathbf{q}_i$	37	$(w_i \times 1)$	physical wheel position vector of wheel i
$\hat{\mathbf{q}}_i$	36	(4×1)	pseudo-wheel position vector of wheel i
$\mathbf{q}_{ia}$	52	$(a_i \times 1)$	actuated position vector of wheel i
$\mathbf{q}_{iu}$	52	$(u_i \times 1)$	unactuated position vector of wheel i
$\mathbf{q}_{is}$	58	$(s_i \times 1)$	sensed position vector of wheel i
$\mathbf{q}_{in}$	58	$(n_i \times 1)$	not-sensed position vector of wheel i
$\mathbf{q}_p$	53	$(w \times 1)$	partitioned wheel position vector
$\delta\mathbf{p}_R$	68	(3×1)	differential robot displacement vector
$\delta\mathbf{q}_a$	68	$(a \times 1)$	differential actuator displacement vector
$\mathbf{e}$	72	$(3N \times 1)$	least-squares error vector
$\mathbf{e}_R$	68	(3×1)	robot position error vector
$\mathbf{k}_f$	68	(3×1)	control system feedforward gain vector
$\mathbf{k}_a$	68	$(a \times 1)$	actuator gain vector
$\mathbf{k}_s$	68	$(s_i \times 1)$	sensor gain vector

Matrices			
Matrix	Page	Dimension	Definition
$\hat{\mathbf{J}}_i$	36	(3×4)	pseudo-Jacobian matrix of wheel i
$\mathbf{J}_i$	37	$(3 \times w_i)$	Jacobian matrix of wheel i
$\mathbf{J}_{ia}$	52	$(3 \times a_i)$	actuated Jacobian matrix of wheel i
$\mathbf{J}_{iu}$	52	$(3 \times u_i)$	unactuated Jacobian matrix of wheel i
$\mathbf{J}_{is}$	57	$(3 \times s_i)$	sensed Jacobian matrix of wheel i
$\mathbf{J}_{in}$	57	$(3 \times n_i)$	not-sensed Jacobian matrix of wheel i
$\mathbf{U}$	44	$(c \times d)$	an arbitrary matrix
$\Delta(\mathbf{U})$	44	$(c \times c)$	delta function
$\mathbf{J}_a^{-1}$	53	$(a \times 3)$	actuated inverse Jacobian matrix
$\mathbf{J}_s$	58	$(3 \times s)$	sensed forward Jacobian matrix
$\mathbf{W}_i$	37	$(4 \times w_i)$	wheel matrix of wheel i
$\mathbf{V}$	38	(3×3)	motion matrix
$\mathbf{A}_0$	43	$(3N \times 3)$	lefthand side of composite robot equation
$\mathbf{B}_0$	43	$(3N \times w)$	righthand side of composite robot equation
$\mathbf{B}_{0p}$	53	$(3N \times w)$	righthand side of partitioned composite robot equation
$\mathbf{A}_a$	54	$(3N \times 3)$	lefthand side of robot actuation equation
$\mathbf{B}_a$	54	$(3N \times a)$	righthand side of robot actuation equation
$\mathbf{A}_n$	58	$(3N \times \{3+n\})$	lefthand side of the partitioned robot sensing equation
$\mathbf{B}_s$	58	$(3N \times s)$	righthand side of robot sensing equation
$\mathbf{A}_s$	60	$(3N \times 3)$	lefthand side of robot sensing equation

3. Appendix 3: Wheel Jacobian Matrices

3.1 Introduction

In this appendix, we develop the wheel Jacobian matrices for conventional wheels, steered conventional wheels, omnidirectional wheels and ball wheels. The wheel Jacobian matrix (as introduced in Section 4.7.3) relates the velocities of the WMR to the velocities of the wheel. The wheel Jacobian matrix is the product of the pseudo-Jacobian matrix $\hat{\mathbf{J}}_i$ and the wheel matrix $\mathbf{W}_i$:

$$\mathbf{J}_i = \hat{\mathbf{J}}_i \mathbf{W}_i . \qquad (A3.1.1)$$

The pseudo-Jacobian matrix relates the wheel pseudo-velocities to the robot velocities, as described in Section 4.7:

$$\hat{\mathbf{J}}_i = \begin{pmatrix} \cos {}^R\theta_{C_i} & -\sin {}^R\theta_{C_i} & {}^Rd_{C_iy} & -{}^Rd_{H_iy} \\ \sin {}^R\theta_{C_i} & \cos {}^R\theta_{C_i} & -{}^Rd_{C_ix} & {}^Rd_{H_iz} \\ 0 & 0 & 1 & -1 \end{pmatrix} . \qquad (A3.1.2)$$

The wheel matrix in (4.7.13) relates the pseudo-velocities to the actual wheel velocities. The wheel equations-of-motion in Figure 3.2 are applied to construct the wheel matrices. The pseudo-velocities ${}^{\overline{C_i}}v_{C_ix}$, ${}^{\overline{C_i}}v_{C_iy}$ and ${}^{\overline{C_i}}\omega_{C_i}$ are the velocities v_x, v_y, and ω_z in Figure 3.1. The actual wheel velocities are the angular velocities of the wheel and rollers ω_{w_ix}, ω_{w_iy}, ω_{w_iz}, and ω_{w_ir} about their respective axes. With these observations, the wheel matrix for each wheel is written directly from the wheel equations-of-motion in Figure 3.2. The wheel Jacobian matrix is then formed by multiplying the pseudo-Jacobian matrix in (A3.1.2) by the wheel matrix. We consider each of the aforementioned wheels in turn.

3.2 Conventional Non-Steered Wheel

The conventional non-steered wheel has two DOFs: motion in the direction of the wheel orientation, and rotational slip about the point of contact, corresponding to the two wheel pseudo-velocities ${}^{\overline{C_i}}v_{C_iy}$, and ${}^{\overline{C_i}}\omega_{C_i}$, respectively. The actual wheel velocities are the angular velocity of the wheel about its axle ω_{w_ix} and the angular velocity of the rotational slip ω_{w_iz}. These velocities are related by the (4×2) wheel matrix $\mathbf{W}_i$ in (A3.2.1).

$$\dot{\mathbf{q}}_i = \begin{pmatrix} 0 & 0 \\ R & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \omega_{w_ix} \\ \omega_{w_iz} \end{pmatrix} = \mathbf{W}_i \mathbf{q}_i \qquad (A3.2.1)$$

The wheel matrix is multiplied by the pseudo-Jacobian matrix in (A3.1.2) to form the (3×2) Jacobian matrix:

Conventional Non-Steered Wheel Jacobian Matrix

$$\mathbf{J}_1 = \begin{pmatrix} -R_i \sin {}^R\theta_{C_i} & {}^Rd_{C_iy} \\ R_i \cos {}^R\theta_{C_i} & -{}^Rd_{C_ix} \\ 0 & 1 \end{pmatrix}. \qquad (A3.2.2)$$

This wheel is termed *degenerate* because the Jacobian is non-square and thus non-invertible. Even though a robot velocity vector can be calculated from a wheel velocity vector, it is not always possible to compute a wheel velocity vector from a robot velocity vector. The degenerate nature of the kinematic equations-of-motion of the non-steered conventional wheel precludes its application to three DOF WMRs.

3.3 Conventional Steered Wheel

The conventioanl steered wheel has an additional DOF provided by the steering joint corresponding to the pseudo-velocity ${}^{H_i}\omega_{S_i}$. The actual steering velocity $\omega_{s_i z}$ (in Figure 3.2) is equal to the steering pseudo-velocity. The (4×3) wheel matrix and the (3×3) wheel Jacobian matrix are, respectively:

$$\dot{\mathbf{q}}_i = \begin{pmatrix} 0 & 0 & 0 \\ R & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \omega_{w_i z} \\ \omega_{w_i z} \\ \omega_{s_i z} \end{pmatrix} = \mathbf{W}_i \, \mathbf{q}_i \qquad (A3.3.1)$$

and

Conventional Steered Wheel Jacobian Matrix

$$\mathbf{J}_1 = \begin{pmatrix} -R_i \sin {}^R\theta_{C_i} & {}^Rd_{C_iy} & -{}^Rd_{H_iy} \\ R_i \cos {}^R\theta_{C_i} & -{}^Rd_{C_ix} & {}^Rd_{H_ix} \\ 0 & 1 & -1 \end{pmatrix}. \qquad (A3.3.2)$$

The Jacobian matrix is invertible if its determinant is nonzero; i.e., if

$$det(\mathbf{J}_i) \;=\; R_i\,({}^{S_i}d_{C_iy}\cos{}^{S_i}\theta_{C_i} - {}^{S_i}d_{C_ix}\sin{}^{S_i}\theta_{C_i}) \neq 0\;. \qquad (A3.3.3)$$

The determinant is zero and the conventional steered wheel is redundant if the steering axis intercepts the wheel point of contact (i.e., if ${}^{S_i}d_{C_ix} = {}^{S_i}d_{C_iy} = 0$) or if the wheel is oriented perpendicular to the steering link (i.e., if ${}^{C_i}d_{S_iy} = {}^{S_i}d_{C_ix}sin{}^{S_i}\theta_{C_i} - {}^{S_i}d_{C_iy}\cos{}^{S_i}\theta_{C_i} = 0$).

3.4 Omnidirectional Wheel

The omnidirectional wheel possesses three DOFs without a steering joint. The DOFs are motion in the direction of the wheel orientation, motion in the direction of the roller orientation and rotational slip, which correspond respectively to the actual wheel velocities ω_{w_ix}, ω_{w_ir}, and ω_{w_iz}. The pseudo-velocities $\dot{\mathbf{q}}_i$ are linear combinations of the actual velocities $\mathbf{q}_i$:

$$\dot{\mathbf{q}}_i \;=\; \begin{pmatrix} 0 & r\sin\eta & 0 \\ R & -r\cos\eta & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \omega_{w_ix} \\ \omega_{w_ir} \\ \omega_{w_iz} \end{pmatrix} \;=\; \mathbf{W}_i\,\mathbf{q}_i\;. \qquad (A3.4.1)$$

The wheel Jacobian matrix is:

Omnidirectional Wheel Jacobian Matrix

$$\mathbf{J}_i \;=\; \begin{pmatrix} -R_i\sin{}^{R}\theta_{C_i} & r_i\sin({}^{R}\theta_{C_i}+\eta_i) & {}^{R}d_{C_iy} \\ R_i\cos{}^{R}\theta_{C_i} & -r_i\cos({}^{R}\theta_{C_i}+\eta_i) & -{}^{R}d_{C_ix} \\ 0 & 0 & 1 \end{pmatrix}\;. \qquad (A3.4.2)$$

The determinant of the omnidirectional wheel Jacobian matrix is $-R_ir_i\sin\eta_i$, and consequently the Jacobian matrix is invertible whenever the rollers are not aligned with the wheel (i.e., whenever $\eta_i \neq 0$).

3.5 Ball Wheel

The ball wheel possesses three DOFs of rotation about the three normal axes positioned at the wheel center. The wheel matrix relating the actual wheel velocities ω_{w_ix}, ω_{w_iy} and ω_{w_iz} to the

pseudo-velocities is:

$$\hat{\mathbf{q}}_i = \begin{pmatrix} R & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \omega_{w_i x} \\ \omega_{w_i y} \\ \omega_{w_i z} \end{pmatrix} = \mathbf{W}_i \, \mathbf{q}_i \, . \qquad (A3.5.1)$$

The wheel Jacobian matrix is:

Ball Wheel Jacobian Matrix

$$\mathbf{J}_i = \begin{pmatrix} R_i \cos {}^R\theta_{C_i} & -R_i \sin {}^R\theta_{C_i} & {}^R d_{C_i y} \\ R_i \sin {}^R\theta_{C_i} & R_i \cos {}^R\theta_{C_i} & -{}^R d_{C_i x} \\ 0 & 0 & 1 \end{pmatrix} . \qquad (A3.5.2)$$

Since the determinant of the ball wheel Jacobian matrix is R_i^2, it is invertible for all non-zero wheel radii.

In Section 7, the wheel Jacobian matrices developed in this appendix are applied to obtain the kinematic equations-of-motions of specific WMRs.

4. Appendix 4: Actuated Inverse Solution Matrix Calculations

In this appendix, we detail the matrix manipulations leading to the actuated inverse solution in Section 5.5 . We solve the composite partitioned robot equation in (5.5.3)

$$\mathbf{A}_0\dot{\mathbf{p}} = \begin{pmatrix} \mathbf{I}_1 \\ \mathbf{I}_2 \\ \vdots \\ \mathbf{I}_N \end{pmatrix} \dot{\mathbf{p}} = \begin{pmatrix} \mathbf{J}_{1a} & 0 & \dots & 0 & \mathbf{J}_{1u} & 0 & \dots & 0 \\ 0 & \mathbf{J}_{2a} & \ddots & \vdots & 0 & \mathbf{J}_{2u} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \mathbf{J}_{Na} & 0 & \dots & 0 & \mathbf{J}_{Nu} \end{pmatrix} \begin{pmatrix} \dot{\mathbf{q}}_{1a} \\ \dot{\mathbf{q}}_{2a} \\ \vdots \\ \dot{\mathbf{q}}_{Na} \\ \dot{\mathbf{q}}_{1u} \\ \dot{\mathbf{q}}_{2u} \\ \vdots \\ \dot{\mathbf{q}}_{Nu} \end{pmatrix} = \mathbf{B}_{0p} \begin{pmatrix} \dot{\mathbf{q}}_a \\ \dot{\mathbf{q}}_u \end{pmatrix} \tag{A4.1}$$

to calculate the actuated wheel velocities $\dot{\mathbf{q}}_a$ in the least-squares solution in (5.5.4):

$$\begin{pmatrix} \dot{\mathbf{q}}_a \\ \dot{\mathbf{q}}_u \end{pmatrix} = (\mathbf{B}_{0p}^T\mathbf{B}_{0p})^{-1}\,\mathbf{B}_{0p}^T\,\mathbf{A}_0\,\dot{\mathbf{p}}\,. \tag{A4.2}$$

We begin by forming the matrix product:

$$(\mathbf{B}_{0p}^T\mathbf{B}_{0p}) = \begin{pmatrix} \mathbf{J}_{1a}^T\mathbf{J}_{1a} & 0 & \dots & 0 & \mathbf{J}_{1a}^T\mathbf{J}_{1u} & 0 & \dots & 0 \\ 0 & \mathbf{J}_{2a}^T\mathbf{J}_{2a} & \ddots & \vdots & 0 & \mathbf{J}_{2a}^T\mathbf{J}_{2u} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \mathbf{J}_{Na}^T\mathbf{J}_{Na} & 0 & \dots & 0 & \mathbf{J}_{Na}^T\mathbf{J}_{Nu} \\ \mathbf{J}_{1u}^T\mathbf{J}_{1a} & 0 & \dots & 0 & \mathbf{J}_{1u}^T\mathbf{J}_{1u} & 0 & \dots & 0 \\ 0 & \mathbf{J}_{2u}^T\mathbf{J}_{2a} & \ddots & \vdots & 0 & \mathbf{J}_{2u}^T\mathbf{J}_{2u} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \mathbf{J}_{Nu}^T\mathbf{J}_{Na} & 0 & \dots & 0 & \mathbf{J}_{Nu}^T\mathbf{J}_{Nu} \end{pmatrix}$$

$$= \begin{pmatrix} \mathbf{D}_{aa} & \mathbf{D}_{au} \\ \mathbf{D}_{au}^T & \mathbf{D}_{uu} \end{pmatrix}. \tag{A4.3}$$

To invert $(\mathbf{B}_{0p}^T\mathbf{B}_{0p})$, we have written the matrix in block form with four components, each one a block diagonal matrix. We let the block matrix $\mathbf{X}$ be the inverse of the matrix in (A4.3).

To compute the block components of the matrix inverse in terms of the block components of the matrix in (A4.3), we apply the fact that the inverse of a matrix times the matrix itself is the identity matrix; i.e.,

$$\begin{pmatrix} \mathbf{X}_{11} & \mathbf{X}_{12} \\ \mathbf{X}_{21} & \mathbf{X}_{22} \end{pmatrix} \begin{pmatrix} \mathbf{D}_{aa} & \mathbf{D}_{au} \\ \mathbf{D}_{au}^T & \mathbf{D}_{uu} \end{pmatrix} = \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} . \tag{A4.4}$$

Since we seek only the upper (actuated) components of the wheel velocity vector $\dot{\mathbf{q}}_a$ in (A4.1), we calculate only the two components in the top row of the block matrix inverse. We thus seperate the solution of the actuated wheel velocities

$$\dot{\mathbf{q}}_a = (\mathbf{X}_{11} \quad \mathbf{X}_{12}) \mathbf{B}_{0p}^T \mathbf{A}_0 \dot{\mathbf{p}} \tag{A4.5}$$

from the solution of the unactuated ones. We expand (A4.4) to obtain

$$\mathbf{X}_{11}\mathbf{D}_{aa} + \mathbf{X}_{12}\mathbf{D}_{au}^T = \mathbf{I} \tag{A4.6}$$

and

$$\mathbf{X}_{11}\mathbf{D}_{au} + \mathbf{X}_{12}\mathbf{D}_{uu} = \mathbf{0} . \tag{A4.7}$$

From (A4.6) and (A4.7), we find

$$\mathbf{X}_{12} = -\mathbf{X}_{11}\mathbf{D}_{au}\mathbf{D}_{uu}^{-1} \tag{A4.8}$$

and

$$\mathbf{X}_{11} = (\mathbf{D}_{aa} - \mathbf{D}_{au}\mathbf{D}_{uu}^{-1}\mathbf{D}_{au}^T)^{-1} , \tag{A4.9}$$

where

$$\mathbf{D}_{uu}^{-1} = \begin{pmatrix} (\mathbf{J}_{1u}^T\mathbf{J}_{1u})^{-1} & 0 & \cdots & 0 \\ 0 & (\mathbf{J}_{2u}^T\mathbf{J}_{2u})^{-1} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & (\mathbf{J}_{Nu}^T\mathbf{J}_{Nu})^{-1} \end{pmatrix} . \tag{A4.10}$$

The matrix $\mathbf{X}_{11}$ in (A4.9) is

$$\mathbf{X}_{11} = \begin{pmatrix} -[\mathbf{J}_{1a}^T\Delta(\mathbf{J}_{1u})\mathbf{J}_{1a}]^{-1} & 0 & \cdots & 0 \\ 0 & -[\mathbf{J}_{2a}^T\Delta(\mathbf{J}_{2u})\mathbf{J}_{2a}]^{-1} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -[\mathbf{J}_{Na}^T\Delta(\mathbf{J}_{Nu})\mathbf{J}_{Na}]^{-1} \end{pmatrix} \tag{A4.11}$$

The matrix $\mathbf{X}_{12}$ in (A4.8) is

$$\mathbf{X}_{12} = \begin{pmatrix} \mathbf{A}_1 & 0 & \dots & 0 \\ 0 & \mathbf{A}_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \mathbf{A}_N \end{pmatrix} \qquad (A4.12)$$

where,

$$\mathbf{A}_i = [\mathbf{J}_{ia}^T \Delta(\mathbf{J}_{iu}) \mathbf{J}_{ia}]^{-1} \mathbf{J}_{ia}^T \mathbf{J}_{iu} (\mathbf{J}_{iu}^T \mathbf{J}_{iu})^{-1} \quad . \qquad (A4.13)$$

We substitute (A4.12) and (A4.11) into (A4.5) to obtain the actuated wheel velocity vector

$$\dot{\mathbf{q}}_a = \begin{pmatrix} [\mathbf{J}_{1a}^T \Delta(\mathbf{J}_{1u}) \mathbf{J}_{1a}]^{-1} \mathbf{J}_{1a}^T \Delta(\mathbf{J}_{1u}) \\ [\mathbf{J}_{2a}^T \Delta(\mathbf{J}_{2u}) \mathbf{J}_{2a}]^{-1} \mathbf{J}_{2a}^T \Delta(\mathbf{J}_{2u}) \\ \vdots \\ [\mathbf{J}_{Na}^T \Delta(\mathbf{J}_{Nu}) \mathbf{J}_{Na}]^{-1} \mathbf{J}_{Na}^T \Delta(\mathbf{J}_{Nu}) \end{pmatrix} \dot{\mathbf{p}} \, . \qquad (A4.14)$$

Equation (A4.14) is the least-squares solution for the actuated wheel velocity vector. We note that this solution is applicable only when the matrix in (A4.3) is invertible. The conditions under which this solution is applicable are specified by the soluble motion criterion in (5.4.1).

5. Appendix 5: Sensed Forward Solution Matrix Calculations

In this appendix, we detail the matrix manipulations leading to the least-squares sensed forward solution. We solve the partitioned robot sensing equation in (5.7.2)

$$\mathbf{A}_n \begin{pmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{q}}_n \end{pmatrix} = \begin{pmatrix} \mathbf{I}_1 & -\mathbf{J}_{1n} & 0 & \dots & 0 \\ \mathbf{I}_2 & 0 & -\mathbf{J}_{2n} & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ \mathbf{I}_N & 0 & \dots & 0 & -\mathbf{J}_{Nn} \end{pmatrix} \begin{pmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{q}}_{1n} \\ \dot{\mathbf{q}}_{2n} \\ \vdots \\ \dot{\mathbf{q}}_{Nn} \end{pmatrix} \tag{A5.1}$$

$$= \begin{pmatrix} \mathbf{J}_{1s} & 0 & \dots & 0 \\ 0 & \mathbf{J}_{2s} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \mathbf{J}_{Ns} \end{pmatrix} \begin{pmatrix} \dot{\mathbf{q}}_{1s} \\ \dot{\mathbf{q}}_{2s} \\ \vdots \\ \dot{\mathbf{q}}_{Ns} \end{pmatrix} = \mathbf{B}_s \dot{\mathbf{q}}_s$$

to calculate the robot velocities $\dot{\mathbf{p}}$ in the least-squares solution in (5.7.4):

$$\begin{pmatrix} \dot{\mathbf{p}} \\ \dot{\mathbf{q}}_n \end{pmatrix} = (\mathbf{A}_n^T \mathbf{A}_n)^{-1} \mathbf{A}_n^T \mathbf{B}_s \dot{\mathbf{q}}_s \, . \tag{A5.2}$$

We begin by forming the matrix product

$$(\mathbf{A}_n^T \mathbf{A}_n) = \begin{pmatrix} N\mathbf{I} & -\mathbf{J}_{1n} & -\mathbf{J}_{2n} & \dots & -\mathbf{J}_{Nn} \\ -\mathbf{J}_{1n}^T & \mathbf{J}_{1n}^T \mathbf{J}_{1n} & 0 & \dots & 0 \\ -\mathbf{J}_{2n}^T & 0 & \mathbf{J}_{2n}^T \mathbf{J}_{2n} & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ -\mathbf{J}_{Nn}^T & 0 & \dots & 0 & \mathbf{J}_{Nn}^T \mathbf{J}_{Nn} \end{pmatrix} \tag{A5.3}$$

$$= \begin{pmatrix} N\mathbf{I} & \mathbf{T} \\ \mathbf{T}^T & \mathbf{D} \end{pmatrix} ,$$

where N is the number of wheels and $\mathbf{I}$ is the (3×3) identity matrix. We let the block matrix $\mathbf{X}$ be the inverse of the symmetric matrix $(\mathbf{A}_n^T \mathbf{A}_n)$ in (A5.3). Since the inverse of a matrix times the matrix is the identity matrix,

$$\begin{pmatrix} \mathbf{X}_{11} & \mathbf{X}_{12} \\ \mathbf{X}_{21} & \mathbf{X}_{22} \end{pmatrix} \begin{pmatrix} N\mathbf{I} & \mathbf{T} \\ \mathbf{T}^T & \mathbf{D} \end{pmatrix} = \begin{pmatrix} \mathbf{I} & 0 \\ 0 & \mathbf{I} \end{pmatrix} . \tag{A5.4}$$

We use the top block row of the matrix inverse to seperate the robot velocity vector $\dot{\mathbf{p}}$ from the non-sensed wheel velocity vector $\dot{\mathbf{q}}_n$:

$$\dot{\mathbf{p}} = (\mathbf{X}_{11} \quad \mathbf{X}_{12})\,\mathbf{A}_n^T\mathbf{B}_s\dot{\mathbf{q}}_s\,. \tag{A5.5}$$

From (A5.4), we obtain

$$\mathbf{X}_{11}N\mathbf{I} + \mathbf{X}_{12}\mathbf{T}^T = \mathbf{I} \tag{A5.6}$$

and

$$\mathbf{X}_{11}\mathbf{T} + \mathbf{X}_{12}\mathbf{D} = \mathbf{0}\,, \tag{A5.7}$$

from which

$$\mathbf{X}_{12} = -\mathbf{X}_{11}\mathbf{T}\mathbf{D}^{-1} \tag{A5.8}$$

and

$$\mathbf{X}_{11} = (N\mathbf{I} - \mathbf{T}\mathbf{D}^{-1}\mathbf{T}^T)^{-1}\,. \tag{A5.9}$$

The inverse of the block diagonal matrix $\mathbf{D}$ is:

$$\mathbf{D}^{-1} = \begin{pmatrix} (\mathbf{J}_{1n}^T\mathbf{J}_{1n})^{-1} & 0 & \cdots & 0 \\ 0 & (\mathbf{J}_{2n}^T\mathbf{J}_{2n})^{-1} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & (\mathbf{J}_{Nn}^T\mathbf{J}_{Nn})^{-1} \end{pmatrix}\,. \tag{A5.10}$$

We expand the block elements in (A5.8) and (A5.9) to obtain

$$\mathbf{X}_{12} = -\mathbf{X}_{11}\left[-\mathbf{J}_{1n}(\mathbf{J}_{1n}^T\mathbf{J}_{1n})^{-1} \quad -\mathbf{J}_{2n}(\mathbf{J}_{2n}^T\mathbf{J}_{2n})^{-1} \quad \cdots \quad -\mathbf{J}_{Nn}(\mathbf{J}_{Nn}^T\mathbf{J}_{Nn})^{-1}\right] \tag{A5.11}$$

where

$$\begin{aligned}\mathbf{X}_{11} &= [N\mathbf{I} - \mathbf{J}_{1n}(\mathbf{J}_{1n}^T\mathbf{J}_{1n})^{-1}\mathbf{J}_{1n}^T - \mathbf{J}_{2n}(\mathbf{J}_{2n}^T\mathbf{J}_{2n})^{-1}\mathbf{J}_{2n}^T - \ldots - \mathbf{J}_{Nn}(\mathbf{J}_{Nn}^T\mathbf{J}_{Nn})^{-1}\mathbf{J}_{Nn}^T]^{-1} \\ &= -[\Delta(\mathbf{J}_{1n}) + \Delta(\mathbf{J}_{2n}) + \ldots + \Delta(\mathbf{J}_{Nn})]^{-1}\end{aligned} \tag{A5.12}$$

Finally, we substitute (A5.11) and (A5.12) into (A5.5) to obtain the least-squares solution for the robot velocity vector:

$$\dot{\mathbf{p}} = [\Delta(\mathbf{J}_{1n}) + \Delta(\mathbf{J}_{2n}) + \ldots + \Delta(\mathbf{J}_{Nn})]^{-1}[\Delta(\mathbf{J}_{1n})\mathbf{J}_{1s} \;\; \Delta(\mathbf{J}_{2n})\mathbf{J}_{2s} \;\; \cdots \;\; \Delta(\mathbf{J}_{Nn})\mathbf{J}_{Ns}]\dot{\mathbf{q}}_s\,. \tag{A5.13}$$

In Section 5.8, we develop the adequate sensing criterion which ensure the invertability of the matrix $(\mathbf{A}_n^T\mathbf{A}_n)$ in (A5.3) and thereby the applicability of the least-squares sensed forward solution in (A5.13).