Feedback Control of a Class of Nonholonomic Hamiltonian Systems

Sørensen, Mathias Jesper

Publication date:
2005

Document Version
Publisher's PDF, also known as Version of record

Feedback Control of a Class of Nonholonomic Hamiltonian Systems

Ph.D. Thesis

Mathias Jesper Sørensen

Department of Control Engineering
Institute of Electronic Systems
Aalborg University
Fredrik Bajers Vej 7, DK-9220 Aalborg Ø, Denmark.

ISBN 87-90664-28-0
August 2005

PREFACE

This thesis is submitted as partial fulfillment of the requirements for the Doctor of Philosophy at the Department of Control Engineering at the Institute of Electronic Systems, Aalborg University, Denmark. The work has been carried out in the period from August 2002 to August 2005 under the supervision of Associate Professor Tom Søndergaard Pedersen.

Aalborg, August 2005 Mathias Jesper Sørensen

ACKNOWLEDGMENT

Three years have passed since the beginning of this venture and you are now holding the final outcome. It has been a great experience; at times frustrating, at times the most interesting thing in the world. Nevertheless, it has been a great personal learning experience for me, but this thesis could not have been completed without the help from all the people who have supported me during the past three years and during my time as a student at the Department of Control Engineering.

I would like to thank my supervisor Tom S. Pedersen for keeping my head above water with his support and guidance during the past three years. A special thanks goes to the rest of the 'team': Jan Dimon Bendtsen for his great and contagious enthusiasm, Palle Andersen for his thorough and well places questions, which has lead to many interesting discussions, and Jens Dalsgaard for his practical help. I would also like to thank all the students who have worked on building the AV robot. Without you it would still have been stuck in the laboratory. A very broad thanks to everybody at the Department for upholding what I find to be a superb social atmosphere. Keep it up in the future!

During spring 2004 I was a guest at the Control Laboratory at Twente University, and I would like to thank Professor Stefano Stramigioli, Carla Gouw-Banse, and every one (there are too many to mention) at the Control Laboratory for welcoming me and making my stay a very pleasant one. And to everyone who lived in Macandra during spring 2004; I wish you all a bright future.

Finally, I would like to thank Pernille Dahl for always supporting me, even when we have been on different continents, and for bearing with me and my absent-mindedness during the past months.

ABSTRACT

Feedback control of nonholonomic systems has always been problematic due to the nonholonomic constraints that limit the space of possible system velocities. This property is very basic, and Brockett proved that a nonholonomic system cannot be asymptotically stabilized by a time-invariant smooth feedback. This thesis presents a novel way of controlling a special class of nonholonomic Hamiltonian systems. The basic idea is to split the configuration coordinates in two; a primary part that we wish to asymptotically stabilize, and a secondary part that not necessarily has to be stabilized, but is useful when controlling the primary part. The secondary part is introduces as the integral of so-called kinematic inputs. The kinematic inputs have the property that they cannot change the amount of energy in the system, i.e., the Hamiltonian function is invariant with respect to the kinematic inputs. The resulting nonholonomic Hamiltonian system with kinematic inputs shares many of the properties of the classical Hamiltonian system, and some of the methods involved in controlling classical systems are proved to also apply to the augmented system. The extra degree of freedom provided by the kinematic inputs turns out to be useful when stabilizing the nonholonomic system. If the system is properly actuated it is possible to asymptotically stabilize the primary part of the configuration coordinates via a passive energy shaping and damping injecting feedback. The feedback is smooth and time-invariant, but since it does not asymptotically stabilize the secondary part of the configuration coordinates, it does not violate Brockett's obstruction.

The results from the general class of nonholonomic Hamiltonian systems with kinematic inputs are applied to a real implementation of a four wheel steered, four wheel driven nonholonomic robotic vehicle, where the velocity of the steering motors are assumed to satisfy the conditions of proper kinematic inputs. The proposed controller is general enough to achieve both global asymptotic stabilization and path tracking for the robot. To improve the operation of the closed loop system some extensions are provided: integral action for asymptotic stabilization under the influence of disturbances, and an adaptive damping scheme ensuring that the robot travels at a predefined speed when tracking a path. Both of these extensions are defined in the framework of Hamiltonian systems.

CONTENTS

List of Figures **xiii**

List of Tables **xvii**

Glossary & Nomenclature **xix**

1 Introduction **1**

1.1 The API Project - Background and Motivation 2

1.2 Delimitation of Study 5

1.3 Previous and Related Work 6

1.4 Contributions of This Work 9

1.5 Thesis Overview 11

2 Hardware and Software on the AV **13**

2.1 Hardware 14

2.1.1 AV Frame 15

2.1.2 Steering System 16
2.1.3 Propulsion System 18
2.1.4 Sensors 19
2.1.5 Computer System 26
2.1.6 Hardware Communication Structure 29
2.2 Software 30
2.2.1 Overview of Major Software Components 30
2.2.2 Real-Time Workshop 32
2.2.3 PC/104 Software 35
2.2.4 LH-Agros Software 36

3 Lagrange Model of the AV 39
3.1 Vehicle Definition 40
3.2 Constraints 42
3.3 Lagrange Model 45

4 The Hamiltonian Formulation and Model Reduction 49
4.1 The Constrained Tangent Bundle 50
4.2 Kinematic Inputs 50
4.3 The Hamiltonian Model 52
4.4 Eliminating the Lagrange multipliers 54
4.5 Inertia Matrix Scaling 57
4.6 Discussion 58

5 Reduction and Validation of the AV Model 61

5.1 The Reduced Hamiltonian Equivalent 61
5.2 Model Validation 68

6 Feedback Control of Systems with Kinematic Inputs 79

6.1 Passivity of Dynamic Systems 80
6.2 Example of Passivity Based Control 81
6.3 Energy Shaping 86
6.4 Damping Injection 88
6.5 Asymptotic Stability 88
6.6 Discussion 92

7 Feedback Control of the AV 95

7.1 Energy Shaping and Damping Injection 96
7.2 Torque Distribution 99
7.3 Convergence Towards $\mathbf{Q_0}$ 102
7.4 Disturbances and Integral Action 104
7.5 Path Tracking 112
7.5.1 Tracking a Circle 112
7.5.2 Inter Crop Row Potentials 116
7.5.3 Adaptive Damping 118
7.5.4 Putting it All Together 121
7.6 Discussion 123

8 Physical Tests 125

8.1 Convergence toward a Single Point 125

8.2 Tracking a Line 126
8.3 Tracking a Circle 129
8.4 Putting it All Together 132
8.5 Discussion 133

9 Conclusions and Future Work **135**
9.1 Summary of the Results 135
9.2 Recommendations for Future Work 137

A Hamilton's Principle and Lagrange's Equation **139**
A.1 Lagrange's Equation 139
A.2 Extension to Nonholonomic Systems 142

Bibliography **145**

LIST OF FIGURES

1.1 The Autonomous Vehicle 3

1.2 Weed mapping 4

1.3 Bead moving along a fixed path 7

2.1 Crop row changing with car-like steering 15

2.2 The AV, right side view 16

2.3 Crop row changing with front+rear wheel steer 17

2.4 ICR of a front+rear wheel steer 17

2.5 ICR of a full four wheel steer 18

2.6 The AV, rear view 21

2.7 The AV, rear instrument box 21

2.8 The AV, front instrument box 22

2.9 Deviation in dead-reckoning 23

2.10 ECO-DAN crop row guidance camera 26

2.11 Hardware communication structure on the AV 29
2.12 Overview of software structure 31
2.13 Code generation and compilation procedure of Real-Time Workshop . . 33
2.14 The Simulink user interface during simulation 34
2.15 Introducing hardware in the loop 35
2.16 Software components on the PC/104 36
2.17 First order motor model 37

3.1 Definition of the AV body frame coordinates 40
3.2 Definition of coordinates, inputs and parameters related to each wheel . 41
3.3 The ICR revisited 42

5.1 Driving along a straight line 66
5.2 Rotating around the geometric center 67
5.3 Step response of special case 1 and 2 70
5.4 AV and model step responses of special case 1 and 2 72
5.5 Block diagram of the verification setup 73
5.6 Inputs used for verification 73
5.7 Measured steering angles 74
5.8 Measured and simulated translational velocity 75
5.9 Measured and simulated translational velocity (closeup) 75
5.10 Measured and simulated angular velocity of the AV 76
5.11 Measured and simulated angular velocity of the AV (closeup) 76
5.12 Model verification, position 77

5.13 Model verification, orientation 78

6.1 A mass moving on a frictionless surface 82
6.2 Energy shaping 83
6.3 Energy shaping + Damping injection 84
6.4 Responses of the simple example 85
6.5 A knife's edge moving in a potential field 89
6.6 Set of equilibrium points of the knife's edge example 90

7.1 Two methods of actuating of the AV 98
7.2 Unit vectors of rotating forces 104
7.3 Stabilization at a single point 105
7.4 Stabilization at a single point on a non-horizontal field. 105
7.5 Stabilization at a single point using integral action 109
7.6 Position and angular errors using integral action 110
7.7 Closed loop of the simple example 111
7.8 Potential function of a circle 113
7.9 Potential pull along the circle 114
7.10 Combined potential function 114
7.11 Tracking the circle 115
7.12 Inter crop row potentials 117
7.13 Tracking the circle with inter row potentials, low gain 119
7.14 Tracking the circle with inter row potentials, high gain 119
7.15 Velocity of the AV during path tracking with constant damping 120

7.16 Velocity of the AV during path tracking with adaptive damping 121

7.17 Tracking four crop rows, position and orientation 122

7.18 Tracking four crop rows, velocity 123

8.1 Converging to a single point 126

8.2 Converging to a single point with integral action 127

8.3 Position and orientation errors with and without integral action 127

8.4 Tracking a line. Constant damping 128

8.5 Velocity of the line tracking test with constant damping 128

8.6 The 40kg anchor used as external disturbance 129

8.7 Tracking a line. Adaptive damping 130

8.8 Velocity and damping values of the line tracking test with adaptive damping 130

8.9 Tracking a circle, low gain 131

8.10 Tracking a circle, high gain 132

8.11 Tracking four crop rows in practice; position and orientation 133

8.12 Tracking four crop rows in practice; velocity 134

LIST OF TABLES

2.1 The steering motors 18

2.2 The RTK-GPS 22

2.3 The compass and tilt sensor 24

2.4 The fiber optic gyro 24

2.5 The steering motor encoders 25

2.6 The ground speed radar 25

2.7 The crop row guidance camera 26

2.8 The LH-Agro embedded computer 28

2.9 The PC/104 stack 28

2.10 Parameters associated with the motors 37

5.1 Directly measurable AV model parameters. (*) taken from [7] 69

5.2 Motor and friction parameters 71

GLOSSARY & NOMENCLATURE

4WD	Four wheel drive, 18
4WS	Four wheel steer, 18
$\boldsymbol{A}$	Constraint matrix, 44
APF	Artificial Potential Field, 9
API	Autonomous Platform and Information system for registration of crops and weeds, 2
AV	Autonomous Vehicle, 2
$\boldsymbol{B}$	Input matrix, 51
$\boldsymbol{\mathcal{B}}$**-frame**	Coordinate frame fixed instantaneously at the position and orientation of the AV, 40
$\boldsymbol{\beta_i,\ i = 1,2,3,4}$	Steering angle of the i'th wheel ($\beta' = [\beta_1\ \beta_2]^T$), 40
$\boldsymbol{C_1}$	Free rolling constraint matrix, 44
$\boldsymbol{C_2}$	Non-slipping constraint matrix, 45
$\boldsymbol{\chi}$	$\chi = [x_1\ x_2\ \theta]^T$, the position and orientation of the AV with respect to the $\mathcal{N}$-frame, 40
$\boldsymbol{D}$	Damping matrix, 86
$\boldsymbol{(\gamma_i, \kappa_i),\ i = 1,2,3,4}$	Position in polar coordinates of the i'th wheel with respect to the geometric center of the AV, 40
$\boldsymbol{H}$	Hamiltonian function, 52
$\boldsymbol{\tilde{H}}$	Hamiltonian function of the reduced system, 55

$\bar{H}$	Shaped Hamiltonian function, 87
ICR	Instantaneous Center of Rotation, 16
J	Interconnection matrix, 56
j_f	Moment of inertia of AV body, 46
$j_{w,\phi}$	Moment of inertia of each wheel about the ϕ rotation axes, 46
L	Lagrangian function, 45
λ	Lagrange undetermined multipliers, 45
$\mathcal{M}$	Configuration manifold, 40
M	Inertia matrix, 46
$\bar{M}$	Augmented inertia matrix, 48
m_f	AV body mass, 46
m_w	Wheel mass, 46
$\mathcal{N}$-frame	Earth fixed coordinate frame, 40
Ω_c	Constrained tangent bundle, 50
Ω_c^*	Constrained cotangent bundle, 54
p	Generalized momentum, 52
$\tilde{p}_1$	Reduces momentum, 55
PCHS	Port-Controlled Hamiltonian System, 8
$\phi_i,\ i = 1,2,3,4$	Forward rotation angle of the i'th wheel, 40
q	Generalized coordinates, 45
Q_0	Set of minima of $U + \bar{U}$, 87
R	Rotation matrix from $\mathcal{N}$-frame to $\mathcal{B}$-frame, 42
r_w	Wheel radius, 40
r	Coordinates related to the kinematic inputs ($\dot{r} = v$), 51
RTK-GPS	Real Time Kinematics - Global Positioning System, 20
RTW	Real-Time Workshop, 32
S	Coordinate transformation matrix, 55
s	Supply rate, 80
Σ	AV kinematics vector, 64

$\boldsymbol{T}$	Kinetic energy, 45
$\boldsymbol{\tau_{\phi_i},\ i = 1,2,3,4}$	Effective torque of the i'th propulsion motor, 40
$\boldsymbol{T\mathcal{M}}$	Tangent bundle of $\mathcal{M}$, 50
$\boldsymbol{T^*\mathcal{M}}$	Cotangent bundle of $\mathcal{M}$, 52
$\boldsymbol{T_q\mathcal{M}}$	Tangent space of $\mathcal{M}$ at q, 50
$\boldsymbol{T_q^*\mathcal{M}}$	Cotangent space of $\mathcal{M}$ at q, 52
$\boldsymbol{U}$	Potential energy, 45
$\boldsymbol{\bar{U}}$	Shaping potential energy, 87
$\boldsymbol{\Upsilon}$	Inertia scaling matrix, 58
$\boldsymbol{V}$	Storage function, 80
$\boldsymbol{v}$	Kinematic input, 51
$\boldsymbol{\zeta}$	Angular velocity of steering motor 1 and 2, 48

CHAPTER 1

INTRODUCTION

Throughout modern history agricultural research has been, and still is, an area of large economic, environmental, and political interest, and with the introduction of state-of-the-art technologies, innovative new tools for increasing the size and quality of agricultural outputs are emerging.

Agricultural science has in recent years made great advances on the use of robotics in agriculture. Farmers are already beginning to implement automated fruit pickers, weeding vehicles, pigsty cleaners, and milking machines in their production. These advances are often governed by the need for more efficient production methods, or methods that reduce the strain on the environment or increases animal welfare. With more than 60% of Denmark cultivated and a total sales profit of 88.3 billion DKK in 2000[1], Danish agronomic and horticultural knowledge is playing an important role on today's international agricultural scene.

[1] From *Facts & Figures, Agriculture in Denmark.* Published by The Danish Agricultural Council, 2002

1.1 The API Project - Background and Motivation

The acronym API stands for *Autonomous Platform and Information system for registration of crops and weeds*, and it is a joint research project with the purpose of prototyping a standalone system that is able to collect local data on the weed and crop state in a field. This data can then be used to build a map showing the spatial distribution of crops and weeds.

A detailed spatial weed map of a field is useful for the farmer when planning his field treatment, since knowledge of the coverage of different species of weeds gives the farmer the opportunity to mix an optimal herbicide agent. Using a variable rate sprayer combined with a positioning system the farmer is then able to precisely apply the agent in the right amounts at the right locations. And when considering crop treatment, the map can be used for precise application of fertilizers and water. This *precision spraying/fertilizing*, which is a subset of the broader terminology *precision farming*, should ultimately result in a reduction in herbicide and fertilizer use and an increase in crop quality and stability. Developing methods for generating detailed weed and crop maps is therefore important for both environmental and economic reasons.

For crop and weed information gathering there already exists a range of different sensors that can be mounted on tractors, combine harvesters, or other human operated machinery. Gathering weed and crop information is a time consuming task, and even if the farmer uses a vehicle mounted sensor, data collection is usually done only in conjunction with sowing, spraying, fertilizing, or harvesting. Often this is either too early or too late, as the crop treatment is most effective when the crops have just germinated. The API project is therefore focused on an alternative method of gathering the data. In this project the main platform for carrying the sensors into the field is a small autonomous vehicle, from now on denoted the AV. See figure 1.1.

The AV is of a relatively small size with a sideways and front to rear tread distance of $1\mathrm{m}$ and a total weight of approximately $230\mathrm{kg}$. The major benefit of the small size and weight is that the tread pressure is decreased, as compared with that of a tractor, and the soil compaction and crop damage is hence greatly reduces. To accommodate a large degree of freedom, the AV has been constructed as a four wheel driven, four wheel steered (4WD-4WS) vehicle. This gives the AV the ability to rotate around any point and hence also drive sideways. The AV is designed to carry any kind of sensor or implement into the field; it might be vision sensors, like color or infrared cameras, or it might be sensors that need direct physical contact with the soil, such as soil sampling equipment or soil compaction sensors. It has also been discussed to put a mechanical weeding implement on the AV, so that weeding can be done on the fly.

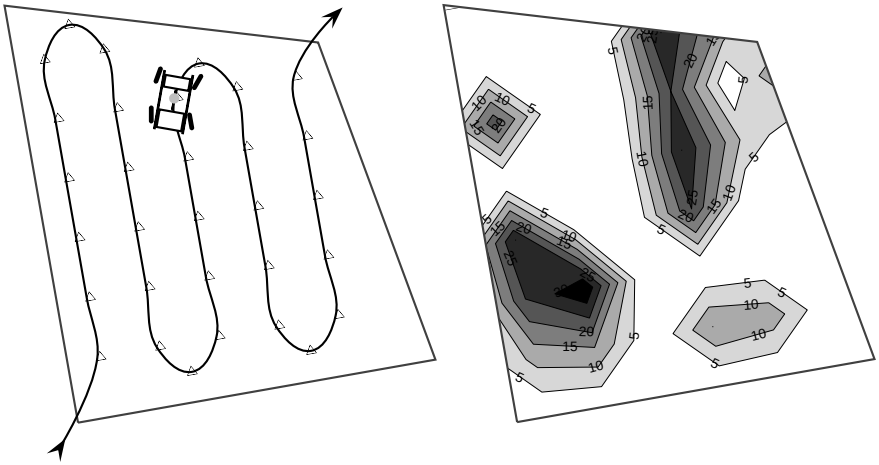
Figure 1.1: *The Autonomous Vehicle*

The implement used in the API project is limited to a high resolution color camera, and a large separate part of the project is to develop robust computer vision algorithms to identify a range of different weed species from color photographs. Apart from the AV itself and the primary sensor, the API project includes one more important segment. With or without the implement the AV cannot complete the task of mapping a field by itself. The project therefore also incorporates a base station, typically located at the farm, for job planning. The base station also handles high level task management and data handling, and it is the primary interface between the farmer and the API. At the base station the farmer can plan future tasks and study current and previous weed and crop maps.

A typical field mapping job would progress as follows:

1. The farmer defines a new job and sets it up in the base station software. The job description includes, as a minimum, the boundaries of the field to be mapped, sowing direction, and information on the time of execution

2. The base station generates a grid of way-points in the current field. The way-points includes sample points that the AV must drive to and take a photo, and intermediate navigation points that the AV should just pass through



(a) Field traversal (b) Percentile weed coverage map.

Figure 1.2: *Weed mapping*

3. The AV is driven manually to the field, and a wireless connection is established with the base station
4. The AV then asks the base station to transmit the first way-point, and it is now up to the AV to reach this point without damaging the crops. When it has reached the point, and if it is a sample point, it sends a signal to the camera telling it to take a photo. The camera then takes a photo and transmits it back to the base station for further processing
5. The AV then asks for the next way-point, drives to it, and continues in this manner until the entire field has been traversed. Figure 1.2(a) shows how the traversal of a (very small) field might look
6. The collected data is then processed, and a weed coverage map is generated. An example of a weed map can be seen in figure 1.2(b). The map shows the percentile coverage of White Goosefoot on an imaginary field

Thus, when the farmer has driven the AV into the field, he can forget about it until it has traversed the entire field and is ready to be taken home. The in-field driving might last several hours, but because the AV is driving autonomously, it will not be considered as a time consuming operation for the farmer. Building a system with so much autonomy is not an easy task, and it is further complicated by the growth state in the field. Gathering weed and crop information for spraying or fertilizing must be done

when the plants have just germinated, since spraying is most effective at this state of growth. At this stage the weeds are most vulnerable, but so are the crops, and this raises several interesting problems for the autonomy of the AV. Letting an autonomous vehicle drive in a just germinated field imposes strict demands on the precision and execution of the path planning algorithms and control laws to avoid unnecessary crop damage.

1.2 Delimitation of Study

As indicated in the previous section the development of the API consists of three distinct tasks:

1. Development of the camera vision system
2. Development of the base station
3. Development of the AV

The first task was handed to researchers at the *Danish Agricultural Research Center* at Bygholm, where they have developed a method using *active shape modeling* to identify different weed species from digital photos [57, 56]. The general idea is to build parametric models of the shape of each individual weed specie. By changing the parameters it is possible to change main features of the shape, such as growth stage, number of leaves, deformities, etc., and hence each model covers a large variation of each specie. Photos gathered from a field is then analyzed; the first step is to isolate every single weed and crop plant, and the second is to match them with all the weed models until a 'best fit' is reached.

The second task was handed to researchers at the *Department of Computer Science* at Aalborg University. They have created an abstract model of the entire API in order to design a suitable base station system. The base station also includes an efficient way of generating sample points in the field based on a dynamic sampling strategy, where the sample grid is adapted to local variations, such as previously measured weed density, or occurrence of special precarious weed species [59].

The third task was also handed to Aalborg University, but to researchers and students at the *Department of Control Engineering*. This thesis focuses on this task and on some of the instrumentation and control aspects of the AV.

Developing and prototyping the AV has involved several people, and it has spawned many sub-projects for students and researchers alike. The mechanical construction of the frame and mounting of the motors was carried out at Research Center Bygholm, but

everything else has been built, mounted, and tested by people (mostly students) at the department. This includes a power supply and safety system, construction of hardware interfaces for actuators and sensors, mounting of sensors, and design of a distributed software testbed for developing and testing navigation and control algorithms in real-time. Pedersen et al. [50] and Nielsen et al. [45] gave a conceptual description of the control system architecture on the AV. The vehicle in these two papers was a predecessor to the current AV, but the system architecture has been left virtually unchanged on the new vehicle. The system architecture was also the focus of Nielsen et al. [44], but only as an example when designing control architectures for a generic autonomous vehicles. Several papers have also been published on the control aspects of the AV. Andersen et al. [3] introduced a robust nonlinear controller based on feedback linearization, and Sørensen [61] introduced a controller based on artificial potential fields. This was extended to cover a more general class of systems in [62].

This thesis is specifically concerned with the software and control aspects on the AV with main focus on the control aspects. For proper operation in the field the AV has to be able to converge to single way-points as dictated by the base station, and while approaching the way-points, the AV has to drive along the crop rows to minimize crop damage. The control algorithms for the AV should hence be able to solve the standard problems of path tracking and asymptotic stabilization of a wheeled robot. The latter imposes an interesting problem, since asymptotic convergence toward a single point has always been an intrinsic difficulty in the control of wheeled robots.

1.3 Previous and Related Work

So what makes autonomous vehicles and mobile robots different from other electro-mechanical systems? Mobile robots are often imposed with non-integrable constraints that cannot directly be used to reduce the dimension of the system. Consider the bead on figure 1.3 moving along a fixed curved path in $\mathbb{R}^2$.

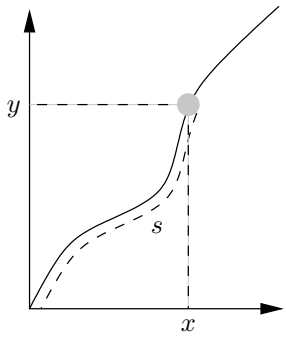


Figure 1.3: *Bead moving along a fixed path*

The position of the bead can be described using Cartesian coordinates x and y, but since the bead is constrained to lie on the curve, one coordinate would suffice; the arc length s along the curve, for example. This is an example of an integrable constraint that can be used to reduce the number of generalized coordinates of the system. A non-integrable or nonholonomic constraint, on the other hand, is defined as any constraint that cannot directly be used to reduce the number of coordinates. Nonholonomic constraints can have many different forms. They can be inequalities on the configuration space; if, in the previous example, the bead was constrained to lie on either side of the curve. When working with vehicles the nonholonomic constraints appear as constraints on the allowed direction of system velocities. Consider an ideal free rolling disk, for example; it would be imposed with constraints that guarantee the disk never slips sideways and never does a wheel-spin. These constraints are also known as free rolling and non-slipping constraints.

Ways of overcoming the obstacle of having nonholonomic constraints in vehicle control have been studied extensively in the last decades, and in the beginning, this was more or less the only problem that was addressed. The objectives back then was, and still is today, to develop tracking controllers and controllers that were able to asymptotically stabilize the vehicle, but basically it was all about solving or working around the problem of nonholonomicity. Brockett and Sussmann [14] proved that nonholonomic systems could not be asymptotically stabilized by any smooth time-invariant controller, and alternative approaches have been proposed ever since. Typically with some kind of switching involved making the feedback non-smooth or time-variant or both. A simple and very famous example of this, which can be found in almost any book on nonlinear systems, is the use of Lie brackets to parallel park a car like vehicle. A more general method for motion planning of driftless systems using Lie brackets were proposed by Lafferriere and Sussmann [39] who introduced a trajectory generating algorithm based on higher order Lie brackets. Many of the Lie bracket methods use non-smooth switching between constant inputs, and to avoid this hard switching Murray and Sastry [42] proposed to switch between smooth sinusoidal inputs instead. The steering were basically the same though. Other researchers worked on avoiding switching altogether, and Barraquand and Latombe [5] introduced an optimal path planner, where the input switching was minimized, and Kanayama et al. [34] proposed a tracking controller that used a virtual robot moving along a predefined trajectory as reference. This strategy required that the control actions for the virtual robot, and also to some degree for the actual robot, were known beforehand, and undesired controls could be avoided. Still using a switching strategy Canudas de Wit and Sørdalen [70] proposed a piecewise smooth controller that was able to exponentially stabilize a nonholonomic system. The method

was extended to trajectory tracking in [58]. Self-learning control has also been used successfully to control nonholonomic systems, and Nguyen and Widrow [43] used a neural network to drive a truck-and-trailer system in reverse.

Most of the pioneering work from around 1990 was focused on the kinematic part of the vehicles. The kinematics of a vehicle is the relation between velocities and configuration coordinates, and this is typically where the nonholonomic constraints enter. Good results were achieved with pure kinematic control as long as the robot was equipped with powerful enough actuators, so that the velocities of the robot could be controlled directly. This is usually the case for small robots, but when the robots increased in both size and weight the rigid body dynamics began to play an important role. Throughout the nineties the major contribution to the work on control of mobile robots was the inclusion of the natural dynamic part associated with rigid body motion. The full dynamics were already incorporated into the model of a general nonholonomic system by Bloch and McClamroch [10], and some considerations were given on stabilizability of the full system. This work was further elaborated in [11]. Thuilot et al. [66, 67] introduced a dynamic model for a general class of mobile robots. The authors also looked at a feedback linearization scheme, where the dynamic part suddenly played a significant role in the physical understanding of the linearization. The kinematic part of a mobile robot is only feedback linearizable through a dynamic extension, and the dynamic part of the full model is the natural choice. This was also exploited by Fierro and Lewis [22] who used a backstepping technique together with a dynamic extension to achieve path following and stabilization about a desired posture. Apart from introducing the dynamics, much of the work in the nineties was on improving existing methods. Godhavn and Egeland [29] and Samson [51] proposed a unified approach to stabilize nonholonomic systems on different forms, and Fukao et al. [26], Dong et al. [17], and Soetano et al. [55] proposed adaptive control schemes for a robot with parametric uncertainties. Many other methods have been proposed in the last decade, but common to many of these methods is that it is easy to loose track of the structural properties of the underlying physical systems; useful structural properties that are inherent in mobile robots and indeed all electro-mechanical systems. Especially feedback linearization will often result in a complex controller, and physical insight into the system is lost.

To keep and exploit some of the intrinsic structures of the system, some researchers have turned to new ways of representing general classes of electro-mechanical systems. A very useful one is the port-controlled Hamiltonian system (PCHS) that is general enough to describe almost any passive system (and hence any natural electro-mechanical system), while still capturing the inherent structures [53]. The PCHS is a generalization of the classical Hamiltonian representation of dynamic systems. Apart from the intrinsic symmetry, which the PCHS inherits from the classical Hamiltonian system, it is also equipped with an input/output port, hence the term *port-controlled*. The input/output port has the feature that the product between input and output is always power. The port

is therefore often called a *power-port*, and the energy flowing into the system is exactly the integral of this power-port.

Because the PCHS captures the physical structures, many of the proposed control methods for this kind of system have intuitive physical interpretations. Terms like *energy shaping* and *damping injection* are often encountered, and they are directly related to physical laws, such as energy conservation and the laws of motion. *Energy shaping* refers to a method with which it is possible to add or subtract artificial potential energy through feedback, thereby generating a closed loop system with shaped potential energy. *Damping injection* refers to increasing or decreasing the natural damping of the system, typically by a velocity feedback. These terms are all well described by Ortega et al. [49, 48] who introduced a general *Interconnection and Damping Assignment - Passivity Based Controller*, or IDA-PBC, for the PCHS. In theory, the IDA-PBC is able to convert any well behaved PCHS into any other PCHS of the same order. In practice though, the structure and controllability of the original PCHS often prohibits this, but the IDA-PBC can still lead to very useful results.

Many of the PCHS references already mentioned only deal with holonomic systems, but the theory applies largely to nonholonomic systems as well. van der Schaft and Maschke [54] introduced a PCHS description of a general nonholonomic system, and in [40] some results were presented on stabilizing the nonholonomic PCHS. Khennouf et al. [36] also described some preliminary results on asymptotic stabilization by switching between two different artificial potential energy functions in the energy shaping feedback. Fujimoto and Sugie [25] used canonical transformations to stabilize a nonholonomic PCHS, and in [23] it was extended to trajectory tracking. Duindam and Stramigioli [18] used energy considerations in a PCHS to accelerate a heavily under-actuated vehicle called a 'snakeboard'. The notion of *energy shaping* in a nonholonomic PCHS, which was used in many of these references, is closely related to the notion of *artificial potential fields* (APF), which have also been used successfully for path planning and control of mobile robots [13, 1, 27]. An artificial potential field is basically another word for the resulting potential energy in *energy shaping*.

1.4 Contributions of This Work

This thesis considers a class of nonholonomic Hamiltonian systems that encompass the AV and many other types of wheeled robots. Common to these systems is that the total physical energy is invariant with respect to a part of the configuration coordinates. This lays the grounds for defining the so-called *kinematic inputs* that can be used to asymptotically stabilize the system if it is sufficiently actuated. The important difference with this procedure, as opposed to many of the procedures described in the previous

section, is that the resulting feedback is smooth and time-invariant. This does not conflict with the results by Brockett and Sussmann [14], because the controlled system is on a slightly different form than the form used by Brockett. The proposed method is applied to the AV and tested in practice and have shown to be effective for both asymptotic convergence to a single way-point and for general crop row tracking. The controller for the AV is further improved by including integral action to guarantee asymptotic stability under the influence of a constant disturbance; and an adaptive damping scheme used to control the traveling velocity when tracking rows. All the improvements are given in the context of Hamiltonian systems.

The main contributions are summarized here:

- A **full dynamic model of the AV**, which is suitable for control, is developed using Lagrangian and Hamiltonian mechanics
- A **verification** of a simulation model is carried out from physical measurements on the real AV, and the model is shown to capture the dynamics of the physical system
- The introduction of so-called **kinematic inputs** in the framework of nonholonomic Hamiltonian systems is shown to give new insight in the control and stabilization of these systems. Kinematic inputs are able to directly change a subset of the configuration manifold through their first derivative. Furthermore, the total energy function must be invariant with respect to the kinematic inputs. With this formulation, it is possible to set up sufficient conditions that guarantee asymptotic stability on the remaining part of the configuration manifold
- A nonlinear **passivity based controller** is developed for controlling systems with kinematic inputs. The controller is applied to the AV, and it is general enough to handle both crop row tracking and asymptotic stabilization of the AV. Especially the ability to asymptotically stabilize the system is a useful feature, as this is a general problem for nonholonomic systems. The controller applied to the AV is further improved by two extensions. The first is integral action to eliminate the effect of external disturbances, and the second is an adaptive damping scheme that enables velocity control when path tracking. Both extensions are given in the framework of Hamiltonian systems
- A **proof of concept** is presented by tests of the controller on the physical vehicle

Some of the results have already been presented and published in three conference papers and one transaction paper that has been submitted for publication. Sørensen [61] focused on deriving a suitable model of the AV, and a path tracking controller was introduced using APFs. In [62] the kinematic inputs were introduced, and a controller

for asymptotic stabilization of a class of nonholonomic Hamiltonian systems with kinematic inputs was presented. The controller was applied to the AV, and a successful test on the physical system was also presented. Nielsen et al. [44] presents some conceptual pointers for design of embedded software for autonomous vehicles. Sørensen et al. [63] presents some additional results on a general nonholonomic Hamiltonian system and is a summary of the results presented in the thesis at hand.

1.5 Thesis Overview

Chapter 2. Hardware and Software on the AV. This chapter gives a description of the hardware and software on the AV. It can be read independently from the rest of the thesis.

Chapter 3. Lagrange Model of the AV. Before turning to the problem of controlling the AV a full dynamic model of the vehicle is developed in this chapter. The model is developed using the Lagrangian equations for nonholonomic electro-mechanical systems.

Chapter 4. The Hamiltonian Formulation and Model Reduction. This chapter deals with a general class of nonholonomic Hamiltonian systems are augmented with properly defined *kinematic inputs*. The chapter describes a method of eliminating the Lagrange multipliers and thereby reducing the system.

Chapter 5. Reduction and Validation of the AV Model. The method described in chapter 4 is applied to the model of the AV to arrive at a suitable model for control. This model is then verified based on measurements from physical test runs with the vehicle.

Chapter 6. Feedback Control of Systems with Kinematic Inputs. This chapter is devoted to developing a controller for the general class of nonholonomic systems with kinematic inputs. The chapter focuses on energy shaping and damping injection to achieve asymptotic stabilization of the system.

Chapter 7. Feedback Control of the AV. The controller developed in chapter 6 is applied to the AV model. Several extension to the controller is also given, such as general crop row tracking, crop row tracking with constant velocity, and integral action.

Chapter 8. Physical Tests. This chapter illustrates the results from several physical tests with the AV and the applied controller.

Chapter 9. Conclusions and Future Work. Conclusion and final remarks.

CHAPTER 2

HARDWARE AND SOFTWARE ON THE AV

As mentioned earlier the main source of interest in this thesis is the development of a working prototype of an AV, and this chapter describes some of the hardware and software issues involved in this task. The prototype is to be used as a development platform, and this has influenced both the mechanical design and the design of software systems. The AV is constructed with a high degree of mobility by letting all of its wheels be steerable and drivable, and this 'over' actuation of the vehicle has two important advantages: it enables the AV to minimize the damage to the crops in the field, and it gives the control designer an opportunity to test many different control strategies, as the AV is able to emulate different steering methods, such as all wheel steer, front axle steer, skid steer, etc. To relieve the control designer from having to know every little intrinsic detail about the hardware on the AV a graphical control design environment has also been developed. It includes an interface to the sensors and actuators on the AV, and an automated code generation feature that enables the control designer to build a feedback controller in a graphical environment and then convert it to an executable program for the AV.

2.1 Hardware

As mentioned in the introduction, the idea behind the API project is to have a small AV drive along crop rows in a field, doing as little damage to the crops as possible. The AV must be able to traverse many kinds of fields with crops at different growth stages, but a typical working environment is a crop field with newly germinated vegetation. At this stage there is very little plant material to bind the soil, and just a little rainfall will make the soil soft and muddy. The AV must therefore be equipped with a traction and propulsion system that is powerful enough to drive the AV in this environment.

The crop and weed registration also puts constraints on the size and mechanical construction of the AV. As the acronym implies, the API is a *platform* capable of carrying a range of small implements into the field. These implements include, but are not limited to, passive sensors like CCD and infrared cameras, or mechanical implements like weeders and soil samplers. The AV must hence be constructed with enough room, clearance, and mounting options for this variety of implements. A high clearance is also necessary when the AV has to operate in a field with a late crop growth state.

The AV also needs a high degree of maneuverability to be able to navigate a field without damaging the crops. Most of the time the AV drives along a crop row, and this is a relatively simple maneuver that can be accomplished by a simple car-like steering. But the AV needs to do more complicated maneuver as well; it has to be able to change crop rows and to align itself to a given crop row, both in-field and when reaching the end of a crop row. Car-like steering is not a good choice in these situations as figure 2.1 shows. The rear wheel tread deviates from the front wheel tread as soon as the vehicle starts turning, and this results in crop damage by all four wheels while crossing the row. To minimize the damage, the AV must use a different steering approach with a larger degree of maneuverability.

The most important requirements to the mechanical construction are summarized below. The AV must have:

1. Good traction
2. A powerful propulsion system
3. A large degree of maneuverability
4. High clearance
5. Mounting options for implements
6. Suitable for control inputs

Figure 2.1: *In-field crop row changing with car-like steering. The triangles marks areas of crop damage*

The last requirement should be evident in the design of any autonomous vehicle; there must be a well defined interface between the physical components of the AV and the control system, and the actuation of the system must render the system fully controllable.

2.1.1 AV Frame

To take the third, fourth, and fifth requirement into account and to give the AV a good loading capability, it has been designed as a four wheel vehicle with the wheels placed at the corners of a $1 \times 1\mathrm{m}$ square. This configuration enables the AV to drive along crop rows planted with a relative distance of one meter (or an integer fraction thereof). Figure 2.2 depicts a side view of the AV, and it shows the large working area and high clearance at the center of the vehicle together with the triangular shaped boom for implement mounting. The symmetric structure of the wheel configuration makes the AV stable under varying load conditions, because the weight of an implement, if placed at the center, will distribute evenly onto all four wheels.

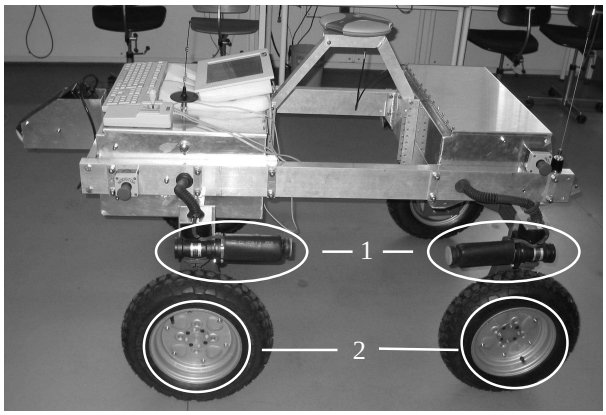


Figure 2.2: *The AV right side view. 1: DC steering motors, 2: DC hub drive motors*

2.1.2 Steering System

As already noted, car-like steering is not the prime choice of steering configuration for the AV if the crop damage should be kept at a minimum. If, on the other hand, it is possible to steer both the front and rear wheels of the vehicle in a synchronous manner the wheel tread deviation between front and rear wheels change, and a crop changing maneuver would look like the one shown in figure 2.3. The deviation between front and rear wheel tread has been brought to zero, and sharper turns can generally be carried out with this steering configuration. Only having two treads over the crop rows, while also intersecting the crop rows at a sharper angle, clearly reduces the damage.

The maneuver illustrated on figure 2.3 can be executed by any vehicle where the *instantaneous center of rotation* (ICR) is located on a line passing through the center of the vehicle, see figure 2.4. The ICR is defined as the point where the four lines perpendicular to the wheels meet. The maneuverability of a front+rear wheel steered vehicle has not really been increased when compared with the car-like steering; the vehicle can still only drive back and forth at an arbitrary rotation rate. The front+rear wheel steered vehicle is not able to drive directly sideways, which is a desirable maneuver when the AV has to align itself to a crop row. Of cause, any car-like or front+rear wheel steered vehicle can do sideways motion by a sequence of forward and backward motions – much like parallel parking a car. This is not an ideal motion in a crop field though, where the crops are planted very close together, and there is no room for these maneuvers. Pure sideways movement and greater maneuverability can be achieved by allowing all four wheels to be steered independently. This allows the ICR to lie anywhere in $\mathbb{R}^2$, hence allowing the vehicle to rotate around any point and to drive sideways. Rotation about an arbitrary ICR is illustrated in figure 2.5. Sideways driving can then be achieved by

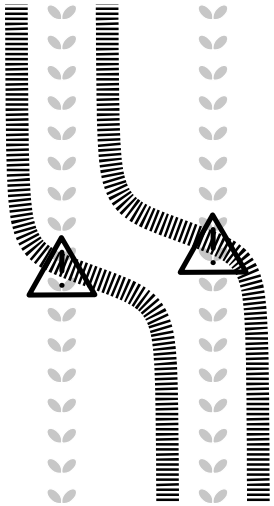


Figure 2.3: *Crop row changing with front+rear wheel steer*

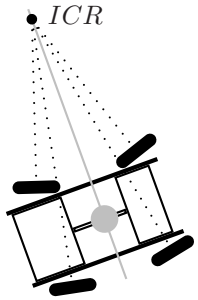


Figure 2.4: *ICR of a front+rear wheel steer*

letting the ICR tend to infinity in the direction perpendicular to the desired sideways motion. This steering configuration will henceforth be called four wheel steer (4WS).

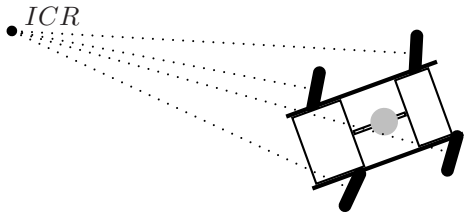


Figure 2.5: *ICR of a full four wheel steer*

To achieve 4WS on the AV it is equipped with four steering actuators marked with (1) on figure 2.2; one for each wheel. The steering actuators are four DC-motors that enable each wheel suspension to be rotated 360° through a worm gear. Table 2.1 shows some of the characteristics of the steering actuators. The 4WS has the added benefit that it can be used to mimic any other steering method (car-like, front+rear, skid steer, etc.), and this feature is useful when testing and comparing motion control algorithms for different wheel steering configurations.

Manufacturer	maxon motor
Model	F 2260 (885)
Type	Graphite brushes DC motor
Stall torque	$1.67\mathrm{Nm}$
Assigned power rating	$80\mathrm{W}$
Maximum efficiency	80%
Website	`http://www.maxonmotor.com`

Table 2.1: *The steering motors*

2.1.3 Propulsion System

For driving in a muddy field with little traction it is desirable to have a four wheel drive (4WD) propulsion system on the AV. The 4WS of the AV puts constraints on the design of the propulsion system though. Because of the independent steering of each wheel, it is not feasible to design a transmission system going from a central propulsion engine and passing through each steerable wheel suspension. This would result in a complex mechanical construction of the suspensions, and at the same time it would

most likely result in a propulsion distribution that is difficult to control. A different approach is to mount four independent propulsion motors directly in each wheel, thereby eliminating the need for a central transmission system all together. This is achieved on the AV with the use of four brushless DC hub motors; one mounted in each wheel, marked (2) on figure 2.2. The motors are custom built for the AV by Heinzmann GmbH (`http://heinzmann.de`). Unfortunately, the customization means that very little is known of the electrical and mechanical properties of the motors, but they are assumed to exhibit linear behavior. The four hub motors allows us to consider each wheel as a self contained mechanical subsystem in the sense that only an electrical power and signal connection needs to pass through the wheel suspension, thereby greatly decreasing the mechanical complexity of the overall system. The system is also simplified, from a control engineering point of view, by the similarity of each wheel set and the useful properties of having to control simple DC motors instead of, for example, a central fuel based propulsion system.

2.1.4 Sensors

The 4WD-4WS structure of the AV results in a highly actuated and fully controllable system, but in order to be fully observable as well, it is important to identify the information needed to describe the instantaneous state of the AV and its surroundings. First of all, the absolute position and orientation is needed to relate to the way-points received from the base station. Knowledge of their time derivatives, i.e., the velocity and rotation rate, is also imperative when it comes to good motion control of the AV. Each particular crop field might not be completely horizontal, so pitch and roll information is also useful if any compensating for the effect of gravity is needed. All this information is related to the global state of the AV, but some local information on actuator states is also necessary. For steering the AV, information on the angular position of each steering motor is needed. And knowing the angular velocity of each individual steering and propulsion motor is useful for good low level motor control. To navigate along crop rows, the AV also needs information on the position and orientation of a target crop row. This is most likely not available from the base station (depending on what information the farmer collected when sowing the crop), so the AV needs some other method of obtaining the information.

The necessary information is:

- The two dimensional position in world coordinates
- The two dimensional velocity in world coordinates
- Heading, pitch, and roll of the AV

- Rotation rate of the AV
- Angular velocity of each propulsion motor
- Angular position and velocity of each of the four steering motor
- Location of a nearby crop row relative to the AV

This information is imperative for good control of the AV, and it must be available by direct measurements or through an observer. To obtain the information, the following range of sensors is equipped on the AV:

- A **GPS receiver** for measuring position and velocity of the AV in world coordinates
- A magnetic **compass and tilt sensors** to measure the heading, pitch, and roll
- A single axis **fiber optic gyro** for measuring the rotation rate about an axis perpendicular to the ground
- **Tachometers** in each propulsion motor for measuring the angular velocity
- **Encoders** in each steering motor for measuring the angular position
- A **ground speed radar** for measuring the forward velocity of the AV relative to the ground
- A **crop row guidance camera** for measuring the offset and direction of a crop row in front of the AV

Figures 2.6 to 2.8 shows the mounting locations of the sensors.

GPS Receiver

The global positioning system equipped on the AV is a Real Time Kinematics - Global Positioning System (RTK-GPS). The RTK-GPS receiver is used to generate a position and velocity solution in world coordinates. The solution is based on signals received from the GPS (and GLONASS when available) satellites together with correctional information via a radio modem from a reference station. The corrections are based on information in the signals from the satellites and phase measurements on the modulated signals themselves. By including phase measurements in the corrections, it is possible to achieve a precision of a couple of centimeters. More specifications are summarized in table 2.2. The high precision is important for the quality and repeatability of the

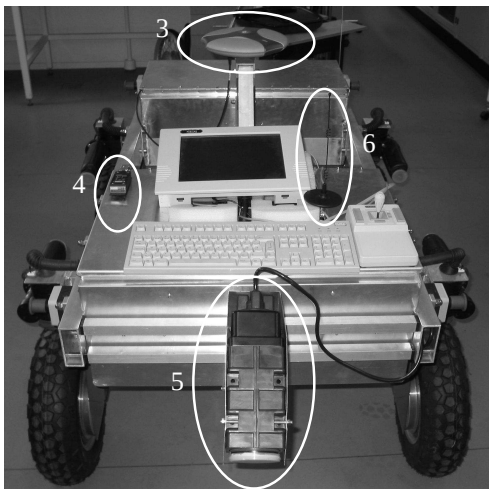


Figure 2.6: *The AV rear view. 3: GPS antenna, 4: Magnetometer and tilt sensors, 5: Ground speed radar, 6: WLAN antenna. The joystick, monitor, and keyboard are used for interfacing the on-board computer system*

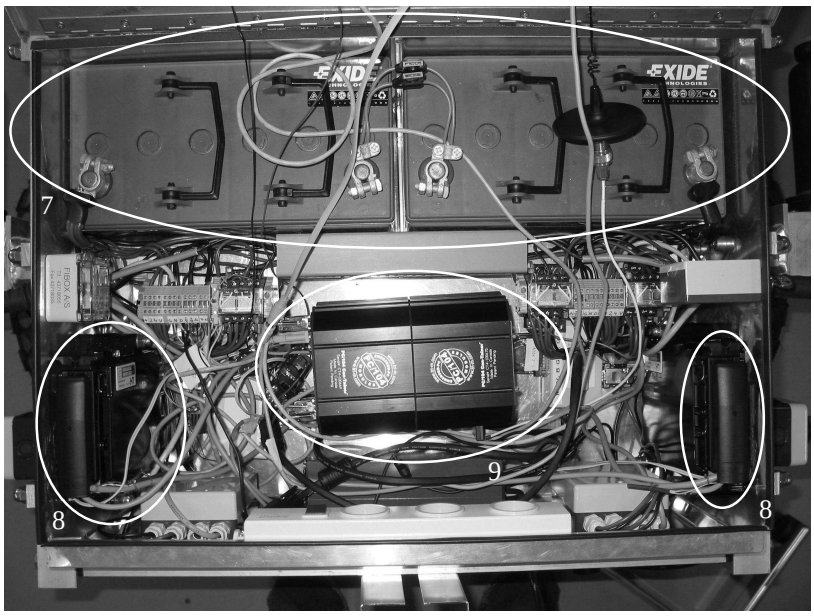


Figure 2.7: *The AV rear instrument box. 7: Power supply, 8: LH Agro embedded computers, 9: PC/104 main computer*

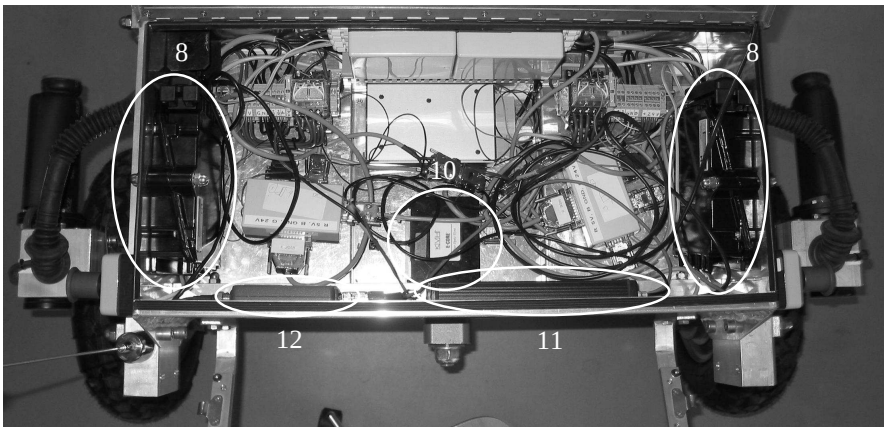


Figure 2.8: *The AV front instrument box. 8: LH Agro embedded computers, 10: 1-axis fiber optic gyro, 11: RTK-GPS receiver, 12: GPS radio modem*

field mapping. Repeatability refers to the ability to return to the exact same location on a crop field several times, and good repeatability is useful when the farmer wants to see the temporal variation on a field or check the long term effects of his applied crop treatment.

Manufacturer	JAVAD
Model	Legacy-E
Type	40-channels dual frequency GPS+GLONASS receiver
RTK accuracy (horizontal)	15mm
Maximum sampling rate	10Hz
Website	`http://www.javad.com`

Table 2.2: *The RTK-GPS*

The GPS receiver is unfortunately a rather unreliable sensor. The receiver is only able to return a position if the antenna has line-of-sight with at least four satellites. This is usually not a problem in clear weather on an open field, but as soon as the receiver is in the vicinity of solid objects like buildings or trees, the positioning solution is often lost; even thick clouds can severely degrade the performance of the receiver. This problem can partly be solved by relying on the rest of the sensors in a *dead-reckoning* configuration during periods of GPS drop-outs. Dead-reckoning is a term used when estimating the position of a system based on integration of local measurements only. If the steering position and angular velocity of each wheel and the orientation of the body of the AV is known it is possible to estimate the trajectory of the AV from the point where the GPS

solution drops out. In the ideal situation, the dead-reckoning estimate would coincide with the absolute position of the AV, but a dead-reckoning observer is by construction a divergent filter. Uncertainties in the system, such as wheel slip or biased measurements, will eventually make the dead-reckoning position deviate from the actual position. This is illustrated in figure 2.9.

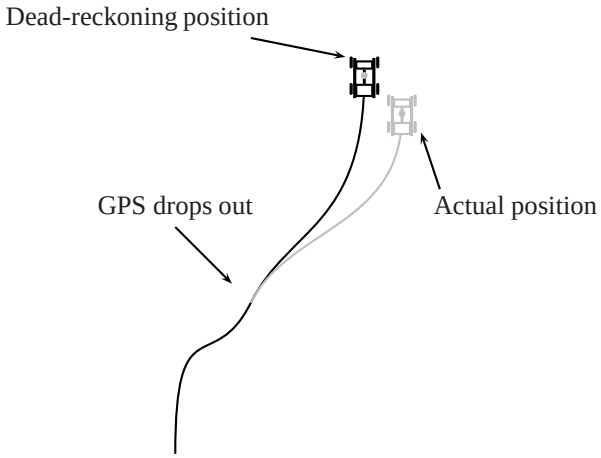


Figure 2.9: *Deviation in dead-reckoning*

During short drop-outs of the GPS solution, the availability of the AV can be greatly increased by applying dead-reckoning, but if the drop-out is long, the dead-reckoning will eventually become too unreliable. The AV will then have to stop until the GPS receiver has reestablished a positioning solution. The use of the rest of the sensors is therefore twofold; they are used both for the full state estimation, when the GPS is operational, and to help estimate the position of the AV when dead-reckoning.

Compass and Tilt Sensors

To obtain the global attitude (heading, pitch, and roll), the AV is equipped with compass and tilt sensors combined in a single housing. The housing comprises three magneto resistive magnetic sensors and a two-axis liquid filled tilt sensor to produce tilt compensated attitude measurements. More specifications can be found in table 2.3. Because of the multitude of magnetic sensors, the compass can somewhat compensate for magnetic interference from ferrous metallic objects and stray magnetic fields. This compensation is by no means ideal, and the output heading from the compass is unreliable. It can,

however, be combined with other measurements to yield a good estimate of the heading.

Manufacturer	Honeywell
Model	HMR3000
Type	Digital compass module
Heading accuracy	$0.5°$
Pitch/roll accuracy	$\pm 4°$
Maximum sampling rate	20Hz
Website	`http://www.ssec.honeywell.com`

Table 2.3: *The compass and tilt sensor*

Fiber Optic Gyro

The primary output of the 1-axis fiber optic gyro is the angular velocity about an axis perpendicular to the ground. See table 2.4 for more specifications. The output is directly related to the rate of change of the heading. The heading readings from the compass can therefore be reconstructed by integrating the measurements from the gyro while taking pitch and roll into account, but due to a temperature varying bias in the gyro, the reconstructed heading will diverge from the actual heading if no compensation it present. To solve this problem, the readings from the gyro and compass can be combined to give a more reliable estimate on the heading.

Manufacturer	KVH Industries
Model	E•Core 2000 (RD2100)
Type	Single-axis fiber optic gyro
Accuracy	$0.014°/\mathrm{s}$
Bias stability over full temperature range	$0.4°/\mathrm{s}$
Maximum sampling rate	10Hz
Website	`http://www.kvh.com`

Table 2.4: *The fiber optic gyro*

Propulsion Motor Tachometers

Each propulsion motor is equipped internally with a tachometer measuring the angular velocity of each wheel relative to the wheel suspension. The tachometer readings are

used both for low level control of each actuator and for dead-reckoning where the distance traveled by each wheel can be calculated by integrating the tachometer readings.

Steering Motor Encoders

To measure the steering angles of the wheels each steering motor is equipped with an encoder. See table 2.5 for specifications. Combining these measurements with the measurements from the tachometers enables the calculation of the direction of movement of the AV relative to itself. The encoders are also used for low level control of the steering actuators.

Manufacturer	maxon motor
Model	HEDL 5540
Type	Digital encoder
Accuracy	$0.0144°$/pulse (through worm gear)
Website	`http://www.maxonmotor.com`

Table 2.5: *The steering motor encoders*

Ground Speed Radar

Using the Doppler effect the ground speed radar mounted on the rear of the AV is able to measure the velocity relative to the ground. See table 2.6 for specifications. The use of the radar is limited, because it only measures the absolute longitudinal component of the velocity. Hence, the radar output is zero if the AV is driving sideways. The primary reason for having the radar is to be able to estimate the occurrence of wheel slip by comparing the output of the radar with the output of the four tachometers in the propulsion motors.

Manufacturer	DICKEY-john
Model	RVSII
Type	Ground speed radar
Accuracy	5% $(0.53 - 3.2\mathrm{km/h})$ 3% $(3.2 - 7.0\mathrm{km/h})$
Website	`http://www.dickey-john.com`

Table 2.6: *The ground speed radar*

Crop Row Guidance Camera

To obtain local crop row information the AV is equipped with a crop row guidance camera. See figure 2.10 and table 2.7 for specifications. The camera uses a vision system to output the lateral offset of a crop row in its field of vision.

Figure 2.10: *ECO-DAN crop row guidance camera*

Manufacturer	ECO-DAN
Model	11-410-02-01
Type	Single plant camera
Accuracy	$1\text{mm} \pm 0.5\text{mm}$
Maximum sampling rate	$25Hz$
Website	`http://www.eco-dan.com`

Table 2.7: *The crop row guidance camera*

2.1.5 Computer System

To interface the different hardware components the AV is equipped with a computer system that also handles internal coordination and external communication tasks. The overall purpose of the computer system is to execute the motion control algorithms needed for autonomous driving. Even though no control algorithm has been introduced yet it is possible to identify two distinct levels of control in the AV [44].

Low level motor control of each wheel set. The purpose of this control is to achieve precise position, velocity, or torque control of each motor on the AV. The low level control closes loops around the motor hardware, and it should be run at a high sampling rate at an order of magnitude of 10^2Hz.

High level motion control of the AV body. This control handles the mutual coordination of all four wheel sets and the generation of position, velocity, or torque references for the low level controllers. The high level controller closes the loop around the AV body, which has a relatively slow response compared with the response of the motors. Hence, the sampling rate of this control need not be as fast as the motor control and is of an order of magnitude of 10^1Hz.

Not only do the two levels of control put different timing requirements on the computer system; the complexity of the controllers also puts requirements on the computational power. The low level motor control is expected to consist of simple, possibly linear controllers, while the high level motion control is expected to be more complex. Because of this layered control structure, the computer system on the AV is distributed over five separate computers:

- 4 small embedded computers for low level control. Marked (8) on figures 2.7 and 2.8
- 1 powerful main computer for high level control and data communication with the base station. Marked (9) on 2.7

The Four Embedded Computers

The four embedded computers are manufactured by LH-Agro, and they are small rugged computers designed specifically for controlling farming applications. They run a real-time operating system and are equipped with several I/O ports for interfacing sensors and actuators. They also include a CAN-bus interface for real-time external communication. The primary task of the LH-Agro computers is to implement the fast local loops around the motors, and each LH-Agro controls one propulsion and one steering motor. Additional specifications can be found in table 2.8.

The Main Computer

The main computer on the AV is a standard PC/104 industrial computer running Debian Linux. This computer runs high level motion planning and coordination tasks, and it interfaces a subset of the sensors. The PC/104 stack comprises a main CPU module, a power supply module, a RS-232 four port module for sensor communication, a PCMCIA module for a WLAN adapter, and a CAN bus module for communicating with the LH-Agros. Additional specifications can be found in table 2.9.

Manufacturer	LH-Agro
Model	IC28
Processor	Infinion C167CR
Input ports	16 digital 2 analog
Output ports	6 digital 4 PWM
Communication ports	1 CAN 1 RS-232
Website	`http://www.lh-agro.com`

Table 2.8: *The LH-Agro embedded computer*

Module	CPU
Manufacturer	ICOP
Model	ICOP-6070
Website	`http://www.icop.com.tw`
Module	RS-232
Manufacturer	ICOP
Model	ICOP-1800
Website	`http://www.icop.com.tw`
Module	PCMCIA
Model	PCM-210A
Module	CAN
Model	PCM-3680

Table 2.9: *The PC/104 stack*

2.1.6 Hardware Communication Structure

Figure 2.11 shows an overview of the hardware interconnections. Each sensor is connected to the computer on which their output is first needed. The tachometers and encoders are connected to the LH-Agros, where they are used for the low level control, and except for the ground speed radar, the rest of the sensors are directly connected to the PC/104. (The physical properties of the radar signal necessitates the connection to one of the LH-Agros, but the data is passed unused to the PC/104 through the CAN-bus.) The joystick is used for manual driving.

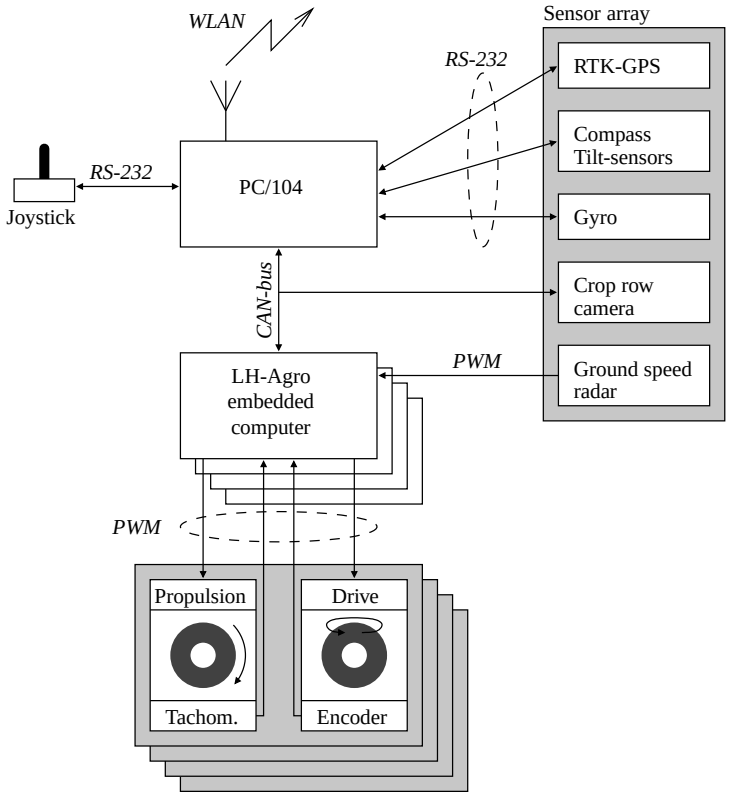


Figure 2.11: *Hardware communication structure on the AV*

2.2 Software

Once the hardware is mounted and connected to the computer system the next step is to combine the data flowing through the system in a good navigation and control scheme for the AV. The software handles every aspect of communication and data handling on the AV; from interfacing sensors and actuators to scheduled execution of high level motion control algorithms. The AV is intended to be a development platform for testing motion control algorithms, and from a control engineering point of view, many of the software aspects are not really of interest; the control engineer basically needs only to be served the data that is available from the sensors, act on this data, and then generate control signals for the actuators. How the system actually interfaces the sensors and actuators is of less importance and should be encapsulated in a intuitive control development environment. This section is focused on the development of such an environment for testing motion control algorithms.

2.2.1 Overview of Major Software Components

In the previous section two levels of control were identified. One was the low level control of each motor, which is to be executed on the LH-Agros, and one was the high level motion control of the entire AV, which is to be executed on the PC/104. It is assumed that good low level controllers can be designed and implemented on the LH-Agros once and for all, and the need for future changes in the software is minimal. The software on the PC/104, on the other hand, is assumed to be of a more varying nature, since it should be possible to test many different motion control, path following, and trajectory tracking algorithms.

Figure 2.12 identifies the major software components on the AV and their mutual data dependencies. Note that it is very similar to the hardware interconnection in figure 2.11, because of the distribution of computer systems. At the heart of the structure is the PC/104 software that coordinates the motion of the AV. In order to do this, it has to have access to the sensors, the joystick, and the LH-Agros through the CAN bus. The PC/104 software interfaces the sensors, and it closes the motion control loop by sending reference signals to the motors through the LH-Agro software. When driving autonomously, it makes sense to have the ability to interface the AV remotely, hence the external PC. From the external PC it should be possible to monitor the motion control software, transmit new parameters, and start and stop the AV.

The data flowing between the components are as follows.

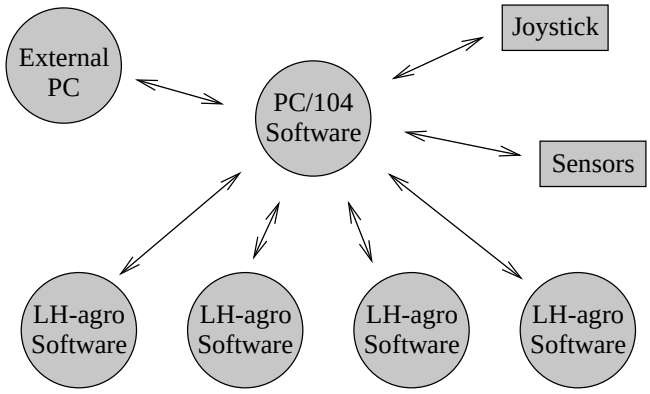


Figure 2.12: *Overview of software structure*

PC/104 ⟷ LH-Agros The LH-Agros receive position, velocity, or torque references for the eight motors, and start and stop commands from the PC/104. They send back tachometer, encoder, and ground speed radar readings to the PC/104. Any asynchronous parameter changes for the LH-Agros software can also be sent by the PC/104.

PC/104 ⟷ Sensors At this interface, data is sent from the GPS receiver, gyro, compass, and crop row guidance camera to the PC/104, which in return sends any setup information or polling requests to the sensors.

PC/104 ⟷ Joystick The joystick is only to be used for manual driving, and under these circumstances the state of the joystick is sent to the PC/104.

PC/104 ⟷ External PC Supervisory information on the state of the AV is sent from the PC/104 to an external PC. In return, the external PC is able to send new parameters for the controllers on the PC/104 and LH-Agros. Asynchronous commands, such as starting or stopping a controller and changing between manual and autonomous mode, can also be sent by the External PC.

Getting data to flow in and out of each subsystem is one thing, but reacting on the data in an advanced motion control scheme is a completely different issue. For the control engineer the latter is, by far, the most interesting, but testing a developed control scheme is difficult if there is no well defined data flow infrastructure. To relieve the control engineer of knowing every little hardware and software detail on the AV and to generally

ease the implementation of a designed controller, a graphical control development environment that encapsulates and hides all data communication on the AV is a great help. The desired features of a development environment for the AV is:

- A graphical user interface that gives easy access to sensors and actuators
- Automatic code generation and compilation based on the graphically designed controller
- On-line graphical data representation and monitoring of the closed loop system
- On-line changing of controller parameters

All of these features are embedded in the mathematical software suite Matlab/Simulink through its toolbox Real-Time Workshop. They can be applied to the AV by tailoring the PC/104 software to interface Real-Time Workshop.

2.2.2 Real-Time Workshop

The Real-Time Workshop (RTW) is a comprehensive Matlab/Simulink toolbox that is able to generate and compile code based on Simulink block diagrams. It is mainly used for rapid prototyping on a variety of different operating systems, and it can be extended to fit more specialized systems, like the AV, as well. The source code generated by RTW is in C and can be extended by any additional custom written C source code. This feature enables RTW to communicate with any hardware or software component, as long as it is accessible via C. RTW can generate generic executable code for the Linux operating system, but without extensions it has limited hardware interface capability. Since there is a lot of specialized hardware on the AV, several issues has to be resolved before enabling the control engineer to design a controller through Simulink and generate the appropriate code automatically. The major issues are:

- RTW does not include an interface to the sensors, actuators, and CAN bus on the AV, and it must hence be extended in order to communicate with these components
- When generating code to the Linux operating system, RTW does not take timing and scheduling into account. The resulting executable will run ‘as fast as possible’, not obeying the desired sampling frequency

Fortunately, the extensibility of RTW makes these issues resolvable. The first issue can be addresses directly by writing custom S-functions for inclusion in the Simulink diagram. An S-function is a peace of C-code that defines how external components should

be interfaced in a format that RTW understands. The second issue is somewhat more difficult to resolve. To understand this, it is necessary to look closer at how RTW generates the code. Figure 2.13 shows an overview of the code generation and compilation procedure of RTW. When the *build* button in Simulink is clicked RTW generates pre-

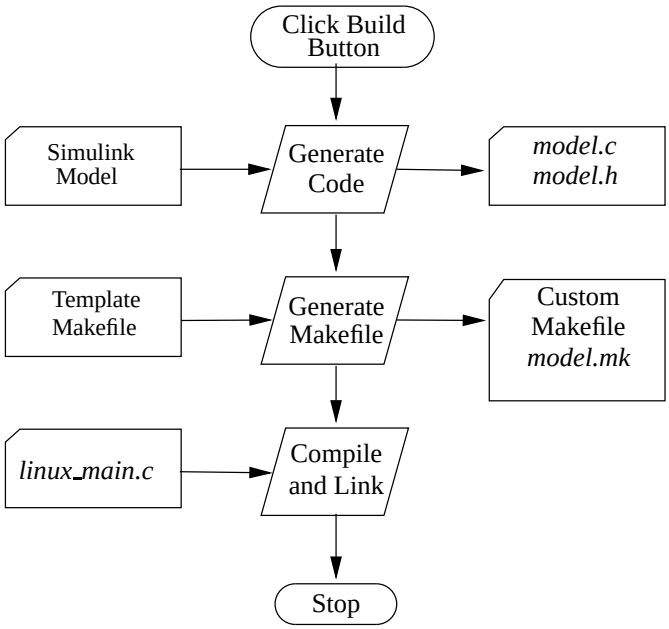


Figure 2.13: *Code generation and compilation procedure of Real-Time Workshop*

liminary code from the Simulink model, including any custom made S-functions. The generated code is a basically a group of C-functions describing the model. This includes functions for calculating the model states at the next time instance, retrieving the output, and so forth. How, and in which order, these functions are called is not included yet. The next step is to generate a custom makefile for the target executable. This is where RTW is told which compiler to use and any options related to the compilation. The template makefile also defines which file should be used to execute the code that was generated in the first step. The file is typically a C source file that includes the **main** function in which the functions from the code generation step are called in the right sequence. In this example the file is called `linux_main.c`, and it is included when compiling and linking the executable. The `linux_main.c` file also determines time of execution of each iteration step of the model, and any operating system specific timing and scheduling should be included in this file in order to iterate the model at the correct sampling interval.

In the `linux_main.c` file shipped with RTW there is no timing in the execution of the model; the next iteration of the model is executed as soon as the previous one has finished, resulting in an executable model that runs 'as fast as possible', thereby not providing the correct sampling period. The problem has been resolved by modifying this file and inserting a timer that triggers the next iteration of the model when the correct amount of time has elapsed.

The executable file generated by RTW includes one more very useful element. By default, RTW facilitates communication with the running executable through what is referred to as *External Mode*; typically via a TCP/IP connection. When the executable is started it registers itself on the network stack on the machine and starts listening for incoming connections. Anyone with the same Simulink model and a running Matlab/Simulink is then able to connect to the executable, provided that they are on the same network. When a connection is made, it is then possible to view on-line data from within Simulink and to upload new parameters to the running model.

When RTW is fully integrated with the PC/104 software it provides the following intuitive work flow for the control engineer:

1. The engineer starts by designing and building a controller in Simulink. The design would typically be based on a mathematical model of the AV and knowledge of what sensors and actuators are present in the physical system. How sensors and actuators are interfaced is not of importance, since this is hidden behind the user interface and made available through appropriate Simulink blocks
2. The next step is to test the controller on a simulation model of the AV. Figure 2.14 shows how this would appear to the engineer for a simple SISO system

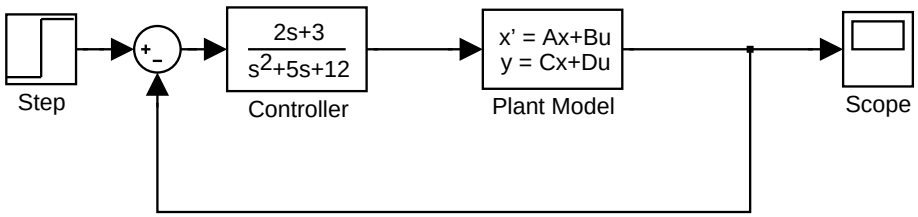


Figure 2.14: *Simple example of the Simulink user interface when simulating*

3. When design requirements have been met the hardware is introduced into the loop by exchanging the simulation model with a set of S-function blocks, see figure 2.15. Executable code is then automatically generated, built, and downloaded through the wireless network to the PC/104

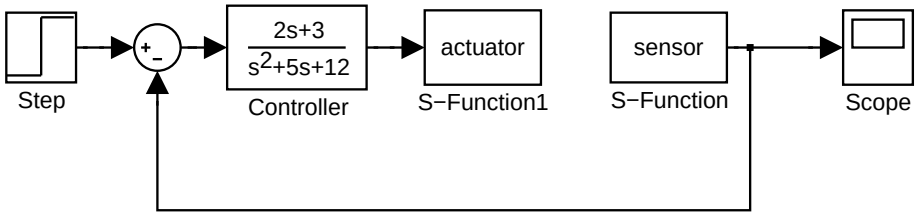


Figure 2.15: *Simple example of introducing hardware in the loop in Simulink*

4. The executable code is started and the engineer is able to connect to it, monitor the system, and upload new parameters on-line from within Simulink running on the external PC

2.2.3 PC/104 Software

The S-functions introduced in the previous subsection allows RTW to interface any piece of hardware or software that is accessible through C. For controlling the AV the RTW executable must know how to access the sensors, the CAN bus, and the joystick. To make these subsystems available, the following three drivers are identified:

1. A **sensor driver** that handles the RS-232 interfaces to the GPS receiver, the gyro, and the compass. The sensor driver must wait for incoming raw data from the sensors, interpret it, and make it available for the RTW executable. The sensor driver must also handle any configuration of the sensors at start up
2. A **CAN bus driver** that handles all data communication to and from the CAN bus. It must transmit references generated by the RTW executable and listen to incoming data from tachometers, encoders, and crop row camera to make it available for the RTW executable
3. A **joystick driver** that listens for incoming data from the joystick, interprets this data and makes it available for the RTW executable

The inter process communication between the drivers and the RTW executable can be handled in different ways. One option is to embed every part of the drivers in the S-functions used by RTW. This solution will ultimately mean that the system will only be able to run with a working RTW executable. Any external non-RTW programs will then be unable to communicate with the hardware unless going through RTW. Even though Simulink and RTW are powerful tools they may not always be the most useful

ones. When writing simple programs for the AV, such as testing or debugging programs, it may be more convenient to access the drivers directly through a stand alone program. To provide for a more flexible software structure the drivers have been split into individual programs; all communicating through a segment of shared memory. This allows any other program (written in a language that supports communication with shared memory) to access sensors and actuators. A schematic illustration of the shared memory communication is shown in figure 2.16. The figure shows the flow of sensor data and actuator

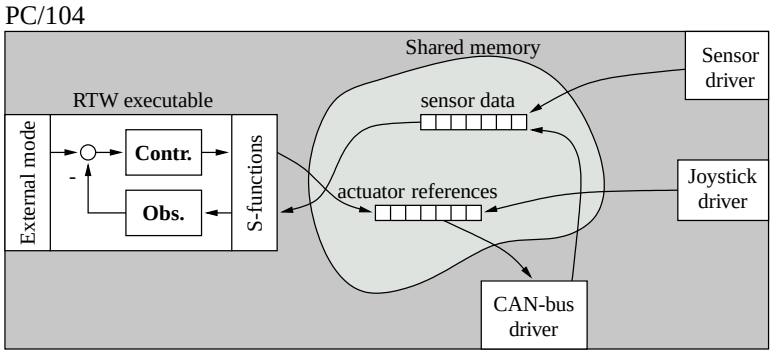
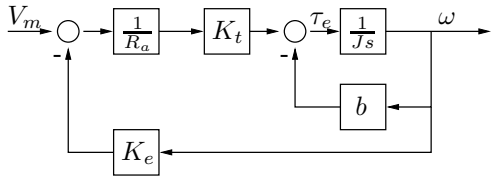


Figure 2.16: *Software components on the PC/104*

references between the drivers and the RTW executable. As the figure shows, both the RTW executable and the joystick generates actuator references, but they should not run simultaneously. In general, the joystick is only used for manual driving, while the RTW executable is used for autonomous driving. Both are depicted in the figure to illustrate how the use of shared memory enables direct manual driving of the AV while bypassing the RTW executable.

2.2.4 LH-Agros Software

When the actuator references have been generated by the software on the PC/104 they are transmitted via the CAN bus to the four LH-Agros embedded computers. The LH-Agro software then closes the fast local loops around each actuator and returns measurements from encoders and tachometers to the PC/104. The steering and propulsion motors both exhibit simple first order linear behavior [7], and the model used to describe them is the basic first order linear model of a DC motor as shown on figure 2.17. The parameters associated with the unloaded steering and propulsion motors were also identified in [7]. They are summarized in table 2.10.



V_m	input voltage
τ_e	applied torque
ω	angular velocity
R_a	armature resistance
K_t	torque constant
K_e	electromotive force constant
J	moment of inertia of anchor
b	coefficient of kinetic friction

Figure 2.17: *First order motor model*

Parameter		Steer	Propulsion
R_a	[Ω]	1.43	0.017
K_t/K_e	[Nm/A] / [Vs]	0.1	0.31
J	[kg m^2]	$3.1 \cdot 10^{-4}$	0.18
b	[Nms]	0.003	0.37

Table 2.10: *Parameters associated with the motors*

The Steering Control Loop

To encapsulate the internal dynamics of the steering motors individual position and velocity controllers around each motor were designed by Bisgaard et al. [7] and implemented in the LH-Agros. The position controller is a second order lead-lag controller, and it is used to precisely position each steering motor at an angle relative to the wheel suspension. The velocity controller is a pure proportional controller and is used to rotate each steering motor at a given angular velocity. The position and velocity controllers on each wheel are not designed to run simultaneously, and the controller used is determined by the kind of reference transmitted by the PC/104. This structure allows the designer of the high level controller on the PC/104 to focus on generating position or velocity references to the steering actuators, while the actual reference tracking and the underlying dynamics are encapsulated by the LH-Agros.

The Propulsion Control Loop

The custom built propulsion motors are already controlled by velocity controllers embedded inside each motor. According to the manufacturer (Heinzmann GmbH) there is a linear first order behavior from input voltage to angular velocity of the closed loop. Measurements have shown that this claim is valid with the assumption that the controller is just a simple proportional feedback. In this sense, the proportional feedback can be

viewed as a scaling of either K_t, K_e, or R_a, and the closed loop can be viewed as a linear DC motor with parameters shown in table 2.10. The parameters are based on measurements on the closed loop.

The propulsion motors should not be velocity controlled though, because the four velocities of the propulsion motors are not independent – assuming that the AV should be driven with the minimum amount of slippage. Imagine that the AV is driving along a straight line on a flat field. Obviously, the angular velocity of each propulsion motor will be identical if the AV drives without slipping; the velocities of all wheels are determined by the velocity of one. With the same velocity reference a wheel slip may occur if the AV passes a bump or a depression in the field. If the motors are controlled by setting references to the applied torque (τ_e on figure 2.17) instead the inputs are independent, and slippage of the wheels will be reduced by the presence of natural friction forces between wheel and ground. In the proceeding chapters it is assumed that the input to the propulsion motors is the applied torque, where the following simple trick is used to approximate the corresponding input voltage V_m

$$V_m = \frac{R_a}{K_t}\tau_e + \left(K_e + \frac{R_a b'}{K_t}\right)\omega.$$

Note that the value for the coefficient of kinetic friction b in table 5.1 only describes the internal friction of the motor. There are other sources of friction as well, such as ground friction and increased friction in the motor bearings as a result of the load on the wheels. The parameter b' describes the total kinetic friction, but this parameter it is difficult to determine as it largely depends on various ground and soil conditions. No effort will be taken to determine the exact value of b', but an approximate mean value of $K_e + \frac{R_a b'}{K_t}$ is assumed to be known for various environmental conditions.

CHAPTER 3

LAGRANGE MODEL OF THE AV

In the Matlab/Simulink setup the interface to the AV consists of reference signals to the motors and state information from the sensors. For control purposes, and to understand the underlying dynamics of the AV, the connection from input reference signals to the position and velocity of the AV is worth investigating, and in this chapter a mathematical model describing the motion of the AV is introduced. The AV is subject to free rolling and non-slipping constraints that should be accounted for in the model, and it is therefore developed using the Lagrange's equation[1] for nonholonomic systems with some modifications due to the special 4WS structure of the AV. The resulting Lagrangian model consists of a set of second order differential equations describing the motion of the AV on its configuration manifold. On this form the equations are not suitable for control, but they are the starting point for defining the Hamiltonian equivalent in the subsequent chapter. The Hamiltonian equivalent, on the other hand, will be very useful when turning to motion control of the AV.

[1] For an introduction to Lagrange's equation please refer to appendix A.

3.1 Vehicle Definition

The AV consists of a rigid rectangular body frame and four wheels. The wheels are mounted at the four corners of the vehicle and are all both steerable and drivable. The operational environment of the AV is assumed to be a horizontal field with a constant height above sea level; the pitch and roll of the AV is hence negligible. The position and orientation of the vehicle frame is defined on the manifold $\mathcal{M} \subset \mathbb{R}^2 \times \mathbb{S}$, and a point on this manifold is denoted $\chi \in \mathcal{M}$. Consider a coordinate system $\mathcal{N}$ fixed to the earth and with its axes pointing east and north. Then there exist local coordinates $\chi^{\mathcal{N}} = [x_1 \ x_2 \ \theta]^T$ with respect to the $\mathcal{N}$-frame, as shown in figure 3.1. The figure indicates that there also exists a second coordinate frame $\mathcal{B}$ fixed instantaneously at the position and orientation of the AV. A point χ on the manifold can be described in both the $\mathcal{N}$- and the $\mathcal{B}$-frame and are denoted $\chi^{\mathcal{N}}$ and $\chi^{\mathcal{B}}$ respectively. For notational convenience the superscript will often be dropped, and if nothing else is stated, χ is the coordinate representation in the $\mathcal{N}$-frame.

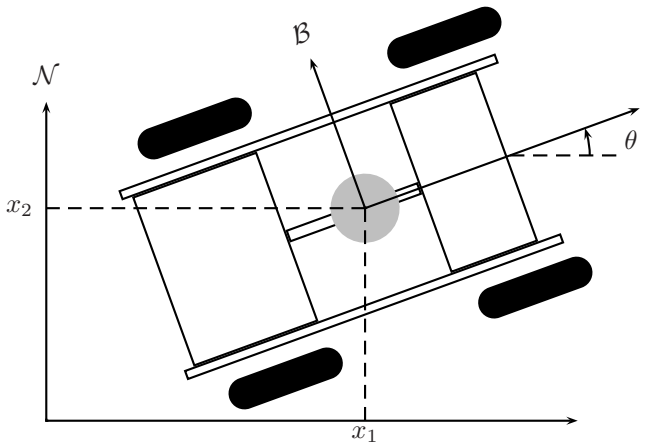


Figure 3.1: *Definition of the vehicle body frame coordinates. $\mathcal{N}$ is an inertial coordinate frame fixed to the earth, and $[x_1 \ x_2]^T$ is the position of the geometric center of the vehicle. The geometric center coincides with the center of the GPS antenna. A second coordinate frame is fixed to the AV at the geometric center and is denoted the $\mathcal{B}$-frame. θ is the rotation between the $\mathcal{B}$- and the $\mathcal{N}$-frame*

The position of the i'th wheel is described by two angles β_i and ϕ_i, as shown in figure 3.2. The figure also shows the torque inputs τ_{β_i} and τ_{ϕ_i}. The three constant parameters γ_i, κ_i, and r_w describe the mounting position and radius of each wheel.

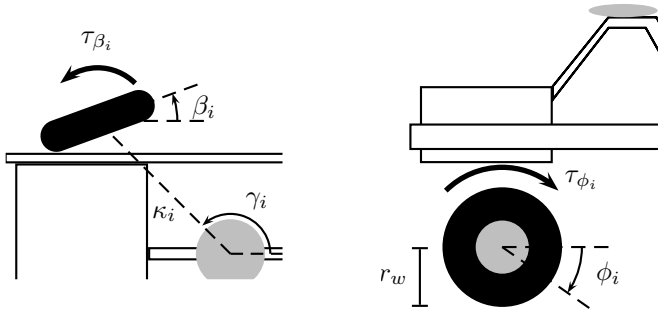


Figure 3.2: *Definition of parameters and coordinates related to the i'th wheel (the figure shows top and side views). β_i defines the steering angle of the wheel relative to the AV frame, and ϕ_i defines the angular position of the propulsion motor. τ_{β_i} and τ_{ϕ_i} are the input torques to the steering and propulsion motors. κ_i and γ_i constitutes a constant polar coordinate pair that defines the mounting position of the wheel relative to the geometric center of the AV. r_w is the radius the wheel*

With these definitions it is possible to define the set q_T of configuration coordinates that is able to completely describe the instantaneous configuration of the AV

$$q_T = \begin{bmatrix} \chi \\ \beta \\ \phi \end{bmatrix}, \tag{3.1}$$

with $\beta = [\beta_1 \ \beta_2 \ \beta_3 \ \beta_4]^T \in \mathbb{S}^4$ and $\phi = [\phi_1 \ \phi_2 \ \phi_3 \ \phi_4]^T \in \mathbb{S}^4$. This is not a minimal set though, and the next section describes constraints that will allow the number of configuration coordinates to be reduced.

The set u_T of input torques to the system is defined as

$$u_T = \begin{bmatrix} \tau_\beta \\ \tau_\phi \end{bmatrix},$$

with $\tau_\beta = [\tau_{\beta_1} \ \tau_{\beta_2} \ \tau_{\beta_3} \ \tau_{\beta_4}]^T \in \mathbb{R}^4$ and $\tau_\phi = [\tau_{\phi_1} \ \tau_{\phi_2} \ \tau_{\phi_3} \ \tau_{\phi_4}]^T \in \mathbb{R}^4$. The alert reader might have noticed that the inputs to the local steering motor control loops are angular velocity or position references and not torques. For now, however, it is assumed that the inputs are all torques, as this will fit into the Lagrangian framework. The consequence of changing the input to velocities will be discussed later in this chapter.

For later use, it is also worthwhile to define the transformation that relates a velocity vector described in coordinates related to the $\mathcal{N}$-frame to the same vector described in coordinates related to the $\mathcal{B}$-frame. If the $\mathcal{B}$-frame is considered to be instantaneously fixed at $[x_1 \ x_2]^T$ and at a rotation θ the transformation is a simple rotation matrix

$$\dot{\chi}^{\mathcal{B}} = R(\theta)\dot{\chi}^{\mathcal{N}} \quad \Rightarrow \quad \dot{\chi}^{\mathcal{N}} = R^T(\theta)\dot{\chi}^{\mathcal{B}},$$

with

$$R(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad R^{-1}(\theta) = R^T(\theta).$$

$\dot{\chi}^{\mathcal{N}}$ and $\dot{\chi}^{\mathcal{B}}$ denotes the same vector expressed in the $\mathcal{N}$- and the $\mathcal{B}$-frame respectively.

Remark. Note that $\dot{\chi}^{\mathcal{N}}$ is *not* expressed in the same coordinates as $\chi^{\mathcal{N}}$. $\dot{\chi}^{\mathcal{N}}$ is expressed in terms of the basis vectors $\partial/\partial x_1$, $\partial/\partial x_2$, and $\partial/\partial\theta$ that span the tangent space of $\mathcal{M}$ at χ. △

3.2 Constraints

The 4WS-4WD structure of the AV might give it a large degree of mobility, but the motion of the AV is still not the motion of an unconstrained rigid body. The wheels of the AV restricts the motion, and the direction of acceleration is always determined by the orientation of the wheels. The multitude of steerable wheels means that they have to be strictly coordinated for sensible driving. Imagine a situation where two wheels are pointing north and two are pointing east. This is not a sensible configuration, and it will ultimately result in sideways dragging of some of the wheels and maybe even physical damage the wheel suspensions. The insensible configuration arises when the instantaneous center of rotation is not uniquely defined. Recall that the ICR is defined as the point of intersection of the four lines perpendicular to the wheels, see figure 3.3.

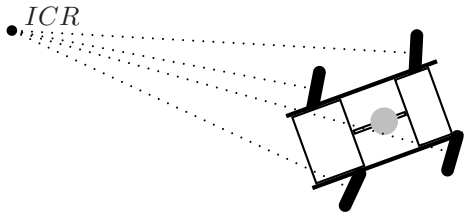


Figure 3.3: *The Instantaneous Center of Rotation*

Only two wheels are needed to uniquely define the ICR, but no constraints ensure automatically that the perpendicular lines from the two other wheels also intersects the ICR. On car-like steered vehicles, where only the front axle is steerable, the mechanical construction of the steering system will ensure that a unique ICR is always located on the line extended along the rear axle. On the AV, on the other hand, it is up to the motion control algorithms to ensure that the mechanical constraint is replaced by sensible steering references for all four wheels. In practice this can be done by letting two wheels,

for example the two left side wheels, define the ICR. Based on their steering angles β_1 and β_2 the steering angles for the two right side wheels (β_3 and β_4) that renders the ICR uniquely defined can be determined by

$$\beta_3 = \arctan\left(\frac{\cos\beta_1 \sin\beta_2}{\sin(\beta_1 - \beta_2) + \cos\beta_1 \cos\beta_2}\right), \tag{3.2}$$

$$\beta_4 = \arctan\left(\frac{\sin\beta_1 \cos\beta_2}{\sin(\beta_1 - \beta_2) + \cos\beta_1 \cos\beta_2}\right). \tag{3.3}$$

Henceforth, it is assumed that the local position control loops around the steering motors on wheel 3 and wheel 4 are fast enough to guarantee that β_3 and β_4 are always satisfy (3.2) and (3.3).

Remark. Note that the ICR is not uniquely defined when the two perpendicular lines from wheel 1 and 2 coincide, i.e., when $\beta_i = \pi/2 + n\pi,\ i = 1, 2,\ n \in \mathbb{Z}$. All we know is that it lies somewhere on the line passing through wheel 1 and 2. The problem is overcome by using wheel 1 and 4 instead to define the ICR. △

The mutual dependency between the steering angles is an example of a holonomic constraint on the general form [30]

$$f(q_1, q_2, q_3, \ldots, t) = 0, \tag{3.4}$$

A constraint on this form can be used to reduce the number of configuration coordinates; in this case from all four β-angles to only β_1 and β_2. (β_3 and β_4 will be encountered later, but are then considered as functions of β_1 and β_2 and not as an independent part of the configuration coordinates.) Having introduced the constraint of a uniquely defined ICR the set of configuration coordinates q_T and inputs u_T have been reduced to

$$q'_T = \begin{bmatrix} \chi \\ \beta' \\ \phi \end{bmatrix}, \qquad u'_T = \begin{bmatrix} \tau_{\beta'} \\ \tau_\phi \end{bmatrix},$$

with $\beta' = [\beta_1\ \beta_2]^T$ and $\tau_{\beta'} = [\tau_{\beta_1}\ \tau_{\beta_2}]^T$.

Not all constraints can be expressed on the form (3.4). A constraint might be expressed by an inequality instead of an equality, or it might be expressed as a constraint on the time derivatives of the configuration coordinates. If a constraint is not expressible by (3.4) it is said to be nonholonomic or non-integrable [4]. For sensible driving of the AV the vehicle is imposed with the nonholonomic constraints of *free rolling* and *non-slipping*. The free rolling constrain means that no sideways velocity of any wheel is allowed, and this is closely related to the existence of a unique ICR; free rolling can only be achieved if the ICR is unique. The non-slipping constraint, on the other hand, means that the velocity of the point of contact between each wheel and the ground is

always zero in the direction of the wheel. Both of these constraints are related to the velocity of the system and cannot be described as holonomic constraints. Instead, they are described by a set of nonholonomic constraints on the form

$$A^T(q)\dot{q} = 0, \tag{3.5}$$

where $A(q)$ is an $n \times k$ matrix describing k constraints, and n is the number of configuration coordinates. The constraints are assumed to be independent, i.e., $k \leq n$ and rank$(A(q)) = k$.

To derive expressions of the constraints consider the velocity of the i'th wheel relative to the $\mathcal{B}$-frame

$$w_i^{\mathcal{B}} = \begin{bmatrix} \dot{x}_1^{\mathcal{B}} \\ \dot{x}_2^{\mathcal{B}} \end{bmatrix} + \begin{bmatrix} -\kappa_i \sin\gamma_i \\ \kappa_i \cos\gamma_i \end{bmatrix} \dot{\theta} = \begin{bmatrix} 1 & 0 & -\kappa_i \sin\gamma_i \\ 0 & 1 & \kappa_i \cos\gamma_i \end{bmatrix} \dot{\chi}^{\mathcal{B}}.$$

According to the free rolling constraint this velocity vector should point in the exact same direction as the wheel, and according to the non-slipping constraint, the velocity of the contact point between wheel and ground should be zero, and hence the length of the velocity vector should be equal to $r_w\dot{\phi}_i$.

Now, define two unit vectors in the $\mathcal{B}$-frame. One along and one perpendicular to the wheel

$$e_{i,\parallel}^{\mathcal{B}} = \begin{bmatrix} \cos\beta_i \\ \sin\beta_i \end{bmatrix}, \qquad e_{i,\perp}^{\mathcal{B}} = \begin{bmatrix} -\sin\beta_i \\ \cos\beta_i \end{bmatrix}.$$

If the velocity of the wheel is in the direction of the wheel itself the dot product between w_i and $e_{i,\perp}^{\mathcal{B}}$ must equal zero. The free rolling constraint can hence be expressed by

$$< w_i, e_{i,\perp} >= 0 = \begin{bmatrix} -\sin\beta_i & \cos\beta_i & \kappa_i \cos(\gamma_i - \beta_i) \end{bmatrix} \dot{\chi}^{\mathcal{B}}, \qquad i = 1, 2,$$

and expressing the constraints for all wheels in the $\mathcal{N}$-frame yields

$$C_1(\beta')R(\theta)\dot{\chi} = 0, \qquad C_1(\beta') = \begin{bmatrix} -\sin\beta_1 & \cos\beta_1 & \kappa_1\cos(\beta_1 - \gamma_1) \\ -\sin\beta_2 & \cos\beta_2 & \kappa_2\cos(\beta_2 - \gamma_2) \end{bmatrix}. \tag{3.6}$$

Note that the constraint is only imposed on two of the wheels. If the ICR is uniquely defined, and if two of the wheels satisfy the free rolling constraint, then all wheels satisfy the constraint. Hence there are only two independent constraints.

An expression for the non-slipping constraint can be derived in a similar fashion. If the length of the velocity vector $w_i^{\mathcal{B}}$ should be equal to $r_w\dot{\phi}_i$ the inner product between $w_i^{\mathcal{B}}$ and the unit vector along the wheel should also be equal to $r_w\dot{\phi}_i$

$$< w_i, e_{\parallel} >= r_w\dot{\phi}_i = \begin{bmatrix} \cos\beta_i & \sin\beta_i & \kappa_i \sin(\gamma_i - \beta_i) \end{bmatrix} \dot{\chi}^{\mathcal{B}}, \qquad i = 1, 2, 3, 4.$$

Expressing the constraints for all wheels in the $\mathcal{N}$-frame yields

$$C_2(\beta')R(\theta)\dot{\chi} = r_w\dot{\phi}, \qquad C_2(\beta') = \begin{bmatrix} \cos\beta_1 & \sin\beta_1 & \kappa_1\sin(\beta_1-\gamma_1) \\ \cos\beta_2 & \sin\beta_2 & \kappa_2\sin(\beta_2-\gamma_2) \\ \cos\beta_3 & \sin\beta_3 & \kappa_3\sin(\beta_3-\gamma_3) \\ \cos\beta_4 & \sin\beta_4 & \kappa_4\sin(\beta_4-\gamma_4) \end{bmatrix}. \tag{3.7}$$

All four wheels are included since there is no direct relation between the slippage of each wheel and hence all four constraints are independent. Collecting the constraints and expressing them on the form (3.5) yields

$$A^T(q_T')\dot{q}_T' = \begin{bmatrix} C_1(\beta')R(\theta) & 0 & 0 \\ C_2(\beta')R(\theta) & 0 & -r_wI \end{bmatrix}\dot{q}_T' = 0.$$

3.3 Lagrange Model

Having defined the nonholonomic constraints the dynamics of the AV can be described directly by applying Lagrange's equation for nonholonomic systems (see appendix A for an introduction to this equation)

$$\begin{gathered} \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\right) - \frac{\partial L}{\partial q} = A(q)\lambda + \mathcal{U}, \\ A^T(q)\dot{q} = 0, \end{gathered} \tag{3.8}$$

where $\lambda \in \mathbb{R}^k$ is a vector of Lagrange multipliers, and $\mathcal{U}$ includes the inputs and any dissipation or friction forces in the system. The Lagrangian function $L = T - U$ is defined as the difference in kinetic energy T and potential energy U of the entire system. Driving on a horizontal field the AV has no potential energy, but if the field has a slope, the potential energy will be nonzero. An exact expression of the potential energy function is not available, but it is assumed that the energy is a function of χ and that it is bounded from below

$$U:\ \mathcal{M} \to \mathbb{R}, \qquad U(\chi) \geq C.$$

The expression for kinetic energy, on the other hand, is possible to derive immediately; it is a combination of translational and rotational energy of the individual moving parts of the AV. Assume that the following constants are known:

m_f	Mass of AV body (excluding wheels)
j_f	Moment of inertia of AV body (excluding wheels) about the axis perpendicular to the ground and passing through the geometric center of the AV
m_w	Mass of each individual wheel
$j_{w,\beta}$	Moment of inertia of each wheel plus suspension about the β rotation axes
$j_{w,\phi}$	Moment of inertia of each wheel about the ϕ rotation axes

The total kinetic energy of the AV can then be expressed as

$$\begin{aligned} T(q'_T, \dot{q}'_T) = &\frac{1}{2}\left[m_f(\dot{x}_1^2 + \dot{x}_2^2) + j_f\dot{\theta}^2\right] \\ &+ \frac{1}{2}m_w\sum_{i=1}^{4}(\dot{x}_1 - \kappa_i\sin(\gamma_i+\theta)\dot{\theta})^2 + (\dot{x}_2 + \kappa_i\cos(\gamma_i+\theta)\dot{\theta})^2 \\ &+ \frac{1}{2}j_{w,\beta}\sum_{i=1}^{4}(\dot{\beta}_i + \dot{\theta})^2 \\ &+ \frac{1}{2}j_{w,\phi}\sum_{i=1}^{4}\dot{\phi}_i^2. \end{aligned}$$

The first element is the translational and rotational energy of the AV frame, the second element is the translational energy of the wheels, the third element is the rotational energy of the wheels about the β rotation axes, and the fourth element is the rotational energy of the wheels about the ϕ rotation axes.

On a more compact matrix form the Lagrangian function is

$$L(q'_T, \dot{q}'_T) = \frac{1}{2}\left[\dot{\chi}^T R^T(\theta) M R(\theta)\dot{\chi} + j_{w,\beta}\sum_{i=1}^{4}(\dot{\beta}_i + \dot{\theta})^2 + j_{w,\phi}\dot{\phi}^T\dot{\phi}\right] - U(\chi),$$

with a constant positive definite symmetric inertia matrix

$$M = \begin{bmatrix} m_f + 4m_w & 0 & -m_w\sum_{i=1}^{4}\kappa_i\sin\gamma_i \\ 0 & m_f + 4m_w & m_w\sum_{i=1}^{4}\kappa_i\cos\gamma_i \\ -m_w\sum_{i=1}^{4}\kappa_i\sin\gamma_i & m_w\sum_{i=1}^{4}\kappa_i\cos\gamma_i & j_f + m_w\sum_{i=1}^{4}\kappa_i^2 \end{bmatrix}.$$

With the Lagrangian function defined the Lagrange equation 3.8 can be partitioned into three separate parts; each describing the dynamics of the three parts of the configuration

coordinates

$$\left.\begin{aligned}
\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\chi}}\right) - \frac{\partial L}{\partial \chi} &= R^T(\theta)C_1^T(\beta')\lambda_1 + R^T(\theta)C_2^T(\beta')\lambda_2, \\
\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\beta}'}\right) - \frac{\partial L}{\partial \beta'} &= \tau_{\beta'}, \\
\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\phi}}\right) - \frac{\partial L}{\partial \phi} &= -r_w\lambda_2 + \tau_\phi, \\
C_1(\beta')R(\theta)\dot{\chi} &= 0, \\
C_2(\beta')R(\theta)\dot{\chi} - r_w\dot{\phi} &= 0,
\end{aligned}\right\} \tag{3.9}$$

where λ_1 and λ_2 are of appropriate size. Because the inputs $\tau_{\beta'}$ and τ_ϕ are generalized forces acting directly on the generalized coordinates β' and ϕ they are simply added to the last two Lagrange equations.

There is a problem with the second equation of (3.9) though. The rotational energy about the β axes of all four wheels are included in the kinetic energy, but only the torque input to wheel 1 and 2 can change the energy. It was assumed in the previous section that wheel 3 and 4 can be controlled in such a way that β_1 to β_4 will always uniquely define the ICR. But how does this affect the kinetic energy? This is not answered by the Lagrange equations, and one has to look closer at the local control loops of wheel 3 and 4 in order to find the answer. On the other hand, one has to realize that the rotational energy of the wheels constitutes a very small part of the total kinetic energy. The mass m_f of the AV body is approximately 200kg, and at the relatively low speed of $1m/s$ the translational energy in the body frame alone is 100J. Assuming that the wheels can be regarded as solid disks the moment of inertia $j_{w,\beta}$ of each wheel is approximately 0.5kg m^2. Even with a fast turning rate of $\pi\text{rad/s}$ the total sum of the rotational energy in the four wheels is only about 5J. Not very much compared to the 100J in the AV frame.

Realizing that the rotational energy in the wheels only plays a very small role in the overall Lagrangian function it is assumed that this part of the energy can be neglected all together and the model modified accordingly. Neglecting the rotational energy of the wheels about the β axes is equivalent to letting $j_{w,\beta}$ tend to zero, but this raises an issue of controlling β'. Applying a torque to a body with zero moment of inertia would result in an infinite acceleration, in theory at least, of that body. To overcome this problem the steering motors are controlled by local velocity controllers as already described in chapter 2. By neglecting the dynamics of the β angles they are no longer a part of the generalized coordinates; instead, they are treated as time varying parameters that can be manipulated directly through their first derivative. In the next chapter they will lay the grounds for the formal definition of *kinematic inputs*.

The resulting modified Lagrange equations are

$$\left.\begin{aligned} \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\chi}}\right) - \frac{\partial L}{\partial \chi} &= R^T(\theta)C_1^T(\beta')\lambda_1 + R^T(\theta)C_2^T(\beta')\lambda_2, \\ \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\phi}}\right) - \frac{\partial L}{\partial \phi} &= -r_w\lambda_2 + \tau_\phi, \\ \dot{\beta}' &= \zeta, \\ C_1(\beta')R(\theta)\dot{\chi} &= 0, \\ C_2(\beta')R(\theta)\dot{\chi} - r_w\dot{\phi} &= 0. \end{aligned}\right\} \tag{3.10}$$

The model is still not useful for control. There are still the two undetermined Lagrange multipliers that have to be eliminated. Fortunately, the model contains more information than is needed and can be reduced even further. The AV should be able to follow crop rows, navigate way-points, etc., which is related to the position and velocity of the AV frame. Knowing the exact angle ϕ_i of each wheel is therefore irrelevant. The velocities $\dot{\phi}$ of the wheels are important though, since they contribute to the kinetic energy of the system. Fortunately, the direct dependency of $\dot{\phi}$ can be eliminated by using the non-slipping constraint (3.7) that relates $\dot{\phi}$ to $\dot{\chi}$. By exchanging $\dot{\phi}$ with $\frac{1}{r_w}C_2(\beta')R(\theta)\dot{\chi}$ the Lagrangian function becomes a function of χ, $\dot{\chi}$ and β' alone

$$L'(\chi, \dot{\chi}, \beta') = \frac{1}{2}\dot{\chi}^T R^T(\theta)\bar{M}(\beta')R(\theta)\dot{\chi} - U(\chi),$$

with an augmented inertia matrix

$$\bar{M}(\beta') = M + \frac{j_{w,\phi}}{r_w^2}C_2^T(\beta')C_2(\beta').$$

With this reformulated Lagrangian function the left hand side of the second equation of (3.10) becomes zero and hence $\lambda_2 = \frac{1}{r_w}\tau_\phi$. Inserting this reduces (3.10) to

$$\left.\begin{aligned} \frac{d}{dt}\left(\frac{\partial L'}{\partial \dot{\chi}}\right) - \frac{\partial L'}{\partial \chi} &= R^T(\theta)C_1^T(\beta')\lambda_1 + R^T(\theta)C_2^T(\beta')\frac{1}{r_w}\tau_\phi, \\ \dot{\beta}' &= \zeta, \\ C_1(\beta')R(\theta)\dot{\chi} &= 0. \end{aligned}\right\} \tag{3.11}$$

There is still the problem of eliminating λ_1. This will be the topic of the next chapter, which describes a general method of eliminating the Lagrange multipliers. The method is based on the equivalent Hamiltonian formulation of a more general system. Once the multipliers have been eliminated the model can be verified based on measurements on the physical system. This is postponed until chapter 5. The Hamiltonian equivalent of model (3.11) will also be the starting point for developing control algorithms for the AV.

CHAPTER 4

THE HAMILTONIAN FORMULATION AND MODEL REDUCTION

In the previous chapter the Lagrangian formulation of a mechanical system was applied to the AV. The resulting dynamic model is an example of a standard constrained Lagrangian model with the exception that a part of it, β', can be controlled directly through its first derivative. This phenomenon is formally introduced in this chapter as *kinematic inputs* in the framework of general nonholonomic Lagrange systems.

Although the Lagrange equation is a powerful way of deriving the dynamic equations of a mechanical system it is not well suited for control. An equivalent formulation of the dynamics, which will be introduced in this chapter, is the Hamiltonian formulation of the system. This formulation adds no new information to the system, and it is essentially just a transformation of the coordinates of the Lagrange equation, but it has a structure that makes is very useful for feedback control; the Hamiltonian function, which is the dual of the Lagrangian function, can often be used as a Lyapunov function, and basic theorems of passive and dissipative systems have a natural application to the Hamiltonian equation.

Feedback control will be postpone until chapter 6, and the focus in this chapter is on deriving a suitable Hamiltonian equation of a general system with kinematic inputs. The

undetermined Lagrange multipliers will still exist in the Hamiltonian equivalent of the original Lagrangian equation, and the last part of this chapter describes a method of reducing the system by eliminating the multipliers.

4.1 The Constrained Tangent Bundle

Although the configuration manifold of the AV was introduced in chapter 3 the full configuration space has not yet been precisely defined. The position and orientation of the AV is describes by points on the configuration manifold $\mathcal{M}$, but this is not enough to describe the instantaneous state of the system. The system also has a velocity, and we need to choose a space in which both the position and velocity can reside. In mechanical systems the natural choice is the tangent bundle, which defines a space of velocities associated to every point on the configuration manifold. In nonholonomic systems the space of velocities is further limited by the constraints, and this is the background for defining the constrained tangent bundle for a general nonholonomic system.

Consider an n-dimensional configuration manifold $\mathcal{M}$ and a physical system with $q \in \mathcal{M}$ describing its instantaneous configuration on the manifold. Let the tangent space at q be denoted $T_q\mathcal{M} \subset \mathbb{R}^n$. Any vector $\xi \in T_q\mathcal{M}$ belonging to the tangent space is called a tangent vector to $\mathcal{M}$ at q, and the velocity $\dot{q}$ of the point q is an example of an element of the tangent space. The union of all tangent spaces $\bigcup_{q\in\mathcal{M}} T_q\mathcal{M}$ is called the tangent bundle of $\mathcal{M}$ and is denoted $T\mathcal{M}$. An element on the tangent bundle consists of a point q and a vector ξ belonging to the corresponding tangent space at q

$$(q,\ \xi) \in T\mathcal{M} \quad (2n\text{-dimensional}).$$

Now, consider a system imposed with a set of nonholonomic constraints

$$A^T(q)\dot{q} = 0,$$

where $A(q)$ is an $n \times k$ differentiable matrix of constant rank k defining k independent nonholonomic constraints. The constraints limit the set of allowed velocities on $T_q\mathcal{M}$, and the constrained tangent bundle is defined as a subset of the full tangent bundle

$$\Omega_c = \{(q, \dot{q}) \in T\mathcal{M} \mid A^T(q)\dot{q} = 0\}.$$

4.2 Kinematic Inputs

As stated in the previous chapter the direct steering of the wheels lays the grounds for defining the notion of kinematic inputs. The basic property of the kinematic inputs is that

they cannot directly change the amount of energy in the system; the energy is invariant with respect to the coordinates related to the kinematic inputs. The formal definition of the kinematic inputs of a mechanical system is as follows.

Definition 4.1. *Consider a set of coordinates $\bar{q} = [q^T \ r^T]^T \in \mathcal{M} \times \mathcal{R}$ defining the instantaneous configuration of a mechanical system. Assume that the l-dimensional subset r of the coordinates can be controlled directly through the first time derivative by an input v, such that*

$$\dot{r} = v \quad v \in T_r\mathcal{R} \subset \mathbb{R}^l,$$

and that

$$\frac{\partial T}{\partial r} = 0 \qquad \textit{for} \qquad (q, \dot{q}) \in \Omega_c,$$

where $T : \ \Omega_c \times \mathcal{R} \to \mathbb{R}$ and $U : \ \mathcal{M} \to \mathbb{R}$ describe the kinetic and potential energy of the system. Then v is a kinematic input.

As a result of this definition it is possible to define the Lagrangian equation of motion of a nonholonomic system augmented with kinematic inputs

$$\left.\begin{aligned} \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\right) - \frac{\partial L}{\partial q} &= A(q,r)\lambda + B(q,r)u, \\ \dot{r} &= v, \\ A^T(q,r)\dot{q} &= 0, \end{aligned}\right\} \tag{4.1}$$

with a Lagrangian function $L(q, \dot{q}, r) = T(q, \dot{q}, r) - U(q)$, $\lambda \in \mathbb{R}^k$, and $u \in \mathbb{R}^m$. The AV model (3.11) is an example of a system on this form. Note that r is allowed to change the structure of both the constraint matrix A and the input matrix B. The dependency on r in $A(q, r)$ has an interesting effect on the shape of Ω_c. Changing r will also change the space of allowed velocities, and the constrained tangent bundle becomes dependent on r

$$\underset{r \in \mathcal{R}}{\Omega_c} = \{(q, \dot{q}) \in T\mathcal{M} \mid A^T(q,r)\dot{q} = 0\}.$$

The dependency on r means that it is possible to change the shape of the constrained tangent bundle through $\dot{r} = v$, and this will prove to be a useful feature when controlling such systems. Consider the union of all possible constrained tangent bundles $\bar{\Omega}_c = \bigcup_{r \in \mathcal{R}} \Omega_c$. Clearly $\Omega_c \subseteq \bar{\Omega}_c$ for any $r \in \mathcal{R}$, which implies that the reachable space of the system can be increased by changing r. Ultimately, the system can be viewed as a (nearly) unconstrained system if $\bar{\Omega}_c = \mathcal{M} \times \mathbb{R}^n$

4.3 The Hamiltonian Model

The Lagrangian formulation of the dynamics in (4.1) describes the motion of the system in terms of n second order differential equations in q, and the state of the system is described by coordinates q and velocities $\dot{q}$ (for the moment r is assumed to be constant). The Hamiltonian formulation, on the other hand, seeks to describe the motion of the system by $2n$ first order differential equations in $2n$ variables. One can argue that the Lagrangian formulation already describes $2n$ first order equations if q and $\dot{q}$ are chosen as the $2n$ variables, but there exists a more suitable choice that will make the system equations almost symmetric. This new set of variables comprises the original generalized coordinates q and a new n-dimensional generalized momentum

$$p = \frac{\partial L}{\partial \dot{q}}. \tag{4.2}$$

The generalized momentum is said to lie in the cotangent space of $\mathcal{M}$, which is denoted $p \in T_q^*\mathcal{M}$. The cotangent space of $\mathcal{M}$ is defined as the vector space of linear functions $f : T_q\mathcal{M} \to \mathbb{R}$ mapping elements on the tangent space to the real axis. In the same fashion, as with the tangent bundle, the cotangent bundle can be defined as the union of all cotangent spaces on $\mathcal{M}$

$$T^*\mathcal{M} = \bigcup_{q \in \mathcal{M}} T_q^*\mathcal{M}.$$

The dimension of cotangent bundle is the same as the dimension of the tangent bundle.

The procedure of switching from $(q, \dot{q}, r)$ in the Lagrangian formulation to (q, p, r) in the Hamiltonian formulation is provided by the Legendre transformation that transforms functions on a vector space (the tangent bundle) to functions on the dual vector space (the cotangent bundle). A detailed discussion of the Legendre transformation and its use in physical systems can be found in many books on mechanical systems and mathematical analysis, see for example [4]. In this setting the Legendre transformation is used to transform the Lagrangian function, which is a function on the tangent bundle, to a new function on the cotangent bundle. The new function is the Hamiltonian function

$$H(q, p, r) = p^T \dot{q} - L(q, \dot{q}, r), \tag{4.3}$$

in which $\dot{q}$ is expressed in terms of p by the relation (4.2). There is a simple physical relation between the Lagrangian and Hamiltonian functions if the kinetic energy is described by a quadratic function in $\dot{q}$

$$L(q, \dot{q}, r) = \frac{1}{2}\dot{q}^T M(q, r)\dot{q} - U(q),$$

where $M(q, r)$ is a positive definite inertia matrix. The generalized velocity is then related to the generalized momentum by

$$p = \frac{\partial L}{\partial \dot{q}} = M(q, r)\dot{q} \qquad \Leftrightarrow \qquad \dot{q} = M^{-1}(q, r)p,$$

and the Hamiltonian function becomes

$$\begin{aligned} H(q,p,r) &= p^T\dot{q} - L(q,\dot{q},r) \\ &= \frac{1}{2}p^T M^{-1}(q,r)p + U(q). \end{aligned} \tag{4.4}$$

The first term is the kinetic energy described in the generalized momentum and the Hamiltonian function is hence the sum of kinetic and potential energy, whereas the Lagrangian is the difference.

Having defined the Hamiltonian function the next step is to derive the equations of motion of the system (4.1) on the cotangent bundle. The resulting equations for an ordinary Lagrangian system without kinematic inputs is a basic result from classical mechanics and are referred to as the Hamiltonian equations of motion. As the following theorem shows, the same equivalent exists for systems with kinematic inputs.

Theorem 4.1. *The system* (4.1) *is equivalent to the system of first order Hamiltonian equations*

$$\left.\begin{aligned} \begin{bmatrix}\dot{q}\\ \dot{p}\end{bmatrix} &= \begin{bmatrix}0 & I\\ -I & 0\end{bmatrix}\begin{bmatrix}\frac{\partial H}{\partial q}\\ \frac{\partial H}{\partial p}\end{bmatrix} + \begin{bmatrix}0\\ B(q,r)\end{bmatrix}u + \begin{bmatrix}0\\ A(q,r)\end{bmatrix}\lambda, \\ \dot{r} &= v, \\ 0 &= A^T(q,r)\frac{\partial H}{\partial p}, \end{aligned}\right\} \tag{4.5}$$

where $H(q,p,r) = p^T\dot{q} - L(q,\dot{q},r)$ *is the Legendre transformation of the Lagrangian function viewed as a function of* $\dot{q}$*.* $p = \frac{\partial L}{\partial \dot{q}}$ *is the generalized momentum.*

Proof. By definition, the Hamiltonian function is a function of q, p, and r and the total derivative of the function is

$$dH = \frac{\partial^T H}{\partial q}dq + \frac{\partial^T H}{\partial p}dp + \frac{\partial^T H}{\partial r}dr, \tag{4.6}$$

but from the definition of the Hamiltonian function (4.3) we can also write

$$\begin{aligned} dH &= \dot{q}^T dp + p^T d\dot{q} - \frac{\partial^T L}{\partial q}dq - \frac{\partial^T L}{\partial \dot{q}}d\dot{q} - \frac{\partial^T L}{\partial r}dr \\ &= \dot{q}^T dp - \frac{\partial^T L}{\partial q}dq - \frac{\partial^T L}{\partial r}dr. \end{aligned} \tag{4.7}$$

From (4.1) we have that

$$\frac{\partial L}{\partial q} = \dot{p} - A(q,r)\lambda - B(q,r)u,$$

and inserting this into (4.7) yields

$$dH = \dot{q}^T dp - [\dot{p} - A(q,r)\lambda - B(q,r)u]^T \, dq - \frac{\partial^T L}{\partial r} dr.$$

The Hamiltonian system (4.5) is obtained by matching terms with (4.6). □

Remark. From the definition of the kinematic input it can also be concluded from the proof that $\partial H/\partial r = -\partial L/\partial r = 0$. △

Remark. The drift vector field exhibits an almost symmetric structure, which is evident in any Hamiltonian system. It is perhaps more clear in the classical system, where $\dot{q} = \partial H/\partial p$ and $\dot{p} = -\partial H/\partial q$. The Hamiltonian system is said to have a *symplectic* structure [4]. △

Analogous to the definition of the constrained tangent bundle we can also define the constrained cotangent bundle for the Hamiltonian system

$$\underset{r \in \mathcal{R}}{\Omega_c^*} = \{(q,p) \in T^*\mathcal{M} \mid A^T(q,r)\frac{\partial H}{\partial p} = 0\}.$$

The Hamiltonian system (4.5) describes the motion of a system with kinematic inputs on the constrained cotangent bundle Ω_c^*.

4.4 Eliminating the Lagrange multipliers

The Hamiltonian equations still contain the undetermined Lagrange multipliers, and if the model is to be used for control, the multipliers need to be eliminated. This section describes a method of doing this for a system on the form (4.5). The method is largely based on the method described by van der Schaft and Maschke [54], but it is extended to cover systems with kinematic inputs as well. The basic idea is to define a coordinate transformation $p \mapsto \tilde{p}$, $\tilde{p} = [\tilde{p}_1^T \ \tilde{p}_2^T]^T$, where $\tilde{p}_1$ is invariant with respect to λ, and the Hamiltonian function can be rewritten in terms of q, r, and $\tilde{p}_1$ alone. λ then only affects the dynamics of $\tilde{p}_2$, which can be safely disregarded.

Remark. For convenience we use the following notation. Consider an n-dimensional column vector $X(x)$, where each entry is a function of the m-dimensional vector x. The partial derivative of X with respect to x is defined as the matrix

$$\frac{\partial X}{\partial x} = \begin{bmatrix} \frac{\partial X_1}{\partial x_1} & \cdots & \frac{\partial X_1}{\partial x_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial X_n}{\partial x_1} & \cdots & \frac{\partial X_n}{\partial x_m} \end{bmatrix}$$

△

If the constraint matrix $A(q,r)$ is differentiable and rank$(A) = k$ there exists an $n \times (n-k)$ differentiable matrix $S(q,r)$ of rank $n-k$ whose columns completely span the kernel of $A^T(q,r)$

$$A^T(q,r)S(q,r) = 0 \qquad \Leftrightarrow \qquad \ker[A^T(q,r)] = \text{img}[S(q,r)]. \tag{4.8}$$

Define a diffeomorphic coordinate change $(q,p) \mapsto (q, \tilde{p}_1, \tilde{p}_2)$

$$\begin{bmatrix} \tilde{p}_1 \\ \tilde{p}_2 \end{bmatrix} = \begin{bmatrix} S^T(q,r) \\ A^T(q,r) \end{bmatrix} p. \tag{4.9}$$

van der Schaft and Maschke [54] showed that $\frac{\partial \tilde{H}}{\partial \tilde{p}_2} = 0$ on Ω_c^*, but actually the Hamiltonian is completely independent of $\tilde{p}_2$ as the following theorem shows.

Theorem 4.2. *Consider a Hamiltonian function on the form* (4.4) *and a coordinate transformation* $(q,p) \mapsto (q, \tilde{p}_1, \tilde{p}_2)$ *on the form* (4.9)*, where* $A(q,r)$ *and* $S(q,r)$ *satisfy* (4.8)*. In the new coordinates the Hamiltonian function is described by*

$$\tilde{H}(q,r,\tilde{p}_1) = \frac{1}{2}\tilde{p}_1^T \left(S^T(q,r)M(q,r)S(q,r)\right)^{-1} \tilde{p}_1 + U(q). \tag{4.10}$$

Proof. From the constraint we know that $\dot{q} \in \ker[A^T(q,r)]$. Hence, from (4.8) it can be concluded that $\dot{q} \in \text{img}[S(q,r)]$, and there exists a set of $n-k$ independent signals η such that

$$\dot{q} = S(q,r)\eta.$$

The kinetic energy can then be written as $T = \frac{1}{2}\eta^T S^T M S \eta$, and the Hamiltonian function becomes

$$H_\eta(q,r,\eta) = \frac{1}{2}\eta^T S^T(q,r)M(q,r)S(q,r)\eta + U(q).$$

Using the definition of the generalized momentum we also know that

$$\tilde{p}_1 = S^T(q,r)p = S^T(q,r)M(q,r)\dot{q} = S^T(q,r)M(q,r)S(q,r)\eta.$$

$M(q,r)$ is positive definite, rank$(S(q,r)) = n-k$ everywhere, and the square matrix $S^T(q,r)M(q,r)S(q,r)$ is hence full rank and invertible. The Hamiltonian on the constrained cotangent bundle Ω_c^* can thus be described by (4.10). □

Remark. The Lagrange multipliers can be interpreted as the forces that guarantee that the nonholonomic constraints are always satisfied. On the AV they make sure that there are no sideways displacement of the wheels and that there are no displacement of the contact point between each wheel and the ground. In this sense they are not doing any work on the system. As the results show the constraint forces act in the direction of $\tilde{p}_2$, but since the constraint forces cannot contribute to the total energy, it makes sense that the Hamiltonian function is invariant with respect to $\tilde{p}_2$. △

The next step is to derive the system equations in terms of the new Hamiltonian function and the new coordinates. The dynamics of the generalized coordinates are

$$\dot{q} = \frac{\partial H}{\partial p} = \left(\frac{\partial \tilde{p}_1}{\partial p}\right)^T \frac{\partial \tilde{H}}{\partial \tilde{p}_1} + \left(\frac{\partial \tilde{p}_2}{\partial p}\right)^T \frac{\partial \tilde{H}}{\partial \tilde{p}_2} = S(q,r)\frac{\partial \tilde{H}}{\partial \tilde{p}_1}. \tag{4.11}$$

To find the expressions for $\dot{\tilde{p}}_1$ we first derive $\dot{p}$ in terms of the new coordinates

$$\begin{aligned}
\dot{p} &= -\frac{\partial H}{\partial q} + A\lambda + Bu \\
&= -\frac{\partial \tilde{H}}{\partial q} - \left(\frac{\partial \tilde{p}_1}{\partial q}\right)^T \frac{\partial \tilde{H}}{\partial \tilde{p}_1} + A\lambda + Bu \\
&= -\frac{\partial \tilde{H}}{\partial q} - \begin{bmatrix} \frac{\partial p^T S_1}{\partial q} & \cdots & \frac{\partial p^T S_{n-k}}{\partial q} \end{bmatrix} \frac{\partial \tilde{H}}{\partial \tilde{p}_1} + A\lambda + Bu,
\end{aligned}$$

where S_i denotes the i'th column of $S(q,r)$

The time derivative of the i'th element of the new generalized momentum $\tilde{p}_1$ is

$$\begin{aligned}
\dot{\tilde{p}}_{1,i} &= \frac{d}{dt}\left(S_i^T p\right) = p^T \frac{\partial S_i}{\partial q}\dot{q} + p^T \frac{\partial S_i}{\partial r} v + S_i^T \dot{p} \\
&= p^T \frac{\partial S_i}{\partial q} S \frac{\partial \tilde{H}}{\partial \tilde{p}_1} + p^T \frac{\partial S_i}{\partial r} v + S_i^T \left(-\frac{\partial \tilde{H}}{\partial q} - \begin{bmatrix} \frac{\partial p^T S_1}{\partial q} & \cdots & \frac{\partial p^T S_{n-k}}{\partial q} \end{bmatrix} \frac{\partial \tilde{H}}{\partial \tilde{p}_1} + A\lambda + Bu \right) \\
&= \left(p^T \frac{\partial S_i}{\partial q} S - \begin{bmatrix} p^T \frac{\partial S_1}{\partial q} S_i & \cdots & p^T \frac{\partial S_{n-k}}{\partial q} S_i \end{bmatrix} \right) \frac{\partial \tilde{H}}{\partial \tilde{p}_1} + p^T \frac{\partial S_i}{\partial r} v - S_i^T \frac{\partial \tilde{H}}{\partial q} + S_i^T B u.
\end{aligned} \tag{4.12}$$

The two dynamic equations (4.11) and (4.12) do not depend on neither λ nor $\tilde{p}_2$. The Lagrange multipliers have thus been eliminated, and the motion of the constrained system is described by (4.11) and (4.12). This leads to the following Hamiltonian system

$$\left.\begin{aligned}
\begin{bmatrix} \dot{q} \\ \dot{\tilde{p}}_1 \end{bmatrix} &= J(q, \tilde{p}_1, r) \begin{bmatrix} \frac{\partial \tilde{H}}{\partial q} \\ \frac{\partial \tilde{H}}{\partial \tilde{p}_1} \end{bmatrix} + \begin{bmatrix} 0 \\ S^T(q,r)B(q,r) \end{bmatrix} u + \begin{bmatrix} 0 \\ B_v(q, \tilde{p}_1, r) \end{bmatrix} v, \\
\dot{r} &= v, \\
\tilde{H}(q,r,\tilde{p}_1) &= \frac{1}{2}\tilde{p}_1^T \left(S^T(q,r)M(q,r)S(q,r)\right)^{-1} \tilde{p}_1 + U(q),
\end{aligned}\right\} \tag{4.13}$$

with interconnection matrix

$$J(q, \tilde{p}_1, r) = \begin{bmatrix} 0 & S(q,r) \\ -S^T(q,r) & \left(-p^T[S_i, S_j](q)\right)_{i,j} \end{bmatrix},$$

and kinematic input vector field

$$B_v(q, \tilde{p}_1, r) = \begin{bmatrix} p^T \frac{\partial S_1}{\partial r} \\ \vdots \\ p^T \frac{\partial S_{n-k}}{\partial r} \end{bmatrix},$$

where $[S_i, S_j](q)$ denotes the Lie bracket of S_i and S_j in q

$$[S_i, S_j](q) = \frac{\partial S_j}{\partial q}(q, r)S_i - \frac{\partial S_i}{\partial q}(q, r)S_j.$$

Note that $J(q, r) = -J^T(q, r)$. The symplectic structure of the original system has hence been maintained in the reduced system.

4.5 Inertia Matrix Scaling

Although the Lagrange multipliers have been eliminated, the reduction scheme has introduced the additional input term $B_v v$. The new input term is a result of the form of the new inertia matrix of the new Hamiltonian function of the reduced system. The Hamiltonian function $\tilde{H}$ is still invariant with respect to r but the inertia matrix $S^T M S$ might not be. Hence, if the inertia matrix is changed by the kinematic input v the momentum $\tilde{p}_1$ must be changed accordingly for $\tilde{H}$ to be rendered invariant. This explains the input term $B_v v$, which acts directly on $\tilde{p}_1$.

The only difference between the unreduced system with kinematic inputs (4.5) and the classical Hamiltonian system is the dependency on r in the constraint and input matrices. The similarity with the classical system is no longer as clear with the introduction of the extra input term $B_v v$, but fortunately, the choice of $S(q, r)$ is not unique, and as the following theorem shows, it is possible to eliminate the term $B_v v$ by an appropriate scaling of the inertia matrix $S^T M S$. By eliminating the extra input term, we will end up with a system very similar to a classical Hamiltonian system.

Theorem 4.3. *There exists an $n \times (n-k)$ matrix $S_I(q, r)$ of rank $n-k$ that satisfies $A^T(q, r)S_I(q, r) = 0$ and the relation*

$$S_I^T(q, r)M(q, r)S_I(q, r) = I, \tag{4.14}$$

where $M(q, r)$ is a strictly positive definite symmetric inertia matrix of the system (4.5)*. Using $S = S_I$ in* (4.13) *eliminates the input vector field*

$$B_v(q, r, p) = 0. \tag{4.15}$$

Proof. Assume that there exists a matrix $S_I(q,r)$ that satisfies (4.14) and let $S = S_I$. Then the Hamiltonian function is $\tilde{H} = \frac{1}{2}\tilde{p}_1^T\tilde{p}_1 + U(q)$, and from the definition of the kinematic input we know that $\partial\tilde{H}/\partial r = \partial\tilde{p}_1/\partial r = 0$ and hence

$$\frac{\partial\tilde{p}_1}{\partial r} = \frac{\partial}{\partial r}S^T(q,r)p = B_v(q,r,p) = 0.$$

To prove that S_I exists consider an $n \times (n-k)$ matrix $\bar{S}(q,r)$ of rank $n-k$ that satisfies

$$A^T(q,r)\bar{S}(q,r) = 0.$$

$A(q,r)$ and $\bar{S}(q,r)$ constitutes a proper transform pair, but so does $A(q,r)$ and $\bar{S}(q,r)\Upsilon(q,r)$, where $\Upsilon(q,r)$ is any $(n-k)\times(n-k)$ non-singular matrix. If we take $S_I = \bar{S}\Upsilon$ then (4.14) is satisfied if there exists an $\Upsilon(q,r)$ such that

$$\Upsilon^T\bar{S}^T M\bar{S}\Upsilon = I. \tag{4.16}$$

The symmetric matrix $\bar{S}^T M\bar{S}$ is a congruence transformation of M, and the positive definiteness of M is hence maintained in $\bar{S}^T M\bar{S}$. For a positive definite symmetric matrix there always exists a diagonalization on the form $\bar{S}^T M\bar{S} = Q^T\Lambda Q$, where Λ is a diagonal matrix composed of the strictly positive eigenvalues, and Q is a matrix composed of orthonormal eigenvectors. Now choose $\Upsilon = (Q^T\Lambda^{\frac{1}{2}}Q)^{-1}$, where $\Lambda^{\frac{1}{2}}$ denotes the diagonal matrix of square roots of the individual eigenvalues ($\Lambda = \Lambda^{\frac{1}{2}}\Lambda^{\frac{1}{2}}$). Then (4.16) becomes

$$(Q^T\Lambda^{\frac{1}{2}}Q)^{-T}Q^T\Lambda Q(Q^T\Lambda^{\frac{1}{2}}Q)^{-1} = Q^T\Lambda^{-\frac{1}{2}}QQ^T\Lambda QQ^T\Lambda^{-\frac{1}{2}}Q = I.$$

□

With the choice of $S = S_I$, the system (4.13) is reduced to

$$\left.\begin{aligned}\begin{bmatrix}\dot{q}\\ \dot{\tilde{p}}_1\end{bmatrix} &= J(q,\tilde{p}_1,r)\begin{bmatrix}\frac{\partial\tilde{H}}{\partial q}\\ \frac{\partial\tilde{H}}{\partial\tilde{p}_1}\end{bmatrix} + \begin{bmatrix}0\\ S^T(q,r)B(q,r)\end{bmatrix}u,\\ \dot{r} &= v,\\ \tilde{H}(q,\tilde{p}_q) &= \frac{1}{2}\tilde{p}_1^T\tilde{p}_1 + U(q).\end{aligned}\right\} \tag{4.17}$$

4.6 Discussion

By eliminating the Lagrange multipliers, and after an appropriate scaling of the inertia matrix, the resulting system equations have been reduced to a system of $2n-k$ first order

Hamiltonian equations with a very simple Hamiltonian function; plus the l original first order linear equations for the kinematic inputs.

The only difference between the reduced system (4.17) and a classical Hamiltonian system without kinematic inputs is the dependency on r in the interconnection and input matrices. The passivity properties of the classical Hamiltonian system therefore also apply to this system, since fixing r in (4.17) converts it to a classical system. Since the Hamiltonian function in invariant with respect to r the passivity is also retained when r varies. The introduction of the kinematic input means that the direction of movement can be changed without affecting the total energy of the system and without sacrificing the useful passivity property.

CHAPTER 5

REDUCTION AND VALIDATION OF THE AV MODEL

The Lagrange model of the AV, which was derived in chapter 3, is a special case of the general Lagrange model of a system with kinematic inputs. The role of the kinematic inputs are played by the velocities of the steering motors on the AV, and an equivalent Hamiltonian model of the AV can be obtained by applying the reduction scheme introduced in the previous chapter. This chapter introduces the reduced Hamiltonian equations for the AV, where the undetermined Lagrange multipliers have been eliminated. With the Lagrange multipliers gone, and the AV equations written on a much simpler form, the model is verified against measurements on the physical system.

5.1 The Reduced Hamiltonian Equivalent

Before applying the reduction scheme to the AV Lagrange model we must first check that the model has the correct structure. When comparing (3.11) on page 48 to (4.1) on

page 51 it is evident that the AV Lagrange model is on the correct form, with

$$\begin{aligned} L &\mapsto L', \\ q &\mapsto \chi, \\ A &\mapsto R^T C_1^T, \\ \lambda &\mapsto \lambda_1, \\ B &\mapsto R^T C_2^T \frac{1}{r_w}, \\ u &\mapsto \tau_\phi, \\ r &\mapsto \beta', \\ v &\mapsto \zeta. \end{aligned}$$

The second thing to check is whether or not the velocities of the steering motors satisfy the conditions of proper kinematic inputs. According to definition 4.1 on page 51 ζ is a proper kinematic input if both the kinetic energy T and the potential energy U are invariant with respect to β' on the constrained tangent bundle Ω_c. The potential energy U at a given point in Ω_c is derived from the height above sea level (or any other constant reference point), and it is assumed to be completely independent of β'. All that is left to be checked is if the kinetic energy is also invariant with respect to β'. It is invariant if the partial derivative of T with respect to β' is zero on Ω_c:

$$\frac{\partial T}{\partial \beta_i} = \frac{\partial}{\partial \beta_i}\left(\frac{1}{2}\dot{\chi}^T R^T(\theta)\bar{M}(\beta')R(\theta)\dot{\chi}\right) = 0, \quad i = 1,2 \qquad \text{for} \qquad (\chi, \dot{\chi}) \in \Omega_c, \tag{5.1}$$

with the β' dependent inertia matrix $\bar{M}$

$$\bar{M}(\beta') = M + \frac{j_{w,\phi}}{r_w^2} C_2^T(\beta') C_2(\beta').$$

Since M is constant, (5.1) reduces to checking whether

$$\dot{\chi}^T R^T(\theta)\left[\frac{\partial C_2^T}{\partial \beta_i} C_2(\beta') + C_2^T(\beta')\frac{\partial C_2}{\partial \beta_i}\right] R(\theta)\dot{\chi} = 0, \quad i = 1,2 \qquad \text{for} \qquad (\chi, \dot{\chi}) \in \Omega_c \tag{5.2}$$

is true. Consider the derivative of the first row of C_2 denoted by $C_{2,1}$ with respect to β_1

$$\frac{\partial C_{2,1}}{\partial \beta_1} = \begin{bmatrix} -\sin\beta_1 & \cos\beta_1 & \kappa_1 \cos(\beta_1 - \gamma_1) \end{bmatrix}.$$

The right side is exactly the same as the first row of C_1, see (3.6) in page 44. The relationship between C_1 and the derivative of C_2 can be extended even further by looking at all possible derivatives of the rows of C_2, but this will not be shown here. It can easily

be verified that the derivative of any row of C_2 will result in a vector that is a linear combination of the rows of C_1, i.e.,

$$\frac{\partial C_{2,j}}{\partial \beta_i} = \sum_{k=1}^{4} \alpha_{kji} C_{1,k}, \qquad \alpha_k \in \mathbb{R}.$$

Consider a vector v that lies in the kernel of C_1. This means that $C_{1,k}v = 0$ and hence also that v lies the kernel of the space spanned by the derivatives of C_2. From the free rolling constraint we know that $R\dot{\chi} \in \ker[C_1] \subset \ker\left[\frac{\partial C_2}{\partial \beta_i}\right]$ on Ω_c. The multiplication terms involving $R\dot{\chi}$ and $\frac{\partial C_2}{\partial \beta_i}$ in (5.2) vanish, and the equation is therefore true. ζ is hence a proper kinematic input. Note that the equality will not be true for an arbitrary χ and $\dot{\chi}$ outside Ω_c.

Finding the Hamiltonian equivalent of the AV Lagrange model is now done in three steps; step one is to derive the unreduced Hamiltonian equivalent of the Lagrange model, where the Lagrange multipliers are left untouched; the second step is to find a matrix S that can be used to reduce the system by elimination of the Lagrange multipliers; the third and final step is to find a suitable positive definite matrix Υ (it turns out to be a scalar function) that renders the new momentum $\tilde{p}_1$ invariant with respect to the kinematic input ζ.

Step 1. Finding the unreduced Hamiltonian equivalent

The Hamiltonian equivalent of the Lagrangian model (3.11) is obtained by direct application of theorem 4.1 on page 53. The Hamiltonian equivalent of the AV model is

$$\left.\begin{aligned} \begin{bmatrix}\dot{\chi}\\ \dot{p}\end{bmatrix} &= \begin{bmatrix}0 & I\\ -I & 0\end{bmatrix}\begin{bmatrix}\frac{\partial H}{\partial \chi}\\ \frac{\partial H}{\partial p}\end{bmatrix} + \begin{bmatrix}0\\ R^T(\theta)C_2^T(\beta')\frac{1}{r_w}\end{bmatrix}\tau_\phi + \begin{bmatrix}0\\ R^T(\theta)C_1^T(\beta')\end{bmatrix}\lambda_1,\\ \dot{\beta}' &= \zeta,\\ 0 &= C_1(\beta')R(\theta)\frac{\partial H}{\partial p}, \end{aligned}\right\} \tag{5.3}$$

with $p = R^T(\theta)\bar{M}(\beta)R(\theta)\dot{\chi}$ and Hamiltonian function

$$H(\chi, p, \beta') = \frac{1}{2}p^T[R^T(\theta)\bar{M}(\beta')R(\theta)]^{-1}p + U(\chi).$$

The system defines the motion of the AV on the constrained cotangent bundle

$$\Omega_c^* = \{(\chi, p) \in T^*\mathcal{M} \mid C_1(\beta')R(\theta)\frac{\partial H}{\partial p} = 0\}.$$

Step 2. Eliminating the Lagrange multipliers

To apply the reduction method and eliminate the Lagrange multipliers we need to find a diffeomorphic change of coordinates $\tilde{p}_1 = \Upsilon^T \bar{S}^T p$ and $\tilde{p}_2 = C_1 R p$ where

$$C_1(\beta')R(\theta)\bar{S}(\chi,\beta') = 0. \tag{5.4}$$

The two rows of $C_1 R$ defines two independent constraints on the three dimensional manifold $\mathcal{M}$. The kernel of $C_1 R$ is hence one dimensional, and we are seeking a three dimensional nonzero column vector $\bar{S}$, which spans this kernel. One choice is

$$\bar{S}(\chi,\beta') = R^T(\theta)\Sigma(\beta'),$$
$$\Sigma(\beta') = \begin{bmatrix} \cos\beta_2\kappa_1\cos(\beta_1-\gamma_1) - \cos\beta_1\kappa_2\cos(\beta_2-\gamma_2) \\ \sin\beta_2\kappa_1\cos(\beta_1-\gamma_1) - \sin\beta_1\kappa_2\cos(\beta_2-\gamma_2) \\ \sin(\beta_1-\beta_2) \end{bmatrix}.$$

$\bar{S}$ is nonzero (full rank) except when $\beta_i = \pi/2 + n\pi,\ i = 1, 2,\ n \in \mathbb{Z}$. See the remark on page 43 on how to overcome this singularity. With this $\bar{S}$ and $\Upsilon = 1$ it is possible to write the system on the reduced form as in (4.13), but there is still the problem of an undesired dependency between the new momentum and the kinematic input ζ.

Step 3. Making the new momentum invariant with respect to the kinematic input

To further reduce the system by the application of theorem 4.3 $S = \bar{S}\Upsilon = R^T\Sigma\Upsilon$ has to satisfy the equation

$$S^T(\chi,\beta')R^T(\theta)\bar{M}(\beta')R(\theta)S(\chi,\beta') = 1, \tag{5.5}$$

which will make $\tilde{p}_1$ invariant with respect to the kinematic inputs.

To satisfy (5.5) a scalar nonzero function Υ must be chosen such that

$$\Sigma^T(\beta')\bar{M}(\beta')\Sigma(\beta')\Upsilon^2(\chi,\beta') = 1.$$

The choice of Υ is obviously

$$\Upsilon(\beta') = \frac{1}{\sqrt{\Sigma^T(\beta')\bar{M}(\beta')\Sigma(\beta')}}.$$

Having found an $S(\chi,\beta')$ that satisfies (5.5) the coordinate transformation $p \mapsto \tilde{p}_1, \tilde{p}_2$ is

$$\tilde{p}_1 = \Upsilon(\beta')\Sigma^T(\beta')R(\theta)p \qquad \text{1-dimensional,}$$
$$\tilde{p}_2 = C_1(\beta')R(\theta)p \qquad \text{2-dimensional,}$$

and the reduced Hamiltonian equivalent of the AV Lagrange model is

$$\left.\begin{aligned}\begin{bmatrix}\dot{\chi}\\ \dot{\tilde{p}}_1\end{bmatrix} &= J(\chi,\beta')\begin{bmatrix}\frac{\partial \tilde{H}}{\partial \chi}\\ \frac{\partial \tilde{H}}{\partial \tilde{p}_1}\end{bmatrix} + \begin{bmatrix}0\\ B_\phi(\beta')\end{bmatrix}\tau_\phi,\\ \dot{\beta}' &= \zeta,\end{aligned}\right\} \tag{5.6}$$

with

$$\begin{aligned}J(\chi,\beta') &= \begin{bmatrix}0 & R^T(\theta)\Sigma(\beta')\Upsilon(\beta')\\ -\Upsilon(\beta')\Sigma^T(\beta')R(\theta) & 0\end{bmatrix},\\ B_\phi(\beta') &= \Upsilon(\beta')\Sigma^T(\beta')C_2^T(\beta')\frac{1}{r_w},\\ \tilde{H}(\tilde{p}_1) &= \frac{1}{2}\tilde{p}_1^2.\end{aligned}$$

The original Lagrange model equations (3.11) have now been reduced to a set of first order equations. By using the nonholonomic constraints the problem has been reduced from a problem of solving three nonlinear second order differential equations, four nonlinear first order equations, and two linear first order equations, to a problem of only four nonlinear first order equations and two linear first order equations.

The system (5.6) is still highly nonlinear, however, and it is easy to lose track of the physical interpretation of the different elements in the system. To gain some insight into the physical nature of the system we look at two special cases where the system is expected to demonstrate a simpler, maybe even linear, behavior.

Special case 1. Driving along a straight line

When driving along a straight line all the steering motors are fixed at the same angle β_0, and the ICR lies at infinity in the direction perpendicular to the direction of motion, see figure 5.1.

The wheels are mounted at the corners of a $1 \times 1\mathrm{m}$ square, and the following relations hold: $\kappa_1 \cos\gamma_1 - \kappa_2 \cos\gamma_2 = 1$ and $\kappa_1 \sin\gamma_1 - \kappa_2 \sin\gamma_2 = 0$. In this configuration the components of the model hence reduce to much simpler forms

$$\Sigma(\beta_0) = \begin{bmatrix}\cos^2\beta_0\\ \sin\beta_0\cos\beta_0\\ 0\end{bmatrix}, \qquad \Upsilon(\beta_0) = \frac{1}{\cos\beta_0\sqrt{m_f + 4m_w + 4\frac{j_{w,\phi}}{r_w}}},$$

$$C_2(\beta_0) = \begin{bmatrix}\cos\beta_0 & \sin\beta_0 & \kappa_1\sin(\beta_0-\gamma_1)\\ \cos\beta_0 & \sin\beta_0 & \kappa_2\sin(\beta_0-\gamma_2)\\ \cos\beta_0 & \sin\beta_0 & \kappa_3\sin(\beta_0-\gamma_3)\\ \cos\beta_0 & \sin\beta_0 & \kappa_4\sin(\beta_0-\gamma_4)\end{bmatrix}.$$

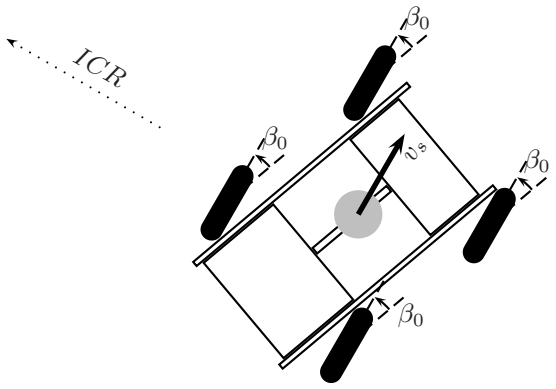


Figure 5.1: *Driving along a straight line. v_s is the velocity in the direction of motion*

The model (5.6) is then reduced to

$$\left.\begin{aligned} \dot{\chi} &= \begin{bmatrix} \cos(\theta+\beta_0) \\ \sin(\theta+\beta_0) \\ 0 \end{bmatrix} \frac{1}{\sqrt{m_f + 4m_w + 4\frac{j_{w,\phi}}{r_w}}} \tilde{p}_1, \\ \dot{\tilde{p}}_1 &= \frac{1}{\sqrt{m_f + 4m_w + 4\frac{j_{w,\phi}}{r_w}}} \sum_{i=1}^{4} \frac{\tau_{\phi_i}}{r_w}. \end{aligned}\right\} \tag{5.7}$$

The first equation describes how the AV moves in the direction $\theta + \beta_0$. As expected, there is no rotation of the AV, i.e., $\dot{\theta} = 0$. The equation can be rewritten in terms of the translational velocity v_s along the direction of motion

$$\dot{\chi} = \begin{bmatrix} \cos(\theta+\beta_0) \\ \sin(\theta+\beta_0) \\ 0 \end{bmatrix} v_s,$$

with $\tilde{p}_1$ and v_s related by

$$v_s = \frac{1}{\sqrt{m_f + 4m_w + 4\frac{j_{w,\phi}}{r_w}}} \tilde{p}_1. \tag{5.8}$$

With this change of coordinates the second equation of (5.7) becomes

$$m_s \dot{v}_s = \sum_{i=1}^{4} \frac{\tau_{\phi_i}}{r_w}, \qquad m_s = m_f + 4m_w + 4\frac{j_{w,\phi}}{r_w},$$

and we end up with a dynamic model that describes the acceleration of a mass m_s along a line with a constant slope $\theta + \beta_0$. m_s is the total mass of the AV plus the four moments

of inertia of the rolling wheels translated to 'masses' through the wheel radius r_w. Thus, m_s describes all the parts of the AV that are set in motion when driving along a straight line.

Special case 2. Rotation about the geometric center

In this situation all the wheels are oriented so that the ICR is located at the geometric center of the AV. In this case $\beta_1 = -\beta_2 = 3\pi/4$ and $\beta_4 = -\beta_3 = \pi/4$, see figure 5.2.

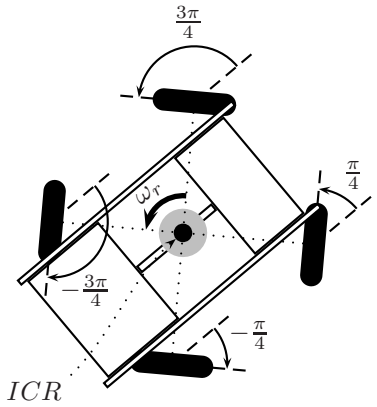


Figure 5.2: *Rotating around the geometric center. ω_r is the angular velocity about the geometric center*

As in the previous case the components of the model reduces to much simpler forms

$$\Sigma = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}, \qquad \Upsilon = \frac{1}{\sqrt{j_f + (m_w + \frac{j_{w,\phi}}{r_w}) \sum_{i=1}^{4} \kappa_i^2}},$$

$$C_2 = \begin{bmatrix} -\sqrt{2} & \sqrt{2} & \kappa_1 \\ -\sqrt{2} & -\sqrt{2} & \kappa_2 \\ \sqrt{2} & -\sqrt{2} & \kappa_3 \\ \sqrt{2} & \sqrt{2} & \kappa_4 \end{bmatrix},$$

and (5.6) is reduced to

$$\left.\begin{aligned}\dot{\chi} &= \begin{bmatrix}0\\0\\1\end{bmatrix} \frac{-1}{\sqrt{j_f + (m_w + \frac{j_{w,\phi}}{r_w}) \sum_{i=1}^{4} \kappa_i^2}} \tilde{p}_1, \\ \dot{\tilde{p}}_1 &= \frac{-1}{\sqrt{j_f + (m_w + \frac{j_{w,\phi}}{r_w}) \sum_{i=1}^{4} \kappa_i^2}} \sum_{i=1}^{4} \frac{\kappa_i \tau_{\phi_i}}{r_w}.\end{aligned}\right\} \tag{5.9}$$

As expected, there is no translational motion, i.e., $\dot{x}_1 = \dot{x}_2 = 0$. The first equation describes the rotation about the geometric center, and the rotation rate is related to the angular velocity ω_r by

$$\dot{\chi} = \begin{bmatrix}0\\0\\1\end{bmatrix} \omega_r.$$

In this case $\tilde{p}_1$ is related to ω_r through the relation

$$\omega_r = \frac{-1}{\sqrt{j_f + (m_w + \frac{j_{w,\phi}}{r_w}) \sum_{i=1}^{4} \kappa_i^2}} \tilde{p}_1.$$

Using this coordinate transformation the second equation in (5.9) is reduced to

$$j_r \dot{\omega}_r = \sum_{i=1}^{4} \frac{\kappa_i \tau_{\phi_i}}{r_w}, \qquad j_r = j_f + (m_w + \frac{j_{w,\phi}}{r_w}) \sum_{i=1}^{4} \kappa_i^2.$$

The result is similar to that of the previous case, but this time the equation describes an angular acceleration of a body with moment of inertia j_r. j_r is the total moment of inertia of the AV about its geometric center plus the moments of inertia of the rotating wheels, first translated through r_w to 'masses' at the wheel mounting points and then translated through κ_i to contributions to the total moment of inertia.

In each of the two special cases it is possible to relate the generalized momentum $\tilde{p}_1$ to either the translational or rotational velocity through the square root of the mass or moment of inertia. This is only valid for pure translational and pure rotational movement, and in general, when the AV is exhibiting both translational and rotational motion, $\tilde{p}_1$ is a combination of both.

5.2 Model Validation

The model (5.6) describes the dynamics of the AV with the assumption that all parameters in the model are constant and known. Most of the parameters can be measured

directly using a scale or a tape measure, but parameters, such as moments of inertia and friction coefficients, are somewhat harder to estimate. Based on empirical measurements the values of the parameters used in the model have been estimated and they are summarized in table 5.1.

Parameter		Value	Description
m_f	[kg]	187	Mass of frame (*)
j_f	[kg m^2]	95	Moment of inertia of frame (*)
m_w	[kg]	10	Mass of each wheel
$j_{w,\phi}$	[kg m^2]	0.5	Moment of inertia of a single wheel about the ϕ rotation axis
κ_i	[m]	$\sqrt{0.5}$	Polar position of the i'th wheel relative to the geometric center of the AV, $i = 1, 2, 3, 4$
γ_i	[rad]	$\frac{\pi}{4} + \frac{\pi}{2}(i-1)$	
r_w	[m]	0.23	Radius of wheels

Table 5.1: *Directly measurable AV model parameters. (*) taken from [7]*

To make the model complete, an estimate of the friction coefficient b' (see page 37) is also needed. Recall that the physical input to the propulsion motors are not torques, but voltages V_m related to τ_ϕ by

$$\tau_{\phi_i} = D_1 V_{m_i} - D_2(b')\dot{\phi}_i, \qquad i = 1, 2, 3, 4, \tag{5.10}$$

with $D_1 = K_t/R_a$ and $D_2(b') = K_e K_t/R_a + b'$. D_1 is the voltage to torque relationship of the motor at zero velocity, and an estimate of D_1 was found by Bisgaard et al. [7]. The parameter D_2 captures both the electromotive force generated in the motor and any kinetic friction affecting the motor. The internal kinetic friction of the unloaded motor was estimated by Bisgaard et al. [7], but it is assumed that the propulsion motors will be subject to an increased friction from external sources when driving in the field. The parameter D_2 is assumed to be equal for all four wheels.

Consider a situation where the AV is driving along a straight line as in special case 1. Combining (5.8) with (5.10) and applying the same voltage V_m to all four wheels yields the following linear first order system

$$m_s \dot{v}_s = \frac{4}{r_w}\left(D_1 V_m - \frac{D_2(b')}{r_w} v_s\right). \tag{5.11}$$

A similar approach can be taken for special case 2 where the AV is rotating about the geometric center. In this case the first order system is

$$j_r\dot{\omega}_r = \frac{1}{r_w}\sum_{i=1}^{4}\kappa_i\left(D_1V_m - \frac{\kappa_i D_2(b')}{r_w}\omega_r\right). \tag{5.12}$$

Figure 5.3 shows two measured step responses of the AV. The applied voltage to the propulsion motors are shown in the two top plots, and the resulting velocities are shown in the bottom plots. The input to each experiment consists of two identical steps, and

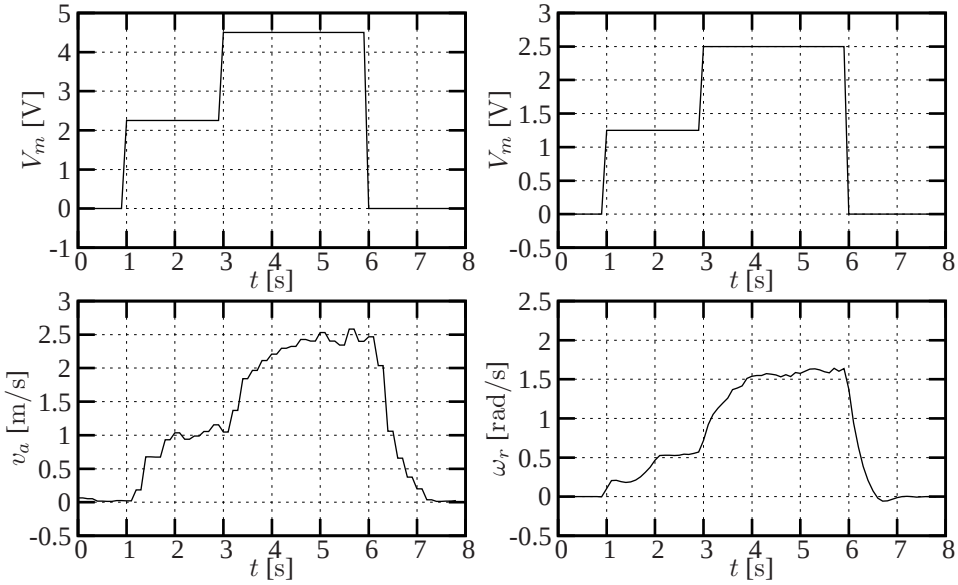


Figure 5.3: *Step response of special case 1 and 2*

if the dynamics of the AV are linear, we would expect two identical steps on the output as well. This is not the case though, as the first step in both experiments are lower than the second step. This indicates that there are additional unmodeled non-linearities in the system. This is most likely due to static friction, and this is therefore introduced in the model by the following modified input torque

$$\bar{\tau}_{\phi_i} = D_1V_{m_i} - D_2(b')\dot{\phi}_i - \tau_{s_i}(\phi_i, D_1V_{m_i}), \qquad i = 1,2,3,4,$$

where the static friction τ_{s_i} is determined by

$$\tau_{s_i} = \begin{cases} D_1V_{m_i} & \text{if} \quad \sum_{i=1}^{4} D_1V_{m_i} < 4\tau_{s0} \quad \text{and} \quad \dot{\phi}_i = 0, \\ \text{sign}(D_1V_{m_i})\tau_{s0} & \text{if} \quad \sum_{i=1}^{4} D_1V_{m_i} \geq 4\tau_{s0} \quad \text{and} \quad \dot{\phi}_i = 0, \\ \text{sign}(\phi_i)\tau_{s0} & \text{else.} \end{cases}$$

When the AV is not in motion the net torque is zero until the applied torque from the motors grows larger than the total static friction $4\tau_{s0}$. When the AV is in motion the static friction in each wheel is constant in the opposite direction of rotation. For simplicity the static friction during motion τ_{s0} is assumed to be constant, positive, and identical for all four wheels.

We are now ready to estimate the coefficient $D_2(b')$ and the static friction τ_{s0} from the step responses of figure 5.3. When the AV drives at constant non-zero velocity the models (5.11) and (5.12) including static friction reduces to

$$0 = D_1 V_m - \frac{D_2(b')}{r_w} v_s - \text{sign}(v_s)\tau_{s0},$$

and

$$0 = \sum_{i=1}^{4} \kappa_i \left(D_1 V_m - \frac{\kappa_i D_2(b')}{r_w} \omega_r - \text{sign}(\omega_r)\tau_{s0} \right).$$

These equations are linear equations in the two variables $D_2(b')$ and τ_{s0} when V_m, v_s, and ω_r are known. Four steady state velocities can be read from figure 5.3, and these readings can then be used to do a simple least square approximation of $D_2(b')$ and τ_{s0}. The resulting values are shown in table 5.2. Note that $D_2(b')$ will most likely change

Parameter		Value
D_1	[Nm/V]	18.2
$D_2(b')$	[Nm/rad/s]	6.5
τ_{s0}	[Nm]	11.6

Table 5.2: *Estimated friction parameters. (D_1 is derived from the data in table 2.10 on page 37).*

from one ground/soil condition to the next. In these experiments the AV was driving on gravel ground, and the same is true for any subsequent experiments unless otherwise specified.

With the estimates in tables 5.1 and 5.2 it is possible to compare the step responses of the physical system with those of the model. They are depicted in figure 5.4, and the model seems to capture the dynamics of the physical system in the two special cases well. The rise times of the model and the AV also fits nicely, which indicates that the estimated total mass and moment of inertia are correct.

To see how the model behaves in a more general setting the output of the model is compared with that of the AV during a 90s test run where the AV is driven manually by joystick. Figure 5.5 shows a block diagram of how the AV and model data are generated. The inputs V_m and β'_{ref} are generated by the joystick. The same V_m is passed directly

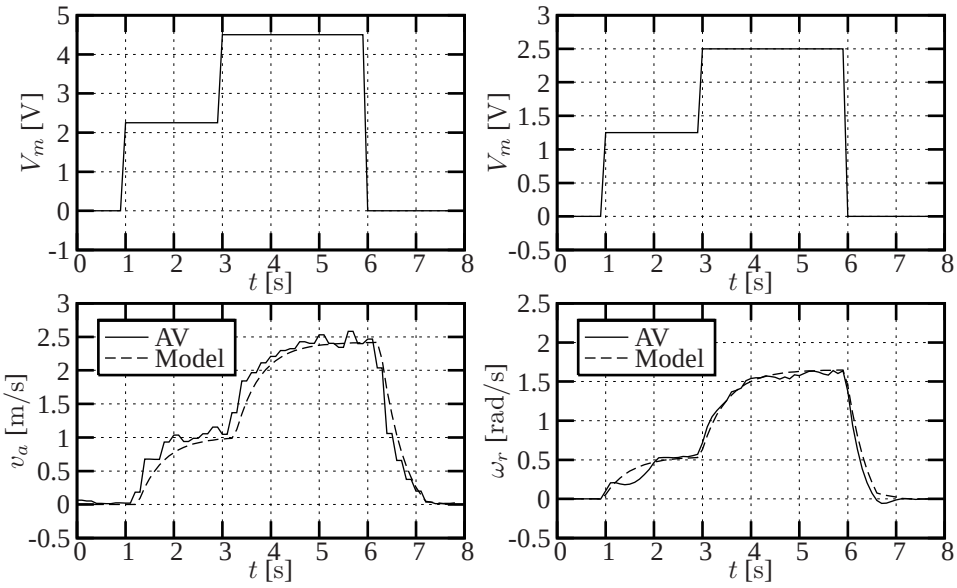


Figure 5.4: *AV and model step responses of special case 1 and 2*

to all four propulsion motors, but since the model expects four input torques, they have to be calculated based on the estimated parameters D_1, D_2, and τ_{s0}. Note that the kinematic input ζ does not enter the system explicitly. The joystick generates the desired steering angles β'_{ref}, and the software in the LH-Agros uses these references in the angular position control loops. Because ζ has been eliminated in the model equation (5.6) it suffices to pass the measured steering angles $\hat{\beta}'$ to the model.

Figure 5.6 shows the input V_m and the corresponding estimated torque input $\bar{\tau}_\phi$ to the model. The input is by no means simple and should in that sense excite most of the dynamics in the AV.

Figure 5.7 shows the measured steering angles fed into the model. Looking closely at the figure, it is possible to identify the varying ways the AV is being steered. Around $t = 11$s, for example, the steering angles are $\beta_1 = -\beta_2$ and the AV is rotating around an ICR that lies on the line passing through the side of the AV and the geometric center. A different mode of steering can be identified around $t = 70$s, where $\beta_1 = \beta_2 = -\pi/2$ and the AV is driving sideways with all four wheels fixed at the same angle.

Figure 5.8 shows the resulting translational velocity ($v_s = \sqrt{\dot{x}_1^2 + \dot{x}_2^2}$) of the AV (measured by the GPS receiver) and the model. The model captures the dynamics of the physical system well, even though there are some discrepancies between the two graphs;

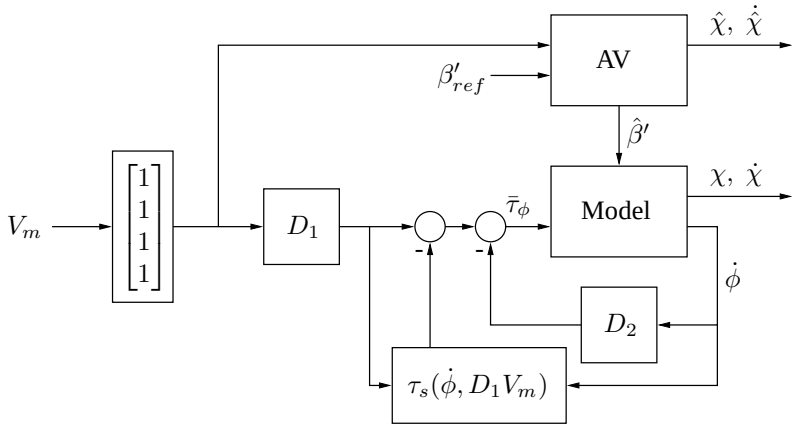


Figure 5.5: *Block diagram of the verification setup*

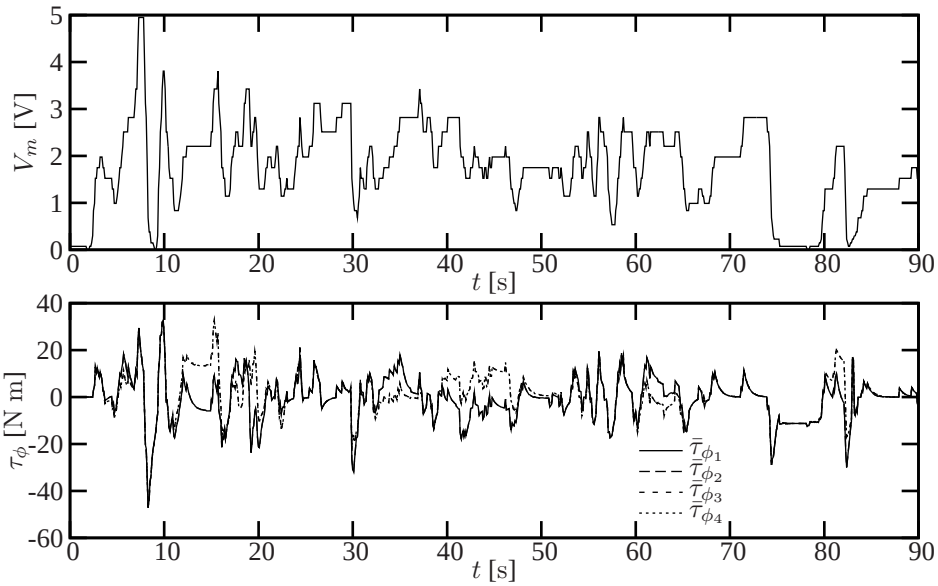


Figure 5.6: *Input voltage to the propulsion motors and the corresponding estimated torque input to the model*

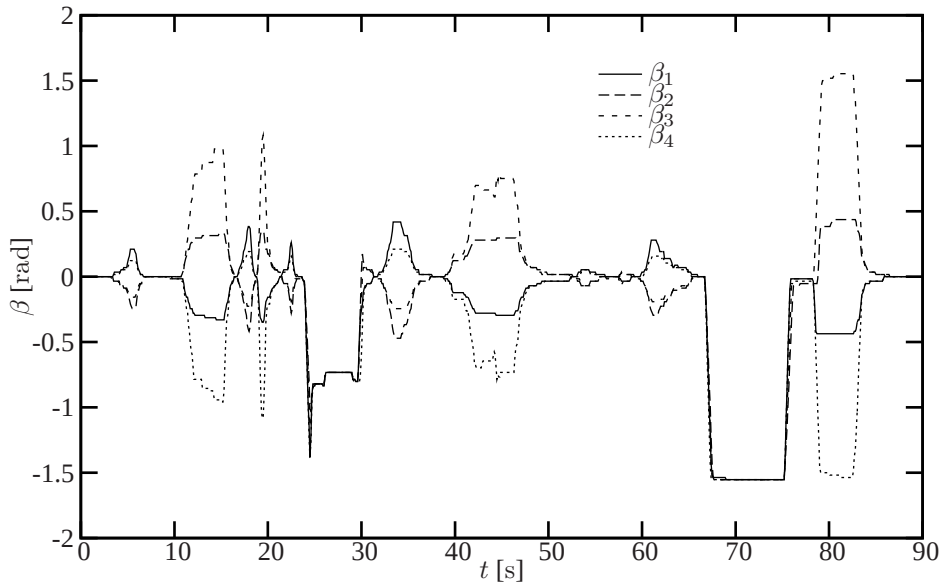


Figure 5.7: *Measured steering angles*

in the interval from 20 to 50 seconds the velocity seem to be a bit off at the peaks. A clearer picture is shown in figure 5.9 where a closeup of that interval is shown. Even though the model fails to hit the peaks of the measured velocity exactly it still captures the dominating dynamics of the system, and the discrepancies at the peaks are not large enough to invalidate the model.

The discrepancies between AV and model are even smaller when turning to the measured and simulated rotational velocities. Figure 5.10 shows the rotational velocities of the AV (measured by the gyro) and model during the full test, and figure 5.11 shows a closeup of the interval from 20 to 50 seconds.

Calculating the position and orientation of the AV based on the model is essentially a question of integrating the dashed graphs in figures 5.8 and 5.10, but due to the discrepancies between model and AV, the resulting modeled position and orientation are expected to diverge over time from those measured by the GPS and compass. Figures 5.12 and 5.13 shows the measured and simulated position and orientation of the AV. As expected, the position and orientation deviate over time (even though it is difficult to see for the orientation), but the model still captures the general behavior of the AV, and the deviation can easily be eliminated by implementing a suitable observer for the system. It is therefore concluded that the derived model of the AV describes the real

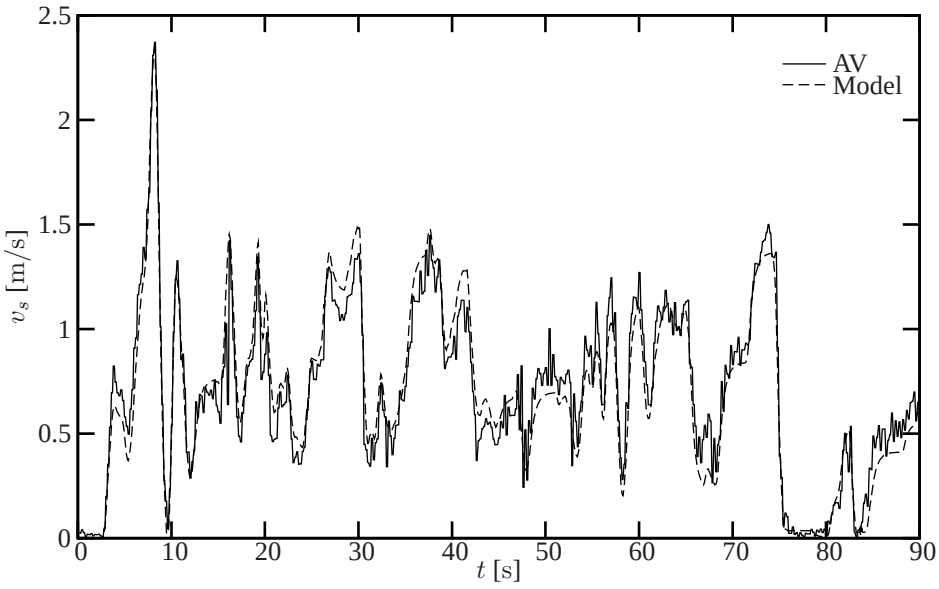


Figure 5.8: *Measured and simulated translational velocity*

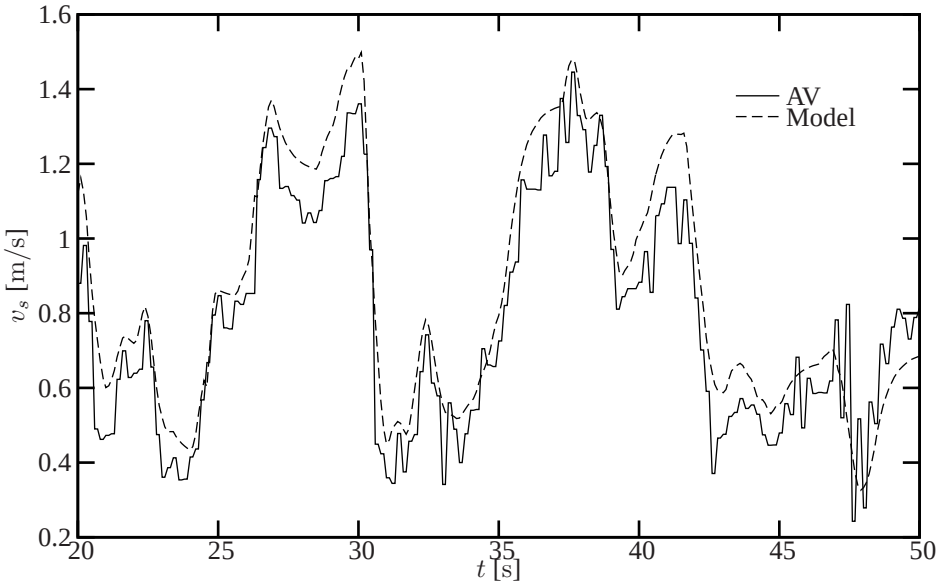


Figure 5.9: *Measured and simulated translational velocity (closeup)*

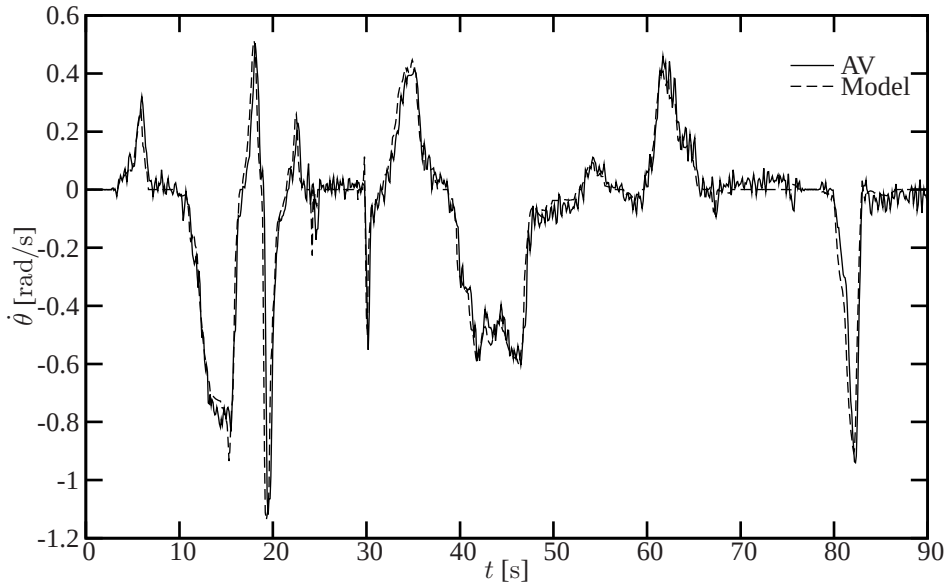


Figure 5.10: *Measured and simulated angular velocity of the AV*

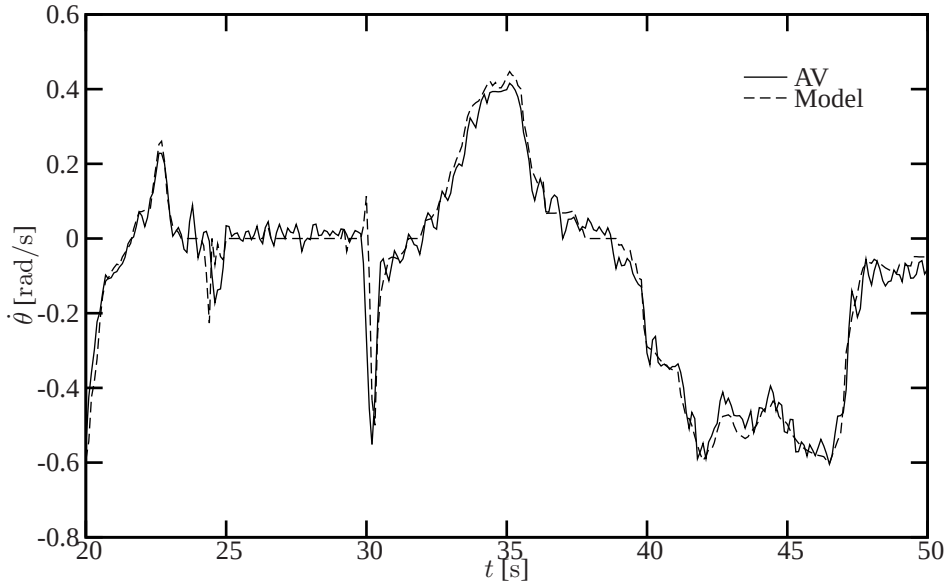


Figure 5.11: *Measured and simulated angular velocity of the AV (closeup)*

dynamics of the physical system to an extent that makes the model suitable for deriving and simulating control algorithms for the AV.

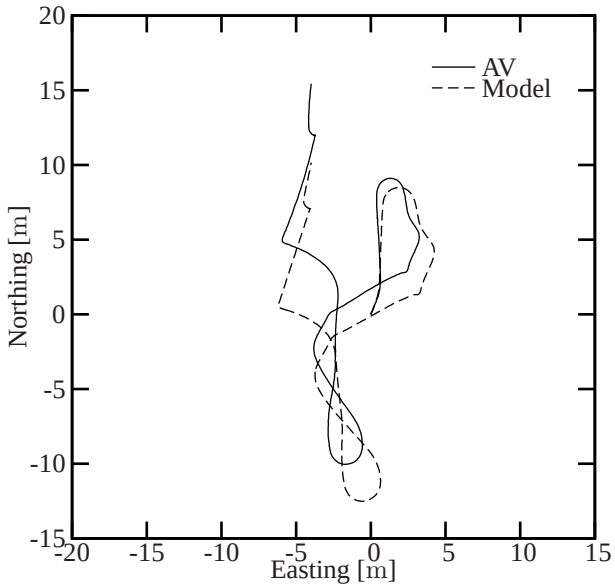


Figure 5.12: *Comparison between measured and simulated position*

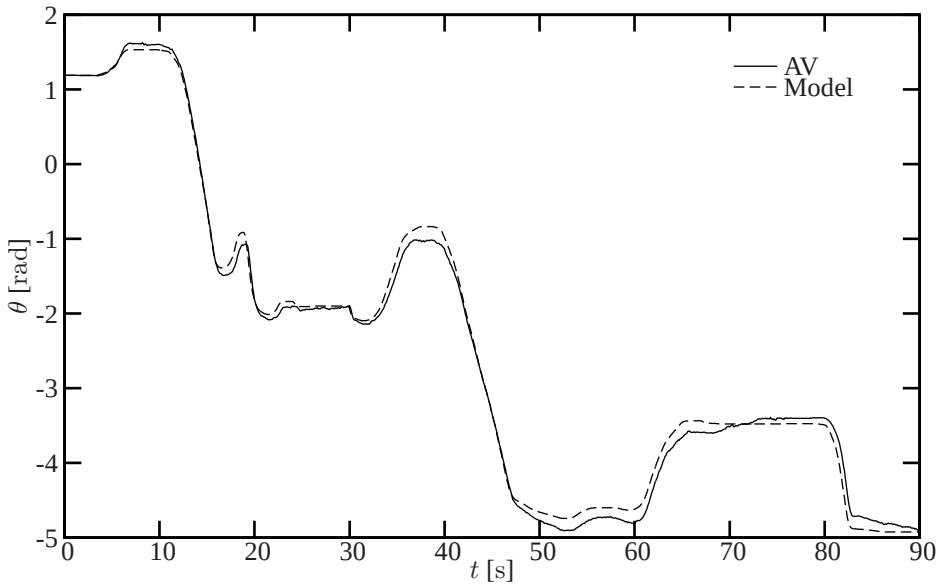


Figure 5.13: *Comparison between measured and simulated orientation*

CHAPTER 6

FEEDBACK CONTROL OF SYSTEMS WITH KINEMATIC INPUTS

A general model of a nonholonomic Hamiltonian system with kinematic inputs was introduced in chapter 4, and we now turn to feedback control of that system. This chapter focuses on asymptotic stabilization of the system by means of *energy shaping* and *damping injection*. The two concepts are, in their simplest form, smooth time invariants feedbacks, but Brockett and Sussmann [14] proved that smooth time invariant feedbacks alone cannot asymptotically stabilize a nonholonomic system. The system will, however, converge to an open subset of the configuration manifold [40]. The kinematic inputs constitutes an additional degree of control, and these inputs can be used to force the set of convergence to only containing a set of desired stable equilibria and asymptotic stability can be achieved.

The control methods described in this chapter are based on passivity properties of dynamic systems, and a short introduction to the passivity approach is valid before moving to the control aspects.

6.1 Passivity of Dynamic Systems

Consider the general nonlinear dynamic system

$$\Pi : \begin{cases} \dot{x} = f(x,u) \\ y = h(x,u) \end{cases},$$

with $x \in \mathcal{X}$, $u \in \mathcal{U} \subset \mathbb{R}^m$ and $y \in \mathcal{Y} \subset \mathbb{R}^k$. $\mathcal{X}$ is an n-dimensional manifold. Define the *supply rate* $s : \mathcal{U} \times \mathcal{Y} \to \mathbb{R}$ and a continuous differentiable *storage function* $V : \mathcal{X} \to \mathbb{R}^+$. $\mathbb{R}^+$ denotes all non-negative real values[1].

Definition 6.1. *The dynamic system* Π *is said to be* passive *with respect to the supply rate* s *if there exists a storage function* V*, such that for any initial condition* $x(0) \in \mathcal{X}$ *and for all* $t_1 > 0$ *the passivity inequality is satisfied*

$$V(x(t_1)) - V(x(0)) \leq \int_0^{t_1} s(u(t), y(t))dt. \tag{6.1}$$

This is a quite general definition of passivity, and for mechanical systems it merely states that the system is passive if the increase in stored energy from $t = 0$ to $t = t_1$ is never greater than the amount of energy supplied externally to the system; the system cannot, by itself, generate energy.

In electrical and mechanical systems a useful choice of storage function is usually the total physical energy of the system, and the supply rate is then usually chosen to be the instantaneous externally applied power to the system. One important supply rate is defined as follows. Consider an output space defined as the dual of the input space $\mathcal{Y} = \mathcal{U}^*$. Then we can define a supply rate

$$s(u,y) = y^T u.$$

This type of supply rate is encountered in many physical systems, and it often has a clear physical meaning. In electrical systems u and y may be voltages and currents, and in mechanical system u and y may be generalized forces and velocities. Other choices of inputs and outputs exist, but if the storage function is real energy the product between input and output must equal power.

The passivity inequality (6.1) can also be written in terms of the instantaneous change of the storage function. Taking the time derivative on both sides yields

$$\dot{V}(x(t)) \leq s(u(t), y(t)).$$

[1] The storage function only have to be bounded from below, but without loss of generality we assume that it is strictly non-negative.

This inequality states that the system is passive if the rate of change of the storage function at any given time instance is never larger than the supply rate. If the relation is a strict equality the system is said to be *conservative*. This formulation of the passivity condition can be directly applied to show that the Hamiltonian system with kinematic inputs (4.17) is a conservative system. Take the Hamiltonian function $\tilde{H}$ as storage function and look at the time derivative

$$\begin{aligned}\dot{\tilde{H}} &= \begin{bmatrix} \frac{\partial \tilde{H}}{\partial q} \\ \frac{\partial \tilde{H}}{\partial \tilde{p}_1} \end{bmatrix}^T \left(J \begin{bmatrix} \frac{\partial \tilde{H}}{\partial q} \\ \frac{\partial \tilde{H}}{\partial \tilde{p}_1} \end{bmatrix} + \begin{bmatrix} 0 \\ S^T B \end{bmatrix} u \right) \\ &= \frac{\partial^T \tilde{H}}{\partial \tilde{p}_1} S^T B u.\end{aligned}$$

Defining the output y as

$$y = B^T S \frac{\partial \tilde{H}}{\partial \tilde{p}_1} \quad (= B^T \dot{q}),$$

yields

$$\dot{\tilde{H}} = y^T u. \tag{6.2}$$

Hence, with storage function $\tilde{H}$ the system is passive (and conservative) with respect to the supply rate $s(u, y) = y^T u$ with y defined as above. The choice of output may seem random, but it usually has a clear physical interpretation. On the AV, for example, the inputs are the four propulsion motor torques τ_ϕ, and the output y, if defined as above, turns out to be the angular velocities $\dot{\phi}$ of the propulsion motors. The supply rate is then equal to the total power supplied by the propulsion motors. Since the constraint forces are not doing any work on the system, and since there are not yet any non-conservative friction forces in the model, the supply rate is exactly equal to the rate of change of total energy of the AV.

6.2 Example of Passivity Based Control

The passivity of (4.17) does not automatically infer stability of the system. The states of a passive system will not grow exponential, but if the uncontrolled system is conservative, and has a nonzero initial momentum, the system will never come to rest either. Even if energy dissipation or non-conservative forces are introduced in the system there is no guarantee that the system will come to rest at the desired equilibrium. The equilibrium point, or set of equilibrium points, are given by the minima of the potential energy function U, and the equilibrium points may, or may not, coincide with the desired set of equilibrium points. To solve the problems of having an undesired potential energy function and the absence of energy dissipation, the concepts of *potential energy shaping*

Figure 6.1: *A mass moving on a frictionless surface*

and *damping injection* are introduced. Before moving to the general situation, we first consider the simple linear example in figure 6.1 of a mass m moving on a flat frictionless horizontal surface.

The system can be written as a Hamiltonian system

$$\begin{aligned}\begin{bmatrix}\dot{x}\\ \dot{p}\end{bmatrix} &= \begin{bmatrix}0 & 1\\ -1 & 0\end{bmatrix}\begin{bmatrix}\frac{\partial H}{\partial x}\\ \frac{\partial H}{\partial p}\end{bmatrix} + \begin{bmatrix}0\\ F\end{bmatrix},\\ y &= \frac{\partial H}{\partial p} \quad (= \dot{x}),\\ H(p) &= \frac{1}{2}m^{-1}p^2,\end{aligned}$$

where x is the position of the mass along the horizontal, p is the momentum of the mass, and F is the applied input force.

Assume that we wish to asymptotically stabilize the system at the origin $x = 0$. Depending on the initial momentum, the mass will either move at a constant velocity along the horizontal if $p(0) \neq 0$, or it will stay at its initial position if $p(0) = 0$. The lack of potential energy in the Hamiltonian function means that any initial point on the horizontal plane is an equilibrium point if $p(0) = 0$, but we are only interested in one: $x = 0$. To eliminate the rest we define a function $U(x)$ that has a global minimum at $x = 0$ and the following control law

$$F = -\frac{\partial U}{\partial x} + \bar{F},$$

where $\bar{F}$ is the new input. The feedback applies the negative gradient of the function U to the mass, thereby always pulling the mass toward the minimum, see figure 6.2.

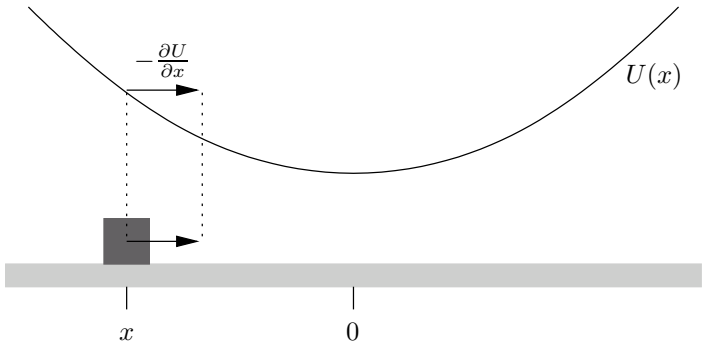


Figure 6.2: *Energy shaping*

The resulting closed loop Hamiltonian system with the new input is

$$\begin{bmatrix}\dot{x}\\ \dot{p}\end{bmatrix} = \begin{bmatrix}0 & 1\\ -1 & 0\end{bmatrix}\begin{bmatrix}\frac{\partial \bar{H}}{\partial x}\\ \frac{\partial \bar{H}}{\partial p}\end{bmatrix} + \begin{bmatrix}0\\ \bar{F}\end{bmatrix},$$
$$y = \frac{\partial \bar{H}}{\partial p},$$
$$\bar{H}(x,p) = \frac{1}{2}m^{-1}p^2 + U(x).$$

The system is of the exact same structure as before, but now with a *shaped* potential energy. Had there been any *real* potential energy in the original system the shaped potential energy would just have been the sum of the original *real* potential energy and the new *artificial* potential energy. Examining the new system we see that the only equilibrium point is $(x,p) = (0,0)$.

The system is still not asymptotically stable. Because of energy conservation ($\bar{H}$ is constant if $\bar{F} = 0$) the introduction of the artificial potential energy has resulted in a marginally stable system, which oscillates about the origin. To make the system asymptotically stable the origin must not only be an equilibrium of the system, but also an asymptotically stable one. Take the positive definite Hamiltonian function $\bar{H}$ as a Lyapunov function candidate. The function is positive definite in a neighborhood of $(0,0)$, and the time derivative of it is

$$\dot{\bar{H}} = y\bar{F}.$$

Choosing $\bar{F} = -k_d y,\ k > 0$ renders $\dot{\bar{H}} \leq 0$. The function $\dot{\bar{H}}$ is not identically negative outside the origin, and we turn to LaSalles invariance principle to determine the asymptotic behavior of the system.

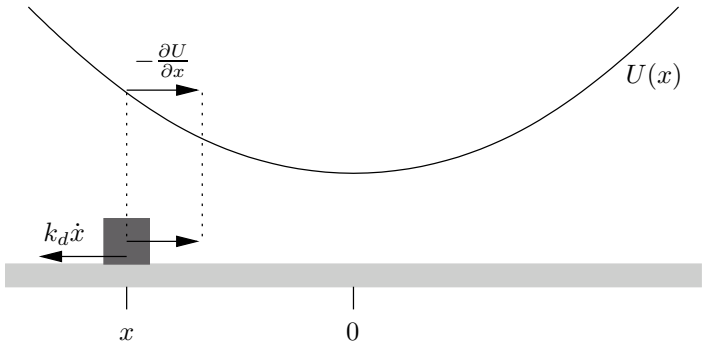


Figure 6.3: *Energy shaping + Damping injection*

Theorem 6.1 (LaSalle's Invariance Principle). *Let $\Omega \subset D$ be a compact set that is positively invariant with respect to*

$$\dot{x} = f(x),$$

where $f : D \to \mathbb{R}^n$. Let $V : D \to \mathbb{R}$ be a continuously differentiable function such that $\dot{V}(x) \leq 0$ in Ω. Let E be the set of all points in Ω where $\dot{V}(x) = 0$. Let Q be the largest invariant set in E. Then every solution starting in Ω approaches Q as $t \to \infty$.

Proof. See [35] □

We conclude that the system converges asymptotically to the largest invariant set of the system where $\dot{\bar{H}} = 0$, that is, when $\dot{x} = 0$. The set is identical to the single point $(0, 0)$ and the origin is hence asymptotically stable.

This particular choice of $\bar{F}$ is called *damping injection*, and in the simple example the $\bar{F}$ feedback can be compared to the presence of a non-conservative friction force between the mass and the ground, see figure 6.3.

The resulting closed loop system is

$$\left.\begin{aligned} \begin{bmatrix} \dot{x} \\ \dot{p} \end{bmatrix} &= \left(\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ 0 & k_d \end{bmatrix} \right) \begin{bmatrix} \frac{\partial \bar{H}}{\partial x} \\ \frac{\partial \bar{H}}{\partial p} \end{bmatrix}, \\ \bar{H}(x,p) &= \frac{1}{2} m^{-1} p^2 + U(x). \end{aligned} \right\} \tag{6.3}$$

The responses of the original system, the system with energy shaping, and the system with both energy shaping and damping injection are shown in figure 6.4. The system is in

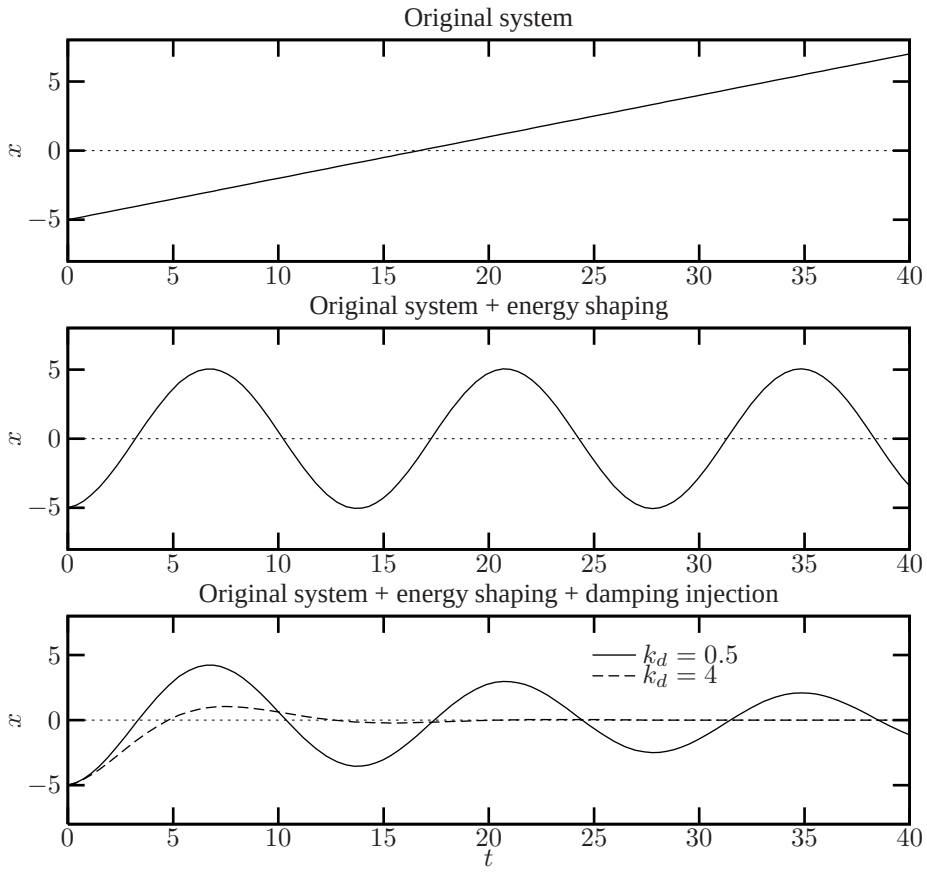


Figure 6.4: *Responses of the simple example*

all three cases started with a nonzero momentum at $x = -5$. As expected, the response of the uncontrolled system in the top figure continues to grow at constant velocity. In the middle figure energy shaping is introduced, and the position no longer grows linearly, but oscillates around the origin. In the bottom figure damping injection is introduced with two different damping coefficients, and the responses converge asymptotically to zero at different rates; the higher the damping, the faster the system settles.

If the potential energy function is defined as a quadratic function $U(x) = \frac{1}{2}k_p x^2,\ k_p > 0$, the total feedback is $F = -k_p x - k_d \dot{x}$. One could argue that this example is just a complicated way of deriving a simple linear PD controller, but the example is meant as a brief introduction to the physical interpretation of energy shaping and damping

injection. A PD controller is by construction linear, but the concepts of energy shaping and damping injection works equally well for a large class of nonlinear systems.

Remark. The damped system (6.3) is an example of a dissipative Hamiltonian system. A more general form is

$$\begin{bmatrix}\dot{q}\\ \dot{p}\end{bmatrix} = [J - D]\begin{bmatrix}\frac{\partial H}{\partial q}\\ \frac{\partial H}{\partial p}\end{bmatrix} + \begin{bmatrix}0\\ B\end{bmatrix} u,$$
$$y = B^T \frac{\partial H}{\partial p}.$$

This system differs from the conservative Hamiltonian system by the introduction of a positive semi-definite damping matrix D. The system still satisfies the passivity condition

$$\dot{H} = y^T u - \frac{\partial^T H}{\partial p} D \frac{\partial H}{\partial p} \qquad \Rightarrow \qquad \dot{H} \leq y^T u,$$

but it is no longer conservative. △

6.3 Energy Shaping

To put the concept of energy shaping into a more general context we look at the Hamiltonian system with kinematic inputs from chapter 4. The system equations are repeated below for convenience

$$\left.\begin{aligned}\begin{bmatrix}\dot{q}\\ \dot{\tilde{p}}_1\end{bmatrix} &= J(q,\tilde{p}_1,r)\begin{bmatrix}\frac{\partial \tilde{H}}{\partial q}\\ \frac{\partial \tilde{H}}{\partial \tilde{p}_1}\end{bmatrix} + \begin{bmatrix}0\\ S^T(q,r)B(q,r)\end{bmatrix} u,\\ y &= B^T(q,r)S(q,r)\frac{\partial H}{\partial \tilde{p}_1},\\ \tilde{H}(q,\tilde{p}_1) &= \frac{1}{2}\tilde{p}_1^T\tilde{p}_1 + U(q),\end{aligned}\right\} \tag{4.17}$$

with a skew symmetric interconnection matrix

$$J(q,\tilde{p}_1,r) = \begin{bmatrix}0 & S(q,r)\\ -S^T(q,r) & \left(-\tilde{p}_1^T S^T(q,r) M(q,r)[S_i,S_j](q)\right)_{i,j}\end{bmatrix}.$$

As already seen, the system is passive (and conservative) with respect to supply rate $s(u,y) = y^T u$ and storage function $\tilde{H}$. That is,

$$\dot{\tilde{H}} = y^T u,$$

and the set of equilibrium points $(\dot{q}, \dot{\tilde{p}}_1) = 0$ of this system with a fixed r is $\{(q,0) \in \Omega_c^* \mid S^T(q,r)\frac{\partial U}{\partial q} = 0\}$. The extreme points of the potential energy function are equilibrium points of the system, but so is any point q for which the gradient $\frac{\partial U}{\partial q}$ lies in the kernel of S^T. This is one of the direct consequences of having a nonholonomic system, and the problem will receive more attention later. For now we focus on shaping the potential energy function so that the extreme points of the shaped potential function coincides with a desired set of equilibrium points.

Suppose the point, or the set of points, Q_0 that we want the system to approach can be represented as local minima of a known potential function $U + \bar{U}$, where $U(q)$ is the original *real* potential energy, and $\bar{U}(q)$ is the *artificial* or *shaping* potential energy of the designers choice. It is assumed that the artificial potential function $\bar{U}$ can be designed in such a way that $U + \bar{U}$ are strictly non-negative and that the set of extreme points only comprises a closed set of minimum points. The set Q_0 is then defined as

$$Q_0 = \{(q,0) \in \Omega_c^* \mid \frac{\partial (U + \bar{U})}{\partial q} = 0\}.$$

The object of energy shaping is to find a feedback that will add $\bar{U}$ to the original potential energy of the system, thereby shaping the total potential. Maschke and van der Schaft [40] proposed an input that shapes the potential energy for nonholonomic systems without kinematic inputs, but it applies equally well for systems with kinematic inputs. Consider an input u_{es} satisfying

$$-S^T(q,r)\frac{\partial \bar{U}}{\partial q} = S^T(q,r)B(q,r)u_{es}. \tag{6.4}$$

Inserting the input $u = u_{es} + \bar{u}$ into (4.17) yields the modified Hamiltonian system

$$\begin{bmatrix} \dot{q} \\ \dot{\tilde{p}}_1 \end{bmatrix} = J(q,r,\tilde{p}_1) \begin{bmatrix} \frac{\partial \bar{H}}{\partial q} \\ \frac{\partial \bar{H}}{\partial \tilde{p}_1} \end{bmatrix} + \begin{bmatrix} 0 \\ S^T(q,r)B(q,r) \end{bmatrix} \bar{u}, \tag{6.5}$$

where $\bar{H}$ is similar to the original $\tilde{H}$, but now with shaped potential energy

$$\bar{H}(q,r,\tilde{p}_1) = \tilde{H}(q,r,\tilde{p}_1) + \bar{U}(q).$$

The set of equilibrium points of the new system has thus been modified to include all the minimum points of the desired potential function $U + \bar{U}$

$$Q = \{(q,0) \in \Omega_c^* \mid S^T(q,r)\frac{\partial (U + \bar{U})}{\partial q} = 0\}. \tag{6.6}$$

6.4 Damping Injection

The introduction of artificial potential energy has not changed the conservative nature of the original system. The time derivative of the new storage function $\bar{H}$ still equals the supply rate, that is, $\dot{\bar{H}} = y^T \bar{u}$. If we consider the new Hamiltonian function as a Lyapunov function candidate the time derivative should be rendered negative to guarantee asymptotic stability. The obvious choice of feedback that will render $\dot{\bar{H}}$ non-positive is

$$\bar{u} = -Ky, \qquad K > 0, \tag{6.7}$$

with $K \in \mathbb{R}^{m \times m}$. $\dot{\bar{H}}$ only becomes negative *semi*definite, but from theorem 6.1 we know that the system states will asymptotically converge to the largest invariant set where $\dot{\bar{H}} = 0$. The structure of the system implies that if $y = 0$ then $\tilde{p}_1 = 0$, and the largest invariant set is exactly Q.

The closed loop system with energy shaping and damping injection can be written on the simple form

$$\begin{bmatrix} \dot{q} \\ \dot{\tilde{p}}_1 \end{bmatrix} = [J(q, r, \tilde{p}_1) - D(q, r)] \begin{bmatrix} \frac{\partial \bar{H}}{\partial q} \\ \frac{\partial \bar{H}}{\partial \tilde{p}_1} \end{bmatrix}, \tag{6.8}$$

with an unchanged interconnection matrix J and a positive semi-definite dissipation matrix

$$D(q, r) = \begin{bmatrix} 0 & 0 \\ 0 & S^T(q, r) B(q, r) K B^T(q, r) S(q, r) \end{bmatrix}.$$

Remark. The controlled system is an example of a non-conservative, or dissipative, Hamiltonian system. There was no energy dissipation in the original system, but it was introduced by feeding back the outputs. If the original system had inherent dissipative elements – like friction in mechanical systems or resistive elements in electrical circuits – they can usually be modeled by a similar dissipation matrix. Feeding back the outputs would then result in either an increased dissipation if $K > 0$, or a decreased dissipation if $K < 0$. As long as the total resulting dissipation matrix is positive semi-definite the system remains stable. △

6.5 Asymptotic Stability

So far, energy shaping and damping injection have transformed the original system into a system that converges asymptotically to the set Q. This set contains the desired set of convergence Q_0, but Q is generally larger than Q_0, which is a direct result of the nonholonomic nature of the system. To visualize this consider the simple example of a

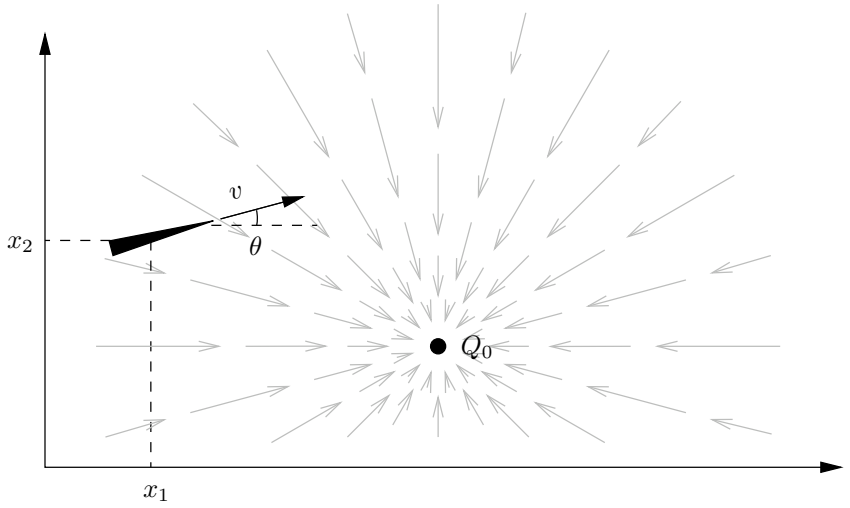


Figure 6.5: *A knife's edge moving in a potential field*

knife's edge with mass m moving in a potential field, see figure 6.5. The knife's edge is only allowed to move in the direction it is pointing, and this constraint renders the system nonholonomic. The gray arrows in the figure represent the negative gradients of a potential function U that has a global minimum at Q_0.

Suppose the knife's edge has negligible moment of inertia and that the rotation rate can be controlled directly through its first derivative. We choose this as a kinematic input $\dot{\theta} = \zeta$ and choose $q = [x_1 \; x_2]^T$ as generalized coordinates. The generalized momentum is then $p = m\dot{q}$ and the Hamiltonian function is

$$H(q,p) = \frac{1}{2m} p^T p + U(q).$$

The system is subject to the nonholonomic constraint

$$\dot{q} = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} v \quad \Rightarrow \quad \begin{bmatrix} \sin\theta & -\cos\theta \end{bmatrix} \dot{q} = 0.$$

The unreduced system can then be written as a Hamiltonian system

$$\begin{bmatrix} \dot{q} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix} \begin{bmatrix} \frac{\partial H}{\partial q} \\ \frac{\partial H}{\partial p} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \sin\theta \\ -\cos\theta \end{bmatrix} \lambda.$$

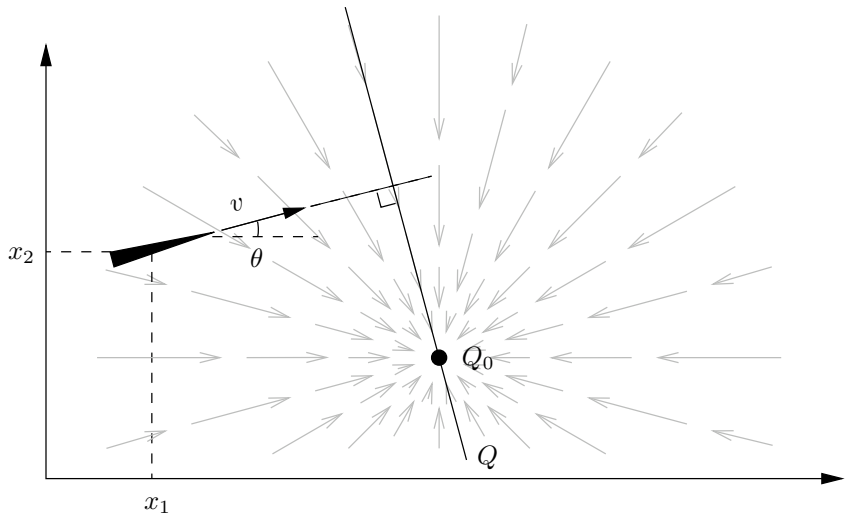


Figure 6.6: *Set of equilibrium points of the knife's edge example*

To eliminate the Lagrange multipliers we choose $S(\theta) = [\cos\theta \ \ \sin\theta]^T$ and define the new momentum

$$\tilde{p}_1 = \cos\theta p_{x_1} + \sin\theta p_{x_2}.$$

This is exactly the momentum along the direction of motion, and with this change of coordinates the system can be rewritten on the reduced form

$$\begin{bmatrix} \dot{q} \\ \dot{\tilde{p}}_1 \end{bmatrix} = \begin{bmatrix} 0 & S(\theta) \\ -S^T(\theta) & 0 \end{bmatrix} \begin{bmatrix} \frac{\partial \tilde{H}}{\partial q} \\ \frac{\partial \tilde{H}}{\partial \tilde{p}_1} \end{bmatrix},$$

$$\tilde{H}(q, \tilde{p}_1) = \frac{1}{2m}\tilde{p}_1^2 + U(q).$$

Using (6.6), the set of equilibrium points is described by

$$Q = \{(q, 0) \in \Omega_c^* \mid \begin{bmatrix} \cos\theta & \sin\theta \end{bmatrix} \frac{\partial U}{\partial q} = 0\}. \tag{6.9}$$

At a fixed θ the set is defined by the union of all points on the manifold with associated gradients $\frac{\partial U}{\partial q}$ that are perpendicular to the direction of motion. The set is illustrated in figure 6.6 for a fixed θ.

Without changing θ the knife's edge would move toward the intersection between its own line of motion and Q and not toward Q_0 as desired. In this simple example it is easy to see what could be done to make Q_0 an asymptotically stable equilibrium. We

could for example use the kinematic input to make sure that the gradient $\frac{\partial U}{\partial q}$ always lies in the image of S^T. This is the same as saying that the knife's edge should always be pointing in the same direction as the gradient. In this example the only solution to (6.9) would then be the trivial one $\frac{\partial U}{\partial q} = 0$, and asymptotic convergence to the desired point Q_0 is achieved.

The same rationale can be used for the general situation.

Theorem 6.2. *Consider the feedback controlled system* (6.8)*. Let n be the dimension of the configuration manifold $\mathcal{M}$ and let $S(q, r)$ be full rank and defined according to* (4.8)*. Furthermore, let the shaped potential energy $U(q) + \bar{U}(q)$ be a smooth function whose extreme points only comprises a closed set Q_0 of minima. If*

$$\bigcup_{r \in \mathcal{R}} \textit{img}[S(q, r)] = \mathbb{R}^n, \qquad q \in \mathcal{M} \tag{6.10}$$

there always exists a reference for the coordinates r related to the kinematic inputs v that renders the system asymptotically stable at Q_0.

Proof. There has been made no assumptions on the size of $\frac{\partial (U+\bar{U})}{\partial q}$, and in general it can lie anywhere in $\mathbb{R}^n$. If (6.10) is true it is always possible to find at least one r_q that satisfies

$$\frac{\partial (U + \bar{U})}{\partial q} \in \text{img}[S(q, r_q)] = \ker[A^T(q, r_q)], \qquad q \in \mathcal{M}. \tag{6.11}$$

Using this r_q as reference for the kinematic control of r the solution of (6.6) is the trivial one, i.e, $Q = Q_0$ and asymptotic stability is achieved. □

From an energy perspective the situation can be interpreted as follows. Consider the gradient as a generalized force $F_U(q) = -\frac{\partial (U+\bar{U})}{\partial q}$. This is the force that pulls the system toward the set Q_0, but in order to do any work on the system the force has to lie in the space of allowed velocities. The nonholonomic constraint forces will partially or completely cancel it if it does not. Since the space of allowed velocities is img$[S(q, r)]$, (6.11) implies that a nonzero generalized force $F_U(q)$ is guaranteed to do work on the system and pull it toward Q_0.

The requirement (6.10) implies that a suitable r_q for $q \in \mathcal{M}$ is only guaranteed to exist if it possible to go in any direction on $T_q\mathcal{M}$ by changing r. The requirement is satisfied for the simple knife's edge example and also for the AV as we shall see in the next chapter. In the general situation though, the requirement may not be satisfied. Consider for example a car like vehicle, where the angle of the steering axle is controlled by a kinematic input. No matter how the steering wheels are oriented, the vehicle cannot drive sideways.

6.6 Discussion

The general problem, when trying to asymptotically stabilize nonholonomic systems, is that the constrained tangent space seldom 'points in the right direction'. That is, it does not point toward Q_0, but toward the bigger set Q that includes Q_0. By introducing the kinematic inputs we are able to redirect the constrained tangent space so that it always points toward Q_0.

In chapter 4 we defined the kinematic inputs as time derivatives of a part of the configuration coordinates, but the inputs could have been introduced as external inputs as well. However the inputs are introduced they have to satisfy $\frac{\partial T}{\partial r} = 0$. The trick is to leave the coordinates that we wish to control as regular generalized coordinates and use the part related to the kinematic inputs to stabilize q, i.e., stabilizing r is of no concern and they are used only as a tool to stabilize q. If $U + \bar{U}$ is a smooth function and S is full rank for any q and r, and if each entry in S is a smooth function, then the references r_q will also be smooth functions, and it is possible to design smooth time-invariant feedbacks for the kinematic inputs. The energy shaping and damping injecting feedback is also smooth and time-invariant, and this implies that we have achieved global asymptotic stability of a nonholonomic system using a smooth time-invariant feedback. This is *not* in contradiction with the results by Brockett and Sussmann [14] who proved that a nonholonomic system cannot be asymptotically stabilized by a smooth time-invariant feedback. In systems with kinematic inputs we only consider asymptotic convergence of the q coordinates and not the r coordinates related to the kinematic inputs. Controlling the r coordinates is used to asymptotically stabilize the q coordinates, but once $q \in Q_0$ the references to the kinematic inputs are undefined, or they will be defined by the direction of the gradient when close to Q_0, i.e., the final position of r is determined by the initial configuration of the system. Of cause, r can be changed to any value by the kinematic input when $q \in Q_0$, since changing r will not change q, but this involves a non-smooth switching. In this sense the feedback algorithm presented in this chapter does not violate the results by Brockett and Sussmann [14].

The passivity of the closed loop system implies that it is also robustly stable with respect to parameter variations in the inertia matrix (these are often the parameters that are hardest to estimate). As long as the net damping is positive the Hamiltonian function will decrease, and the system will eventually stop when all the energy has been dissipated. The performance, on the other hand, will most likely be sacrificed, since it is determined by the shape of the potential shaping function, which should be designed with the estimated inertia matrix in mind, i.e, the rate of convergence toward Q_0 is determined by the steepness of the shaping function. This is an issue that deserves some future attention, but it will not be considered further in this thesis. It would require that a general definition of performance in Hamiltonian systems is developed, or at least a definition

that can be applied to nonholonomic Hamiltonian systems with kinematic inputs, but performance is an issue that is generally difficult to handle for nonlinear systems.

CHAPTER 7

FEEDBACK CONTROL OF THE AV

The energy shaping and damping injecting feedbacks will now be applied to the AV. Under normal operation the AV receives a set of way-points that it has to reach in sequence. When driving in between way-points the AV has to do as little damage to the crop as possible. This implies that is should follow the crop rows and the wheels should follow the space between the crop rows.

The method introduced in the previous chapter is used to asymptotically stabilize the AV toward either a single (way-)point or a path (crop row) in the field. Both kinds of convergence can be achieved with the same controller by changing the shape of the potential energy. In the simple linear example introduced on page 81 the energy shaping and damping injecting feedback, with an appropriately chosen potential, led to a simple PD controller. For nonlinear systems the resulting controller will in general be nonlinear, but some of the intrinsic features and limitations of a linear PD controller are still present. One major limitation is the absence of integral action, and small disturbances may lead to situations where the asymptotic behavior is severely degraded. This chapter presents a solution to this problem by introducing an additional integral state on the AV. The new state does not integrate the position error directly, as a normal integral state would do. Instead it integrates the potential energy, which means that it can be included directly in the Hamiltonian framework. The energy shaping and damping injecting con-

troller is by definition a position controller, but there may be practical considerations that puts restrictions on the allowed velocity of the AV as well. To accommodate for a certain degree of velocity control an extension is introduced that uses a velocity dependent damping to control the velocity of the AV.

7.1 Energy Shaping and Damping Injection

In this section a energy shaping and damping injecting feedback for the AV is introduced. For convenience, the reduced Hamiltonian model of the AV from chapter 5 is rewritten below

$$\left.\begin{aligned} \begin{bmatrix} \dot{\chi} \\ \dot{\tilde{p}}_1 \end{bmatrix} &= J(\chi,\beta') \begin{bmatrix} \frac{\partial \tilde{H}}{\partial \chi} \\ \frac{\partial \tilde{H}}{\partial \tilde{p}_1} \end{bmatrix} + \begin{bmatrix} 0 \\ B_\phi(\beta') \end{bmatrix} \tau_\phi, \\ \dot{\beta}' &= \zeta, \\ y &= B_\phi^T(\beta') \frac{\partial \tilde{H}}{\partial \tilde{p}_1} \quad (= \dot{\phi}), \\ \tilde{H}(\tilde{p}_1) &= \frac{1}{2}\tilde{p}_1^2, \end{aligned}\right\} \tag{7.1}$$

with

$$J(\chi,\beta') = \begin{bmatrix} 0 & R^T(\theta)\Sigma(\beta')\Upsilon(\beta') \\ -\Upsilon(\beta')\Sigma^T(\beta')R(\theta) & 0 \end{bmatrix},$$
$$B_\phi(\beta') = \Upsilon(\beta')\Sigma^T(\beta')C_2^T(\beta')\frac{1}{r_w}.$$

Since the AV is assumed to be driving on a horizontal field, it does not have any initial potential energy. Suppose we want the AV to converge asymptotically to a set Q_0 defined as the set of minimum points of a potential function

$$Q_0 = \{(\chi,0) \in \Omega_c^* \mid \frac{\partial \bar{U}}{\partial \chi} = 0\}.$$

For simplicity it is assumed that Q_0 is a closed connected set, and that $\bar{U}$ has no other extreme points than those in Q_0.

First of all, we wish to add the artificial potential energy $\bar{U}$ to the system by means of an energy shaping feedback, because we know that the system will then converge to a set that includes Q_0. We are therefore looking for a feedback u_{es} that satisfies (6.4), or in the AV case, a feedback $\tau_{\phi,es}$ that satisfies

$$-\Upsilon(\beta')\Sigma^T(\beta')R(\theta)\frac{\partial \bar{U}}{\partial \chi} = \Upsilon(\beta')\Sigma^T(\beta')C_2^T(\beta')\frac{1}{r_w}\tau_{\phi,es}. \tag{7.2}$$

The relation constitutes one equation with four unknown, and some additional constraints between the four wheel torques have to be introduced in order to solve the equation. Because of the non-slipping and free rolling constraints, the AV can, in theory, be driven by one propulsion motor only. A solution to (7.2) is therefore to set three of the wheel torques equal to zero and solve for the last one. This obviously puts an unnecessary high strain on this single motor. Another solution is to let all four propulsion torques be equal. Both solutions share the property that the torque vector can be written as

$$\tau_{\phi,es} = X\tau_s, \tag{7.3}$$

where τ_s is a scalar, and X is a 4-dimensional *torque distribution* vector. $X = [1\ 0\ 0\ 0]^T$ in the case of driving the AV with only the torque to the first wheel, and $X = [1\ 1\ 1\ 1]^T$ if all four torques are equal. Combining the constraint (7.3) with (7.2) results in an equation with just one unknown, and the solution is

$$\tau_{\phi,es} = -X\frac{r_w\Sigma^T(\beta')R(\theta)}{\Sigma^T(\beta')C_2^T(\beta')X}\frac{\partial\bar{U}}{\partial\chi}. \tag{7.4}$$

The vector X must be chosen such that the denominator is nonzero. The torque distribution vector does not necessarily have to be constant, and in section 7.2 a varying $X(\beta')$, which minimizes the instantaneous electrical power supplied to the propulsion motors, is found. Note that the choice of X has no influence on the motion of the system, and for now it is just assumed that an X that renders the denominator of (7.4) nonzero exists.

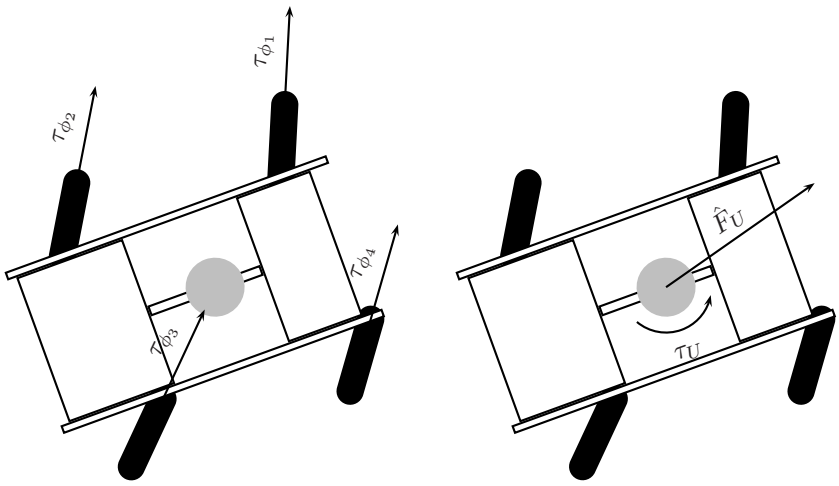
Remark. The derivation of this energy shaping feedback could have been derived by physical consideration alone without the help of (7.2). Consider two almost identical AVs; one (system 1) is actuated at the wheels, as is the case with the real AV, and one (system 2) is actuated by applying a force $\hat{F}_U$ and a torque τ_U at the geometric center. See figure 7.1.

The configuration of the two systems are defined on the same manifold, and they are both subject to the same nonholonomic constraints; the only thing separating them is the point of entry of the inputs. The input space of system 2 is directly related to the generalized momentum p – the three dimensional vector describing the translational and rotational momentum of the AV.

What we aim to do, when shaping the potential energy of the AV, is to find a feedback that applies the negative gradient, viewed as a generalized force, of the desired potential energy function. The negative gradient cannot directly be applied to system 1, but it can be applied directly to system 2. By setting

$$F_U = -\frac{\partial\bar{U}}{\partial\chi}, \qquad F_U = \begin{bmatrix}\hat{F}_U \\ \tau_U\end{bmatrix}$$

the potential energy of system 2 has been shaped. To shape the energy of system 1 we need to find an input τ_ϕ that will make system 1 move along the exact same trajectory as



(a) System 1 (b) System 2

Figure 7.1: *Two methods of actuating of the AV*

system 2. Without going into too much detail the dynamics of system 2 can be written in the reduced Hamiltonian form[1]

$$\begin{bmatrix} \dot{\chi} \\ \dot{\tilde{p}}_1 \end{bmatrix} = J(\chi, \beta') \begin{bmatrix} \frac{\partial \tilde{H}}{\partial \chi} \\ \frac{\partial \tilde{H}}{\partial \tilde{p}_1} \end{bmatrix} - \begin{bmatrix} 0 \\ \Upsilon(\beta')\Sigma^T(\beta')R(\theta) \end{bmatrix} \frac{\partial \bar{U}}{\partial \chi}.$$

Equating this system with (7.1) and using (7.3) yields relation (7.4). △

From (6.6) on page 87 we know that the conservative AV with shaped energy has equilibrium points defined by

$$Q = \{(\chi, 0) \in \Omega_c^* \mid \Sigma^T(\beta')R(\theta)\frac{\partial \bar{U}}{\partial \chi} = 0\}. \tag{7.5}$$

Shaping the potential energy has not changed the conservative property of the AV, and damping has to be introduced to guarantee that Q is also an asymptotically stable set. Using (6.7) the damping injection takes on the simple form

$$\tau_{\phi,di} = -k_d y = -k_d \dot{\phi}, \qquad k_d > 0,$$

[1] It is a matter of exchanging the input vector field $R^T(\theta)C_2^T(\beta')\frac{1}{r_w}\tau_\phi$ of the unreduced system 1 with that of system 2. The input vector field of system 2 is just the identity matrix. System 2 is then reduced with the coordinate transformation defined in chapter 5.

with $k_d \in \mathbb{R}$. The damping injection is essentially just an addition of kinetic friction in the wheels with the same scalar friction coefficient k_d for all four wheels. They do not necessarily have to be equal, but they are chosen this way for simplicity.

Including both energy shaping and damping injection, the total feedback is $\tau_\phi = \tau_{\phi,es} + \tau_{\phi,di}$ and the closed loop becomes

$$\begin{bmatrix} \dot{\chi} \\ \dot{\tilde{p}}_1 \end{bmatrix} = [J(\chi, \beta') - D(\beta')] \begin{bmatrix} \frac{\partial \bar{H}}{\partial \chi} \\ \frac{\partial \bar{H}}{\partial \tilde{p}_1} \end{bmatrix}, \tag{7.6}$$

with a damping matrix

$$D(\beta') = \begin{bmatrix} 0 & 0 \\ 0 & k_d B_\phi(\beta') B_\phi^T(\beta') \end{bmatrix},$$

and a shaped Hamiltonian function

$$\bar{H}(\tilde{p}_1, \chi) = \tilde{H}(\tilde{p}_1) + \bar{U}(\chi).$$

7.2 Torque Distribution

The torque distribution vector X was introduced as a prerequisite for solving (7.2). Although the torque distribution vector is not unique, it has to satisfy the additional constraint of rendering the denominator of (7.4) nonzero. The denominator will become zero (or very small) if the four motors counteracts each other. Consider the situation when the two front wheels are pointing straight ahead ($\beta_1 = \beta_4 = 0$), and the two rear wheels are pointing in the opposite direction ($\beta_2 = \beta_3 = \pi$). If we choose an equal torque distribution between the four wheels $X = [1\ 1\ 1\ 1]^T$ the front and rear torques will cancel each other, the denominator of (7.4) becomes zero, and the torques grow to infinity. To generate these torques the propulsion motors would have to draw an infinitely high current from the power supply, but physical limitations will most likely not allow this. In this situation a torque distribution on the form $X = [1\ 1\ -1\ -1]^T$ would be more appropriate. This example implies that a constant torque distribution vector is a poor choice in some situations, and it is more appropriate to use a varying X; one that always renders the denominator of (7.4) nonzero. An obvious choice is

$$X(\beta') = C_2(\beta')\Sigma(\beta'). \tag{7.7}$$

As long as X is nonzero the denominator of (7.4) is also nonzero[2].

[2] $X = 0$ at the singularity $\beta_i = \pi/2 + n\pi,\ i = 1, 2,\ n \in \mathbb{Z}$ The singularity can easily be avoided. Please refer to the remark on page 43.

The previous example also implies that the torque distribution has an effect on the current drawn from the power supply and hence the energy used by the propulsion motors. Apart from rendering the denominator of (7.4) nonzero, the particular choice of torque distribution in has yet another useful property related to the electrical power. Consider the total instantaneous electrical power supplied to the four propulsion motors

$$P_e = \sum_{i=1}^{4} V_{m_i} I_i,$$

where V_{m_i} is the input voltage applied to the i'th motor, and I_i is the resulting current through the motor. The motors are modeled as first order DC motors, and the input voltages needed to generate the desired torque at a particular angular velocity of each wheel is

$$V_{m_i} = \frac{R_a}{K_t}\left(\tau_{\phi_i} + b'\dot{\phi}_i\right) + K_e\dot{\phi}_i, \qquad i = 1, 2, 3, 4.$$

(See figure 2.17 on page 37 and the subsequent subsections for a description of the parameters.) The currents are proportional to the effective propulsion torques through the motor torque constant K_t

$$I_i = \frac{1}{K_t}\left(\tau_{\phi_i} + b'\dot{\phi}_i\right), \qquad i = 1, 2, 3, 4.$$

The power can thus be rewritten as a function of the torques and the wheel velocities

$$P_e = \sum_{i=1}^{4} \frac{R_a}{K_t^2}\left(\tau_{\phi_i} + b'\dot{\phi}_i\right)^2 + b'\dot{\phi}_i^2 + \dot{\phi}_i\tau_{\phi_i},$$

or in matrix form

$$P_e = \frac{R_a}{K_t^2}\left(\tau_\phi + b'\dot{\phi}\right)^T\left(\tau_\phi + b'\dot{\phi}\right) + b'\dot{\phi}^T\dot{\phi} + \dot{\phi}^T\tau_\phi. \tag{7.8}$$

Note that in consistent units K_t and K_e are equal, i.e., $\frac{K_e}{K_t} = 1$. (7.8) shows that the total input power to the propulsion motors can be divided into three distinct parts. The first part is the power loss in the armature resistance of the motors, the second part is the power lost to friction, and the last part is the remaining power transformed into mechanical power. Suppose that we have designed an energy shaping and damping injecting feedback with some arbitrary torque distribution vector X

$$\tau_\phi = X\frac{r_w\Sigma^T RF}{\Sigma^T C_2^T X} - k_d\dot{\phi}, \qquad F = -\frac{\partial\bar{U}}{\partial\chi}.$$

The dependency on β' and χ have been dropped for notational convenience. Inserting the feedback into (7.8) yields

$$P_e(X) = \frac{R_a}{K_t^2}\left(X\frac{r_w\Sigma^T RF}{\Sigma^T C_2^T X} + (b' - k_d)\dot{\phi}\right)^T \left(X\frac{r_w\Sigma^T RF}{\Sigma^T C_2^T X} + (b' - k_d)\dot{\phi}\right) + b'\dot{\phi}^T\dot{\phi} + \dot{\phi}^T\left(X\frac{r_w\Sigma^T RF}{\Sigma^T C_2^T X} - k_d\dot{\phi}\right).$$

Suppose that we wish to find a vector X that minimizes the power P_e. To find possible candidates we first find the extreme points of the function by solving for X in $\frac{\partial P_e}{\partial X} = 0$. P_e is a rather lengthy term, but many of the terms vanish in the derivative. The square elements in $\dot{\phi}$, for instance, are independent of X and vanish. Next, consider the elements involving

$$\frac{1}{\Sigma^T C_2^T X}\dot{\phi}^T X. \tag{7.9}$$

The individual elements of $\dot{\phi}$ are not independent due to the nonholonomic constraints, but they are functions of the single independent variable $\tilde{p}_1$ (using (3.7) on page 45 and (7.1))

$$\dot{\phi} = C_2\Sigma\frac{\Upsilon}{r_w}\tilde{p}_1.$$

Inserting this into (7.9) yields

$$\frac{1}{\Sigma^T C_2^T X}\tilde{p}_1\frac{\Upsilon}{r_w}\Sigma^T C_2^T X = \tilde{p}_1\frac{\Upsilon}{r_w}.$$

All the elements involving (7.9) hence vanish in $\frac{\partial P_e}{\partial X}$, and finding the extreme points of P_e reduces to finding the extreme points of the function

$$\tilde{P}_e(X) = \frac{1}{\left(\Sigma^T C_2^T X\right)^2}X^T X. \tag{7.10}$$

$\tilde{P}_e$ is related to the power loss in the armature resistance, and we conclude that this is the only loss that can be minimized by a suitable X. Taking the partial derivative with respect to X and setting it equal to zero yields

$$\frac{\partial\tilde{P}_e}{\partial X} = -C_2\Sigma\frac{2}{\left(\Sigma^T C_2^T X\right)^3}X^T X + X\frac{2}{\left(\Sigma^T C_2^T X\right)^2} = 0.$$

Multiplying with the scalar $\frac{1}{2}\left(\Sigma^T C_2^T X\right)^2$ yields

$$-C_2\Sigma\frac{1}{\Sigma^T C_2^T X}X^T X + X = 0,$$

which implies that a vector $X = cC_2\Sigma$, $c \in \mathbb{R}_{\setminus\{0\}}$ is an extreme point of P_e. In fact, any nonzero vector X in the set img$[C_2\Sigma]$ will result in a minimum value of P_e. This can be seen directly by inspection of (7.10). Start by choosing a random $X \in$ img$[C_2\Sigma]$, for example $X = C_2\Sigma$. Any other vector in img$[C_2\Sigma]$ can be reached by a subsequent scaling of X, but this will not change the value of P_e, since the scaling factor will be canceled by the division. Now consider a situation where X is rotated, so that it moves outside img$[C_2\Sigma]$, but still retains its length. The value of the factor X^TX will remain the same, whereas the value of the denominator $(\Sigma^T C_2^T X)^2$ will decrease, leading to an increase of $\tilde{P}_e$ (and hence also P_e). Any four dimensional vector can be reached by first moving along $C_2\Sigma$ followed by a rotation, and we conclude that a minimum value of $P_e(X)$ implies that $X \in$ img$[C_2\Sigma]$. So by using the torque distribution vector (7.7) the electrical power drawn from the power supply has been minimized.

7.3 Convergence Towards Q_0

In section 7.1 a feedback, which asymptotically stabilizes the AV toward the set Q, was introduced. This set is generally larger than the desired set Q_0 due to the nonholonomic nature of the AV. To guarantee asymptotic stability toward Q_0 we need to find a suitable reference for the kinematic inputs so that (6.11) is satisfied. In the AV case it amounts to solving for β' in

$$\frac{\partial \bar{U}}{\partial \chi} \in \text{img}[R^T(\theta)\Sigma(\beta')] = \ker[C_1(\beta')R(\theta)], \qquad \chi \in \mathcal{M}. \tag{7.11}$$

This is solvable for any χ if the following statement is true (6.10)

$$\bigcup_{\beta' \in \mathbb{S}^2} \text{img}[R^T(\theta)\Sigma(\beta')] = \mathbb{R}^3, \quad \chi \in \mathcal{M}. \tag{7.12}$$

The validity of the statement can be rephrased as follows: is it possible to orient the vector $R^T\Sigma$ in any direction in $\mathbb{R}^3$ by turning the wheels? We already know that the velocity of the AV satisfies $\dot{\chi} \in$ img$[R^T\Sigma]$, and since the ICR of a 4WS vehicle can be placed anywhere, any direction on $\mathbb{R}^3 \ni \dot{\chi}$ can be reached. (7.12) is hence true, and there always exists at least one solution to (7.11).

The right side of (7.11) implies that

$$C_1(\beta')R(\theta)\frac{\partial \bar{U}}{\partial \chi} = 0,$$

and written in details for the i'th wheel

$$-\sin(\beta_i + \theta)\frac{\partial \bar{U}}{\partial x_1} + \cos(\beta_i + \theta)\frac{\partial \bar{U}}{\partial x_2} + \kappa_i \cos(\beta_i - \gamma_i)\frac{\partial \bar{U}}{\partial \theta} = 0.$$

Using the cosine addition formula on the third term yields

$$-\sin(\beta_i+\theta)\frac{\partial \bar{U}}{\partial x_1}+\cos(\beta_i+\theta)\frac{\partial \bar{U}}{\partial x_2} + \kappa_i\left[\cos(\beta_i+\theta)\cos(\theta+\gamma_i)+\sin(\beta_i+\theta)\sin(\theta+\gamma_i)\right]\frac{\partial \bar{U}}{\partial \theta}=0.$$

Collecting terms of sines and cosines yields

$$\sin(\beta_i+\theta)\left[\frac{\partial \bar{U}}{\partial x_1}-\kappa_i\sin(\theta+\gamma_i)\frac{\partial \bar{U}}{\partial \theta}\right]=\cos(\beta_i+\theta)\left[\frac{\partial \bar{U}}{\partial x_2}+\kappa_i\cos(\theta+\gamma_i)\frac{\partial \bar{U}}{\partial \theta}\right],$$

and finally we have that

$$\beta_i=\arctan\left(\frac{\frac{\partial \bar{U}}{\partial x_2}+\kappa_i\cos(\theta+\gamma_i)\frac{\partial \bar{U}}{\partial \theta}}{\frac{\partial \bar{U}}{\partial x_1}-\kappa_i\sin(\theta+\gamma_i)\frac{\partial \bar{U}}{\partial \theta}}\right)-\theta.$$

This is equivalent to

$$\beta_i=\angle\left(\begin{bmatrix}\frac{\partial \bar{U}}{\partial x_1}\\ \frac{\partial \bar{U}}{\partial x_2}\end{bmatrix}+\kappa_i\frac{\partial \bar{U}}{\partial \theta}e_i\right)-\theta, \tag{7.13}$$

where e_i is a unit vector perpendicular to the line connecting the geometric center of the AV and the center of the i'th wheel, see figure 7.2. To achieve asymptotic convergence to Q_0 the wheels should hence point in the directions of linear combinations of a translational force vector and four force vectors uses to rotate the AV.

Whether or not it is possible to design steering controllers that guarantees that (7.13) is always satisfied depends largely on the shape of the potential energy function, physical saturation limits in the steering motors, and the velocity of the AV. Steep curvature changes in the energy function $\bar{U}$ will result in fast changes of the desired steering angles, but since the turning rate of steering motors are physically limited, the desired steering angles may not be met. This is not a real problem though. If the references are not met the AV will converge to the set Q, and on approach the AV will slow down due to the damping. A slower moving AV leads to a decreasing rate of change of the gradient direction and hence also the steering angle references. Even slow controllers will eventually be able to meet the references.

Figure 7.3 shows a closed loop simulation with the energy shaping and damping injecting controller. The simulation is an example of asymptotic stabilization of the AV at a target point χ_0 using a simple quadratic potential shaping function

$$\bar{U}(\chi)=\frac{1}{2}(\chi-\chi_0)^T K_p(\chi-\chi_0),\qquad \chi_0=\begin{bmatrix}10\\-10\\\pi\end{bmatrix},\qquad K_p=\begin{bmatrix}160&0&0\\0&80&0\\0&0&80\end{bmatrix},$$

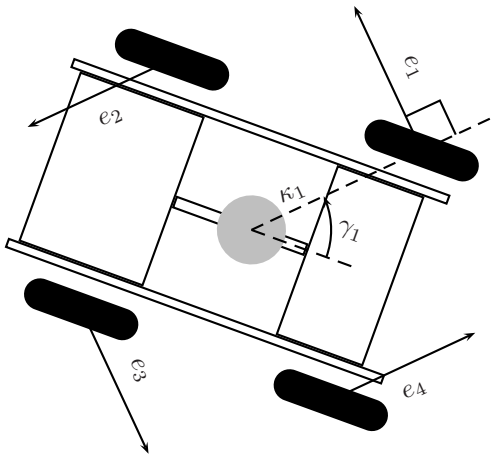


Figure 7.2: *Unit vectors of rotating forces*

and with damping constant $k_d = 8$. The dashed lines in the figure represent contour curves of the x_1, x_2-components of $\bar{U}$. This part of the function is not completely symmetric; it has a steeper descent along the x_1-axis than along the x_2-axis, which explains why the AV converges faster in the x_1-direction than in the x_2-direction. The AVs drawn in the figure shows the position and orientation of the AV for every 2 seconds.

7.4 Disturbances and Integral Action

The energy shaping and damping injecting controller inherits some of the characteristics of a linear PD controller. In the simple linear example on page 81 the resulting controller was in fact a linear PD controller when the potential energy function was a quadratic function, and without integral action the asymptotic convergence of the linear system is often sacrificed if there are external disturbances. The same is true for the AV where external disturbances comprise unmodeled slopes in the field, uneven soil tracks, rocks, etc. Consider the same simulation as shown in figure 7.3, but this time the AV is driving on a field with a slope. A 5° slope is modeled by a constant translational force pulling the AV toward north-west, and the resulting simulation in shown in figure 7.4. The gray AV represents the target configuration, but the AV stops short of it by a few meters because of the slope.

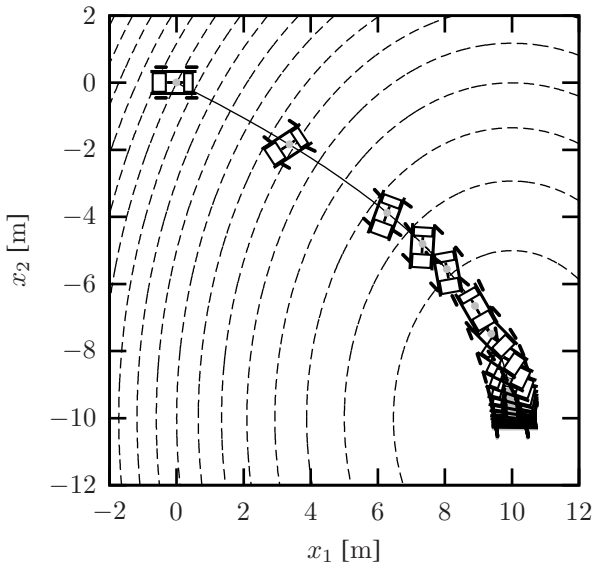


Figure 7.3: *Stabilization at a single point*

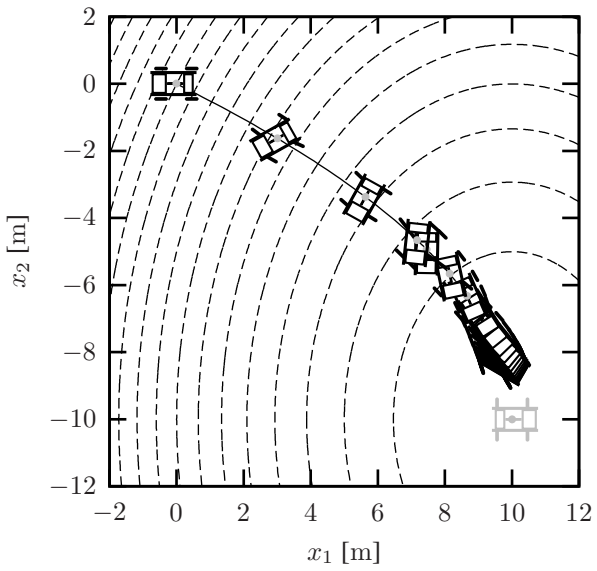


Figure 7.4: *Stabilization at a single point on a non-horizontal field.*

The remainder of this section describes a method of introducing integral action into the Hamiltonian formulation of the AV. The resulting closed loop with integral action maintains the Hamiltonian structure and hence also the passivity property of the system. The integral feedback that maintains the Hamiltonian structure has itself a special structure, and it has an interesting consequence for the stability of the system; because of the preserved passivity, increased feedback gains cannot destabilize the system. To relate this to linear systems, the simple example of a mass moving on a horizontal plane is revisited at the end of the section.

Integral action is only considered when the AV has to converge to a single point, i.e., when the potential energy function $\bar{U}$ has a single global minimum. In this situation any unmodeled structures in the field, such as an unknown slope, rocks, or soil tracks, may inhibit the asymptotic convergence. These unknown physical structures can all be modeled as additional unknown potential energy. So instead of the Hamiltonian function used to prove asymptotic stability in the ideal case, the disturbed Hamiltonian function of the real system includes additional potential energy

$$\bar{H}_d(\tilde{p}_1, \chi) = \underbrace{\frac{1}{2}\tilde{p}_1^2 + \bar{U}(\chi)}_{\bar{H}} + U_d(\chi). \tag{7.14}$$

U_d is the disturbance potential energy function capturing the unknown structures in the field. The addition of the disturbance function means that the system will not converge to the minimum of the shaped potential energy $\bar{U}$, but will instead converge to the minimum of the total potential energy $\bar{U} + U_d$. The introduction of an unknown potential can either be introduced in the Hamiltonian function as in (7.14), or it can be introduced in the closed loop system as an additional energy shaping term

$$\begin{bmatrix} \dot{\chi} \\ \dot{\tilde{p}}_1 \end{bmatrix} = [J(\chi, \beta') - D(\beta')] \begin{bmatrix} \frac{\partial \bar{H}}{\partial \chi} \\ \frac{\partial \bar{H}}{\partial \tilde{p}_1} \end{bmatrix} - \begin{bmatrix} 0 \\ \Upsilon(\beta')\Sigma^T(\beta)R(\theta)\frac{\partial U_d}{\partial \chi} \end{bmatrix}. \tag{7.15}$$

To introduce integral action we are seeking an additional feedback that will make the augmented system converge to the original desired set Q_0. When the AV is far away from Q_0 integral action is not really necessary because $\frac{\partial \bar{U}}{\partial \chi}$ is generally much larger than $\frac{\partial U_d}{\partial \chi}$, but when the AV approaches Q_0 the disturbance begins to dominate, and integral action must be used to drive the last distance to Q_0. To simplify the problem some assumptions have to be made. It is assumed that the disturbance is constant locally around Q_0 in the sense that $\frac{\partial U_d}{\partial \chi}$ is constant. This rules out some disturbances, such as stones and other small structures, while larger structures, such as unmodeled slopes, are still allowed. It is further assumed that the AV is sufficiently damped so when approaching Q_0 it is driving slowly, and if a smooth well behaved $\bar{U}$ is used, the direction of $\frac{\partial \bar{U}}{\partial \chi}$ is also changing very slowly. Since the steering angles are derived from this direction, it is assumed that β' is fixed during periods when integral actions is turned on.

The first issue in the design of integral action is to decide on an appropriate quantity to integrate. Suppose that the closed loop system (7.15) is experiencing a nonzero disturbance. After a while the system will stop ($\tilde{p}_1 = 0$), where the artificial force $-\frac{\partial \bar{U}}{\partial \chi}$, which should pull the AV toward Q_0, is canceled by the disturbance.

$$\dot{\tilde{p}}_1 = -\Upsilon(\beta')\Sigma^T(\beta')R(\theta_0)\frac{\partial \bar{U}}{\partial \chi} - \Upsilon(\beta')\Sigma^T(\beta')R(\theta_0)\frac{\partial U_d}{\partial \chi} = 0.$$

The artificial force is a function of the generalized coordinates χ and is related to the error between the current configuration of the AV and the desired target configuration Q_0; if $\bar{U}$ is a quadratic function of χ the gradient is proportional to the error. One choice of integral state is therefore to define a state that is related to the integral of the artificial force and then feed back this state. Define the integral state

$$\dot{p}_I = -\Upsilon(\beta')\Sigma^T(\beta)R(\theta)\frac{\partial \bar{U}}{\partial \chi}.$$

Inserting this into (7.1) yields yet another Hamiltonian system

$$\begin{bmatrix} \dot{\chi} \\ \dot{\tilde{p}}_1 \\ \dot{p}_I \end{bmatrix} = [J_I(\chi,\beta') - D_I(\beta')] \begin{bmatrix} \frac{\partial \bar{H}}{\partial \chi} \\ \frac{\partial \bar{H}}{\partial \tilde{p}_1} \\ \frac{\partial \bar{H}}{\partial p_I} \end{bmatrix} + \begin{bmatrix} 0 \\ B_\phi(\beta') \\ 0 \end{bmatrix} \tau_{\phi,I} - \begin{bmatrix} 0 \\ \Upsilon(\beta')\Sigma^T(\beta)R(\theta)\frac{\partial U_d}{\partial \chi} \\ 0 \end{bmatrix}, \tag{7.16}$$

where $\tau_{\phi,I}$ is an additional input to be used for the integral state feedback (now $\tau_\phi = \tau_{\phi,es} + \tau_{\phi,di} + \tau_{\phi,I}$) and

$$J_I(\chi,\beta') = \begin{bmatrix} 0 & R^T(\theta)\Sigma(\beta')\Upsilon(\beta') & R^T(\theta)\Sigma(\beta')\Upsilon(\beta') \\ -\Upsilon(\beta')\Sigma^T(\beta')R(\theta) & 0 & 0 \\ -\Upsilon(\beta')\Sigma^T(\beta')R(\theta) & 0 & 0 \end{bmatrix},$$

$$D_I(\beta') = \begin{bmatrix} 0 & 0 & 0 \\ 0 & k_d B_\phi(\beta')B_\phi^T(\beta') & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

The upper right element of J_I has been set to the current value to maintain the skew-symmetric property of the matrix. It does not change the dynamics of the system because $\frac{\partial \bar{H}}{\partial p_I} = 0$, but the skew-symmetric property will be useful in the final closed loop system. To close the loop we need to feed back the integral state, while maintaining the stability of the closed loop system. It is a well known fact from linear systems analysis that an integral feedback may destabilize the system if the integral feedback gain is too high. Care should therefore be taken when choosing the feedback, but if it is possible to find a feedback that maintains the Hamiltonian structure and passivity of the system we can also prove that it is stable. Consider the feedback

$$\tau_{\phi,I} = -k_d B_\phi^T(\beta')k_I(\tilde{p}_1 - p_I), \qquad k_I > 0, \tag{7.17}$$

and the new Hamiltonian function

$$\bar{H}_I = \bar{H} + \frac{1}{2}k_I(\tilde{p}_1 - p_I)^2 = \frac{1}{2}\tilde{p}_1^2 + \frac{1}{2}k_I(\tilde{p}_1 - p_I)^2 + \bar{U}(\chi).$$

With this feedback the closed loop system becomes

$$\begin{bmatrix} \dot{\chi} \\ \dot{\tilde{p}}_1 \\ \dot{p}_I \end{bmatrix} = [J_I(\chi, \beta') - D_I(\beta')] \begin{bmatrix} \frac{\partial \bar{H}_I}{\partial \chi} \\ \frac{\partial \bar{H}_I}{\partial \tilde{p}_1} \\ \frac{\partial \bar{H}_I}{\partial p_I} \end{bmatrix} - \begin{bmatrix} 0 \\ \Upsilon(\beta')\Sigma^T(\beta)R(\theta)\frac{\partial U_d}{\partial \chi} \\ 0 \end{bmatrix}.$$

This system is again a Hamiltonian system with skew-symmetric interconnection matrix J_I, a positive semi-definite dissipation matrix D_I, and a new strictly positive Hamiltonian function $\bar{H}_I$. The system is still stable, which can be seen by taking $\bar{H}_I$ as a Lyapunov function candidate, and looking at its time derivative

$$\dot{\bar{H}}_I = -\begin{bmatrix} \frac{\partial \bar{H}_I}{\partial \chi} \\ \frac{\partial \bar{H}_I}{\partial \tilde{p}_1} \\ \frac{\partial \bar{H}_I}{\partial p_I} \end{bmatrix}^T D_I(\beta') \begin{bmatrix} \frac{\partial \bar{H}_I}{\partial \chi} \\ \frac{\partial \bar{H}_I}{\partial \tilde{p}_1} \\ \frac{\partial \bar{H}_I}{\partial p_I} \end{bmatrix} = -k_d B_\phi(\beta') B_\phi^T(\beta') \left(\frac{\partial \bar{H}_I}{\partial \tilde{p}_1}\right)^2 \leq 0.$$

The system is stable, but does it still converge asymptotically to the same set as before? Application of theorem 6.1 on page 83 proves that the system converges asymptotically to the largest invariant set contained in the set of points where $\dot{\bar{H}}_I = 0 \Rightarrow \frac{\partial \bar{H}_I}{\partial \tilde{p}_1} = 0$, or

$$\tilde{p}_1 = \frac{k_I}{1 + k_I} p_I.$$

Since $k_I > 0$ the relation implies that $\dot{\tilde{p}}_1 = \dot{p}_I = 0$ in the set where $\dot{\bar{H}}_I = 0$, and the system converges to the following set on the constrained tangent bundle

$$Q_I = \{(\chi, \tilde{p}_1) \in \Omega^* \mid \Sigma^T(\beta')R(\theta)\frac{\partial \bar{U}}{\partial \chi} = 0\}. \tag{7.18}$$

Note that the set of convergence is not exactly the same as (7.5); in this new set $\tilde{p}_1$ is not necessarily zero. Imagine the case where there is no potential energy in the system, and p_I is given an initial value different from zero. The lack of potential energy means that p_I will never change from its initial value, and $\tilde{p}_1$ will hence also remain constant and nonzero in the set where $\dot{\bar{H}}_I = 0$. If, on the other hand, the potential energy of the system has only a single global minimum, then the set (7.18) only contains the points where $\tilde{p}_1 = 0$. This can best be seen by contradiction. If $\tilde{p}_1 \neq 0$, then $\dot{\chi} \neq 0$. This implies that if $\bar{U}$ has only one minimum, then after a short while $\dot{p}_I \neq 0$ and $\dot{\bar{H}}_I \neq 0$. The points where $\tilde{p}_1 \neq 0$ do therefore not belong to an invariant set where $\dot{\bar{H}}_I = 0$.

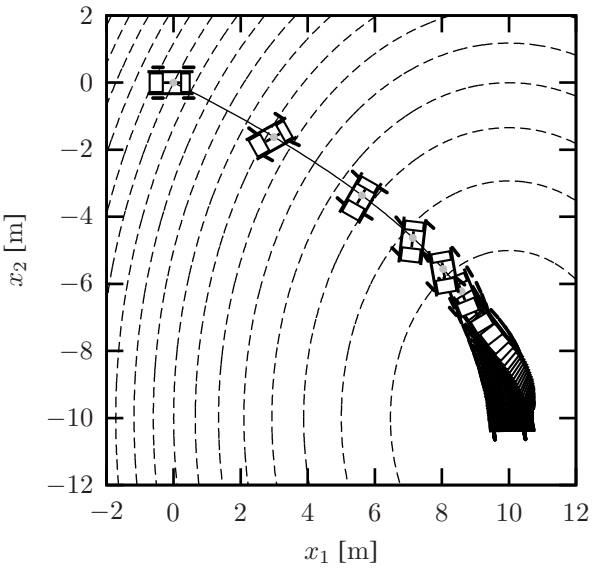


Figure 7.5: *Stabilization at a single point using integral action*

For practical purposes it is assumed that the set (7.18) coincides with (7.5), because the integral action is only to be used for asymptotic stabilization of the AV at a single desired configuration.

Figure 7.5 shows the same simulation as in figure 7.4, but now with integral action. An integral feedback gain of $k_I = 0.01$ has been used, and figure 7.6 shows the evolution of the position and orientation errors of the simulation. To avoid integrator windup, the integral state is not updated, and the integral feedback is not switched on, until the kinetic energy of the AV has reached a lower bound, i.e., $|\tilde{p}_1| \leq 1$. This happens around $t = 34\text{s}$, where integral action is turned on, and the AV is forced to converge asymptotically toward the minimum of $\bar{U}$.

Remark. Feeding back integral states in a control system can often have a destabilizing effect on the closed loop system. It may therefore seem illogical that the closed loop system remains stable, even if k_I is increased to an arbitrary high positive value. The stability is maintained because the integral feedback (7.17) also introduces an additional damping, which is proportional to k_I. To have a closer look at the effect of the integral feedback we return to the simple example from page 81 of a mass m sliding on a

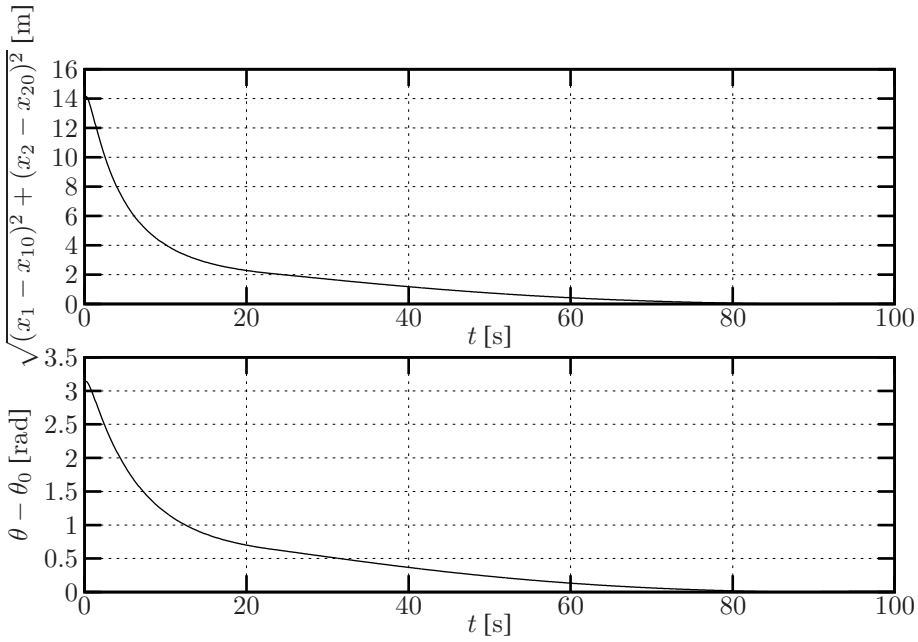


Figure 7.6: *Position and angular errors when stabilizing to a point using integral action*

frictionless surface. The Hamiltonian representation of the system is

$$\begin{bmatrix} \dot{x} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} \frac{\partial H}{\partial x} \\ \frac{\partial H}{\partial p} \end{bmatrix} + \begin{bmatrix} 0 \\ F \end{bmatrix},$$
$$H(p) = \frac{1}{2} m^{-1} p^2.$$

In the example an energy shaping and damping injecting feedback was introduced

$$F = -\frac{\partial U}{\partial x} - k_d \dot{x},$$

and the resulting closed loop system was

$$\begin{bmatrix} \dot{x} \\ \dot{p} \end{bmatrix} = \left(\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ 0 & k_d \end{bmatrix} \right) \begin{bmatrix} \frac{\partial \bar{H}}{\partial x} \\ \frac{\partial \bar{H}}{\partial p} \end{bmatrix},$$
$$\bar{H}(x, p) = \frac{1}{2} m^{-1} p^2 + U(x).$$

We now introduce the integral state

$$p_I = -\frac{\partial U}{\partial x},$$

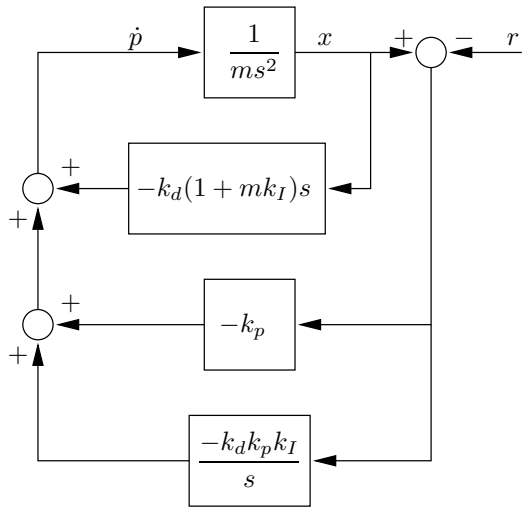


Figure 7.7: *Closed loop of the simple example*

and feed it back so that the total feedback becomes

$$F = -\frac{\partial U}{\partial x} - k_d\dot{x} - k_dk_I(p - p_I).$$

Suppose that we wish to stabilize the moving mass at a reference point r. The artificial potential energy function to achieve this may then be chosen as $U(x) = \frac{1}{2}k_p(x-r)^2$, which has a single global minimum at $x = r$. A block diagram of the closed loop system is depicted in figure 7.7, and the transfer function from reference to position is

$$\frac{X(s)}{R(s)} = \frac{k_ps + k_dk_pk_I}{ms^3 + k_d(1+mk_I)s^2 + k_ps + k_dk_pk_I}.$$

To investigate the stability of the system we look at the Routh array

$$\begin{array}{lll} s^3: & 1 & m^{-1}k_p \\ s^2: & m^{-1}k_d(1+mk_I) & m^{-1}k_dk_pk_I \\ s^1: & m^{-1}k_p\left(1 - \dfrac{k_I}{m^{-1}+k_I}\right) & 0 \\ s^0: & m^{-1}k_dk_pk_I & . \end{array}$$

If the coefficients of the left column are all positive the system is stable. We immediately see that as long as k_p, k_d, and k_I are all positive the system is stable. So, in conclusion,

we have designed a simple PID controller for the linear case, but by giving the controller a Hamiltonian structure the closed loop system is always stable, in theory, no matter the size of the feedback gains. △

7.5 Path Tracking

The focus of this chapter has up until now been on stabilizing the AV at a single target configuration, but the energy shaping and damping injecting feedback can easily be extended to path tracking as well. The first step toward path tracking was already seen in figure 7.3 on page 105, where the potential shaping function was designed to have a steeper slope in the x_1-direction than the x_2-direction, thereby forcing the AV to converge faster in the x_1-direction. The simulation showed how it is possible to define the path, along which the AV travels, by shaping the potential energy function $\bar{U}$. By construction, the closed loop system will always be pulled toward the minimum of $\bar{U}$, and convergence to a desired path can be achieved by designing the function such that the path represents a set of low values. This is illustrated in the following example, where energy shaping is used to track a circular path.

7.5.1 Tracking a Circle

Consider the situation where the desired path is a circle in the x_1, x_2-plane with center at the origin and radius r_0. The initial position of the AV may be anywhere in the plane. The goal of path tracking is then to force the AV to converge to the circle and then track it indefinitely. It is assumed that damping is already present in the system (if not, it can be injected) and that any disturbances can be neglected. Integral action is turned off. We then seek a potential function $\bar{U}$, which is able to attract the AV to the circle, and when on the circle, the function should be able to pull the AV along it. To converge to the path the following shaping function is constructed:

$$\bar{U}_c = \frac{1}{2}K_c(r - r_0)^2, \quad r^2 = x_1^2 + x_2^2, \quad K_c > 0.$$

The function is depicted in figure 7.8.

Starting anywhere in the x_1, x_2-plane (except at the origin, which is a singular point and should be avoided) the negative gradient of this function will always pull the AV toward the circle, which constitutes the set of minima of $\bar{U}_c$. Although the AV is expected to converge to the circle it has not yet been defined how the AV should move when on the circle. If $\bar{U}_c$ is used alone as shaping function the AV will simply drive toward the point on the path, which is closest to the initial position of the AV, and eventually stop at this

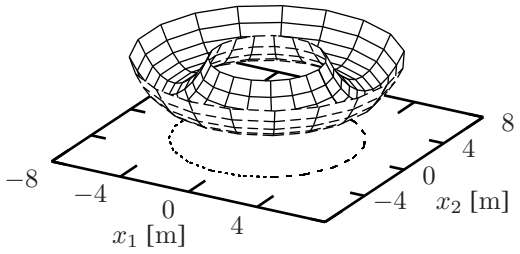


Figure 7.8: *The potential function $\bar{U}_c$, whose minima comprises a circle*

point when all kinetic energy has been dissipated. This function is therefore not useful by itself, and an additional potential has to be introduced to achieve tracking along the path. If the AV is to drive clockwise along the circular path the following potential function with a constant length gradient along the path is constructed:

$$\bar{U}_a = K_a \psi, \quad K_a > 0,$$

where $\psi = \arctan \frac{x_2}{x_1}$ is the angle of the line connecting the origin of the x_1, x_2-plane with the AV. Note that ψ is not limited to $0 < \psi < \pi$, but is allowed to evolve indefinitely, as illustrated on figure 7.9.

The negative gradient of this function will always pull the AV in the clockwise direction parallel to the tangent of the circle. By adding $\bar{U}_c$ and $\bar{U}_a$ a new potential function is constructed that pulls the AV toward and along the path. The combined potential function $\bar{U}_c + \bar{U}_a$ is shown in figure 7.10. Just by looking at this figure one would immediately expect the AV to exhibit some kind of circular movement in the x_1, x_2-plane if $\bar{U}$ is applied as shaping function. Figure 7.11 shows a simulation of the AV with $\bar{U} = \bar{U}_c + \bar{U}_a$, parameters $r_0 = 5$, $K_c = 1600$, $K_a = 1600 r_0$, and a damping constant of $k_d = 8$.

The simulation shows the trajectory and position of the AV $1\mathrm{s}$ apart during a period of $10\mathrm{s}$. The desired path is marked by a dotted line, and the dashed lines represent contour curves of the potential function. Initially, the AV is started with zero velocity at $(x_1, x_2) = (4, 0)$. The figure shows how the AV converges smoothly to the path and stays there for the remaining time. Only one revolution is shown, but the AV will in principle continue due to the lack of a absolute minimum in the potential function. Since the damping is nonzero, the AV reaches a constant velocity when the gradient of the

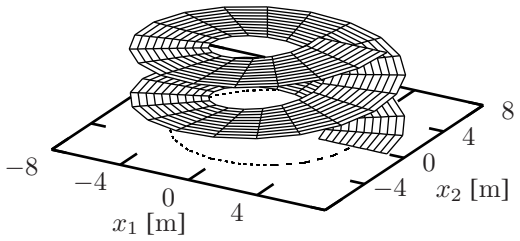


Figure 7.9: *The potential $\bar{U}_a$ pushing the AV along the circle*

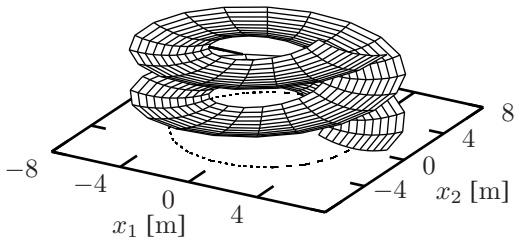


Figure 7.10: *Combined potential function $\bar{U}_c + \bar{U}_a$*

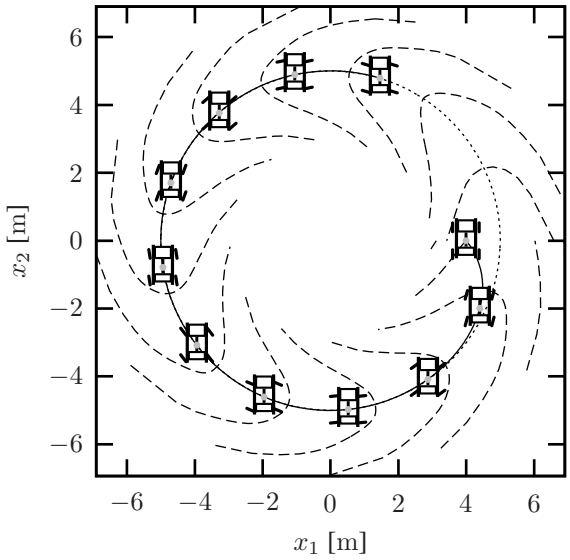


Figure 7.11: *Tracking the circle*

potential function pulling it along the path is canceled by the damping. The orientation of the AV is constant during the simulation because $\bar{U}$ is independent of θ.

If the AV is supposed to stop at a certain point on the path the gradient along the path should be designed so that it has constant length when the AV is far away from the target point, and when the AV approaches the point, the length of the gradient should decrease and eventually vanish at that point.

It may seem illogical that there is no overshoot when the AV hits the path. Indeed, a free moving mass, or a rolling ball would oscillate about the path, but the nonholonomic nature of the AV forces the momentum perpendicular to the path, which is generate when approaching it, to be directed along the path instead. Consider the gradient of the potential function

$$\frac{\partial \bar{U}}{\partial \chi} = K_c(r - r_0) \begin{bmatrix} \cos\psi \\ \sin\psi \\ 0 \end{bmatrix} + \frac{K_a}{r} \begin{bmatrix} -\sin\psi \\ \cos\psi \\ 0 \end{bmatrix}.$$

The gradient consists of two distinct parts; the first part, which is perpendicular to the circle and vanishes on the circle, and the second part, which is tangent to the circle and have constant length on the circle. When the AV approaches and eventually hits the path the gradient of $\bar{U}$ points along the path. Since the direction of the gradient is

used as reference for the steering motors the wheels are pointing in the direction of the tangent, and the AV cannot overshoot the path without violating the constraints. In the practical case though, care should be taken when choosing values for the constants in the potential function. If K_c is very large compared to K_a the direction of the gradient will change very rapidly when the AV approaches the path, and the AV may be forced to do an almost 90° turn when it hits the path. If the velocity is high the result may be sideways slip of the wheels, or even worse, the AV may roll over.

The potential function introduced here has effectively achieved path tracking to a circle in the x_1, x_2-plane, but there are still some issues that have to be resolved before the path tracking algorithm can be used for effective crow row tracking. First of all, the control of the orientation of the AV must be addressed. In the circle example the orientation of the AV was left unchanged, but this is of cause not a good idea if the crop rows should be left undamaged. This issue will be addressed in the next subsection. The second issue is related to the desirable traveling velocity during path tracking. The final traveling velocity along the path is in the ideal case determined by the length of the gradient along the path and the amount of damping in the system. In the non-ideal case the velocity is also influenced by disturbances, such as unknown slopes in the field, unmodeled friction, etc. In subsection 7.5.3 an adaptive damping scheme is introduced to address this issue and to achieve a great deal of velocity control along the path.

7.5.2 Inter Crop Row Potentials

In the example of tracking a circle the AV maintained a constant orientation, and at certain points on the circle the wheels of the AV crossed the path. If the path represents a crop row this is not very desirable. The potential function $\bar{U}_c$ used to converge to the circle was defined to have a minimum when the geometric center of the AV was on the circle. When it comes to crop row tracking it makes more sense to define a function, which has a minimum when the wheels are on the inter crop row space instead. Figure 7.12 shows a schematic drawing of what is defined as the *inter row potentials* (dashed graphs). Instead of the geometric center tracking a single potential on the crop row the left wheels should track the left inter row potential and the right wheels should track the right inter row potential.

To exemplify the inter row potential tracking the circle tracking example from earlier is revisited. The left and right inter row potentials can now be defined as follows. If the AV is to drive clockwise along the circle the left wheels (wheel 1 and 2) should track a circle of radius $r_0 + 0.5$, and the right wheels (wheel 3 and 4) should track a circle of radius $r_0 - 0.5$. Define the position of the i'th wheel relative to the center of the circles as a two dimensional vector w_i. Assuming that the center of the circle and the origin of

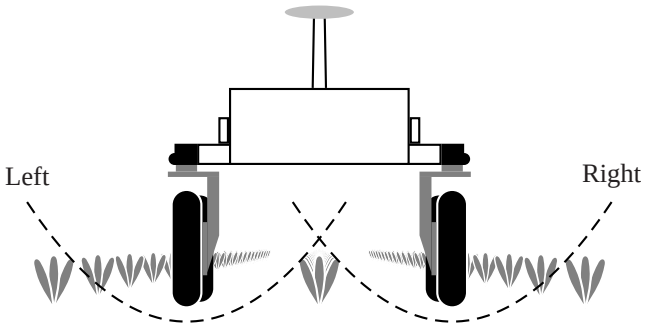


Figure 7.12: *Inter crop row potentials*

the $\mathcal{N}$-frame coincide the vector is given by

$$w_i = \begin{bmatrix} x_1 + \kappa_i \cos(\gamma_i + \theta) \\ x_2 + \kappa_i \sin(\gamma_i + \theta) \end{bmatrix}, \qquad i = 1, \ldots, 4.$$

The error between each wheel and the circle it has to track is then $|w_i| - (r_0 + \delta r_i)$, where $\delta r_i = 0.5$ for $i = 1, 2$ (left wheels), and $\delta r_i = -0.5$ for $i = 3, 4$ (right wheels). Let us then define the total potential function as a sum of squares of these errors

$$\bar{U}_{ic} = \frac{1}{2} K_c \sum_{i=1}^{4} \left(|w_i| - (r_0 + \delta r_i) \right)^2,$$

where K_c is a design parameter. This function clearly has a minimum when all four wheels are situated in the inter crop row space, though the errors cannot vanish completely due to the curvature of the circle and the rigid body frame of the AV. To apply the potential function in the feedback the gradient of the function must first be found

$$\frac{\partial \bar{U}_{ic}}{\partial \chi} = K_c \sum_{i=1}^{4} (|w_i| - (r_0 + \delta r_i)) \frac{\partial |w_i|}{\partial \chi}.$$

With

$$|w_i| = \sqrt{(x_1 + \kappa_i \cos(\gamma_i + \theta))^2 + (x_2 + \kappa_i \sin(\gamma_i + \theta))^2},$$

each individual element of $\frac{\partial |w_i|}{\partial \chi}$ is

$$\frac{\partial |w_i|}{\partial x_1} = \frac{1}{2|w_i|} 2(x_1 + \kappa_i \cos(\gamma_i + \theta)) = \cos(\angle w_i),$$

$$\frac{\partial |w_i|}{\partial x_2} = \frac{1}{2|w_i|} 2(x_2 + \kappa_i \sin(\gamma_i + \theta)) = \sin(\angle w_i),$$

where $\angle w_i$ denotes the angle of the vector w_i. The third element of the gradient is

$$\begin{aligned}\frac{\partial |w_i|}{\partial \theta} &= \frac{1}{2|w_i|}\big[2(x_1 + \kappa_i \cos(\gamma_i + \theta))(-\kappa_i \sin(\gamma_i + \theta)) \\ &\quad + 2(x_2 + \kappa_i \sin(\gamma_i + \theta))\kappa_i \cos(\gamma_i + \theta)\big] \\ &= \kappa_i[-\cos(\angle w_i)\sin(\gamma_i + \theta) + \sin(\angle w_i)\cos(\gamma_i + \theta)] \\ &= -\kappa_i \sin(\gamma_i + \theta - \angle w_i).\end{aligned}$$

The final gradient is then

$$\frac{\partial \bar{U}}{\partial \chi} = \underbrace{K_c \sum_{i=1}^{4} (|w_i| - (r_0 + \delta r_i)) \begin{bmatrix} \cos(\angle w_i) \\ \sin(\angle w_i) \\ -\kappa_i \sin(\gamma_i + \theta - \angle w_i) \end{bmatrix}}_{\frac{\partial \bar{U}_{ic}}{\partial \chi}} + \underbrace{\frac{K_a}{r} \begin{bmatrix} -\sin\psi \\ \cos\psi \\ 0 \end{bmatrix}}_{\frac{\partial \bar{U}_a}{\partial \chi}}. \tag{7.19}$$

Note that $\bar{U}_a$ has been left unchanged from the previous example, as its only purpose is to pull the AV along the circle. The major difference between this gradient and the one from earlier is the nonzero term $\frac{\partial \bar{U}_c}{\partial \theta}$, which has been introduced by using the inter row potentials. Figure 7.13 shows a simulation using the inter row potentials with $K_c = 300$, $K_a = 600 r_0$, and a damping factor of $k_d = 8$. The AV is drawn 1s apart.

The simulation shows that the AV has started to orient itself, and the wheels no longer crosses the path. The orientation does seem to lag behind though when choosing a relatively low valued K_c. By increasing it with a factor 4 to $K_c = 1200$ the lag is greatly reduced as shown on figure 7.14. Unfortunately, the AV experiences a huge attraction to the path from its initial position, which results in a very sharp turn at high velocity when the AV approaches the path. This is the drawback of only having one design parameter (K_c) to determine the gain of the translational attraction to the path and the subsequent tracking. The problem can be solved by using a low K_c when initially approaching the path and then increasing K_c when on the path.

7.5.3 Adaptive Damping

In all the preceding tracking examples nothing has been stated about the velocity of the AV. The AV will eventually reach a constant velocity when the gradient of the potential energy is canceled by the damping, and in theory, this steady state velocity can be determined if the damping factor is known and there are no disturbances. If there exists specifications on desired traveling velocity along the path the gains in the potential

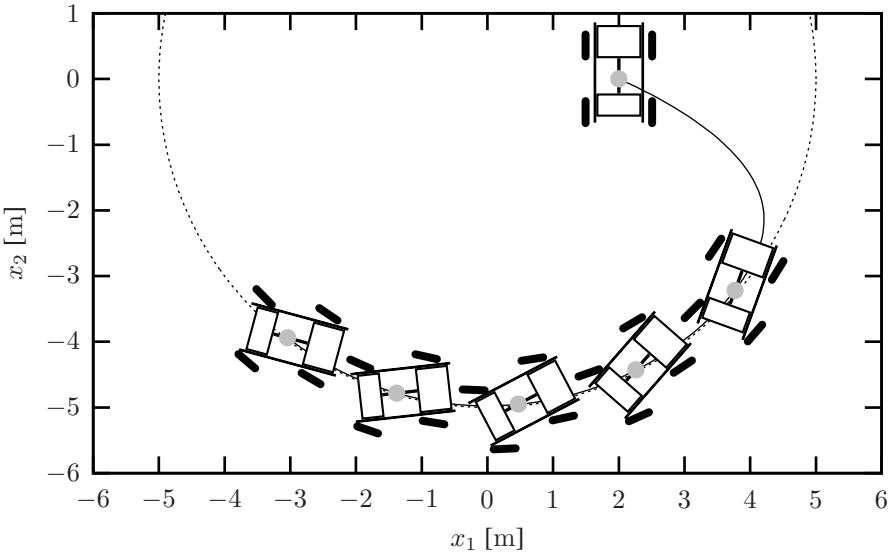


Figure 7.13: *Tracking the circle with inter row potentials ($K_c = 80$)*

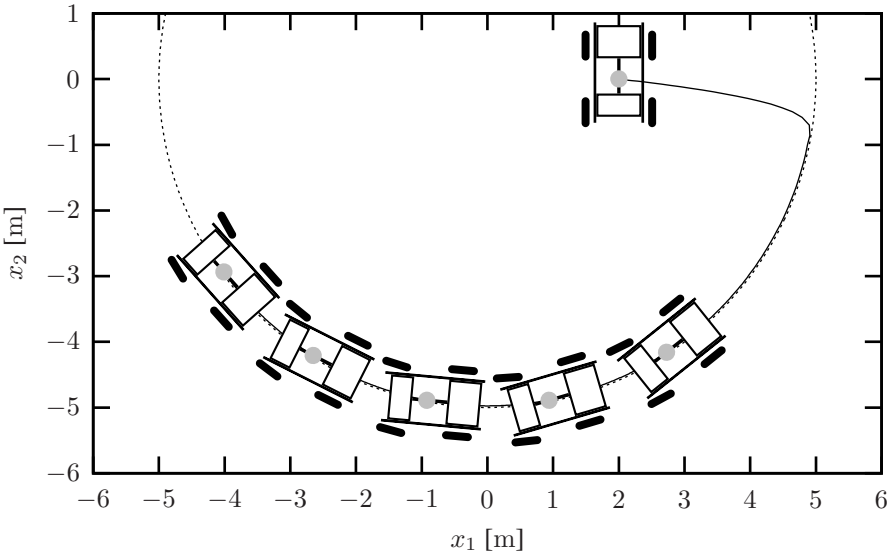


Figure 7.14: *Tracking the circle with inter row potentials ($K_c = 320$)*

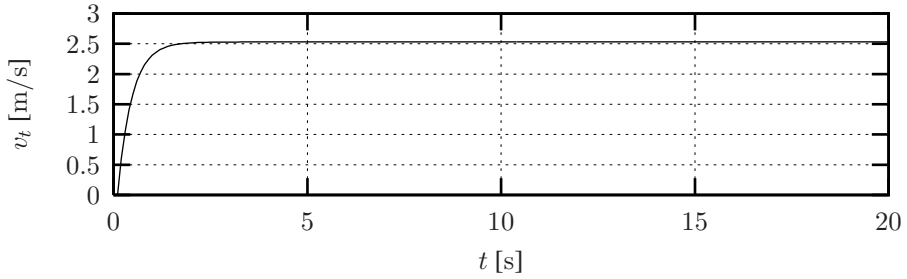


Figure 7.15: *Velocity of the AV during path tracking with constant damping*

energy function can be chosen accordingly. In practice though, the kinetic friction, unknown slopes in the field, parameter variations, etc., will influence the velocity of the AV, and an additional velocity feedback is necessary to achieve the desired velocity. One option is to use a velocity feedback to increase or decrease the length of the gradient along the path. In other words, let the gradient along the path pull the AV more if the velocity is lower than the desired velocity, and pull it less if the velocity is higher than the desired velocity. This idea poses a basic problem though. The gradient should still be considered as a gradient of a potential energy function, and if the length of the gradient varies, the potential energy on the manifold also varies as a function of velocity. This means that the AV can generate its own potential energy, and the passivity property of the AV has be lost. Another way of solving the velocity control problem, while still maintaining the passivity property of the system, is to vary the damping factor. The damping factor can be varied, and the passivity maintained, as long as the damping factor is positive. Consider an adaptive damping on the form

$$\dot{k}_d = \frac{1}{T_d}(v_t - v_{t0}), \qquad v_t = \sqrt{\dot{x}_1^2 + \dot{x}_2^2},$$
$$k_d = \text{sat}(k_d, k_{d,min}).$$

When the velocity of the AV diverges from the desired velocity v_{t0} the damping factor is either increased or decreased at a rate proportional to the velocity error. T_d is the integration time. The second part is a saturation function ensuring that the damping is bounded from below by a positive constant minimum damping $k_{d,min}$. This is introduces to guarantee that the system remains stable.

Figure 7.15 shows the translational velocity of the AV in the circle tracking example from before with constant damping factor $k_d = 8$. The only difference from the previous example is that the initial position of the AV is on the path.

Figure 7.16 shows the same simulation, but this time with the adaptive damping applied. The integration time is set to $T_d = 0.1$, the minimum damping to $k_{d,min} = 8$, and the

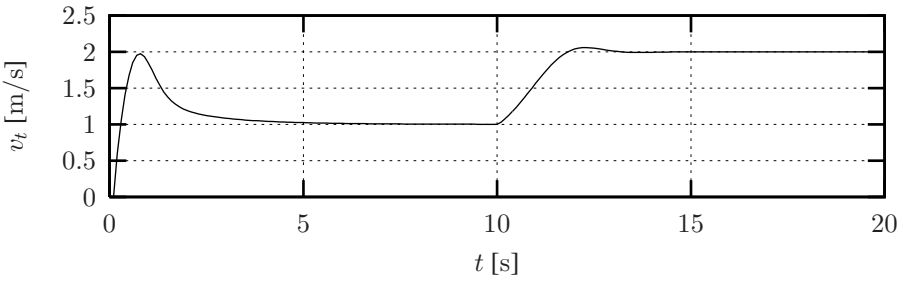


Figure 7.16: *Velocity of the AV during path tracking with adaptive damping*

desired velocity is set to $v_{t0} = 1\mathrm{m/s}$ for $t < 10\mathrm{s}$ and then changed to $v_{t0} = 2\mathrm{m/s}$ for $t \geq 10\mathrm{s}$.

The figure shows that the desired velocities are reached asymptotically with a considerable overshoot in the beginning. If desired, the overshoot can avoided by choosing a more appropriate initial value of k_d. In this simulation $k_d = 8$ at $t = 0\mathrm{s}$.

7.5.4 Putting it All Together

The combined feedback with energy shaping, damping injection, integral feedback, and adaptive damping is

$$\begin{aligned}\tau_\phi = &- C_2(\beta')\Sigma(\beta')\frac{r_w\Sigma^T(\beta)R(\chi)}{\Sigma^T(\beta)C_2^T(\beta)C_2(\beta')\Sigma(\beta')}\frac{\partial\bar{U}}{\partial\chi} && \text{(energy shaping)}\\ &- k_d\dot{\phi} && \text{(damping injection)}\\ &- k_d B_\phi^T(\beta')k_I(\tilde{p}_1 - p_I), && \text{(integral feedback)}\end{aligned}$$

with the derivative of p_I

$$\dot{p}_I = -\Upsilon(\beta')\Sigma^T(\beta)R(\theta)\frac{\partial\bar{U}}{\partial\chi},$$

and the adaptive damping coefficient

$$\begin{aligned}\dot{k}_d &= \frac{1}{T_d}(v_t - v_{t0}), \qquad v_t = \sqrt{\dot{x}_1^2 + \dot{x}_2^2}\\ k_d &= \mathrm{sat}(k_d, k_{d,min}).\end{aligned}$$

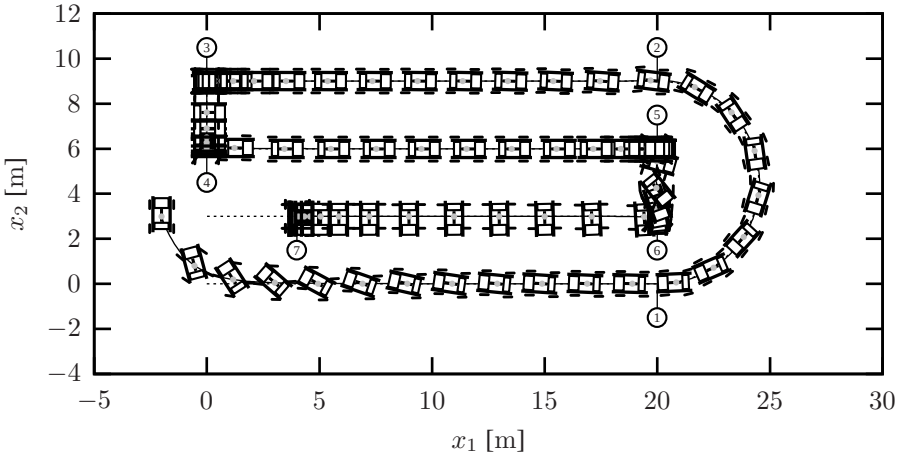


Figure 7.17: *Tracking four crop rows*

Last, but not least, the references to the steering motors are

$$\beta_{i,ref} = \arctan\left(\frac{\frac{\partial \bar{U}}{\partial x_2} + \kappa_i \cos(\theta + \gamma_i)\frac{\partial \bar{U}}{\partial \theta}}{\frac{\partial \bar{U}}{\partial x_1} - \kappa_i \sin(\theta + \gamma_i)\frac{\partial \bar{U}}{\partial \theta}}\right) - \theta.$$

Figure 7.17 shows an example simulation of a simple row tracking operation, where both crop row tracking and convergence toward single points are used. Along the four crop rows parallel to the x_1-axis and during the turn between point 1 and 2 the AV is tracking the path using inter crop row potentials and a constant pull from the potential function along the path. From point 3 to point 4 and again from point 5 to point 6 the AV is given a simple potential function with a single minimum at point 4 and 6 respectively. In the latter case the potential function includes an additional rotational potential that forces the AV to execute a 90° rotation.

The translational velocity of the AV is shown in figure 7.18. The figure illustrates how the adaptive damping effectively forces the AV to travel at the desired speeds. Along the four straight lines and the semicircle the desired speed is set to $1\mathrm{m/s}$. During the point stabilization between point 3 and 4 and again between point 5 and 6 no desired speed is set.

All the examples shown in this chapter are based on pure simulation of the AV. In the next chapter the control principles will be applied to the real AV.

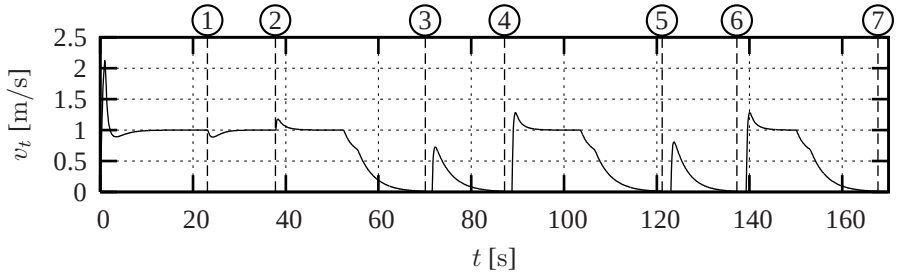


Figure 7.18: *Translational velocity of the tracking example. The numbers on top mark the points in time, when the AV reaches the correspondingly numbered way-points in figure 7.17*

7.6 Discussion

During modeling of the AV the Hamiltonian function played a crucial part in defining the dynamics, but in the energy shaping and damping injecting feedback it seems to have completely vanished. The kinematic parts are still there in the form of the matrices $C_2(\beta')$, $\Sigma(\beta')$, and $R(\theta)$, but the inertia matrix is gone. The Hamiltonian function does not enter the feedback explicitly, but implicitly through the design of the energy function $\bar{U}$. The shape and size of the shaping energy function should be designed with the inertia matrix in mind to give the closed loop system a decent performance; moving a huge mass may require a steep potential energy function to perform properly, while a small mass may require a less steep function. In other words, the feedback itself only guarantees stability and does not take performance into account, and it is up to the designer of $\bar{U}$ to define the performance.

The performance of a dynamic system is related to the concept of time (just consider the rise time and settling time of linear systems), and the 'lack' of performance in the feedback is also related to the 'lack' of time dependency when tracking, i.e., the AV is able to track paths and not trajectories. Tracking paths is preferred to tracking trajectories in the application of the AV, since trajectory tracking along the crop rows implies that we need to set the exact traveling speed of the AV at any point on the trajectory. Some parts of the field may be difficult to traverse and the desired velocity may not be met. In these situations the AV is likely to fall behind the trajectory, which can have undetermined side effects. By using path tracking and adaptive damping instead, the velocity of the AV is allowed to drop below the desired velocity without sacrificing stability and convergence to the path.

CHAPTER 8

PHYSICAL TESTS

The control algorithms derived in chapter 7 will now be applied to the physical system with all its limitations such as noisy measurements, higher order dynamics, and actuator saturation. Note that every graph and figure in this chapter is based on data collected from the real system.

8.1 Convergence toward a Single Point

The first test is to see if the AV is able to converge asymptotically to a single point. The following potential function is used

$$\bar{U} = \frac{1}{2}(\chi - \chi_0)^T K_p (\chi - \chi_0), \qquad \chi_0 = \begin{bmatrix} 0 \\ 0 \\ -\frac{\pi}{2} \end{bmatrix}, \qquad K_p = \begin{bmatrix} 1600 & 0 & 0 \\ 0 & 1600 & 0 \\ 0 & 0 & 1600 \end{bmatrix},$$

and the damping factor is fixed at $k_d = 8$. The AV is started from rest to the west of the target at orientation $\theta(0) = 0$. Figure 8.1 shows the path taken by the AV with this potential function and with no integral action ($k_I = 0$). The AV is drawn for every 2s, and the shaded AV shows the desired configuration at the target χ_0. The AV is instructed

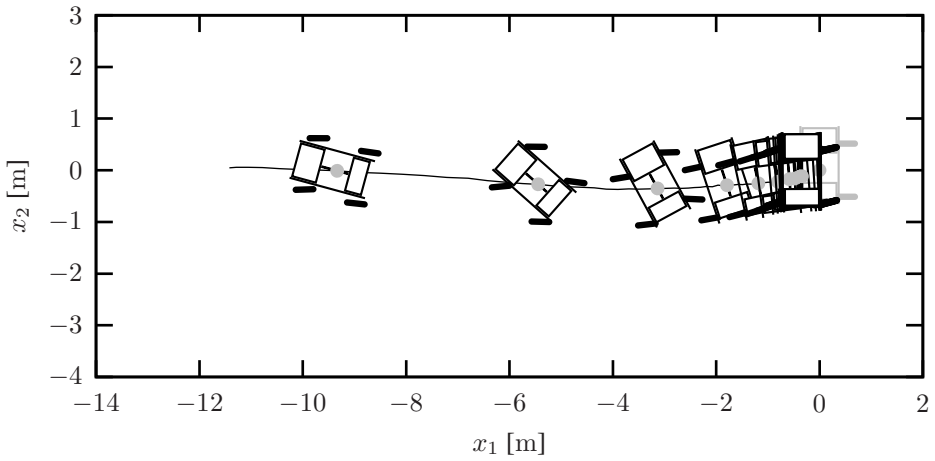


Figure 8.1: *Converging to a single point*

to stop when it reaches the target configuration, but without integral action, the AV is prevented from converging asymptotically due to a external disturbances. Figure 8.2 shows the same test, but now integral action is turned on ($k_I = 0.001$) when the AV is within $5\mathrm{m}$ of the target. In this case the AV smoothly converges to the desired target with asymptotically vanishing errors. The errors for both the tests are shown in figure 8.3.

8.2 Tracking a Line

In this section the AV's ability to track a single straight line at $x_2 = 0$ at constant velocity during various load conditions is tested. To track a line the AV is given the following potential function

$$\bar{U} = \frac{1}{2} K_c \sum_{i=1}^{4} (x_2 + \kappa_i \sin(\gamma_i + \theta) - \delta_i)^2 - K_a x_1,$$

with constants $K_c = 320$, $K_a = 1600$, $\delta_i = 0.5$ for $i = 1, 2$, and $\delta_i = -0.5$ for $i = 3, 4$. The first part of the potential function comprises four inter row potentials, one for each wheel. Each potential is a quadratic function of the distance between the i'th wheel and the line $x_2 = \delta_i$. The second part of the potential gives the constant pull along the line.

Figure 8.4 shows a test with a constant damping factor $k_d = 7$, i.e, there is no adaptive damping. The figure shows that the AV quickly converges to the path and stays there for the duration of the test. Since there is no adaptive damping there is little control of the

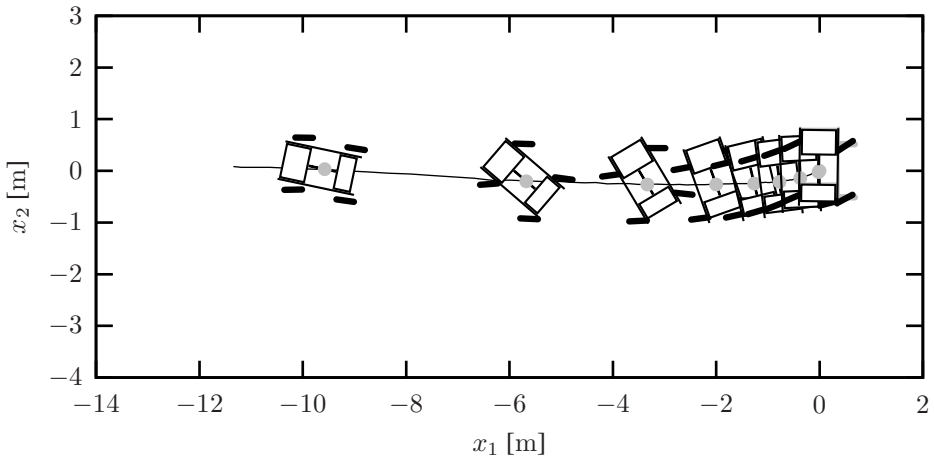


Figure 8.2: *Converging to a single point with integral action*

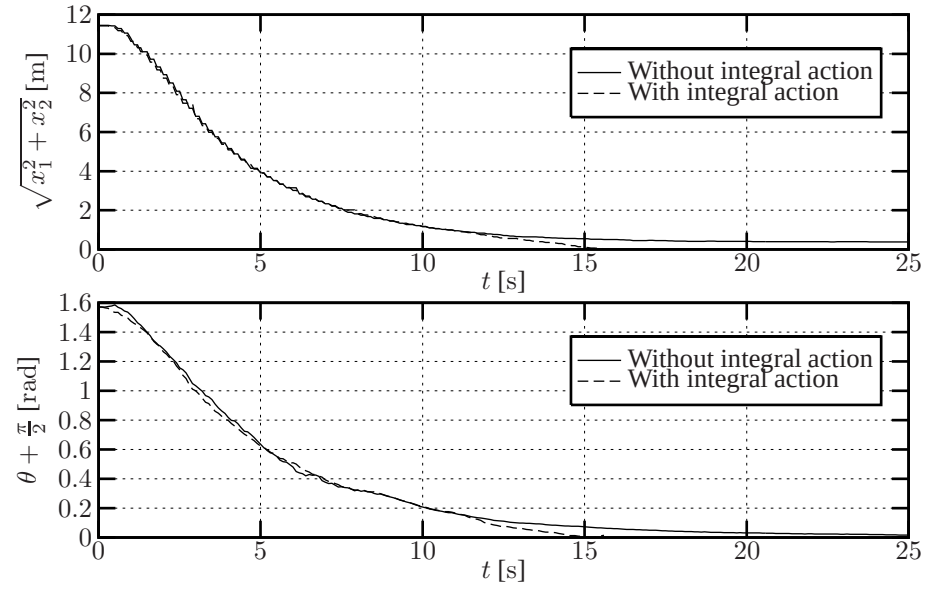


Figure 8.3: *Position and orientation errors with and without integral action*

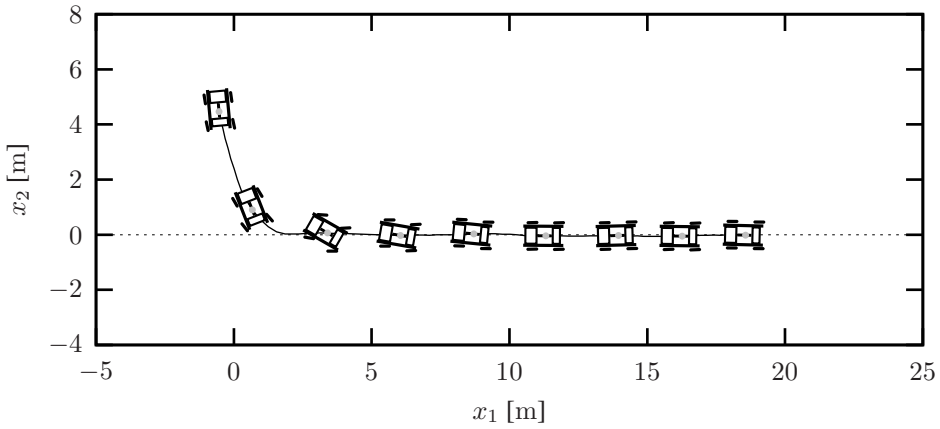


Figure 8.4: *Tracking a line. Constant damping*

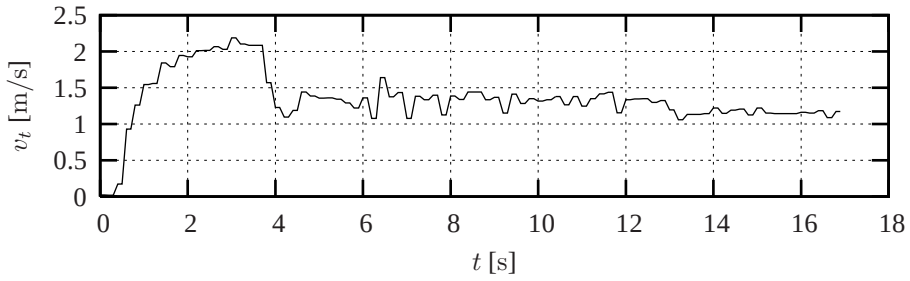


Figure 8.5: *Velocity of the line tracking test with constant damping*

resulting velocity along the path. Figure 8.5 shows the translational velocity of the AV as measured by the GPS receiver. In the beginning, when the AV is far away from the path, the velocity reaches a maximum of around $2\mathrm{m/s}$. This upper limit is determined by saturation limits in the propulsion motors, and the AV reaches its maximum speed due to a large contribution from the inter crop row potentials. When the AV reaches the path the inter crop row potentials vanish quadratically, which explains the sudden drop in velocity at around $t = 3.8\mathrm{s}$. After that, the AV continues along the path at a constant velocity until $t = 13\mathrm{s}$. At this point an external disturbance is introduces by dropping a $40\mathrm{kg}$ anchor behind the AV, see figure 8.6. With a constant damping factor the velocity of the AV drops slightly, which was expected since there is no effort to maintain a constant velocity.

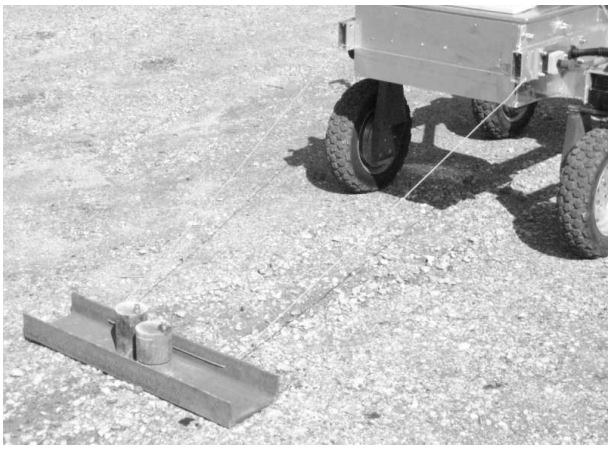


Figure 8.6: *The* $40\mathrm{kg}$ *anchor used as external disturbance*

The same test *with* adaptive damping is shown in figures 8.7 and 8.8. An integration time of $T_d = 0.2$ and a desired velocity of $v_{t0} = 1\mathrm{m/s}$ are used. The ability to track the line is unchanged, but now the velocity stabilizes at the desired velocity and quickly returns to it after introducing the disturbance at $t = 17\mathrm{s}$. The oscillations from $t = 3\mathrm{s}$ to $t = 7\mathrm{s}$ are due to a relatively short integration time T_d that makes the damping factor k_d fluctuate. The oscillations could have been avoided by choosing a larger integration time, but then the controller would not have been so quick to reach the desired velocity when the disturbance is introduced. A better solution would be to fix the damping factor until the AV is close to the path and then turn on the adaptive damping, but this will not be pursued further.

8.3 Tracking a Circle

We now turn to the physical implementation of the circle tracking example of section 7.5.2. We wish to clockwise track a circle with radius r_0 and centered at the origin of the $\mathcal{N}$-frame. In the first test a potential function with inter row potentials is used (it is the same function that was used to generate the gradient (7.19) on page 118)

$$\bar{U} = \frac{1}{2}K_c \sum_{i=1}^{4} \left(|w_i| - (r_0 + \delta r_i)\right)^2 + K_a \psi. \tag{8.1}$$

r_0 is the radius of the circle, $|w_i|$ is the distance from the center of the circle to the i'th wheel. Wheel 1 and 2 should track the inter row spacing outside the circle, and wheel 3

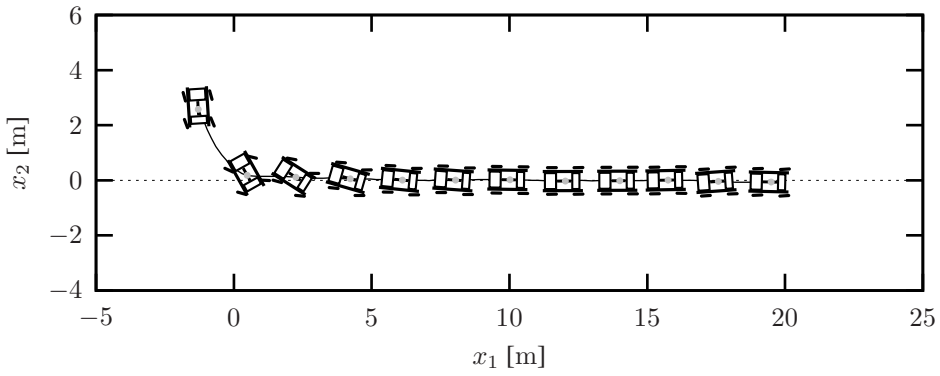


Figure 8.7: *Tracking a line. Adaptive damping*

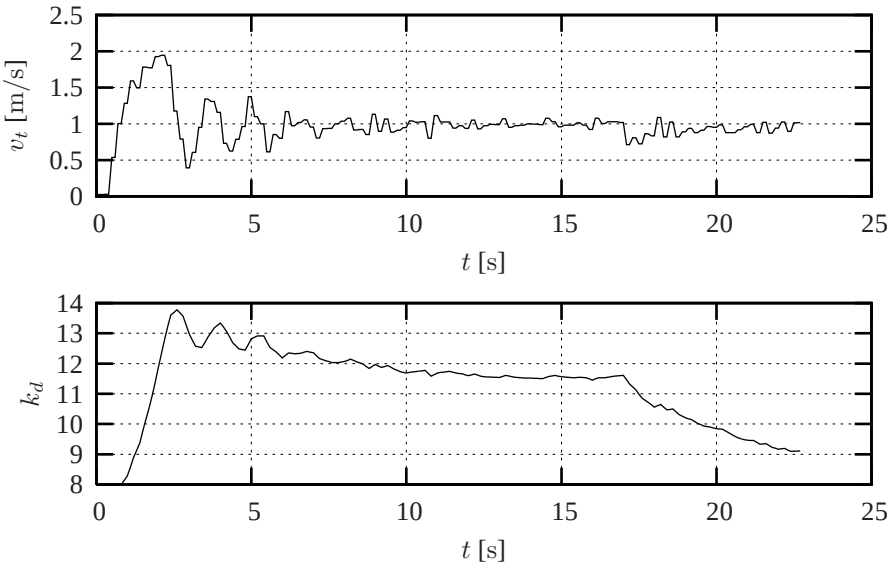


Figure 8.8: *Velocity and damping values of the line tracking test with adaptive damping*

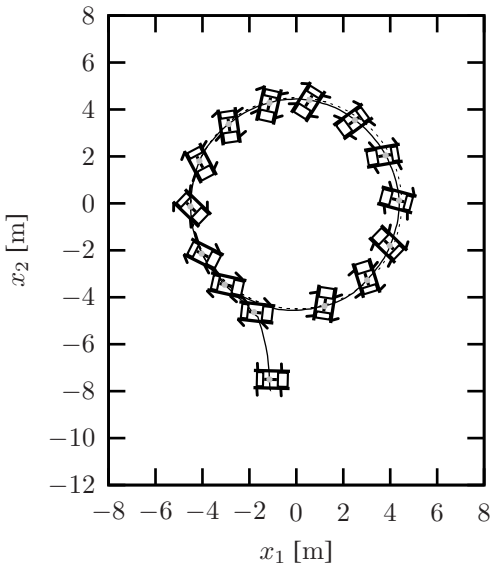


Figure 8.9: *Tracking a circle.* $K_c = 320$, $K_a = 1600 r_0$

and 4 should track the inter row spacing inside the circle, i.e., $\delta r_i = 0.5$ for $i = 1, 2$ and $\delta r_i = -0.5$ for $i = 3, 4$. ψ is the angle of the line connecting the center of the circle to the geometric center of the AV.

Figure 8.9 shows how the AV behaves with one choice of K_c and K_a. From an initial position below the circle the AV quickly converges to the circle and stays there with little variation for the duration of the test. Because of the curvature of the path the wheels never hit the inter row spacing, and at the end of the test, when the AV has reached a steady velocity, the orientation of the AV is almost perpendicular to the tangent of the circle. This deviation from the inter row spacing is due to a low K_c, and a better inter row tracking can be achieved by increasing K_c (see for example the simulation in figure 7.14 on page 119), thereby punishing deviations from the inter row spacing more severely. Unfortunately, K_c also determines the rate of translational convergence of the AV toward the circle, and increasing K_c to much more than 400 will result in a huge net pull of the AV, even at small errors. This, in turn, results in jagged, non-smooth, and generally unwanted motion of the AV (the almost 90° turn on figure 7.14 is a good example of this). Clearly, this is a drawback of using inter row potentials, and a different potential should therefore be used when tracking paths with a large curvature. One of many options is the following function

$$\bar{U} = \frac{1}{2} K_c \left(r - r_0\right)^2 + \frac{1}{2} K_\theta \left(\theta - \left(\psi + \frac{\pi}{2}\right)\right)^2 + K_a \psi. \tag{8.2}$$

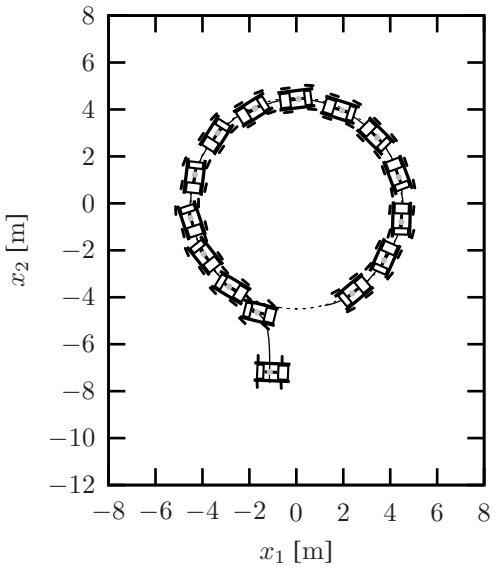


Figure 8.10: *Tracking a circle.* $K_c = 1280$, $K_\theta = 1600$, $K_a = 1600 r_0$

The first part has a minimum when the AV is on the circle, the second part has a minimum when the AV is oriented along the tangent to the circle, and the last (unchanged) part accounts for pulling the AV along the circle. With the introduction of the third design parameter K_θ the rate of convergence of the orientation and the position can be controlled independently. Figure 8.10 shows the behavior of the AV with this potential function. With the additional design parameter K_θ the AV is now able to smoothly track the circle while maintaining an orientation along the tangent of the circle.

Whether the inter row potential function (8.1) or the function (8.2) is best suited depends on the situation at hand. This example just illustrates that the motion of the AV it greatly influenced by the structure of the potential function.

8.4 Putting it All Together

As a last test the different modes of operation – convergence toward a single point, tracking a line, and tracking a circle – are combined in a test where the AV drives along four parallel lines. The test is the physical implementation of the simulation in figure 7.17 on page 122. Figure 8.11 shows the configuration of the AV during the test. The wheels of

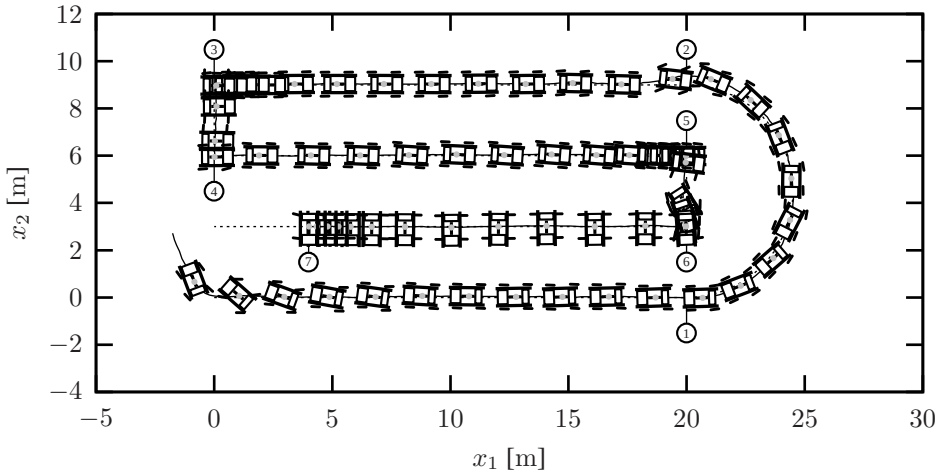


Figure 8.11: *Tracking four crop rows in practice*

the AV are kept well away from the crop row represented by the straight lines, except at the very beginning when the AV approaches the first row. The desired speed was set to $1\mathrm{m/s}$ along the rows and the semicircle between point 1 and 2, and figure 8.12 shows that it is precisely met.

8.5 Discussion

The tests shown in this chapter are all based on convergence to simple geometric objects, such as points, straight lines, and circles. The simplicity of these objects facilitates easy construction of the potential function $\bar{U}$, but a general path or crop row in a field may comprise more than just lines and circles. On the other hand, one should note that the control algorithm does not need access to the complete potential function, but just the gradient of $\bar{U}$ at the instantaneous configuration of the AV on the manifold. The complete potential function may be difficult to generate in the general case, but the gradient is easily reconstructed from local measurements, since the gradient represents the positional error between the AV and the target path or point. The crop row camera (see page 26), for example, outputs the offset and orientation error relative to a nearby crop row and these two measurements can directly be related to the gradient if the object is to track the row.

From the tests shown in this chapter it is concluded that the feedback controlled AV behaves as expected from the simulation. Even with noisy sensor data. The data collected from the sensors are in most cases unfiltered when entering the feedback loop. The only

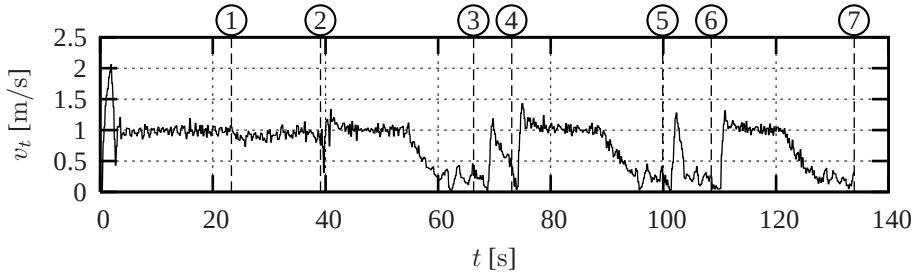


Figure 8.12: *Translational velocity of the tracking example. The numbers on top mark the points in time, when the AV reaches the correspondingly numbered way-points in figure 8.11*

exception is the estimated orientation of the AV, which is a combination of the compass reading and the integral of the gyro. By controlling the AV based on a dynamic model of the vehicle, the dynamics of the AV itself acts as a kind of filter. The damping injecting feedback, for example, is based on the measured velocity from the GPS receiver. This signal is quite noisy, but since it is passed through the natural dynamics of the AV the resulting motion of the AV is still smooth.

What have been accomplished here is the design and implementation of a smooth time invariant controller for the AV that is general enough to deal with both asymptotic stabilization and path tracking. A second and important property of the controller is the absence of any singularities. It has a singularity when $\beta_1 = \beta_2 = \frac{\pi}{2}$, but this is related to the structure of the AV and not the controller. In [66] a linearization of a the same type of robot was introduced, but it was also shown that the linearizing feedback is only defined at a nonzero velocity. This is not a problem for path tracking applications, but for asymptotic stabilization some other method has to be applied instead. By avoiding linearization all together and exploiting the nonlinear structures of the AV the resulting passivity based controller is defined for any configuration and any velocity of the AV.

CHAPTER 9

CONCLUSIONS AND FUTURE WORK

This thesis has focused on modeling and feedback control of a class of nonholonomic systems. The major contribution has been the introduction of so-called kinematic inputs in the framework of Lagrangian and Hamiltonian systems. Previous results on feedback control of classical nonholonomic Hamiltonian systems have been shown to be applicable for this type of system as well, but the introduction of kinematic inputs has proved to provide for the design of a global asymptotically stabilizing feedback. The feedback control of Hamiltonian systems with kinematic inputs has been simulated and tested successfully on a real nonholonomic system. The system in question was a four wheel steered, four wheel driven mobile robot (the AV) used for surveying crops and weeds in an agricultural field. For proper operation, the AV has to be able to asymptotically stabilize itself and follow crop rows to minimize crop damage. Both of these operational modes were provided by the proposed controller.

9.1 Summary of the Results

In chapter 3 a full dynamic and unreduced model of the AV was introduced based on the Lagrange equation for nonholonomic systems. The motion of the AV was assumed

to satisfy free rolling and non-slipping constraints, and the appropriate expressions for these nonholonomic constraints were incorporated in the model. Some considerations were discussed where a part of the configuration coordinates (the steering angles) was separated from the rest of the coordinates to lay the grounds for defining the notion of kinematic inputs.

In chapter 4 the kinematic inputs were formally defined in the context of general nonholonomic Lagrangian systems. A kinematic input was defined as an input that could control a subset of the configuration coordinates through its first time derivative. Furthermore, the total physical energy of the system had to be invariant with respect to this subset. The link between a Lagrange system with kinematic inputs and the corresponding Hamiltonian system was also given in this chapter, and a reduction scheme that eliminated the Lagrange multipliers was introduced. The reduction scheme was a coordinate transformation based on results by van der Schaft and Maschke [54], and it was shown that the reduction also applied to systems with kinematic inputs. The reduced system was again a Hamiltonian system, but with an additional input that was a result of including kinematic inputs. Finally, this chapter introduced an additional constraint on the coordinate transformation that was able to eliminate the additional input. The final reduced system was on a simple form that made it particularly useful for control.

The results from chapter 4 was applied to the model of the AV in chapter 5. It was first checked that the steering angle velocities of the AV were fully qualified kinematic inputs. Once this was established the reduction scheme was applied to the model. The resulting model was then validated based on measurements on the real AV, and the model was seen to precisely predict the behavior of the real system.

Chapter 6 dealt with feedback control of the reduced nonholonomic Hamiltonian system with kinematic inputs. The object was to asymptotically stabilize the system at a desired closed set Q_0 on the configuration manifold. The set Q_0 was defined as the set of minima of a potential energy function. An energy shaping and damping injecting feedback was introduced, and the closed loop dissipative system was shown to converge to an open set Q. The set Q contained Q_0, but was generally larger, and asymptotic stability toward Q_0 was not yet achieved. This was where the kinematic inputs came into play. By designing a proper feedback for the kinematic inputs it was possible to force the set Q to only contain Q_0, and asymptotic stability was achieved. A feedback for the kinematic inputs was not guaranteed to exist, so a sufficient condition for existence was also given. The condition was satisfied for the four wheel steered AV and other systems with a similar degree of mobility.

The feedback was applied to the model of the AV in chapter 7 to yield a closed loop dissipative system that enabled global asymptotic stabilization at an arbitrary position and orientation on a horizontal field. While the four wheel steer made it possible to asymptotically stabilize the AV, the four wheel drive supplied the freedom to define the

propulsion torque distribution between the four DC drive motors. Many torque distributions were possible, and a distribution that minimized the electrical power input to the motors was proposed.

The energy shaping and damping injecting feedback was by construction very similar to a PD controller for a linear system, and it inherited some of the same limitations; if there were external disturbances to the system the asymptotic stability was sacrificed. To solve this problem integral action was introduced to deal with constant disturbances, such as non-horizontal fields, soil tracks, rocks, etc. The integral feedback was chosen such that the closed loop system was again on a Hamiltonian form. By imposing this structure on the feedback the system was guaranteed to be stable, even when choosing high feedback gains.

Asymptotic stabilization of the AV was not enough for proper operation in the field. The AV also had to be able to track crop rows. This was achieved by designing a potential energy function, where the path was represented by a cleft or valley in the function. This function was then fed back through the energy shaping feedback, and path tracking were achieved with the same controller structure. When traveling along the path, the velocity of the AV was determined by the steepness of the potential energy function and the amount of damping, and to give the system a certain degree of velocity control, an adaptive damping scheme was introduced. By changing the damping it was possible to control the steady state velocity without sacrificing the useful dissipative property of the closed loop system.

The proposed feedbacks were tested on a simulation model of the AV in chapter 7, and in chapter 8 they were tested on the real AV, which was subject to unknown disturbances, actuator saturation, unmodeled dynamics, etc. The closed loop system was proved to perform as expected with very little discrepancies between real and simulated outputs.

9.2 Recommendations for Future Work

Three important issues that have not received attention in the thesis are worth noting here. These include both practical issues and problems of a more theoretical nature.

1. To be able to asymptotically stabilize a nonholonomic Hamiltonian system with kinematic inputs condition (6.10) on page 91 should be satisfied. It is easily checked for the AV, but in general, checking the validity of this condition is not an easy task. A local proof is easy to find based on the inverse function theorem, but this must be extended to a global proof if we wish to prove the existence of a global asymptotically stabilizing feedback.

2. There has been very little attention on the performance of the closed loop system, and any performance requirements have been left to be resolved by the designer of the potential shaping function. An unambiguous concept of performance is always difficult to define for nonlinear systems, but the Hamiltonian structure might help. In the authors opinion it would be worthwhile to construct a suitable performance index so that energy shaping and damping injecting feedbacks for the AV could be evaluated for performance. Or an index that allows for a qualitative comparison between the passivity based controller presented here and existing controllers for vehicles of the same type.

3. The simulation model of the AV was a set of continuous differential equations, and simulation was done based on a high-order numerical integration method. The physical implementation of the feedback, on the other hand, was based on an less precise approximation where the inputs to the actuators were passed through a zero-order hold filter. This had no visible effect on the results, since the sampling period was short. Increasing the sampling period will almost certainly have a detrimental effect on the stability and performance of the closed loop system, and it should be further investigated just how far we can push the system without destabilizing it.

APPENDIX A

HAMILTON'S PRINCIPLE AND LAGRANGE'S EQUATION

This appendix gives a short introduction to *Hamilton's principle of least action* and is based on excerpts from the two books [30] and [41]. Hamilton's principle is a very basic principle, and it applies to a wide range of physical systems. Newton's laws of motion are just one example of equations that can be deduced from this principle. Because of the generality of Hamilton's principle it is well suited to handle mechanical systems with nonholonomic constraints, and in the end of the chapter the principle will be used to derive the extended Lagrange's equation for nonholonomic systems.

A.1 Lagrange's Equation

We start by introducing Hamilton's principle for a conservative unconstrained *monogenic* system and use it to derive the classical Lagrange's equation. The term monogenic indicates that all forces acting on the system are generated by a single potential function, and this function is only depending on the position coordinates of the system (consider for example a mass moving in a gravitational field). The position and velocity of the

system are described by the generalized coordinated $q_i,\ i = 1, 2, \ldots, n$ and their corresponding generalized velocities $\dot{q}_i,\ i = 1, 2, \ldots, n$. For the time being it is assumed that the system is holonomic, which implies that the q_is and $\dot{q}_i$s are independent. Constraints on the $\dot{q}_i$s will be introduced later when moving to nonholonomic systems. Hamilton's principle states that the motion from time t_1 to time t_2 is such that the line integral

$$I = \int_{t_1}^{t_2} L dt, \qquad L = T - U,$$

is stationary for any arbitrary variations of the correct path between t_1 and t_2 – provided all these variations vanish at t_1 and t_2. The scalar function $L(q_1, \ldots, q_n, \dot{q}_1, \ldots, \dot{q}_n, t)$ is called the Lagrangian function and is defined as the difference between kinetic energy T and potential energy U.

Stationarity of a line integral implies that the integral along the correct path has the same value as the integral along any neighboring path to within first order. Consider the variations in the coordinates

$$\begin{aligned} q_1(t, \alpha) &= q_1(t, 0) + \alpha \eta_1(t), \\ &\vdots \\ q_n(t, \alpha) &= q_n(t, 0) + \alpha \eta_n(t), \end{aligned}$$

where the η_is are arbitrary independent function with the only constraints that they vanish at t_1 and t_2 and that they are continuous through their second derivative. The line integral I is stationary if the *variation* of I is zero

$$\delta I = \delta \int_{t_1}^{t_2} L(q_1, \ldots, q_n, \dot{q}_1, \ldots, q_n, t) dt = 0. \tag{A.1}$$

The variation of I is defined as

$$\delta I = \frac{\partial I}{\partial \alpha} d\alpha,$$

and (A.1) is

$$\delta I = \int_{t_1}^{t_2} \sum_i \left(\frac{\partial L}{\partial q_i} \frac{\partial q_i}{\partial \alpha} d\alpha + \frac{\partial L}{\partial \dot{q}_i} \frac{\partial \dot{q}_i}{\partial \alpha} d\alpha \right) dt = 0. \tag{A.2}$$

Integrating the second term by parts

$$\int_{t_1}^{t_2} \frac{\partial L}{\partial \dot{q}_i} \frac{\partial \dot{q}_i}{\partial \alpha} dt = \left[\frac{\partial L}{\partial \dot{q}_i} \frac{\partial q_i}{\partial \alpha} \right]_{t_1}^{t_2} - \int_{t_1}^{t_2} \frac{\partial q_i}{\partial \alpha} \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) dt.$$

Because η_i vanishes at the end points the first term is zero and (A.2) becomes

$$\delta I = \int_{t_1}^{t_2} \sum_i \left(\frac{\partial L}{\partial q_i} - \frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} \right) \delta q_i dt = 0, \tag{A.3}$$

where the variation or *virtual displacement* of q_i is

$$\delta q_i = \frac{\partial q_i}{\partial \alpha} d\alpha.$$

The term virtual displacement refers to an infinitesimal displacement of the configuration coordinate consistent with any forces and constraints imposed on the system at time instance t. For the time being we only consider unconstrained movement, but later in this appendix nonholonomic constraints will be introduced, which will limit the set of possible displacements. The displacement is called virtual to distinguish it from an actual displacement occurring in a time interval dt, during which forces and constraints may change.

Since the q_is are independent (no constraints), the virtual displacements δq_is are also independent. The condition that $\delta I = 0$ implies that each coefficient to the δq_is vanish separately[1]. This implies that

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = 0, \qquad i = 1, 2, \ldots, n. \tag{A.4}$$

This is the well known Lagrange's equation for a conservative monogenic system.

As a control system the basic Lagrange's equation is of little use, because it does not provide any controllable inputs to the system. In the monogenic system is was assumed that all the forces in the system was derivable from a single potential function U, which only depended on the q_is. That is

$$\bar{Q}_i = -\frac{\partial U}{\partial q_i},$$

and Lagrange's equation can be rewritten as

$$\frac{d}{dt}\frac{\partial T}{\partial \dot{q}_i} - \frac{\partial T}{\partial q_i} = \bar{Q}_i, \qquad i = 1, 2, \ldots, n.$$

Whether $\bar{Q}_i$ is an internal force of a system with $L = T - U$, or an externally applied generalized force to a system with no potential energy ($L = T$), the motion of the system will be the same. We conclude that if the forces acting on the system consists of both an internal part derived from a potential function and an external part Q_i, then Lagrange's equation can be extended to cover system with external inputs as well

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = Q_i, \qquad i = 1, 2, \ldots, n. \tag{A.5}$$

[1] We use the *fundamental lemma* of the calculus of variations. The lemma states that if $\int_a^b M(x)h(x)dx = 0$, $\forall h(x)$ with continuous second partial derivatives, then $M(x) = 0$ [4].

A.2 Extension to Nonholonomic Systems

In the step going from (A.3) to Lagrange's equation (A.4) it was assumed that the virtual displacements δq_i were all independent, but this is not true for nonholonomic systems. A system is nonholonomic if there exists constraints that cannot be expressed as holonomic constraints between the configuration coordinates, as in

$$f(q_1, \ldots, q_n, t) = 0. \tag{A.6}$$

In many applications (including mobile robots) nonholonomic constraints are encountered as linear relationships between the differentials of the q_is

$$\sum_i a_{li} dq_i + a_{lt} dt = 0, \tag{A.7}$$

where $l = 1, \ldots, m$ indicates that there may be more than one constraint. Note that the a_{li}s and a_{lt} may depend on both time and the generalized coordinates. Sometimes the constraints can be integrated to yield constraints on the form (A.6), but then it is a holonomic constraint. From the definition of the virtual displacement the nonholonomic constraints can be viewed as constraints on the δq_is

$$\sum_i a_{li} \delta q_i = 0. \tag{A.8}$$

The virtual displacements are hence no longer independent, and we need to reduce the n equations of (A.3) to independent ones. The trick to do this is to introduce *Lagrange undetermined multipliers*. If (A.8) holds, then it is also true that

$$\lambda_l \sum_i a_{li} \delta q_i = 0$$

for some undetermined quantities $\lambda_l,\ l = 1, \ldots, m$. Summing over l and integrating the result from t_1 to t_2 yields

$$\int_{t_1}^{t_2} \sum_{i,l} \lambda_l a_{li} \delta q_i dt = 0.$$

This equation can then be added directly to (A.3)

$$\delta I = \int_{t_1}^{t_2} \sum_i \left(\frac{\partial L}{\partial q_i} - \frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} + \sum_l \lambda_l a_{li} \right) \delta q_i dt = 0. \tag{A.9}$$

The δq_is are still not independent though. The first $n - m$ equations may be chosen independently, but the remaining m are fixed by (A.8). The Lagrange multipliers, however, are still at our disposal. Suppose that we choose the λ_ls such that

$$\frac{\partial L}{\partial q_i} - \frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} + \sum_l \lambda_l a_{li} = 0, \qquad i = n - m + 1, \ldots, n. \tag{A.10}$$

With the λ_ls determined we know that the last m equations of the sum in (A.9) are all zero, but we still have to satisfy

$$\int_{t_1}^{t_2} \sum_i \left(\frac{\partial L}{\partial q_i} - \frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} + \sum_l \lambda_l a_{li} \right) \delta q_i dt = 0, \qquad i = 1, \ldots, n-m.$$

The only δq_is involved are the independent ones, and we conclude that it is satisfied if and only if

$$\frac{\partial L}{\partial q_i} - \frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} + \sum_l \lambda_l a_{li} = 0, \qquad i = 1, \ldots, n-m \tag{A.11}$$

(again using the *fundamental lemma*). Combining (A.10) and (A.11) and adding an external input by the same reasoning, which led to (A.5), we end up with the final Lagrange's equation for nonholonomic systems

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = \sum_l \lambda_l a_{li} + Q_i, \qquad i = 1, \ldots, n. \tag{A.12}$$

This is not enough to describe the motion though. We have introduced the m Lagrange multipliers, which are generally functions of the coordinates and time, and hence increased the system to having $2n + m$ unknowns, but the Lagrange's equations (A.12) only gives a total of n second order differential equations. The last m equations are embedded in the constraint equations (A.7), but this time they are considered as first order differential equations

$$\sum_i a_{li}\dot{q}_i + a_{lt} = 0. \tag{A.13}$$

The two sets of equations (A.12) and (A.13) can be also be written on a more compact matrix form

$$\begin{aligned} \frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} &= A^T(q,t)\lambda + Q, \\ 0 &= A(q,t)\dot{q} + A_0(q,t), \end{aligned}$$

with

$$q = \begin{bmatrix} q_1 \\ \vdots \\ q_n \end{bmatrix}, \quad Q = \begin{bmatrix} Q_1 \\ \vdots \\ Q_n \end{bmatrix}, \quad \lambda = \begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_m \end{bmatrix},$$

$$A(q,t) = \begin{bmatrix} a_{11} & \cdots & a_{m1} \\ \vdots & \ddots & \vdots \\ a_{1n} & \cdots & a_{mn} \end{bmatrix}, \quad A_0(q,t) = \begin{bmatrix} a_{1t} \\ \vdots \\ a_{mt} \end{bmatrix}.$$

BIBLIOGRAPHY

[1] Adams, M. D. High speed target pursuit and asymptotic stability in mobile robotics. *IEEE Transactions on Robotics and Automation*, 15(2):230–237, 1999.

[2] Alexander, J. C., Maddocks, J. H., et al. Shortest distance paths for wheeled mobile robots. *IEEE Transactions on Robotics and Automation*, 14(5):657–662, 1998.

[3] Andersen, P., Bendtsen, J. D., et al. Robust feedback linearization-based control design for a wheeled mobile robot. In *Proceedings of the 6th International Symposium on Advanced Vehicle Control*. Hiroshima, Japan, 2002.

[4] Arnold, V. I. *Mathematical Methods of Classical Mechanics*. Springer, 2nd edition, 1989.

[5] Barraquand, J. and Latombe, J.-C. On nonholonomic mobile robots and optimal maneuvering. In *Proceedings of the 4th International Symposium on Intelligent Control*, pages 340–347. Albany, New York, 1989.

[6] Barshan, B. and Durrant-Whyte, H. F. Inertial navigation systems for mobile robots. *IEEE Transactions on Robotics and Automation*, 11(3):328–342, 1995.

[7] Bisgaard, M., Vinther, D., et al. *Modelling and Fault-Tolerant Control of an Autonomous Wheeled Robot*. Master's thesis, Aalborg University, Denmark, 2004.

[8] Bloch, A. M., Chang, D. E., et al. Controlled Lagrangians and the stabilization of mechanical systems II: Potential shaping. *IEEE Transactions on Automatic Control*, 46(10):1556–1571, 2001.

[9] Bloch, A. M., Leonard, N. E., et al. Controlled Lagrangians and the stabilization of mechanical systems I: The first matching theorem. *IEEE Transactions on Automatic Control*, 45(12):2253–2270, 2000.

[10] Bloch, A. M. and McClamroch, N. H. Control of mechanical systems with classical nonholonomic constraints. In *Proceedings of the 28th Conference on Decision and Control*, pages 201–205. Tampa, Florida, 1989.

[11] Bloch, A. M., Reyhanoglu, M., et al. Control and stabilization of nonholonomic dynamic systems. *IEEE Transactions on Automatic Control*, 37(11):1746–1757, 1992.

[12] Bonnifait, P. and Garcia, G. A multisensor localization algorithm for mobile robots and its real-time experimental validation. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1395–1400. 1996.

[13] Borenstein, J. and Koren, Y. The vector field histogram - fast obstacle avoidance for mobile robots. *IEEE Transactions on Robotics and Automation*, 7(3), 1991.

[14] Brockett, R. W. and Sussmann, H. J. *Differential Geometric Control Theory*. Birkhauser Boston, 1982.

[15] Brown, R. G. and Hwang, P. Y. C. *Introduction to Random Signals and Applied Kalman Filtering*. John Wiley & Sons, 3 edition, 1997. ISBN 0-471-12839-2.

[16] Desaulniers, G. and Soumis, F. An efficient algorithm to find a shortest path for a car-like robot. *IEEE Transactions on Robotics and Automation*, 11(6):819–828, 1995.

[17] Dong, W., Huo, W., et al. Tracking control of uncertain dynamic nonholonomic system and its application to wheeled mobile robots. *IEEE Transactions on Robotics and Automation*, 16(6):870–874, 2000.

[18] Duindam, V. and Stramigioli, S. Energy-based model-reduction and control of nonholonomic mechanical systems. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*, pages 4584–4589. New Orleans, Louisiana, 2004.

[19] Duindam, V., Stramigioli, S., et al. Passive compensation of nonlinear robot dynamics. *IEEE Transactions on Robotics and Automation*, 20(3):480–487, 2004.

[20] Edwards, C. H., Jr. and Penney, D. E. *Calculus with Analytic Geometry*. Prentice-Hall Inc., 4 edition, 1994.

[21] Farrell, J. A. and Barth, M. *The Global Positioning System & Inertial Navigation*. McGraw-Hill, 1999. ISBN 0-07-022045-X.

[22] Fierro, R. and Lewis, F. L. Control of a nonholonomic mobile robot: Backstepping kinematics into dynamics. In *Proceedings of the 34th Conference on Decision & Control*, pages 3805–3810. New Orleans, LA, 1995.

[23] Fujimoto, K., Sakurama, K., et al. Trajectory tracking control of port-controlled Hamiltonian systems via generalized canonical transformations. *automatica*, 39:2059–2069, 2003.

[24] Fujimoto, K. and Sugie, T. Canonical transformation and stabilization of generalized Hamiltonian systems. *Systems & Control Letters*, 42:217–227, 2001.

[25] Fujimoto, K. and Sugie, T. Stabilization of Hamiltonian systems with nonholonomic constraints based on time-varying generalized canonical transformations. *System & Control Letters*, 44:309–319, 2001.

[26] Fukao, T., Nakagawa, H., et al. Adaptive tracking control of a nonholonomic mobile robot. *IEEE Transactions on Robotics and Automation*, 16(5):609–615, 2000.

[27] Ge, S. S. and Cui, Y. J. New potential functions for mobile robot path planning. *IEEE Transactions on Robotics and Automation*, 16(5):615–620, 2000.

[28] Gillespie, T. D. *Fundamentals of Vehicle Dynamics*. Society of Automotive Engineers, Inc., 1992. ISBN 1-56091-199-9.

[29] Godhavn, J.-M. and Egeland, O. A Lyapunov approach to exponential stabilization of nonholonomic systems in power form. *IEEE Transactions on Automatic Control*, 42(7):1028–1032, 1997.

[30] Goldstein, H. *Classical Mechanics*. Addison-Wesley, 2 edition, 1980. ISBN 0-201-02918-9.

[31] Gray, A. *Modern Differential Geometry of Curves and Surfaces*. CRC Press, 1993.

[32] Hagras, H., Callaghan, V., et al. A fuzzy-genetic based embedded-agent approach to learning & control in agricultural autonomous vehicles. In *Proceedings of the 1999 IEEE International Conference on Robotics & Automation*, pages 1005–1010. Detroit, Michigan, 1999.

[33] Hague, T., Marchant, J. A., et al. Autonomous robot navigation for precision horticulture. In *Proceedings of the 1997 IEEE International Conference on Robotics and Automation*, pages 1880–1885. Albuquerque, New Mexico, 1997.

[34] Kanayama, Y., Kimura, Y., et al. A stable tracking control method for an autonomous mobile robot. In *Proceedings of the IEEE Conference on Robotics and Automation*, pages 384–389. Cincinatti, USA, 1990.

[35] Khalil, H. K. *Nonlinear Systems*. Prentice Hall, 2 edition, 1996. ISBN 0-13-228024-8.

[36] Khennouf, H., Canudas de Wit, C., et al. Preliminary results on asymptotic stabilization of Hamiltonian systems with nonholonomic constraints. In *Proceedings of the 34th Conference on Decision & Control, New Orleans, USA*, pages 4305–4310. 1995.

[37] Kolmanovsky, I. and McClamroch, N. H. Developments in nonholonomic control problems. *IEEE Control Systems Magazine*, 15(6):20–36, 1995.

[38] Koren, Y. and Borenstein, J. Potential field methods and their inherent limitations for mobile robot navigation. In *IEEE Conference on Robot. Automat.*, pages 1398–1404. 1991.

[39] Lafferriere, G. and Sussmann, H. Motion planning for controllable systems without drift. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1148–1153. Sacramento, California, 1991.

[40] Maschke, B. M. and van der Schaft, A. A Hamiltonian approach to stabilization of nonholonomic mechanical systems. In *Proc. of the 33rd Conf. on Decision and Control, Lake Buena Vista, FL*, pages 2950–2954. 1994.

[41] Meirovitch, L. *Methods of Analytic Dynamics*. McGraw-Hill, 1970.

[42] Murray, R. M. and Sastry, S. S. Steering nonholonomic systems using sinusoids. In *Proceedings of the 29th Conference on Decision and Control*, pages 2097–2101. Honolulu, Hawaii, 1990.

[43] Nguyen, D. H. and Widrow, B. Neural networks for self-learning control systems. *IEEE Control Systems Magazine*, 10(3):18–23, 1990.

[44] Nielsen, J. D., Bendtsen, J. D., et al. A conceptual framework for design of embedded systems and data communication for autonomous vehicles. In *Proceedings of the 5th IFAC/EURON symposium on intelligent autonomous vehicles*. Lisbon, Portugal, 2004.

[45] Nielsen, K. M., Andersen, P., et al. Control of an autonomous vehicle for registration of weed and crop in precision agriculture. In *Proceedings of the IEEE Conference on Control Applications CCA/CACSD 2002*. Glasgow, Scotland, 2002.

[46] Nijmeijer, H. and van der Schaft, A. *Nonlinear Dynamical Control Systems*. Springer, 1995.

[47] Oprea, J. *Differential Geometry*. Prentice-Hall Inc., 1997.

[48] Ortega, R., van der Schaft, A., et al. Interconnection and damping assignment passivity-based control of port-controlled Hamiltonian systems. *Automatica*, 38:585–596, 2002.

[49] Ortega, R., van der Schaft, A. J., et al. Putting energy back in control. *IEEE Control Systems Magazine*, pages 18–33, 2001.

[50] Pedersen, T. S., Nielsen, K. M., et al. Development of an autonomous vehicle fore weed and crop registration. In *Proceedings of the EurAgEng 2002*. Budapest, Hungary, 2002.

[51] Samson, C. Control of chained systems. Application to path following and time-varying point-stabilization of mobile robots. *IEEE Transactions on Automatic Control*, 40(1):64–77, 1995.

[52] Sastry, S. *Nonlinear Systems: Analysis, Stability and Control*. Springer, New York, 1999. ISBN 0387985131.

[53] van der Schaft, A. *L2-Gain and Passivity Techniques in Nonlinear Control*. Springer, 2000.

[54] van der Schaft, A. and Maschke, B. M. On the Hamiltonian formulation of non-holonomic mechanical systems. *Reports on Mathematical Physics*, 34(2):225–233, 1994.

[55] Soetano, D., Lapierre, L., et al. Adaptive, non-singular path-following control of dynamic wheeled robots. In *Proceedings of the 42nd IEEE Conference on Decision and Control*, pages 1765–1770. Maui, Hawaii, 2003.

[56] Søgaard, H. T. and Heisel, T. Machine vision identification of weed species based on active shape models. In *Proceedings of the European Weed Research Society 12th international symposium*. Wageningen, The Netherlands, 2002.

[57] Søgaard, H. T. and Heisel, T. Weed classification by active shape models. In *Proceedings of the EurAgEng 2002*. Budapest, Hungary, 2002.

[58] Sørdalen, O. J. and Canudas de Wit, C. Exponential control law for a mobile robot: Extension to path following. In *Proceedings of the 1992 IEEE International Conference on Robotics and Automation*, pages 2158–2163. Nice, France, 1992.

[59] Sørensen, C. G., Olsen, H. J., et al. Planning and operation of an autonomous vehicle. In *Proceedings of the 2002 ASEA / CIGR XVth World Congress*. Chicago, Illinois, 2002.

[60] Sørensen, M. J. *Modeling and Control of Wheeled Farming Robot*. Master's thesis, Aalborg University, Department of Control Engineering, 2002. Http://www.control.auc.dk/■mjs/publications/sorensen_01_2002.pdf.

[61] Sørensen, M. J. Artificial potential field approach to path tracking for a non-holonomic mobile robot. In *Proceedings of the 11th Mediteranean Conference on Control And Automation*. Rhodes, Greece, 2003.

[62] Sørensen, M. J., Bendtsen, J. D., et al. Asymptotic stabilization of non-holonomic port-controlled Hamiltonian systems. In *Proceedings of the 5th IFAC/EURON Symposium on Intelligent Autonomous Vehicles*. Lisbon, Portugal, 2004.

[63] Sørensen, M. J., Bendtsen, J. D., et al. Stabilization and path tracking of non-holonomic systems with kinematic inputs. *IEEE Transactions on Robotics*, 2006. Submitted for publication.

[64] Strang, G. *Linear Algebra*. Wellesley-Cambridge Press, 2 edition, 1998. ISBN 0-9614088-5-5.

[65] Thuilot, B., Cariou, C., et al. Automatic guidance of a farm tractor along curved paths, using a unique CP-DGPS. *Proceedings of the International Conference on Intelligent Robots and Systems*, pages 674–679, 2001.

[66] Thuilot, B., d'Andra Novel, B., et al. Modeling and feedback control of mobile robots equipped with several steering wheels. *IEEE Transactions on Robotics and Automation*, 12(3):375–390, 1996.

[67] Thuilot, B., d'Andra Novel, B., et al. Structural properties and classification of kinematic and dynamic models of wheeled robots. *IEEE Transactions on Robotics and Automation*, 12(1):47–62, 1996.

[68] Walsh, G., Tilbury, D., et al. Stabilization of trajectories for systems with non-holonomic constraints. *IEEE Transactions on Automatic Control*, 39(1):216–222, 1994.

[69] Williams, R. L., II, Carter, B. E., et al. Dynamic model with slip for wheeled omnidirectional robots. *IEEE Transactions on Robotics and Automation*, 18(3):285–293, 2002.

[70] Canudas de Wit, C. and Sørdalen, O. J. Exponential stabilization of mobile robots with nonholonomic constraints. In *Proceedings of the 30th Conference on Decision and Control*, pages 692–697. Brighton, England, 1991.

[71] Wong, J. Y. *Theory of Ground Vehicles*. John Wiley & Sons, 1976. ISBN 0-471-03470-3.

[72] Yang, J.-M. and Kim, J.-H. Sliding mode control for trajectory tracking of non-holonomic wheeled mobile robots. *IEEE Transactions on Robotics and Automation*, 15(3):578–587, 1999.