HOKKAIDO UNIVERSITY

Title	Motion Control and Stability Improvement of Autonomous Mobile Robots with Suspended Wheels
Author(s)	鍾, 国梁
Degree Grantor	北海道大学
Degree Name	博士(工学)
Dissertation Number	甲第11127号
Issue Date	2013-09-25
DOI	https://doi.org/10.14943/doctoral.k11127
Doc URL	https://hdl.handle.net/2115/53855
Type	doctoral thesis
File Information	Guoliang_Zhong.pdf

Hokkaido University Collection of Scholarly and Academic Papers : HUSCAP

Motion Control and Stability Improvement of Autonomous Mobile Robots with Suspended Wheels

サスペンションを有する移動ロボットの運動制御および安定性向上に関する研究

Doctoral Dissertation

Guoliang Zhong

Division of Human Mechanical Systems and Design

Graduate School of Engineering

Hokkaido University

September 2013

Contents

Dissertation Abstract IV

Chapter 1 Background and Introduction 1

1.1 Research Background 1

1.2 Contributions and Outline 7

Chapter 2 Vibration Control of Mobile Manipulator 8

2.1 Introduction 8

2.2 PSO and PSO with Chaos 10

2.2.1 PSO Algorithm 10

2.2.2 PSO with Chaos Method 12

2.3 Problem One: Suspended Platform Motion with a Static Manipulator 14

2.3.1 Dynamic Model 14

2.3.2 Controller Design and Optimization 16

2.3.3 Results and Discussion 21

2.4 Problem Two: Suspended Platform Motion with a Dynamic Manipulator 28

2.4.1 DPM Concept 28

2.4.2 Dynamic Model 30

2.4.3 Optimal Trajectory Generation 31

2.4.4 Results and Discussion 34

2.5 Summary 39

Appendix A: Dynamic Model for Problem One 40

Appendix B: Dynamic Model for Problem Two 43

Chapter 3 Tracking Control Considering Interaction Effects and Uncertainties 45

3.1 Introduction 45

3.2 System Model 47

3.2.1 Kinematic Model 48
3.2.2 Dynamic Model 51
3.3 Backstepping Control Based on Kinematics 51
3.3.1 Controller Design 52
3.3.2 Results and Discussion 54
3.4 Adaptive Backstepping Control with Fuzzy Compensator Based on Dynamics 57
3.4.1 Controller Design 57
3.4.2 Results and Discussion 60
3.5 Summary 70

Chapter 4 Dynamic Stability Control of Robots in Rough Terrain 72
4.1 Introduction 72
4.2 Robot with Semi-Active Suspension Systems 74
4.3 Dynamic Stability Measure 77
4.4 Controller Design and Optimization 81
4.4.1 Semi-Active Damper Control Model 81
4.4.2 Optimization Using the CPSO Algorithm 85
4.5 Results and Discussion 89
4.6 Summary 95

Chapter 5 Teleoperation of Robots Based on Virtual Reality and WiFi 96
5.1 Introduction 96
5.2 Robot System Design 98
5.2.1 Mechanical Structure 98
5.2.2 Control System Architecture 100
5.3 Kinematics Analysis 100
5.3.1 Kinematic Model of Platform 100
5.3.2 Kinematic Model of Manipulator 102
5.4 Teleoperation Based on the Virtual Reality Technology via WiFi 104
5.5 Experimental Results 106
5.6 Summary 108

Chapter 6 Conclusions and Further Works 109

6.1 Conclusions 109

6.2 Further Works 111

References 112

Acknowledgment 122

Dissertation Abstract

学 位 論 文 内 容 の 要 旨

博士の専攻分野の名称　博士（工学）　氏名　鍾 国梁 (Zhong Guoliang)

学 位 論 文 題 名

Motion Control and Stability Improvement of Autonomous Mobile Robots with Suspended Wheels

（サスペンションを有する移動ロボットの運動制御および安定性向上に関する研究）

Autonomous mobile manipulators have been given extensive attention in recent years since they have many applications such as transporting operation of materials in modern factories, handling radiation infected objects in specific environments, and serving disabled persons at home. In order to satisfy the mobility performance requirements of a challenging rough terrain, a suspended wheeled mobile robot with a dexterous manipulator becomes an appropriate mobile robot design. This thesis focuses on the suspended wheeled mobile robot and deal with the following problems: Firstly, mobile robots with suspension system can absorb vibration induced by rough roads, but due to center-of-gravity (CG) shift and the dynamic of manipulator, the suspended platform is subject to vibration when the robot moves with acceleration. Secondly, trajectory tracking of a mobile platform and a manipulator simultaneously is a challenging work because of its complex nonlinearity and dynamic interaction between the platform and manipulator. Thirdly, when robots move through rough terrain, it is necessary to improve their stability to eliminate external interference that degrades the performance of vision systems. Finally, although application of the teleoperation has been a benefit to remote control, how to generate an intuitive user interface for teleoperation is still a troublesome problem. In order to solve these problems, this thesis is organized as follows:

The research background about the suspended wheeled robot with a manipulator is introduced in Chapter 1. In addition, the contributions and outline are presented in this chapter.

On the aspect of vibration suppression, Chapter 2 presents approaches based on the particle swarm optimization (PSO) algorithm to overcome the following vibration problems: (1) when the suspended platform moves with static manipulator, vibration of the suspended platform occurs due to CG shift; (2) when the suspended platform and the manipulator move simultaneously, the vibration is caused by the movement of manipulator. For the first problem, a method for the optimization of multi-input shapers is designed by using PSO with chaos to reduce the residual vibration. For the second problem, an approach based on PSO with chaos is developed to suppress the vibration by searching for the time-jerk synthetic optimal trajectories of the manipulator. Then, the resulting shapers and optimal trajectories are performed on the presented models and demonstrate that the vibration can be controlled to a desired level effectively in both problems.

Chapter 3 presents an adaptive control strategy for trajectory tracking of a mobile manipulator system that consists of a wheeled platform and a modular manipulator. When a robot system moves in the presence of sliding, it is difficult to accurately track its trajectory by applying the backstepping approach, even if a non-ideal kinematic model is employed. To address this problem, a combination of adaptive fuzzy control and backstepping approach based on a dynamic model is proposed. This control scheme considers the dynamic interaction between the platform and manipulator. To accurately track the trajectory, a fuzzy compensator is proposed to compensate for modeling uncertainties such as friction and external disturbances. Moreover, to reduce approximation errors and ensure system stability, a robust term is included in the adaptive control law. Simulation results obtained by comparing several cases reveal the presence of the dynamic interaction and confirm the robustness of the designed controller. Then, the effectiveness and merits of the proposed control strategy to counteract the modeling uncertainties and accurately track the trajectory are demonstrated.

In Chapter 4, semi-active suspensions are mounted between the wheels and

platform of a robotic vehicle to absorb the vibrations caused by rough terrain. The semi-active suspension consists of a spring and a magneto rheological damper. By combining the dynamic model of the suspended robotic vehicle and the control model of the damper, a new methodology is proposed to evaluate the dynamic stability of the vehicle. The model considers the configuration of semi-active suspensions and the road-holding ability of robotic vehicles. Based on the stability criterion, the chaotic PSO (CPSO) method is used to search the optimum semi-active damping characteristics. The control model of the semi-active damper is checked by sinusoidal response analysis. To verify the dynamic stability criterion and the control method, the proposed methodology is evaluated by simulating a rough pavement condition and comparing the effectiveness of the method to a passive suspension. The results show that the proposed stability criterion is feasible, and the optimal control method yields a substantially improved dynamic stability when the vehicle moves through rough terrain.

Chapter 5 reports the construction and the remote control of a nonholonomic robotic system. The system consists of a three-wheeled mobile platform to keep the robot stable and a 6-degree-of-freedom (DOF) articulate manipulator on top of the platform. First, how its design and work are explained, then the kinematic model of the robotic platform is presented considering nonholonomic constraints and the inverse kinematics of the manipulator is described. In order to generate a visualized user interface, a new development in information technology, which is known as virtual reality, is adopted. All different movements and manual control of the manipulator are shown in a 3-D virtual environment. Teleoperation is realized by using WiFi to transmit the commands and remote desktop from the lower computer to the upper computer. Then, experiments are conducted to prove that the proposed remote operation of the robotic system is reliable and efficient.

Finally, the conclusions of the whole study are presented and the further works are described in Chapter 6.

Chapter 1 Background and Introduction

1.1 Research Background

Mobile manipulators have been given extensive attention in recent years since they have many applications such as materials transport and service for disabled persons. A mobile modular manipulator is normally composed of a *m*-wheeled mobile platform and a *n*-degree-of-freedom (DOF) onboard modular manipulator. This combination extends the workspace of the entire robot dramatically. Building up the dynamic model for such kind of robots is a challenging task due to the interactive motions between the manipulator and the mobile platform [1, 2]. Also the vibration control is an important research topic in mobile robots, especially when robots move through rough terrain [3]. As for a mobile manipulator, the trajectory tracking task becomes even more complex and difficult to achieve since the platform and the manipulator move simultaneously [4, 5]. Furthermore, stability is another concerning issue since the probability of tip-over increases due to this kind of mechanical structure [6].

To ensure steady movement of robots on rough terrain, usually suspension systems are installed between the wheels and the platform to absorb the vibration induced by road. Typically, Fig. 1-1 shows a suspended wheeled robot called Seekur, which is made by Adept MobileRobots LLC [7]. Seekur is a large, all-weather robot that can traverse rugged terrain. Its four independently controlled wheels allow the platform omnidirectional steering capability. Further, as shown in Fig. 1-2, if a manipulator is mounted on the platform, it will be a suspended mobile manipulator. The suspended mobile manipulator can achieve tasks in the field so that more and more researchers have been devoted to the researches involving this kind of robot. This study mainly investigates the vibration reduction, tracking control, stability improvement, and teleoperation of the suspended mobile manipulator.

Fig. 1-1 Seekur robot.

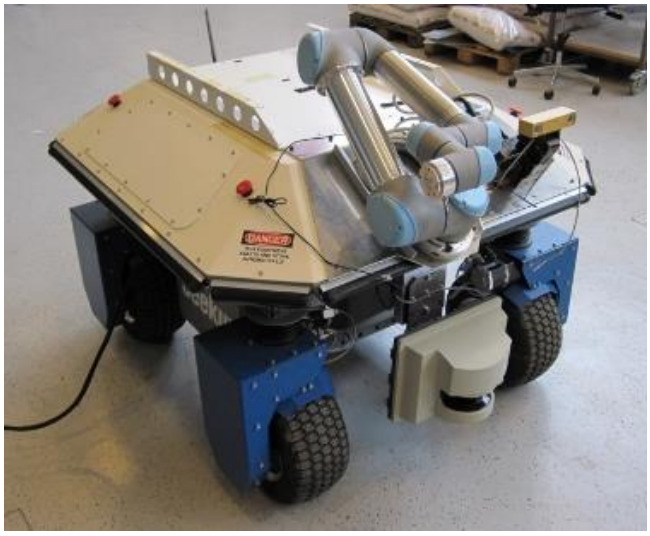


Fig. 1-2 Seekur mobile manipulator.

For modeling the robot system, the interaction between the platform and the manipulator results in more complicated dynamics equations, which in turn requires a more sophisticated control. Obviously, using a suspension system with the mobile platform will add the complexity of the platform-manipulator interaction. However, on the other hand, a system will be gotten with more flexibility and better performance even in undesirable situations such as moving on uneven environments. To apply model-based control laws, it is required to extract explicit system dynamics model. Saha and Angeles [8] obtained a systematic method for the kinematics and dynamics modeling of a two DOFs automated guided vehicle. They employed the

notion of natural orthogonal complement to eliminate the Lagrange multipliers. In [9] the dynamic interaction between a 1-DOF manipulator and platform of the mobile manipulator was investigated for a planar robotic system. In [10] direct path method (DPM) was utilized for deriving the dynamics of a space robotic system equipped with multiple arms which was proved to have noticeably less time-consuming mathematical calculations as compared to other criteria. To obtain a concise explicit set of dynamics equations, this useful method is extended in this study for including the terms induced by the interaction between platform and manipulator.

Similarly, the vibration isolation performance of a vehicle suspension system has been of paramount importance in the automotive industries. Accordingly, there are many studies about how to improve the riding comfort by reducing the vibration from road. However, unlike vehicles, the position of center-of-gravity (CG) for a suspended mobile manipulator is variable when the manipulator is moving to perform tasks. Therefore, it is necessary to suppress the vibration caused by the CG shift especially in the case of a long manipulator with a small-sized platform. Many related studies concerned with reduction of the vibration caused by rough road and prevention of tip-over have been done by other researchers [11, 12]. In particular, Korayem et al. [13] proposed a novel approach for the determination of the maximum payload-carrying capacity of a mobile manipulator considering the tip-over stability on zero moment point (ZMP) criterion. Zhong et al. [14] presented a method for the optimization of multi-input shapers using the particle swarm optimization (PSO) algorithm [15] to suppress the vibration of the suspended mobile robot. However, for suspended mobile robot, there is a dearth of research in previous literatures addressing the suppression of vibration resulted from the CG shift which must be taken into account in dynamic control. In this study, a feed-forward control is presented based on optimal input shaping technique to reduce the vibration.

In robotics, trajectory generation is a fundamental problem. The problem can be defined in this way: find a temporal motion law along a given geometric path, in such a way that certain requirements on the trajectory properties are fulfilled. Generating trajectories which satisfy specific requirements concerning position, velocity, acceleration, and jerk values is crucial to ensure optimal results from the viewpoint of

motion performance, especially for high-speed operations required in many current applications. A smooth trajectory (i.e., a trajectory with a limited jerk value) can reduce the vibration during the movement. Low-jerk trajectories can be executed more rapidly and accurately. Moreover, the jerk minimization can limit the excitation of resonance frequencies, since vibration induced by non-smooth trajectories can damage the robot actuators, and introduce large errors while the robot is performing tasks such as a trajectory tracking. The aim of the method is to minimize the value of the jerk during the execution by generating a jerk-optimal trajectory, which can reduce the vibration and save the execution time. Some examples of this approach can be found in [16–18]. In [16] a computationally efficient nonlinear trajectory generation (NTG) algorithm was developed and its software implementation was described to solve, in real-time, nonlinear optimal trajectory generation problems for constrained systems. NTG is a nonlinear trajectory generation software package that combines nonlinear control theory, B-spline basis functions, and nonlinear programming. In [17] the so-called interval analysis was used in an algorithm that globally minimizes the maximum absolute value of the jerk along a trajectory whose execution time is set a priori. The trajectories were expressed by cubic splines and the intervals between the via-points were computed so that the lowest jerk peak was produced. In [18] a time-jerk optimal trajectory planning algorithm is presented. This method was an extended version of the algorithm reported in [19, 20], because into the optimization constraints, the joint variable limits and the instantaneous power consumption of the actuating motors were also considered.

In the tracking problem, the robot is to follow a desired trajectory. Using kinematic models of mobile robots, various approaches such as backstepping [21], neural networks [5], neural fuzzy [22], linearization [23] have been proposed to solve the tracking problem. However, the dynamic interaction effect was neglected. Moreover, a mobile robot is an uncertain multiple-input multiple-output (MIMO) nonlinear system, which suffers from the model uncertainties and external disturbances. Recently, some approaches have been presented to integrate a kinematic controller and a torque controller for mobile robots with uncertainties [24, 25]. The adaptive control methods, which linearly parameterize the dynamic models of mobile

robots, have been employed to solve the tracking problem of mobile robots. For example, Fukao et al. [26] developed an approach combined an adaptive kinematic controller in consideration of unknown parameters in the robot kinematics and an adaptive torque controller. But it is noted that external disturbances in the robot dynamics were neglected, which may seriously degrade the performance of the tracking control. Moreover, in the existing literatures many of them studied either the trajectory tracking of platform or the trajectory tracking of manipulator, few of them considered the tracking control problem of the platform and the manipulator simultaneously.

For improving the stability of the robot when it moves through rough terrain, semi-active suspensions are employed in this study. The advantages of semi-active suspensions over traditional passive suspensions have been addressed in many studies. For example, Karnopp et al. [27] studied the performance of a skyhook controlled semi-active suspension, and compared the performance with that of a conventional passive suspension. The name “skyhook” is derived from the fact that it is a passive damper hooked to an imaginary inertial reference point. Semi-active control strategies can maintain the reliability of passive devices using a very small amount of energy, yet provide the versatility, adaptability and higher performance of fully active systems [28]. The particular benefits of semi-active methods are that (as with adaptive-passive methods) the parameters of the system can be changed with time to retain optimal performance and (unlike adaptive-passive methods) higher levels of optimization can be achieved due to the rapid time-variation capability.

To improve the stability of a semi-active suspended mobile manipulator, an adequate stability measure is a premise. Several researchers examined the problem of how one should define the instantaneous stability margin for a mobile manipulator. McGhee and Iswandhi [29] proposed the use of the shortest horizontal distance between the CG and the support pattern boundary projected onto a horizontal plane. This measure was refined by Song and later by Sugano et al. [30], yet it was insensitive to top-heaviness and was only an approximation for systems on uneven terrain. Furthermore, this measure failed in the presence of angular loads. Davidson and Schweitzer [31] also extended the work of McGhee by using screw mechanics to

provide a measure which eliminated the need for a projection plane while allowing for angular loads. But they recognized that their measure was not sensitive to top-heaviness. Korayem et al. [32] proposed a measure which was sensitive to CG height by using the minimum work required to tip-over the vehicle. Their energy-based approach was extended by themselves to include inertial and external loads [33]. However, it was subjected to the same assumption of constant load magnitude and direction throughout the motion.

Teleoperation is the manipulation of an object (in this case a robot), in such a way as to allow an operator to perform a task at a distance. This is usually done because of a hostile environment where human access is difficult, but human intelligence is necessary. Teleoperation system mainly consists two parts, one is the user interface and the other is transmission mode. Many studies concerning teleoperation of robots have done in recent. A robotic system for fetching objects in home environment was developed by Cristancho et al. [34], where fetching task was implemented using multimodal user interface with touch and voice inputs and intelligent adaptive manipulation strategy, based on image processing of images from on-board cameras. Livatino et al. [35] discussed the stereo viewing and virtual reality technologies in mobile robot teleguide. In these studies, the virtual reality was used to develop user interface. For telecommands transmission mode, these are many ways. Reference [36] adopted a wireless application protocol (WAP) based system for mobile robot teleoperation. WAP has integrated Internet and mobile phone technologies and it can transmit information around the globe. WAP runs on GPRS or CDMA, but these two communication networks can not support enough bandwidth to transmit videos smoothly nowadays. Comparing with WAP, wireless WiFi has enough bandwidth to transmit data. Kelly et al. [37] developed a real-time photorealistic virtualized reality interface for remote mobile robot control by using virtual reality and wireless techniques. This study establishes a teleoperation system of mobile robots based on virtual reality and WiFi. Further, the upper computer is used to deal with algorithms, decisions, and commands so that the errors and time delay can be reduced during communication.

1.2 Contributions and Outline

The main focus of this study is motion control and stability improvement of suspended mobile robots. On the basis of the research background presented above, the thesis spreads out discussion mainly from the following six chapters:

Chapter 1 presents the research background about the suspended mobile robot. The contributions are listed with the outline of this thesis.

Chapter 2 endeavors to present methods by using PSO algorithm for solving the following vibration problems: (1) Using optimal multi-input shaping technique to reduce the vibration of suspended platform occurs due to the CG shift; (2) Suppressing the vibration caused by dynamic manipulator by generating the time-jerk synthetic optimal trajectory.

Chapter 3 proposes an adaptive control method, which is designed by applying backstepping idea, to track the trajectory of mobile manipulator in presence of the interaction and uncertainty. By using this method, the mobile platform and manipulator can accurately track the desired trajectories simultaneously.

Chapter 4 investigates the stability of the suspended wheeled robot by installing semi-active suspensions between the platform and wheels. A top-heaviness sensitive stability measure is proposed and a modified model of semi-active suspension is presented. Proposed stability criterion and control method is feasible to substantially improve the dynamic stability when robots move through rough terrain.

Chapter 5 develops a novel teleoperation technique by using the virtual reality and wireless WiFi. The proposed remote control method generates an intuitive user interface for teleoperation. Also experiments prove that the proposed remote operation of the robotic system is reliable and efficient.

Chapter 6 concludes the whole thesis and presents the further works based on this study.

Chapter 2 Vibration Control of Mobile Manipulator

2.1 Introduction

A mobile robot with a manipulator has attracted extensive attention for its wide area of applications and dexterity. Suspended mobile robot generally consists of a suspended mobile platform and a modular manipulator, and this combination extends the workspace of the robot dramatically, furthermore, the suspension system can absorb vibration induced by rough roads. But in the case of a small-sized platform with suspension system, severe vibration and even the dangers of tip-over on account of center-of-gravity (CG) shift may occur. This chapter proposes a method to reduce this type of vibration by using optimal multi-input shaping.

Input command shaping is an attractive feed-forward control technique, which is essentially a “hands off” vibration reduction method. With input command shaping, inputs can be fed through shapers and into the system, and ideally the resulting system output will be vibration free. Yin et al. [38] designed a controller involving the input shaping technique to suppress the residual vibration of a flexible manipulator. Gurleyuk and Cinal [39] presented an unsophisticated method for tuning the amplitudes and time locations of a three-impulse sequence input shapers such as zero vibration (ZV), zero vibration derivative (ZVD) and extra insensitive (EI) or specified insensitive (SI) shapers. In this chapter, the particle swarm optimization (PSO) with chaos [40] is used to obtain the optimal multi-input ZVD shapers.

When the manipulator moves to accomplish some tasks while the platform is moving, the suspension system causes vibration which results from the dynamic manipulator. There have been numerous similar studies which can be divided into two groups depending on the problem-solving method. One of them uses the time optimal control with limited acceleration [41–43], the other is the method to search a

minimum-jerk trajectory [17, 44, 45]. However, in the reports described above, they ignored the interaction between the platform and manipulator, and only considered either the dynamics of platform or of manipulator.

In this chapter, the main configuration of the robot is a three-link manipulator mounted on a mobile suspended platform. Two rear driving wheels and one front caster wheel support the platform, as shown in Fig. 2-1. The multi-input shaping technique is applied to reduce the vibration of a mobile robot when the manipulator is static. Considering the CG shift, the input shapers evaluating both robustness and settling time are optimized by PSO with chaos. The introduced shapers have the vibration reduction capabilities of the original shapers, and exhibit improvement in robustness and settling time. When the manipulator and suspended platform move simultaneously, the direct path method (DPM) [10] is introduced to model the system considering the dynamics of the manipulator and the interaction between the platform and manipulator. Based on cubic splines, the time-jerk synthetic optimal trajectory is described mathematically by taking account of both the minimum execution time and minimax approach of jerks with kinematics constraints such as the upper bounds of velocity and acceleration. The optimal trajectory of each joint is obtained by using PSO with chaos. The residual vibration can be reduced by forcing the manipulator along the generated optimal trajectory.

Fig. 2-1 Picture of suspended mobile robot with a manipulator.

The remainder of this chapter is organized as follows: Section 2.2 introduces PSO algorithm and PSO with chaos. Problems one and two are presented in Section 2.3 and 2.4, respectively, through formulating the mathematical models, designing the optimization schemes, and discussing about the results. Finally, conclusions are provided in Section 2.5.

2.2 PSO and PSO with Chaos

2.2.1 PSO Algorithm

PSO is a stochastic optimization technique inspired by social behavior of bird flocking or fish schooling [15]. Unlike genetic algorithm (GA), PSO has no evolution operators such as crossover and mutation, which makes it simpler, easily completed and fewer parameters needed. In response, PSO has been given extensive attention in vibration control field [46–48].

The implementation of the PSO based on a state machine (SM) is outlined in Fig. 2-2. The arrow leading from one state to another is called a transition, and describes how the SM transits from a state to another state. The label for the transition describes the condition that triggers the transition. The process for implementing the global version of the PSO is as follows:

Step 1: Initialize a population (array) of particles with random positions and velocities in *d* dimensions in the problem space.

Step 2: For each particle, evaluate the desired optimization fitness function in *j* variables.

Step 3: Compare the particle fitness evaluation of each particle with its *local_best*. If current value is better than the *local_best*, set the *local_best* value to the current value and the *local_best* location equal to the current location in the *j*-dimensional space.

Step 4: Compare the fitness evaluation with the overall previous best of the population. If the current value is better than the *global_best*, reset the *global_best* to

the current array index and value of the particle.

Step 5: Change the velocity and the position of the particle according to Eqs. (2-1) and (2-2), respectively.

$$v_{ij}^{k+1} = \sigma \cdot v_{i,j}^{k} + \chi_1 \cdot rand_1 \cdot \left(p_{i,j}^{k} - x_{i,j}^{k}\right) + \chi_2 \cdot rand_2 \cdot \left(g_{i,j}^{k} - x_{i,j}^{k}\right), \tag{2-1}$$

$$x_{i,j}^{k+1} = x_{i,j}^{k} + v_{i,j}^{k}, \tag{2-2}$$

where $v_{i,j}^{k}$ and $x_{i,j}^{k}$ are the velocity and position of the *j*th dimension of the *i*th particle in the *k*th iteration. Learning rates are represented by χ_1 and χ_2; $rand_1$ and $rand_2$ are two pseudo-random values in the range of [0, 1]; $p_{i,j}^{k}$ and $g_{i,j}^{k}$ are the current personal best position and global best position respectively; σ is the inertial weighting which determines how much the particle inherits from the former velocity.

Step 6: Loop back to step 2 until a set criterion, which is usually a sufficiently good fit or a maximum number of iteration, is reached.

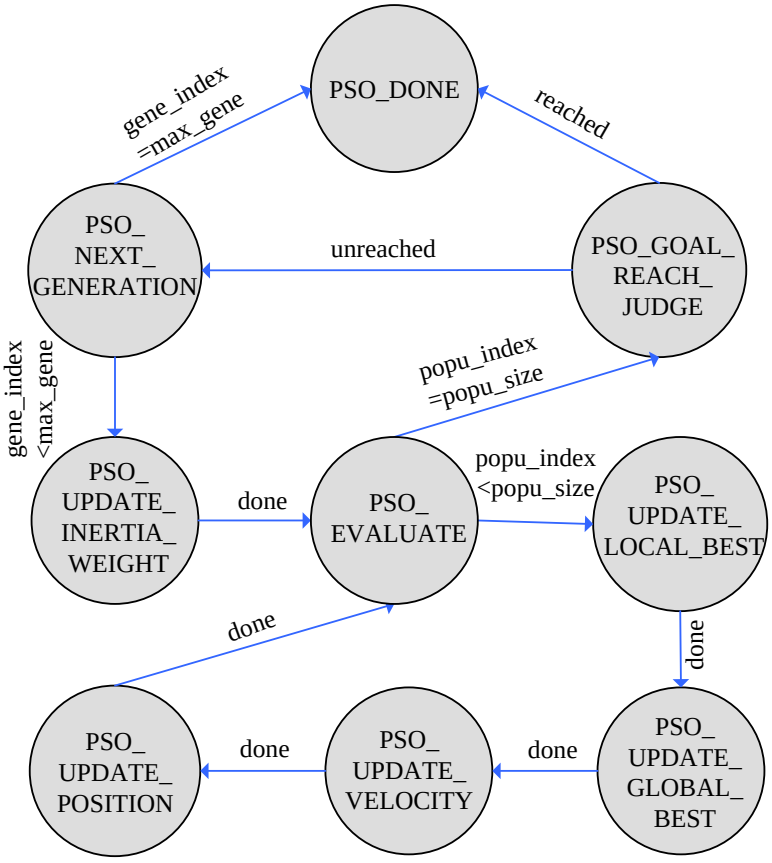


Fig. 2-2 Implementation of PSO based on a state machine.

2.2.2 PSO with Chaos Method

In order to avoid the local minimum, the chaotic algorithm is introduced into the evolutionary process. Generally, chaos optimization process can be divided into two steps: the initiation and the perturbation. In the initiation stage, chaos variable instead of random number is used. Chaos variable is given as

$$\xi_{i,j}^{1} = rand\,, \tag{2-3}$$

$$\xi_{i+1,j}^{1} = \rho \xi_{i,j}^{1}\left(1 - \xi_{i,j}^{1}\right), \tag{2-4}$$

where *rand* is a pseudo-random value in the range of [0, 1]. The sequence number of chaos variable is *i*. The dimension of the particles is *j*. Chaos variable $\xi_{i,j}$ is in [0, 1] interval. The rate for reproduction ρ is set to $\rho = 4$.

In the perturbation stage, chaos disturbance *r* to the particles is added, and then the cost values are calculated. Solutions at the initial stage should have a big variation and explore the search space in large-scale, then every solution gradually approaches to optimal point with increase in iterations, therefore, a smaller variation is needed so that current solution can search in a small range. The disturbance *r* can be defined by

$$r_{i,j}^{1} = -rand + \chi_0 \xi_{i,j}^{1} rand\,, \tag{2-5}$$

$$\xi_{i,j}^{k+1} = \rho \xi_{i,j}^{k}\left(1 - \xi_{i,j}^{k}\right), \tag{2-6}$$

where χ_0 is a learning factor which is set to 2. The current iteration number is *k*. When $k = 1$, $\xi_{i,j}^{k}$ can be gotten from Eq. (2-4).

Figure 2-3 shows the logistic map of chaotic algorithm. Figure 2-4 shows the flow chart of PSO with chaos method.

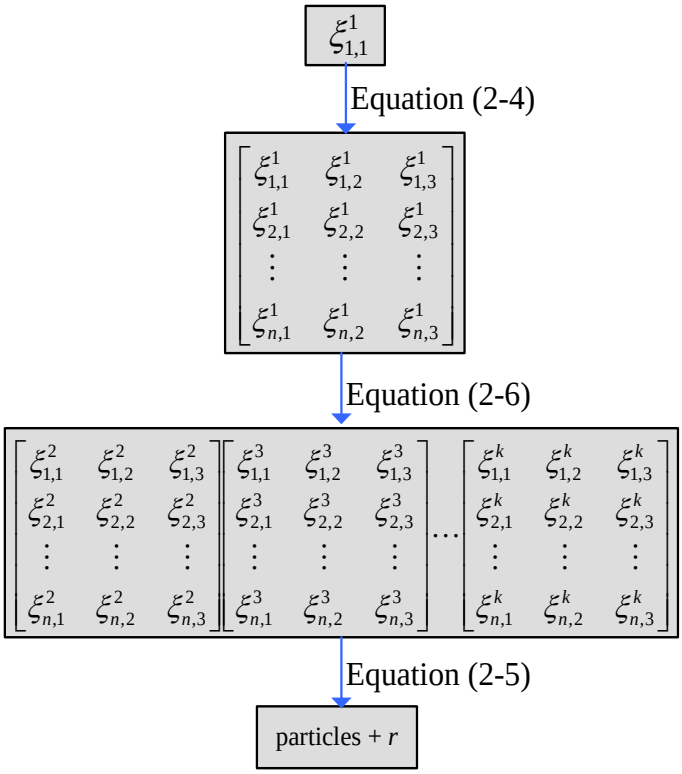


Fig. 2-3 Logic diagram of mathematical operation.

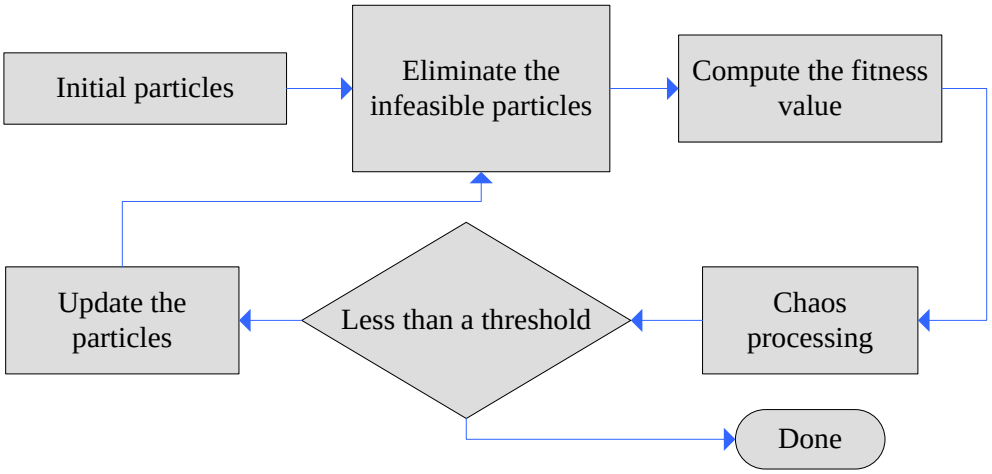


Fig. 2-4 Flow chart of PSO with chaos method.

2.3 Problem One: Suspended Platform Motion with a Static Manipulator

2.3.1 Dynamic Model

The robot with an upright static manipulator is illustrated in Fig. 2-5. The CG is shifted because of the pose of the manipulator. The suspension systems are installed between the rear wheels and platform. The O-XY coordinate system is centered on the mobile platform, the line through centers of the two rear wheels is the X axis, the line perpendicular to the X axis and through the center of the front wheel is the Y axis.

To express the dynamics of the robot, determining the velocity of the CG is necessary. As illustrated in Fig. 2-5, the velocity of the CG is decomposed into the X and Y directions

$$V_{Gx} = -\frac{l_r}{w}\left(v_R - v_L\right), \tag{2-7}$$

$$V_{Gy} = \frac{1}{2}\left(v_L + v_R\right) + \frac{e}{w}\left(v_R - v_L\right), \tag{2-8}$$

where V_{Gx} and V_{Gy} are the velocity components in the X and Y directions, l_r is the CG shift relative to the X axis, e is the CG shift relative to the Y axis, w is the distance between centers of the two rear wheels, v_L and v_R are the linear velocities of the left and right wheels. In the case of the static manipulator, l_r and e are constant values.

From the Lagrange equation of the second type, the dynamics equation of the system can be defined by

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right) - \frac{\partial T}{\partial q_i} + \frac{\partial U}{\partial q_i} + \frac{\partial U_F}{\partial \dot{q}_i} = Q_i, \qquad i = 1, \cdots, N, \tag{2-9}$$

where T, U and U_F are the robotic system's kinetic, potential and dissipated energies, respectively. Here, N describes the degree-of-freedom (DOF) of the system. And q_i, $\dot{q}_i$ and Q_i are the ith element of the vector of the generalized coordinates, speeds, and forces, respectively.

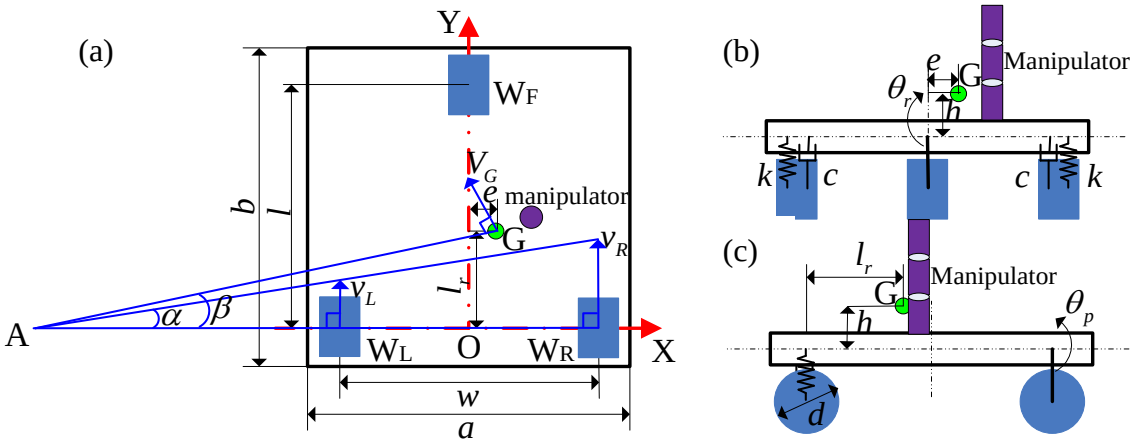


Fig. 2-5 Schematic drawings of suspended mobile robot with a manipulator.

(a) Top view, (b) Rear view, and (c) Side view.

The generalized coordinate is defined by

$$\boldsymbol{q} = \{\varphi_L, \varphi_R, \theta_p, \theta_r\}^T, \tag{2-10}$$

where φ_L and φ_R describe the rotational angles of the left and right wheels, θ_p is the pitch angle of the platform, and θ_r is the roll angle of the platform.

The total kinetic energy of the system can be expressed as

$$T = \frac{1}{2}\left[M\left(V_{Gx}{}^2 + V_{Gy}{}^2\right) + I_p\dot{\theta}_p{}^2 + I_r\dot{\theta}_r{}^2 + I_w\dot{\varphi}_L{}^2 + I_w\dot{\varphi}_R{}^2\right], \tag{2-11}$$

where M is the mass of the robot, I_p and I_r are the total moments of inertia with respect to the pitch and roll axes, and I_w is the moment of inertia of a wheel with respect to the center of the wheel.

The total potential energy includes both gravitational energy and the potential energy stored in the springs and can be expressed as

$$U = \frac{1}{2}k\left[\left(\boldsymbol{\delta}_L \cdot \boldsymbol{\delta}_L + \boldsymbol{\delta}_R \cdot \boldsymbol{\delta}_R\right) + Mg \cdot \boldsymbol{\delta}_G\right], \tag{2-12}$$

where k is the stiffness coefficient for the springs, g is the acceleration of gravity, $\boldsymbol{\delta}_L$, $\boldsymbol{\delta}_R$ and $\boldsymbol{\delta}_G$ are the corresponding displacements for left wheel spring, right wheel spring, and CG, respectively.

Next, the total dissipated energy of the system is obtained as

$$U_F = \frac{1}{2}c\left(\dot{\boldsymbol{\delta}}_L \cdot \dot{\boldsymbol{\delta}}_L + \dot{\boldsymbol{\delta}}_R \cdot \dot{\boldsymbol{\delta}}_R\right) + \mu N_L \int_0^t \left(\frac{1}{2} d\dot{\varphi}_L\right) dt + \mu N_R \int_0^t \left(\frac{1}{2} d\dot{\varphi}_R\right) dt, \tag{2-13}$$

where c is the viscous damping coefficient, N_L and N_R are the normal pressures on the left and right wheels, μ is the friction coefficient, $\dot{\boldsymbol{\delta}}_L$ and $\dot{\boldsymbol{\delta}}_R$ are the rates of the corresponding left and right displacements for the suspension mounted on each wheel.

Substituting the system kinetic, potential, and dissipated energies in Eq. (2-9), the dynamic model can be obtained in the state space as

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \\ \boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} \end{cases}, \tag{2-14}$$

where $\boldsymbol{x} = \left[\dot{\theta}_p, \dot{\theta}_r, \theta_p, \theta_r\right]^T$ is the state vector, $\boldsymbol{y} = \left[\theta_p, \theta_r\right]^T$ is the output vector, $\boldsymbol{u} = \left[\tau_L, \tau_R\right]^T$ (τ_L, τ_R are the torques of the left and right drive motors) is the control force vector, $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{C}$ are the system, control and output matrices, respectively. It is assumed that θ_p, θ_r are small, i.e., $\sin\theta_p \approx \theta_p$, $\sin\theta_r \approx \theta_r$, $\cos\theta_p \approx 1$ and $\cos\theta_r \approx 1$. Detailed derivation of the state equation is described in Appendix A.

2.3.2 Controller Design and Optimization

The introduced control strategy consists of a feed-forward control unit and a feedback control unit, and the two units are designed separately, as shown in Fig. 2-6.

The feed-forward control unit involves an optimal multi-input shaping technique. It is implemented outside of the feedback loop and is used to suppress induced vibration of the reference model at the starting and the stop stages of the robot motion. The motion between the two stages is controlled by a feedback control unit making error compensation. This subsection focuses on the starting maneuvers from some prescribed input state to the achievement of uniform rectilinear motion, where a dynamic balance has been reached between a specific coulomb friction and input torques. A feed-forward controller is developed to reduce the vibration of the suspension system by using optimal ZVD shapers which are optimized by employing the PSO with chaos method.

Each of the ZVD shapers functions with three sets of impulses (i.e. six variables including three timings and three amplitudes). Then, as the first amplitude timing is at

zero and the three amplitudes are constrained to add up to unity, the number of variables for each shaper is reduced to four. Two shapers act on the left and right input torques for the robot considered here, and the optimization of multi-input shapers is transformed into an optimization problem of eight variables which are expressed as

$$E = \left[A_{L1}, A_{L2}, A_{R1}, A_{R2}, t_{L2}, t_{L3}, t_{R2}, t_{R3} \right]^T, \tag{2-15}$$

where A_{L1}, A_{L2} and A_{R1}, A_{R2} are the impulse amplitudes for the left and right drive wheels, t_{L2}, t_{L3} and t_{R2}, t_{R3} are the impulse timings for left and right drive wheels.

Considering the energy of the vibration during motion, the robustness with respect to frequency, the settling time of the system, and the saturation limits of the input torques, the cost function is defined as

$$J = \frac{1}{2} \sum_{i=1}^{3} k_i \int_{t_0}^{t_f} \left[\boldsymbol{x}(t) \boldsymbol{K} \boldsymbol{x}^T(t) + \boldsymbol{u}(t) \boldsymbol{H} \boldsymbol{u}^T(t) \right] dt \Big|_{\omega_o = c_i \omega_d} + k_4 t_s, \tag{2-16}$$

where ω_o is the operating frequency, ω_d is the design frequency, c_i is a factor that determines the permissible error range of the frequency, and k_i is the weighting factor for each operating frequency. Matrix $\boldsymbol{K}$ is the positive semi-definite weighting matrix and $\boldsymbol{H}$ is the positive definite weighting matrix. The settling time is represented as t_s, and k_4 is the weighting factor for the settling time. Since the error in the damping affects the vibration much smaller than the error in frequency does [49], the above cost function does not consider the robustness with respect to damping.

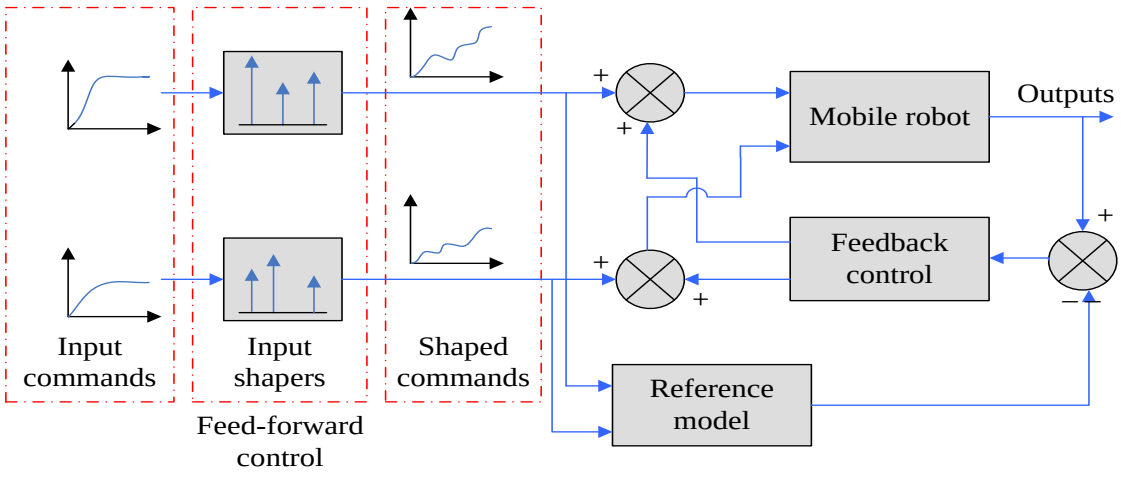


Fig. 2-6 Control scheme of the suspended mobile robot with a static manipulator.

From the above, the optimization of the multi-input shapers can be viewed as a particle evolution in an eight-dimensional space with respect to the cost function. The search space can be defined by

$$\boldsymbol{S}=\begin{bmatrix} A_{L1}^{1} & A_{L2}^{1} & A_{R1}^{1} & A_{R2}^{1} & t_{L2}^{1} & t_{L3}^{1} & t_{R2}^{1} & t_{R3}^{1} \\ A_{L1}^{2} & A_{L2}^{2} & A_{R1}^{2} & A_{R2}^{2} & t_{L2}^{2} & t_{L3}^{2} & t_{R2}^{2} & t_{R3}^{2} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ A_{L1}^{s} & A_{L2}^{s} & A_{R2}^{s} & A_{R3}^{s} & t_{L2}^{s} & t_{L3}^{s} & t_{R2}^{s} & t_{R3}^{s} \end{bmatrix}^{T}, \tag{2-17}$$

where s is the number of particles. The solution to the problem corresponds to the particle in the search space.

Table 2-1 Parameters of the suspended mobile robot and manipulator.

Symbol	Parameter	Value
M	mass of the robot	25.30 kg
M_w	mass of each wheel	0.52 kg
M_b	mass of the platform	19.72 kg
${M_m}^n$	mass of each link of the manipulator	1.86 kg
g	acceleration of gravity	9.81m/s^2
$2w$	distance between the two rear wheels	0.46 m
l	length of the wheelbase	0.46 m
d	diameter of each wheel	0.13 m
e	CG shift relative to X axis	0.10 m
l_r	CG shift relative to Y axis	0.25 m
h	relative height of the CG	0.37 m
${l_m}^n$	length of each link of the manipulator	0.20 m
I_b	moment of inertia of the platform	0.37 kg·m^2
${I_m}^n$	moment of inertia of each link	0.06 kg·m^2
I_p	moment of inertia of the robot with respect to pitch axis	0.26 kg·m^2
I_r	moment of inertia of the robot with respect to roll axis	0.18 kg·m^2
k	stiffness coefficient of each suspension	2.50 kN/m
c	damping property of each suspension	5.00 N/m/s

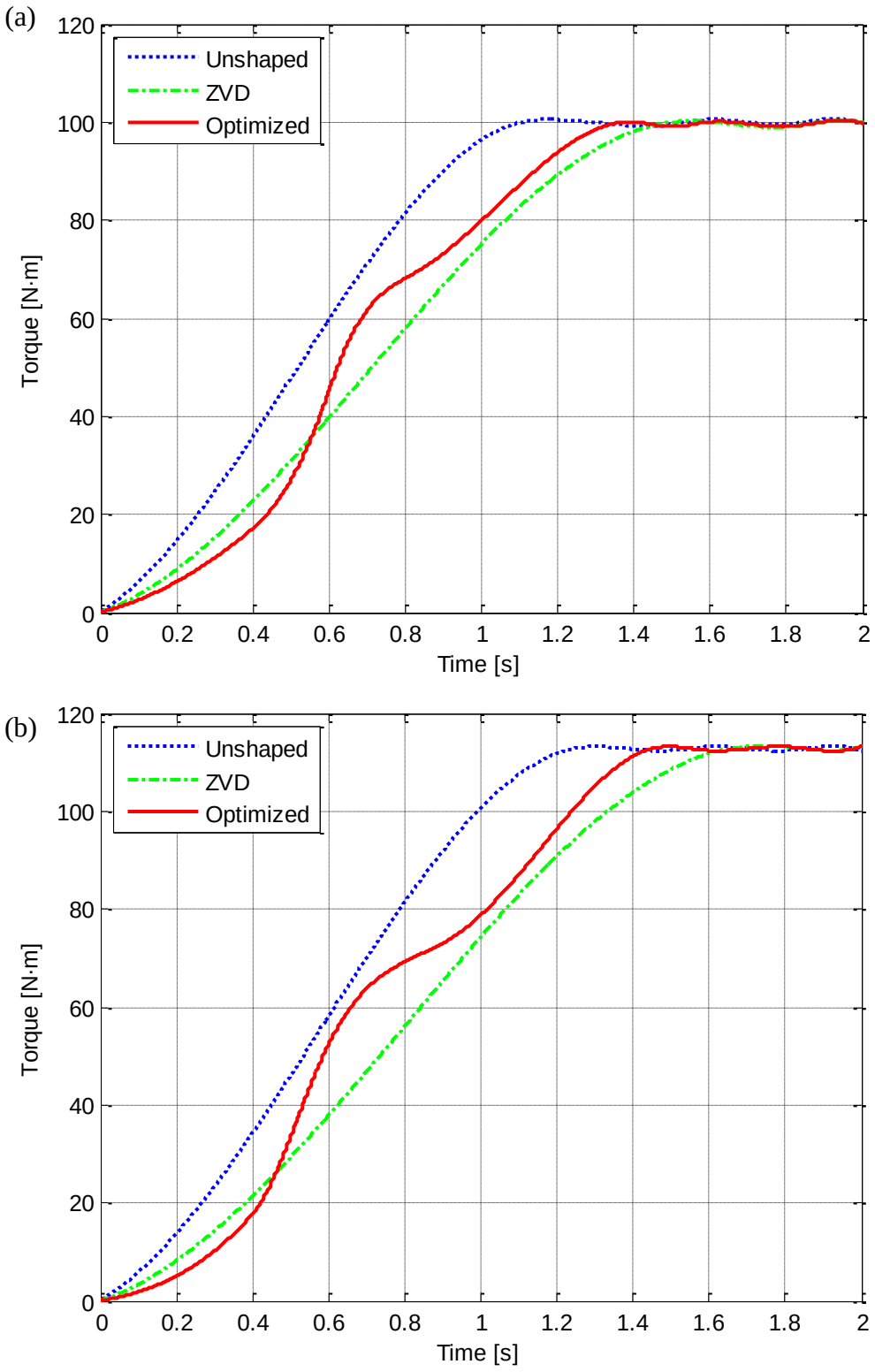


Fig. 2-7 Time history of control input torques.

(a) Input torque of left wheel, and (b) Input torque of right wheel.

Weighting factors in the cost function are set as $k_1 = 10$, $k_2 = 5$, $k_3 = 5$, $k_4 = 3$, $c_1 = 1$, $c_2 = 0.75$, $c_3 = 1.25$, $\boldsymbol{K}$ = diag[1 1 10 10], and $\boldsymbol{H}$ = diag[1 1]. Parameters of the abovementioned model are listed in Table 2-1.

According to the values of the model parameters and the state equation, the operating frequencies in the pitch and roll directions can be obtained from the eigenvalues of the system matrix $\boldsymbol{A}$ as follows:

$$\omega_o^p = Eig_1(\boldsymbol{A}) = 15.49 \text{ rad/s}, \tag{2-18}$$

$$\omega_o^r = Eig_2(\boldsymbol{A}) = 8.14 \text{ rad/s}. \tag{2-19}$$

As the vibration in the pitch direction is larger than that in the roll direction during rectilinear motion, and thus ω_o^p is considered as the design frequency ω_d and ω_o^r is ignored in the design and optimization of the ZVD shapers. When the ZVD shapers are designed by traditional method [50], the variables for the left and right input shapers are $A_{L1} = A_{R1} = 0.34$, $A_{L2} = A_{R2} = 0.49$, $T_{L2} = T_{R2} = 0.58$, $T_{L3} = T_{R3} = 1.16$. While the optimal variables obtained by using PSO after 100 iterations are $A_{L1} = 0.21$, $A_{L2} = 0.65$, $T_{L2} = 0.74$, $T_{L3} = 1.12$, $A_{R1} = 0.16$, $A_{R2} = 0.69$, $T_{R2} = 0.63$, $T_{R3} = 1.20$.

Figure 2-7 shows the time histories of the left and right input torques under the condition of a specific coulomb friction. Comparing the optimal input torques with the original input torques and the input torques after ZVD shaping, though the reduction in the vibration amplitude comes at a trade-off with the long settling time, the value of the cost function for the optimal shaping obtained by PSO with chaos is lower than that for the ZVD shaping.

In addition, to keep the robot move along a straight line, first the input torque for the left wheel is defined as a step response of 100 N m, then the input torque for the right wheel is obtained by trial and error. From the model in Fig. 2-5, the CG is in the geometrical right half part of the robot, and then the right wheel carries more weight of the robot than the left wheel does. Consequently, as seen in Fig. 2-7, when the robot moves on a straight line, the input torque for right wheel is larger than that for the left wheel in the final steady-state.

2.3.3 Results and Discussion

To demonstrate the effectiveness of the proposed method, several numerical simulations are performed. The input torques as shown in Fig. 2-7 are employed as the control input. Figure 2-8 shows the comparison among the unshaped, ZVD shaped, and optimal multi-input shaped responses for the pitch and roll angles.

When $\omega_o = \omega_o^p = \omega_d$, as indicated by the curves in Fig. 2-8(a), both the ZVD and optimal shapers yield lower residual vibration levels than the unshaped response in the pitch direction. Further, the optimal shapers with improved settling time result in a better performance than the ZVD shapers. Fig. 2-8(b) indicates that the ZVD shapers and optimal shapers do not give rise to significant vibration in the roll direction. It is because of this insignificant impact that the vibration frequency in the pitch direction only is necessary to design shapers in Subsection 2.3.2.

Approaches to develop robust shapers loosely fall into four categories: derivative methods, tolerable vibration limit methods, ad hoc methods, and numerical optimization methods. In this study, the numerical optimization is used to improve the robustness. Next, in order to examine the robustness, the design frequency is fixed as $\omega_d = 15.49$ rad/s, and change the operating frequency ω_o through changing the relative height of CG.

Figure 2-9 shows the vibratory responses in pitch direction with the frequency $\omega_o = 0.78\omega_d = 12.08$ rad/s and $\omega_o = 1.34\omega_d = 20.76$ rad/s. To allow comparison, the performance of the unshaped command is superimposed on the curves. From Fig. 2-9(a), it is observed that the optimal input shapers with the improved robustness give a better performance than the ZVD shapers when $\omega_o = 0.78\omega_d$. This advantage results from the considerations of robustness with respect to frequency in the cost function. However, when the operating frequency $\omega_o \notin [0.75\omega_d,\ 1.25\omega_d]$, as shown in Fig. 2-9(b), the optimal shaped command is better than the unshaped command but not superior to the ZVD shaping since the error range considered in the cost function is $[0.75\omega_d,\ 1.25\omega_d]$.

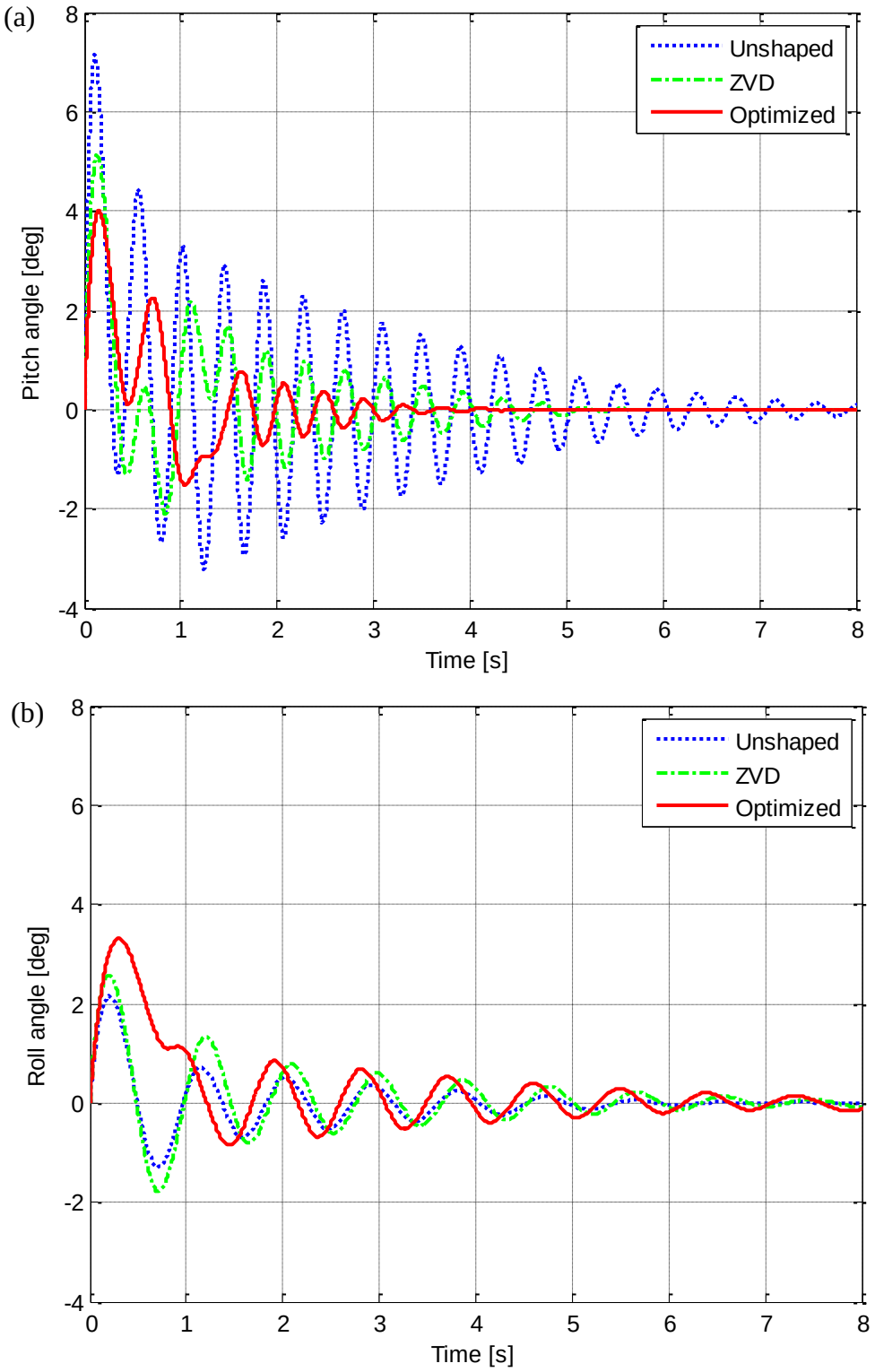


Fig. 2-8 Time responses of suspended mobile platform when $\omega_o = \omega_d$.

(a) In pitch direction, and (b) In roll direction.

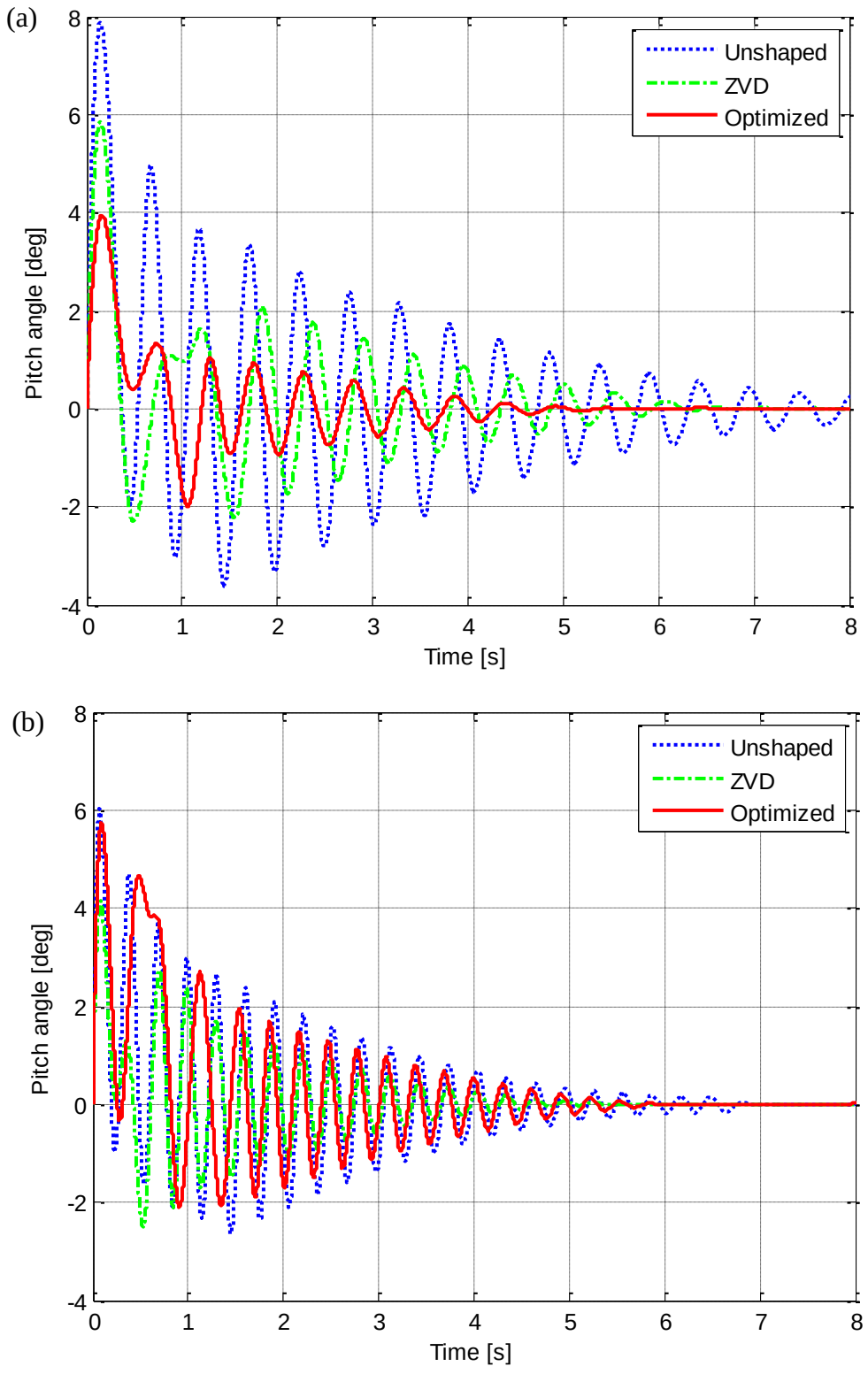


Fig. 2-9 Time responses of suspended mobile platform in pitch direction.

(a) $\omega_o = 0.78\omega_d$, and (b) $\omega_o = 1.34\omega_d$.

Figure 2-10 plots numerous trials based on various frequency errors. In Fig. 2-10, the abscissa denotes the frequency ratio expressed as ω_o / ω_d, the ordinate is the pitch angle when the movement has settled. The optimal input shapers offer better performance than the ZVD shapers when $\omega_o \in [0.75\omega_d,\ 1.25\omega_d]$, but at $\omega_o < 0.75\omega_d$ or $\omega_o > 1.25\omega_d$ the performance of the optimal input shapers is poorer than that of the ZVD shapers. This result suggests that the improvement of the performance at $0.75\omega_d \le \omega_o \le 1.25\omega_d$ comes at the deteriorating performance when the operating frequency ω_o is beyond this range. Such deterioration can be expected since there is no factor in the cost function to reflect the performance of the optimal shapers in that range. Additionally, the improvement comes at a trade-off with the long settling time.

On the basis of the above discussion, the optimal input shapers shorten the settling time compared with the ZVD shapers, and they are capable of decreasing vibration and are more robust than the ZVD shapers when the frequency error changes are in the acceptable range.

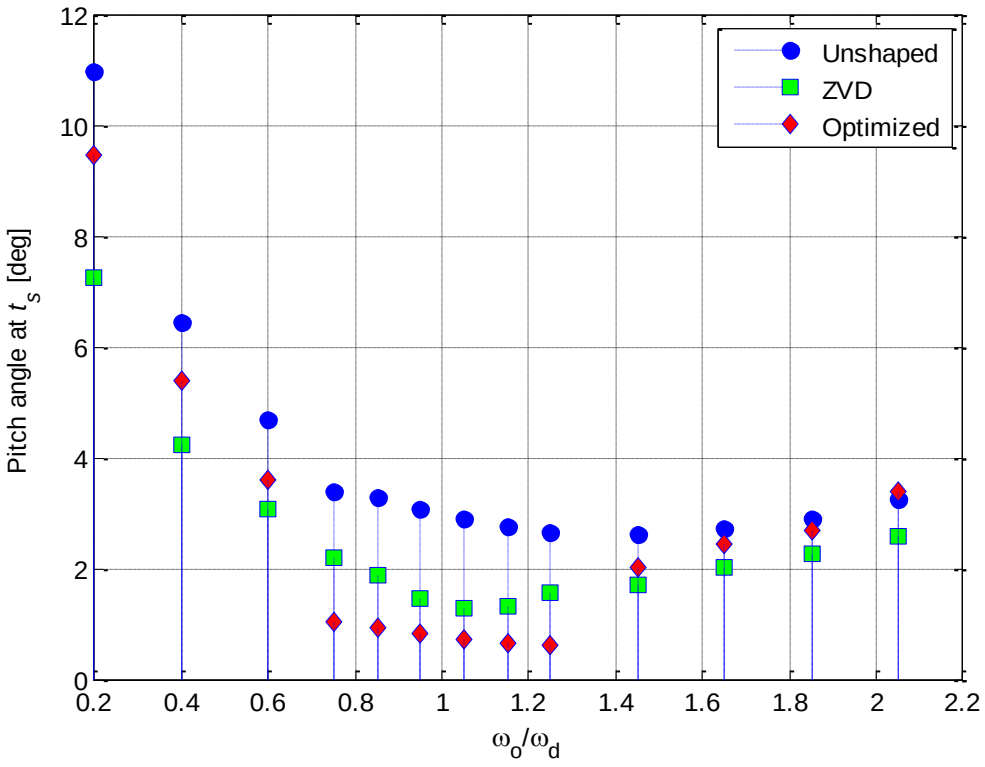


Fig. 2-10 Numerous trial results of pitch angle at settling time t_s vs ω_o / ω_d.

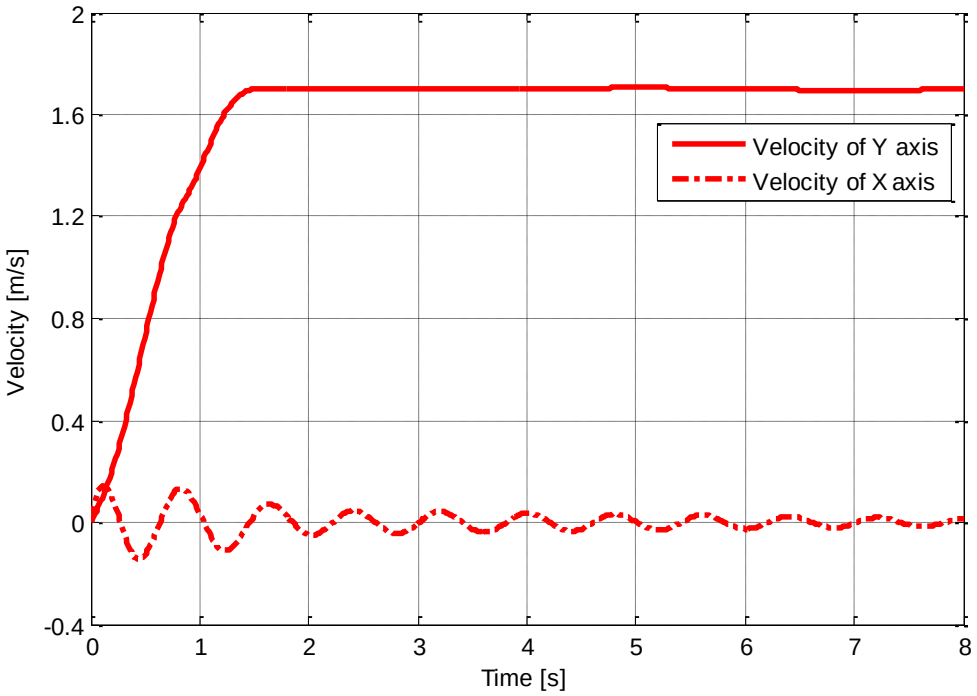


Fig. 2-11 Velocities of X and Y axis.

Furture insight into the system is gained by examining the velocity and motion trace after the optimal multi-input shaping when ω_o is equal to ω_d as depicted in Figs. 2-11−2-15. Figure 2-11 shows the velocities of the CG in the X and Y directions during the motion. At the beginning the velocity of the Y axis increases sharply and then reaches an almost constant value after a settling time, which corresponds with the profile of the input torques as show in Fig. 2-7. The velocity of the X axis does not settle but fluctuates around zero, because of the vibratory roll angle as shown in Fig. 2-8(b), weakening with time as a result of the damping properties.

Figure 2-12 shows the motion trace versus time and its three-plane projections are shown in Figs. 2-13−2-15. Corresponding with Fig. 2-11, before settling, the displacement of the Y axis is not in direct proportion to the elapsed time, the displacement of the X axis displays stronger vibration and the motion trace follows a snake like progress. After settling, as the vibration of the roll angle weakens gradually, the amplitude of the vibration and the extent of deviation from the straight forward movement become smaller as shown in Figs. 2-14 and 2-15.

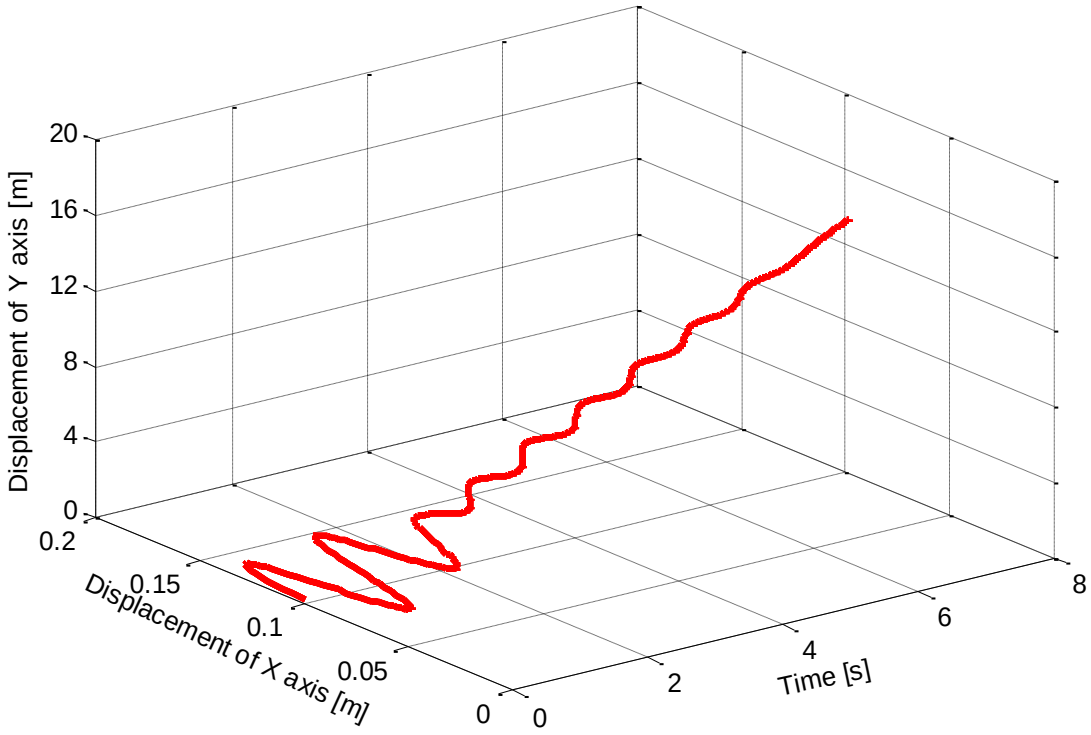


Fig. 2-12 Motion trace with time passage.

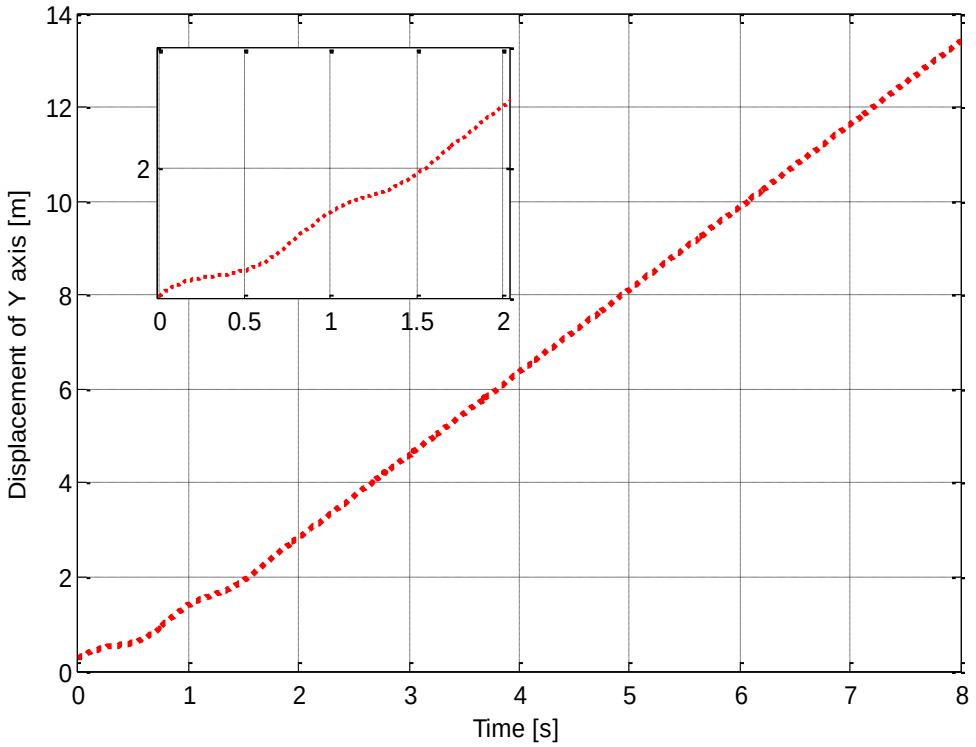


Fig. 2-13 Displacement of Y axis (projection on T-Y plane).

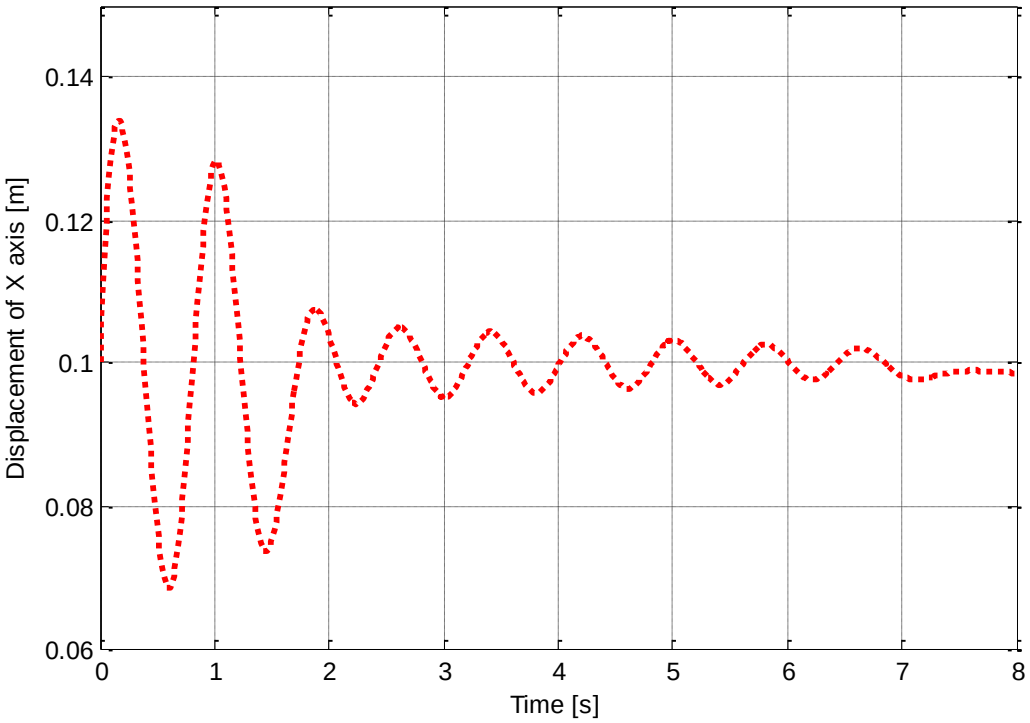


Fig. 2-14 Displacement of X axis (projection on T-X plane).

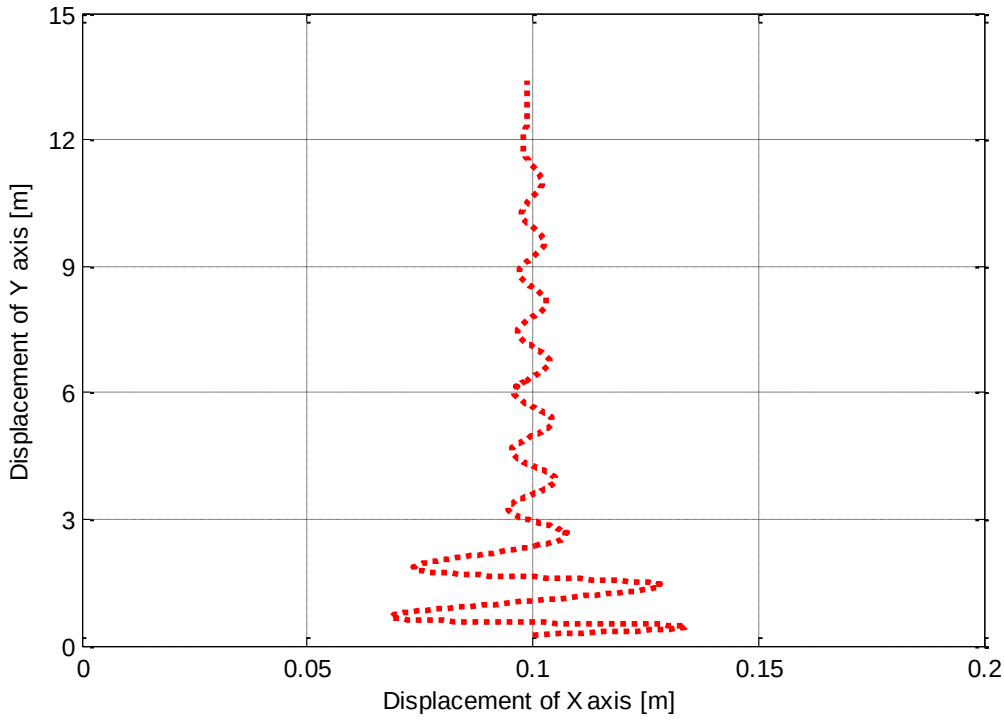


Fig. 2-15 Motion trace (projection on X-Y plane).

2.4 Problem Two: Suspended Platform Motion with a Dynamic Manipulator

2.4.1 DPM Concept

The DPM concept is used to obtain a dynamic model that accounts for the interaction between the platform and manipulator. According to the DPM concept, a point on the base platform (preferably its center of mass) represents the translational motion of the system [10]. As shown in Fig. 2-16, the kinematics of the mobile manipulator system can be developed by using a set of body-fixed geometric vectors to formulate the position and velocity with respect to a representative point *p*. The motion of the center of mass (CM) is used to describe the system translation with respect to the inertial frame of reference, O_g-$X_gY_gZ_g$. The CMs of each body are defined as c_*, the joints are defined as j_*, and the rest of the definitions are described in Fig. 2-16.

The inertial position of the representative point *p* is R_p , which can be written as

$$\boldsymbol{R}_p = \boldsymbol{R}_{c_b} + \boldsymbol{r}_{p/c_b}, \qquad n = 1,2,3,b. \tag{2-20}$$

$$\boldsymbol{r}_{p/c_b} = \boldsymbol{r}_{p/c_n} + \boldsymbol{r}_{c_n/c_b}, \qquad n = 1,2,3. \tag{2-21}$$

$$\boldsymbol{r}_{c_n/c_b} = \begin{cases} \boldsymbol{r}_{c_b/c_b} = 0, & n = b. \\ \boldsymbol{r}_{c_n/c_b} = \boldsymbol{r}_{j_0/c_b} - \boldsymbol{r}_{j_{n-1}/c_n} + \sum_{k=1}^{n-1}\left(\boldsymbol{r}_{j_k/c_k} - \boldsymbol{r}_{j_{k-1}/c_k}\right), & n = 1,2,3. \end{cases} \tag{2-22}$$

Substituting Eq. (2-22) in Eq. (2-21) yields

$$\boldsymbol{r}_{p/c_b} = \begin{cases} \boldsymbol{r}_{p/c_b}, & n = b. \\ \boldsymbol{r}_{p/c_n} + \boldsymbol{r}_{j_0/c_b} - \boldsymbol{r}_{j_{n-1}/c_n} + \sum_{k=1}^{n-1}\left(\boldsymbol{r}_{j_k/c_k} - \boldsymbol{r}_{j_{k-1}/c_k}\right), & n = 1,2,3. \end{cases} \tag{2-23}$$

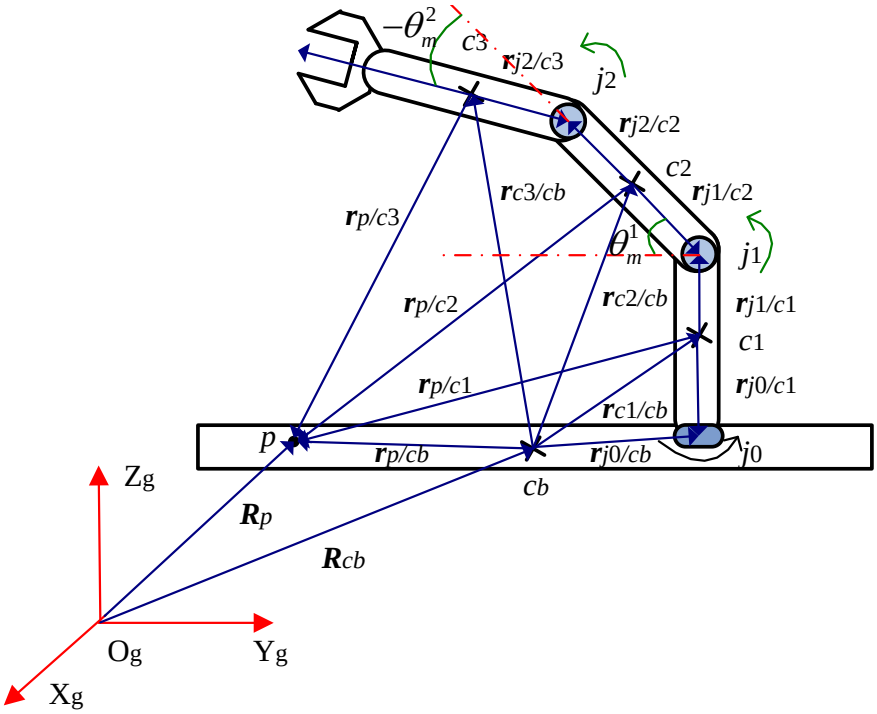


Fig. 2-16 Vector diagram of the suspended mobile robot with a dynamic manipulator for DPM concept.

Substituting Eq. (2-23) in Eq. (2-20) completes the position analysis and yields

$$\boldsymbol{R}_p = \begin{cases} \boldsymbol{R}_{c_b} + \boldsymbol{r}_{p/c_b}, & n = b. \\ \boldsymbol{R}_{c_b} + \boldsymbol{r}_{p/c_n} + \boldsymbol{r}_{j_0/c_b} - \boldsymbol{r}_{j_{n-1}/c_n} + \sum_{k=1}^{n-1}\left(\boldsymbol{r}_{j_k/c_k} - \boldsymbol{r}_{j_{k-1}/c_k}\right), & n = 1,2,3. \end{cases} \tag{2-24}$$

To obtain the inertial velocity of point *p*, Eq. (2-20) is differentiated as

$$\dot{\boldsymbol{R}}_p = \dot{\boldsymbol{R}}_{c_b} + \dot{\boldsymbol{r}}_{c_n/c_b} + \boldsymbol{\omega}_m^n \times \boldsymbol{r}_{p/c_n}, \tag{2-25}$$

where $\boldsymbol{\omega}_m^n$ denotes the angular velocity of *i*th link, "×" is the cross product operation. Similar to the position analysis presented above, $\dot{\boldsymbol{r}}_{c_n/c_b}$ is obtained by differentiating Eq. (2-22) as follows:

$$\dot{\boldsymbol{r}}_{c_n/c_b} = \begin{cases} \dot{\boldsymbol{r}}_{c_b/c_b} = 0, & n = b. \\ \boldsymbol{\omega}_b \times \boldsymbol{r}_{j_0/c_b} - \boldsymbol{\omega}_m^n \times \boldsymbol{r}_{j_{n-1}/c_n} + \sum_{k=1}^{n-1}\left\{\boldsymbol{\omega}_m^k \times \left(\boldsymbol{r}_{j_k/c_k} - \boldsymbol{r}_{j_{k-1}/c_k}\right)\right\}, & n = 1,2,3. \end{cases} \tag{2-26}$$

Substituting Eq. (2-26) in Eq. (2-25), the inertial velocity is derived as follows:

$$\dot{\boldsymbol{R}}_p = \begin{cases} \dot{\boldsymbol{R}}_{c_b} + \boldsymbol{\omega}_b \times \boldsymbol{r}_{p/c_b}, & n = b. \\ \dot{\boldsymbol{R}}_{c_b} + \boldsymbol{\omega}_b \times \boldsymbol{r}_{j_0/c_b} + \boldsymbol{\omega}_m^n \times \left(\boldsymbol{r}_{p/c_n} - \boldsymbol{r}_{j_{n-1}/c_n}\right) + \sum_{k=1}^{n-1} \left\{\boldsymbol{\omega}_m^k \times \left(\boldsymbol{r}_{j_k/c_k} - \boldsymbol{r}_{j_{k-1}/c_k}\right)\right\}, & n = 1, 2, 3. \end{cases}$$

(2-27)

2.4.2 Dynamic Model

To express the dynamics of the mobile manipulator system, the Lagrange approach is used in conjunction with the DPM concept. Similar to the problem one, the Lagrange equation of the second kind can be written as

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}_i}\right) - \frac{\partial T}{\partial q_i} + \frac{\partial U}{\partial q_i} + \frac{\partial U_F}{\partial \dot{q}_i} = Q_i \qquad i = 1, \cdots, N, \tag{2-28}$$

The generalized coordinate is defined as follows:

$$\boldsymbol{q} = \left\{\boldsymbol{R}_b, \boldsymbol{\Theta}_b, \boldsymbol{\Theta}_m\right\}^T, \tag{2-29}$$

where $\boldsymbol{R}_b = \left(x_G, y_G, z_G\right)$ and $\boldsymbol{\Theta}_b = \left(\theta_y, \theta_p, \theta_r\right)$ describe position vectors of the CM of the suspended platform and its Euler angles. The vector of joint angles for the manipulator is represented by $\boldsymbol{\Theta}_m = \left(\theta_m^0, \theta_m^1, \theta_m^2\right)$. Furthermore, x_G, y_G, z_G are the CM position of the platform with respect to the inertial O_g-$X_gY_gZ_g$ coordinate axes; θ_y, θ_p, θ_r are the yaw, pitch and roll angles, respectively; and θ_m^0, θ_m^1, θ_m^2 denote each joint angle of the manipulator.

By ignoring extra forces, the vector of force can be written as

$$\boldsymbol{Q} = \begin{Bmatrix} 0^{3\times1} \\ \tau_m^{3\times1} \end{Bmatrix} + \boldsymbol{J}_b^T \begin{Bmatrix} 0 \\ \tau_b \end{Bmatrix}, \tag{2-30}$$

where τ_b and τ_m are torques applied on the platform and each joint, $\boldsymbol{J}_b$ is the Jacobian matrix.

To derive the terms of the total kinetic energy of the system, it is divided into two parts

$$T = T_b + \sum_{n=1}^{3} T_m^n = \frac{1}{2} M_b \dot{\boldsymbol{R}}_{c_b} \cdot \dot{\boldsymbol{R}}_{c_b} + \frac{1}{2} \sum_{n=1}^{3} M_m^n \dot{\boldsymbol{R}}_{c_b} \cdot \dot{\boldsymbol{R}}_{c_b} + \frac{1}{2} \boldsymbol{\omega}_b \cdot I_b \boldsymbol{\omega}_b + \dot{\boldsymbol{R}}_{c_b} \cdot \left(\sum_{n=1}^{3} M_m^n \dot{\boldsymbol{r}}_{c_i/c_b} \right) + \frac{1}{2} \sum_{n=1}^{3} \left\{ M_m^n \dot{\boldsymbol{r}}_{c_i/c_b} \cdot \dot{\boldsymbol{r}}_{c_i/c_b} + \boldsymbol{\omega}_m^n \cdot I_m^n \boldsymbol{\omega}_m^n \right\}, \tag{2-31}$$

where T_b and T_m^n are the kinetic energy for the suspended platform and *n*th link of the manipulator, "$\cdot$" is the inner product operator. The mass and the moment of inertia of the platform and *n*th link are represented by M_b, M_m^n and I_b, I_m^n, respectively.

The potential energy is divided into three parts, as shown in Eq. (2-32). Using the geometric vectors presented in Fig. 2-11, it can be written as

$$\begin{aligned} U &= U_b + \sum_{n=1}^{3} U_m^n + U_s \\ &= M_b g \cdot \boldsymbol{R}_{c_b} + \sum_{n=1}^{3} \left\{ M_m^n g \cdot \left(\boldsymbol{R}_{c_b} + \boldsymbol{r}_{c_i/\,c_b} \right) \right\} + \frac{1}{2} k \left(\boldsymbol{\delta}_L \cdot \boldsymbol{\delta}_L + \boldsymbol{\delta}_R \cdot \boldsymbol{\delta}_R \right), \end{aligned} \tag{2-32}$$

where U_b and U_m^n are the gravitational potential energy of the platform and the *n*th link of the manipulator, and U_s is the spring potential energy.

To obtain the dissipated energy generated by the damping systems, the concept of Rayleigh's dissipation function is utilized. Therefore, the dissipated energy can be expressed as

$$U_F = \frac{1}{2} c \left(\dot{\boldsymbol{\delta}}_L \cdot \dot{\boldsymbol{\delta}}_L + \dot{\boldsymbol{\delta}}_R \cdot \dot{\boldsymbol{\delta}}_R \right). \tag{2-33}$$

Using this approach, all the three main components of Eq. (2-28) is obtained. Moreover, by taking the derivatives of these terms, the dynamic model can be represented as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\dot{\boldsymbol{q}}, \boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{Q}, \tag{2-34}$$

where $\boldsymbol{M}(\boldsymbol{q})$, $\boldsymbol{C}(\dot{\boldsymbol{q}}, \boldsymbol{q})$, $\boldsymbol{G}(\boldsymbol{q})$ and $\boldsymbol{Q}$ denote the mass matrix, nonlinear velocity vector, gravity vector and torque matrix, respectively. Detailed derivation of the dynamic model is described in Appendix B.

2.4.3 Optimal Trajectory Generation

In this subsection, the jerks of manipulator are minimized by using PSO with chaos when the workspace of the robot is collision-free. First, the mathematical

description of the trajectory optimization is presented, and then the PSO with chaos method is employ to search for the optimal trajectory of each joint of the manipulator.

In order to formulate the trajectory, the cubic splines are applied to each joint to interpolate the joint trajectory between every two neighbor knot points [17]. The trajectory in the *j*th time interval of the *n*th joint is expressed as

$$\Gamma_{n,j}(t) = \alpha_{n,j} + \beta_{n,j}t + \gamma_{n,j}t^2 + \lambda_{n,j}t^3, \qquad n = 0,1,2. \tag{2-35}$$

As shown in Fig. 2-1, the manipulator has three joints. Supposing the trajectory for each joint has M + 1 knot points including the first and last knot points, the trajectory is expressed as

$$\{q_{n,1}, q_{n,2}, \cdots, q_{n,k}, \cdots, q_{n,M+1}\}, \qquad n = 0,1,2, \tag{2-36}$$

where $q_{n,k}$ denotes the rotational angle of *n*th joint in the (*k*-1)th time interval.

Thus, for each joint, there exist M time intervals. Because the displacements and velocities of the starting and ending knot points are constrained, the second and penultimate knot points are chosen as the extra points to represent the robot trajectory. The interpolation point can be described in Table 2-2. Furthermore, Fig. 2-17 visualizes the trajectory with extra knots.

To derive the coefficients, the constraints of the displacements and velocities of the starting and ending knot points are used. The coefficients can be derived as

$$\alpha_{n,j} = q_{n,j}, \tag{2-37}$$

$$\beta_{n,j} = v_{n,j}, \tag{2-38}$$

$$\gamma_{n,j} = \frac{3}{h_j^2}\left(q_{n,j+1} - q_{n,j}\right) - \frac{1}{h_j}\left(v_{n,j+1} + 2v_{n,j}\right), \tag{2-39}$$

$$\lambda_{n,j} = -\frac{2}{h_j^3}\left(q_{n,j+1} - q_{n,j}\right) + \frac{1}{h_j^2}\left(v_{n,j+1} + v_{n,j}\right), \tag{2-40}$$

where $v_{n,j+1}$ denotes the velocity of the *n*th joint in the *j*th time interval, h_j denotes the interval length of the *j*th time interval for all the joints.

Table 2-2 Knot position for each joint of the manipulator.

Joint number	Knot (degree)					
	1	2	3	4	5	6
0	-10	extra knot	60	20	extra knot	55
1	20		45	70		35
2	15		-10	50		30

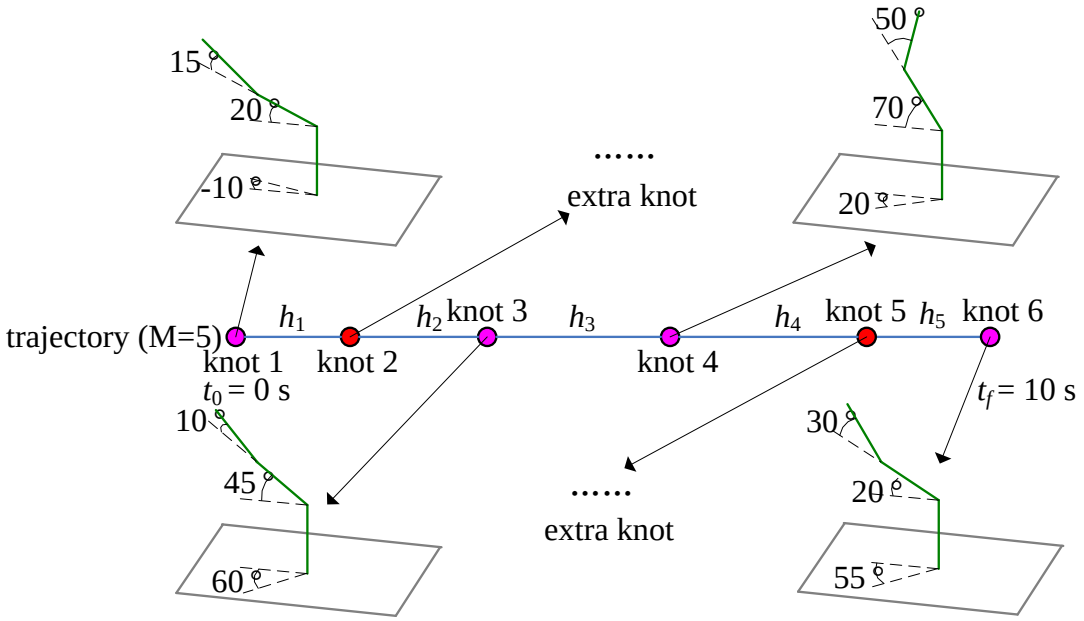


Fig. 2-17 Graphical representation of the trajectory with extra knots.

Substituting these coefficients into Eq. (2-35), the jerk of the trajectory of the *n*th joint, i.e. the third derivative of Eq. (2-35), can be obtained as

$$J_{n,j}(t) = \dddot{\Gamma}_{n,j}(t) = -\frac{12}{h_j^3}\left(q_{n,j+1} - q_{n,j}\right) + \frac{6}{h_j^2}\left(v_{n,j+1} + v_{n,j}\right), \qquad n = 0,1,2. \tag{2-41}$$

To this end, the optimal trajectory generation problem can be treated as an optimization problem. The design variables are defined by

$$\boldsymbol{D} = [h_1, h_2, \cdots, h_5, q_{n,2}, q_{n,5}] \qquad n = 0,1,2. \tag{2-42}$$

The search space is given as

$$\boldsymbol{S} = \begin{bmatrix} h_1^1 & h_2^1 & h_3^1 & h_4^1 & h_5^1 & q_{n,2}^1 & q_{n,5}^1 \\ h_1^2 & h_2^2 & h_3^2 & h_4^2 & h_5^2 & q_{n,2}^2 & q_{n,5}^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ h_1^s & h_2^s & h_3^s & h_4^s & h_5^s & q_{in,2}^s & q_{n,5}^s \end{bmatrix}^T, \qquad n = 0,1,2\,. \tag{2-43}$$

Considering the high operating efficiency and low manipulator vibration, the cost function is defined to obtain the time-jerk synthetic optimal trajectory as

$$f = \min\left(\zeta_T \sum_{n=0}^{2} T_n + \zeta_J \sum_{n=0}^{2} \max\left| J_{n,j}(t) \right| \right), \tag{2-44}$$

where T_n is the execution time of the nth joint, ζ_T and ζ_J are weighting factors of the total execution time and the total minimax jerks.

2.4.4 Results and Discussion

A robot system is illustrated as shown in Fig. 2-1. The parameters of the mobile robot are listed in Table 2-1. The time for the motion is set as t_f = 10 s, the number of the particle as s = 100, and the iterations as k = 100. After using the proposed approach to generate the time-jerk synthetic optimal trajectory based on the dynamic model, some results for different kinds of weighting ratios ζ_J/ζ_T can be obtained.

The optimal trajectories of each joint and the response of the platform and manipulator are shown in Figs. 2-18−2-20. From these figures, some assertions can be obtained as follows:

(1) Owing to use the cubic splines interpolation, the position, velocity and acceleration of each joint are continuous after optimization.

(2) Increasing the weighting ratio value ζ_J/ζ_T has a beneficial effect in reducing the jerks but lengthens the execution time. Therefore, the total execution time is necessary in the cost function.

(3) The vector of torque is derived as the inertia matrix multiplied by the vector of jerk (i.e. $\dot{\boldsymbol{\tau}} = \boldsymbol{I} \cdot \boldsymbol{J}$). It is easy to infer that maximizing the value ζ_J/ζ_T helps to reduce the jerks which in turn reduce the jumps of torque.

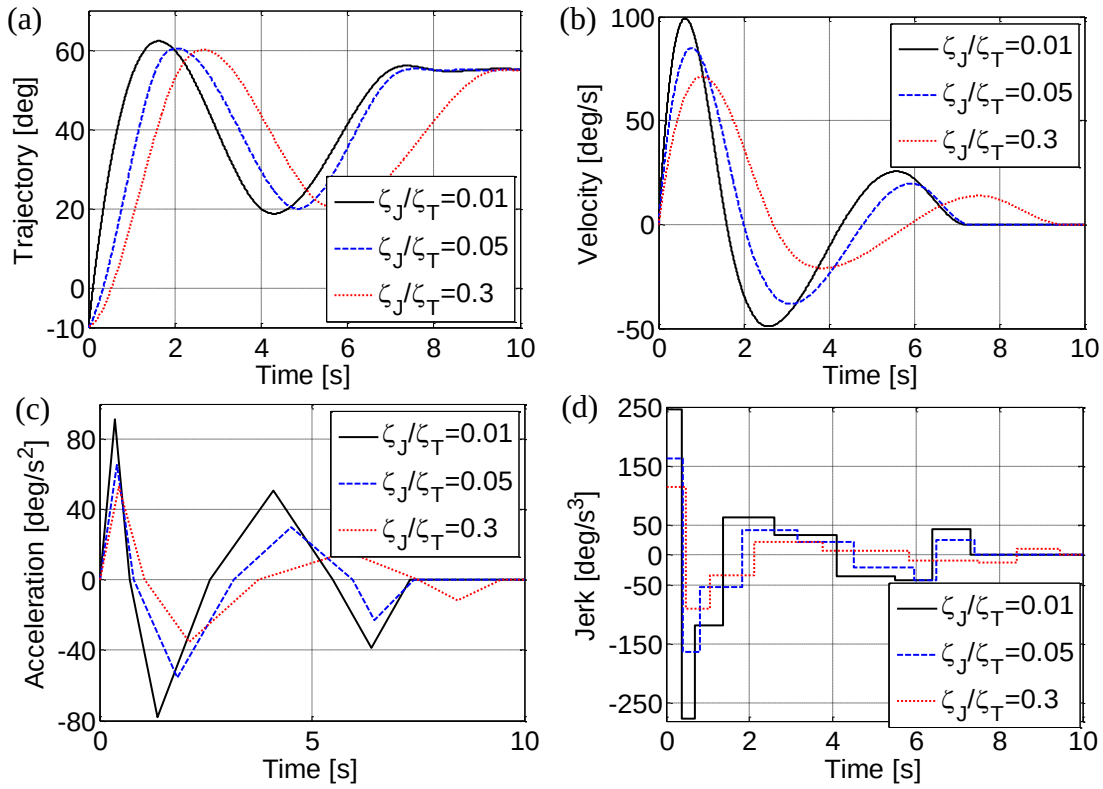


Fig. 2-18 Time responses of joint 0 for several kinds of ζ_J/ζ_T.

(a) Trajectory, (b) Velocity, (c) Acceleration, and (d) Jerk.

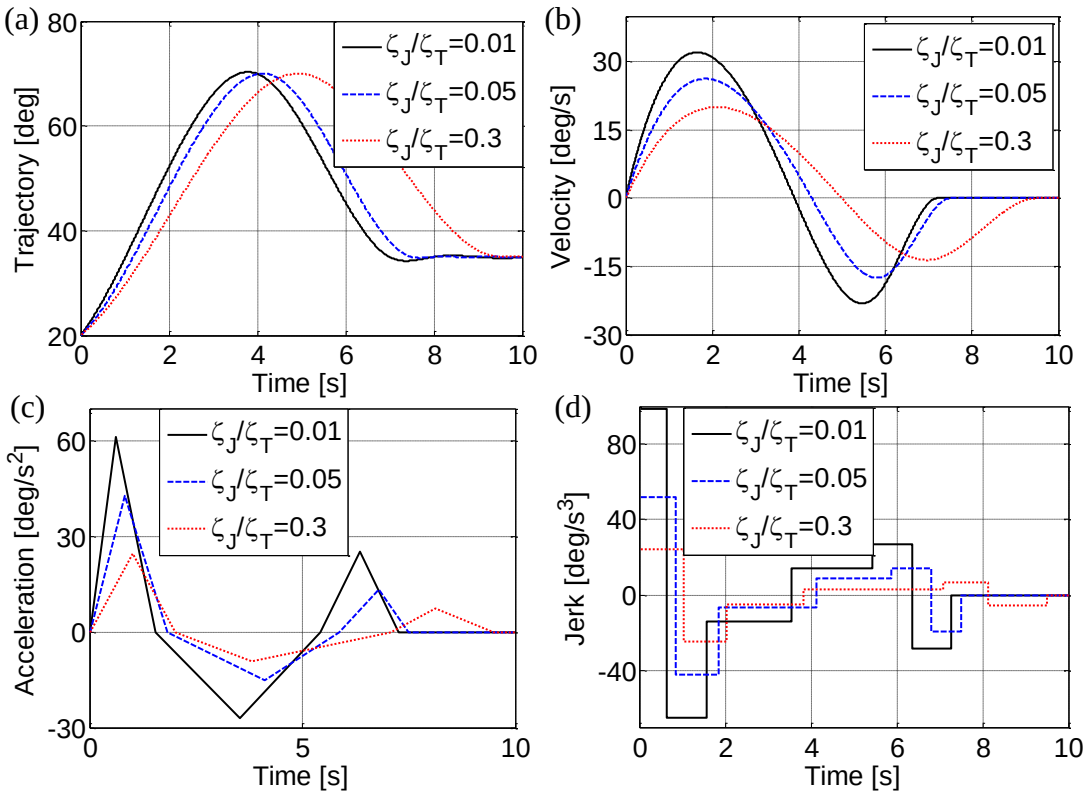


Fig. 2-19 Time responses of joint 1 for several kinds of ζ_J/ζ_T.

(a) Trajectory, (b) Velocity, (c) Acceleration, and (d) Jerk.

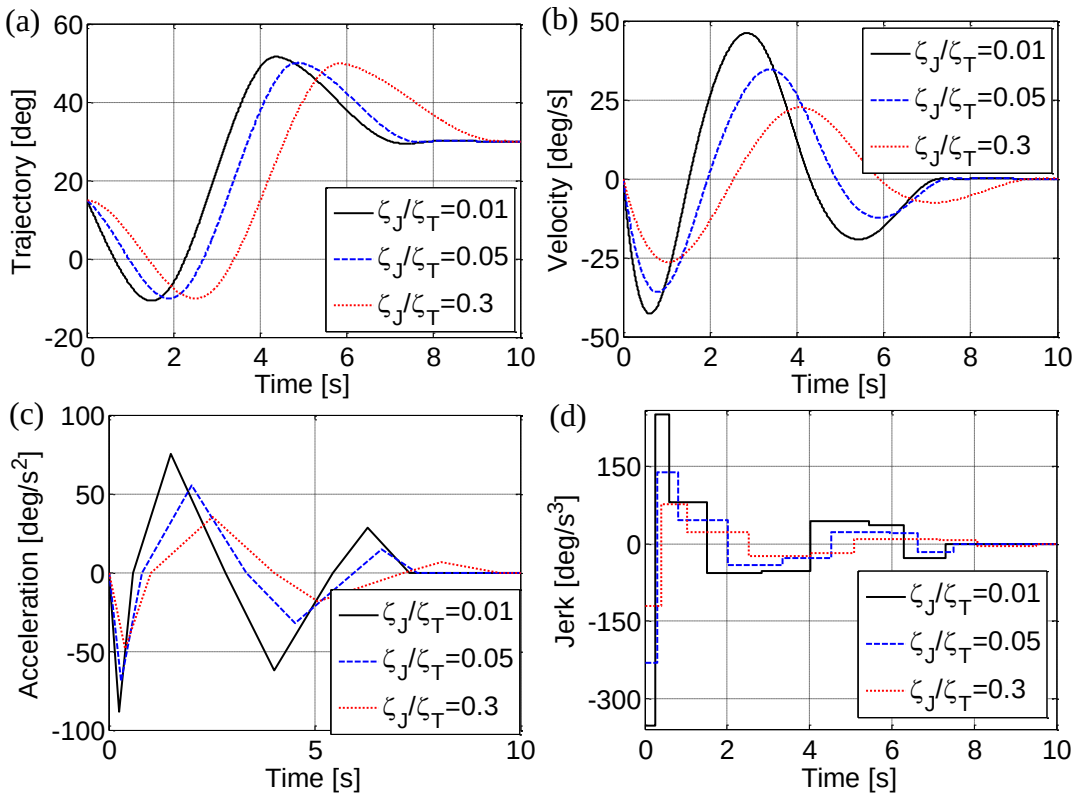


Fig. 2-20 Time responses of joint 2 for several kinds of ζ_J/ζ_T.

(a) Trajectory, (b) Velocity, (c) Acceleration, and (d) Jerk.

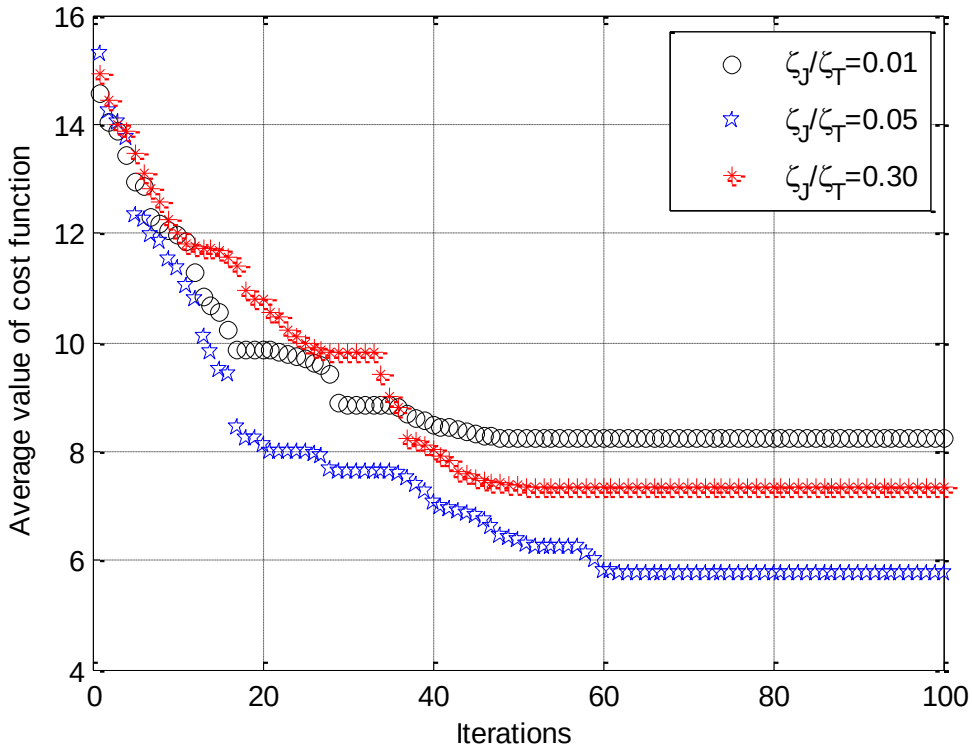


Fig. 2-21 Average value of cost function versus iterations for several kinds of ζ_J/ζ_T.

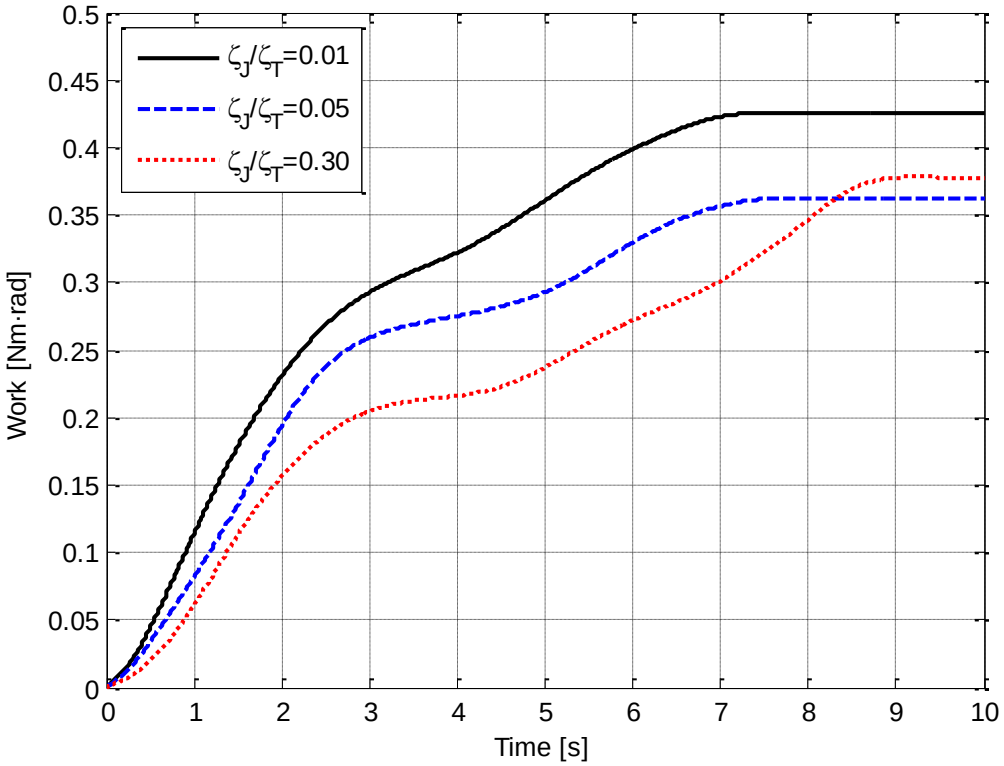


Fig. 2-22 Time history of control energy for several kinds of ζ_J/ζ_T .

Figure 2-21 shows the convergent characteristics of cost function versus iterations. A continuous decrease in average of cost function is also indicative of a smooth convergence to a solution. As seen in Fig. 2-21, when ζ_J/ζ_T = 0.05, the time-jerk synthetic optimal trajectory is the best one comparing with other cases.

Figure 2-22 shows the total control energy of the manipulator for different kinds of weighting ratios. It is obvious that the control input requires relative minimum energy comparing with other cases when ζ_J/ζ_T = 0.05. The work of the control input can be obtained as

$$W(t)=\sum_{n=0}^{2} I_m^{n+1}\cdot\int_{t_0}^{t_f}\left\{\left|\alpha_n(t)\right|\cdot\left|\omega_n(t)\right|\right\}dt\ , \tag{2-45}$$

where $\alpha_n(t)$ and $\omega_n(t)$ are the angular acceleration and velocity of *n*th joint at time *t*. As shown in Eq. (2-45), the work of the control input is related to acceleration and execution time which are considered in the cost function.

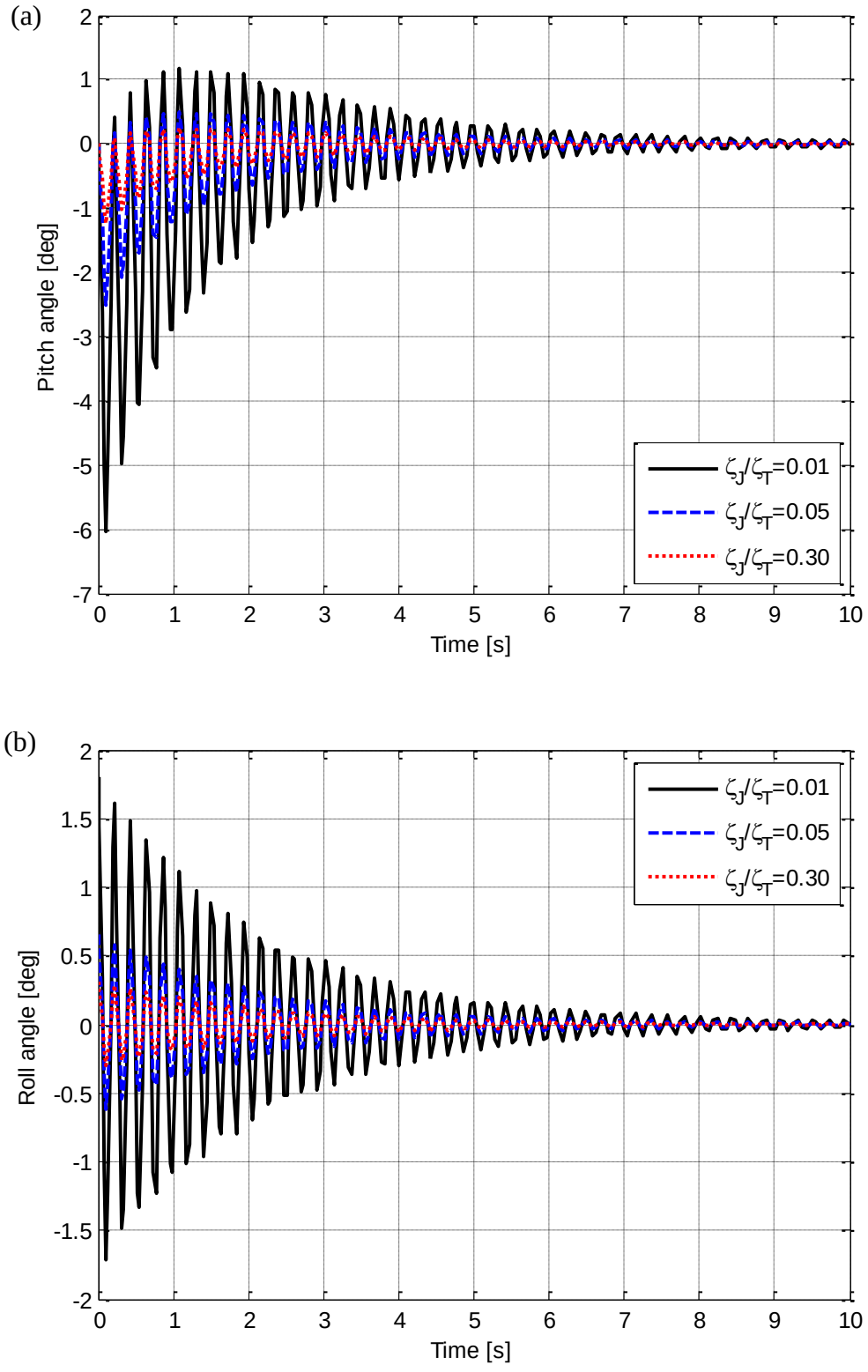


Fig. 2-23 Time responses of suspended mobile platform for several kinds of ζ_J/ζ_T.

(a) In pitch direction, and (b) In roll direction.

Finally, the jerks and the jumps of torque of each joint are decreased after optimization. In consequence, when each joint of the manipulator is forced along the optimal trajectories, the vibrations of the pitch and roll angles are reduced as shown in Fig. 2-23. These results reveal the effectiveness of the presented PSO with chaos method to solve the contradictory problem between high operating efficiency and low residual vibration using limited control energy. Furthermore, the nature frequencies of the pitch and roll vibrations in problems one and two are different. The causes of this difference may be the relatively large mass of the upright manipulator and the frictions considered in problem one.

2.5 Summary

In this chapter, two problems of vibration reduction of wheeled mobile manipulator with suspension system were investigated. To overcome the first problem, an explicit dynamic model involving the incorporation of the CG shift and suspension characteristic was formulated. In order to minimize residual vibration as well as settling time, the PSO with chaos method was employed to optimize the multi-input shapers. Other objectives such as improved robustness and saturation limits of the input torques were also incorporated in the optimization scheme. To overcome the second problem, a dynamic model was presented considering the dynamics of manipulator and the interaction between suspended platform and manipulator by introducing the DPM concept. In order to generate the optimal trajectory, the PSO with chaos method was applied to optimize the parameters of inter-knots which were used to polynomial interpolation. The optimal trajectories considering minimum execution time and minimax jerks were obtained after optimization.

When the suspended platform moves with static manipulator, the optimal multi-input shaping is effective to suppress vibration with shorter settling time and robustness. When the suspended platform moves with dynamic manipulator, the PSO with chaos method was effective to reduce the vibration by finding the time-jerk synthetic optimal trajectory.

Appendix A: Dynamic Model for Problem One

To obtain the dynamic model of problem one, the kinetic, potential, and dissipated energies can be derived as follows:

$$\begin{aligned} T &= \frac{1}{2}Mv_G^2 + \frac{1}{2}I_r\dot{\theta}_r^2 + \frac{1}{2}I_p\dot{\theta}_p^2 + \frac{1}{2}I_w\dot{\varphi}_L^2 + \frac{1}{2}I_w\dot{\varphi}_R^2 \\ &= \frac{1}{2}M\left\{\left[\frac{1}{2}\left(v_L + v_R\right) + \frac{e}{w}\left(v_L - v_R\right)\right]^2 + \left[\frac{l_r}{w}\left(v_L - v_R\right)\right]^2\right\} \\ &\quad + \frac{1}{2}\left(I_r\dot{\theta}_r^2 + I_p\dot{\theta}_p^2 + I_w\dot{\varphi}_L^2 + I_w\dot{\varphi}_R^2\right) \\ &= \frac{Md^2\left(w^2 + 4e^2 - 4we + 4l_r^2 + 2w^2M_w\right)}{32w^2}\dot{\varphi}_L^2 \\ &\quad + \frac{Md^2\left(w^2 + 4e^2 + 4we + 4l_r^2 + 2w^2M_w\right)}{32w^2}\dot{\varphi}_R^2 \\ &\quad + \frac{Md^2\left(w^2 - 4e^2 - 4l_r^2\right)}{16w^2}\dot{\varphi}_L\dot{\varphi}_R + \frac{1}{2}M\left(e^2 + h^2\right)\dot{\theta}_r^2 + \frac{1}{2}M\left[\left(l - l_r\right)^2 + h^2\right]\dot{\theta}_p^2, \end{aligned} \tag{A-1}$$

$$\begin{aligned} U &= \frac{1}{2}k\left[\left(\frac{w}{2}\theta_r + l\theta_p\right)^2 + \left(-\frac{w}{2}\theta_r + l\theta_p\right)^2\right] - Mgh\left(1 - \cos\theta_p\cos\theta_r\right) \\ &= kl^2\theta_p^2 + \frac{1}{4}kw^2\theta_r^2 - Mgh + Mgh\cos\theta_p\cos\theta_r, \end{aligned} \tag{A-2}$$

$$U_F = \frac{1}{2}c\left(2l^2\dot{\theta}_p^2 + \frac{w^2}{2}\dot{\theta}_r^2\right) = cl^2\dot{\theta}_p^2 + \frac{1}{4}cw^2\dot{\theta}_r^2. \tag{A-3}$$

Lagrange equations are given as follows:

$$\frac{d}{dt}\left(\frac{\partial T}{\partial\dot{\varphi}_L}\right) - \frac{\partial T}{\partial\varphi_L} + \frac{\partial U}{\partial\varphi_L} + \frac{\partial U_F}{\partial\dot{\varphi}_L} = \tau_L, \tag{A-4}$$

$$\frac{d}{dt}\left(\frac{\partial T}{\partial\dot{\varphi}_R}\right) - \frac{\partial T}{\partial\varphi_R} + \frac{\partial U}{\partial\varphi_R} + \frac{\partial U_F}{\partial\dot{\varphi}_R} = \tau_R, \tag{A-5}$$

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\theta}_p}\right)-\frac{\partial T}{\partial \theta_p}+\frac{\partial U}{\partial \theta_p}+\frac{\partial U_F}{\partial \dot{\theta}_p}=-hMa_y=-hM\left[\frac{1}{2}\left(\dot{v}_L+\dot{v}_R\right)+\frac{e}{w}\left(\dot{v}_L-\dot{v}_R\right)\right]$$
$$=-hM\left[\left(\frac{d}{4}-\frac{ed}{2w}\right)\ddot{\varphi}_L+\left(\frac{d}{4}+\frac{ed}{2w}\right)\ddot{\varphi}_R\right], \tag{A-6}$$

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\theta}_r}\right)-\frac{\partial T}{\partial \theta_r}+\frac{\partial U}{\partial \theta_r}+\frac{\partial U_F}{\partial \dot{\theta}_r}=-hMa_x$$
$$=-hM\frac{l_r}{w}\left(\dot{v}_L-\dot{v}_R\right)=-hM\frac{dl_r}{2w}\left(\ddot{\varphi}_L-\ddot{\varphi}_R\right). \tag{A-7}$$

Substituting energy components in Eqs. (A-4)−(A-7) obtains

$$\frac{Md^2\left(w^2+4e^2-4we+4l_r^2+2w^2M_w\right)}{16w^2}\ddot{\varphi}_L+\frac{Md^2\left(w^2-4e^2-4l_r^2\right)}{16w^2}\ddot{\varphi}_R=\tau_L, \tag{A-8}$$

$$\frac{Md^2\left(w^2+4e^2+4we+4l_r^2+2w^2M_w\right)}{16w^2}\ddot{\varphi}_R+\frac{Md^2\left(w^2-4e^2-4l_r^2\right)}{16w^2}\ddot{\varphi}_L=\tau_R, \tag{A-9}$$

$$M\left[\left(l-l_r\right)^2+h^2\right]\ddot{\theta}_p+2cl^2\dot{\theta}_p+2kl^2\theta_p-Mgh\theta_p$$
$$=-\frac{Mhd}{4}\left(\ddot{\varphi}_L+\ddot{\varphi}_R\right)+\frac{Mhed}{2w}\left(\ddot{\varphi}_L-\ddot{\varphi}_R\right), \tag{A-10}$$

$$M\left(e^2+h^2\right)\ddot{\theta}_r+\frac{1}{2}kw^2\theta_r-Mgh\theta_r+\frac{1}{2}cw^2\dot{\theta}_r=\frac{Mhdl_r}{2w}\left(\ddot{\varphi}_R-\ddot{\varphi}_L\right). \tag{A-11}$$

Subtraction of Equation (A-9) from Equation (A-8) yields

$$\frac{Md^2\left(8e^2+8l_r^2+2w^2M_w\right)}{16w^2}\left(\ddot{\varphi}_L-\ddot{\varphi}_R\right)+\frac{4ew}{16w^2}\left(\ddot{\varphi}_L+\ddot{\varphi}_R\right)=\tau_L-\tau_R. \tag{A-12}$$

Addition of Equation (A-8) and Equation (A-9) yields

$$\frac{Md^2\left(2w^2+2w^2M_w\right)}{16w^2}\left(\ddot{\varphi}_L+\ddot{\varphi}_R\right)-\frac{4ew}{16w^2}\left(\ddot{\varphi}_L-\ddot{\varphi}_R\right)=\tau_L+\tau_R. \tag{A-13}$$

Substituting Equations (A-12) and (A-13) in Equations (A-10) and (A-11), the rotational angles of left and right wheels can be expressed by input torques. Finally, the dynamic model of the system in the state space can be obtained as shown in Eq.

(2-14). The coefficient matrices of Eq. (2-14) are derived as follows:

The system matrix

$$\boldsymbol{A}=\begin{bmatrix} A_{11} & 0 & A_{13} & 0 \\ 0 & A_{22} & 0 & A_{24} \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \tag{A-14}$$

where

$$A_{11}=\frac{-2cl^2}{M\left[(l-l_r)^2+h^2\right]}, \quad A_{13}=\frac{Mgh-2kl^2}{M\left[(l-l_r)^2+h^2\right]}, \quad A_{22}=\frac{-cw^2}{2M\left(e^2+h^2\right)}, \quad \text{and}$$

$$A_{24}=\frac{Mgh-kw^2}{2M\left(e^2+h^2\right)}.$$

The control matrix

$$\boldsymbol{B}=\begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \tag{A-15}$$

where

$$B_{11}=\frac{-hd\left[16ew+8Md^2\left(4e^2+4l_r^2+w^2M_w\right)-16Med^2w\left(1+M_w\right)-32e^2\right]}{4\left[(l-l_r)^2+h_G{}^2\right]\left[M^2d^4\left(4e^2+4l_r^4+w^2M_w\right)\left(1+M_w\right)-4e^2\right]},$$

$$B_{12}=\frac{-hd\left[16Med^2w\left(1+M_w\right)-32e^2-16ew\right]}{4\left[(l-l_r)^2+h_G{}^2\right]\left[M^2d^4\left(4e^2+4l_r^4+w^2M_w\right)\left(1+M_w\right)-4e^2\right]},$$

$$B_{21}=\frac{-hdl_r\left[8Md^2w\left(1+M_w\right)+16e\right]}{2\left(e^2+h^2\right)\left[M^2d^4\left(4e^2+4l_r^4+w^2M_w\right)\left(1+M_w\right)-4e^2\right]},$$

$$B_{22}=\frac{hdl_r\left[8Md^2w\left(1+M_w\right)-16e\right]}{2\left(e^2+h^2\right)\left[M^2d^4\left(4e^2+4l_r^4+w^2M_w\right)\left(1+M_w\right)-4e^2\right]}.$$

The output matrix

$$\boldsymbol{C}=\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{A-16}$$

Appendix B: Dynamic Model for Problem Two

Similarly, after deriving the energy components, the coefficient matrices of Eq. (2-34) are obtained as follows:

The mass matrix

$$\boldsymbol{M}=\begin{bmatrix} M_{11} & M_{12} & \cdots & M_{1N} \\ M_{21} & M_{22} & \cdots & M_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ M_{N1} & M_{N2} & \cdots & M_{NN} \end{bmatrix}, \tag{B-1}$$

where

$$M_{ij}=M_b\frac{\partial \boldsymbol{R}_{c_b}}{\partial q_i}\cdot\frac{\partial \boldsymbol{R}_{c_b}}{\partial q_j}+\frac{\partial \boldsymbol{\omega}_b}{\partial \dot{q}_i}\cdot I_b\frac{\partial \boldsymbol{\omega}_b}{\partial \dot{q}_j}+\sum_{n=1}^{3}\left(M_m^n\frac{\partial \boldsymbol{r}_{c_k/c_b}}{\partial q_i}\cdot\frac{\partial \boldsymbol{r}_{c_k/c_b}}{\partial q_j}+\frac{\partial \boldsymbol{\omega}_m^n}{\partial \dot{q}_i}\cdot I_n\frac{\partial \boldsymbol{\omega}_m^n}{\partial \dot{q}_j}\right)$$
$$+\frac{\partial \boldsymbol{R}_{c_b}}{\partial q_j}\cdot\left(\sum_{n=1}^{3}M_m^n\frac{\partial \boldsymbol{r}_{c_k/c_b}}{\partial q_i}\right)+\frac{\partial \boldsymbol{R}_{c_b}}{\partial q_i}\cdot\left(\sum_{n=1}^{3}M_m^n\frac{\partial \boldsymbol{r}_{c_k/c_b}}{\partial q_j}\right).$$

The nonlinear velocity vector

$$\boldsymbol{C}=\boldsymbol{C}_1\dot{\boldsymbol{q}}+\boldsymbol{C}_2, \tag{B-2}$$

where $\boldsymbol{C}_1=\begin{bmatrix} C_{111} & C_{112} & \cdots & C_{11N} \\ C_{121} & C_{122} & \cdots & C_{12N} \\ \vdots & \vdots & \vdots & \vdots \\ C_{1N1} & C_{1N2} & \cdots & C_{1NN} \end{bmatrix}$, where

$$C_{1ij}=M_b\frac{\partial \boldsymbol{R}_{c_b}}{\partial q_i}\cdot\sum_{s=1}^{N}\frac{\partial^2 \boldsymbol{R}_{c_b}}{\partial q_s\partial q_j}+\frac{\partial \boldsymbol{\omega}_b}{\partial \dot{q}_i}\cdot I_b\frac{\partial \boldsymbol{\omega}_b}{\partial \dot{q}_j}+\boldsymbol{\omega}_b\cdot I_b\frac{\partial^2 \boldsymbol{\omega}_b}{\partial \dot{q}_i\partial q_j}$$
$$+\frac{\partial \boldsymbol{R}_{c_b}}{\partial q_i}\cdot\sum_{n=1}^{3}\left(M_m^n\sum_{s=1}^{N}\frac{\partial^2 \boldsymbol{r}_{c_k/c_b}}{\partial q_s\partial q_j}\cdot\dot{q}_s\right)+\sum_{s=1}^{N}\left(\frac{\partial^2 \boldsymbol{r}_{c_k/c_b}}{\partial q_s\partial q_j}\cdot\dot{q}_s\right)\cdot\sum_{n=1}^{3}\left(M_m^n\frac{\partial \boldsymbol{r}_{c_k/c_b}}{\partial q_i}\right)$$
$$+\sum_{n=1}^{3}\left\{M_m^n\frac{\partial \boldsymbol{r}_{c_k/c_b}}{\partial q_i}\cdot\sum_{s=1}^{N}\left(\frac{\partial^2 \boldsymbol{r}_{c_k/c_b}}{\partial q_s\partial q_j}\cdot\dot{q}_s\right)+\frac{\partial \boldsymbol{\omega}_m^n}{\partial \dot{q}_i}\cdot I_n\frac{\partial \boldsymbol{\omega}_m^n}{\partial q_j}+\boldsymbol{\omega}_m^n\cdot I_n\frac{\partial^2 \boldsymbol{\omega}_m^n}{\partial \dot{q}_i\partial q_j}\right\}$$
$$+c\left(\frac{\partial \boldsymbol{\delta}_L}{\partial q_i}\cdot\frac{\partial \boldsymbol{\delta}_L}{\partial q_j}+\frac{\partial \boldsymbol{\delta}_R}{\partial q_i}\cdot\frac{\partial \boldsymbol{\delta}_R}{\partial q_j}\right).$$

And $\boldsymbol{C}_2 = \begin{bmatrix} C_{21} & C_{22} & \cdots & C_{2N} \end{bmatrix}^T$, where

$$C_{2i} = -\left(\boldsymbol{\omega}_b \cdot I_b \frac{\partial \boldsymbol{\omega}_b}{\partial q_i} + \sum_{n=1}^{3} \boldsymbol{\omega}_m^n \cdot I_n \frac{\partial \boldsymbol{\omega}_m^n}{\partial q_i} \right).$$

The gravity vector

$$\boldsymbol{G} = \begin{bmatrix} G_1 & G_2 & \cdots & G_N \end{bmatrix}^T, \tag{B-3}$$

where $G_i = k\left(\boldsymbol{\delta}_L \cdot \frac{\partial \boldsymbol{\delta}_L}{\partial q_i} + \boldsymbol{\delta}_R \cdot \frac{\partial \boldsymbol{\delta}_R}{\partial q_i} \right) + M_b g \frac{\partial \boldsymbol{R}_{c_b}}{\partial q_i} + g \sum_{n=1}^{3} M_m^n \left(\frac{\partial \boldsymbol{R}_{c_b}}{\partial q_i} + \frac{\partial \boldsymbol{r}_{c_k/c_b}}{\partial q_i} \right).$

Chapter 3 Tracking Control Considering Interaction Effects and Uncertainties

3.1 Introduction

Since the application of mechanical automation to agricultural, industrial, and service sectors, researchers have long been fascinated by the development of autonomous mobile robots, especially the wheeled mobile robot. In recent years, there has been considerable interest in the design of advanced systems for the motion control of robots [51−54]. The problem of motion control addressed in the literature can be classified into three groups: 1) point stabilization, where the goal is to stabilize a robot at a given target point with a desired orientation; 2) path planning, where the robot is required to converge to and follow a desired path without any temporal specifications; and 3) trajectory tracking, where the robot is required to track a time parameterized reference. The point stabilization of a wheeled robot is a challenging problem in the field of motion control of nonholonomic systems because the systems do not meet the necessary condition of smooth feedback control presented in [55]. In response, a few new control methods and tools have been proposed such as discontinuous feedback laws [56], continuous time-varying control laws [57], and input feed-forward control laws [14].

Based on the abovementioned classification and definitions, the path following problem can be regarded as a special case of the trajectory tracking problem. Several researchers have investigated these problems. Njah [21] employed recursive and active backstepping nonlinear techniques to design control functions for trajectory tracking of the new hyperchaotic Liu system. Yang and Red [58] constructed an on-line Cartesian trajectory control of mechanism along complex curves, and Red [59] presented a dynamic optimal trajectory generator for Cartesian path following. However, their work does not address the problem of simultaneously tracking the

trajectory of the platform and manipulator, i.e., they neglected the interaction between the platform and manipulator. Furthermore, the issues of modeling uncertainties such as friction and external disturbances were not considered.

The wheeled mobile manipulator (i.e., wheeled mobile robot with a manipulator) is extensively used in tasks such as welding, paint spraying, and accurate positioning systems. In these practical applications, when following a given path or tracking the obtained optimal trajectory, the interaction between the platform and manipulator cannot be neglected because it inevitably causes sliding [60, 61]. Yamamoto and Yun [1, 62] discussed the effect of the dynamic interaction between a mobile platform and a mobile manipulator on the performance of a task. Meghdari et al. [9] investigated the dynamic interaction between a one-degree-of-freedom (DOF) manipulator and the vehicle of a mobile manipulator of a planar robotics system. Moosavian and Papadopoulos [10] utilized the direct path method (DPM) to derive the dynamics of a space robotic system equipped with multiple arms, and showed that the DPM concept requires fewer computations compared to other methods. In this study, the DPM concept is extended to obtain an explicit dynamic model of the wheeled mobile manipulator.

Most of the abovementioned results consider only the nominal model of the system. Only a few authors have addressed the control problems while taking into consideration of the modeling uncertainties [63, 64]. In reality, robotic manipulator is inevitably subject to structured and unstructured uncertainties, which are very difficult to accurately model. Motivated by these considerations, the goal of this chapter is to use adaptive controllers for tracking the trajectory of a three-link mobile manipulator in the presence of modeling uncertainties. In particular, an adaptive fuzzy controller is designed by using the backstepping approach to track the trajectory of a mobile manipulator. To accurately track the trajectory, a fuzzy compensator is incorporated to counteract the modeling uncertainties such as friction and external disturbances [65−67]. In addition, a robust term (RT) is introduced to reduce approximation errors and ensure system stability.

The rest of this chapter is organized as follows. In Section 3.2, the kinematic and dynamic models of the mobile manipulator system are presented, and the DPM

concept is introduced. In Section 3.3, the kinematic models with and without sliding between the wheels and floor are used to design backstepping controllers for tracking the trajectory of a robot. In Section 3.4, the dynamic model is used to design an adaptive fuzzy controller with compensator in order to simultaneously track the desired trajectory of the mobile platform and manipulator. Finally, in Section 3.5 concluding remarks are presented.

3.2 System Model

The main configuration of the mobile manipulator is a modular manipulator mounted on a mobile platform. Figure 3-1 shows a prototype of the mobile manipulator. Two rear driving wheels and one front caster wheel support the platform. Independent motors actuate the two rear wheels, and the caster wheel is free to attain any orientation according to the motion of the robot. In modeling the robot, it is assumed that the wheels, platform, and each link of the manipulator are rigid.

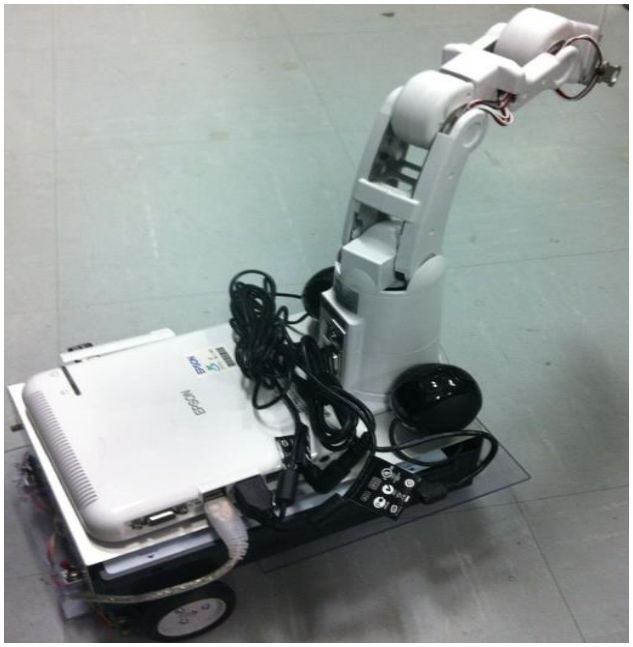


Fig. 3-1 Mobile manipulator system.

3.2.1 Kinematic Model

The objective of the trajectory tracking problem is to track a time-parameterized reference. Figure 3-2 shows the configuration of the robotic platform, where ϕ is the heading angle. The coordinate systems are defined as follows: O_g-$X_gY_gZ_g$ is an inertial base frame (global frame) fixed on the motion plane and O-XYZ is a local frame fixed on the mobile platform. The configuration of the path can be defined by the DOF of the robotic platform:

$$P = \begin{bmatrix} x, & y, & \phi \end{bmatrix}^T . \tag{3-1}$$

Therefore, considering the nonholonomic constraints of a wheeled robot [68], the robot kinematics can be described by

$$\dot{P} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos\phi & 0 \\ \sin\phi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}, \tag{3-2}$$

where v and ω are the linear and angular velocities with respect to the inertial base frame, respectively.

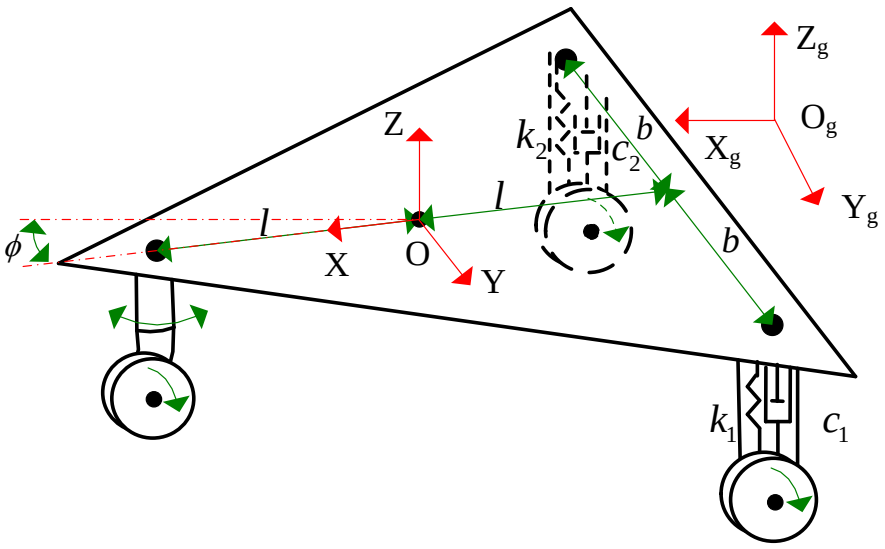


Fig. 3-2 Configuration of the robotic platform.

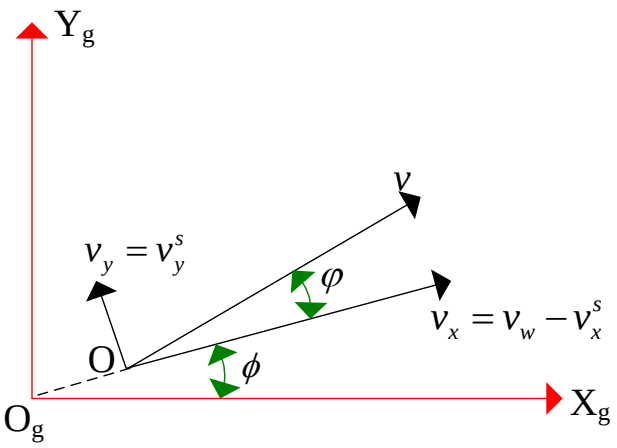


Fig. 3-3 Graphical representation of the parameters of robotic platform.

Consider a desired trajectory with velocities v_d and ω_d, and path configuration $P_d = [x_d, \quad y_d, \quad \phi_d]^T$. In the local coordinate system with respect to the mobile robot system, the configuration error $P_e = [x_e, \quad y_e, \quad \phi_e]^T$ can be presented by

$$P_e = \begin{bmatrix} x_e \\ y_e \\ \phi_e \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_d - x \\ y_d - y \\ \phi_d - \phi \end{bmatrix}. \tag{3-3}$$

By differentiating Eq. (3-3) and substituting Eq. (3-2) into the result, it can be obtained as

$$\begin{cases} \dot{x}_e = (v_d \cos\phi_d - v\cos\phi)\cos\phi - \omega(x_d - x)\sin\phi \\ \qquad + (v_d \sin\phi_d - v\sin\phi)\sin\phi + \omega(y_d - y)\cos\phi, \\ \dot{y}_e = -(v_d \cos\phi_d - v\cos\phi)\sin\phi - \omega(x_d - x)\cos\phi \\ \qquad + (v_d \sin\phi_d - v\sin\phi)\cos\phi - \omega(y_d - y)\sin\phi, \\ \dot{\phi}_e = \omega_d - \omega. \end{cases} \tag{3-4}$$

Considering Eq. (3-3), the time derivative of the configuration error for the robot is

$$\dot{P}_e = \begin{bmatrix} \dot{x}_e \\ \dot{y}_e \\ \dot{\phi}_e \end{bmatrix} = \begin{bmatrix} \omega y_e - v + v_d \cos\phi_e \\ -\omega x_e + v_d \sin\phi_e \\ \omega_d - \omega \end{bmatrix}. \tag{3-5}$$

However, when the robot moves on slippery ground or swerves, longitudinal and lateral sliding inevitably occurs [61, 69]. In this case, because Eq. (3-2) is no longer valid, a non-ideal kinematic model must be considered. The violation of the pure

rolling constraints is described by introducing four parameters: the longitudinal sliding velocity v_x^s, lateral sliding velocity v_y^s, linear velocity of wheel v_w, and bias of the side-sliding angle φ. Figure 3-3 shows a graphical representation of these parameters.

In the presence of sliding, the longitudinal and lateral velocities satisfy the following constraints:

$$\dot{P} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos(\phi+\varphi) & 0 \\ \sin(\phi+\varphi) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}, \tag{3-6}$$

$$\begin{cases} \dot{x}_d = v_d \cos\phi, \\ \dot{y}_d = v_d \sin\phi, \end{cases} \tag{3-7}$$

where

$$v = \sqrt{v_x^2 + v_y^2}, \tag{3-8}$$

$$\begin{cases} v_x = v_w - v_x^s, \\ v_y = v_y^s, \end{cases} \tag{3-9}$$

and

$$\varphi = \arctan\left(\frac{v_y}{v_x}\right). \tag{3-10}$$

Introducing Eqs. (3-6) and (3-7) into Eq. (3-4) yields

$$\begin{cases} \dot{x}_e = (v_d \cos\phi_d - v\cos(\phi+\varphi))\cos\phi - \omega(x_d - x)\sin\phi \\ \quad + (v_d \sin\phi_d - v\sin(\phi+\varphi))\sin\phi + \omega(y_d - y)\cos\phi, \\ \dot{y}_e = -(v_d \cos\phi_d - v\cos(\phi+\varphi))\sin\phi - \omega(x_d - x)\cos\phi \\ \quad + (v_d \sin\phi_d - v\sin(\phi+\varphi))\cos\phi - \omega(y_d - y)\sin\phi, \\ \dot{\phi}_e = \omega_d - \omega. \end{cases} \tag{3-11}$$

After performing simple algebra deduction it can be obtained as

$$\begin{cases} \dot{x}_e = -v\cos(\varphi) + v_d \cos\phi_e + \omega y_e, \\ \dot{y}_e = -v\sin(\varphi) + v_d \sin\phi_e - \omega x_e, \\ \dot{\phi}_e = \omega_d - \omega. \end{cases} \tag{3-12}$$

Due to Eqs. (3-8) and (3-10), it is obvious that

$$\begin{cases} \dot{x}_e = -v_x + v_d \cos\phi_e + \omega y_e, \\ \dot{y}_e = -v_y + v_d \sin\phi_e - \omega x_e, \\ \dot{\phi}_e = \omega_d - \omega. \end{cases} \tag{3-13}$$

Therefore, when sliding is considered, the time derivative of the configuration error can be written as

$$\dot{P}_e = \begin{bmatrix} \dot{x}_e \\ \dot{y}_e \\ \dot{\phi}_e \end{bmatrix} = \begin{bmatrix} \omega y_e - v_x + v_d \cos\phi_e \\ -\omega x_e + v_d \sin\phi_e - v_y \\ \omega_d - \omega \end{bmatrix}. \tag{3-14}$$

In the absence of sliding, it is obvious that $v_x = v_w = v$ and Eq. (3-14) is equal to Eq. (3-5).

3.2.2 Dynamic Model

The DPM is used to obtain a dynamic model that accounts for the interaction between the platform and manipulator as described in Chapter 2.

Using this approach, the dynamic model can be obtained as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\dot{\boldsymbol{q}},\boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{Q}. \tag{3-15}$$

where $\boldsymbol{M}(\boldsymbol{q})$, $\boldsymbol{C}(\dot{\boldsymbol{q}},\boldsymbol{q})$, $\boldsymbol{G}(\boldsymbol{q})$, and $\boldsymbol{Q}$ denote the inertial matrix, Coriolis force, gravity component, and torque matrix, respectively.

3.3 Backstepping Control Based on Kinematics

It is well known that smooth static-state feedback laws cannot be applied to stabilize nonholonomic systems [52]. At present, the widely used classical control methods are employed in cascaded form to control nonholonomic systems. Among them, the chained system theory and backstepping approach are the most important nonlinear control methods with numerous applications. The backstepping design

procedure is systematic and simple because it breaks down complex nonlinear systems into smaller subsystems, and designs virtual-input partial Lyapunov functions onto the actual controller by "back stepping" through the system and reassembling the complex system from its component subsystems. Backstepping control has two advantages: first, it is very effective for complex dynamic systems such as mobile manipulator systems because it uses recursive methodology to design the control law; and second, it is useful for uncertain systems since it designs the control law based on Lyapunov stability theory. This is why the backstepping approach is used to design the controllers in this chapter.

3.3.1 Controller Design

Consider the ideal kinematic model described in Eq. (3-5). Because the desired configuration variables (x_d, y_d) and ϕ_d are not independent of each other, the desired trajectory is defined by

$$\begin{cases} \dot{x}_d = v_d \cos\phi_d, \\ \dot{y}_d = v_d \sin\phi_d. \end{cases} \tag{3-16}$$

Step 1: Introduce the virtual-input angle ϑ. Then, according to Eq. (3-16)

$$\begin{cases} \dot{x} = v\cos\vartheta, \\ \dot{y} = v\sin\vartheta. \end{cases} \tag{3-17}$$

Next, the Lyapunov function candidate for the first step is chosen as

$$V_1 = \frac{1}{2}x_e^2 + \frac{1}{2}y_e^2. \tag{3-18}$$

Then, the derivative of V_1 is

$$\dot{V}_1 = x_e(\dot{x}_d - v\cos\vartheta) + y_e(\dot{y}_d - v\sin\vartheta). \tag{3-19}$$

According to the Lyapunov theorem of stability, it can be set as

$$v\cos\vartheta = \dot{x}_d + c_1 x_e, \tag{3-20}$$

$$v\sin\vartheta = \dot{y}_d + c_2 y_e, \tag{3-21}$$

where $c_1 > 0$ and $c_2 > 0$. After substitution, Eq. (3-19) can be rewritten as

$$\dot{V}_1 = -c_1 x_e^2 - c_2 y_e^2 , \tag{3-22}$$

where $\dot{V}_1 < 0$ satisfies the Lyapunov theorem of stability. Therefore, the control laws of the linear velocity and virtual-input angle are

$$v = \sqrt{\left(\dot{x}_d + c_1 x_e\right)^2 + \left(\dot{y}_d + c_2 y_e\right)^2} , \tag{3-23}$$

$$\vartheta = \arctan \frac{\dot{y}_d + c_2 y_e}{\dot{x}_d + c_1 x_e}. \tag{3-24}$$

If $x_e = 0$ and $y_e = 0$, the virtual-input angle becomes $\vartheta = \arctan\left(\dot{y}_d / \dot{x}_d\right) = \phi_d$. Therefore, during the second step, it must be ensured that the heading angle ϕ tracks the virtual-input angle ϑ.

Step 2: Set $e = \vartheta - \phi$, and consider the following Lyapunov function

$$V_2 = V_1 + \frac{1}{2} e^2 . \tag{3-25}$$

Thus, the derivative of V_2 is

$$\dot{V}_2 = -c_1 x_e^2 - c_2 y_e^2 + e\left(\dot{\vartheta} - \omega\right). \tag{3-26}$$

Similarly, according to the Lyapunov theorem of stability, the control law of the angular velocity is defined as

$$\omega = \dot{\vartheta} + c_3 e , \tag{3-27}$$

where $c_3 > 0$. After substitution, Eq. (3-26) can be rewritten as

$$\dot{V}_2 = -c_1 x_e^2 - c_2 y_e^2 - c_3 e^2 , \tag{3-28}$$

where $\dot{V}_2 \le 0$ satisfies the Lyapunov theorem of stability.

Similarly, considering the non-ideal kinematic model described in Eq. (3-14), when the mobile platform does not purely roll but moves in the presence of sliding, the resulting control laws are

$$v = \sqrt{\left(\dot{x}_d + c_1 x_e\right)^2 + \left(\dot{y}_d + c_2 y_e\right)^2} - v_x^s , \tag{3-29}$$

$$\vartheta = \arctan \frac{\dot{y}_d + c_2 y_e}{\dot{x}_d + c_1 x_e} - \psi_b , \tag{3-30}$$

$$\omega = \dot{\vartheta} + c_3 e. \tag{3-31}$$

3.3.2 Results and Discussion

For tracking a desired trajectory, the desired trajectory is designed as "∞" and is described by $x_d = \cos 0.5t$ and $y_d = \sin t$. Moreover, the initial configuration of the robotic platform is given by $P_s = [1 \quad 1 \quad 0]$, which implies that the robot starts from point [1, 1]; the initial heading angle is zero. Table 3-1 shows parameters of a mobile manipulator for simulation. The parameters of the controller are set as $c_1 = 10$, $c_2 = 10$, and $c_3 = 50$, and the control laws are defined in Subsection 3.3.1.

Figure 3-4 displays simulation result in the absence of sliding. As shown in Fig. 3-4, the robot accurately follows the desired trajectory. Furthermore, the accuracy of trajectory tracking in the X and Y directions is confirmed by Fig. 3-5. Figure 3-6 depicts the time histories of the control inputs for the robot. Because the backstepping controller is designed on the basis of the kinematic model, the control inputs are the linear and angular velocities that result from the rotations of the wheels.

Table 3-1 Parameters of the mobile manipulator.

Symbol	Parameter	Value
M_b	mass of the platform	20.56 kg
$M_m{}^2 = M_m{}^3$	mass of the 2nd and 3rd links of the manipulator	2.20 kg
l	half of the length of the wheelbase	0.23 m
$l_m{}^1 = l_m{}^2 = l_m{}^3$	length of each link of the manipulator	0.20 m
b	half of the distance between the two rear wheels	0.23 m
I_b	moment of inertia of the platform	0.37 kg·m^2
$I_2 = I_3$	moment of inertia of the 2nd and 3rd links	0.06 kg·m^2
$k_1 = k_2$	stiffness coefficient of left and right suspensions	2.50 kN/m
$c_1 = c_2$	damping property of left and right suspensions	5.26 N/m/s

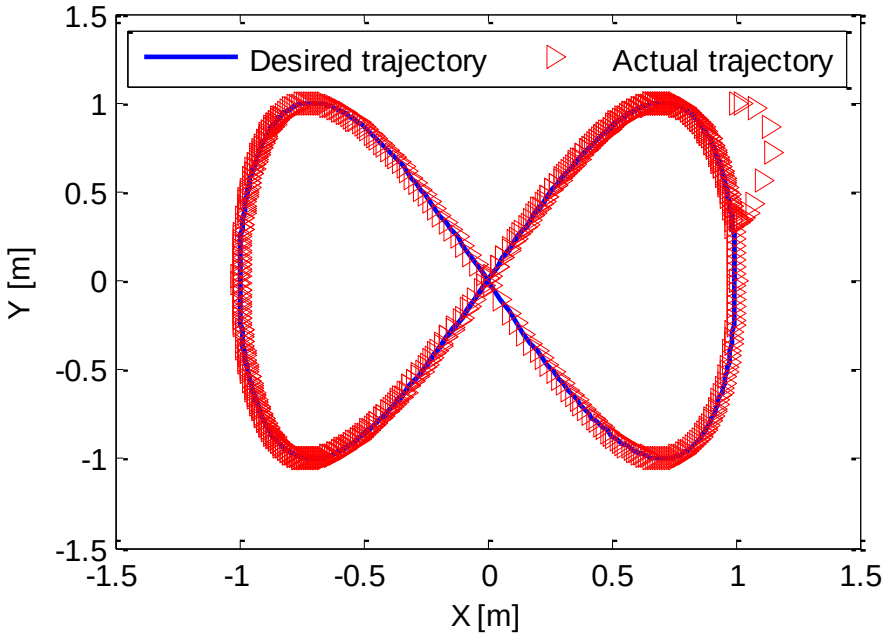


Fig. 3-4 Trajectory tracking of the robot in the absence of sliding.

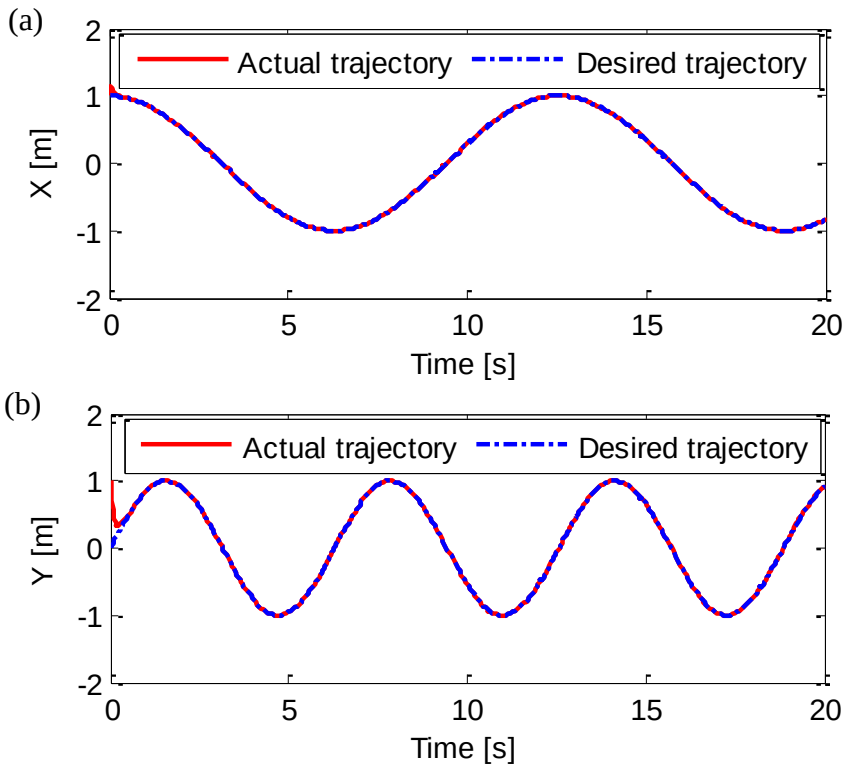


Fig. 3-5 Trajectory tracking of the robot.

(a) In X direction, and (b) In Y direction.

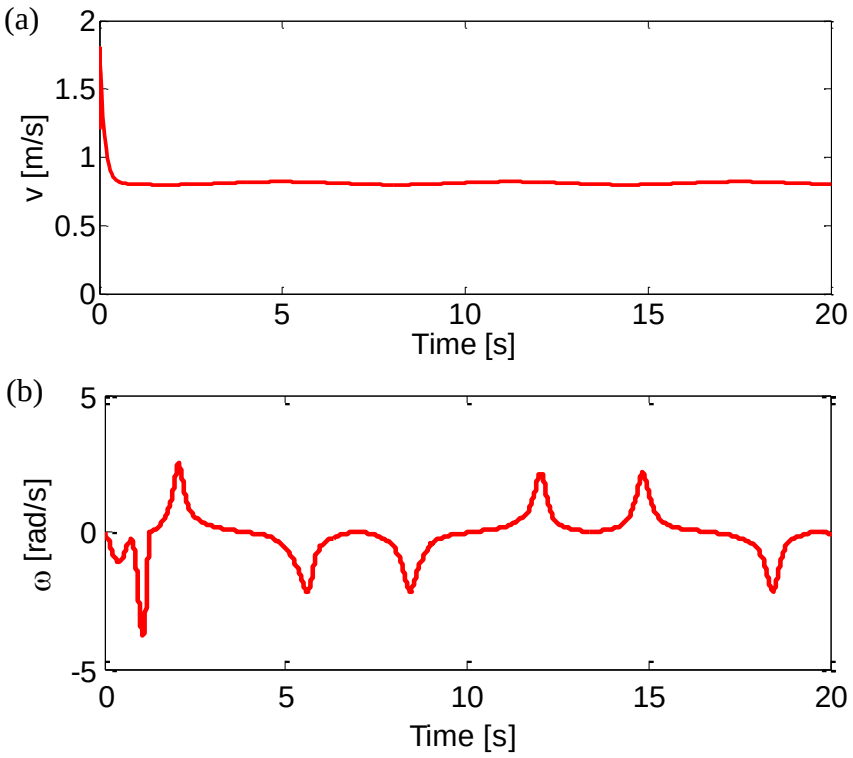


Fig. 3-6 Control inputs for the robot.

(a) Linear velocity, and (b) Angular velocity.

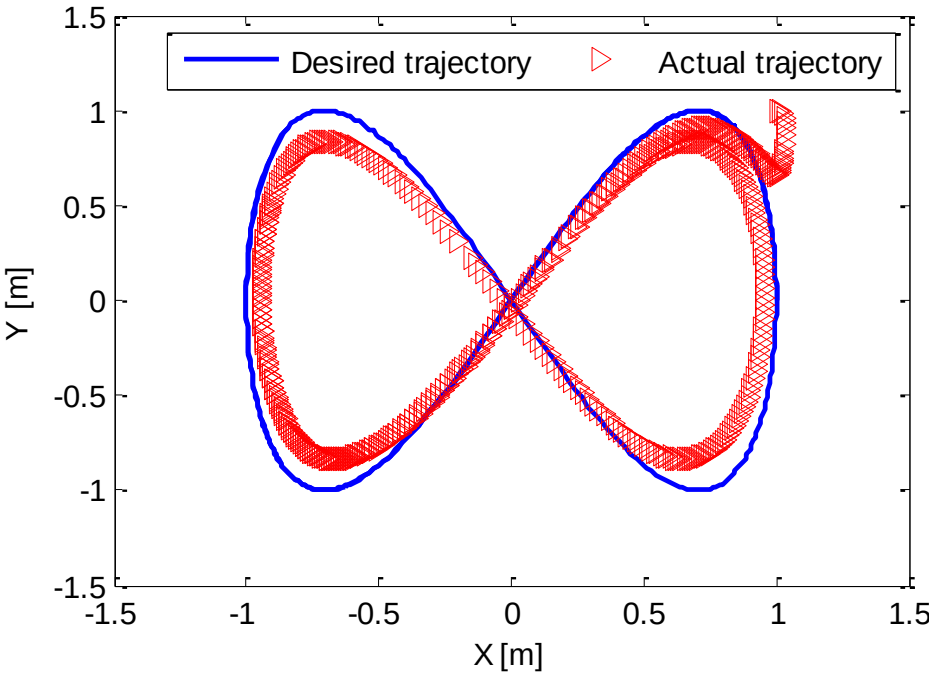


Fig. 3-7 Trajectory tracking of the robot in the presence of sliding.

However, when the sliding is considered, it becomes difficult to track the desired trajectory using the backstepping controller at the kinematic level. Figure 3-7 shows the results of trajectory tracking when sliding occurs. The violation of the pure rolling is described by selecting the sliding parameters as $v_x^s = 0.01v_w$ and $v_y^s = 0.01\omega$. As shown in Fig. 3-7, the robot cannot track the desired trajectory as accurately as in Fig. 3-4, and the system tracking errors become large and even unstable. In the next section, this problem at the dynamic level is studied by considering the interaction between the platform and manipulator as well as the modeling uncertainties such as friction and disturbances. To this end, an adaptive backstepping control with a fuzzy compensator is designed.

3.4 Adaptive Backstepping Control with Fuzzy Compensator Based on Dynamics

At the kinematic level, the immediate cause of sliding is velocity deviation. However, at the dynamic level, the root causes of sliding are friction and the interaction between the platform and manipulator. Therefore, in this section, an adaptive backstepping controller that considers the effect of this dynamic interaction is designed. Then a fuzzy compensator is added to counteract the modeling uncertainties such as friction and disturbances.

3.4.1 Controller Design

Similar to Section 3.3, consider the dynamic model rewritten as

$$\boldsymbol{M}(\boldsymbol{q})\cdot\ddot{\boldsymbol{q}} + \boldsymbol{C}(\dot{\boldsymbol{q}},\boldsymbol{q})\cdot\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) + \boldsymbol{F}(\dot{\boldsymbol{q}},\boldsymbol{q}) = \boldsymbol{Q}, \tag{3-32}$$

where $\boldsymbol{F}(\dot{\boldsymbol{q}},\boldsymbol{q})$ denotes the modeling uncertainties including friction and external disturbances. To compensate for $\boldsymbol{F}(\dot{\boldsymbol{q}},\boldsymbol{q})$, the sliding surface $\boldsymbol{s}$ = 0 is chosen as the hyper plane [70]

$$\boldsymbol{s} = \dot{\tilde{\boldsymbol{q}}} + \Lambda\tilde{\boldsymbol{q}}, \tag{3-33}$$

where $\boldsymbol{\Lambda}$ is a positive-definite matrix with eigenvalues strictly in the right half of the complex plane, and $\tilde{\boldsymbol{q}}$ is the tracking error vector. This study defines

$$\dot{\boldsymbol{q}}_r = \dot{\boldsymbol{q}}_d - \boldsymbol{\Lambda}\tilde{\boldsymbol{q}}, \tag{3-34}$$

where $\dot{\boldsymbol{q}}_d$ is the derivative of the desired trajectory.

The Lyapunov function candidate is chosen as

$$V(t) = \frac{1}{2}\left(\boldsymbol{s}^T \boldsymbol{M}\boldsymbol{s} + \sum_{i=1}^{n} \tilde{\boldsymbol{\Theta}}_i^T \boldsymbol{\Gamma}_i \tilde{\boldsymbol{\Theta}}_i \right), \tag{3-35}$$

where $\tilde{\boldsymbol{\Theta}}_i = \boldsymbol{\Theta}_i^* - \boldsymbol{\Theta}_i$ is the *i*th column vector of the optimal parameter matrix, $\boldsymbol{\Gamma}_i$ is a strictly positive real constant. Differentiating $V(t)$ with respect to time yields

$$\dot{V}(t) = -\boldsymbol{s}^T \left(\boldsymbol{M}\ddot{\boldsymbol{q}}_r + \boldsymbol{C}\dot{\boldsymbol{q}}_r + \boldsymbol{G} + \boldsymbol{F} - \boldsymbol{Q} \right) + \sum_{i=1}^{n} \tilde{\boldsymbol{\Theta}}_i^T \boldsymbol{\Gamma}_i \dot{\tilde{\boldsymbol{\Theta}}}_i, \tag{3-36}$$

where $\boldsymbol{F}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is a completely unknown nonlinear function vector. Therefore, $\boldsymbol{F}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is replaced by a multi-input multi-output (MIMO) fuzzy logic system [71] of the form $\hat{\boldsymbol{F}}(\boldsymbol{q}, \dot{\boldsymbol{q}} | \boldsymbol{\Theta})$. The adaptive control law (ACL) is defined as

$$\boldsymbol{Q} = \boldsymbol{M}\ddot{\boldsymbol{q}}_r + \boldsymbol{C}\dot{\boldsymbol{q}}_r + \boldsymbol{G} + \hat{\boldsymbol{F}}(\boldsymbol{q}, \dot{\boldsymbol{q}} | \boldsymbol{\Theta}) - \boldsymbol{K}_D \boldsymbol{s}, \tag{3-37}$$

where $\boldsymbol{K}_D = \mathrm{diag}(K_i)$ and $K_i > 0$ for $i = 1, 2, \cdots, n$, and

$$\hat{\boldsymbol{F}}(\boldsymbol{q}, \dot{\boldsymbol{q}} | \boldsymbol{\Theta}) = \begin{bmatrix} \hat{F}_1(\boldsymbol{q}, \dot{\boldsymbol{q}} | \boldsymbol{\Theta}_1) \\ \hat{F}_2(\boldsymbol{q}, \dot{\boldsymbol{q}} | \boldsymbol{\Theta}_2) \\ \vdots \\ \hat{F}_n(\boldsymbol{q}, \dot{\boldsymbol{q}} | \boldsymbol{\Theta}_n) \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Theta}_1^T \boldsymbol{\xi}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \\ \boldsymbol{\Theta}_2^T \boldsymbol{\xi}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \\ \vdots \\ \boldsymbol{\Theta}_n^T \boldsymbol{\xi}(\boldsymbol{q}, \dot{\boldsymbol{q}}) \end{bmatrix}. \tag{3-38}$$

By selecting the optimal parameter matrix of the fuzzy logic system as $\boldsymbol{\Theta}^*$, the minimum approximation error vector can be defined as

$$\boldsymbol{w} = \boldsymbol{F}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \hat{\boldsymbol{F}}(\boldsymbol{q}, \dot{\boldsymbol{q}} | \boldsymbol{\Theta}^*). \tag{3-39}$$

By substituting Eq. (3-37) in Eq. (3-36), $\dot{V}(t)$ can be rewritten as

$$
\begin{aligned}
\dot{V}(t) &= -\boldsymbol{s}^T\left(\boldsymbol{F}(\boldsymbol{q},\dot{\boldsymbol{q}}) - \hat{\boldsymbol{F}}(\boldsymbol{q},\dot{\boldsymbol{q}}|\boldsymbol{\Theta}) + \boldsymbol{K}_D\boldsymbol{s}\right) + \sum_{i=1}^{n}\tilde{\boldsymbol{\Theta}}_i^T\boldsymbol{\Gamma}_i\dot{\tilde{\boldsymbol{\Theta}}}_i \\
&= -\boldsymbol{s}^T\left(\boldsymbol{F}(\boldsymbol{q},\dot{\boldsymbol{q}}) - \hat{\boldsymbol{F}}(\boldsymbol{q},\dot{\boldsymbol{q}}|\boldsymbol{\Theta}) + \hat{\boldsymbol{F}}(\boldsymbol{q},\dot{\boldsymbol{q}}|\boldsymbol{\Theta}^*) - \hat{\boldsymbol{F}}(\boldsymbol{q},\dot{\boldsymbol{q}}|\boldsymbol{\Theta}^*) + \boldsymbol{K}_D\boldsymbol{s}\right) + \sum_{i=1}^{n}\tilde{\boldsymbol{\Theta}}_i^T\boldsymbol{\Gamma}_i\dot{\tilde{\boldsymbol{\Theta}}}_i \\
&= -\boldsymbol{s}^T\left(\tilde{\boldsymbol{\Theta}}^T\boldsymbol{\xi}(\boldsymbol{q},\dot{\boldsymbol{q}}) + \boldsymbol{w} + \boldsymbol{K}_D\boldsymbol{s}\right) + \sum_{i=1}^{n}\tilde{\boldsymbol{\Theta}}_i^T\boldsymbol{\Gamma}_i\dot{\tilde{\boldsymbol{\Theta}}}_i \\
&= -\boldsymbol{s}^T\boldsymbol{K}_D\boldsymbol{s} - \boldsymbol{s}^T\boldsymbol{w} + \sum_{i=1}^{n}\left(\tilde{\boldsymbol{\Theta}}_i^T\boldsymbol{\Gamma}_i\dot{\tilde{\boldsymbol{\Theta}}}_i - s_i\tilde{\boldsymbol{\Theta}}_i^T\boldsymbol{\xi}(\boldsymbol{q},\dot{\boldsymbol{q}})\right),
\end{aligned}
\tag{3-40}
$$

where $\tilde{\boldsymbol{\Theta}} = \boldsymbol{\Theta}^* - \boldsymbol{\Theta}$, $\boldsymbol{\xi}(\boldsymbol{q},\dot{\boldsymbol{q}})$ is a fuzzy basis function. Thus, the adaptive laws are

$$
\dot{\tilde{\boldsymbol{\Theta}}}_i = \boldsymbol{\Gamma}_i^{-1}s_i\boldsymbol{\xi}(\boldsymbol{q},\dot{\boldsymbol{q}}), \qquad i = 1,2,\cdots,n. \tag{3-41}
$$

Then

$$
\dot{V}(t) = -\boldsymbol{s}^T\boldsymbol{K}_D\boldsymbol{s} - \boldsymbol{s}^T\boldsymbol{w} \le 0. \tag{3-42}
$$

Equation (3-42) satisfies the Lyapunov theorem of stability.

Because Eq. (3-40) contains the term "$\boldsymbol{s}^T\boldsymbol{w}$", which is of the order of the minimum approximation error, from the universal approximation theorem [72], $\boldsymbol{w}$ is expected to be very small. Therefore, to reduce approximation errors and ensure system stability, a robust term (RT) is added to the ACL defined in Eq. (3-37). In particular, the ACL with RT (ACL-RT) is defined as

$$
\boldsymbol{Q} = \boldsymbol{M}\ddot{\boldsymbol{q}}_r + \boldsymbol{C}\dot{\boldsymbol{q}}_r + \boldsymbol{G} + \hat{\boldsymbol{F}}(\boldsymbol{q},\dot{\boldsymbol{q}}|\boldsymbol{\Theta}) - \boldsymbol{K}_D\boldsymbol{s} - \boldsymbol{W}\,\mathrm{sgn}(\boldsymbol{s}), \tag{3-43}
$$

where $\boldsymbol{W} = \mathrm{diag}[\,w_{M_1} \quad w_{M_2} \quad \cdots \quad w_{M_n}\,]$ and $w_{M_i} \ge |w_i|$ for $i = 1, 2, \cdots, n$.

Substituting Eq. (3-43) in Eq. (3-36), $\dot{V}(t)$ can be rewritten as

$$
\dot{V}(t) = -\boldsymbol{s}^T\boldsymbol{K}_D\boldsymbol{s} \le 0. \tag{3-44}
$$

The structure of the proposed control scheme is shown in Fig. 3-8.

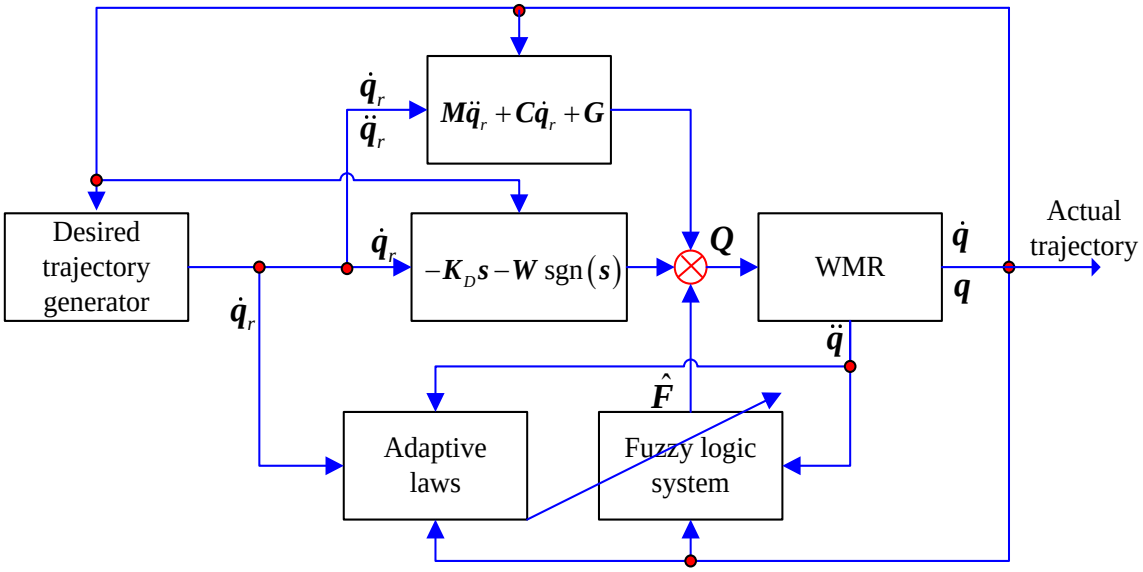


Fig. 3-8 Structure of the control scheme.

3.4.2 Results and Discussion

The mobile manipulator system as shown in Fig. 3-1 is considered again. For illustration purpose and clarity, the dynamics of the DC motor is ignored because the commercial DC motors are usually equipped with their own motion controller with excellent performance. Moreover, the rotation of the first joint j_0 in the posture shown in Fig. 2-11 is not considered, because joint j_0 is in rotary motion in a horizontal plane. When joint j_0 moves, the center-of-gravity of the manipulator will shift, the robot may unstable or even tip-over [73]. Therefore, in this chapter the manipulator is considered as a two-link mechanism. The parameters of the mobile manipulator for simulation are given in Table 3-1. During simulations, the desired trajectories of the mobile platform and manipulator are tracked simultaneously. To investigate the dynamic interaction between the platform and manipulator, the load carried by the end-effector is changed. To demonstrate the robustness of the controller, the results of the ACL controller with those of the ACL-RT controller are compared. Six different cases shown in Table 3-2 are examined.

Table 3-2 Cases examined.

Case	Load carried by the end-effector	Control law
1	2.00 kg	ACL
2	2.00 kg	ACL-RT
3	0.30 kg	ACL
4	0.30 kg	ACL-RT
5	5.00 kg	ACL
6	5.00 kg	ACL-RT

Considering the dynamic model in Eq. (3-32), the desired trajectories of the joints of the manipulator are set as $\theta_{md}^1 = \theta_{md}^2 = 0.5\sin t$. In applications, the desired trajectory can be obtained by using inverse kinematics [74] or can be generated by an optimal trajectory planner [75, 76]. The desired trajectory of the mobile platform is the same as in Section 3.3. Furthermore, Fig. 3-9 visualizes the desired trajectory of the manipulator and platform in 3D space. The configurations of the robot are depicted every 1.33 s.

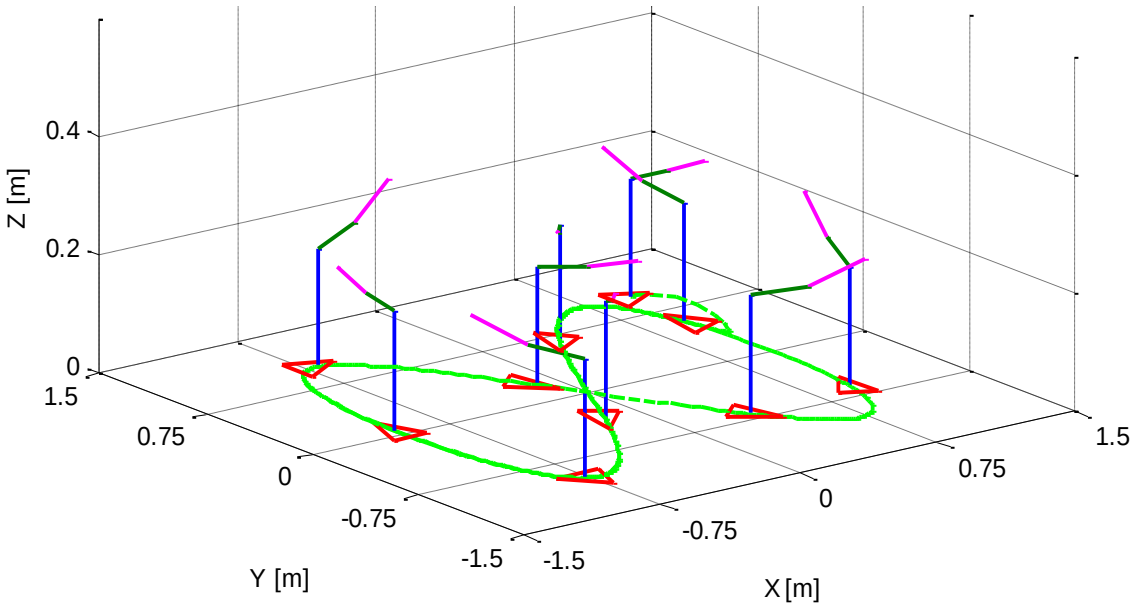


Fig. 3-9 Desired trajectories of manipulator and platform in 3D space.

The membership function of the fuzzy logic system is defined by

$$\mu_{A_i}(x_i) = \exp\left(-\left(\frac{x_i - \bar{x}_i}{\pi/24}\right)^2\right), \tag{3-45}$$

where for $i = 1,2,3,4,5$, $\bar{x}_i = -\pi/6$, $-\pi/12$, 0, $\pi/12$, $\pi/6$, respectively. The fuzzy set A_i is defined by Negative Big (NB), Negative Small (NS), Zero (ZO), Positive Small (PS), and Positive Big (PB).

According to the Coulomb model of friction, the frictions of Joint 1, Joint 2, left wheel, and right wheel can be represented as a vector:

$$\boldsymbol{F}(\dot{\boldsymbol{q}}) = \begin{bmatrix} 5\dot{\theta}_m^1 + 0.2\,\mathrm{sgn}\left(\dot{\theta}_m^1\right) \\ 5\dot{\theta}_m^2 + 0.2\,\mathrm{sgn}\left(\dot{\theta}_m^2\right) \\ 10v_L + 0.5\,\mathrm{sgn}\left(v_L\right) \\ 10v_R + 0.5\,\mathrm{sgn}\left(v_R\right) \end{bmatrix}, \tag{3-46}$$

where v_L and v_R are the linear velocity of the left and right wheels, respectively. By decomposing the velocity, v_L and v_R can be represented as

$$\begin{cases} v_L = v\cos\phi + \dfrac{b}{l}v\sin\phi, \\ v_R = v\cos\phi - \dfrac{b}{l}v\sin\phi. \end{cases} \tag{3-47}$$

The disturbance terms of Joint 1, Joint 2, left and right wheels are defined by

$$\boldsymbol{\tau}_d = \begin{bmatrix} 0 & 0 & 0.05\sin(20t) & 0.1\sin(20t) \end{bmatrix}^T. \tag{3-48}$$

The parameters of the controller are chosen as $\boldsymbol{\Lambda} = 10\boldsymbol{I}$, $\boldsymbol{K}_D = 20\boldsymbol{I}$ (where $\boldsymbol{I}$ is the unit matrix), $\boldsymbol{\Gamma}_1 = \boldsymbol{\Gamma}_2 = \boldsymbol{\Gamma}_3 = \boldsymbol{\Gamma}_4 = 0.0001$, and $W = \mathrm{diag}[0,0,2,2]$.

The results of the manipulator are shown in Figs. 3-10−3-12. To demonstrate the dynamic interaction effect, the disturbance of each joint is not considered, as shown in Eq. (3-48). In the simulations, the dynamic interaction and the robustness of the designed controller are examined by comparison with several cases. Figure 3-10 shows the trajectory tracking of each joint for cases 1 and 2, where the load carried by

the end-effector is 2.00 kg. As shown in Fig. 3-10, when the ACL controller is used for case 1, each joint of the manipulator tracks the desired trajectory with small error. Moreover, when the ACL-RT controller is used for case 2, the error is reduced.

However, as shown in Fig. 3-11, for case 3, where the load carried by the end-effector is 0.30 kg and the ACL controller is used, each joint of the manipulator cannot accurately track the desired trajectory. This is because the effect of dynamic interaction emerges when the equivalent mass ratio of the platform and manipulator exceeds a permissible range. Conversely, for case 4, where the load carried by the end-effector is also 0.30 kg but the ACL-RT controller is used, each joint can track the desired trajectory with small error. This implies that the ACL-RT controller is robust with respect to variable load carried by the end-effector. Robustness is essential and increases the efficiency when the load carried by the end-effector is variable [77].

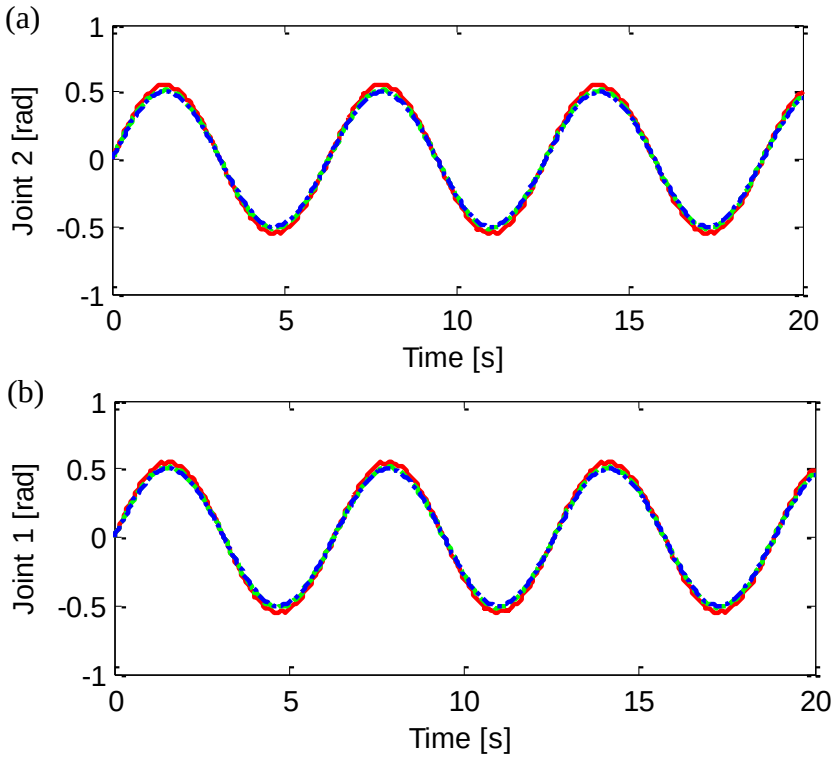


Fig. 3-10 Trajectory tracking of manipulator for cases 1 (—) and 2 (- - -).
(a) Joint 2, and (b) Joint 1, desired trajectory (- · -).

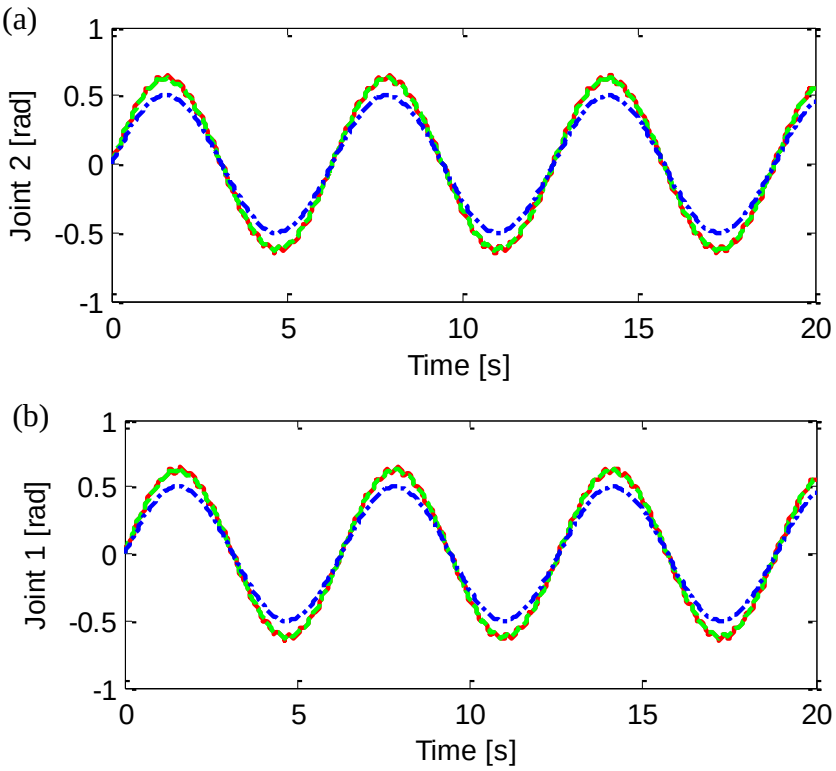


Fig. 3-11 Trajectory tracking of manipulator for cases 3 (—) and 4 (- - -).
(a) Joint 2, and (b) Joint 1, desired trajectory (- · -).

Figure 3-12 depicts the trajectory tracking of each joint when the load carried by the end-effector is 5.00 kg for cases 5 and 6. As indicated in Fig. 3-12, each joint tracks the desired trajectory with high precision when either the ACL or ACL-RT controller is applied. When the platform is fixed or the interaction effect is neglected, it can be asserted that the equivalent mass ratio of the platform and manipulator for cases 5 and 6 is the best one among the cases examined. However, when the mobile platform is subject to the dynamic interaction and the trajectories of both the manipulator and platform are required to accurately track the desired trajectory, this assertion may not be valid. This deduction is confirmed by the following results.

Typically, the nominal model ignores external disturbances. Although this assumption is reasonable in ideal situations, it is still necessary to investigate whether the adaptive control system is robust with respect to external disturbances because rough roads inevitably generate external disturbances in practical situations. Therefore, for controlling the platform, the external disturbance of wheels should be considered. In the following results, the ACL and the ACL-RT controllers are compared.

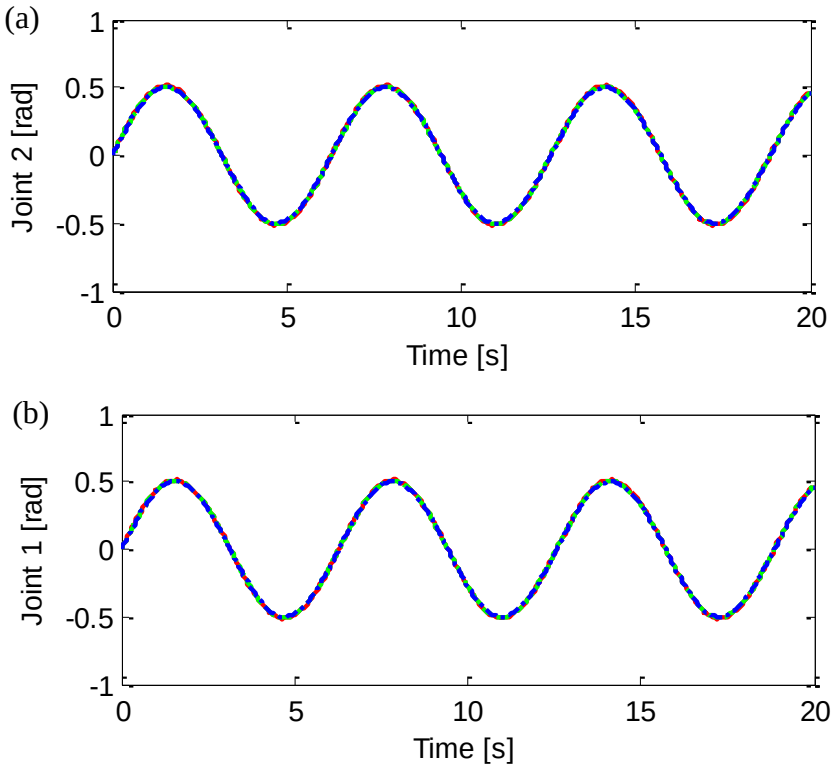


Fig. 3-12 Trajectory tracking of manipulator for cases 5 (—) and 6 (- - -).
(a) Joint 2, and (b) Joint 1, desired trajectory (- · -).

The results of the mobile platform are shown in Fig. 3-13. As opposed to the manipulator, the disturbance of each wheel for the mobile platform is considered. On the basis of the following results, the robustness to the disturbance and the dynamic interaction is discussed. Figure 3-13(a) shows the trajectory tracking of the mobile platform for cases 1 and 2. For case 2, it can be observed that the proposed ACL-RT controller is able to force the robot to converge to the desired trajectory under the influence of friction and external disturbances. Moreover, for case 1, it can be observed that the desired trajectory is tracked with small error when the ACL controller is employed. Figure 3-13 (b) shows the trajectory tracking of the mobile platform for cases 3 and 4. For these cases, when the load carried by the end-effector is 0.30 kg, the mobile platform tracks the desired trajectory with high precision by applying either the ACL or ACL-RT controller. The results for cases 5 and 6 are shown in Fig. 3-13(c). From Fig. 3-13(c), it is clear that the mobile platform cannot accurately track the desired trajectory in either case. Therefore, the above assertion, which states that the equivalent mass ratio of the platform and manipulator for cases 5

and 6 is the best one among the cases examined, is not valid.

As indicated by the results for case 2 shown in Figs. 3-10 and 3-13(a), the proposed ACL-RT controller, which is designed at the dynamic level, accurately tracks the desired trajectory of both the manipulator and platform. This result is obtained because the application of the fuzzy compensator and RT effectively counteracts the friction and disturbances. Furthermore, when the equivalent mass ratio of the platform and manipulator is within a tolerable range, the RT is capable of reducing the influence of the dynamic interaction, as shown in Figs. 3-10, 3-12, 3-13(a), and 3-13(b). The comparison of Fig. 3-11 with Fig. 3-13(b) or of Fig. 3-12 with Fig. 3-13(c) reveals that the improved performance in the trajectory tracking of the manipulator comes at the cost of deteriorated performance in the trajectory tracking of the platform. This deterioration is expected because the dynamic interaction effect between the manipulator and platform is considered.

However, when the equivalent mass ratio of the platform and manipulator exceeds the tolerable range, the RT in the ACL-RT controller becomes ineffective. As shown in Fig. 3-13(c), the trajectory tracking of the platform is not accurate because there exists a specific amount of deviation. However, the manipulator tracks the desired trajectory with high precision. Further, if a load carried by the end-effector is greater than 5.00 kg, the robot becomes unstable or liable to tip-over [77].

On the basis of the above discussion, when the dynamic interaction is considered in case 2, the ACL-RT controller achieves the best performance. For case 2, Fig. 3-14 shows the friction and its compensation for each joint of the manipulator. From Fig. 3-14, it is obvious that the fuzzy compensator provides the convergent estimation of the friction. The time histories of the control input torque for each joint are shown in Fig. 3-15.

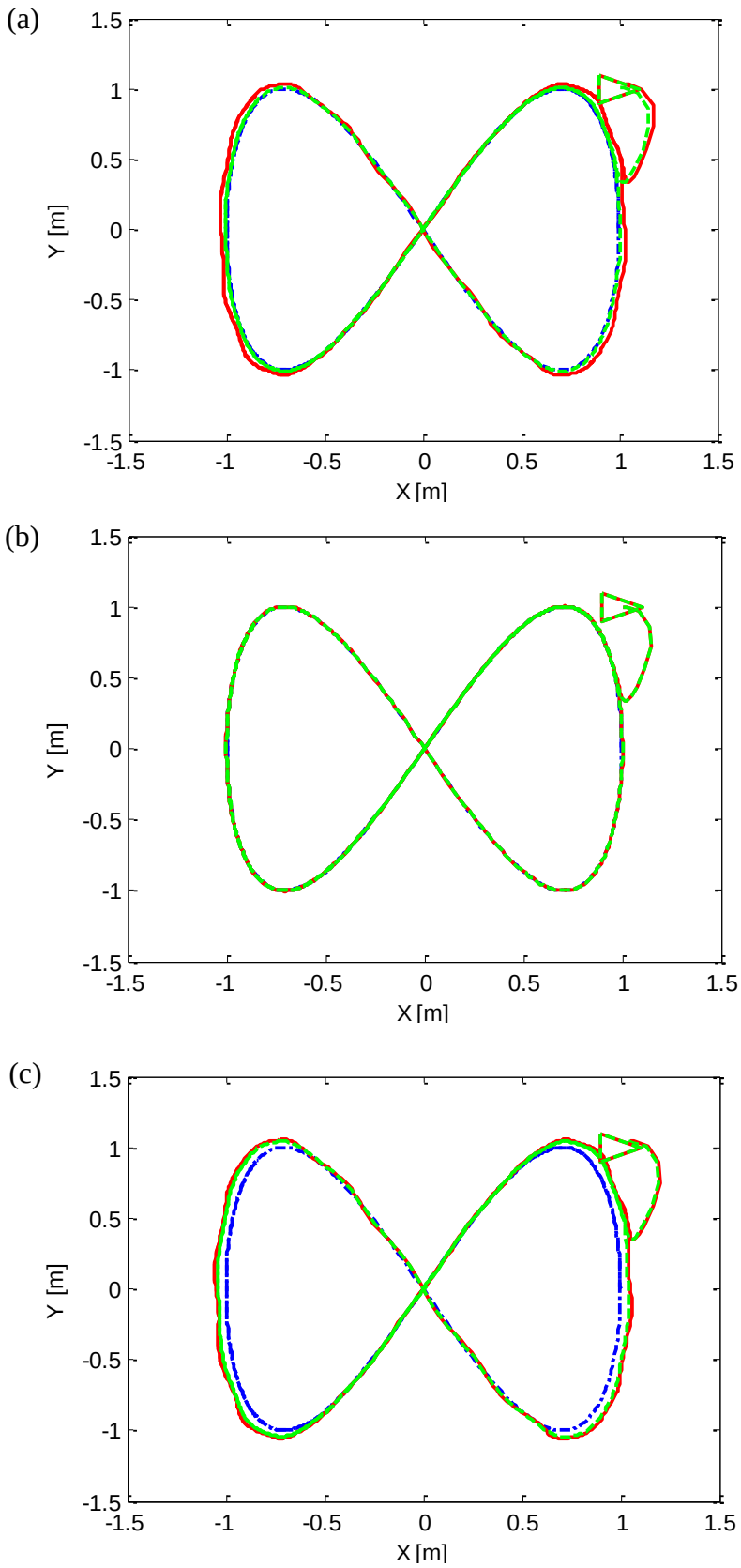


Fig. 3-13 Trajectory tracking of mobile platform for every case.

(a) Cases 1 (—) and 2 (- - -), (b) Cases 3 (—) and 4 (- - -),

(c) Cases 5 (—) and 6 (- - -), and desired trajectory (- - -).

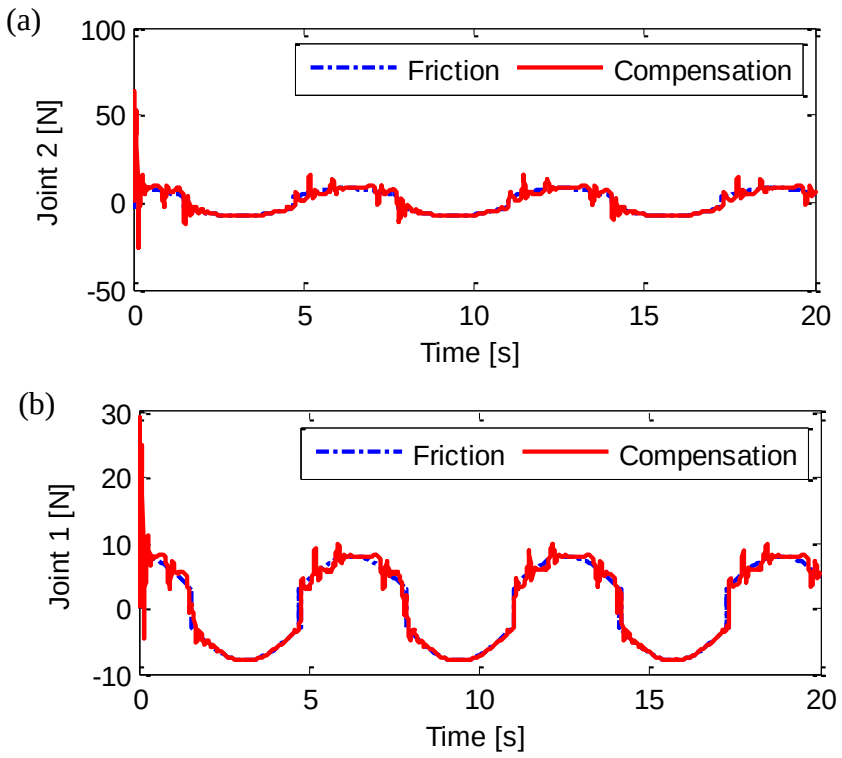


Fig. 3-14 Friction and its compensation for the manipulator (case 2).

(a) Joint 2, and (b) Joint 1.

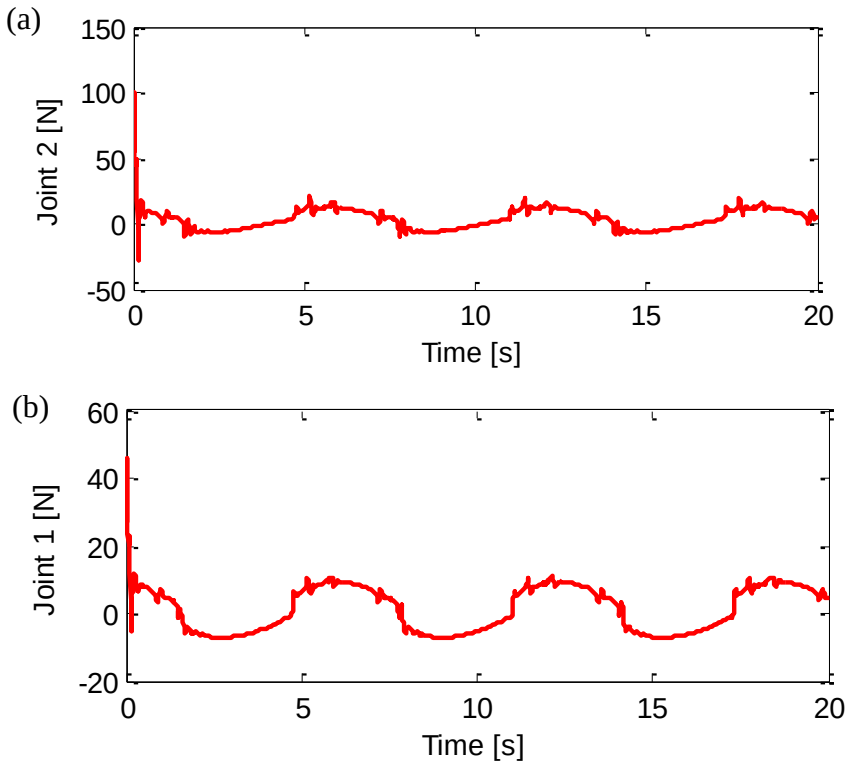


Fig. 3-15 Control input torque for each joint of the manipulator (case 2).

(a) Joint 2, and (b) Joint 1.

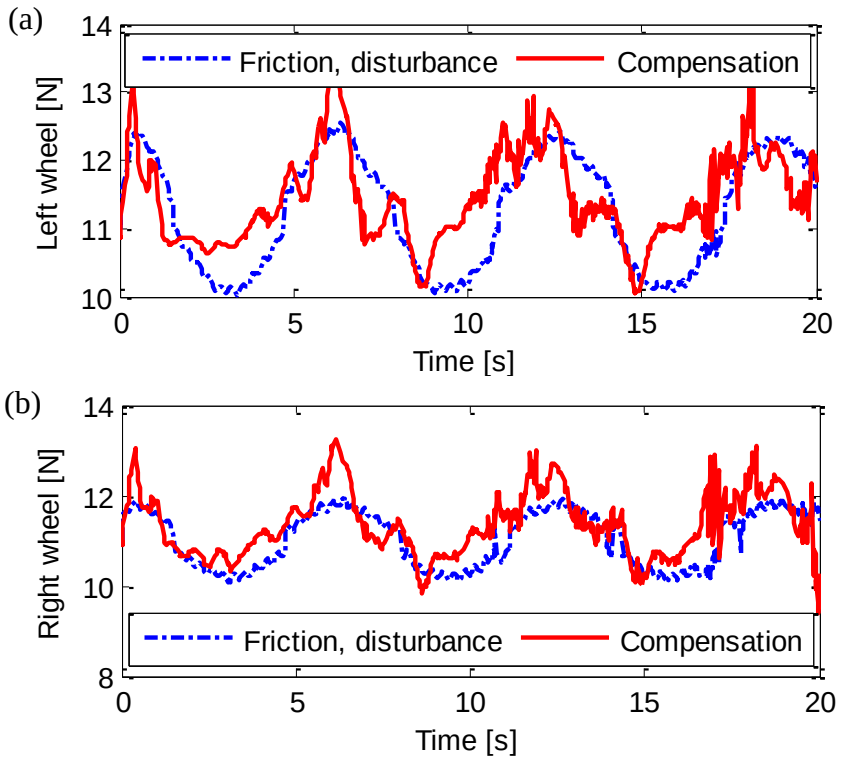


Fig. 3-16 Friction and disturbance for each wheel, and their compensation (case 2).

(a) Left wheel, and (b) Right wheel.

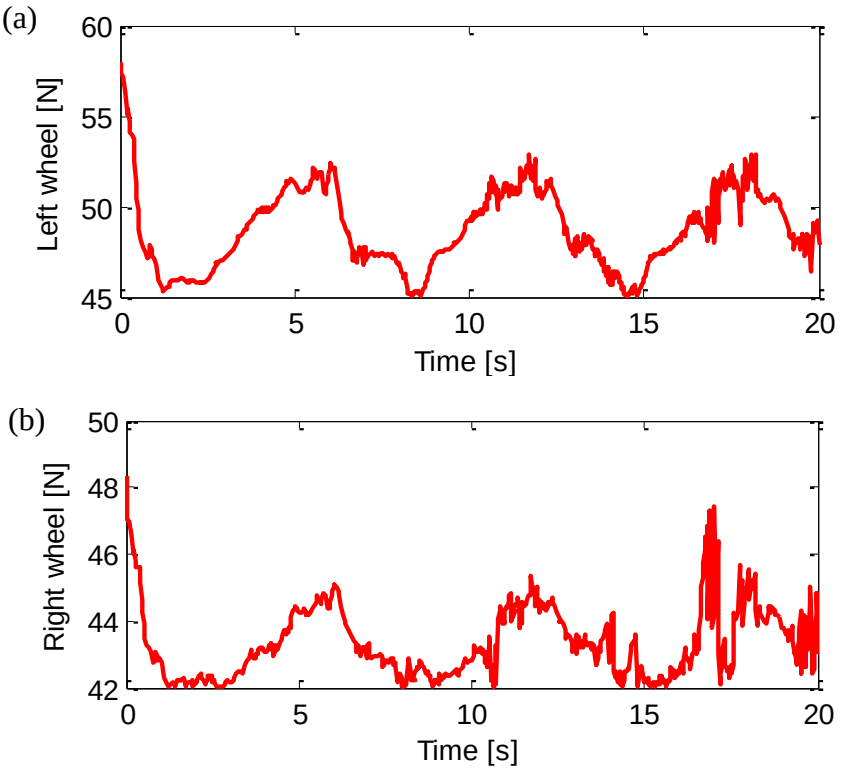


Fig. 3-17 Control input torque for each wheel (case 2).

(a) Left wheel, and (b) Right wheel.

Figure 3-16 plots the friction and disturbance for each wheel, and their compensations. For comparison, the friction for each joint is considered and both friction and disturbance for each wheel are considered since wheels may be subject to disturbance from roads. Through comparative analysis, it can be found that when the disturbance is considered, the compensation effect is degraded as shown in Fig. 3-16. The control input torque for each wheel is shown in Fig. 3-17. Comparing Fig. 3-13(a) with Fig. 3-7, it is obvious that the use of the ACL-RT controller results in accurate trajectory tracking of the platform.

3.5 Summary

In this chapter, the problem of trajectory tracking control of a mobile manipulator was investigated. To solve this problem, an ideal kinematic model assuming the pure rolling of the wheels, a non-ideal kinematic model considering the sliding effects, and a dynamic model considering the dynamic interaction between the mobile platform and manipulator were constructed. From these models, several controllers were designed by using the backstepping approach. In classical trajectory tracking systems, the dynamic interaction and modeling uncertainties are neglected by modeling the mobile platform and manipulator as a whole system at the kinematic level, which may deteriorate the system performance when the robot moves in the presence of sliding. Therefore, in this chapter an adaptive controller based on the backstepping approach was employed. To account for the dynamic interaction effect, the DPM was utilized and the mobile manipulator was modeled as a multi-body system at the dynamic level. Then the load carried by the end-effector was varied and simulation results were obtained to verify the effect of dynamic interaction. To accurately track the desired trajectory when the robot moves in the presence of modeling uncertainties such as friction and external disturbances, a fuzzy compensator was proposed to counteract the uncertainties. In addition, a RT was added to reduce approximation errors and ensure system stability when the platform is subject to external disturbances. Finally, numerical simulations were performed for

the mobile manipulator under several conditions. The results demonstrated the effectiveness of the trajectory tracking and the robustness with respect to the dynamic interaction and uncertainties. The proposed method may well make sense to put it into the high precision control of mobile manipulators.

Chapter 4 Dynamic Stability Control of Robots in Rough Terrain

4.1 Introduction

In the field of robotics, an essential feature of robots is the visual system, which plays a key role in obtaining environmental information. Robots must have the detailed and accurate visual information about the environments in which they operate. Two methods are available to improve the performance of visual systems. One is to employ high-performance visual equipment that can extract more accurate information from an environment. For example, advanced RGB-D cameras (e.g., Microsoft Kinect) are applied to build dense 3D environment maps for robot navigation, manipulation, mapping, and telepresence [78, 79]. The other is to develop mechanical devices combined with control approaches to eliminate external interference that degrades the performance of visual systems. Typically, fully active or semi-active suspensions have been used to improve the stability of robots moving through rough terrain [3, 80]. Semi-active suspensions use a variable damping force as the control force. For example, an electro rheological (ER) damper or a magneto rheological (MR) damper applies an electric field or a magnetic field to control the viscosity of ER or MR fluids [81]. Fully active suspensions produce the vibration control force with a separate hydraulic or pneumatic unit [82]. Therefore, the cost and weight of fully active suspensions are much higher than those of semi-active suspensions. Semi-active suspensions are receiving more attention because of their low cost and competitive performance in comparison with fully active suspensions. This chapter investigates ways to improve the dynamic stability of a suspended wheeled robot, installed with semi-active suspensions, when it passes through rough terrain.

Generally, rough terrain causes external random excitations on a dynamic system.

The problem of rough terrain and its influence on vehicles and robots due to unwanted random vibration remain a subject of research among automotive manufacturers and robot researchers whose objective is to minimize vibration effects on drivers and robotic equipments [83, 84]. Several new applications of fully active and semi-active control procedures and the use of special devices to improve rough terrain stability have been developed recently. Iagnemma and Dubowki [3] presented an approach to control actively articulated suspensions to improve the rollover stability of a planetary rover. Liu et al. [28] compared the vibration isolation characteristics of four established semi-active suspension control strategies based on skyhook and balance controls. Shen et al. [85] discussed an averaging method to provide an approximate analytical solution for a semi-active on−off dynamic vibration absorber. Further, new ideas in suspension control such as adaptive control [86] and optimal control [87] have been implemented and tested in several applications. However, studies regarding mobile manipulator control indicated that top-heaviness sensitivity is a significant factor in stability measure, particularly, when the end effector carries a load [88].

In addition, unlike trucks or cars, wheeled mobile robots can reposition their center of gravity (CG) by repositioning the manipulator. In past studies, Farritor et al. [89] proposed a genetic-algorithm-based method for repositioning a vehicle's manipulator to modify its center of mass location and to aid stability during motion. Huang et al. [90] presented a static stability compensation range to evaluate the capability to compensate for instability by moving the manipulator when performing a planned task. However, the capability to compensate for instability by moving the manipulator is limited because control difficulty actually increases with the redundancy of the manipulator. The entire system becomes unstable when redundancy exceeds a certain value. Considering the above reasons, a control method, in which the inputs of semi-active suspensions are optimized by chaotic particle swarm optimization (CPSO), is proposed to improve the dynamic stability of a robot moving in rough terrain.

The rest of this chapter is organized as follows. In Section 4.2, a dynamic model and the governing equations of a robot with semi-active suspensions are presented. In Section 4.3, the dynamic stability measure approach is described. In Section 4.4, the

controller is designed based on the semi-active damper control model and optimized by employing CPSO. In Section 4.5, results and discussion are presented to demonstrate the quality of both the measuring method and the optimal control of the robot's dynamic stability using the proposed approach. Finally, in Section 4.6, concluding remarks are presented.

4.2 Robot with Semi-Active Suspension Systems

A mobile robot shown in Fig. 4-1 is equipped with an articulated manipulator mounted on a mobile platform that is supported by two rear driving wheels and one front caster wheel. In this section, a mobile robot having a semi-active suspension system mounted on each of the two rear wheels is considered. The equations of motion of a vibratory system can be derived from the Lagrangian function L, expressed in generalized coordinates as

$$L = T - V , \tag{4-1}$$

where T and V represent the kinetic and potential energies of the system, respectively.

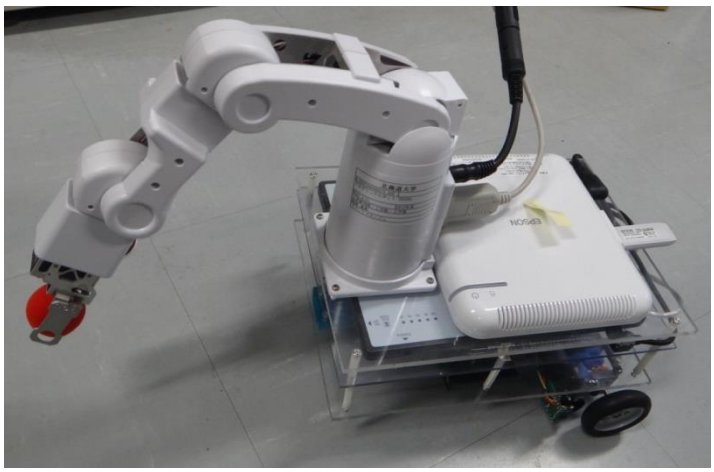


Fig. 4-1 Mobile robot system.

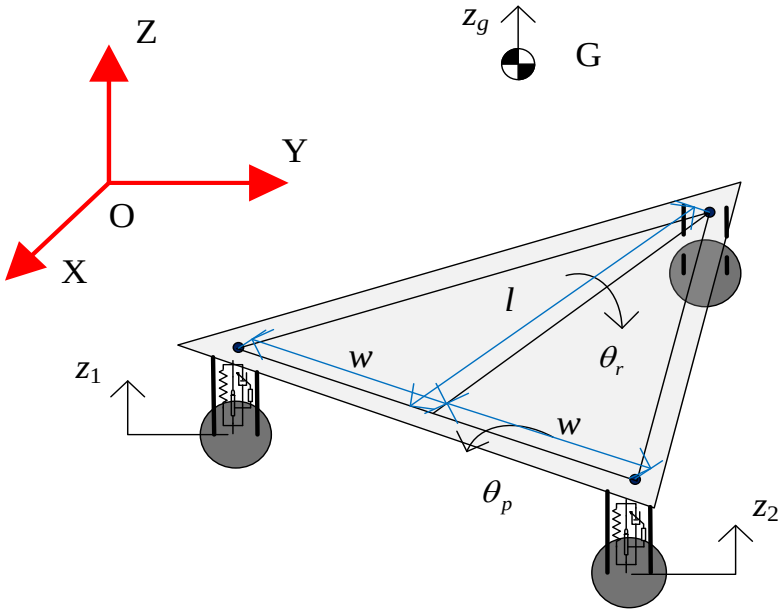


Fig. 4-2 Dynamic model of the robot with suspensions.

The extended Lagrange equation for an n degree-of-freedom (DOF) system, including the dissipative energy term, can be stated as

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right)-\frac{\partial L}{\partial q_i}+\frac{\partial R}{\partial \dot{q}_i}=Q_i\ , \qquad i=1,2,\ldots,n\ , \tag{4-2}$$

where q_i represents the generalized coordinates, Q_i represents non-potential forces, and R represents the Rayleigh dissipation function.

The dynamic model of the robot with semi-active suspensions between the wheels and platform is presented in Fig. 4-2. To account for the bounce, pitch, and roll motions of the robot, L and R can be, respectively, expressed as

$$\begin{aligned} L=T-V=&\frac{1}{2}\left(m\dot{z}_g^2+I_p\dot{\theta}_p^2+I_r\dot{\theta}_r^2\right)-\frac{1}{2}k_1\left(z_g-l\sin\theta_p+w\sin\theta_r-z_1\right)^2\\ &-\frac{1}{2}k_2\left(z_g-l\sin\theta_p-w\sin\theta_r-z_2\right)^2\ , \end{aligned} \tag{4-3}$$

$$\begin{aligned} R=&\frac{1}{2}\hat{c}_1\left(\dot{z}_g-l\cos\theta_p\dot{\theta}_p+w\cos\theta_r\dot{\theta}_r-\dot{z}_1\right)^2\\ &+\frac{1}{2}\hat{c}_2\left(\dot{z}_g-l\cos\theta_p\dot{\theta}_p-w\cos\theta_r\dot{\theta}_r-\dot{z}_2\right)^2\ , \end{aligned} \tag{4-4}$$

where m is the mass of the robot; z_g is the linear displacement of the CG along the

Z-axis; θ_p and θ_r are the pitch and roll angles; I_p and I_r are the moments of inertia of the robot with respect to pitch and roll vibration axes; k_j and $\hat{c}_j$ (j = 1, 2) are spring and continuously variable damping coefficients of the two suspensions; and $\dot{z}_j$ and z_j (j = 1, 2) are the linear velocities and base displacements of the two suspensions along the Z-axis, respectively, acting as external disturbances caused by rough terrain. The meanings of symbols *w* and *l* are indicated in Fig. 4-2. The system parameters of the robot are listed in Table 4-1.

According to Newton's second law of motion, the equation of vertical motion can be written as

$$\begin{aligned} & m\ddot{z}_g + \hat{c}_1\left(\dot{z}_g - l\cos\theta_p\dot{\theta}_p + w\cos\theta_r\dot{\theta}_r - \dot{z}_1\right) \\ & + \hat{c}_2\left(\dot{z}_g - l\cos\theta_p\dot{\theta}_p - w\cos\theta_r\dot{\theta}_r - \dot{z}_2\right) \\ & + k_1\left(z_g - l\sin\theta_p + w\sin\theta_r - z_1\right) \\ & + k_2\left(z_g - l\sin\theta_p - w\sin\theta_r - z_2\right) = 0. \end{aligned} \tag{4-5}$$

Table 4-1 System parameters.

Symbol	Meaning	Value
m	mass of the robot	20.56 kg
l	length of the wheelbase	0.46 m
w	half of the distance between the two rear wheels	0.23 m
I_p	moment of inertia of the robot with respect to pitch axis	0.26 kg·m^2
I_r	moment of inertia of the robot with respect to roll axis	0.18 kg·m^2
k_i	stiffness coefficient of the *i*th suspension	2.50 kN/m
$\hat{c}_{\max}$	maximum damping coefficient of the semi-active damper	5.26 N/m/s
$\hat{c}_{\min}$	minimum damping coefficient of the semi-active damper	0.08 N/m/s

Similarly, according to the moment balance theory, the vibration differential equations of pitch and roll motions can be written as

$$\begin{aligned} & I_p\ddot{\theta}_p - \hat{c}_1 l\cos\theta_p\left(\dot{z}_g - l\cos\theta_p\dot{\theta}_p + w\cos\theta_r\dot{\theta}_r - \dot{z}_1\right) \\ & - \hat{c}_2 l\cos\theta_p\left(\dot{z}_g - l\cos\theta_p\dot{\theta}_p - w\cos\theta_r\dot{\theta}_r - \dot{z}_2\right) \\ & - k_1 l\cos\theta_p\left(z_g - l\sin\theta_p + w\sin\theta_r - z_1\right) \\ & - k_2 l\cos\theta_p\left(z_g - l\sin\theta_p - w\sin\theta_r - z_2\right) = 0, \end{aligned} \tag{4-6}$$

$$\begin{aligned} & I_r\ddot{\theta}_r + \hat{c}_1 w\cos\theta_r\left(\dot{z}_g - l\cos\theta_p\dot{\theta}_p + w\cos\theta_r\dot{\theta}_r - \dot{z}_1\right) \\ & - \hat{c}_2 w\cos\theta_r\left(\dot{z}_g - l\cos\theta_p\dot{\theta}_p - w\cos\theta_r\dot{\theta}_r - \dot{z}_2\right) \\ & + k_1 w\cos\theta_r\left(z_g - l\sin\theta_p + w\sin\theta_r - z_1\right) \\ & - k_2 w\cos\theta_r\left(z_g - l\sin\theta_p - w\sin\theta_r - z_2\right) = 0. \end{aligned} \tag{4-7}$$

In the above equations, the damping coefficients $\hat{c}_1$ and $\hat{c}_2$ are continuous and controllable variables. Their control and optimization will be addressed in Section 4.4.

4.3 Dynamic Stability Measure

In attempts to solve the motion-planning problem to reduce vibrations affecting robots, previous studies have proposed various definitions of the robot stability measure [6, 91–94]. By considering the degree to which a stability constraint is satisfied, these studies have defined measures of the stability margin. However, these measures do not consider top-heaviness sensitivity; thus, when the CG is high off the ground (e.g., when the end effector carries a load), the measures may become invalid. To consider top-heaviness sensitivity in measure of the dynamic stability margin of a robot, it is assume that the robot body is nominally in contact with the ground, as would be the case if wheels provide mobility. Vibrations in the pitch and roll angle directions occur when a nominally upright robot moves through rough terrain, resulting in the reduction of the minimum stability angle, thus deteriorating the stability margin of the robot.

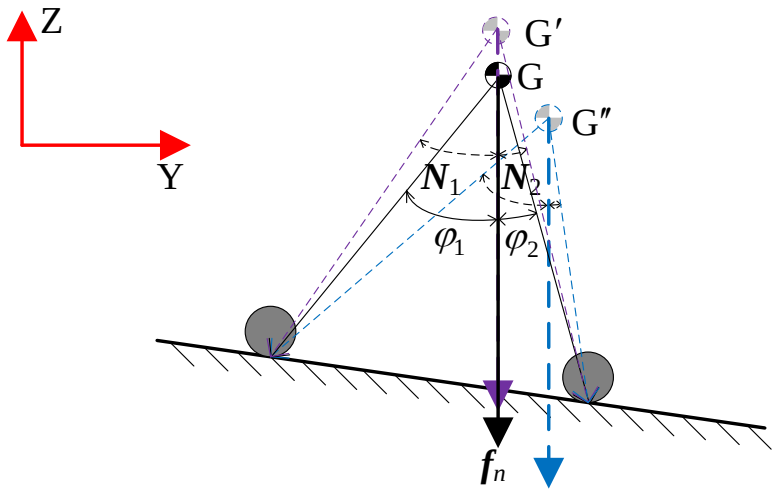


Fig. 4-3 Stability measure with a planar instance.

For the sake of clarity, at first the stability margin measure is analyzed with a planar instance, which highlights its simple graphical nature. Presented in Fig. 4-3 is a two contact points planar system whose center-of-mass is subject to a net force $\boldsymbol{f}_n$, which is the sum of all forces acting on the robot body except the supporting reaction forces. This force vector subtends two stability angles, φ_1 and φ_2, with the two vibration axis normals $\boldsymbol{N}_1$ and $\boldsymbol{N}_2$. Considering top-heaviness sensitivity, the proposed stability margin measure is defined by the minimum of the two stability angles, and weighted by the magnitude of the force vector as

$$\eta = \min(\varphi_1, \varphi_2) \cdot \|\boldsymbol{f}_n\|, \tag{4-8}$$

where $\|\boldsymbol{f}_n\|$ is the modulus of $\boldsymbol{f}_n$.

For the stability margin measure, an increase in CG height clearly results in a smaller minimum stability angle and a reduced margin of the stability. This is illustrated by moving the CG position from G to G′ as shown in Fig. 4-3. Therefore, for a robot which moves in rough terrain, or for a mobile manipulator which carries variable loads, the stability margin should be top-heaviness sensitive. Taking the suspended wheeled robot as an example, when the CG position moves to G′ or G″ by reconfiguring the semi-active suspensions or varying the end-effector loads, stability angles φ_1 and φ_2 are changed, which leads to vibrations or at worst rollover. Critical rollover occurs when φ_1 or φ_2 goes to zero (i.e., $\boldsymbol{f}_n$ coincides with

$\boldsymbol{N}_1$ or $\boldsymbol{N}_2$), or when the modulus of $\boldsymbol{f}_n$ goes to zero and even the smallest disturbance may topple the robot. If $\boldsymbol{f}_n$ lies outside the cone described by $\boldsymbol{N}_1$ and $\boldsymbol{N}_2$ the stability angle becomes negative and rollover is in progress.

Then the stability margin measure is analyzed with a three-wheeled robot instance. As shown in Fig. 4-4, for a three-wheeled robot there are three contact points with the ground, which form a triangle when projected onto the horizontal plane. Similar to the direct path method addressed by Zhong et al. [95], vectors are used to describe the point locations. Let $\boldsymbol{p}_j$ (j=1, 2, 3, 4) represent the locations of the points at which the axes intersect the boundary of the triangle, and $\boldsymbol{p}_g$ represents the location of the CG. The pitch and roll axes of the robot are denoted as $\boldsymbol{a}_p$ and $\boldsymbol{a}_r$, and are expressed by Eqs. (4-9) and (4-10), respectively.

$$\boldsymbol{a}_p = \boldsymbol{p}_3 - \boldsymbol{p}_1, \tag{4-9}$$

$$\boldsymbol{a}_r = \boldsymbol{p}_4 - \boldsymbol{p}_2. \tag{4-10}$$

The pitch and roll vibration axis normals $\boldsymbol{N}_p$ and $\boldsymbol{N}_r$, respectively, which intersect the CG of the robot, are given by

$$\boldsymbol{N}_p = \left(\boldsymbol{E}^{3\times3} - \hat{\boldsymbol{a}}_p \cdot \hat{\boldsymbol{a}}_p{}^T\right)\cdot\left(\boldsymbol{p}_3 - \boldsymbol{p}_g\right), \tag{4-11}$$

$$\boldsymbol{N}_r = \left(\boldsymbol{E}^{3\times3} - \hat{\boldsymbol{a}}_r \cdot \hat{\boldsymbol{a}}_r{}^T\right)\cdot\left(\boldsymbol{p}_4 - \boldsymbol{p}_g\right), \tag{4-12}$$

where $\boldsymbol{E}^{3\times3}$ is a 3×3 identity matrix, $\hat{\boldsymbol{a}}_p = \boldsymbol{a}_p / \|\boldsymbol{a}_p\|$, and $\hat{\boldsymbol{a}}_r = \boldsymbol{a}_r / \|\boldsymbol{a}_r\|$.

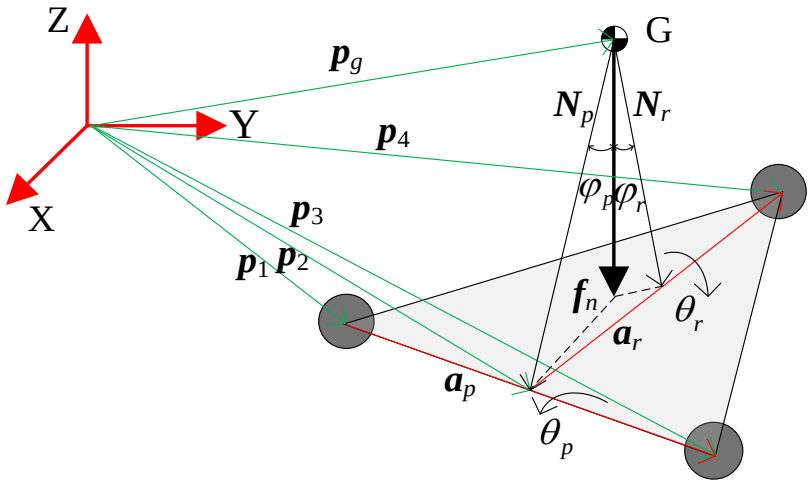


Fig. 4-4 Schematic diagram of stability measure.

From Newton's law, the force equilibrium equation for the robot is defined as

$$\boldsymbol{f}_n = \sum\left(\boldsymbol{f}_g + \boldsymbol{f}_m - \boldsymbol{f}_k - \boldsymbol{f}_d\right), \tag{4-13}$$

where $\boldsymbol{f}_g$ is the gravitational load; $\boldsymbol{f}_m$ is the load transmitted by the manipulator to the robot's body; and $\boldsymbol{f}_k$ and $\boldsymbol{f}_d$ are the spring and damping forces of the suspension, respectively. For any vibration axis, those components of the net force and moment acting along the axis are only concerned with. Thus, the components of $\boldsymbol{f}_n$ for the pitch and roll vibration axes can be described as

$$\begin{aligned}\boldsymbol{f}_i &= \left(\boldsymbol{E}^{3\times3} - \hat{\boldsymbol{a}}_i \cdot \hat{\boldsymbol{a}}_i^{\,T}\right)\cdot \boldsymbol{f}_n \\ &= \left(\boldsymbol{E}^{3\times3} - \hat{\boldsymbol{a}}_i \cdot \hat{\boldsymbol{a}}_i^{\,T}\right)\cdot\left(\boldsymbol{f}_g + \boldsymbol{f}_m - \boldsymbol{f}_k - \boldsymbol{f}_d\right), \qquad i = p, r.\end{aligned} \tag{4-14}$$

For the robot with a manipulator, the moment transmitted by the manipulator is considered because of the loads carried by the end effector. Similarly, the moment equilibrium equation and the components for the vibration axes are defined as

$$\boldsymbol{m}_n = \sum \boldsymbol{m}_m\,, \tag{4-15}$$

$$\boldsymbol{m}_i = \left(\hat{\boldsymbol{a}}_i \cdot \hat{\boldsymbol{a}}_i^{\,T}\right)\cdot \boldsymbol{m}_m\,, \qquad i = p, r\,, \tag{4-16}$$

where $\boldsymbol{m}_n$ is the net moment, and $\boldsymbol{m}_m$ is the moment transmitted by the manipulator. However, this dynamic stability measure focuses on the computation of the stability angle between the net force vector and each of the vibration axis normals. Therefore, it is necessary to replace the pitch and roll moments (i.e., $\boldsymbol{m}_p$ and $\boldsymbol{m}_r$) with an equivalent force couple. For the pitch and roll vibration axes, the member of the force couple action on the CG can be given by

$$\boldsymbol{f}_{m_i} = \frac{\hat{\boldsymbol{N}}_i}{\left\|\boldsymbol{N}_i\right\|} \times \boldsymbol{m}_i = \frac{\hat{\boldsymbol{N}}_i}{\left\|\boldsymbol{N}_i\right\|} \times \left(\hat{\boldsymbol{a}}_i \cdot \hat{\boldsymbol{a}}_i^{\,T}\right)\cdot \boldsymbol{m}_m\,, \qquad i = p, r\,, \tag{4-17}$$

where $\hat{\boldsymbol{N}}_i = \boldsymbol{N}_i / \left\|\boldsymbol{N}_i\right\|$. Then, the total components of the vibration axes are as follows:

$$\begin{aligned}\boldsymbol{f}_i^{\,*} &= \boldsymbol{f}_i + \boldsymbol{f}_{m_i} \\ &= \left(\boldsymbol{E}^{3\times3} - \hat{\boldsymbol{a}}_i \cdot \hat{\boldsymbol{a}}_i^{\,T}\right)\cdot\left(\boldsymbol{f}_g + \boldsymbol{f}_m - \boldsymbol{f}_k - \boldsymbol{f}_d\right) + \frac{\hat{\boldsymbol{N}}_i}{\left\|\boldsymbol{N}_i\right\|} \times \left(\hat{\boldsymbol{a}}_i \cdot \hat{\boldsymbol{a}}_i^{\,T}\right)\cdot \boldsymbol{m}_m\,, \qquad i = p, r.\end{aligned} \tag{4-18}$$

Consequently, the stability angle can be computed as the angle between the total

component force and axis normal for each vibration axis as follows:

$$\varphi_i = \xi_i \arccos\left(\hat{\boldsymbol{f}}_i^* \cdot \hat{\boldsymbol{N}}_i\right), \qquad i = p, r, \tag{4-19}$$

where $\hat{\boldsymbol{f}}_i^* = \boldsymbol{f}_i^* \big/ \left\| \boldsymbol{f}_i^* \right\|$, and ξ_i is defined by

$$\xi_i = \begin{cases} +1 & \left(\hat{\boldsymbol{N}}_i \times \hat{\boldsymbol{f}}_i^*\right) \cdot \hat{\boldsymbol{a}}_i < 0, \\ -1 & \left(\hat{\boldsymbol{N}}_i \times \hat{\boldsymbol{f}}_i^*\right) \cdot \hat{\boldsymbol{a}}_i \geq 0, \end{cases} \qquad i = p, r. \tag{4-20}$$

To include top-heaviness sensitivity, the proposed stability margin measure η^* is defined by the two stability angles and is weighted by the magnitude of the force vector as

$$\eta^* = \sum_i \varphi_i \left\| \boldsymbol{f}_n \right\|, \qquad i = p, r. \tag{4-21}$$

To minimize the vibrations in the pitch and roll directions, the actual values of the stability angles φ_p and φ_r are considered, rather than their minimum values because these values vary while the robot is in bounce motion.

4.4 Controller Design and Optimization

4.4.1 Semi-Active Damper Control Model

A semi-active suspension with an MR damper mounted between the robot body and a wheel subjected to excitation from a rough pavement is shown schematically in Fig. 4-5. The relative displacement (i.e., $z_g - z$) can be measured using a sensor, and the sensor signal is fed into a controller that tunes the MR damping coefficient so that the damping force can be varied as a function of time. Semi-active dampers generally fall into two categories: the on−off type and the continuously variable type [28]. In this study, a continuously variable damper is used by optimizing its coefficient to improve the dynamic stability of the robot. Similar to the on-off damper, the continuously variable damper is also switched between “on” and “off” states. However, in its “on” state, the damping coefficient and corresponding damping force are not invariant but varied.

The force-velocity characteristics of a continuously variable damper are illustrated in Fig. 4-6, where F_d and $\dot{z}_g - \dot{z}$ describe the damping force and relative velocity, respectively. The shaded part of graph in Fig. 4-6 represents the range of achievable damping coefficients in "on" state. Ideally, the off-state damping should be zero, but in practical situations this is not possible because of some constraints. Therefore, in "off" state, the damping coefficient and corresponding damping force are not zero but relatively low as shown in Fig. 4-6.

Rakheja and Sankar [96] noticed that the damping force in a passive damper tends to increase the sprung mass acceleration when the damping force is in the same direction as the spring force. A semi-active damper should ideally have no, or minimal, damping force when the spring and damping forces are in the same direction. In this subsection, this concept is discussed and a continuous balance control strategy is suggested. In this strategy, the damping force is nullified when the damping and spring forces are in the same direction, and the damping force is adjusted to balance the spring force when they act in opposite directions. In this case, the damping coefficient can be continuously varied, depending on relative displacement and relative velocity so that the spring and damper forces are exactly balanced by damper compensation. Mathematically, the restoring force can be expressed as

$$\boldsymbol{f}_d = \begin{cases} -k(z_g - z), & (z_g - z)(\dot{z}_g - \dot{z}) \le 0, \\ 0, & (z_g - z)(\dot{z}_g - \dot{z}) > 0. \end{cases} \tag{4-22}$$

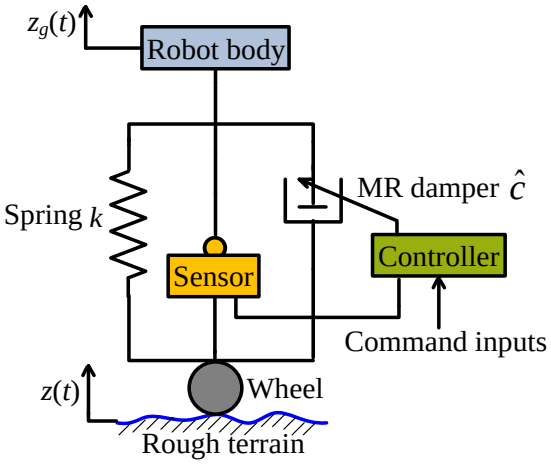


Fig. 4-5 Semi-active suspension system with a MR damper.

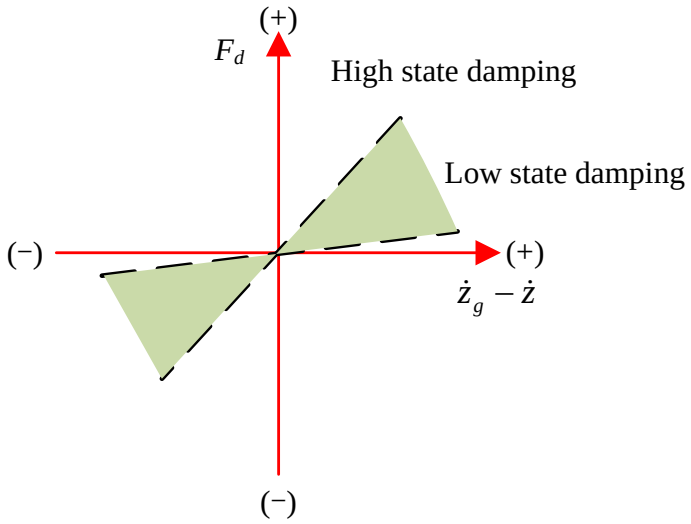


Fig. 4-6 Force-velocity characteristics of a continuously variable damper.

The switching between the "on" and "off" states is controlled by the term $(z_g - z)(\dot{z}_g - \dot{z})$, known as the condition function. The semi-active damping coefficient required for this control algorithm can be written as

$$\hat{c} = \begin{cases} \dfrac{-k(z_g - z)}{(\dot{z}_g - \dot{z})}, & (z_g - z)(\dot{z}_g - \dot{z}) \le 0, \\ 0, & (z_g - z)(\dot{z}_g - \dot{z}) > 0. \end{cases} \tag{4-23}$$

From Eq. (4-23), one can see that when the relative velocity is very small, the required damping coefficient abruptly increases and approaches infinity. However, in reality the semi-active damping coefficient is limited by the physical parameters of the damper, where there are both an upper bound, $\hat{c}_{\max}$, and a lower bound, $\hat{c}_{\min}$. Given this practical consideration, the damping coefficient in Eq. (4-23) can thus be rewritten as

$$\hat{c} = \begin{cases} \max\left\{\hat{c}_{\min}, \min\left[\dfrac{-k(z_g - z)}{(\dot{z}_g - \dot{z})}, \hat{c}_{\max}\right]\right\}, & (z_g - z)(\dot{z}_g - \dot{z}) \le 0, \\ \hat{c}_{\min}, & (z_g - z)(\dot{z}_g - \dot{z}) > 0. \end{cases} \tag{4-24}$$

The use of this damping coefficient model only requires the measurement of the relative motion between the robot body and the wheel. The relative displacement can

be obtained by the sensor shown in Fig. 4-5, and the relative velocity can be obtained by differentiating the measured relative displacement. The system is simple in structure.

However, for the damping coefficient model defined by Eq. (4-24), the denominator in the expressions of the damping equation introduces a high degree of nonlinearity into the system. This will cause jerks, which is defined as sharp changes in the acceleration response of the system, and the jerk response should be reduced when considering the implementation of semi-active dampers. Because of this concern, a shaping function is introduced to smooth the semi-active damping force. The shaping function can be written as a function of the variables that define a condition function, i.e., $F\left(z_g - z,\, \dot{z}_g - \dot{z}\right)$. Recalling the form of the damping force in Eq. (4-22), it is noted that the shaping function for a particular semi-active damper is not unique, allowing for different choices of shaping functions.

To eliminate the denominator in the expression of the damping coefficient, the shaping function must include $\left(\dot{z}_g - \dot{z}\right)$ term. Further, to avoid frequent changes between the negative and positive signs, absolute values are introduced and the final expression of damping coefficient consists of an even power term. Hence, the shaping function can be chosen as

$$F\left(z_g - z,\, \dot{z}_g - \dot{z}\right) = \left|z_g - z\right| \cdot \left|\dot{z}_g - \dot{z}\right|. \tag{4-25}$$

With this shaping function, the damping force becomes

$$\boldsymbol{f}_d = \begin{cases} -\sigma \cdot \left|z_g - z\right| \cdot \left|\dot{z}_g - \dot{z}\right| \cdot \left(z_g - z\right), & \left(z_g - z\right)\left(\dot{z}_g - \dot{z}\right) \le 0, \\ 0, & \left(z_g - z\right)\left(\dot{z}_g - \dot{z}\right) > 0. \end{cases} \tag{4-26}$$

where σ is the gain factor that is optimized by CPSO based on a stability criterion as previously discussed. Considering the practical constraints on the damper, the damping coefficient can once again be rewritten as

$$\hat{c} = \begin{cases} \max\left\{\hat{c}_{\min},\, \min\left[\sigma\left(z_g - z\right)^2,\, \hat{c}_{\max}\right]\right\}, & \left(z_g - z\right)\left(\dot{z}_g - \dot{z}\right) \le 0, \\ \hat{c}_{\min}, & \left(z_g - z\right)\left(\dot{z}_g - \dot{z}\right) > 0. \end{cases} \tag{4-27}$$

4.4.2 Optimization Using the CPSO Algorithm

Based on the previously obtained stability measure criterion in Section 4.3, an optimal control method, which searches the optimum damping force by optimizing σ in the control model of damper, is designed to improve the dynamic stability of the robot. The cost function is defined by

$$F = \left| \eta_0 - \eta^* \right|, \tag{4-28}$$

where η^* is defined by Eq. (4-21), and η_0 denotes the maximum stability margin of the robot measured when the robot is stationary or moving on smooth ground. The cost function not only considers the vibrations from pitch and roll motions but also from bounce motion; it can provide adequate road-holding ability in rough terrain.

The goal of the optimization procedure is to minimize the cost index F subject to practical constraints. To solve this problem, the CPSO is employed in the search process. The velocity and position of the particle at the next iteration are calculated according to Eqs. (4-29) and (4-30), respectively [15].

$$\begin{aligned} v_{i,j}(t+1) = w \cdot v_{i,j}(t) + c_1 \cdot rand_1\left(pbest_{i,j}(t) - x_{i,j}(t)\right) \\ + c_2 \cdot rand_2\left(gbest_{i,j}(t) - x_{i,j}(t)\right), \end{aligned} \tag{4-29}$$

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1), \tag{4-30}$$

where $v_{i,j}(t)$ and $x_{i,j}(t)$ are the velocity and position of the jth dimension of the ith particle in the tth iteration, respectively. Learning rates are represented by c_1 and c_2, usually both are constants set to 2; $rand_1$ and $rand_2$ are two pseudo-random values in the range of [0, 1]; *pbest* and *gbest* are current personal best and global best positions, respectively, and w is an inertial weight that determines how much of the former velocity is inherited by a particle. Specifically, Fig. 4-7 presents a detailed flow chart.

To enrich the searching behavior and to avoid being trapped into local optimum, chaotic dynamics is incorporated into the above PSO. Here, the well-known logistic equation [97], which exhibits the sensitive dependence on initial conditions, is employed for constructing hybrid PSO. The logistic equation is defined as follows:

$$x_{n+1} = \rho \cdot x_n(1 - x_n), \qquad 0 \le x_0 \le 1, \tag{4-31}$$

where ρ is the control parameter, x is a variable and $n = 0, 1, 2, \cdots$. Although the above equation is deterministic, it exhibits chaotic dynamics when $\rho = 4$ and $x_0 \notin \{0, 0.25, 0.5, 0.75, 1\}$. That is, it exhibits the sensitive dependence on initial conditions, which is the basic characteristic of chaos.

A minute difference in the initial value of the chaotic variable would result in a considerable difference in its long time behavior. The track of chaotic variable can travel ergodically over the whole search space. In general the above chaotic variable has special characters, i.e., ergodicity, pseudo-randomness, and irregularity.

The process of the chaotic local search could be defined through the following equation:

$$cx_i^{k+1} = 4cx_i^k\left(1 - cx_i^k\right), \qquad i = 1, 2, \cdots, n, \tag{4-32}$$

where cx_i is the ith chaotic variable, and k denotes the iteration number. Obviously, cx_i^k is distributed in the range (0, 1) under the conditions that the initial $cx_i^0 \in (0, 1)$ and that $cx_i^0 \notin \{0.25, 0.5, 0.75\}$.

The procedures of chaotic local search (CLS) can be illustrated as follows:

Step 1: Set $k = 0$, and map the decision variables x_i^k, $i = 1, 2, \cdots, n$ among the intervals $(\min x_i, \max x_i)$, $i = 1, 2, \cdots n$ to chaotic variables cx_i^k located in the interval (0, 1) using the following equation:

$$cx_i^k = \frac{x_i^k - \min x_i}{\max x_i - \min x_i}, \qquad i = 1, 2, \cdots, n. \tag{4-33}$$

Step 2: Determine the chaotic variables cx_i^{k+1} for the next iteration using the logistic equation according to cx_i^k.

Step 3: Converting the chaotic variables cx_i^{k+1} to decision variables x_i^{k+1} using the following equation:

$$x_i^{k+1} = \min x_i + cx_i^{k+1}(\max x_i - \min x_i), \quad i = 1, 2, \cdots, n. \tag{4-34}$$

Step 4: Evaluating the new solution with decision variables x_i^{k+1}, $i = 1, 2, \cdots, n$.

Step 5: If the new solution is better than $X^0 = [x_1^0, \cdots, x_n^0]$ or the predefined maximum iteration is reached, output the new solution as the result of the CLS, otherwise, let $k = k + 1$ and go back to *Step 2*.

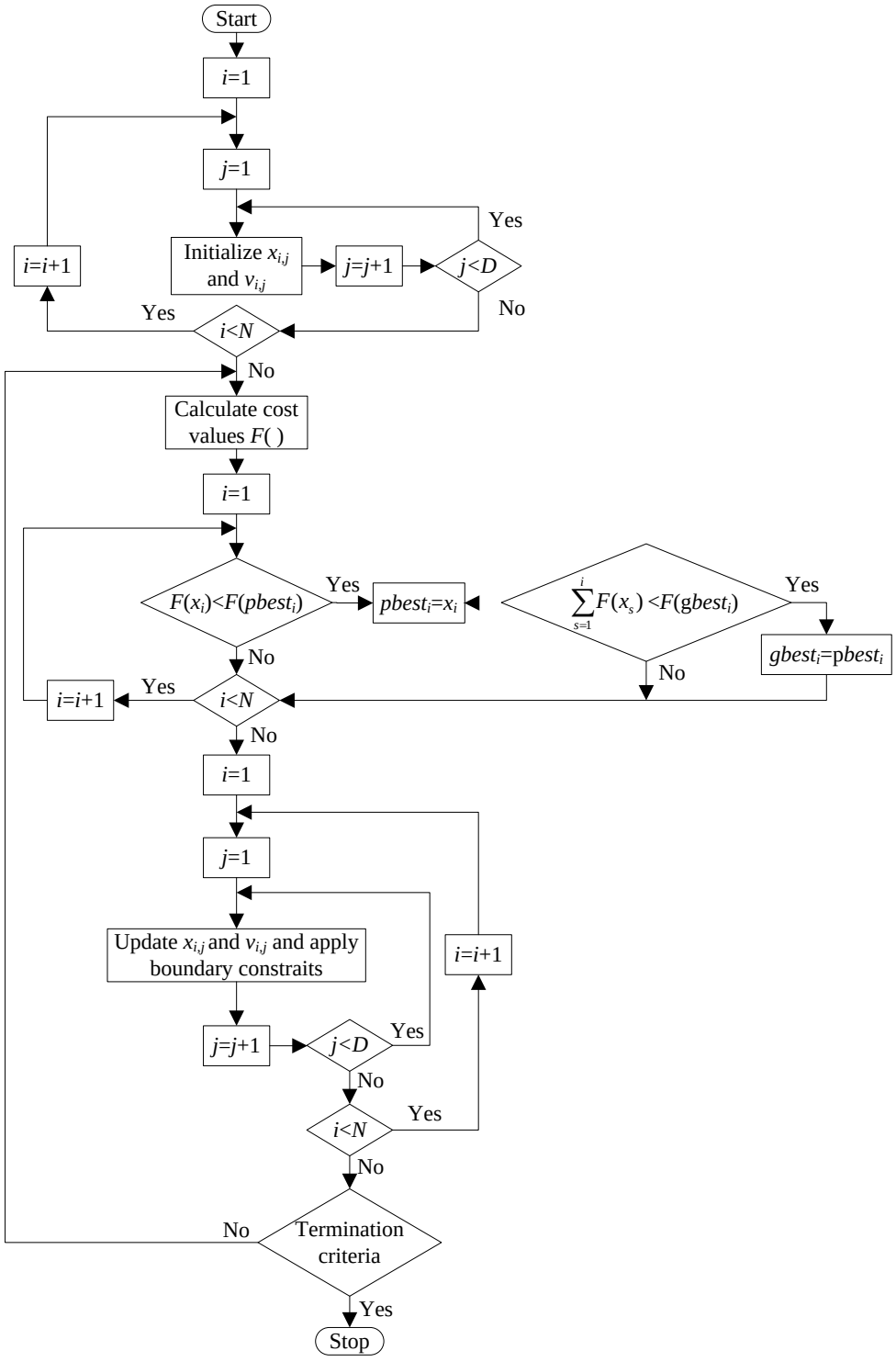


Fig. 4-7 Flow chart of the PSO algorithm.

(*Note: N denotes the number of the particles, D denotes the dimensions of searching space, and F is the cost function.)

Step 1: Set $k = 0$. For each particle i in the population:

Step 1.1: initialize x_i and v_i randomly.

Step 1.2: evaluate f_i.

Step 1.3: initialize *pbest* with the index of the particle with best function value among the population.

Step 1.4: initialize g*best* with a copy of x_i, $\forall i \le N$.

Step 2: Repeat until a stopping criterion is satisfied:

Step 2.1: find *gbest* such that $f(gbest) \le f_i$, $\forall i \le N$.

Step 2.2: for each particle i, $pbest = x_i$ if $f_i < f(pbest)$, $\forall i \le N$.

Step 2.3: for each particle i, update x_i and v_i.

Step 2.4: evaluate f_i for all particles.

Step 3: Reserve the top $N/5$ particles.

Step 4: Implement the CLS for the best particle, and update the best particle using the result of CLS.

Step 5: If a stopping criterion is satisfied, output the solution found best so far.

Step 6: Decrease the search space:

$$\min x_i = \max\left(\min x_i,\ x_i^k - r(\max x_i - \min x_i)\right), \qquad 0 < r < 1,$$

$$\max x_i = \min\left(\max x_i,\ x_i^k + r(\max x_i - \min x_i)\right), \qquad 0 < r < 1.$$

Step 7: Randomly generate $4N/5$ new particles within the decreased search space and evaluate them.

Step 8: Construct The new population Consisting of the $4N/5$ new particles and the old top $N/5$ particles in which the best particle is replaced by the result of CLS.

Step 9: Let $k = k + 1$ and go back to Step 2.

Fig. 4-8 Pseudo-code of CPSO algorithm.

Based on the PSO algorithm and the CLS, a two-phased iterative strategy named CPSO is proposed, in which PSO is applied to perform global exploration and CLS is employed to perform locally oriented search (exploitation) for the solutions resulted by PSO. The procedure of CPSO is described in Fig. 4-8.

4.5 Results and Discussion

To verify the optimal control method and the stability measure, the results of responses of the semi-active suspension and the robot are presented according to the system parameters shown in Table 4-1. First, to check the accuracy of the semi-active damper control model, it is tested by producing a sinusoidal response. Then, to evaluate real-time performance, it is discussed by comparing the step responses of the optimal controlled semi-active suspension with the passive suspension when a robot is performing starting maneuvers [98]. Finally, to investigate the dynamic stability, the vibratory response of the robot is shown while it moves at constant speed in a straight line in rough terrain. Roughness is simulated by Gaussian random disturbance.

When use CPSO to optimize the gain factor, the corresponding relationship between the damping force and condition function becomes not clear. Therefore, to check the semi-active damper control model, the CPSO is not used to optimize the gain factor σ and it is initially considered as a constant, set to 150. Further, to show the relationship more clearly, the sinusoidal input is used rather than random signal. Figure 4-9 shows the sinusoidal excitation input and time histories of the damping force, the condition function, and acceleration responses. In Figs. 4-9(b)–(d), the solid line denotes the damping force, relative velocity, and acceleration of the CG, respectively. The dashed line denotes the spring force, relative position, and acceleration caused by rough terrain, respectively. As shown in Figs. 4-9(b) and (c), the semi-active damper provides a weak damping force (ideally, a zero damping force) whenever the relative displacement and the relative velocity have the same sign. The damper is switched to the “on” state to partially cancel the spring force whenever the relative displacement and relative velocity have different signs. In addition, as shown in the partially enlarged view of Fig. 4-9(b), it is found that there is a possibility for chattering to occur in semi-active dampers when the damping coefficient switches between the “on” and “off” states. Figure 4-9(d) shows that the acceleration response curve of the damper does not reveal any jerks, indicating that the shaping function introduced into the control model successfully reduces any damper jerks of this type.

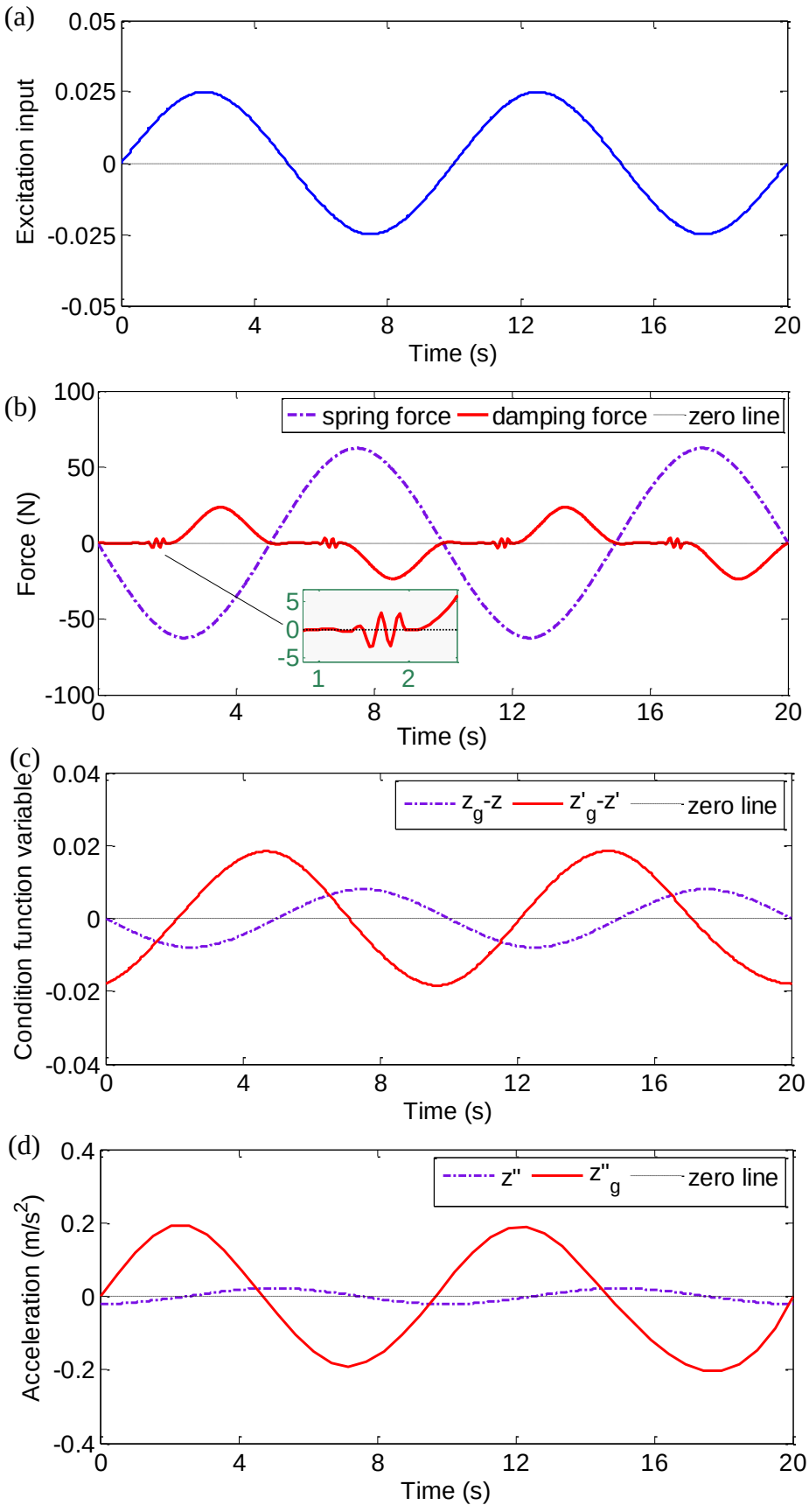


Fig. 4-9 Response of semi-active suspension. (a) sinusoidal input, (b) spring and damping forces, (c) condition function variables, and (d) accelerations.

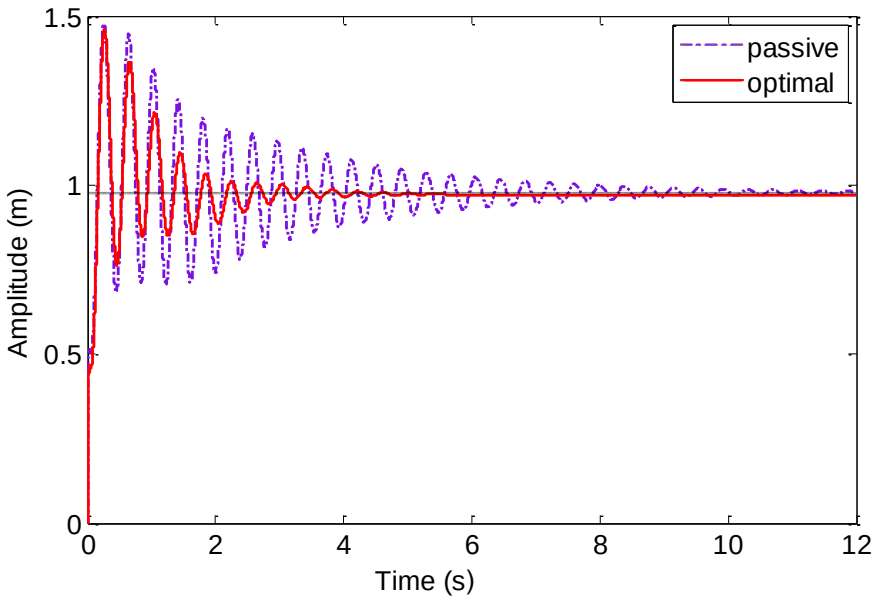


Fig. 4-10 Step responses of semi-active and passive suspensions.

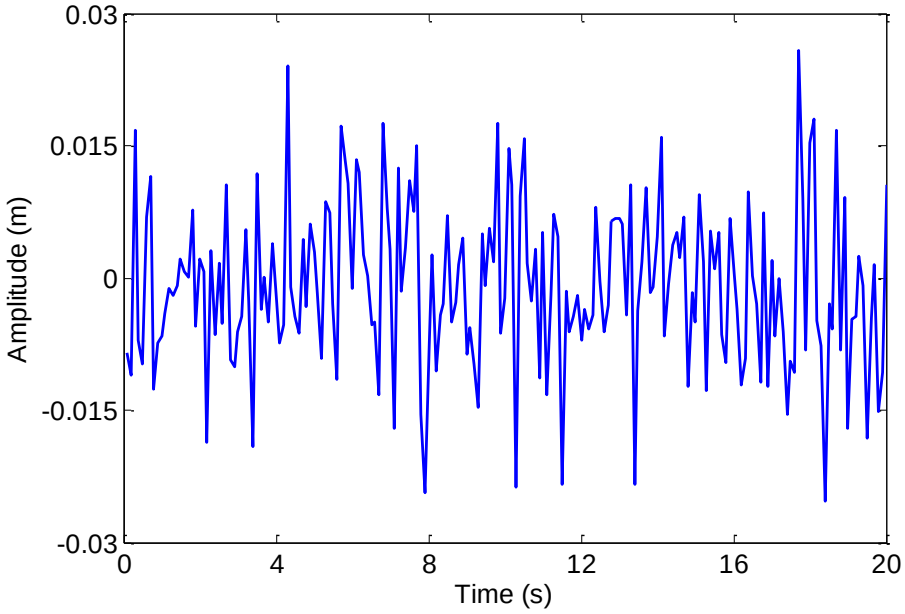


Fig. 4-11 Simulative pavement condition.

Actually, during the time from starting maneuvers to achieving uniform rectilinear motion, real-time performance becomes very important in practical applications, and controllers are required to quickly suppress any residual vibrations. To examine real-time performance, step responses of the optimal controlled

semi-active and passive suspensions are compared. Figure 4-10 shows the results of the step responses. The solid and dashed lines represent the results of semi-active and passive suspensions, respectively. As seen in Fig. 4-10, when the optimal controlled semi-active suspension is used to respond to the step excitation, residual vibrations are eliminated after 5 s, while residual vibrations begin to weaken within 12 s when a passive suspension is evaluated. This clearly shows that the real-time performance of semi-active suspensions when controlled by the proposed method is better than that of uncontrolled passive suspensions.

Another method to evaluate real-time performance is the phase plot, which depicts velocity versus acceleration as limit cycles, and the behavior of a suspension caused by a vertical vibration. This method was presented by Chen et al. [99]. The convergence speeds of the phase plot represent real-time performance. In general, the curves start from the initial point (0, $\boldsymbol{g}$), in which $\boldsymbol{g}$ is gravitational acceleration, and converge to the stable area around (0, 0) clockwise. It would be expected that the phase plot of the optimal controlled semi-active suspension converges faster than that of the passive suspension.

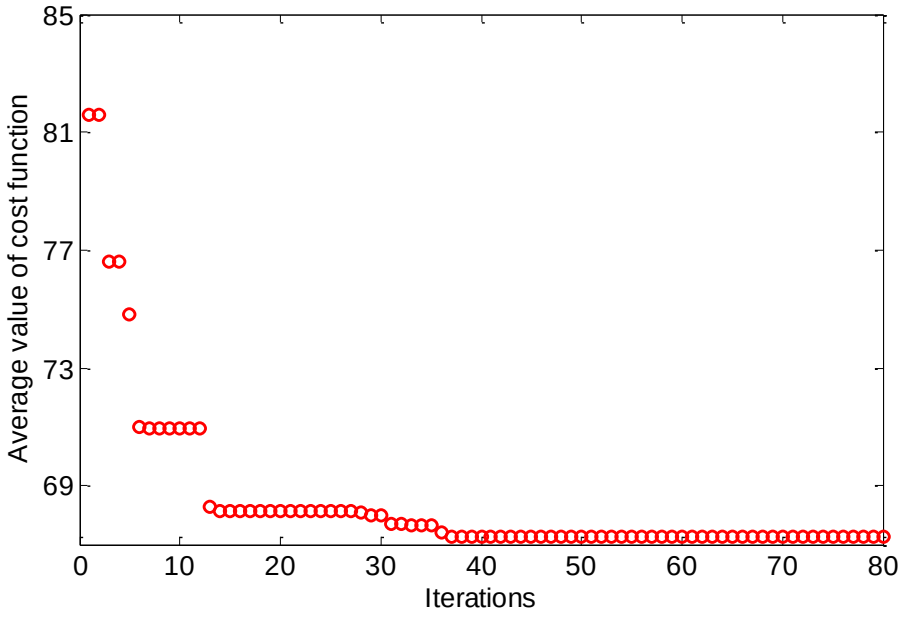


Fig. 4-12 Convergence of the cost function.

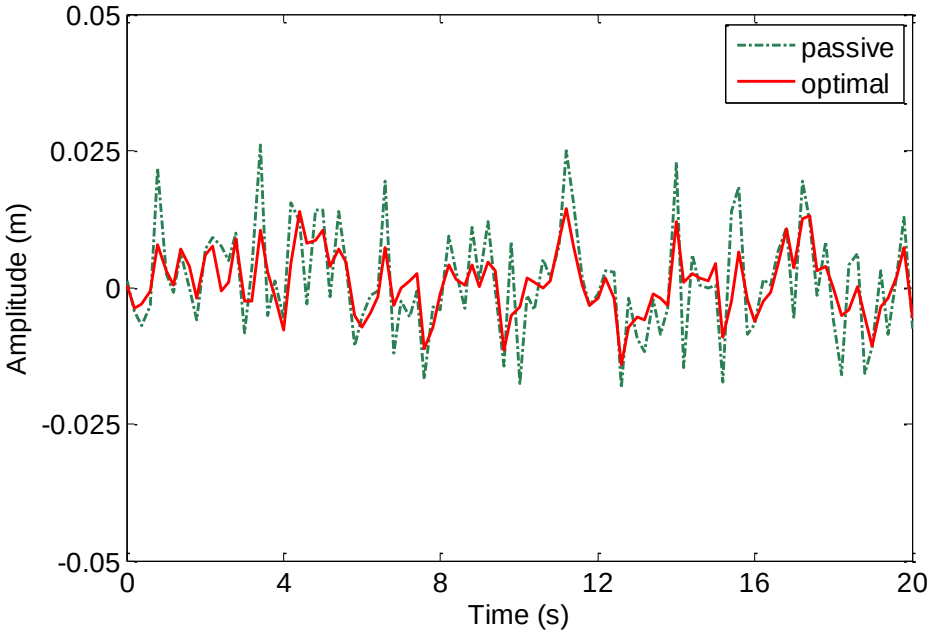


Fig. 4-13 Vibratory response of the robot in bounce motion.

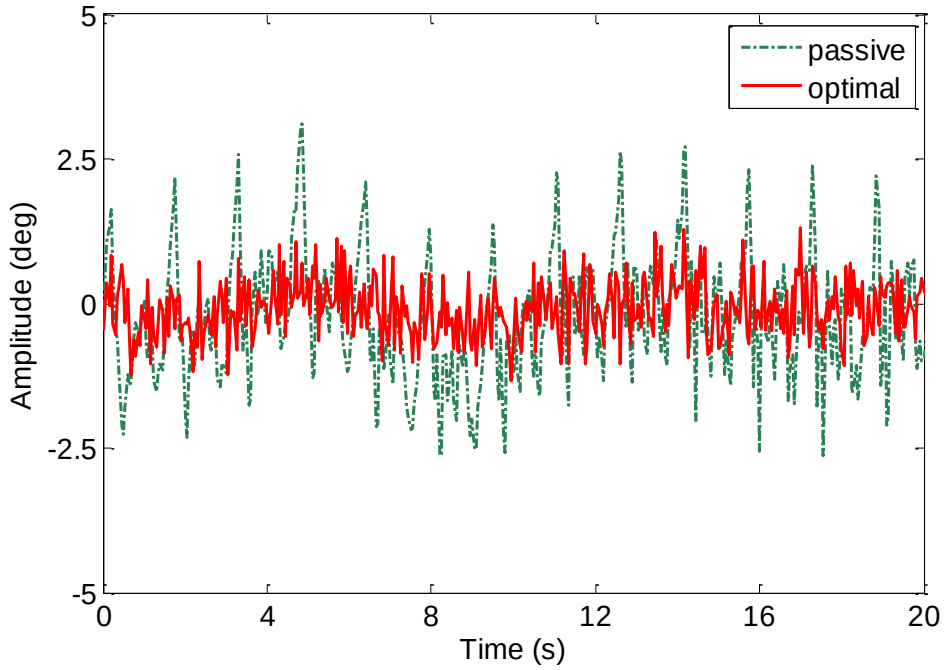


Fig. 4-14 Vibratory response of the robot in pitch motion.

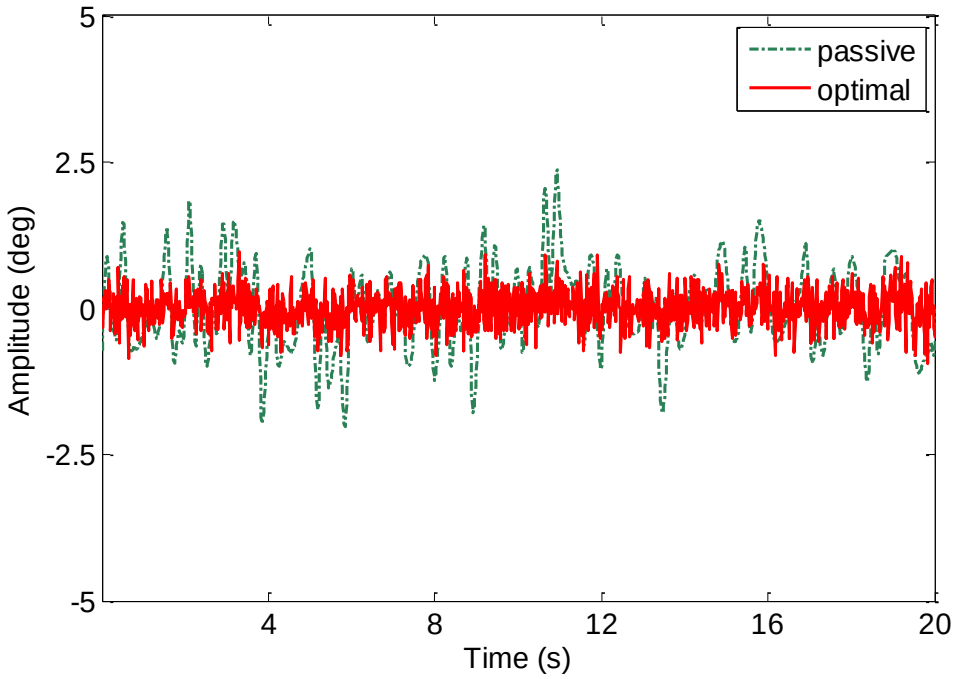


Fig. 4-15 Vibratory response of the robot in roll motion.

To investigate the dynamic stability measure and the optimal control method, the pavement condition is simulated using the Gaussian disturbance shown in Fig. 4-11. At this stage, the CPSO is employed to optimize the gain factor. The number of the particles used in the algorithm is 100, and the number of iterations is 80. Results obtained after applying the CPSO algorithm to generate the optimal gain factor based on minimizing the proposed cost function are shown in Figs. 4-12−4-15. Figure 4-12 shows a plot of the average value of the cost function versus the number of iterations during the optimization process. A gradual decrease in the average value of the cost function indicates a smooth convergence to a solution. Figures 4-13−4-15 show the vibratory responses of the robot in the bounce, pitch, and roll motions where it passes through rough terrain when the end effector does not carry a load. The three response curves of the passive suspension are superimposed to evaluate the performance of the proposed optimal control method. The solid line indicates the vibratory responses of the optimal controlled semi-active suspension, and the dashed line indicates the vibratory responses of the passive suspension. As shown in Figs. 4-13−4-15, the vibrations in the bounce, pitch, and roll motions for the optimal controlled semi-active suspension are less than those for the passive suspension. The results indicate that the

proposed control method improves the dynamic stability in the three directions when a robot passes through rough terrain.

4.6 Summary

The dynamic stability of mobile robots is a challenging problem and has attracted considerable attention since it is difficult to describe the vibration regulation by using a certain function, and therefore, suppressing random vibrations induced by rough terrain can be extremely difficult [100]. This chapter examined the problems of dynamic stability for a wheel-mounted robot equipped with two semi-active suspensions moving through rough terrain. To address the problems associated with random vibrations, this study departed from the dynamic model of the robot, and the vibration differential equations relating the variables of semi-active suspensions to a mobile robot stability measure have been written in a closed form. The control model for a semi-active MR damper was proposed by considering practical constraints of the damper and jerk elimination. The dynamic stability measure was developed by considering gravitational forces due to the weight of the robotic vehicle and the CG positional changes induced by rough terrain. Finally, the effectiveness of the stability measure criterion and the control approach in a rough terrain situation were examined. In contrast to passive suspensions, the proposed semi-active control scheme can suppress random vibrations and improve the dynamic stability of the robot while it moves through rough terrain. In order to improve the adaptability and practicability, a new-type teleoperation system is designed by using WiFi and virtual reality technologies in the following chapter.

Chapter 5 Teleoperation of Robots Based on Virtual Reality and WiFi

5.1 Introduction

Applications of robot systems can be significantly improved by implementing the teleoperation that means using wireless technology to control robot remotely. Currently, in the field of robotics, application of the teleoperation has been a benefit to humans by employing autonomous robots with the ability to explore unknown places, execute tasks in the environment with high potential risks like defuse explosives [101−103]. Because of this, it is necessary to develop teleoperation systems that allow a full control on the robot in real environment. Typically, Sawaragi et al. [104] investigated foundations needed in designing an interface system for robotic teleoperation. Hainsworth [105] presented a discussion of the requirements for user interfaces for teleoperation of mining vehicle systems. In order to get an intuitive user interface, this study adopts the virtual reality technology.

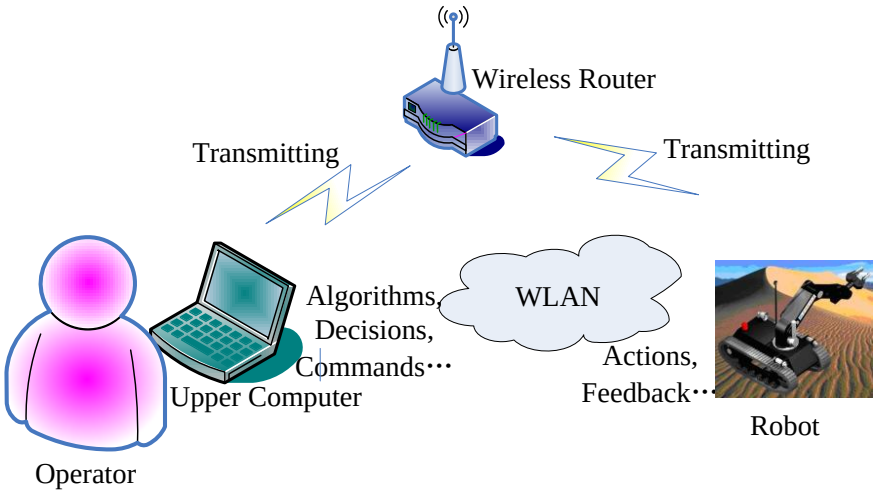


Fig. 5-1 Traditional teleoperation via WLAN.

Virtual reality technology is often used to establish the human-machine interface of teleoperation, then operators manipulate robots on the spot by using joysticks, mouse or keyboards directly [106]. Over the last few decades, the virtual reality technology has been widely adopted in robotics by researchers. Cui et al. [107] established a teleoperation system of mending robots based on virtual reality technology and WLAN. Hill [108] investigated the role of 3-D stereoscopic visualization in applications related to mobile robot teleoperation. However, in most of the studies concerning teleoperation, they regard the upper computer as a host and the lower computer as a slave. As shown in Fig. 5-1, the upper computer deals with the algorithm, decisions, and commands, while lower computer completes specific actions and sends feedback information. This may cause time delay and unpredictable errors introduced during communication between upper and lower computers [109, 110].

In Chapters 3 and 4, problems of the trajectory tracking and stability in rough terrain were addressed. Actually, an important factor affects the practical performance of these problems is teleoperation since the real time performance and errors during communication may degrade the practicability of robots. Therefore, some of the issues related to all the above considerations are necessary to be addressed in this chapter, and it intends to contribute to examining the role of virtual reality technology and WiFi in robot teleoperated applications.

The remainder of this chapter is organized as follows: in Section 5.2, the robotic system design is presented by introducing mechanical structure, and control system architecture is given. In Section 5.3, the kinematic mode of nonholonomic platform and the inverse kinematics of the manipulator are described. In Section 5.4, the teleoperation system is developed by using the virtual reality technology and WiFi. In Section 5.5, experiments are performed and the obtained results are discussed. Finally, in Section 5.6 concluding remarks are presented.

5.2 Robot System Design

The main configuration of the robotic system is an articulated manipulator mounted on a nonholonomic mobile platform. Two rear driving wheels and one front caster wheel support the platform. Independent motors actuate the two rear wheels, and the caster wheel is free to attain any orientation according to the motion of the robot. Figure 5-2 shows the robot prototype that allows users to test different techniques of control for the mobile platform and articulated manipulator. The mobile platform is assembled manually, and the articulated manipulator is a six-axis programmable robot arm, which is made by DENSO. In modelling the robot system, it is assumed that the wheels, platform, and each link of the manipulator are rigid.

5.2.1 Mechanical Structure

The structure of the robot can be decomposed into two parts: the articulated manipulator and the mobile platform. Figure 5-3 shows the schematic diagram of the robot system. As seen in Fig. 5-3(a) [111], the articulated manipulator has three joints and each joint can be rotated in the horizontal and the vertical planes. Therefore, the articulated manipulator has 6 degree-of-freedoms (DOFs). As seen in Fig. 5-3(b), the mobile platform must move in the direction of the axis of symmetry as the two rear driving wheels are not omni-directional wheels. In this case, the robot system is a nonholonomic system.

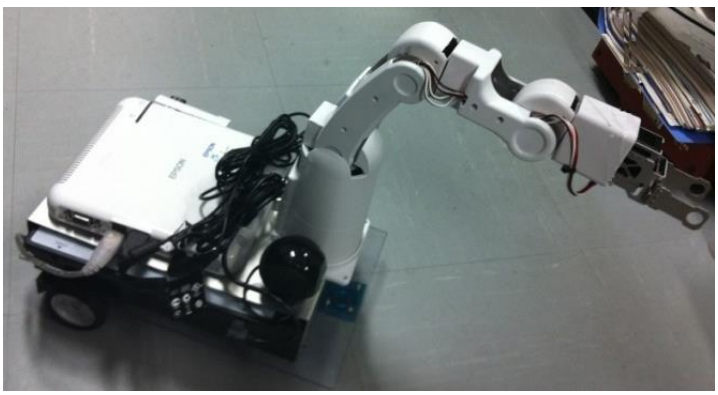


Fig. 5-2 Nonholonomic robot with an articulated manipulator.

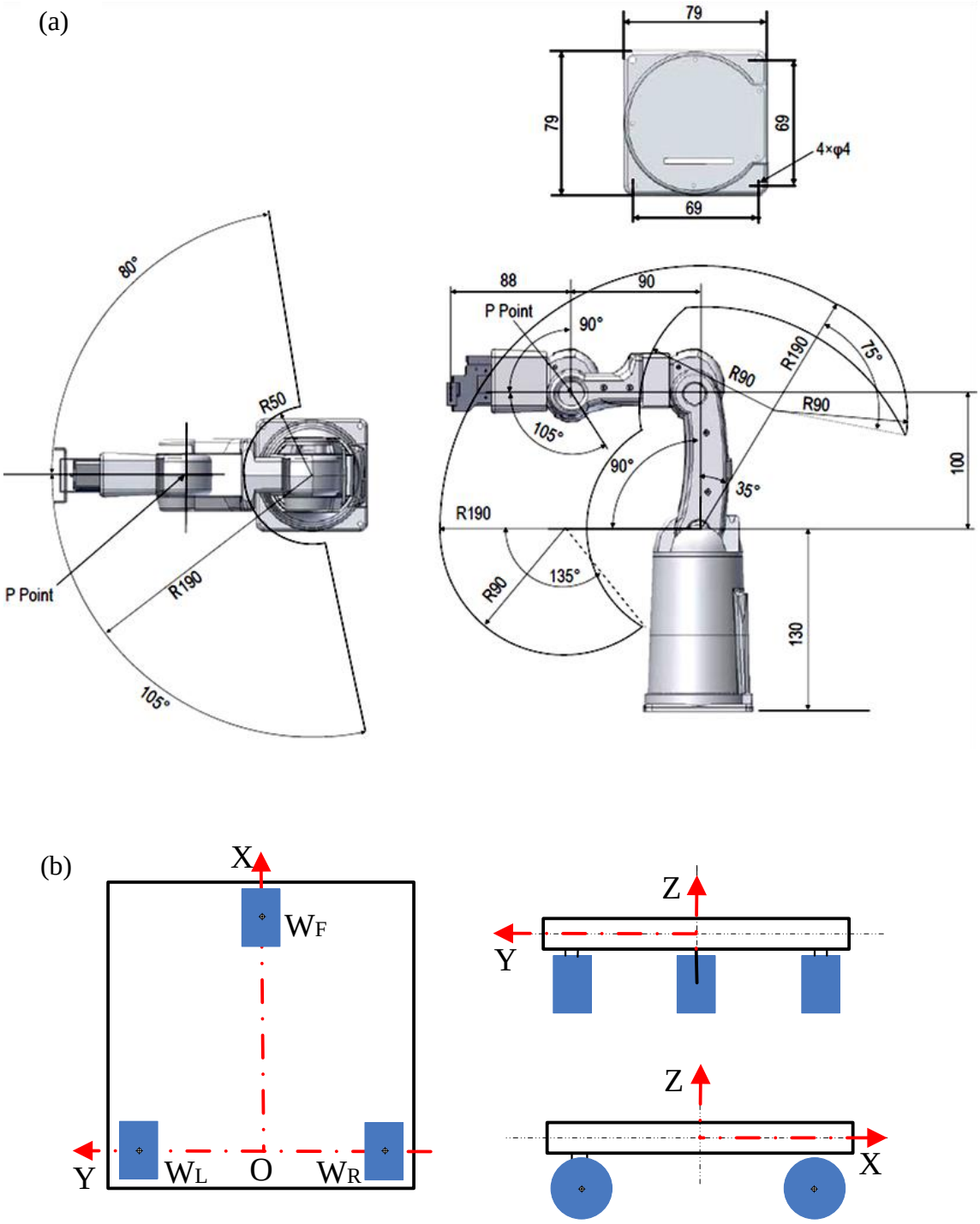


Fig. 5-3 Schematic diagram of the robot system.

(a) 6-DOF articulated manipulator, and (b) Wheeled mobile platform.

5.2.2 Control System Architecture

To attain good teleoperation effect, how to establish a virtual environment and how to transmit telecommands must be considered. To reduce the errors and time delay of communication, this chapter proposes a novel approach. As compared Fig. 5-4 with Fig. 5-1, the trick of the proposed approach is to hand off the processing of algorithms, decisions, and commands to a powerful lower computer, rather than relying on the upper computer.

5.3 Kinematics Analysis

5.3.1 Kinematic Model of Platform

Actually, the main problem of kinematics of robotic platform is the trajectory tracking which focuses on tracking a time-parameterized reference trajectory. Figure 5-5 shows the configuration of the robotic platform, where ϕ is the heading angle. The coordinate systems are defined as follows: O_g-$X_gY_gZ_g$ is an inertial base frame (global frame) fixed on the motion plane and O-XYZ is a local frame fixed on the robotic platform. The configuration of the path can be defined by the DOF of the robotic platform:

$$P = [x, \quad y, \quad \phi]^T . \tag{5-1}$$

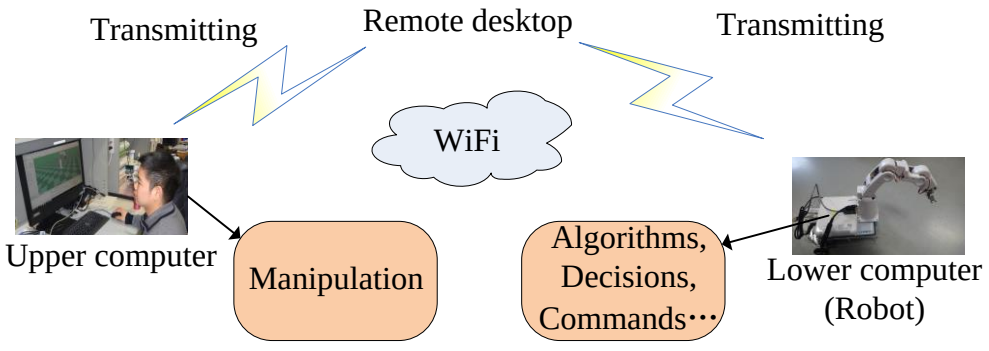


Fig. 5-4 Proposed teleoperation via WiFi.

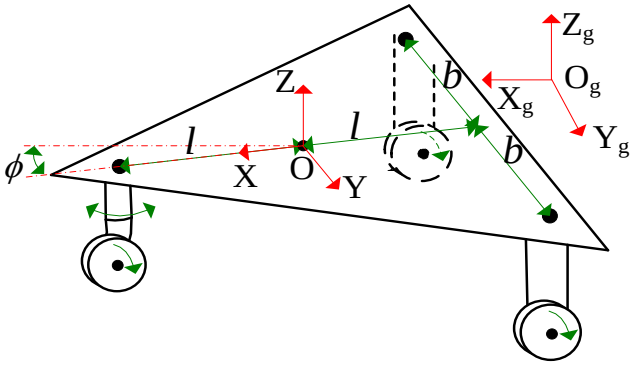


Fig. 5-5 Configuration of the robotic platform.

Therefore, considering the nonholonomic constraints [112] of a wheeled robot, the robot kinematics can be described by

$$\dot{P} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos\phi & 0 \\ \sin\phi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}, \tag{5-2}$$

where v and ω are the linear and angular velocities with respect to the inertial base frame, respectively.

A desired trajectory with velocities v_d and ω_d is considered, and the path configuration as $P_d = [x_d, \quad y_d, \quad \phi_d]^T$. In the local coordinate system with respect to the mobile robot system, the configuration error $P_e = [x_e, \quad y_e, \quad \phi_e]^T$ can be presented by

$$P_e = \begin{bmatrix} x_e \\ y_e \\ \phi_e \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_d - x \\ y_d - y \\ \phi_d - \phi \end{bmatrix}. \tag{5-3}$$

By differentiating Eq. (5-3) and substituting Eq. (5-2) into the result, it can be obtained as

$$\begin{cases} \dot{x}_e = (v_d \cos\phi_d - v\cos\phi)\cos\phi - \omega(x_d - x)\sin\phi \\ \qquad + (v_d \sin\phi_d - v\sin\phi)\sin\phi - \omega(y_d - y)\cos\phi, \\ \dot{y}_e = -(v_d \cos\phi_d - v\cos\phi)\sin\phi - \omega(x_d - x)\cos\phi \\ \qquad + (v_d \sin\phi_d - v\sin\phi)\cos\phi - \omega(y_d - y)\sin\phi, \\ \dot{\phi}_e = \omega_d - \omega. \end{cases} \tag{5-4}$$

Considering Eq. (5-3), the time derivative of the configuration error for the robot is

$$\dot{P}_e = \begin{bmatrix} \dot{x}_e \\ \dot{y}_e \\ \dot{\phi}_e \end{bmatrix} = \begin{bmatrix} \omega y_e - v + v_d \cos\phi_e \\ -\omega x_e + v_d \sin\phi_e \\ \omega_d - \omega \end{bmatrix}. \tag{5-5}$$

5.3.2 Kinematic Model of Manipulator

Kinematics of manipulator falls into two categories, one is the forward kinematics which refers to the use of kinematic equations to compute the position of the end-effector from specified values of the joint parameters, and the other is inverse kinematics which refers to the use of kinematics equations to determine the joint parameters that provide a desired position of the end-effector. In this subsection, first the brief kinematics analysis for 6-DOF manipulator is presented, then a 3-link planar manipulator is simulated by using Matlab, because it is difficult to get the visualized 3-D plots for 6-DOF manipulator which is presented in this chapter.

Based on Denavit-Hartenberg (D-H) approach, the D-H matrix of the manipulator can be derived as follows:

$${}^{i-1}H_i = \begin{vmatrix} \cos\theta_i & -\sin\theta_i \cos\theta_{i+1} & \sin\theta_i \sin\theta_{i+1} & l_i \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cos\theta_{i+1} & -\cos\theta_i \sin\theta_{i+1} & l_i \sin\theta_i \\ 0 & \sin\theta_{i+1} & \cos\theta_{i+1} & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}, \tag{5-6}$$

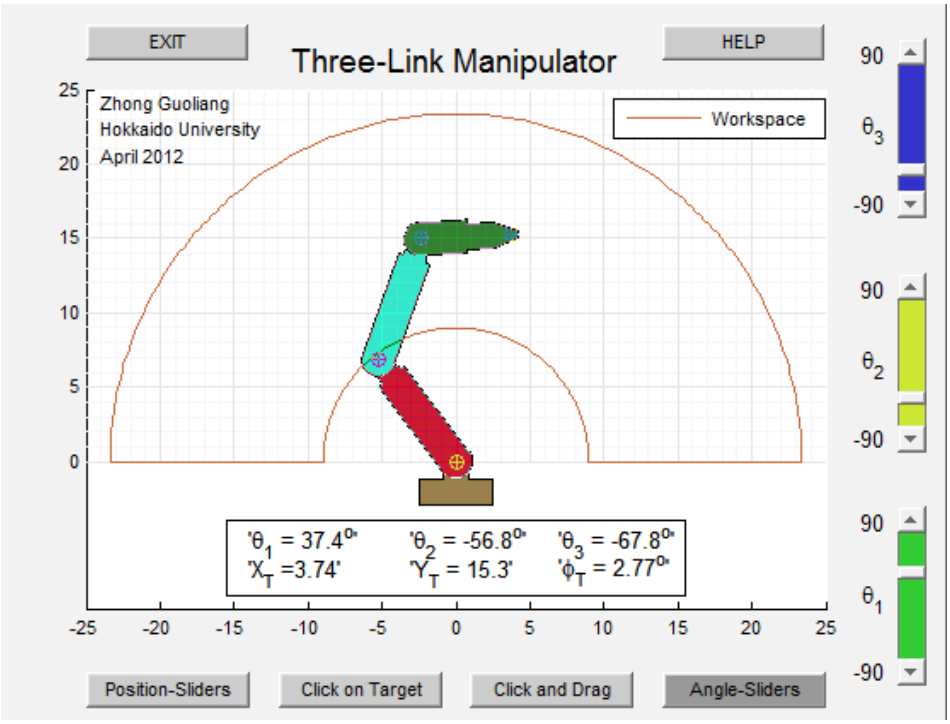
where l_i and θ_i are the length and the rotational angle of the ith link, respectively. For the 6-DOF manipulator, the homogeneous transformation matrix between the coordinate of end-effector and the base coordinate system can be obtained as

$${}^0T_6 = {}^0H_1\,{}^1H_2\,{}^2H_3\,{}^3H_4\,{}^4H_5\,{}^5H_6 = \begin{vmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{vmatrix}, \tag{5-7}$$

where n_x, n_y, n_z, o_x, o_y, o_z, a_x, a_y, and a_z are the direction cosines of the coordinate of end-effector with respect to the base coordinate system, respectively; p_x, p_y, and p_z are the values of the origin of end-effector in the base coordinate system. This is the kinematic equation of the manipulator, which shows the relationship between the pose of end-effector and the rotational angle of each joint. For Eq. (5-7), if variables in the

left side are known and variables in the right side are expected to solve, this is the forward kinematics problem. Contrarily, it is the inverse kinematics problem.

(a)



(b)

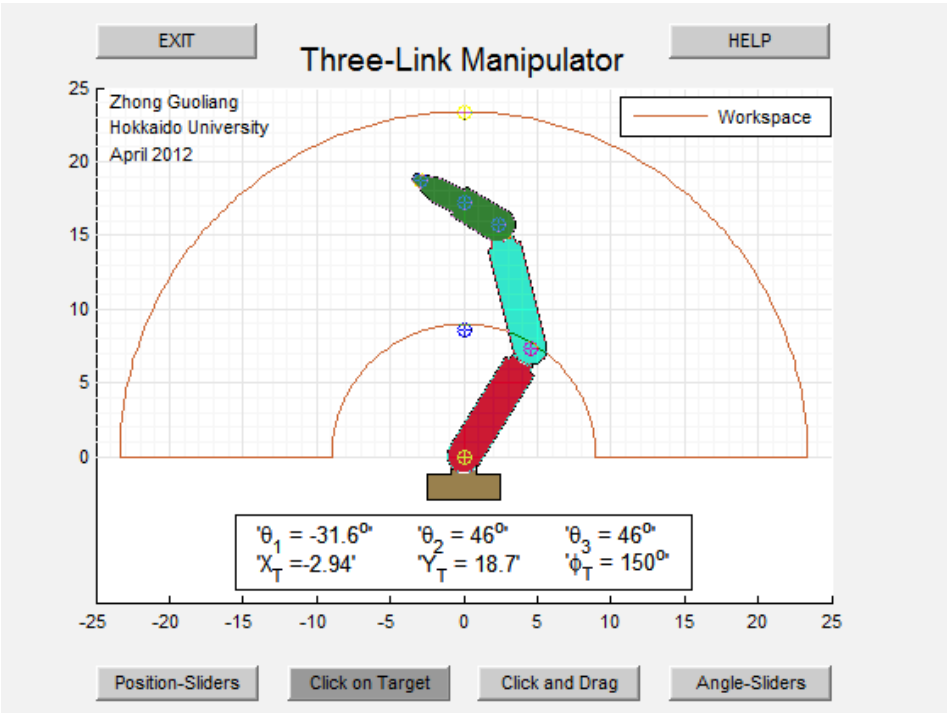


Fig. 5-6 Kinematics simulation of 3-link planar manipulator.

(a) Forward kinematics, and (b) Inverse kinematics.

A 3-link planar manipulator is used to simulate the forward and inverse kinematics for clarity as shown in Fig. 5-6. Figure 5-6(a) reveals the solution of forward kinematics. When the rotational angles of each joint are inputted by dragging the slide blocks, the position of end-effector is determined. Figure 5-6(b) reveals the solution of inverse kinematics. When the position of end-effector is given by clicking in the workspace of manipulator, the rotational angles of each joint will be computed.

5.4 Teleoperation Based on the Virtual Reality Technology via WiFi

The teleoperation system consists of virtual environment and wireless communication. The virtual environment is developed based on virtual manipulator by using WINCAPS III software, which is provided by DENSO [111], which is an offline programming software enables users to conveniently program a robot from a remote PC without operating the robot. The wireless communication is built by accessing the remote desktop via WiFi connection.

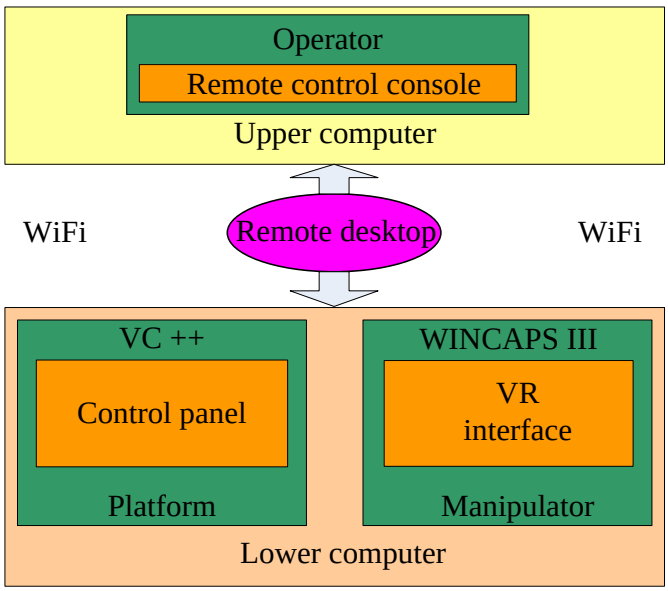


Fig. 5-7 Scheme of the proposed teleoperation system.

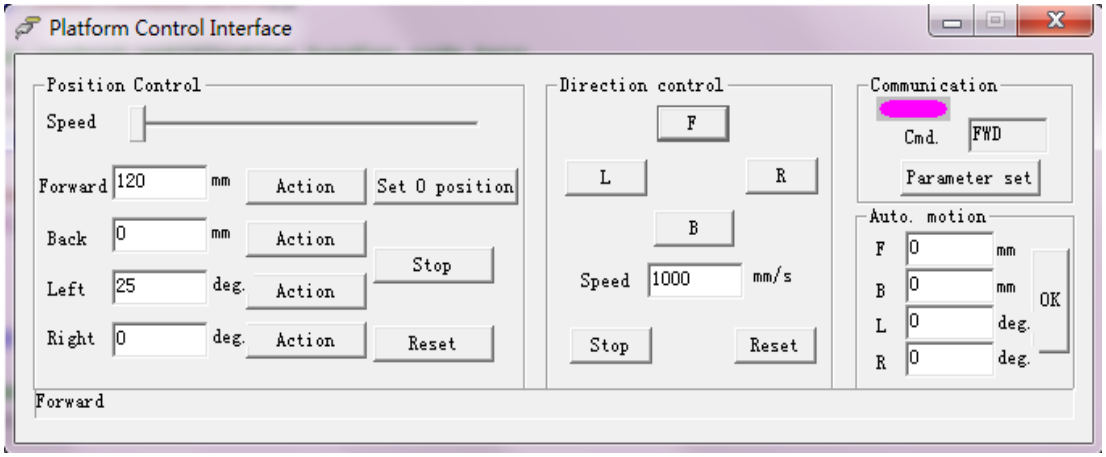


Fig. 5-8 Control panel of the robotic platform.

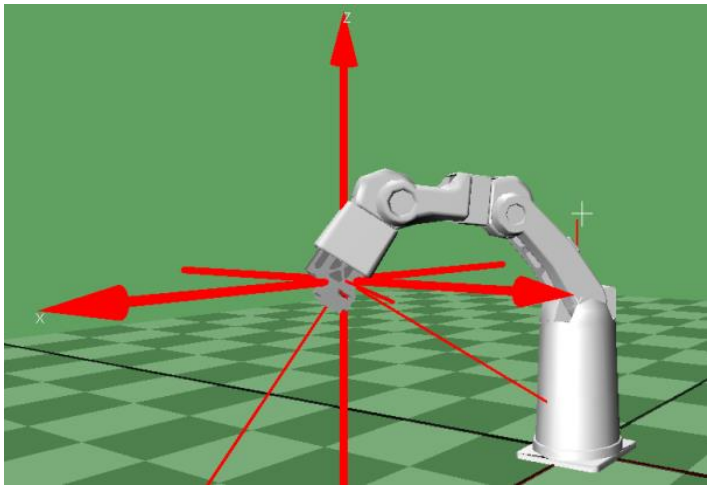


Fig. 5-9 Manipulator in virtual environment.

Figure 5-7 shows the scheme of the proposed teleoperation system. In the lower computer, the VC++ is used to develop a control panel for controlling the robotic platform, and WINCAPS III to establish a virtual reality interface for controlling the manipulator. Then the remote desktop of the lower computer is accessed from the upper computer via WiFi. On the upper computer, operator can manipulate the robot by using the received remote control console.

Figure 5-8 shows the control panel of robotic platform. As shown in Fig. 5-8, the distance, speed, and direction of movement of the robot can be defined by inputting the commands on the panel. The commands of manipulator motion can be inputted through the virtual manipulator in the virtual environment as shown in Fig. 5-9.

5.5 Experimental Results

Based on the kinematic model of robotic platform and the kinematics of manipulator, the teleoperation system is performed on the experimental robot to grasp a ping-pong ball indoors. In this experimental robot, the lower computer is an Endeavor NP30S Mini-PC with usb WiFi adapter, the robot arm is a VE026A articulated manipulator with 6-DOF. The upper computer is a commonly-used computer.

Figure 5-10 shows the motion trace of the robot when it comes in the door and searches for the Ping-Pong ball. At this time, the robot is controlled by the command received from the control panel. Figure 5-11 shows the poses of the manipulator during grasping movement. At this time, the manipulator is teleoperated by virtual manipulator which is transmitted from the lower computer to upper computer via WiFi. Figure 5-12 shows the poses of the virtual manipulator in virtual environment during grasping movement. It can be found from Figs. 5-11 and 5-12, the poses of the real manipulator is consistent with the virtual manipulator, which implies that it is feasible to control the manipulator remotely by using the teleoperation system.

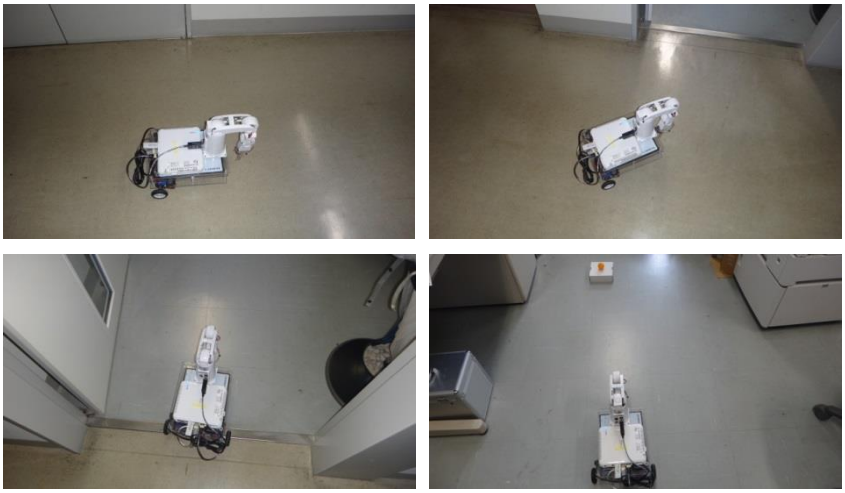


Fig. 5-10 Robot moves from outside to the target in a room.

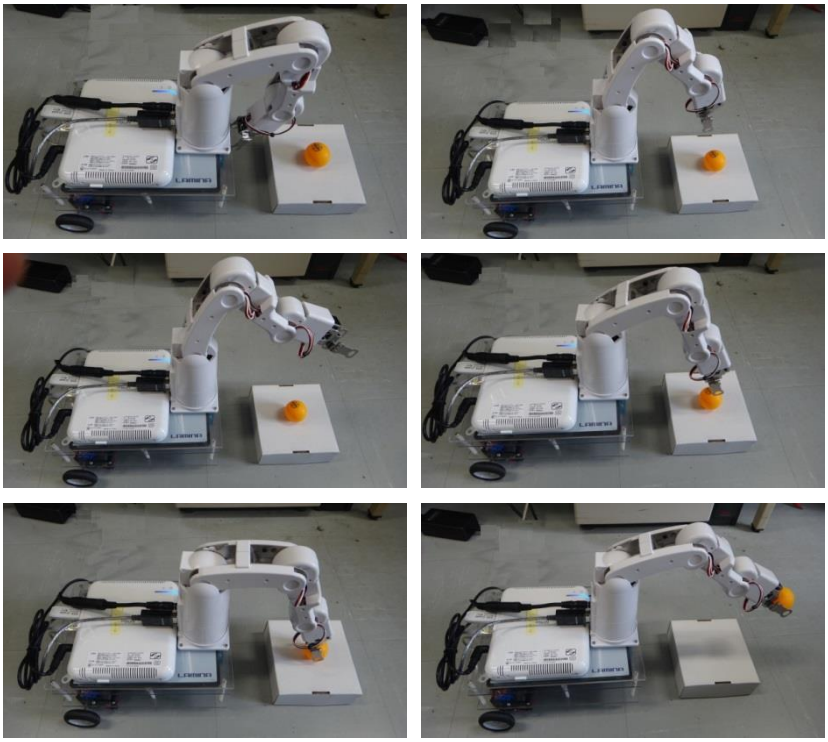


Fig. 5-11 Manipulator grasps the ping-pong ball.

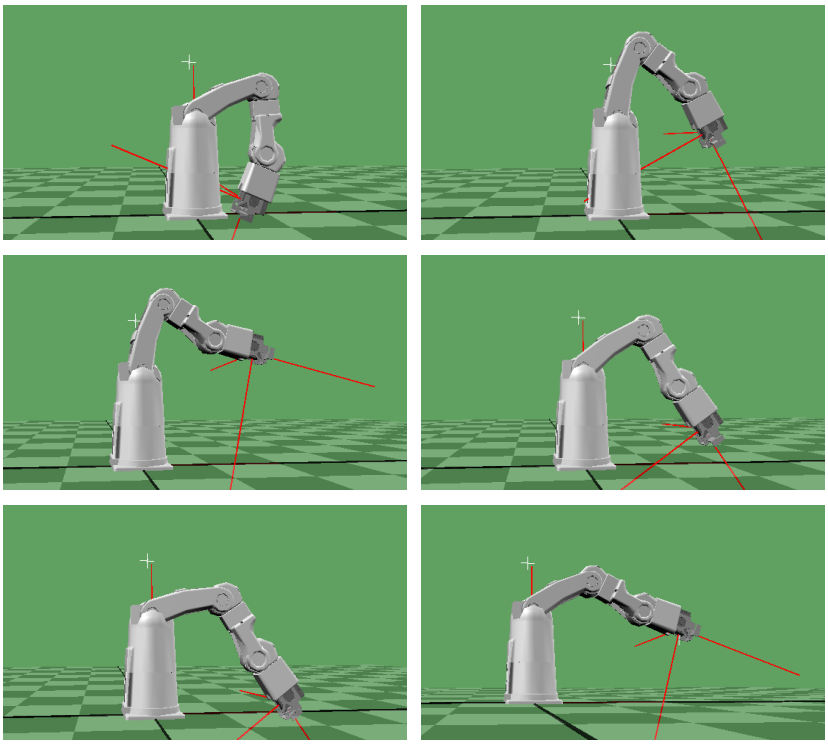


Fig. 5-12 Motion of the virtual manipulator when it grasps the ping-pong ball.

5.6 Summary

Teleoperated robots are expected to do more service for humans. It helps to solve tasks where the human can be in danger or reach inaccessible places. This chapter showed that the proposed teleoperation system was feasible to manipulate mobile robot remotely. Such teleoperation system used virtual reality technology to establish the intuitive user interface based on WINCAPS III, and then transmited the commands and remote desktop service from the lower computer to the upper computer via WiFi. The inverse kinematics of manipulator was described and the kinematic model of mobile platform was derived considering nonholonomic constraints. To prove the effectiveness of the proposed teleoperation system, some experiments were performed based on the proposed method. The results confirmed that the mobile robot can be remotely manipulated by operator using proposed teleoperation scheme.

Chapter 6 Conclusions and Further Works

6.1 Conclusions

The study of this thesis focused on a suspended wheeled mobile robot and investigated the vibration suppression, tracking control, stability improvement, and teleoperation of the robot.

In regard to vibration suppression, methods were investigated to reduce vibration for a mobile robot with a suspension system considering the center-of-gravity (CG) shift. To this end, an explicit dynamic model involving the incorporation of the CG shift and suspension characteristics was formulated. A control scheme consisting of multi-input shaping and feedback control was provided, and an optimization technique called particle swarm optimization (PSO) was employed to optimize the shapers. The objectives of the optimal control were to minimize residual vibration as well as settling time. Other objectives such as improved robustness and saturation limits of the input torques were also incorporated in the optimization scheme. For the numerical study, comparisons among the presented scheme, zero vibration derivative (ZVD) shaping, and unshaped model and analyses of the results show that the optimal multi-input shaping results in input torques with shorter settling times and outperforms the ZVD shaping both in vibration reduction and in robustness. This broadens the range of acceptable performance, thus allowing more room for modeling errors and parameter variation. These improvements can be ascribed to the more elaborate cost function. The results of the simulations verified that the optimization is effective to suppress vibration. Further, the characteristics of the motion can be simply visualized by inspecting the velocity and motion trace and discussing the results based on that.

In addition, the problem of minimum time-jerk trajectory generation was studied to reduce the vibration when the manipulator and the platform move simultaneously. Considering the interaction effect between the platform and manipulator, the direct

path method (DPM) concept was introduced to derive a dynamic model of the complicated system. To generate the optimal trajectory, the chaotic particle swarm optimization (CPSO) method was applied to optimize the parameters of the inter-knots of joint, which were used for polynomial interpolation. Embedding chaos into the original PSO can prevent the local minimum and obtain the global optimal solution. The optimal trajectories considering the minimum execution time and minimax jerks were obtained after optimization. The obtained results illustrated that the vibrations of the suspended platform can be suppressed when the manipulator moves following the minimum time-jerk trajectory. Also, the proposed optimization algorithm was effective in the search for the time-jerk synthetic optimal trajectory.

In regard to tracking control, a dynamic model considering the interaction between platform and manipulator were constructed by introducing the DPM concept. To track the desired trajectory accurately when the wheeled mobile robot moves in the presence of modeling uncertainty such as friction and external disturbance, a fuzzy compensator was proposed to counteract the friction. In addition, a robust term was added to reduce the approximation error and ensure the system stability when the platform subjects to the external disturbance. In the end, the numerical simulations for the robot under the influence of friction and external disturbance were performed. The results demonstrated the effectiveness of the proposed control scheme in the trajectory tracking of platform and manipulator simultaneously.

In regard to stability improvement and teleoperation, the contribution of this study is twofold: concept and design. Conceptually, this study proposed a novel idea to measure the dynamic stability. The method distinguishes itself from others due to the fact that it has the following advantages: 1) The dynamic model of suspended wheeled robot considers the bounce, pitch, and roll motions, and vibrations result from these motions can be completely detected by using the measure method. 2) The practical control model for semi-active damper can reduce the undesirable jerk caused by the discontinuity of damping force at the switching time. 3) The stability margin measure is top-heavy sensitive so that the measuring criterion is valid even if the end-effector carries loads. Accordingly, the optimal control based on the criterion is robust to end-effector loads.

Design-wise, this study designed an optimal controller of semi-active suspension. The effectiveness was demonstrated by examining the control method in difference situations. In contrast to passive suspensions, an interesting observation is that the proposed method can adequately suppress the vibrations and improve the stability when the semi-active suspended robot moves through rough terrain. In addition, a virtual control system was developed for teleoperation. Such teleoperation system uses virtual reality technology to establish the intuitive user interface, and then transmits the commands and remote desktop service from lower computer to upper computer via WiFi. Experiments show that the mobile robot can be remotely manipulated by operator using proposed virtual control.

In principle, semi-active suspension system can deliver the versatility, adaptability and higher performance of fully active systems for a fraction of the power consumption, and the virtual control can provide a visualized interface for teleoperation. From this point of view, the conclusions have certainly practical significance for the design of reasonable control field robots.

6.2 Further Works

This thesis studied the problems of suspended wheeled robots moving in rough terrain. The ideas can be extended to many applications such as field robots or lunar rovers.

In the further study, it will be more complete if the front wheel of the suspended wheeled mobile robot is controllable and some sensors are mounted on the robot to detect the road condition. In this case, the parameters of the road can be obtained in advance so that the vibration can be controlled in real time. In addition, it is an research focus of robotics that researchers improve the vision system by using 3-D depth camera (such as Microsoft Kinect) to orientate the robot and map the external environment, also an emerging broadband wireless access technology called WiMAX is desired to use so that the operators can manipulate the robot several kilometres away.

References

[1] Yamamoto, Y., Yun, X.: Effect of the dynamic interaction on coordinated control of mobile manipulators. *IEEE Trans. Robot. Auto.* **12**, 816−824 (1996).

[2] Yu, Q., and Chen, I.M.: A General Approach to the Dynamics of Nonholonomic Mobile Manipulator Systems. *ASME J. Dyn. Syst., Meas., Control* **124**, 512–521 (2002).

[3] Iagnemma, K., Dubowki, S.: Traction control of wheel mobile robots in rough terrain with applications to planetary rovers. *Int. J. Robot. Res.* **23**, 1029−1040 (2004).

[4] Dong, W.: On trajectory and force tracking control of constrained mobile manipulators with parameter uncertainty. *Automatica* **38**, 1475−1484 (2002).

[5] Xu, D., Zhao, D., Yi, J., Tan, X.: Trajectory tracking control of omnidirectional wheeled mobile manipulators: robust neural network-based sliding mode approach. *IEEE Trans. Syst. Man Cybern. B: Cybern.* **39**, 788−799 (2009).

[6] Ghasempoor, A., Sepehri, N.: A Measure of Machine Stability for Moving Base Manipulators. In: *Proc. IEEE Int. Conf. Robotics and Automation* (ICRA), Nagoya, Japan, pp. 2249−2254 (1995).

[7] Online: *http://www.mobilerobots.com/*.

[8] Saha, S.K., Angeles, J.: Kinematics and dynamics of a three-wheeled 2-DOF AGV. In: *Proc. IEEE Int. Conf. Robotics and Automation* (ICRA), Scottsdale, USA, pp. 1572–1577 (1989).

[9] Meghdari, M., Durali, M., Naderi, D.: Investigating dynamic interaction between the one d.o.f. manipulator and vehicle of a mobile manipulator. *J. Intel. Robot. Syst.* **28**, 277−290 (2000).

[10] Moosavian, S.A., Papadopoulos, E.: Explicit dynamics of space free-flyers with multiple manipulators via spacemaple. *Adv. Robot.* **18**, 223−244 (2004).

[11] Liu, H., Nonami, K., Hagiwara, T.: Semi-active fuzzy sliding mode control of full vehicle and suspensions. *J. Vib. Control* **11**, 1025−1042 (2005).

[12] Moosavian, S.A.A., Alipour, K.: On the dynamic tip-over stability of wheeled

mobile manipulators. *Int. J. of Robot. Auto.* **22**, 322−328 (2007).

[13] Korayem, M.H., Nikoobin, A., Azimirad, V.: Maximum load carrying capacity of mobile manipulators: optimal control approach. *Robotica* **27**, 147−159 (2009).

[14] Zhong, G., Kobayashi, Y., Emaru, T., Hoshino, Y.: Vibration reduction of mobile robot with suspension system using optimal multi-input shaping. In: *Proc. 14th Asia Pacific Vibration Conference* (APVC), HongKong, China, pp. 1981−1990 (2011).

[15] Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: *Proc. IEEE Int. Conf. Neural Networks* (ICNN), Perth, Australia, pp. 1942−1948 (1995).

[16] Milam M.B.: Real-time optimal trajectory generation for constrained dynamical systems. *Ph.D. Thesis*, California Institute of Technology, Pasadena, USA (2003).

[17] Piazzi, A., Visioli, A.: Global Minimum-Jerk Trajectory Planning of Robot Manipulator. *IEEE Trans. Ind. Electron.* **47**, 140−149 (2000).

[18] Lombai, F., Szederkenyi, G.: Throwing motion generation using nonlinear optimization on a 6-degree-of-freedom robot manipulator. In: *Proc. IEEE Int. Conf. Mechatronics* (ICM), Malaga, Spain, pp. 14−19 (2009).

[19] Gasparetto, A., Zanotto, V.: A new method for smooth trajectory planning of robot manipulators. *Mech. Mach. Theory* **42**, 455–471 (2007).

[20] Gasparetto, A., Zanotto, V.: A technique for time-jerk optimal planning of robot trajectories. *Robot. Comput. Integr. Manuf.* **24**, 415–426 (2008).

[21] Njah, A.N.: Trajectory control and synchronization of the new hyperchatic Liu system via backstepping techniques. *Nonlinear Dyn.* **61**, 1−9 (2010).

[22] Hou, Z.G., Zou, A.M., Cheng, L., Tan, M.: Adaptive control of an electrically driven nonholonomic mobile robot via backstepping and fuzzy approach. *IEEE Trans. Control Syst. Technol.* **17**, 803−815 (2009).

[23] Kim, D.H., Oh, J.H.: Tracking control of a two-wheeled mobile robot using input-output linearization. *Control Eng. Practice*. **7**, 369–373 (1999).

[24] Oya, M., Su, C.Y., Katoh, R.: Robust adaptive motion/force tracking control of uncertain nonholonomic mechanical systems. *IEEE Trans. Robot. Auto.* **19**, 175–181 (2003).

[25] Das, T. Kar, I.N.: Design and implementation of an adaptive fuzzy logic-based

controller for wheeled mobile robots. *IEEE Trans. Control Syst. Technol.* **14**, 501–510 (2006).

[26] Fukao, T., Nakagawa, H., Adachi, N.: Adaptive tracking control of a nonholonomic mobile robot. *IEEE Trans. Robot. Auto.* **16**, 609–615 (2000).

[27] Karnopp, D.C., Crosby, M.J., Harwood, R.A.: Vibration control using semi-active force generators. *J. Eng. Ind.* (ASME) **96**, 619–626 (1974).

[28] Liu, Y., Waters, T.P., Brennan, M.J.: A comparison of semi-active damping control strategies for vibration isolation of harmonic disturbances. *J. Sound. Vib.* **280**, 21−39 (2005).

[29] McGhee, R.B., Iswandhi, G.I.: Adaptive locomotion of a multilegged robot over rough terrain. *IEEE Trans. Syst. Man Cybern. B: Cybern.* **9**, 176−182 (1979).

[30] Sugano, S., Huang, Q., Kato, I.: Stability criteria in controlling mobile robotic systems. In: *Proc. IEEE/RSJ Int. Workshop on Intelligent Robots and Systems* (IWIRS), Yokohama, Japan, pp. 832−838 (1993).

[31] Davidson, J.K., Schweitzer, G.: A mechanics-based computer algorithm for displaying the margin of static stability in four-legged vehicles. *Trans. ASME J. Mech. Design* **112**, 480−487 (1990).

[32] Korayem, M.H., Ghariblu, H.: Maximum allowable load on wheeled mobile manipulators imposing redundancy constraints. *Robot. Autonom. Syst.* **44**, 151−159 (2003).

[33] Korayem, M.H., Azimirad, V., Nikoobin, A.: Maximum load carrying capacity of autonomous mobile manipulator in an environment with obstacle considering tip over stability. *Int. J. Adv. Manuf. Technol.* **46**, 811−829 (2010).

[34] Cristancho, C.A., Angelica, F.C., Rodriguez, H.L.: Holonomic platform and six-DOF manipulator robot through virtual reality devices. In: *Proc. IEEE Colombian Conf. Automatic Control and Industry Applications* (CCACIA), Colombian, USA, pp. 37−42 (2011).

[35] Livatino, S., Muscato, G., Privitera, F.: Stereo viewing and virtual reality technologies in mobile robot teleguide. *IEEE Trans. Robot.* **25**, 1343−1355 (2009).

[36] Sgouros, N.M., Gerogiannakis, S.: Robot teleoperation environments featuring

WAP-based wireless devices. *J. Network. Comput. Appl.* 26, 259−271 (2003).

[37] Kelly, A., Chan, N., Herman, H., Huber, D., Meyers, R., Rander, P., Warner, R., Ziglar, J., Capstick, E.: Real-time photorealistic virtualized reality interface for remote mobile robot control. *Int. J. Robot. Res.* **32**, 384−404 (2013).

[38] Yin, H., Kobayashi, Y., Emaru, T., Hoshino, Y.: Decomposed dynamic control for flexible manipulator in consideration of nonlinearity. *J. Syst. Design. Dyn.* **5**, 219−230 (2011).

[39] Gurleyuk, S.S., Cinal, S.: Robust three-impulse sequence input shaper design. *J. Vib. Control* **13**, 1807−1818 (2007).

[40] Alatas, B., Akin, E., and Ozer, A.B.: Chaos embedded particle swarm optimization algorithms. *Chaos, Solotons. Fract.* **40**, 1715−1734 (2009).

[41] Marko, L., Klancar, G., Matko, D., Potocnik, B.: Time optimal path planning considering acceleration limits. *Robot. Autonom. Syst.* **45**, 199−210 (2003).

[42] Jolly, K.G., Kumar, R.S., Vijayakumar, R.: A Bezier curve based path planning in a multi-agent robot soccer system without violating the acceleration limits. *Robot. Autonom. Syst.* **57**, 23−33 (2009).

[43] Misel, B., Ivan, P.: Time-optimal trajectory planning along predefined path for mobile robots with velocity and acceleration constraints. In: *Proc. IEEE Int. Conf. Advanced Intelligent Mechatronics* (ICAIM), Budapest, Hungary, pp. 942−947 (2011).

[44] Faris, W.F., Ata, A.A., Saadeh, M.Y.: Energy minimization approach for a two-link flexible manipulator. *J. Vib. Control* **15**, 497−526 (2009).

[45] Zanotto, V., Gasparetto, A., Lanzutti, A., Boscariol, P., Viloni, R.: Experimental validation of minimum time-jerk algorithms for industrial robots. *J. Intell. Robot. Syst.* **64**, 197−219 (2011).

[46] Chia, N.K., Chia, J.W.: A PSO-tuning method for design of fuzzy PID controllers. *J. Vib. Control* **14**, 375−395 (2008).

[47] Chen, L.X., Cai, G.P., Pan, L.: Experimental study of delayed feedback control for a flexible plate. *J. Sound. Vib.* **322**, 629−651 (2009).

[48] Abe, A.: Trajectory planning for flexible cartesian robot manipulator by using artificial neural network: numerical simulation and experimental verification.

Robotica **29**, 797−804 (2011).

[49] Vaughan, J., Yano, A., Singhose, W.: Comparison of robust input shapers. *J. Sound. Vib.* **315**, 797−815 (2008).

[50] Park, U.H., Lee, J.W., Lim, B.D., Sung, Y.G.: Design and sensitivity analysis of an input shaping filter in the Z-plane. *J. Sound. Vib.* **243**, 157−171 (2001).

[51] Wang, Y., Wang, S., Tan, R., Jiang, Y.: Adaptive control method for path tracking of wheeled mobile robot considering parameter changes. *Int. J. Adv. Mechatronic Syst.* **4**, 41−49 (2012).

[52] Mohammadpour, E., Naraghi, M.: Robust adaptive stabilization of skid steer wheeled mobile robots considering slipping effects. *Adv. Robot.* **25**, 205−227 (2011).

[53] Hendzel, Z.: An adaptive critic neural network for motion control of a wheeled mobile robot. *Nonlinear Dyn.* **50**, 849−855 (2007).

[54] Ma, J.K., Tso, S.K.: Robust discontinuous exponential regulation of dynamic nonholonomic wheeled mobile robots with parameter uncertainties. *Int. J. Robust. Nonlinear Control* **18**, 73−99 (2008).

[55] Alvergue, L., Gu, G., Acharya, S.: A generalized sector-bound approach to feedback stabilization of nonlinear control systems. *Int. J. Robust. Nonlinear Control* **22**, 256−269 (2012).

[56] Rehman, F.U., Michalska, H.: Discontinuous feedback stabilization of wheeled mobile robots. In: *Proc. IEEE Int. Conf. Control Applications* (ICCA), Hartford, USA, pp. 167−172 (1997).

[57] Zhai, G., Takeda, J., Imae, J., Kobayashi, T.: An approach to achieving consensus in nonholonomic systems. In: *Proc. IEEE Int. Symposium on Intelligent Control* (ISIC), Yokohama, Japan, pp. 1476−1481 (2010).

[58] Yang, Z., Red, E.: On-line cartesian trajectory control of mechanism along complex curves. *Robotica* **15**, 263−274 (1997).

[59] Red, E.: A dynamic optimal trajectory generator for cartesian path following. *Robotica* **18**, 451−458 (2000).

[60] Crowther, A.R., Singh, R.: Analytical investigation of stick-slip motions in coupled brake-driveline systems. *Nonlinear Dyn.* **50**, 463−481 (2007).

[61] Wang, D., Low, C.B.: Modeling and analysis of skidding and slipping in wheeled mobile robots: control design perspective. *IEEE Trans. Robot.* **24**, 676−687 (2008).

[62] Yamamoto, Y., Yun, X.: Modeling and compensation of the dynamic interaction of a mobile manipulator. In: *Proc. IEEE Int. Conf. Robotics and Automation* (ICRA), California, USA, pp. 2187−2192 (1994).

[63] Zhong, G., Kobayashi, Y., Emaru, T., Hoshino, Y.: Trajectory tracking of wheeled mobile robot with a manipulator considering dynamic interaction and modeling uncertainty. In: *Proc. Int. Conf. Intelligent Robotics and Applications* (ICIRA), Montreal, Canada, pp. 366−375 (2012).

[64] Fateh, M.M., Khorashadizadeh, S.: Robust control of electrically draiven robots by adaptive fuzzy estimation of uncertainty. *Nonlinear Dyn.* **69**, 1465−1477 (2012).

[65] Peng, C.C., Chen, C.L.: Dynamic controller design for a class of nonlinear uncertain systems subjected to time-varying disturbance. *Nonlinear Dyn.* **57**, 411−423 (2009).

[66] Chen, W., Mills, J.K., Chu, J., Sun, D.: A fuzzy compensator for uncertainty of industrial robots. In: *Proc. IEEE Int. Conf. Robotics and Automation* (ICRA), Seoul, Korea, pp. 2968−2973 (2001).

[67] Damir, O., Leon, Z., Bojan, N.: Compensation of velocity and/or acceleration joint saturation applied to redundant manipulator. *Robotics Autonom. Syst.* **55**, 337−344 (2007).

[68] Ranjbarsahrael, B., Roopael, M., Khosravl, S.: Adaptive fuzzy formation control for a swarm of nonholonomic differentially driven vehicles. *Nonlinear Dyn.* **67**, 2747−2757 (2012).

[69] Low, C.B., Wang, D.W.: GPS-based path following control for a car-like wheeled mobile robot with skidding and slipping. *IEEE Trans. Control Syst. Technol.* **16**, 340−347 (2008).

[70] Farid, Y., Bigdeli, N.: Robust adaptive intelligent sliding model control for a class of uncertain chaotic systems with unknown time delay. *Nonlinear Dyn.* **67**, 2225−2240 (2012).

[71] Labiod, S., Boucherit, M.S., Guerra, T.M.: Adaptive fuzzy control of a class of MIMO nonlinear systems. *Fuzzy Sets. Syst.* **151**, 59−77 (2005).

[72] Gorban, A.N., Wunsch, D.C.: The general approximation theorem. In: *Proc. IEEE World Congress on Computational Intelligence* (WCCI), Anchorage, USA, pp. 1271−1274 (1998).

[73] Zhong, G., Kobayashi, Y., Emaru, T., Hoshino, Y.: Approaches based on particle swarm optimization for problems of vibration reduction of suspended mobile robot with a manipulator. *J. Vib. Control* (2012). doi: 10.1177/1077546312458534.

[74] Menini, L., Tornambe, A.: A Lie symmetry approach for the solution of the inverse kinematics problem. *Nonlinear Dyn.* **69**, 1965−1977 (2012).

[75] Marcos, M.G., Machado, J.A.T., Perdicoulis, T.P.A.: An evolutionary approach for the motion planning of redundant and hyper-redundant manipulators. *Nonlinear Dyn.* **60**, 115−129 (2010).

[76] Zhong, G., Kobayashi, Y., Emaru, T., Hoshino, Y.: Dynamics modeling and trajectory planning of wheeled mobile robot having a manipulator. In: *Proc. Int. Symposium on Robotics* (ISR), Taipei, Taiwan, pp. 206−211 (2012).

[77] Li, Y., Liu, Y.G.: Real-time tip-over prevention and path following control for redundant nonholonomic mobile modular manipulators via fuzzy and neural-fuzzy approaches. *J. Dyn. Syst., Meas. Control* **128**, 753−764 **(**2006).

[78] Henry, P., Krainin, M., Herbst, E., Ren, X., Fox, D.: RGB-D mapping: using Kinect-style depth cameras for dense 3D modeling of indoor environments. I*nt. J. Robot. Res.* **31**, 647−663 (2012).

[79] Krainin, M., Henry, P., Ren, X., Fox, D.: Manipulator and object tracking for in-hand 3D object modeling. *Int. J. Robot. Res.* **30**, 1311−1327 (2011).

[80] Graf, C., Kieneke, R., Maas, J.: Online force estimation for an active suspension control. In: *Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics* (ICAIM), Kaohsiung, Taiwan, pp. 544–549 (2012).

[81] Lai, C.Y., Liao, W.H.: Vibration control of a suspension system via a magnetorheological fluid damper. *J. Vib. Control* **8**, 527−547 (2002).

[82] Nieto, A.J., Morales, A.L., Trapero, J.R., Chicharro, J.M., Pintado, P.: An

adaptive pneumatic suspension based on the estimation of the excitation frequency. *J. Sound. Vib.* **330**, 1891−1903 (2011).

[83] Maciejewski, I.: Control system design of active seat suspensions. *J. Sound. Vib.* **331**, 1291−1309 (2012).

[84] Coulaud, J.B., Campion, G., Bastin, G., Wan, M.D.: Stability analysis of a vision-based control design for an autonomous mobile robot. *IEEE Trans. Robot.* **22**, 1062−1069 (2006).

[85] Shen,Y., Wang, L., Yang, S., Gao, G.: Nonlinear dynamical analysis and parameters optimization of four semi-active on-off dynamic vibration absorbers. *J. Vib. Control* **19**, 143−160 (2012).

[86] Cetin, S., Zergeroglu, E., Sivrioglu, S., Yuksek, I.: A new semiactive nonlinear adaptive controller for structures using MR damper: design and experimental validation. *Nonlinear Dyn.* **66**, 731−743 (2011).

[87] Fateh, M.M., Khorashadizadeh, S.: Optimal robust voltage control of electrically driven robot manipulators. *Nonlinear Dyn.* **70**, 1445−1458 (2012).

[88] Agheli M., Nestinger, S.S.: Study of the foot force stability margin for multi-legged/wheeled robots under dynamic situations. In: *Proc. IEEE/ASME Int. Conf. Mechatronics and Embedded Systems and Applications* (ICMESA), Suzhou, China, pp. 99−104 (2012).

[89] Farritor, S., Hacot, H., Dubowsky, S.: Physics-based planning for planetary exploration. In: *Proc. IEEE Int. Conf. Robotics and Automation* (ICRA), Leuven, Belgium, pp. 278−283 (1998).

[90] Huang, Q., Tanie, K., Sugano, S.: Stability compensation of a mobile manipulator by manipulator motion: feasibility and planning. *Adv. Robot.* **13**, 25−40 (1999).

[91] Zhong, G., Kobayashi, Y., Emaru, T., Hoshino, Y.: Optimal control of the dynamic stability for robotic vehicles in rough terrain. *Nonlinear Dyn.* **73**, 981−992 (2013).

[92] Dubowsky, S., Vance, E.E.: Planning mobile manipulator motions considering vehicle dynamic stability constraints. In: *Proc. IEEE Int. Conf. Robotics and Automation* (ICRA), Scottsdale, USA, pp. 1271−1276 (1989).

[93] Pinto, C.M.A.: Stability of quadruped robot's trajectories subjected to discrete

perturbations. *Nonlinear Dyn.* **70**, 2089−2094 (2012).

[94] Siegwart, R., Lamon, P., Estier, T., Lauria, M., Piguet, R.: Innovative design for wheeled locomotion in rough terrain. *Robot. Autonom. Syst.* **40**, 151−162 (2002).

[95] Zhong, G., Kobayashi, Y., Hoshino, Y., Emaru, T.: System modeling and tracking control of mobile manipulator subjected to dynamic interaction and uncertainty. *Nonlinear Dyn.* **73**, 167−182 (2013).

[96] Rakheja S., Sankar, S.: Vibration and shock isolation performance of a semi-active on-off damper. *J. Vib. Acoust. Stress Reliab. Des.* **107**, 398−403 (1985).

[97] Jiang, H., Kwong, C.K., Chen, Z., Ysim, Y.C.: Chaos particle swarm optimization and T-S fuzzy modeling approaches to constrained predictive control. *Expert Syst. Appl.* **39**, 194−201 (2012).

[98] Smith, M.C., Wang, F.C.: Controller parameterization for disturbance response decoupling: application to vehicle active suspension control. *IEEE Trans. Control Syst. Technol.* **10**, 393−407 (2002).

[99] Chen, Y., Wang, Z.L., Qiu, J., Huang, H.Z.: Hybrid fuzzy skyhook surface control using multi-objective microgenetic algorithm for semi-active vehicle suspension system ride comfort stability analysis. *J. Dyn. Syst. Meas. Control* **134**, 1−14 (2012).

[100] Thueer, T., Krebs, A., Siegwart, R., Lamon, P.: Performance comparison of rough-terrain robots−simulation and hardware. *J. Field Robot.* **24**, 251−271 (2007).

[101] Ryu, D., Hwang, C.S., Kang, S., Kim, M., Song, J.B.: Wearable haptic-based multi-modal teleoperation of field mobile manipulator for explosive ordnance disposal. In: *Proc. IEEE Int. Workshop on Safety, Security and Rescue Robotics* (IESSRR), Kobe, Japan, pp. 75−80 (2005).

[102] Zhong, G., Jiang, L.: Realization of EOD robot embedded control system based on PC/104. *Modern Manuf. Eng.* **32**, 119−122 (2010).

[103] Shiroma, N., Miyauchi, R., Matsuno, F.: Mobile robot teleoperation through virtual robot. In: *Proc. IEEE Int. Symposium on Robot and Human Interactive Communication* (ISRHIC), Munich, Germany, pp. 477−482 (2008).

[104] Sawaragi, T., Shiose, T., Akashi, G.: Foundations for designing an ecological interface for mobile robot teleoperation. *Robot. Autonom. Syst.* **31**, 193−207 (2000).

[105] Hainsworth, D.W.: Teleoperation user interfaces for mining robotics. *Autonom. Robot.* **11**, 19−28 (2001).

[106] Nielsen, C.W., Goodrich, M.A., Ricks, R.W.: Ecological interfaces for improving mobile robot teleoperation. *IEEE Trans. Robot.* **23**, 927−941 (2007).

[107] Cui, F., Zhang, M., Cui, G., Wu, X.: A robot teleoperation system based on virtual reality and WLAN. In: *Proc. World Congress on Intelligent Control and Automation* (WCICA), Dalian, China, pp. 9193−9197 (2006).

[108] Hill, S.L.: Scalable multi-view stereo camera array for real world real-time image capture and three-dimensional displays. *Master Thesis*, Massachusetts Institute of Technology, Cambridge, USA (2004).

[109] Zhong, G., Kobayashi, Y., Hoshino, Y., Emaru, T.: Intuitive teleoperation of nonholonomic mobile robot with a manipulator based on virtual reality and WiFi. In: *Proc. IET Int. Conf. Information Science and Control Engineering* (ICISCE), Shenzhen, China, pp. 802−806 (2012).

[110] Yip, M.C., Tavakoli, M., Howe, R.D.: Performance analysis of a haptic telemanipulator task under time delay. *Adv. Robot.* **25**, 651−673 (2010).

[111] Online: http://www.denso.com/.

[112] Pourboghhrat, F., Karlsson, M.P.: Adaptive control of dynamic mobile robots with nonholonomic constraints. *Comput. Elect. Eng.* **28**, 241−253 (2002).

Acknowledgment

First and foremost I want to thank my supervisor, Professor Yukinori Kobayashi, for providing me with the opportunity to complete my Ph.D. degree at Hokkaido University. It has been an honor to be his Ph.D. student. He has taught me, both consciously and unconsciously, how good research work is done. I appreciate all his contributions of time, ideas, and funding to make my Ph.D. experience productive and stimulating. The joy and enthusiasm he has for his research was contagious and motivational for me, even during tough times in the Ph.D. pursuit. I am also thankful for the excellent example he has provided as a successful researcher and professor.

Then I would like to acknowledge Associate Professor Takanori Emaru for his valuable attention and care. My special thanks also go to Assistant Professor Yohei Hoshino, some method in the thesis were inspired by discussing with him. Additionally, I sincerely appreciate Professor Itsuro Kajiwara, Professor Toshiro Hayashikawa, and Professor Yoshihiro Narita for their time, insightful questions, and helpful comments during the annual academic progress evaluation and review for my dissertation.

The members of the Robotics and Dynamics (R&D) group have contributed immensely to my personal and professional time at Hokkaido University. The group has been a source of friendships as well as good advice and collaboration. I would like to thank other past and present group members that I have had the pleasure to work with or alongside of are grad students Haibin Yin, Koichi Imaoka, Ryosuke Yamamoto, Yizhi Gai, Ravankar Ankit, and Jixin Lv; and the numerous rotation students who have come through the lab, for their skill in spreading happiness on those scientifically dark days. Also I am especially grateful for the English Engineering Education (e3) program members who stuck it out in grad school with me, and Dr. Werawan Manakul who was a coordinator of the e3 program.

I gratefully acknowledge the funding sources that made my Ph.D. work possible. I was funded by the China Scholarship Council for three years. My work was also supported by Hokkaido University.

My time at Hokkaido was made enjoyable in large part due to the many friends and groups that became a part of my life. I am grateful for time spent with classmates and friends, for my backpacking buddies and our memorable trips into the mountains, for R&D member's hospitality as I finished up my degree, and for many other people and memories. My time at Hokkaido University was also enriched by the Chinese Student Association, International Student Center, and many Japanese civil societies.

Lastly, I would like to thank my family for all their love and encouragement, for my parents who raised me with a love of science and supported me in all my pursuits. And most of all for my loving, supportive, encouraging, and patient sister whose faithful support during the final stages of this Ph.D. is so appreciated.