COOPERATIVE COLLISION AVOIDANCE CONTROL AND COORDINATION FOR MULTIAGENT LAGRANGIAN SYSTEMS WITH DISTURBANCES

BY

SHANKAR ANAND DEKA

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Mechanical Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2016

Urbana, Illinois

Adviser:

Associate Professor Dušan M. Stipanović

ABSTRACT

Multi-agent systems like a network of autonomous robots, have tremendous potential in many military and civilian applications. But, even viewed as a pure academic problem, designing controllers for such complex systems is a matter of much interest. Controller design for multi-agent system might focus on achieving several objectives, such as formation control, coverage control, consensus, target capture, pursuit evasion etc., while all at the same time aiming to be optimal in some sense, or following certain constraints imposed by the environment or communication limitations. Whatever is the objective, we always want to have a safety guarantee for the agents; the agents should avoid collisions with themselves and any static obstacles, while performing an objective. This thesis studies one such controller, which guarantees collision avoidance among the agents, in presence of bounded disturbances, while the agents carry out a coordination objective. The agents are assumed to follow a Lagrangian dynamics. The collision avoidance controller takes up the second part of the thesis. In the first part of this thesis, a particular Lagrangian system, the Raven II surgical robot, is studied in with the aim of highlighting the process of modelling and identifying such system. This is done for two reasons. One because Lagrangian dynamics is commonly used to model the agents in a multi-agent system. And second reason that motivates the modelling Raven II in part I, is to aid in future research direction pertaining to the control of Raven II.

ACKNOWLEDGMENT

There are many people whom I would like to extend my sincere gratitude and thank, for their help and support. Foremost, I thank my advisor, Prof. Dušan M. Stipanović, for his constant guidance and for all of the time which he invested in me. Without such an open and willing support, this work would not have been possible. Also, I would like to thank Prof. Kesavadas, for giving me the opportunity to work on the Raven II robot, which forms almost half of this thesis. Finally, I would like to thank my colleague, Xiao Li, who helped through several issues that cropped up during this project.

Table of Contents

List of figures

Chapter 1
Introduction

Multi-agents systems have gained a lot of interest from the control systems community over the last two decades, primarily due to its wide ranging applications. The problems that have been well studied in context of multi-agent systems are formation and flocking, consensus, synchronization, to name a few. Augmenting these problems with certain performance optimality requirements, or constraints, such as restricted communication between agents etc. makes the problem of control of a multi-agent system even more interesting to study, and perhaps more realistic in a practical sense. Since most of the multi-agent systems have a real-time application and require the control laws to be computed in a short time, the study of such systems are done under the framework of distributed, decentralized control. The usefulness of a complex multi-agent systems however comes at the cost of increased complexity in control design, which really depends on the specific objectives and performance goals of the system, such as modularity, flexibility, scalability, reliability, ease of implementation, adaptability to new environments, robustness with respect to sensing and communication unreliability, autonomy, safety etc. Of the foremost importance in any multi-agent system control objective is safety of the agents, and the need for controllers that in some sense provide guaranteed collision avoidance. This thesis studies the problem of collision avoidance and a controller that guarantees it, while allow the agents to carry out other control objective like coordination.

1.1 System dynamics

Each of the individual agents in the multi-agent system are considered to have a non-linear, Lagrangian dynamics. Lagrangian dynamics is an ideal choice for modelling for such an

application because it is very general, in sense that a lot of physical systems actually have such a dynamics, for example, electrical circuits, industrial robotic arms, mobile robots, aerospace systems etc. Moreover, the Lagrangian dynamic equations themselves have a nice structure (discussed in a section below) which can be cleverly exploited while designing controllers for the system. The communication graph used to model the flow or exchange of state information between agents varies from one application to another. In this paper, we considered a connected, undirected communication graph with fixed topology. Part one of this thesis is dedicated to the development of lagrangian dynamics for a robotic system, while part two studies a particular problem in control of systems with lagrangian dynamics.

1.2 Collision avoidance

Collision avoidance in case of multi-agent systems has been studied since the late 1980s. Early works on path planning and collision avoidance in robotic systems focused on the use of artificial potential fields [1] or variants of it. However, Leitmann et al. [2] [3] led pioneering works on a more control theoretic approach to collision avoidance using Lyapunov analysis in the early 1980s, though there seems to be a dichotomy between the computer science community and the controls community in regards to origins of collision avoidance control. A chronological survey with more details on avoidance control is provided in [4] The controller described in this thesis guarantees collision avoidance while allowing the agents to coordinate their positions, all in presence of bounded disturbance. On a more general level, we can think of collision avoidance between agents as a simply as a constraint: avoiding a subset of the configuration space of the system of agents, where any two agents get closer that a certain distance. With this point of view, we can extend the same ideas and approach in order to impose other general constraints, that aren't necessarily avoiding collision.

1.3 Organization of the thesis

This thesis is organized into a total of 6 Chapter.

Chapter 2 introduces a surgical robot called Raven II. This robot is used as an example of a system with Lagrangian dynamics. It mainly highlights the mechanical construction of the surgical robot, including the actuation mechanisms.

Chapter 3 is dedicated to the dynamic modelling of the Raven II. The Euler-Lagrange method for deriving the dynamic equations of a robotic kinematic chain is reviewed, and applied to the Raven II robot. Apart from modelling the links, a model for the motor and the cable transmission is also discussed. The model is computed using symbolic toolbox in Matlab®, but since terms involved are huge, they have not been included in the thesis.

The next chapter, Chapter 4, takes the reader through parameter estimation for robotic arms. It discusses various techniques one can use to estimate the dynamic parameters of a robotic arm, and the same have been used on Raven II, in order to classify the unknown dynamic parameters into appropriate categories based on their identifiability.

Once we have a systems of agents and know/derive their Lagrangian dynamics, the next thing is to design controllers for the multi-agent systems. This is the link between Chapter 4 and Chapter 5, which deals with control for coordination and collision avoidance in multi-agent Lagrangian systems.

The simulations for the controller derived in Chapter 5 are performed on a system of 4 agents and the results are discussed in Chapter 6.

The final chapter, Chapter 7, concludes the thesis, summarizes the work, and suggests some future work.

Chapter 2
Raven II surgical robot

2.1 Brief overview of the Raven II

The Raven II is a surgical robot with 7 degrees of freedom, compact electronics and two wing-like arms which end in tiny gripper claws designed to perform surgery on simulated patients. The robot's software is compatible with Robot Operating System (ROS), an open source robotics coding platform. The main goal of Raven II is to provide common hardware and software to support research innovations in surgical robotics, so that research innovations can be shared and built upon.

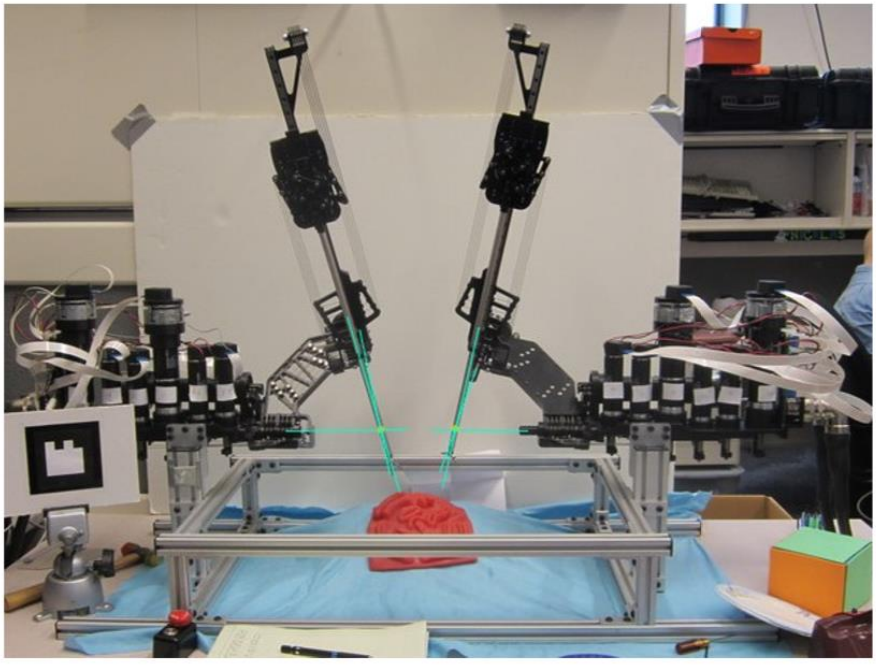


Figure 1: Left and Right arms of Raven-II in the reference pose. Axis lines are superimposed in blue showing their intersection at the remote motion center (Source: [5])

The seven-degree-of-freedom (7-DOF) cable-actuated surgical manipulator can be divided into three main parts:

1. Static base, which holds all of the motors.

2. Spherical mechanism that positions the tool and

3. Detachable tool interface.

Each of the two arms of the Raven 2 have the following 7 motion axes:

1. Shoulder joint (rotational)

2. Elbow joint (rotational)

3. Tool insertion/retraction (translational)

4. Tool rotation (rotational)

5. Tool grasping (rotational)

6. Tool wrist-1 actuation (rotational)

7. Tool wrist-2 actuation (rotational).

The first four joint axes intersect at the surgical port location, creating a spherical mechanism that allows for tool manipulation similar to manual laparoscopy.

2.2 Raven II Hardware

The RAVEN utilizes DC brushless motors located on the stationary base, which actuate all motion axes. Maxon EC-40 motors with 12:1 planetary gearboxes are used for the first three axes, which see the highest forces. The first two axes, those under the greatest gravity load, have power-off brakes to prevent tool motion in the event of a power failure. The fourth axis uses an EC-40 without a gearbox, and Maxon EC32 motors are used for the remaining axes. Maxon DES70/10 series amplifiers drive these brushless motors. The motors are mounted onto

the base via quick-change plates that allow motors to be replaced without the need to disassemble the cable system. [6]

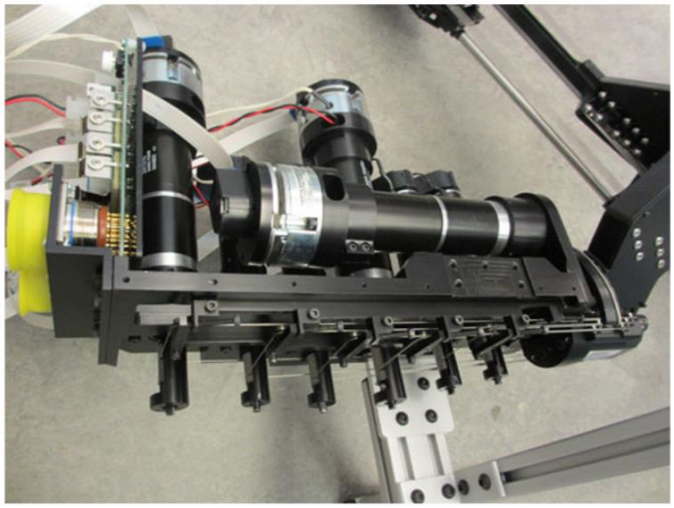


Figure 2: Raven-II motor chassis showing cable tensioners. Arranging six of seven motors in a linear array reduced the width of the base (Source: [7])

The cable transmission system comprises a capstan on each motor, a pretension adjustment pulley, various pulleys to redirect the cables through the links, and a termination point to each motion axis. The shoulder axis is terminated on a single partial pulley. The elbow axis has a dual-capstan reduction stage terminating on a partial pulley. The tool insertion/retraction axis has direct terminations of the cables on the tool holder. The tool rotation, grasping, and wrist cables are terminated on capstans on the tool interface.

Chapter 3
The Raven II dynamics

The dynamic model of a robotic manipulator describes how the robot moves in response to the applied joint torques (in case of rotary joints) and joint forces (in the case of prismatic joints). Mathematically, the dynamic model for a kinematic chain can be formulated as a set of nonlinear, second order, ordinary differential equation which depend on the inertial and kinematic properties of the links. There are several approaches to derive the dynamic model of a kinematic chain, such as Euler-Lagrange approach, Newton-Euler approach or the Lagrange Recursive approach (of these, the first two are most widely used). Each of these approaches results in equivalent set of equations. The choice of any particular method depends on the application or purpose for studying the dynamics. The derivation using Euler-Lagrange method is intuitive and simpler as it considers the links as a whole depends only on the total kinetic energy and the potential energy of the mechanical system. However, it isn't very computationally efficient and therefore less suitable for real-time control purposes which is not an issue in our case since we are building a model to be perfected and implemented off-line as an integral part of the simulator. Newton-Euler method, on the other hand handles each link separately. It relies on recursive techniques and is computationally more efficient [8]. But the resulting equations are not in a closed form, that is, explicit and thus it is more difficult to control them using standard control designs. Therefore, we will choose Lagrangian approach to derive robot dynamics, which allows the equations to be computed in closed form, allowing detailed analysis of the properties of the system. Although the dynamic equations can be derived in principle by summing up all the forces and torques acting on coupled rigid bodies (i.e the robotic links), the Lagrangian approach gives a deeper insight into the structural properties of the dynamic equations. Also, Lagrange's equations reduce the number of equations needed to

describe the motion of the system from m, the number of particles in the system, to n, the number of generalized coordinates (the number of degree of freedom). This technique has the advantage of requiring only the kinetic and potential energies of the system to be computed, and hence tends to be less prone to error than summing together the inertial, Coriolis, centrifugal, actuator, and other forces acting on the robot's links.

There are two basic ways that this problem can be solved. The first, referred to as joint space control, involves converting a given task into a desired path for the joints of the robot. A control law is then used to determine joint torques which cause the manipulator to follow the given trajectory. A different approach is to transform the dynamics and control problem into the task space, so that the control law is written in terms of the end- effector position and orientation. We refer to this approach as workspace control.

For each arm, instead of modeling in the end-effector's workspace, we will do the modeling in the joint space, because essentially we will control each joint's motion independently and then through forward kinematics chain we can reach to the end-effector's configuration.

3.1 Euler-Lagrange formulation

In order to derive the dynamics with the Euler-Lagrangian approach, we first need to calculate the kinetic and the potential energy of each kinematic link. The velocity, inertia matrices and the center of mass coordinates for each link will be calculated with respect to the D-H coordinate system formulated in [5]. The particular choice of coordinate system does not affect the standard way in which kinetic and potential energy of the system is obtained. The figures below show the coordinate frames use to derive the kinematic and the dynamic equations.

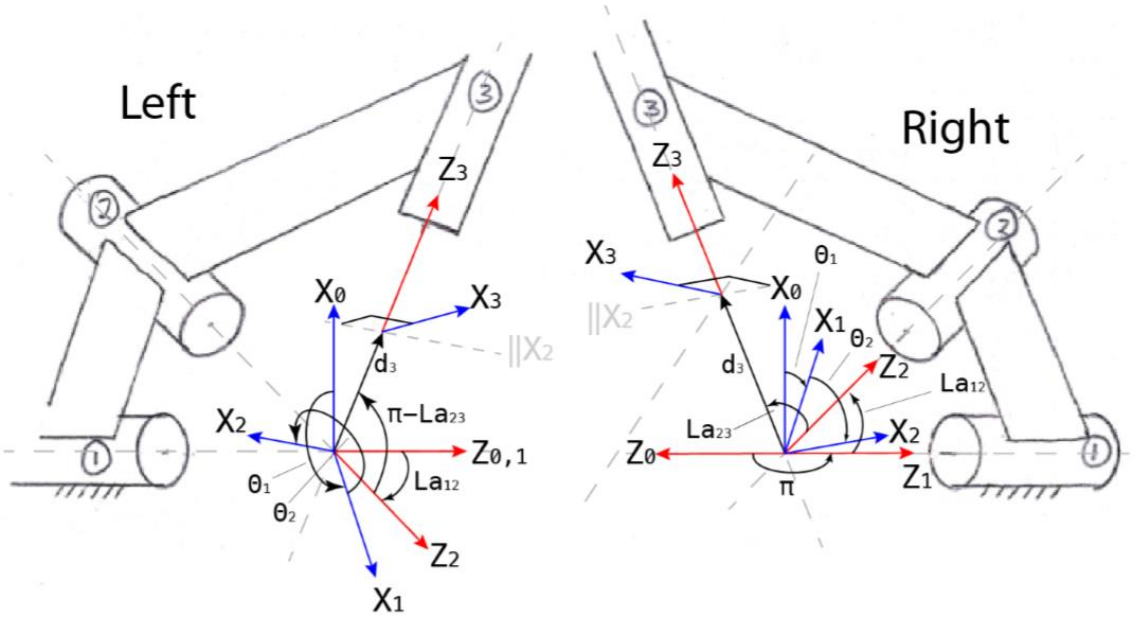


Figure 3: Coordinate frames for link 1, 2 and 3. The origins of frame0, frame1 and frame2 coincide at the center the spherical mechanism. Also, z-axes of frames1, 2, and 3 intersect at that point(Source: [5])

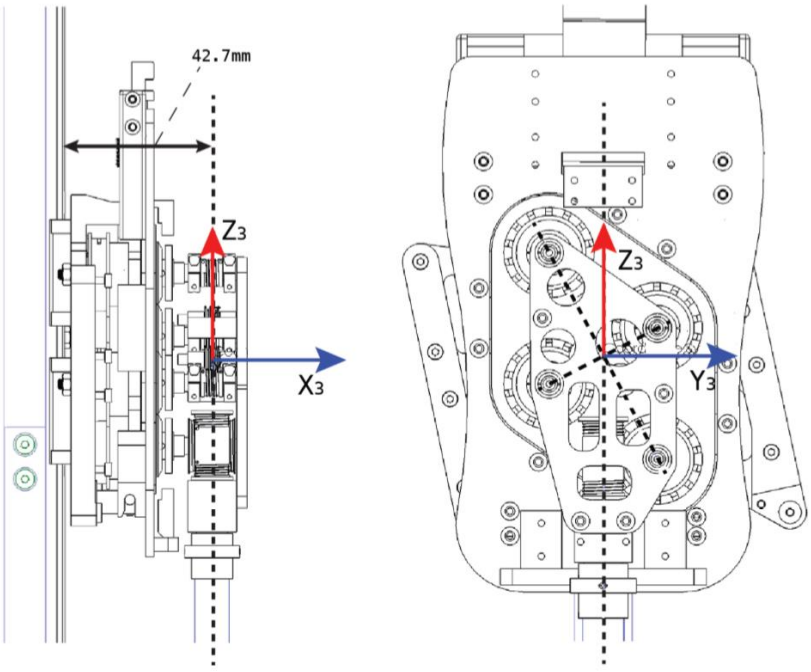


Figure 4: The choice of origin of frame 3 is arbitrary and in our case has been chosen as shown in above.(Source: [5])

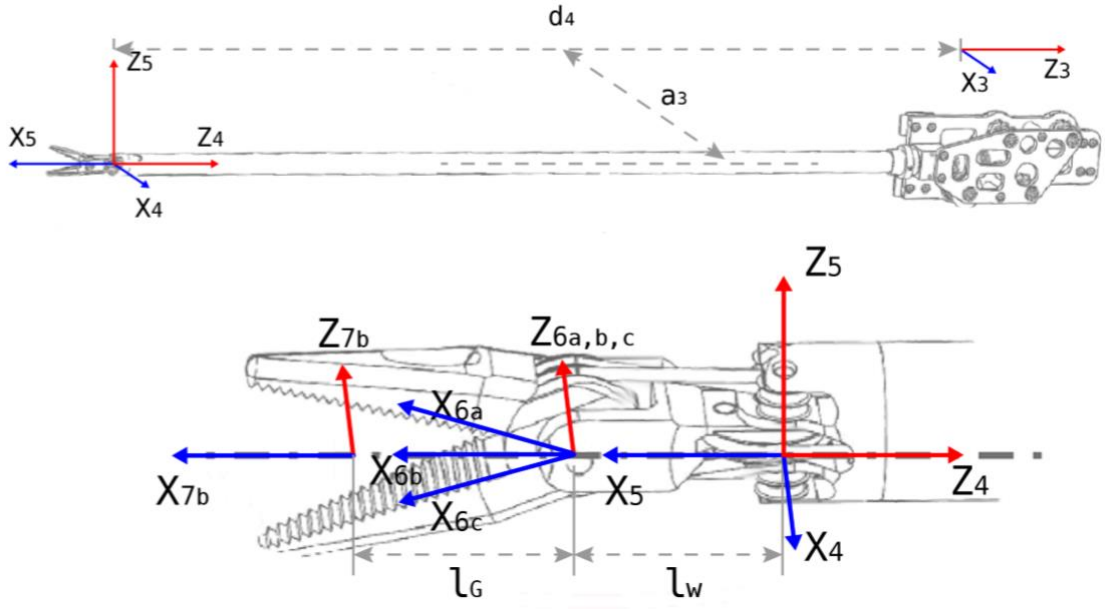


Figure 5: The coordinate frames for links 4, 5 and 6. In this thesis, the dynamics of the last three degrees of freedom have been ignored. (Source: [5])

For an *n*-link open chain, the kinetic energy K_i of link *i* is given by [9]:

$$K_i = \frac{1}{2} m_i \dot{d}_0^{iT} \dot{d}_0^i + \frac{1}{2} \omega_0^{iT} R_0^i I_i R_0^{iT} \omega_0^i - m_i \dot{d}_0^{iT} R_0^i S(r_i) R_0^{iT} \omega_0^i$$

where m_i is the mass of link *i*, $\dot{d}_0^i$ is the linear velocity vector of the origin of frame *i* with respect to frame 0, ω_0^i is the angular velocity vector of frame *i* with respect to frame 0, R_0^i is the rotational matrix of frame *i* with respect to frame 0, I_i is the inertia matrix of link *i* in frame *i*, r_i is the center of mass of link *i* in frame *i*, and $S(\cdot)$ is a skew symmetric matrix operator. The linear velocity $\dot{d}_0^i$ is computed as $\dot{d}_0^i = \frac{\partial d_0^i}{\partial q} \dot{q} = J_{vi} \dot{q}$ where *q* is the joint position vector and $\dot{q}$ is the

joint velocity vector. J_{vi} is defined as $J_{vi} = \begin{bmatrix} \frac{\partial d_0^i}{\partial q_1} & \frac{\partial d_0^i}{\partial q_2} & \cdots & \frac{\partial d_0^i}{\partial q_n} \end{bmatrix}$ where $\frac{\partial d_0^i}{\partial q_1} = 0$ for $j = i+1$, $i+2,\ldots, n$ and for $j = 1, 2, \ldots, i$,

$$\frac{\partial d_0^i}{\partial q_j} = \begin{cases} z_{j-1}, \text{ if } j \text{ is a prismatic joint} \\ z_{j-1} \times [o_n - o_{j-1}], \text{if } j \text{ is a revolute joint} \end{cases}$$

where z_{j-1} is the z axis of frame j-1. o_n is the origin of the frame n and o_{j-1} is the origin of the frame j-1.

The angular velocity vector of frame i with respect to frame 0 if given by

$$\omega_0^i = J_{\omega i}\dot{q}$$

where, $J_{\omega i}$ is defined as $J_{\omega i} = \begin{bmatrix} \rho_1 z_0 & \rho_2 z_1 & \cdots & \rho_n z_{n-1} \end{bmatrix}$. ρ_{1j} is zero if $j = i+1,\ldots,n$ or 1 if $j = 1,2,\ldots,i$ and the joint j is revolute.

The potential energy for link i is calculated as [9]

$$P_i = m_i g^T \left(R_0^i r_i + d_0^i \right),$$

where d_0^i is the position of the origin of frame i with respect to frame 0.

The Kinetic and the Potential energy can also be computed using an alternative way. Since each arm of Raven II robot is an open chain manipulator, we can use twist to represent its motions and model it in a more elegant way than general mechanical systems [10]. Let $g_i = e^{\hat{\xi}_1\theta_1} \cdots e^{\hat{\xi}_i\theta_i}$ denote the robot configuration for frame i relative to the base frame, in which $\left(\hat{\bullet}\right)$

operator denotes the matrix representation of the twist $\xi_i = (v_i, w_i)$. The body velocity of the center of mass for link *i* is given by

$$V_i^b = \begin{bmatrix} \xi_1^\dagger & \cdots & \xi_i^\dagger & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 & \cdots & \dot{\theta}_i & \dot{\theta}_{i+1} & \cdots & \dot{\theta}_n \end{bmatrix}^T = J_i^b \dot{\theta}$$

*w*here $J_i^b(\theta)$ is the body Jacobian corresponding to g_i. Furthermore,

$\xi_j^\dagger = Ad^{-1}\left(e^{\hat{\xi}_{j+1}\theta_{j+1}} \cdots e^{\hat{\xi}_i\theta_i} g_i(0)\right)\xi_j$, $j \le i$, and $Ad_g^{-1} = \begin{bmatrix} R^T & -R^T\hat{p} \\ 0 & R^T \end{bmatrix}$ with respect to the associated homogenous transformation $g = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix}$. Then the total kinetic energy is given by $T(\theta, \dot{\theta}) = \frac{1}{2}\sum_{i=1}^{4} \dot{\theta}^T J_i^{bT} \mathcal{M}_i J_i^b \dot{\theta}$, and the total potential energy is $V(\theta) = \sum_{i=1}^{4} m_i g h_i(\theta)$, where each $h_i(\theta)$ is known from .

Again, since the implementation is done offline the dynamic model of the Raven II is derived using the Euler-Lagrange approach. Specifically each arm of Raven II has seven degrees of freedom, however for the model to be built in the first place (at least) consider five degrees of freedom are considered. The final two degrees of freedom which have been left out of the derivation again at least in the first approximation, correspond to the tool pitch and jaw opening and closing. Having those two extra degree of freedom in the model would increase the computational cost significantly. Since the mass of the gripper jaws are negligible compared to the other links, they do not have any significant effect on the dynamics of the raven II, and therefore do not justify the increased cost of computation. Hence we assume the last two degrees of freedom are of the form $\ddot{\theta}_i = \tau_i$.

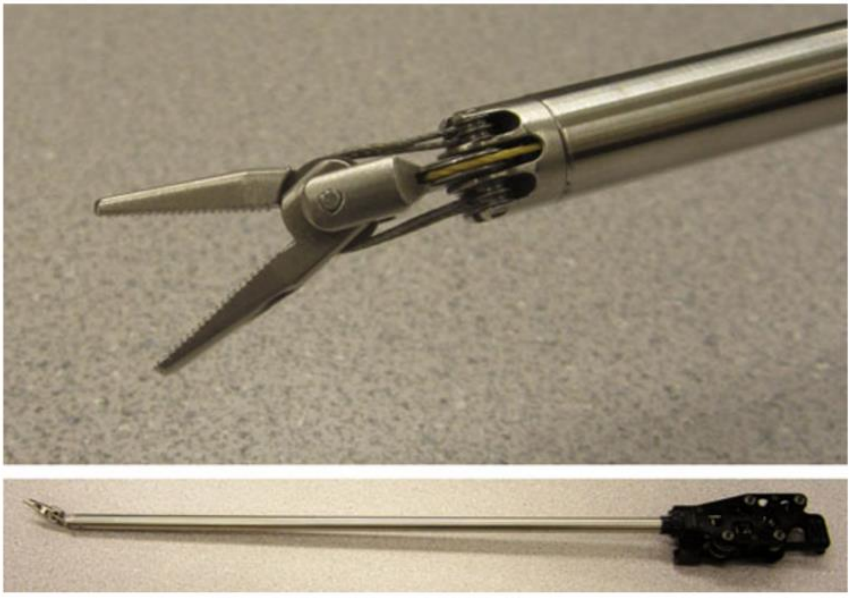


Figure 6: The gripper jaw at the end of the serial kinematic chain. Since, the end effector has negligible mass, its dynamics are simplified (Source: [7])

The kinetic and potential energies of the left arm of Raven II will be calculated using the same joint parameters and reference frame conventions as in [5] while deriving the kinematic model.

The Lagrangian is formulated as

$$L(q,\dot{q}) = K - P$$

where *K* and *P* are respectively, the total kinetic and potential energy of the five or seven degrees of freedom kinematic chain, *q* is the five dimensional joint coordinate vector and its derivative $\dot{q}$ is the joint velocity vector. Once the Lagrangian is obtained, the dynamic equation is given by

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\right) - \frac{\partial L}{\partial q} = \tau$$

where τ is the joint torque vector. Assuming there are no joint friction or external forces acting on the link, the equation takes the form

$$M(q)\ddot{q}+C(q,\dot{q})\dot{q}+g(q)=\tau$$

where *M* is the five dimensional square inertial matrix, *C* is the five dimensional square Coriolis matrix and *g* is the gravity vector of dimension five. The *M* and the *C* matrix have special structural properties that can be exploited while designing controllers for the system [11].

3.2 Modelling cable dynamics

The Raven II robot has a cable driven power transmission mechanism. The cables transfer the torque from the motor capstan to the joint. The advantages of having a cable-pulley mechanism to drive joints over having a direct drive (motor directly drives the joint) is that by having the motors at the base, rather than being attached to the links, the moments of inertia and the mass of the links can be kept small. This leads to smaller values of motor torques in general. Also, they allow for a more compact manipulator design. However, a cable driven robot is a little more difficult to model because one has to consider the cable dynamics as well. Since the cables are elastic and flexible, factors like backlash, compliance introduce a lot of modelling uncertainties, and therefore, necessitate some robustness analysis in the controller design process.

Furthermore cable properties change over time due to cable stretch and creep (a property of cables in which they slowly lengthen under long term load).

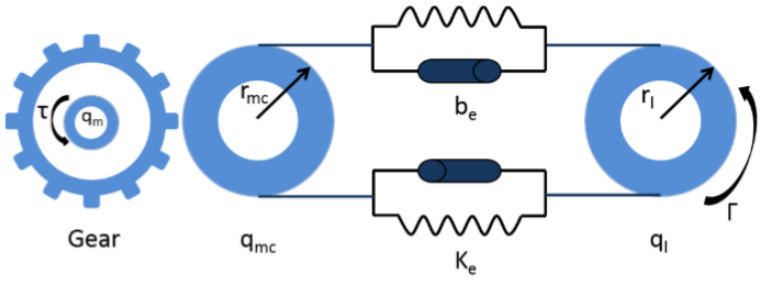


Figure 7: Schematic drawing of a generic cable driven system. Motor shaft is connected to a gearbox and capstan is fixed on the gearbox shaft. (source: [12])

Because the cables are not infinitely stiff, during torque transmission, the motor capstan rotation and the joint capstan rotation would in general be different, and related by a nonlinear dynamic equation. Cable dynamic properties such as stiffness and internal damping are known to vary as a function of tension [4], [5]. However, assuming that there is a very small strain in the cables, we can assume the stiffness and damping constants to be constant in the range of operation of the Raven II. The motor and the joint torques are related by the following non-linear cable coupling dynamics:

$\tau_{load} = (r_{mc} / N) * \gamma$ is the motor load torque, constants r_{mc} and N are the motor capstan radius and the gear-box ratio respectively

$\tau_j = r_l * \gamma$ is the applied joint torque

$\gamma = \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{bmatrix}$ is a 3x1 matrix that describes the cable coupling dynamics

where

$$\gamma_1 = k_e(e^{r_{mc}(\theta_{m1}/N) - r_{l1}\theta_{j1}} - e^{r_{l1}\theta_{j1} - r_{mc}(\theta_{m1}/N)}) + 2b_e(\mathrm{r}_{mc}(\dot{\theta}_{m1} / N) - \mathrm{r}_{l1}\dot{\theta}_{j1})$$

$$\gamma_2 = k_e(e^{r_{mc}(\theta_{m2}/N) - r_{l2}\theta_{j2}} - e^{r_{l2}\theta_{j2} - r_{mc}(\theta_{m2}/N)}) + 2b_e(\mathrm{r}_{mc}(\dot{\theta}_{m2} / N) - \mathrm{r}_{l2}\dot{\theta}_{j2})$$

$$\gamma_3 = k_e(e^{r_{mc}(\theta_{m3}/N) - r_{l3}\theta_{j3}} - e^{r_{l3}\theta_{j3} - r_{mc}(\theta_{m3}/N)}) + 2b_e(\mathrm{r}_{mc}(\dot{\theta}_{m3} / N) - \mathrm{r}_{l3}\dot{\theta}_{j3})$$

3.3 Motor Dynamics

Each of the first three joints of the Raven II (shoulder joint, elbow joint and the tool insertion) are controlled via cable mechanism by three motors that decoupled from one another. However, the last four joints, whose dynamics we shall ignore in this thesis, are in fact coupled, i.e rotation of one motor causes motion in more than one joint.

For the first three motors, the dynamics is given by a second order differential equation:

$\theta_m = [\theta_{m1}, \theta_{m2}, \theta_{m3}]'$ is the motor position vector

$I_m \ddot{\theta}_m + F_{mc} sign(\dot{\theta}_m) + F_{mv} \dot{\theta}_m + \tau_{load} = \tau_m$

where,

I_m, F_{mc}, F_{mv} are 3x3 positive definite diagonal matrices, representing the motor inertia, motor coloumb friction coefficient and motor viscous friction coefficients respectively.

τ_m is 3x1 vector, the control input to the motors

τ_{load} is 3x1 vector, the motor load torque

Due to the presence of the friction term (the coulomb friction in particular), there is a discontinuity in $\ddot{\theta}$ whenever $\dot{\theta}$ crosses zero.

3.4 A note on friction

Friction plays a dominant role in limiting the quality of robot performance. Non compensated friction produces static error, delay, and limit cycle behavior. There are many references on the friction torque in the joint and transmission systems. Various friction models have been proposed in the literature. In general, three kinds of frictions are noted: Coulomb friction, static friction, and viscous friction. The model based on Coulomb friction assumes a constant friction component that is independent of the magnitude of the velocity. The static friction is the torque necessary to initiate motion from rest. It is often greater than the Coulomb friction (Figure 9.2a). The viscous friction is generally represented as being proportional to the velocity, but experimental studies have pointed out the Stribeck phenomenon that arises from the use of fluid lubrication. It results in decreasing friction with increasing velocity at low velocity, then the friction becomes proportional to velocity (Figure 9.2b). A general friction model describing these components is given by:

$$\Gamma_i = F_{ci} sign(\dot{q}_i) + F_{vi} \dot{q}_i + (F_{si} - F_{ci}) sign(\dot{q}_i) e^{-|\dot{q}_i| B_i}$$

In this expression, Γ_i denotes the friction torque of joint i, F_{ci} and F_{vi} indicate the Coulomb and viscous friction parameters respectively. The static torque is equal to $F_{si}sign(\dot{q}_i)$

The most often employed model is composed of Coulomb friction together with viscous friction (Figure 9.2c). Therefore, the friction torque at joint i is written as:

$$\Gamma_i = F_{ci}sign(\dot{q}_i) + F_{vi}\dot{q}_i$$

This friction model can be approximated by a piecewise linear model as shown in Figure 9.2d.

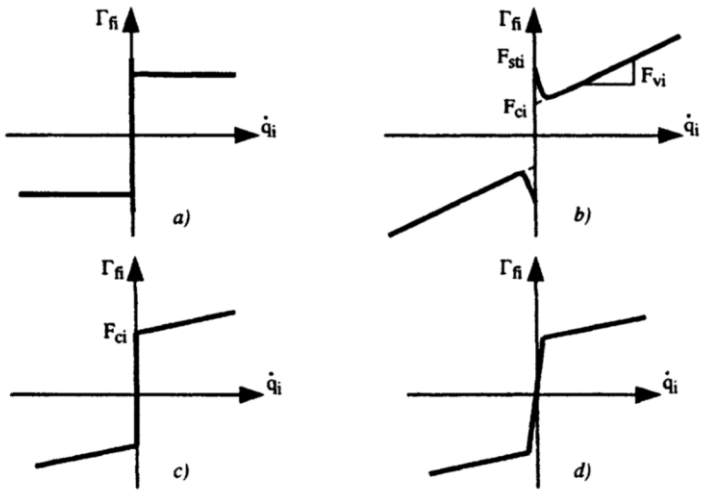


Figure 8: Various friction models (source: [13])

3.5 Remarks on discretization

Although this thesis develops only a continuous time model of the Raven II, an important future work to be addressed is the proper discretization of the dynamics once it is accurately developed in continuous-time in the Euler-Lagrangian form. This is particularly important since the model will be a link between the discrete-time simulator and the continuous-time robot, that is, Raven II. Thus it has to be accurate and the discretization procedure cannot include any inaccuracies which will cause discrepancies between the human operator and the robot. Of

course that the discretization process is never perfect for realistic causal systems yet it must be designed not to significantly affect (that is, affect negligibly) any of the Raven's dynamic model behavior within its operating range.

Discretization of a continuous time dynamic model is necessary and a very crucial step for simulation purposes. Many traditional numerical integrators such as Range-Kutta or Euler method may fail to preserve the structural properties of dynamical equations during discretization. Some of the structural properties of Lagrangian dynamical equations of Raven II (more generally, an open chain robotic arm) can be exploited while designing controller for various purposes, such as collision avoidance, formation control, or simply for developing PD controllers for trajectory tracking as ([10], [14]). Therefore discretization method must be carefully chosen so as to preserve such properties. The idea of using variational integrators to achieve time discretization of the Euler-Lagrange equations is discussed in [15]. The method relies on discretizing the lagrangian action functional of the system and then deriving the equations of motion from a variational principle rather than direct discretization of the equations of motion. The resulting numerical methods have several geometrically desirable properties, including multi-symplecticity, conservation of momentum maps via a discrete version of Noether's theorem, preservation of differential structure and gauge symmetries, lack of spurious modes, and excellent longtime energy conservation behavior. Of these, the longtime energy conservation behavior and preservation of differential structure are of particular interest in the context of Raven II simulations.

Chapter 4
Parameter estimation

4.1 Motivation for parameter estimation

One of the main issues in developing an accurate dynamic model of Raven II is to determine the values of the unknown physical parameters that appear in the dynamic model. The parameters we seek to identify are the moments on inertia of the links and the coordinates of their center of mass. These parameters can be determined using some standard packages, like Solidworks® using the CAD model of the links. However, they could lead to an inaccurate model. A more reliable approach in determining these parameters would be to use a dynamic model along with data from actual experiments on the real robot, and then fit the unknown parameters.

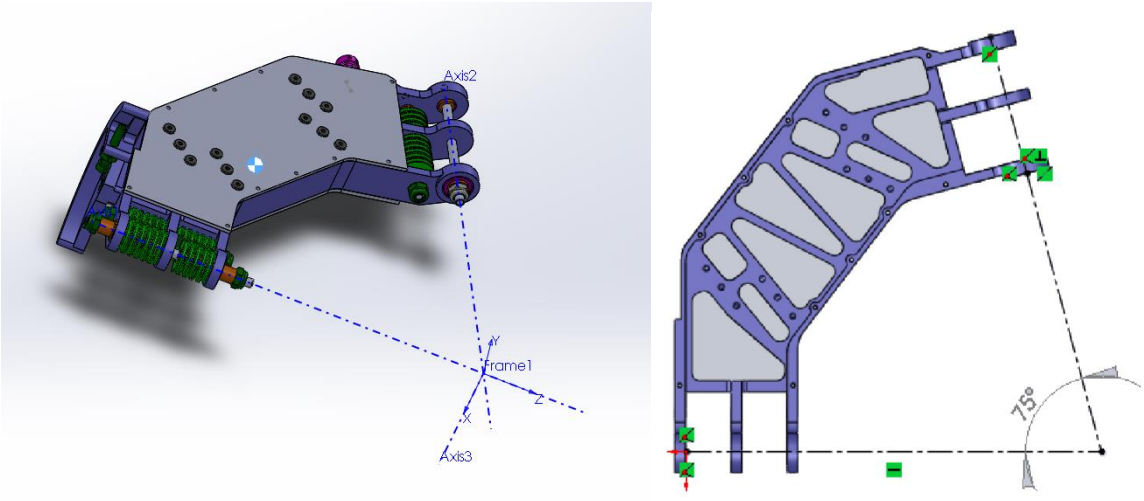


Figure 9: 3D CAD model of Link 1 of RAVEN II.

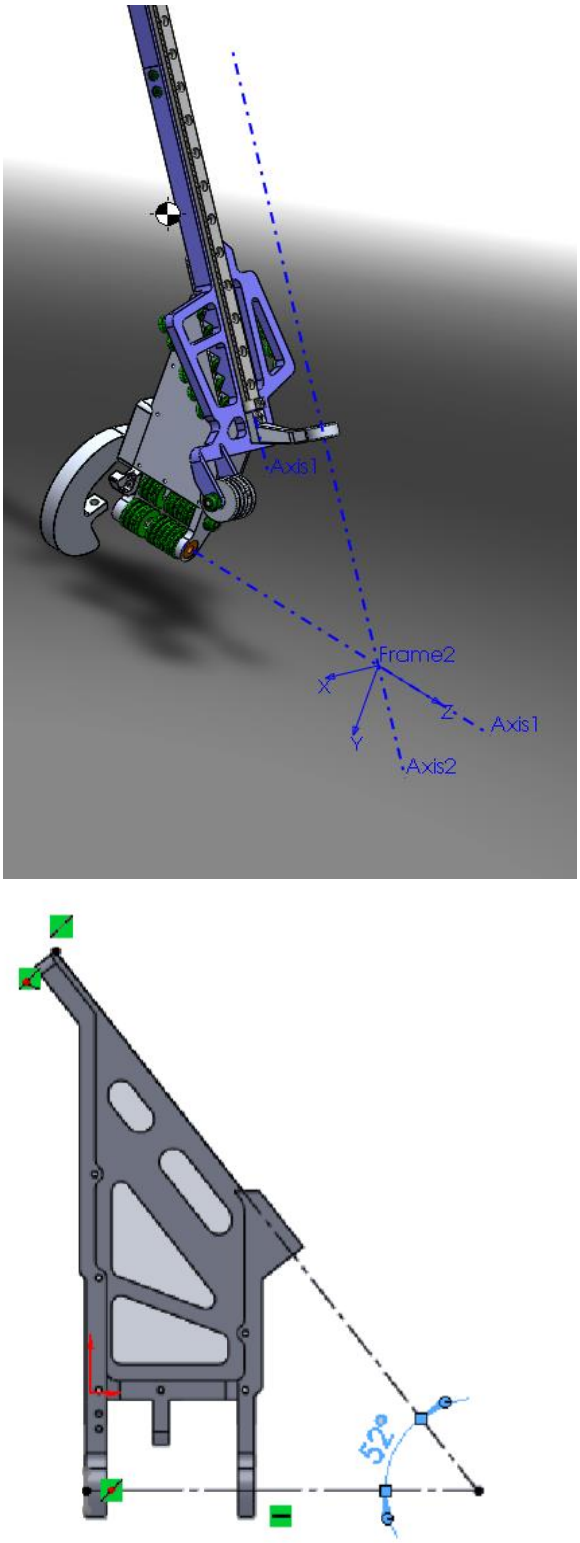


Figure 10: 3D CAD model of Link 2 of Raven II

Figure 11: 3D CAD model of joint 3 of Raven II

If we express all the uncertain parameters as a vector *X*, then it is possible to rearrange the right hand side of the equation above as a linear function of the vector *X* as follows:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \underbrace{A(q,\dot{q},\ddot{q}) + B(q,\dot{q},\ddot{q})X}_{\approx\tau(q,\dot{q},\ddot{q},X)}$$

where vector *A* is computed as $A(q,\dot{q},\ddot{q}) = \tau(q,\dot{q},\ddot{q},0)$ and matrix *B* is obtained by computing the Jacobian of τ with respect to *X*. Again, based on our earlier experience these computations can be performed efficiently using Matlab's symbolic toolbox.

The inertia tensor for each link has 6 unknowns and the center of mass coordinates add another 3 unknowns. Therefore for 5 links, there is a total of 45 unknown parameters. Besides the

identification of the link parameters mentioned earlier, the same idea can be extended to identify friction coefficients as well, if one choses to include frictional forces in the dynamic model. For a Columbic model of joint friction, this can be simply done by extending the *X* vector to also include the unknown friction related parameter, and augmenting the *B* matrix accordingly. One the dynamic model is rearranged in terms of *A* and *B* matrices, the parameters can be estimated from the experimental joint torque data using methods such as least square fit.

Having to fit large number of unknown parameter all at once can lead to very large computational time. To avoid this, one can cleverly make use of the recursive structure of the open kinematic chain. A particularly useful observation to make is that given the values of the vectors q, $\dot{q}$ and $\ddot{q}$, the value of the unknown inertial terms like moment of inertia or the center of mass of a link closer to the base have no effect on the parameter estimation equations for the links farther away from the base. Therefore we can make the computations faster and more efficient by starting with the last link (gripper) and progressively moving towards the first link.

While collecting torque and joint angle data from actual raven robot, the reference joint trajectories is chosen in a way to make the equations simpler yet relevant. Each joint is given a sinusoidal reference trajectory. In some very preliminary calculations only one joint is given a non-zero trajectory reference at a time. The remaining joint reference trajectories are set to zero. For example, while fitting the unknown parameters for link 3, the joint reference trajectories for joint 1, 2, 4, and 5 are set to zero.

4.2 Identifiability of Dynamic Parameters

All Identification models can be written in the form

$$Y(\tau, \dot{\mathrm{q}}) = \mathrm{W}(\mathrm{q}, \dot{\mathrm{q}}, \ddot{\mathrm{q}})\chi + \rho$$

Where χ is the vector of unknown dynamic parameters, $W_{r\,x\,c}$ is an observation matrix, or more commonly known as regressor with r>>c, and ρ is the residual error. With this overdetermined linear system, one can obtain an estimate $\hat{\chi}$, of the unknown parameters by solving

$$\hat{\chi} = \min_{\chi} \|\rho\|^2$$

If W is full rank, the explicit solution is obtained as

$$\hat{\chi} = (W^T W)^{-1} W^T Y$$

The dynamic parameters that we seek to identify for each link are the link mass, the six entries of inertia tensor matrix and the three center of mass coordinates; i.e each link has ten dynamic parameters that need to be estimated. A very obvious but important thing to note is that the dynamics of the any particular link i is only affected the dynamic parameters of link i and those of links that come after the link i in the open kinematic chain, assuming we starting numbering the links from the base (Therefore estimating the parameters of the links starting from the posterior end and progressing backwards towards the links closer to the base would mean each equation having fewer number of unknown parameters to estimate). Another point to note is, not all of the dynamic parameters of link i and those of the links i+1, i+2,…,N, necessarily appear in the dynamic equation for link i. In fact, certain dynamic parameters of the links may not affect the torque of any of the joints, as shown by the example below for the case of Raven II robot. Since these parameters do show up anywhere in the equations, cannot be estimated by the method described in this thesis and for the same reason, are not needed to be estimated far as modelling, simulation and control of the Raven II robot (or any dynamic system in general) is concerned. For example the I_{xx} component of the moment of inertia of the link 1 in the Raven II

has no bearing in the dynamics. In fact, only the I_{zz} component of link 1's inertia tensor appears in the entire dynamics.

Certain parameters appear in the equations only in linear combination with other terms and therefore cannot be uniquely identified. Taking the Raven II as an example, the parameters I_{2xz}, I_{3xy} and I_{3yz} can only be estimated as the linear combination $0.5I_{2xz} + 0.394I_{3xy} + 0.3078I_{3yz}$. Such an estimation is sufficient for computing closed form inverse-dynamics. Note that the numerical values of all the link inertias are required if the inverse dynamics is computed using the recursive Newton-Euler formulation [8].

Put more formally, the dynamic parameters can be classified into three groups: fully identifiable, identifiable in linear combinations, and completely unidentifiable [13].

If there are parameters that are identifiable only in linear combinations, the observation matrix W corresponding to the set of parameters χ is rank deficient (some columns of W are linearly dependent whatever the values of q, q and q). Also, the columns of W corresponding to the unidentifiable parameters would be zero. In order to obtain a unique solution, we have to determine a set of independent identifiable parameters, which are also called base dynamic parameters or minimum dynamic parameters. This means we would need to reformulate the χ vector, so that each of its elements is either a parameter that is fully identifiable or a linear combination of parameters belonging to the second group. Note that the way we choose to combine the parameters of the second group is not unique.

An algorithm to identify the base dynamic parameters is outlined in Chapter 9 in [13]. However, it can also be done by manually inspecting the equations, particularly the entries of the M matrix, as has been done here for the Raven II.

The dynamic parameters of the Raven II are categorized into the three groups as follows:

Fully Identifiable: $m_2, m_3, pc_{21}, pc_{22}, pc_{31}, pc_{32}, pc_{33}$ and all of the motor and joint friction coefficients

Identifiable in linear combinations/ products

$$m_1, pc_{11}, pc_{12}, I_{1zz}, I_{2yz}, I_{2xx}, I_{3yy}, I_{2xy}, I_{2xz}, I_{2yy}, I_{2yz}, I_{2zz}, I_{3xx}, I_{3xy}, I_{3xz}, I_{3yz}, I_{3zz}$$

Unidentifiable: $pc_{13}, pc_{23}, I_{1xx}, I_{1xy}, I_{1xz}, I_{1yy}, I_{1yz}$

Consequently, the total number of dynamic parameters to be actually identified reduces from 42 to only 27 (15 Inertia-related parameters and 12 friction-related parameters). This number can be further reduced. If, just for the purpose of parameter identification, we ignore the cable dynamics (assuming that it wouldn't affect the overall dynamics much), the number of independent states and thereby the order of the overall dynamics decreases by 3. This is because each of the 3 cable couplings between a given joint and the corresponding motor introduces an additional, under-actuated degree of freedom due to extension in the cables. So now, the motor angle and the joint angle are proportional, and the constant of proportion is dependent on the gearbox ratios and the pulley radii, i.e on parameters that are known constants. In the observation matrix W , this leads to the columns corresponding to motor velocities and the joint velocities to be linearly dependent, allowing us to combine the friction coefficients of the motors and the corresponding joints together. The χ vector now becomes a 21x1 vector.

The vector χ is shown below:

$$
\begin{bmatrix}
0.1607I_{3xx} + 0.2511I_{3xz} + 0.2588I_{2zz} + 0.0981I_{3zz} \\
0.933I_{2xx} + 0.0416I_{3xx} + 0.065I_{3xz} + 0.933I_{3yy} + I_{1zz} + 0.06699I_{2zz} + 0.02539I_{3zz} \\
0.933I_{2xy} - 0.5744I_{3xy} + 0.7352I_{3yz} \\
-0.2426I_{3xx} + 0.121I_{3xz} + 0.5I_{2yz} + 0.2426I_{3zz} \\
0.5I_{2xz} + 0.394I_{3xy} + 0.3078I_{3yz} \\
-0.933I_{2xx} + 0.3536I_{3xx} - 0.9053I_{3xz} + 0.933I_{2yy} - 0.933I_{3yy} + 0.5794I_{3zz} \\
m_3 \\
m_3 pc_{31} \\
m_3 pc_{33} \\
m_3 pc_{32} \\
m_1 pc_{11} \\
m_1 pc_{12} \\
m_2 pc_{21} \\
m_2 pc_{22} \\
m_2 pc_{32} \\
f_{c1} \\
f_{v1} \\
f_{c2} \\
f_{v2} \\
f_{c3} \\
f_{v3}
\end{bmatrix}
$$

4.3 Parameter Identification: Structure and Adaptive Algorithm

The problem of identification of all the link masses, the dynamic parameters and the kinematic parameters is a problem of non-linear estimation. However, the nonlinear dynamic equation can be linearized about their nominal values (taken from say, CAD models) to obtain a linear force/torque error model [16]:

$$\varepsilon_i = \tau_i - \tau_i^0 = \phi_i^T \Delta\psi_i$$

Where τ_i is the applied torque/force to link i, τ_i^0 is the computed torque/force at joint i from inverse dynamics using the nominal values of the parameters to be identified, $\Delta\psi_i$ is the correction vector of unknown parameters that affect the torque/force of link i, and ϕ_i is the nonlinear vector function of the kinematic and dynamic parameters and the output measurements (joint positions, velocities and accelerations).

If the value of the kinematic parameters are known, the identification of the remaining dynamic parameters now reduces to a problem of linear estimation. Therefore in our case, ϕ_i^T is just the i[th] row of the observation matrix W and $\Delta\psi_i$ is the estimation error of the vector χ. The adaptive estimation algorithm is illustrated in the figure below, and works in the following way: The joint positions, velocities, and accelerations, along with the current estimate of χ is used in the inverse dynamics of the model to compute the joint torque/force. The difference between the actual torque/force data and computed joint torque/force then gives the output error signal, ε. This term then drives an adaptive estimation error to make the estimation error, $\Delta\chi$ smaller.

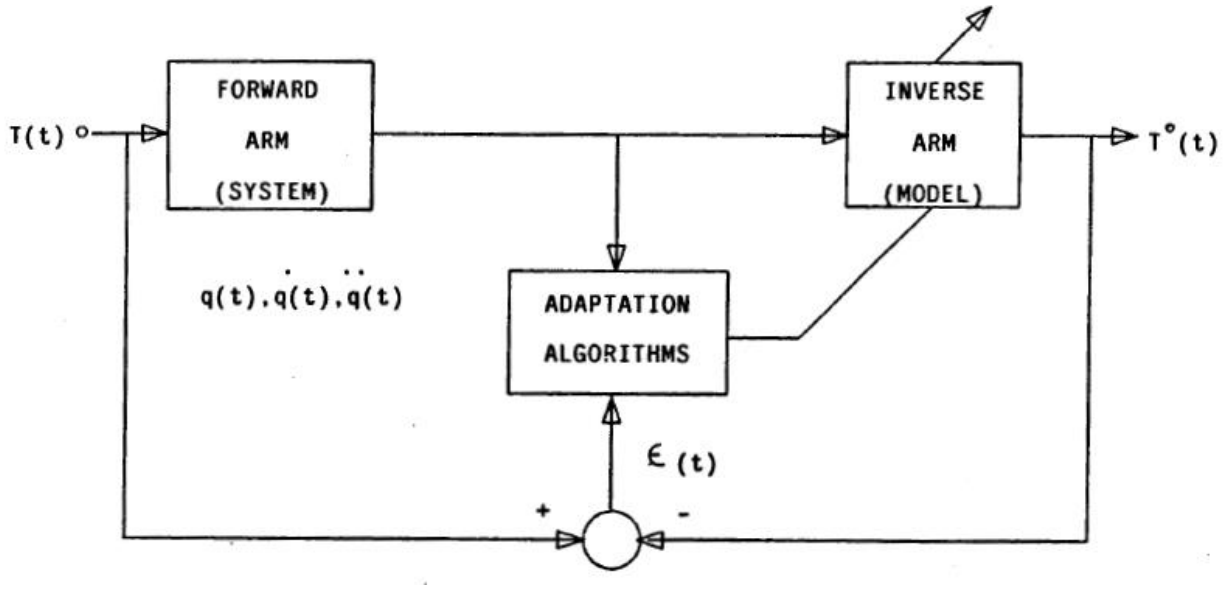


Figure 12: Input error structure for parameter estimation [8]

As remarked in [8], before experimentally implementing an estimation algorithm, one may wish to evaluate its robustness through simulation. The implementation of input-error structure is computation intensive due to the forward arm simulation. So, for simulation studies, an output-error structure (see figure below) for identification can be used instead. In this structure the difference between the outputs torques/forces of the system and the model is the error signal which drives the identification algorithm. Both of these structures can use the same adaptation algorithm, as they are similar in the sense that they both have the same error signal driving the adaptation algorithm. This is a practical advantage because the applicability and performance of an on-line estimation algorithm can be evaluated through off-line simulation studies. However, powerful simulation packages like SIMULINK enable us to handle computations of the input-error structure without much difficulty.

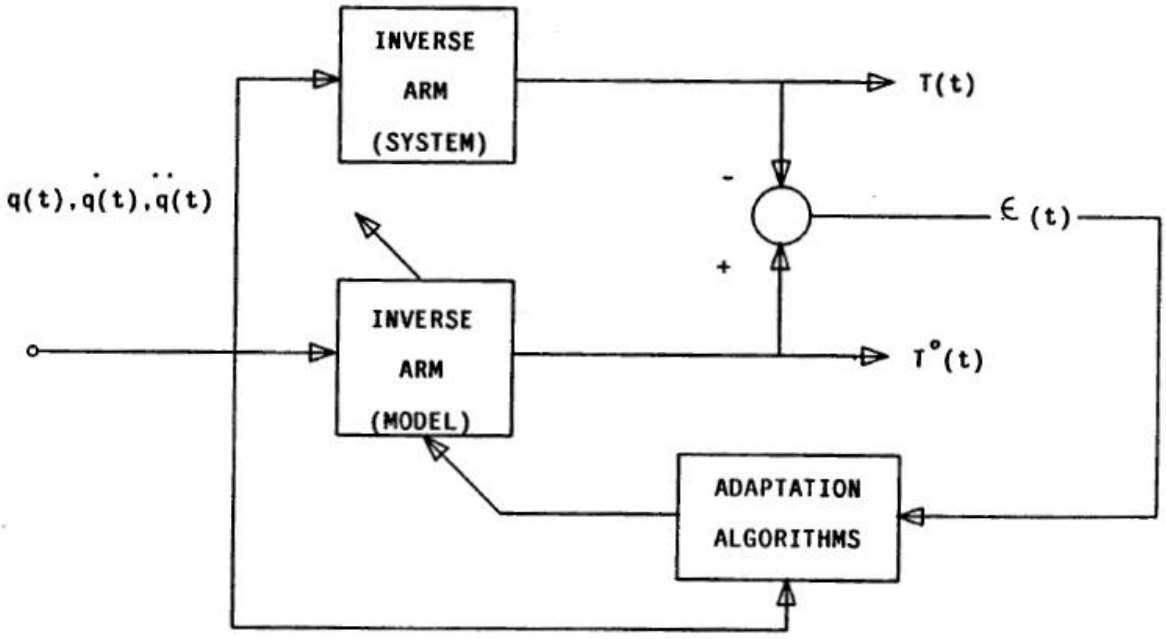


Figure 13: Output error structure for parameter estimation [8]

Chapter 5
Collision Avoidance for Lagrangian Systems

Now that we have developed a lagrangian dynamic model of the raven, we move on to the controls of a general Lagrangian system.

5.1 System Dynamics and Assumptions

We consider the problem of coordination of a system of N robotic agents with guaranteed collision avoidance. We assume that each robot can be described by the nonlinear Lagrange's equations

$$M_i(x_i)\ddot{x}_i + C_i(x_i, \dot{x}_i)\dot{x}_i = \tau_i + d_i, \text{ i=1,2,...,N} \quad (0.1)$$

where $x_i \in \mathbb{R}^n$ are the generalized coordinates, $M_i(x_i)$ is a positive definite inertia matrix $C_i(x_i, \dot{x}_i)$ is comprised of Coriolis and Centrifugal terms, $\tau_i \in \mathbb{R}^n$ is the input torque, and d_i is a disturbance. The disturbance d_i is assumed to be uniformly bounded, i.e,

$$\|d_i\| \le \rho_i < \infty, \forall i \in \{1, 2, ..., N\}$$

for some positive bounds ρ_i. Let $\rho = \max_{i \in \{1,...,N\}} \rho_i$. We assume that the agents' dynamics satisfy the following assumptions:

A1. $\exists$ constants $\bar{\sigma}_i$ and $\underline{\sigma_i}$ such that $\underline{\sigma_i} \le \|M_i(x_i)\| \le \overline{\sigma_i}$

A2. $\exists$ constants k_{C_i} such that $\|\dot{M}_i(x_i)\| \le 2\|C_i(x_i, \dot{x}_i)\| \le 2k_{C_i}\|\dot{x}_i\|$

A3. The matrices $M_i(x_i) - 2C_i(x_i, \dot{x}_i)$ are skew symmetric

These assumptions are satisfied in case of a wide class of robotic manipulators and therefore are not very restrictive.

5.2 Avoidance functions

In order to guarantee collision avoidance while the N robotic agents are coordinating their positions, we define avoidance functions that behave like barrier functions used in static optimization theory. An avoidance function is defined for each pair of agents as following:

$$V_{ij}^{a} = V_{ij}^{a}(x_i, x_j) = \left(\min\left\{ 0, \frac{\|x_i - x_j\|^2 - R^2}{\|x_i - x_j\|^2 - r^2} \right\} \right)^2 \quad (0.2)$$

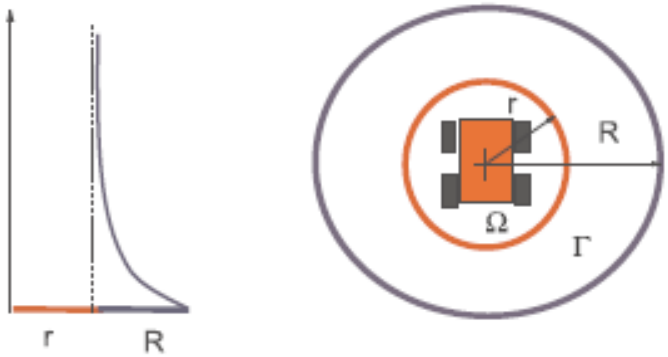


Figure 14: Collision avoidance barrier function

$\forall i, j \in \{1, 2, ..., N\} : i \neq j$ and $R > r > 0$. R denotes the radius of the region in which an agent can sense the presence of another agent, and r is the radius of the avoidance region i.e smallest allowable distance between two agents. A min function is in general not differentiable, however squaring the avoidance function, like in (0.2) makes it differentiable.

Note that such an avoidance function has a radial symmetry. However, in general, an avoidance function could take a more general, elliptical shape if, instead of using the Euclidean norm, we define the distance between the agent and the obstacle as

$$d_a = \sqrt{\left(\frac{x - x_a}{\alpha}\right)^2 + \left(\frac{y - y_a}{\beta}\right)^2}, \alpha, \beta > 0$$

Such an ellipsoidal avoidance function is useful in breaking symmetry when the need arises. Defining d_a with $\alpha = 1, \beta = 1$ gives us(0.2).

The partial derivative of $\frac{\partial V_{ij}^a}{\partial x_i} = \begin{cases} 0, \text{ if } \left\|x_i - x_j\right\| \geq R \text{ or } \left\|x_i - x_j\right\| < r \\ 4\frac{(R^2 - r^2)\left(\left\|x_i - x_j\right\|^2 - R^2\right)}{\left(\left\|x_i - x_j\right\|^2 - r^2\right)^3}\left(x_i - x_j\right)^T, \text{ if } R > \left\|x_i - x_j\right\| > r \end{cases}$ (0.3)

A thing to notice about the functions $V_{ij}^a(x_i, x_j)$ is that they are symmetric with respect to their arguments. Therefore, the partial derivative of $V_{ij}^a(x_i, x_j)$ with respect to x_j can be obtained directly from (0.3) by swapping the indices i and j:

$$\frac{\partial V_{ij}^a}{\partial x_i} = -\frac{\partial V_{ij}^a}{\partial x_j} = \frac{\partial V_{ji}^a}{\partial x_i} = -\frac{\partial V_{ji}^a}{\partial x_j} \quad \forall i, j \in \{1, 2, ..., N\} \quad (0.4)$$

5.3 Communication Graph for Coordination:

Graphs are a handy tool to describe the information exchange among the agents in a multi-agent network. Let $G = (V, E, A)$ be a graph, where $V = (v_1, ..., v_n)$ is a set of vertices and $E \subseteq \{(v_i, v_j) : \mathrm{v}_i, v_j \in V, v_i \neq v_j\}$ is a set of edges. Then, $(v_i, v_j) \in E$ means that agent j can obtain information from agent i. The graph G is directed if $(v_i, v_j) \in E$ doesn't necessarily mean $(v_j, v_i) \in E$ and undirected if $(v_i, v_j) \in E \Leftrightarrow (v_j, v_i) \in E$. A path from node v_{i1} to node v_{ik} is a sequence of edges $(v_{i_1}, v_{i_2}), (v_{i_2}, v_{i_3}), ..., (\mathrm{v}_{i_{k-1}}, v_{i_k})$ with $(v_{i_{j-1}}, v_{i_j}) \in E$ or $(v_{i_j}, v_{i_{j-1}}) \in E$ for

$j \in \{2,...,k\}$. A graph is said to be connected if every two vertices can be connected by a path. In this thesis, we only consider the multi-agent network with a undirected, connected, communication graph. The matrix $A \in \mathbb{R}^{n \times n}$ denotes the adjacency matrix of the graph with nonzero elements satisfying $a_{ij} > 0 \Leftrightarrow (v_i, v_j) \in E$. For an undirected graph G , the adjacency matrix A is symmetric.

A neighbor of an agent $\mathcal{N}_i$ is defined as the set of all other agents that can communicate with agent i i.e $a_{ij} > 0$. If agent i communicates with agent j , then we denote that symbolically as $i \sim j$.

$$\mathcal{N}_i = \{ j \in \{1,2,...,N\} : i \sim j\},\ i \in \{1,2,...,\mathrm{N}\}$$

The problem of coordination and collision avoidance in a communication network with a switching topology, that is undirected and directed, and jointly-connected has been studied in [17].

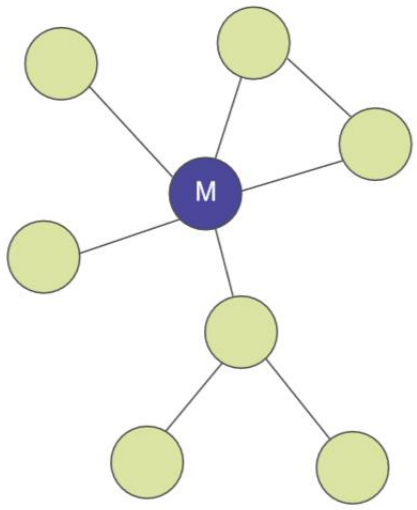


Figure 15: A connected, undirected communication graph. The network topology is assumed fixed throughout the control objective. The node denoted by M is the master agent.

5.4 Control Law for Avoidance and Coordination

In order to achieve coordination amongst the N agents, a linear feedback control law is used for each agent such that the torque input for a given agent i would contain a term proportional to the position error between agent i and agent j, if $i \sim j$. So if agent i and j communicate with each other, then τ_i and τ_j will contain an element of the form $-k(x_i - x_j)$ and $-k(x_j - x_i)$, respectively, where $k > 0$ is a scalar gain. This component of the input torque would try to drive the position error to zero. Although that cannot happen because of the collision avoidance terms in the control law, but it is shown that all the agents converge to some ball of a certain radius around the origin. The control law of the master agent (assumed to be agent 1, without loss of generality) has an additional term of form $-kx_1$, which ensures the convergence of the entire group to a set point (in this case, the origin) provided to the master. The control law for the master agent is given by

$$\tau_1 = \underbrace{-b\dot{x}_1 - kx_1 - k\sum_{\forall j \in \mathrm{N}_1}(x_1 - x_j)}_{\text{coordination}} \underbrace{-\left\{\begin{array}{r} \rho\dfrac{\dot{x}_1}{\|\dot{x}_1\|},\ \dot{x}_1 \neq 0 \\ 0,\ \dot{x}_1 = 0 \end{array}\right\}}_{\text{disturbance mitigation}} + \underbrace{\sum_{j \neq 1}\frac{\partial V_{ij}^{a\ T}}{\partial x_j}}_{\text{collisoin avoidance}} \quad (0.5)$$

And for agents $i = \{2, 3, ..., N\}$, the control law is given by

$$\tau_i = \underbrace{-b\dot{x}_i - k\sum_{\forall j \in \mathrm{N}_i}(x_i - x_j)}_{\text{coordination}} \underbrace{-\left\{\begin{array}{r} \rho\dfrac{\dot{x}_i}{\|\dot{x}_i\|},\ \dot{x}_i \neq 0 \\ 0,\ \dot{x}_i = 0 \end{array}\right\}}_{\text{disturbance mitigation}} + \underbrace{\sum_{j \neq i}\frac{\partial V_{ij}^{a\ T}}{\partial x_j}}_{\text{collisoin avoidance}} \quad (0.6)$$

Where $b > 0$ and ρ is as defined earlier.

It is shown in [18] that the above controller would achieve guaranteed collision avoidance while leading the entire group of agents to converge to a region around the origin. In general the, the

controller for collision avoidance may interfere with the coordination objective of each vehicle, but it is shown that the effects of internal forces that provides this collision avoidance guarantee is bounded.

The closed loop system can be rewritten as one big system of nN states as follows. This makes the stability and collision avoidance proofs more tractable.

$$M\ddot{X} + C(X, \dot{X})\dot{X} = -b\dot{X} - kS^T SX - \rho N(\dot{X}) - L(\mathrm{X}) + \mathrm{D} \quad (0.7)$$

Where $X^T = [\mathrm{x}_1^T, \mathrm{x}_2^T, ..., x_N^T]^T \in \mathbb{R}^{nN}$ is the position vector, $D^T = [\mathrm{d}_1^T, ..., \mathrm{d}_N^T]^T$ is the disturbance vector, $N^T(\dot{X}) = \left[\begin{cases} \frac{\dot{x}_1^T}{\|\dot{x}_1\|}, \dot{x}_1 \neq 0 \\ 0^T, \ \dot{x}_1 = 0 \end{cases}, ..., \begin{cases} \frac{\dot{x}_N^T}{\|\dot{x}_N\|}, \dot{x}_N \neq 0 \\ 0^T, \ \dot{x}_N = 0 \end{cases} \right]^T$ is a component-wise normalized velocity vector, and $L(X) = \left[\sum_{j \in N_1} \frac{\partial V_{ij}^a}{\partial x_j}, ..., \sum_{j \in N_N} \frac{\partial V_{Nj}^a}{\partial x_j} \right]^T$ is the collision avoidance control part. The inertia and Coriolis matrices are block diagonal and are given by $M = diag\{M_1(x_1), ..., M_N(x_N)\}$ and $C = diag\{C_1(x_1, \dot{x}_1), ..., C_N(x_N, \dot{x}_N)\}$ respectively. The matrix S is the connection matrix and is defined as: Let E contain x_1 as an element as well as all the error vectors of the form $x_i - x_j$ if $i \sim j$ are neighbors. Then, S is defined as $E = SX$, with the first element in E being x_1. Since we assume that the communication graph is connected, then S has full column rank and therefore $S^T S$ is a symmetric positive definite matrix.

5.5 Lyapunov Based Proofs for Collision Avoidance and Coordination

In order to prove the claims made above, certain formal definitions about avoidance and detection regions are needed.

The avoidance set for each pair of agents is defined as $\Omega_{ij} = \{X : X \in \mathbb{R}^{nN}, \|x_i - x_j\| \leq r\}$

And the detection region is defined pairwise as $\mathcal{D}_{ij} = \{X : X \in \mathbb{R}^{nN}, \|x_i - x_j\| \leq R\}$

Then, the overall avoidance region and the overall detection region is defined respectively by

$$\Omega = \bigcup_{i,j \in N, j>i} \Omega_{ij}$$
$$\mathcal{D} = \bigcup_{i,j \in N, j>i} \mathcal{D}_{ij}$$

The augmented state dynamical system $\dot{X} = f(X, U(X))$ avoids $\Omega, \Omega \subset \mathbb{R}^{nN}$, if and only if for each solution $X(t, X_0), t \geq 0, X_0 \notin \Omega \Rightarrow X(t, X_0) \notin \Omega, \forall t \geq 0$.

Lemma 1. Hokayem et.al: Consider N agents with Lagrangian dynamics (1.1), and the control laws given above. Let the initial configuration $X(0) \notin \Omega$, where Ω is avoidance set. Then, the set Ω is avoidable in the sense defined above.

Proof:

Consider the Lyapunov type function

$$V_{col} = \frac{1}{2}\dot{X}^T M\dot{X} + \frac{1}{2}kX^T S^T SX + \frac{1}{2}\sum_{i=1}^{N}\sum_{j \neq i} V_{ij}^a$$

Then the derivate of this function along the trajectories of the augmented system (0.7) is given by

$$\dot{V}_{col} = \dot{X}^T(-C(X,\dot{X})\dot{X} - b\dot{X} - \rho N(\dot{X}) - kS^T SX - L(X) + D) + \frac{1}{2}\dot{X}^T\dot{M}\dot{X} + kX^T S^T S\dot{X}$$
$$+ \frac{1}{2}\sum_{i=1}^{N}\sum_{j \neq i}\left(\frac{\partial V_{ij}^a}{\partial x_i}\dot{x}_i + \frac{\partial V_{ij}^a}{\partial x_j}\dot{x}_j\right)$$

Now, using the assumptions A1 and A3, $\dot{V}_{col}$ can be upper bound as:

$$\dot{V}_{col} \le -b\left\|\dot{X}\right\|^2 + \dot{X}^T(D-\rho N(\dot{X})) - \dot{X}^T L(X) + \underbrace{\frac{1}{2}\sum_{i=1}^{N}\sum_{j\neq i}\left(\frac{\partial V_{ij}^a}{\partial x_i}\dot{x}_i + \frac{\partial V_{ij}^a}{\partial x_j}\dot{x}_j\right)}_{=\dot{X}^T L(X),\text{ by change of indices}}$$

Using assumption A2, we get

$$\dot{X}^T(D-\rho N(\dot{X})) \le \sum_{i=1}^{N}\|x_i\|(\|d_i\|-\rho) \le 0$$

Therefore, $\dot{V}_{col}$ is bounded as

$$\dot{V}_{col} \le -b\left\|\dot{X}\right\|^2 \le 0$$

This means that the function V_{col} is non-increasing in the sensing region $\mathcal{D}$. Now, it can be seen that for finite values of X that are outside the region Ω, the function V_{col} is finite. Also, as any two agents approach a distance of r from one another, the function V_{col} blows up because

$$\lim_{\|x_i-x_j\|\to r^+} V_{ij}^a(x_i,x_j) = +\infty, \forall i, j \in \{1,...,N\}, i \neq j$$

Therefore we can conclude from the continuity of the solution of the augmented system, that the partial state trajectory $X(t, X(0), \dot{X}(0))$ will never enter Ω, assuming that $X(0) \in \Omega^C$.

Note that lemma 1 only says that collisions are guaranteed to not occur. It says nothing about the whereabouts of various agents. We can however study the effect of avoidance terms $\frac{\partial V_{ij}^a}{\partial x_i}$ on the performance of the agents by developing certain bounds.

Since we have $\dot{V}_{col}$ and $V_{col} \ge 0$, it implies that $V_{col}(t) \le V_{col}(0), \forall t \ge 0$. This gives us

$$\frac{1}{2}\sum_{i=1}^{N}\sum_{j\neq i}V_{ij}^a \le V_{col}(t) \le V_{col}(0), \forall t \ge 0$$

As this holds for every time $t \geq 0$, in particular, we have

$$\max_{i\in\{1,\ldots,N\},\, j\neq i} \sup_{t\geq 0} V_{ij}^{a} \leq 2V_{col}(0)$$

Due to the continuity and monotonicity of V_{ij}^{a} , there exists a constant $\underline{r} \in (r, R)$,with $\underline{r} > r$,such that

$$\inf_{t\geq 0} \left\| x_i - x_j \right\| \geq \underline{r}, \forall i \in \{1,\ldots,N\},\ j \neq i$$

Using this inequality, we can obtain a bound on the partial derivate, $\frac{\partial V_{ij}^{a}}{\partial x_i}$ as follows:

$$\sup_{t\geq 0} \left\| \frac{\partial V_{ij}^{a}}{\partial x_i} \right\| = \sup_{t\geq 0} \left\| 4\frac{(R^2 - r^2)\left(\left\| x_i - x_j \right\|^2 - R^2\right)}{\left(\left\| x_i - x_j \right\|^2 - r^2\right)^3}\left(x_i - x_j\right)^T \right\| \leq \frac{4R(R^2 - r^2)(R^2 - \underline{r}^2)}{(\underline{r}^2 - r^2)^3} \doteq g$$

where g is a bounded positive constant. Therefore, the effect of the collision avoidance terms on the overall system can be viewed as a disturbance and can be upper bounded as

$$\sup_{t\geq 0} \left\| L(X(t)) \right\| \leq \sqrt{N} \max_{i\in\{1,\ldots,N\}} \sum_{j\neq i} \sup_{t\geq 0} \left\| \frac{\partial V_{ij}^{a}}{\partial x_i} \right\| \leq N^{3/2} g$$

Thus, it can be seen that the effect of the collision avoidance terms can only interfere with the stability and coordination objective in some known bounded fashion characterized by equation above.

Theorem1:

Consider the augmented closed loop system (0.7) with a connected coordination graph. Then, for any initial condition $\xi(0) = [X^T(0), \dot{X}^T(0)]^T$, such that $X(o) \notin \Omega$, there exists positive

control gains b and k large enough such that the trajectory of the augmented system avoids the set Ω, and it satisfies the ultimate bound

$$\|\xi(t)\| \leq \sqrt{\frac{c_2}{c_1}} 2 \frac{\sqrt{N}}{c_3} \sqrt{(Ng)^2 + (2\rho + Ng)^2}, \forall t \geq T$$

Where $T \geq 0$ is a finite time, c_1 and c_2 are positive constants defined in the proof, c_3 is an arbitrarily chosen positive constant, and N, g, and ρ are as defined earlier. Moreover, the positions of all the slave agents ultimately converge to a bounded set around the master's position.

Sketch of proof:

Consider the following candidate Lyapunov function:

$$V_s = \frac{1}{2} \dot{X}^T M \dot{X} + \frac{1}{2} k X^T S^T Z S X + \epsilon \varphi^T(X) M \dot{X}$$

Where $\epsilon > 0$ and $\varphi(X) = \frac{X}{1 + \|X\|}$. Using property P1 and the fact that S has full column rank, it can be shown that for a small enough ϵ there exists two constants c_1 and c_2 such that the following quadratic bound holds on V_s :

$$c_1 \|\xi\|^2 \leq V_s \leq c_2 \|\xi\|^2 \quad (0.8)$$

The derivative of V_s along the trajectories of the augmented closed-loop system is given by:

$$\dot{V}_s = \dot{X}^T \left(-C(X, \dot{X}) \dot{X} - b\dot{X} - \rho N(\dot{X}) - k S^T S X - L(X) + D \right) + \frac{1}{2} \dot{X}^T \dot{M} \dot{X} + k X^T S^T S \dot{X} +$$
$$\epsilon \varphi^T(X) M \dot{X} + \epsilon \varphi^T(X) \left(-C(X, \dot{X}) \dot{X} - b\dot{X} - \rho N(\dot{X}) - k S^T S X - L(X) + D \right)$$

Using properties A.1-A.3, it can be shown that

$$\dot{V}_s \le -b\left\|\dot{X}\right\|^2 - \epsilon\frac{k\|S\|^2\|X\|^2}{1+\|X\|} + \epsilon b\|\varphi(X)\|\left\|\dot{X}\right\| + \left\|\dot{X}\right\|\|L(X)\| + \epsilon(2\bar{\sigma}+k_c)\left\|\dot{X}\right\|^2 + \epsilon\|\varphi(X)\|\left(\left\|\rho N(\dot{X})\right\| + \|L(X)\| + \|D\|\right)$$

Now from Lemma 1, we have $\|L(X)\| \le N^{3/2}g$ and from the bounded disturbance assumption, we have $\|D\| \le \sqrt{N}\rho$. Therefore, $\dot{V}_s$ can be further upper bounded as

$$\dot{V}_s = -\begin{bmatrix}\left\|\dot{X}\right\| \\ \|X\|\end{bmatrix}^T \begin{bmatrix} b-\epsilon(2\bar{\sigma}+k_c) & \frac{\epsilon b}{2(1+\|X\|)} \\ \frac{\epsilon b}{2(1+\|X\|)} & \frac{\epsilon\|S\|^2}{1+\|X\|}\end{bmatrix}\begin{bmatrix}\left\|\dot{X}\right\| \\ \|X\|\end{bmatrix} + \begin{bmatrix}\left\|\dot{X}\right\| \\ \|X\|\end{bmatrix}^T \begin{bmatrix}1 & 0 \\ 0 & \frac{\epsilon}{1+\|X\|}\end{bmatrix}\begin{bmatrix}N^{3/2}g \\ 2\rho+N^{3/2}g\end{bmatrix}$$

Pick any $c_3 > 0$ and choose control gains such that $\min\{b,k\} > \frac{c_3}{\|S\|^2}$. Then, for

$\varepsilon < \min\left\{\frac{b-c_3}{2\bar{\sigma}+k_c}, \frac{2c_3}{b}\right\}$, the following condition is satisfied

$$\|\xi\| \ge \frac{2\sqrt{N}}{c_3}\sqrt{(Ng)^2+(2\rho+Ng)^2} \Rightarrow \dot{V} \le -\frac{c_3}{2}\begin{bmatrix}\left\|\dot{X}\right\| \\ \|X\|\end{bmatrix}^T \begin{bmatrix}1 & 0 \\ 0 & \frac{\varepsilon}{1+\|X\|}\end{bmatrix}\begin{bmatrix}\left\|\dot{X}\right\| \\ \|X\|\end{bmatrix} \quad (0.9)$$

Using (0.8) and (0.9), and by using the comparision principle, there exists a function $\beta \in \mathcal{KL}$ such that

$$V_s \le \max\left\{\beta(c_2\|\xi(0)\|^2, t), c_2\frac{4N}{c_3^{\,2}}\left((Ng)^2+(2\rho+Ng\right)^2\right\}$$

Using the lower bound in (0.8),

$$\|\xi(t)\| \le \max\left\{\sqrt{c_1^{-1}\beta(c_2\|\xi(0)\|^2, t)}, \sqrt{\frac{c_2}{c_1}}\frac{2\sqrt{N}}{c_3}\sqrt{(Ng)^2+(2\rho+Ng)^2}\right\}, \forall t \ge 0$$

But since $\beta \in \mathcal{KL}$, we know it is monotonically decreasing in time, and therefore for some time T, we have $\sqrt{c_1^{-1}\beta(c_2\|\xi(0)\|^2, t)} < \sqrt{\frac{c_2}{c_1}}\frac{2\sqrt{N}}{c_3}\sqrt{(Ng)^2 + (2\rho + Ng)^2}, \forall t > T$ and so, we can establish the bound on $\xi(t)$.

One can further show that the E matrix defined in previous section can be bounded as:

$$\|E(\mathrm{t})\| = \|SX(t)\| \le \|S\|\|X(t)\| \le \|S\|\|\xi(t)\| \le \|S\|\sqrt{\frac{c_2}{c_1}}\frac{2\sqrt{N}}{c_3}\sqrt{(Ng)^2 + (2\rho + Ng)^2}, \forall t \ge T$$

Since the coordination graph is assumed to be connected, the error between master agent 1 and any agent j can be written as $x_1 - x_j = (x_1 - x_{l_1}) + (x_{l_1} - x_{l_2}) + ... + (x_{l_k} - x_j)$ where $l_1, l_2, ..., l_k$ are the agents that connect agent 1 and agent j in the communication graph. Since $\|E(t)\|$ is bounded, for all j, the term $\|x_1 - x_j\|$ is bounded. Thus, we can conclude that all the agents converge to some ball around the master agent.

An interesting treatment of control of network of multiple lagrangian systems with communication delay and quantization error can be found in [19].

Chapter 6
Results

The simulation shown here illustrates the collision avoidance and coordination among a group of 4 agents, when we apply the controllers (0.5) and(0.6). For the first case, we consider the case of communication without any time delay. Each of the 4 agents have a double integrator dynamics given by:

$$\ddot{x}_i = \tau_i + d_i, \forall i \in \{1,...,4\}$$

The agent 1 is chosen as the master agent and is used to drive the whole formation to a given point. The disturbance for agent 1 is taken to be $d_1 = \begin{bmatrix} 5 \\ 5 \end{bmatrix}$ and for the remaining agents, the disturbance is chosen to be uniformly distributed in the range of [-3,3]. The communication graph is taken to be $1 \sim 2 \sim 3 \sim 4$. Thus, the adjacency matrix is given as:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

The figure below shows this graph. The parameters for the control law are chosen to be $b = k = 10$ and $\rho_i = 10 \forall i \in \{1,...,4\}$, such that the requirement of uniform boundedness of the disturbance input is satisfied. The sensing radius R and the avoidance radius r are taken to be 20 and 10 respectively.

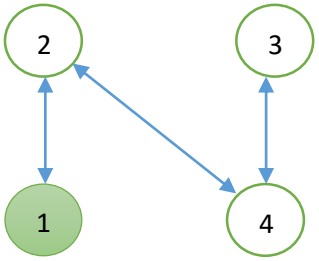


Figure 16: Communication graph and initial relative position of the agents

The agents are initially placed at the corners of a square of length 200 units. The point to which we want agent 1 to converge to is the center of the square. The figure below shows the position of the agents at different times.

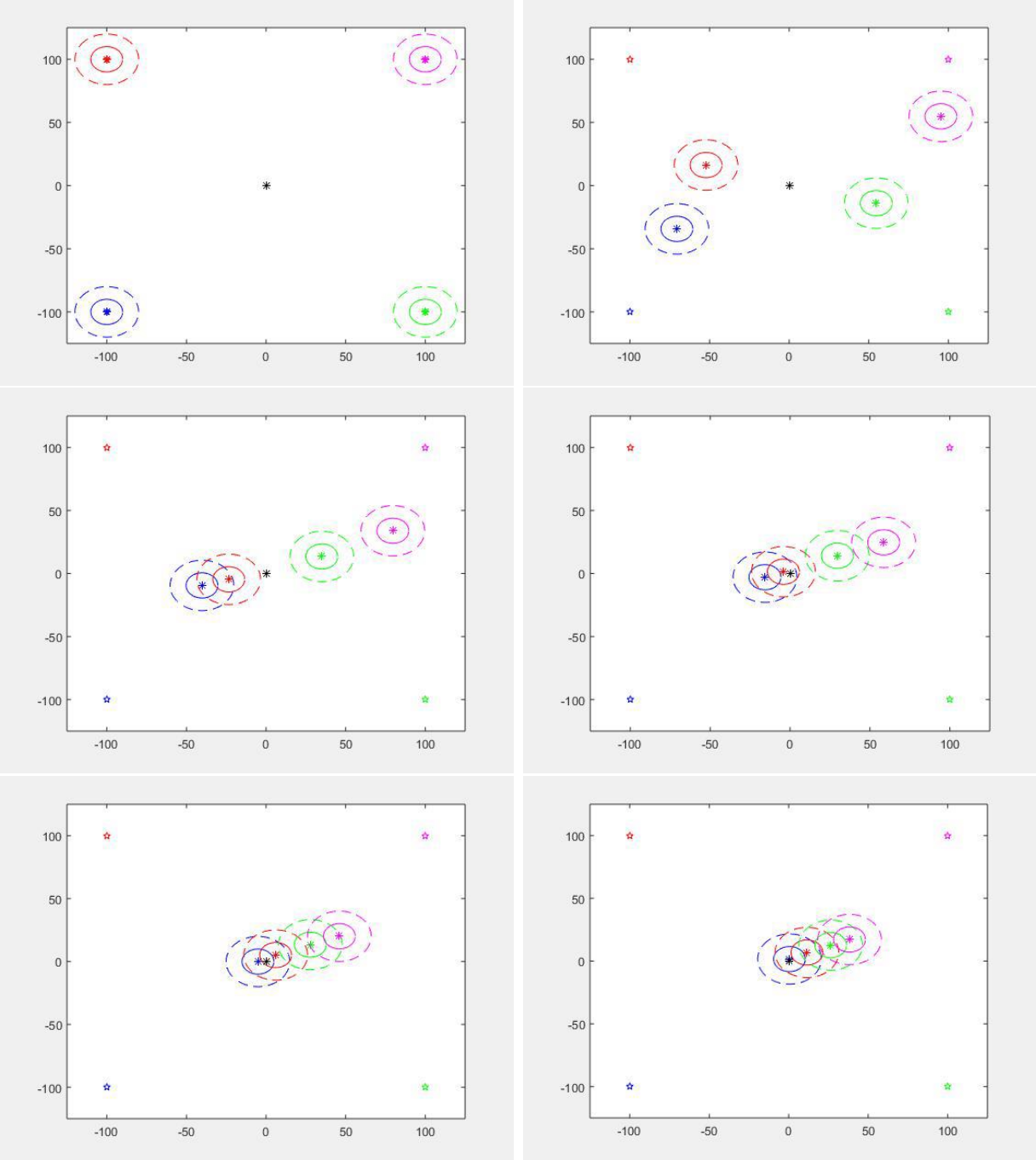
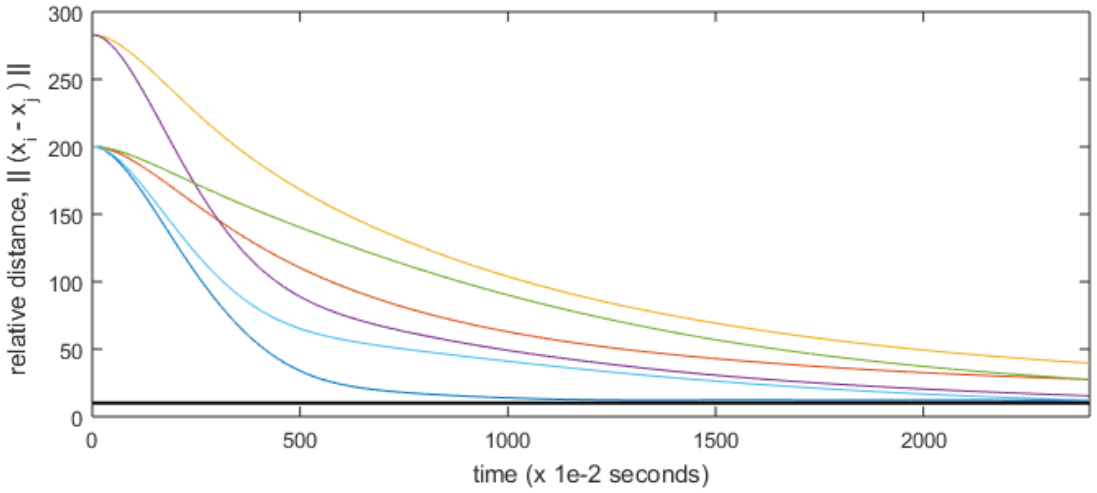
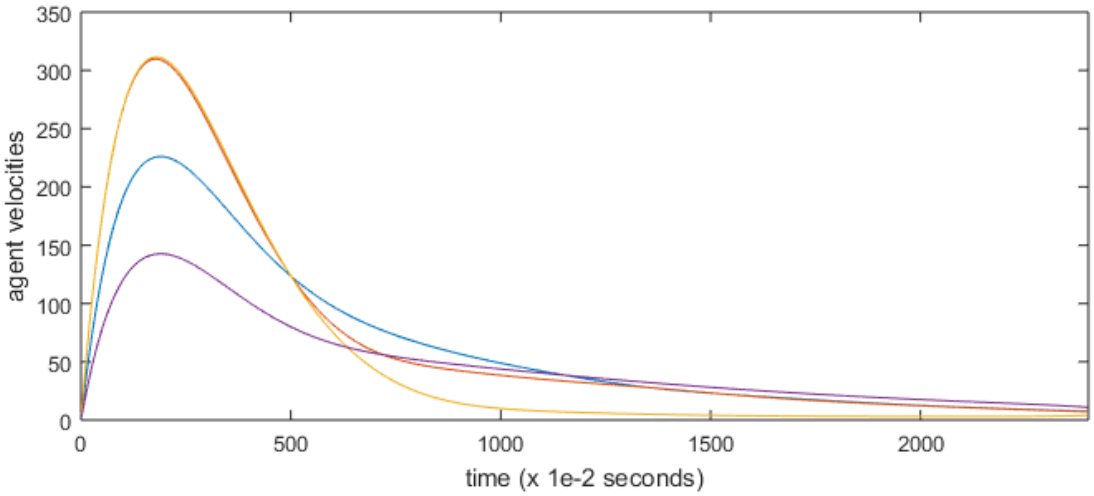


Figure 17: Snapshots of the agents' positions at t=0 s (top left), t=3 s (top right), t=7 s (middle left), t=13 s (middle right), t=18 s (bottom left), t=24 s (bottom right)

The stars mark the starting point of each agent. The current position of each agent is marked by the asterisk (*). The origin, to which agent 1 ought to converge, is marked as a black colored asterisk. The avoidance region for an agent is shown as a solid disk, while the dashed disks represent the sensing region. In the simulation above it can be seen that roughly at time t = 7 seconds, agent one and two enter each other's sensing regions. At and after around t = 18 seconds, agents 1 and 2 are very close, but still outside each other's avoidance region, during which time agent 3 enters agent 2's sensing region and agent 4 enters agent 3's sensing region. At time t = 24 seconds, it can be seen that agent 1 has converged to the origin, and the other agents are very close by.



(a)



(b)

Figure 18: (a) Relative distance between all agents (b) Velocity norms of all agents

In part (a) of the above figure, the bold black line indicates the boundary of avoidance region. Since the curve for every agent pair (six in total) lies above this line, it indicates that there is no collision, as we would expect. Moreover, we see that each of those norms converge over time. Also from part (b) we note that the velocities of each agent converges to zero.

Chapter 7
Conclusion

This thesis talked about multi-agent systems with Lagrangian dynamics and studied a controller that guarantees collision avoidance in presence of bounded disturbances while allowing the agents to perform the task of coordination.

The thesis was broadly divided into two parts. The first part involved in some detail, the derivation of the dynamics of the surgical robot, Raven II. The dynamics of the kinematic chain was derived using Euler-Lagrange approach. A second order linear dynamics was considered for the motor while the cable transmission was modelled using cable strain-torque relationships. Also, techniques to estimate the kinematic parameters of the robot was discussed. Because a cable transmission mechanism adds a lot of nonlinearities and modelling challenges, the open loop performance of the model did not accurately reflect the actual data from experiments, however, the closed loop performance was found to be good enough, given an adequately fast controller. The main purpose behind the first part was not to come up with a precise dynamic model but to illustrate a Lagrangian system and its dynamics, since Lagrangian dynamics are very commonly used to model the agents in a multi-agent system.

In the second part, a unified framework was presented that guarantees the safety of multiple agents with a coordination objective. It was shown that the controller not only guarantees collision avoidance, but the effects of collision avoidance term in the controller is bounded and does not compromise or interfere much with other control objectives that the agents might have. The simulation results verify that this.

Further work on this thesis could include considering communication delays. Along lines of the works of Hokayem et.al [19], it would be interesting also to consider the robustness of collision avoidance controller to quantization errors. Another case that may be of greater practical application would be to consider communication topologies that are not fixed but switching. Although this has been studied in [17] in context of velocity consensus, combining it with coordination would seem like a worthwhile effort.

References

[1] O. Khatib, "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots," in *The International Journal of Robotics Research*, 1986.

[2] G. Leitmann, "Guaranteed Avoidance Strategies," *Journal of Optimization Theory and Applications,* vol. 32, pp. 569-576, 1980.

[3] G. Leitmann and Skowronski, "A note on Avoidance Control," *Optimal Control Application and Methods,* vol. 4, pp. 335-342, 1983.

[4] D. M. Stipanovic, "A Survey and Some New Results in Avoidance Control," in *International Workshop on Dynamics and Control*, Spain, 2009.

[5] H. King, S. N. Kosari, B. Hannaford and J. Ma, "Kinematic Analysis of th eRaven-IITM Research Surgical Robot Platform," Department of Electrical Engineering, The University of Washington, 2014.

[6] J. Rosen, M. N. Sinanan, B. Hannaford and H. King, "The Raven : Design and Validation of a Telesurgery System," in *The International Journal of Robotics Research*, 2009.

[7] B. Hannaford, H. King and et.al, "Raven-II: An Open Platform for Surgical Robotics Research," in *IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING*, 2013.

[8] P. Khosla and T. Kanade, "An Algorithm to Estimate Manipulator Dynamics Parameters," Department of Electrical and Computer Engineering, The Robotics Institute, Carnegie Mellon University, 1987.

[9] M. M. Fateh, "Dynamic Modeling of Robot Manipulators in D-H Frames," *World Applied Sciences Journal,* vol. 6, no. 1, pp. 39-44, 2009.

[10] R. Murray, Z. Li and S. Sastry, A Mathematical Introduction to Robotic Manipulation, CRC Press, 1994.

[11] M. W. Spong, S. Hutchinson and M. Vidyasagar, Robot Modeling and Control, Hoboken, NJ: John Wiley & Sons, 2005.

[12] M. Haghighipanah, Y. Li, M. Miyasaka and B. Hannaford, "Improving Position Precision of a Servo-Controlled Elastic Cable Driven Surgical Robot Using Unscented Kalman Filter," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* , 2015.

[13] W. Khalil and E. Dombre, Modeling, Identification and Control of Robots, Elsevier, 2004.

[14] P. F. Hokayem, D. M. Stipanović and M. W. Spong, "Coordination and collision avoidance for Lagrangian systems with disturbances," *Applied Mathematics and Computation,* vol. 217, no. 3, pp. 1085-1094, 2010.

[15] A. Stern, *Geometric discretization of Lagrangian mechanics and field theories,* Ph.D. thesis, California Institute of Technology, 2009.

[16] C. P. Neuman and P. K. Khosla, "Identification of Robot Dynamics: An Application of Recursive Estimation," in *Proceedings of the Fourth Yale Workshop on Applications of Adaptive Systems Theory*, Yale University, New Haven, 1985.

[17] Y. Wu, D. M. Stipanovic and Z.-H. Guan, "Distributed velocity consensus for Euler-Lagrangian networks with collision avoidance under different switching topologies," 2015.

[18] P. F. Hokayem, D. M. Stipanovic and M. W. Spong, "Coordination and Collision Avoidance for Lagrangian Systems with Disturbances," *Applied Mathematics and Computation,* vol. 217, no. 3, pp. 1085-1094, 2010.

[19] P. F. M. W. Hokayem, "Semiautonomous Control of Multiple Networked Lagrangian systems," in *International Journal of Robust and Nonlinear Control*, 2008.