ANALYSIS AND DESIGN OF MULTI-ARM ROBOTIC SYSTEMS MANIPULATING LARGE OBJECTS

BY

HO SIU YAN

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF PHILOSOPHY

DIVISION OF SYSTEMS ENGINEERING AND ENGINEERING MANAGEMENT THE CHINESE UNIVERSITY OF HONG KONG

JUNE 1995

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my supervisor, Dr X. Shi, for his encouragement and guidance throughout this work. Without his support, this work would not be carried out smoothly as it has been. I would also like to express my gratefulness to all my friends, especially those who have studied with me in CUHK, for their support and camaraderie. Finally, I would like to thank everyone who help me in completing this work.

ABSTRACT

This work is concerned with the manipulation of large objects with multi-arm robotic systems. Large object is defined here as objects which cannot be grasped with ordinary size grippers. The manipulation of such objects, in general, requires the use of multi-arm systems. In this study, the possibility of using multi-arm systems to manipulate large objects without the effect of friction is explored. Approaches for designing and analyzing grasps with frictionless contacts for polyhedral objects are proposed. Using the approaches, grasps provided by small grippers can be designed for large polyhedra. Moreover, an advantage of grasps with frictionless contacts is also demonstrated. An algorithm is derived for a certain type of object to determine position and orientation of the object from the contact locations of the grasp.

In order to analyze the multi-arm systems, a dynamic model is derived. The model utilizes a general joint model to describe the connection between the end effectors and the manipulated object. The general joint model proposed in this study does not just make the modelling of complicated joints possible but it also provides general approaches for the analysis of all types of joints.

To analyze the multi-arm system, computer simulation is helpful. An efficient algorithm is therefore proposed to solve the forward dynamics of the multi-arm system. This algorithm is an improvement over an existing one in that it is in a more general form and can be applied to a larger class of problems. It also avoids complicated computations in certain cases.

NOMENCLATURE

$\boldsymbol{e}$	contact normal, which is a 3×1 unit vector giving the direction pointing towards the corresponding surface and means the fixed vector passing through the contact point.
$\boldsymbol{f}$	wrench, which is expressed as follows:

$$f = \begin{bmatrix} \tilde{f} \\ \tilde{m} \end{bmatrix}$$

where $\tilde{f}$ is the force component

$\tilde{m}$ is the moment component

$\boldsymbol{h}$	torque or wrench due to the gravitational, Coriolis and centrifugal terms
$\boldsymbol{I}$	inertia
$\boldsymbol{I}_n$	the n×n identity matrix
$\boldsymbol{M}$	inertia matrix
$\boldsymbol{m}$	mass
$\boldsymbol{p}$	contact location
$\boldsymbol{q}$	general coordinates
$\boldsymbol{r}$	position and orientation
$\boldsymbol{v}$	velocity
$\boldsymbol{x}$	position vector
θ	a set of Euler angles
$\boldsymbol{0}_{i\times j}$	a $\boldsymbol{i\times j}$ matrix with all elements equal to zero

The symbol ${}_b\boldsymbol{a}_c^{\,d}$ is used to represent a vector contains variables describing the properties of a frame Σ_c relative to the frame Σ_b and expressed in the frame Σ_d. For example, ${}_b\boldsymbol{v}_c^{\,d}$ is used to represent the velocity of the frame Σ_c relative to the frame

Σ_b and expressed in the frame Σ_d.

If $\boldsymbol{a}$ is a 3×1 vector $\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}$, $\boldsymbol{a}\times$ is a skew symmetry 3×3 matrix $\begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}$.

When this matrix is post-multiplied to another 3×1 vector $\boldsymbol{b}$, the result is equal to the cross product $\boldsymbol{a}\times\boldsymbol{b}$.

Symbols with bar, tilde or hat are used to denote modified variables or to distinguish variables with similar meanings.

For a vector $\boldsymbol{a}$ and a matrix $\boldsymbol{A}$, $\dot{\boldsymbol{a}}$ and $\dot{\boldsymbol{A}}$ are the time derivatives of $\boldsymbol{a}$ and $\boldsymbol{A}$ respectively. $\ddot{\boldsymbol{a}}$ is the second time derivative of $\boldsymbol{a}$.

List of Figures

TABLE OF CONTENTS

CHAPTER ONE

INTRODUCTION

There are lots of reasons for the use of several robot arms to manipulate an object instead of only one. It gives a better result in some cases while it is essential in others. When manipulating a very heavy workpiece, using more than one arm may result in a more stiff system. As a result, the accuracy may be increased. A multi-arm system may also be used to manipulate an object exceeding the limit on the loading of each single arm. If the manipulated object is non-rigid, the use of a multi-arm system is a must. The non-rigid object may be a tool or two parts being assembled. Both of this two cases are usual in industrial application. Another process which requires the use of a multi-arm system is the manipulation of large objects. Large objects may be defined in two ways. First, it may be defined as an object which is much larger in size than the manipulators used. When only one arm is used to handle such an object, a large moment may act on the arm even when the mass of the object is not very large. Small disturbance acted on the object may cause a large disturbance moment to act on the arm. The use of more than one arm may remove these problems. Another definition for large objects is an object which cannot be grasped with ordinary size grippers. Although some special grippers or large grippers may be used to handle such objects. There are many advantages in using a multi-arm system to manipulate such objects and this is the subject of this study.

An object with a very large size may not be considered as a large object when the second definition of large object is applied. One example for this is a large object with a small handle. Conversely, under this definition, an object with just the size comparable to the size of the manipulators used can be classified as a large object. For example, a cube, whose sides have a length similar to those of the stretched arms, is considered as a large object. In order to grasp such a large object, more than one gripper must be used unless it can be handled with certain special grippers, e.g. a large gripper or a magnetic device if the object is made of ferrous materials. Therefore, if special end effectors are not used, this task must be carried out with multi-arm systems. When more robot arms are used, it is not just possible to handle a large object using ordinary grippers but the object may also be manipulated without the effect of friction. This is because more contacts may be provided at different locations using just simple grippers. The objective of this study is, thus, aimed at the study on the design and analysis of multi-arm robotic systems which can manipulate such type of large objects without friction.

To design such a multi-arm system, grasping theory should first be studied. Manipulation of object using several arms without grasping it rigidly resembles fine manipulation using a robot hand [1]. The main difference is that there may be more than one contacts for a multi-arm system between each of the manipulators and the object. Moreover, if large objects are manipulated, there will be more constraints on the contact locations. The contacts should be distributed such that the contacts can be provided by small grippers located at separate areas of the large object.

The study of grasping with frictionless contacts had been started more than a century ago as reported in [2]. The term form-closure was also introduced at that time to describe a set of contacts in a grasp which can constrain any motion of the grasped object, irrespective of the magnitude of the contact forces. Grasps using frictionless contacts have several important advantages over ones with friction. First, it dose not depend on friction which can be affected by many different factors including the materials of the contact surfaces, surface finish, cleanness of the contact surfaces, etc. It is difficult to determine the exact effect of all these factors but it may affect the frictional force significantly. Second, it is much easier to reduce friction than increase friction between two bodies. To reduce friction, bearings may be applied but it is impossible to increase the friction for some objects, e.g. a surface coated with grease or oil. Third, a grasp designed for frictionless contacts can also be achieved by contacts with friction. It was shown by Markenscoff et al. [2] that the presence of friction would only result in a better grasp. Moreover, the position and the orientation of the grasped object may be determined from the contacts in such grasps. The main drawback of frictionless contact grasp is that not all objects can be grasped with frictionless contacts. Objects with rotational symmetry cannot be grasped with frictionless contacts only [2]. In fact, most recent studies on grasping are concentrated on grasps using frictional contacts. This is because when friction is present, less contacts are required for the grasp. As a result, the construction and analysis of grasps using frictional contacts are simpler. Nguyen [3] presented approaches for constructing grasps using different types of contacts and concluded that the construction of grasps using frictionless contact is more expensive and harder. Moreover, as more contacts are required, the gripper required to provide the grasp would be more complicated. However, as pointed out above, the situation would be different when a multi-arm system is used. In such case, it is possible to grasp objects with frictionless contacts using ordinary grippers or robot hands.

In [3], Nguyen proposed an approach for constructing grasp for polygons with frictionless point contacts. Four independent region of contact can be obtained using this approach. Grasp with form-closure can be obtained by placing one frictionless contact inside each of the four regions. Approach generalized for polyhedra is outlined but no exact procedure or example is given in the paper. The procedure would be very complex and time consuming. Markenscoff et al. [2] demonstrated how a grasp with frictionless point contacts may be designed for objects without rotational symmetry. The resulted grasp is just one of the possible options and there is no significant reason for selecting that particular grasp in application. The main objective of the study is to demonstrate that seven frictionless point contacts were enough to grasp a 3D object. In a later study [4] by two of the authors of [3], an approach was presented for constructing a grasp with frictionless contacts to balance forces acted through the centre of mass of a polygon. The problem was simplified significantly by considering only a planar system. A more recent paper by Xiong et al. [5] gives methods to analyze grasps with frictionless contacts based on geometric approaches. Discussion on grasp construction is also provided but only brief outlines are presented. Trinkle et al. [6] also studied on the grasping of polygons with frictionless contacts. Enveloping grasping was studied so that in addition to point contacts, edge contacts were involved. In Enveloping grasping, the ability of the gripper to grasp the object depends on the relative size of the object and the gripper, and it is simple to grasp any object with appropriate size. It is possible to provide the same type of grasp for 3D object. The main problems of such kind of grasp is that the object and the gripper should have comparable size and the object is totally enclosed by the gripper. Therefore, Enveloping grasping is not suitable for grasping of large objects.

Nevertheless, among the approaches for constructing grasps for 3D objects with frictionless contacts, none of them can be applied to design grasps for manipulating large objects with several manipulators. In order to design such a grasp, the distances between the contacts must be taken into consideration. Thus, such an approach is presented for polyhedra in Chapter 2. Using the proposed approach, grasps, which can be implemented using two grippers with relative small size as compared with the grasped object, are constructed for several types of polyhedra. The grasps obtained may be applied to grasp objects with different sizes but certain common geometric properties. Approaches are also suggested to test if the grasp can be maintained after small displacements of the contacts. Moreover, a method is proposed to determine the position and orientation of polyhedra with three pairs of parallel surfaces from the contact location when the polyhedra are grasped with form-closure using only the three pairs of parallel surfaces.

An appropriate dynamic model must be available before any analysis can be carried out for the system. The study of multi-arm system has drawn many attentions of many researchers. Many dynamic models are proposed in different studies on multi-arm systems. Among these models, most assumed that each end effector grasps the object rigidly. With this assumption, the relative positions and orientations between the end effectors are constant. The relative velocities and accelerations as well as the internal forces and moments acted on the object are also easier to be determined. The problem may be simplified while most characteristics of the closed-chain systems is retained. This explains why most early studies and studies on control of multi-arm systems [7-14] apply this assumption. References [15] and [16] do not made this assumption exactly. In these two papers, each arm does not grasp the object rigidly but the arms together grasp the object rigidly such that the arms cannot move relative to each other.

In some other discussions, the arms hold the object rigidly but there are relative motion among the arms. In these cases, the objects under consideration are not rigid objects. Zheng [17] studied the assemble operation using two manipulators. In the study, the dynamic models for system with two robot arms assembling two parts were presented for different stages of the operation. Dellinger and Anderson [18] proposed a modelling method of dual-arm systems grasping a non-rigid object which contained a single joint with one or more degrees of freedom. The method is based on variational approach.

There are also some studies which consider certain connection between the arms and the object with certain degree(s) of freedom. Tarn et al. [19] and Yun and Kumar [20] assumed that the object can move relative to the grippers as if the object was connected to the arms with revolute joints. [19] is one of the early attempts in the study of two cooperating robot arms. With such a model for the connections, the system was regarded as a single-loop closed chain in the analysis. [20] is a modified version of [19] with some simulation results. Internal forces was also considered in this study. The situation where only one contact is made by each arm on the workpiece has drawn attentions from many researchers. Several studies on multiple arm systems with rolling contacts were carried out by the same group of researches [21-23]. These studies also aimed at the manipulation of large objects but in their system, there was only one rolling point contact between each arm and the object. Another study [24] by the same group included sliding contact in addition to rolling contact. Other studies [1, 25-28], which consider different types of single point contact between the manipulated object and the manipulators, are available. Among these, [1] and [28] are studies on robot hands which are similar to the case that a multi-arm system manipulates objects with only one point contact between each arm and the workpiece.

Finally, there are studies [29-31] trying to model the joints between the object and the arms in a more general way. However, they only discussed the most commonly used joints. No general method was proposed to determine the matrices required in the modelling of joints. These matrices include the matrices describing the contact forces, the matrices describing the relative motions, and also the time derivatives of these matrices. All these quantities are necessary in the modelling of the multi-arm system. In order to handle the more complicated joints between the arms and object, a method to model all types of joints and the general approaches to compute the required matrices are proposed. A general method is also proposed to compute the value of the corresponding constraint function for the coordinates of the connected bodies and their time derivatives.

To analyze the multi-arm system, computer simulation of the system may be helpful. In fact, real-time simulation may also be applied in some advanced control methods [32]. For this purpose, the solution of the forward dynamic problem must be obtained first. Thus, the forward dynamic problem is also studied. There are lots of references on simulation of multi-body systems [32-34]. Many different approaches were summarized in [32]. Rodriguez et al. proposed a spatial operator algebra [35] such that recursive algorithm can be directly obtained from the model derived; and discussed its application in simulation of multi-arm systems handling one single object [31]. Featherstone [36] formulated the forward dynamic problem of robot arm forming kinematic loops using a spatial notation. With the notation, he developed efficient algorithms to compute the terms in the dynamic model of the system. The forward dynamic model was obtained by combining the dynamic equations for each arm and the kinematic constraint equations due to the connections between the arms and the object. The resulted system of equations is in the same form as those considered in the literatures of simulation of multi-body systems [32-34]. Oh and Orin [37] also extended simulation method for single arm system in the same direction for multi-arm system. Lilly and Orin [30,38] proposed a solution for the simulation of multi-arm systems which have an object connected to several kinematic chains in parallel. This approach is easy to be understood and implemented using parallel processors. (A thorough discussion on simulation approaches for multi-arm system and comparisons with this approach can be found in these references.) Based on the same technique, McMillan et al. [39] considered the simulation when the system lose some degrees of freedom because of singularity of the arms.

Modifying the approaches proposed by Lilly and Orin [38] and McMillan et al. [39], a more suitable algorithm is derived for the system considered in this study. With the modifications, the efficiency of the approaches are retained while several advantages are introduced. The modified approach is also more general and can be used for a larger class of systems.

To summarize, there are three main results in this study. First, the grasping of polyhedra using frictionless point contacts are studied. Approaches are derived for designing and analyzing grasps for multi-arm systems. Secondly, a dynamic model is developed for multi-arm systems handling one single object based on a general joint model. General approaches for computing the corresponding matrices required in the dynamic model and constraint functions can be obtained using this model. Finally, an efficient algorithm is proposed for solving the forward dynamic problem of the multi-arm system.

The arrangement of this thesis is also divided into three parts according to the three main topics. In the next chapter, the grasping of polyhedra using frictionless point contacts is discussed. The dynamic model of multi-arm systems handling one single object and the general joint model are presented in Chapter 3. Chapter 4 presents the approaches for solving the forward dynamic problem. Finally, some possible future works and the conclusion of this study are presented in the last chapter.

CHAPTER TWO

FORM-CLOSURE GRASPING

A grasp is called form-closure grasp [3,5] if it can constrain the object, or provide reaction forces and moments in all directions to the object, irrespective of the magnitudes of the contact forces. To obtain form-closure grasp, all contacts must be considered as frictionless because frictional forces and moments depend on the magnitudes of the corresponding normal forces. Thus, form-closure grasp has the property that it just depends on the geometry of the object but not on the friction between the gripper and the object. The presence of friction would only result in a better grasp of the object [2].

This chapter discusses the design and analysis of form-closure grasp. In the first section, algebraic argument for the minimum number of point contacts required for form-closure is given. Approaches for constructing a form-closure grasp are presented in the second section. The concept of configuration stability of a grasp is introduced in the third section in which methods for testing the property for form-closure grasps are also suggested. In the last section, an approach is proposed to determine the object frame from the contact points of a form-closure grasp. This approach is suitable for the case where the contact points are on three pairs of parallel surfaces only.

2.1 Condition for Form-closure Grasp

It is well-known that for an n-dimensional vector space, at least n+1 uni-sense vectors are required to span the whole vector space with non-negative coefficients [3,5]. A set of N n-dimensional vectors which spans the whole n-dimensional vector space with non-negative coefficients must satisfy the following condition

$$-(\alpha_1 x_1 + \alpha_2 x_2 + \cdots + \alpha_n x_n) = \beta_1 x_{n+1} + \beta_2 x_{n+2} + \cdots + \beta_{N-n} x_N \tag{2.1}$$

where $\alpha_i > 0$

$\beta_i \geq 0$ and not all equal to zero

$\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n$ are n independent vectors

Proof 2.1

Let **A** be a set of n-dimension vectors, which can span the whole n-dimensional vector space with non-negative coefficients. It is obvious that **A** should also be able to span the whole vector space with real coefficients so it must contain at least one basis of the vector space. In other words, it should be able to select n independent vectors from **A**. Considering that any strictly negative combination of $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n$ is a component of the vector space, it can be concluded that Equation 2.1 is a necessary condition for **A**. ■

Equation 2.1 is also a sufficient condition. A set of $\boldsymbol{N}$ n-dimensional vectors can span the n-dimensional vector space with non-negative coefficients if they satisfy Equation 2.1.

Proof 2.2

Consider the following non-negative combination of the $\boldsymbol{N}$ vectors:

$$\gamma_1 x_1 + \gamma_2 x_2 + \cdots + \gamma_n x_n + \gamma_{n+1}(\beta_1 x_{n+1} + \beta_2 x_{n+2} + \cdots + \beta_{N-n} x_N)$$

where $\gamma_i > 0$

Using Equation 2.1, it can be expressed as:

$$(\gamma_1 - \gamma_{n+1}\alpha_1)x_1 + (\gamma_2 - \gamma_{n+1}\alpha_2)x_2 + \cdots + (\gamma_n - \gamma_{n+1})x_n$$

As γ_i - $\gamma_{n+1}\alpha_i$ (for $\boldsymbol{i}$ = 1, ..., n) can be any real number and $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots \boldsymbol{x}_n$ are linearly independent, non-negative combination of the $\boldsymbol{N}$ vectors satisfying Equation 2.1 can be any vector of the n-dimensional vector space. ■

In order to grasp an object, it should be possible to apply reaction forces and moments in all directions. In other words, the reaction wrenches applied on the object must be able to span the six-dimensional wrench space. As each frictionless point contact can only apply a uni-directional wrench on the object being grasped, at least seven point contacts are required in a form-closure grasp.

This is a well-known result. Markenscoff et al. [2] even proved that all three-dimensional objects without rotational symmetry can be grasped with seven point contact with form-closure. Although this result is not new and the proof is similar to some previous work [3,5], they are not exactly the same. First, the condition given by Equation 2.1 is a better condition than that given by Nguyen [3] or Xiong et al. [5]. The conditions used in these previous studies only consider n+1 vectors each time so some marginal cases are neglected. Consider a planar form-closure grasp of a rectangle using five or six point contacts as shown in Figure 2.1. In these cases, the rectangle is form-closure grasped but the wrenches provided by the contacts do not satisfy the conditions given by the previous studies. However, these two cases are included when the condition given by Equation 2.1 is used. Thus, Equation 2.1 is a more complete condition for form-closure. Secondly, this condition can be applied easily to design form-closure grasp for polyhedral objects even when more than seven point contacts are used. This is because this condition takes all contacts into consideration.

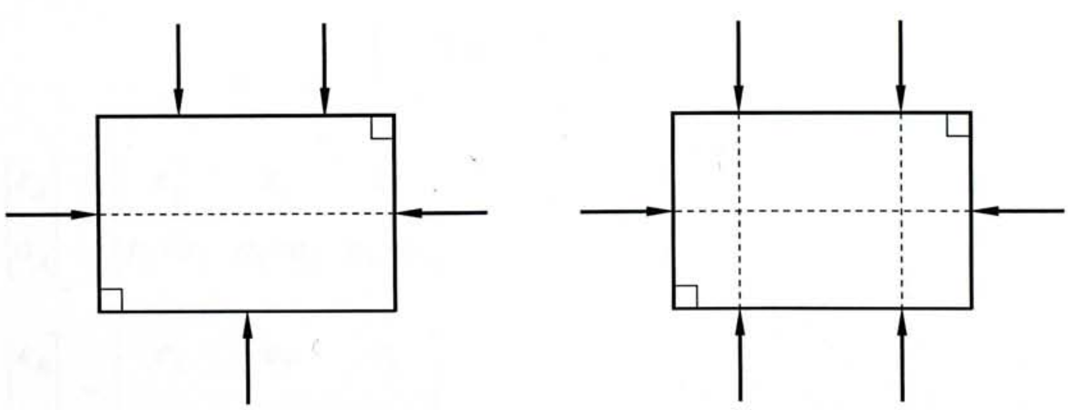

Figure 2.1 Critical cases of planar form-closure grasp

2.2 Construction of Form-closure Grasp

The construction of form-closure grasp can be broken down into two steps according to the condition for form-closure grasp. The condition requires that there should be six contacts providing six independent wrenches and all the wrenches provided by the contact must satisfy Equation 2.1. Thus, the procedures for constructing form-closure grasps can be taken as

1. Select six contacts[1] which can provide six linearly independent wrenches;
2. Select the remaining contact(s) such that the corresponding wrench(es) satisfy Equation 2.1 together with those corresponding to the six contacts selected before.

The construction of form-closure grasp based on these two steps is discussed for polyhedral objects only. For polyhedra, the contact normal depends on the surface normals. Therefore, the construction is simplified as compared with objects having curved surfaces.

2.2.1 Selection of the six independent contacts

The inverse of the matrix whose columns correspond to six zero-pitch wrenches provided by six frictionless point contacts can be expressed as:

$$\begin{bmatrix} (I_3+T_1T_2)e_A^{-1} & T_1a^{-1} \\ T_2e_A^{-1} & a^{-1} \end{bmatrix} \tag{2.2}$$

where $\begin{bmatrix} e_A \\ a_A \end{bmatrix} = \begin{bmatrix} e_1 & e_2 & e_3 \\ p_1\times e_1 & p_2\times e_2 & p_3\times e_3 \end{bmatrix}$

$\begin{bmatrix} e_B \\ a_B \end{bmatrix} = \begin{bmatrix} e_4 & e_5 & e_6 \\ p_4\times e_4 & p_5\times e_5 & p_6\times e_6 \end{bmatrix}$

---

[1] These contacts are called independent contacts for convenience in the discussions that follow.

$p_1, p_2, \cdots, p_6$ are the position vectors of the six independent contacts

$e_1, e_2, \cdots, e_6$ are the contact normals[1] of the corresponding contact points

e_1, e_2, e_3 are linearly independent

$T_1 = -e_A^{-1} e_B$

$a = a_A T_1 + a_B$

$T_2 = -a^{-1} a_A$

I_3 is the 3×3 identity matrix

(The proof for this is given in Appendix A.) Therefore, the basic requirement for the six contacts to be independent is:

1. e_A exists;
2. a, or $a_A T_1 + a_B$, is nonsingular

Using these rules, the contact surfaces and the contact locations can be considered separately in selecting the six independent contacts. The first condition requires that three of the contact normals must be linearly independent. For the contact locations, it does not just require three of the six $p_i \times e_i$ to be linearly independent. The $p_i \times e_i$ must also satisfy the second condition given above. Some necessary conditions for the contact locations can be listed as below:

1. No more than 3 contact normals intersect at one point;
2. No more than 3 contacts normals are parallel;
3. No more than 3 contact normals lie on the same plane;
4. If 3 contact normals intersect at one point,
   (i) the remaining contact normals must not be all parallel;
   (ii) the rest contact normals must not intersect at one point; or
   (iii) the intersection must not lie on the planes formed by the remaining contact

[1] Contact normal is represented by a 3×1 unit vector giving the direction pointing towards the corresponding surface and means the fixed vector passing through the contact point.

normals;

5. If 3 contact normals are parallel, the other 3 contact normals must not intersect at one point;
6. If 3 contact normals lie on the same plane, intersection(s) of the remaining contact normals must not lie on the same plane.

The proofs of these conditions can be found in Appendix A. These give some general rules for selecting the independent contact but are not sufficient conditions. They just narrowed the selection. It is very difficult to specify all the rules for selecting independent contacts in terms of the geometry of the contacts because it depends on many variables. Nevertheless, for some special selections, the problem can be simplified. In the following, three such special cases are discussed.

Case I Symmetric contact points

For objects with three pairs of parallel plane surfaces, we may select one contact on each of the six surfaces. In this case, $\boldsymbol{T_1}$ is equal to $\boldsymbol{I_3}$ and, therefore, $\boldsymbol{a}$ becomes $\boldsymbol{a_A}+\boldsymbol{a_B}$. As the value of $\boldsymbol{p_i}\times\boldsymbol{e_i}$ do not depend on the values of $\boldsymbol{p_i}$ along the direction of $\boldsymbol{e_i}$, it is possible to select the contact locations such that $\boldsymbol{a_B}$ is equal to $\boldsymbol{a_A}$. The problem is then reduced to obtaining three linearly independent $\boldsymbol{p_i}\times\boldsymbol{e_i}$ from three contacts.

The sufficient conditions for three contacts on plane surfaces with linearly independent $\boldsymbol{p_i}\times\boldsymbol{e_i}$ can be listed as follows:

1. The contacts are not coincide with the orthogonal projection[1] of the origin on the corresponding surfaces;
2. For any two contacts on two non-parallel surfaces, the two lines joining the contact points and the projection of the origins on the corresponding surfaces must not both perpendicular to the intersecting line of the surfaces.

[1] This point is called projected origin for simplicity in the later discussions.

3. If two contacts lie on the same plane, the projected origin must not lie on the line joining the contact points.
4. The contact normals are not intersect at one point.
5. The contacts are not all lie on parallel surfaces.
6. For each contact, the line joining the contact and the projected origin must not perpendicular to the intersecting line of the plane on which the contact lies and the surface which parallel to both of the other two $\boldsymbol{p}_i \times \boldsymbol{e}_i$'s.

We can select the three independent contacts in three steps. The first contact can be selected according to Rule (1). The second contact is then selected according to Rules (1) to (3). Finally, the last contact is selected so that it satisfies all the six rules.

Case II   Three contact normals intersect at one point

If we select three contacts whose contact normals are linearly independent and intersect at one point, all elements of $\boldsymbol{a}_A$ become zero when the intersection is taken as the origin. The remaining contacts can then be selected with more freedom such that it only requires the three corresponding $\boldsymbol{p}_i \times \boldsymbol{e}_i$ to be linearly independent. The contacts may then be selected according to the rules given above. In fact, the three intersecting normals can lie on curved surfaces.

Case III   Two contact normals intersect and three contact normals are parallel

We may select two contacts whose contact normals, $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$, are linearly independent and intersect with each other. This makes two columns of $\boldsymbol{a}_A$ become zero if the intersection is chosen as the origin. Furthermore, if another three contacts have parallel contact normals but non-parallel and non-zero $\boldsymbol{p}_i \times \boldsymbol{e}_i$, the last contact can be selected according to the surface normals only. First, the plane of the last contact must have a normal which is not parallel to the three parallel contact normals. In addition, either the normal of the plane or the three parallel contact normals must be linearly independent with $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$. The last contact may then be any point on the plane except points on a line passing through the projected origin. This line is perpendicular to the intersecting line of the plane of the contact and a plane with a normal parallel to the three parallel contact normals.

2.2.2 Determination of the region for the seventh contact

In this section, form-closure grasp using seven point contacts is discussed. In particular, approaches for the selection of the seventh contact on a flat surface for a form-closure grasp are described.

2.2.2.1 General Approach

Consider the case when six independent point contacts are already chosen and it is decided to grasp the object with one more contact on a flat surface. The inward normal of this flat surface must satisfy Equation 2.1 with the other six contact normals. The seventh contact must be selected such that the seven wrenches which can be applied through the contacts satisfy Equation 2.1.

Take $\boldsymbol{p}_1, \boldsymbol{p}_2, \cdots, \boldsymbol{p}_6$ as the position vectors of the six independent contacts

$\boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_6$ as the contact normals of the corresponding contact points

$\boldsymbol{e}_7$ as the inward normal of the flat surface for the seventh contact

$\boldsymbol{p}_7$ as the position vector of the seventh contact

Using Equation 2.1, we have

$$\begin{bmatrix} \boldsymbol{e}_1 & \boldsymbol{e}_2 & \cdots & \boldsymbol{e}_6 \\ \boldsymbol{p}_1 \times \boldsymbol{e}_1 & \boldsymbol{p}_2 \times \boldsymbol{e}_2 & \cdots & \boldsymbol{p}_6 \times \boldsymbol{e}_6 \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{e}_7 \\ \boldsymbol{p}_7 \times \boldsymbol{e}_7 \end{bmatrix} = \begin{bmatrix} \alpha_1/\beta_1 \\ \alpha_2/\beta_1 \\ \vdots \\ \alpha_6/\beta_1 \end{bmatrix}$$

where $\alpha_i > 0$

$\beta_1 > 0$ as N is equal to 7 in this case

Moreover, if $\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3$ are selected such that they are linearly independent, the L.H.S. can be expressed as

$$\begin{bmatrix} (I_3+T_1T_2)e_A^{-1} & T_1a^{-1} \\ T_2e_A^{-1} & a^{-1} \end{bmatrix} \begin{bmatrix} I_3 & 0 \\ 0 & -e_7\times \end{bmatrix} \begin{bmatrix} e_7 \\ p_7 \end{bmatrix} \quad or \quad \begin{bmatrix} (I_3+T_1T_2)e_A^{-1} & T_1a^{-1}(e_7\times)^T \\ T_2e_A^{-1} & a^{-1}(e_7\times)^T \end{bmatrix} \begin{bmatrix} e_7 \\ p_7 \end{bmatrix}$$

where $\begin{bmatrix} e_A \\ a_A \end{bmatrix} = \begin{bmatrix} e_1 & e_2 & e_3 \\ p_1\times e_1 & p_2\times e_2 & p_3\times e_3 \end{bmatrix}$

$$\begin{bmatrix} e_B \\ a_B \end{bmatrix} = \begin{bmatrix} e_4 & e_5 & e_6 \\ p_4\times e_4 & p_5\times e_5 & p_6\times e_6 \end{bmatrix}$$

$$T_1 = -e_A^{-1}e_B$$

$$a = a_AT_1 + a_B$$

$$T_2 = -a^{-1}a_A$$

Using the fact that α_i/β_1 are greater than zero, we can obtain the condition for the seventh contact location so that the grasp is form-closure:

$$\begin{aligned} [(I_3+T_1T_2)e_A^{-1}]e_7 + [T_1a^{-1}(e_7\times)^T]p_7 &< \mathbf{0}_{3\times 1} \\ [T_2e_A^{-1}]e_7 + [a^{-1}(e_7\times)^T]p_7 &< \mathbf{0}_{3\times 1} \end{aligned} \tag{2.3}$$

where $\mathbf{0}_{3\times 1}$ is a 3×1 vector with all elements equal to zero

One more constraint is that the contact should lie on the selected surface. Without loss of generality, we may select a reference frame which has its z-axis parallel to $-e_7$. With this reference frame, the region for the seventh contact may be obtained by substituting zero for the z-component of p_7 in Equation 2.3. The resulted boundary are expressed in the frame with its origin equal to the intersection of the z-axis and the selected surface. The following example demonstrates the use of Equation 2.3.

Example 2.1

Consider a rectangular block and take the reference frame as a frame with the centre of the block as origin and the axes are parallel to the normals of three adjacent surfaces. Three contacts are selected as shown in Figure 2.2. The other three contacts are selected such that they are at the corresponding positions at the opposite corner, i.e.

$$p_4 = -p_1 \quad p_5 = -p_2 \quad p_6 = -p_3$$

This selection follows case I in Section 2.2.1. If the last contact is to be placed on the top surface, the inequalities given in Equation 2.3 can be reduced to

$$\frac{1}{\det(a_A)}\begin{bmatrix} f & -e \\ ad & -bc \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} < \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

where x and y are the x- and y- coordinates of the contact

$\det(a_A)$ is the determinant of a_A and equals to $(ade - bcf)$

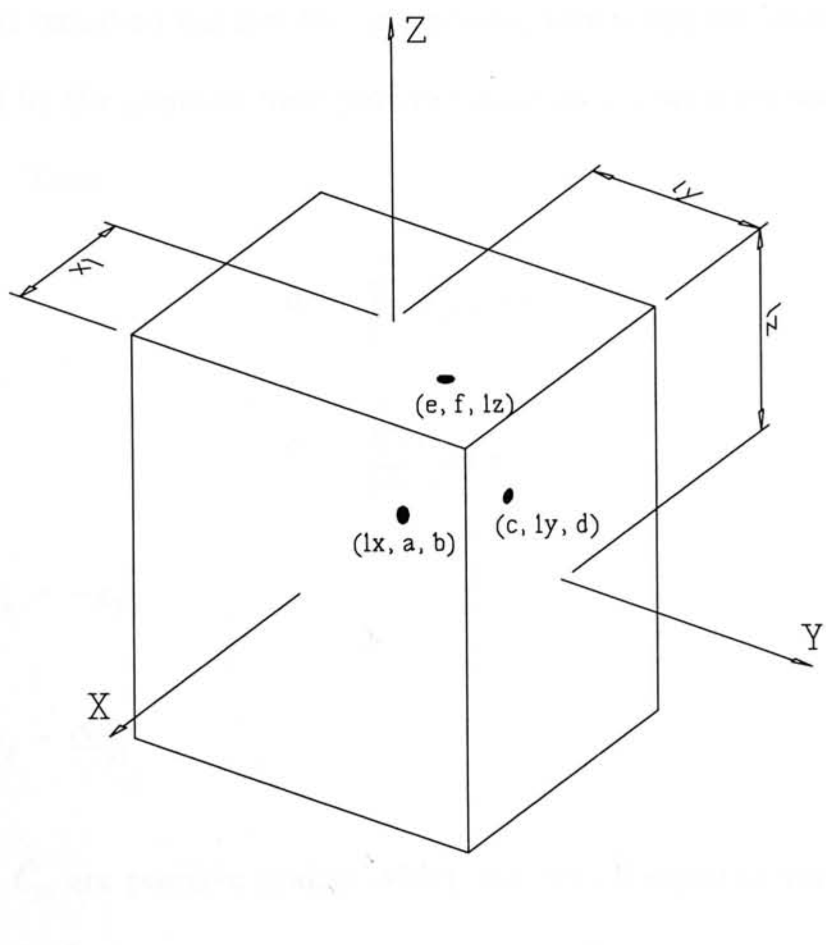

Figure 2.2 Contact point locations in Example 2.1

The region of the possible location of the seventh contact is, therefore, bounded by the following two lines on the surface:

$$fx - ey = 0$$

$$(ad)x - (bc)y = -\det(a_A) \qquad \blacksquare$$

This approach provides a convenient and manageable method to invert the 6×6 matrix formed by the six independent wrenches. It is, however, difficult to apply this approach to determine the region for the last contact when more than seven contacts are used. All sets of six independent contacts must be considered separately. However, this approach can easily be adapted for use in cases where the seventh contact lies on a curved surface[1]. In such cases, the constant value of $\boldsymbol{e}_7$ is replaced with a function of the contact point $\boldsymbol{p}_7$.

2.2.2.2 Approach for Polyhedral objects

This approach is based on the fact that the normal forces and the moments about the origin provided by the contacts must satisfy Equation 2.1 with the same coefficients (i.e. α_i and β_i). Take

$$\boldsymbol{b}_i = \sum_{k=1}^{6} \tilde{\boldsymbol{B}}_{ik} \boldsymbol{p}_k \times \boldsymbol{e}_k$$

$$\boldsymbol{c}_j = \sum_{k=1}^{6} \tilde{\boldsymbol{C}}_{jk} \boldsymbol{p}_k \times \boldsymbol{e}_k$$

where $\sum_{k=1}^{6} \tilde{\boldsymbol{B}}_{ik} \boldsymbol{e}_k = -\boldsymbol{e}_7$

$$\sum_{k=1}^{6} \tilde{\boldsymbol{C}}_{jk} \boldsymbol{e}_k = 0_{3\times1}$$

$\tilde{\boldsymbol{B}}_{ik}$ and $\tilde{\boldsymbol{C}}_{jk}$ are positive scalars which are not all equal to zero for any $\boldsymbol{i}$ and $\boldsymbol{j}$ respectively

---

1 In [3], Nguyen pointed out that the region for a contact in a form-closure grasp must have the following properties:
(1) be either flat or spherical; and
(2) have a convex boundary
However, he used a condition which is not always true in his proof. Actually, it is possible to have contact regions without the above properties. See Appendix A for more details.

$p_7 \times e_7$ must be a strictly negative combination of $p_i \times e_i$ (for i = 1, 2, ..., 6), such that

$$p_7 \times e_7 = \sum_{i=1}^{6} \tilde{D}_i \, p_i \times e_i$$

where $\tilde{D}_i$ are strictly negative scalars

The coefficients $\tilde{D}_i$ must also be used to combine e_i to form e_7, i.e.

$$e_7 = \sum_{i=1}^{6} \tilde{D}_i \, e_i$$

Therefore, $p_7 \times e_7$ is in the following form:

$$\sum_{i=1}^{n_b} B_i b_i + \sum_{j=1}^{n_c} C_j c_j$$

where n_b and n_c are number of b_i and c_i respectively

B_i are negative scalars such that $\sum_{i=1}^{n_b} B_i = -1$

C_j are negative scalars

As $p_7 \times e_7$ is perpendicular to e_7, the coefficients B_i and C_i must satisfy the following condition:

$$e_7^T \left(\sum_{i=1}^{n_b} B_i b_i + \sum_{j=1}^{n_c} C_j c_j \right) = 0 \qquad (2.4)$$

From Equation 2.4, all possible values of $p_7 \times e_7$ can be determined. Using the relationship between $p_i \times e_i$ and p_i on a plane, the possible values of p_7 can then be obtained. The relationship is described in Appendix A. Figure 2.3 shows this relationship.

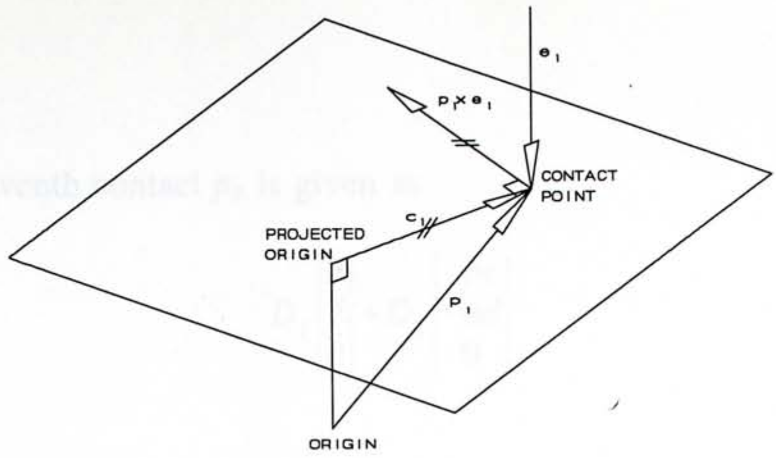

Figure 2.3 Relationship between contact point on a flat surface and the corresponding moment about the origin

The following example shows how the condition given by Equation 2.4 can be used to determine the region for the seventh contact.

Example 2.2

Consider the situation in Example 2.1 again. In this case, there is only one linearly independent $\boldsymbol{b}_i$:

$$\boldsymbol{b}_1 = \begin{bmatrix} -f \\ e \\ 0 \end{bmatrix}$$

and there are three linearly independent $\boldsymbol{c}_i$:

$$c_1 = \begin{bmatrix} 0 \\ -b \\ a \end{bmatrix}, \; c_2 = \begin{bmatrix} d \\ 0 \\ -c \end{bmatrix}, \; c_3 = \begin{bmatrix} -f \\ e \\ 0 \end{bmatrix}$$

Applying Equation 2.4, we have

$$-C_1 a + C_2 c = 0$$

$$C_2 = \frac{a}{c} C_1$$

$\boldsymbol{p}_7 \times \boldsymbol{e}_7$ is in the following form:

$$D_1 \begin{bmatrix} -f \\ e \\ 0 \end{bmatrix} + D_2 \begin{bmatrix} ad \\ -bc \\ 0 \end{bmatrix}$$

where $D_1 < -1$

$D_2 < 0$

Therefore the seventh contact p_7 is given as

$$D_1 \begin{bmatrix} e \\ f \\ 0 \end{bmatrix} + D_2 \begin{bmatrix} -bc \\ -ad \\ 0 \end{bmatrix}$$

This is equivalent to the result obtained in Example 2.1. ■

It is obvious that Equation 2.4 does not always give some valid condition for the B_i's and C_i's. An invalid condition implies that no contact on the selected surface can provide a form-closure grasp with the six independent contacts.

This is a more direct approach. The relation of the contact location of the seventh contact and the α_i's in Equation 2.1 is clear. It is more appropriate for designing grasps although it is only eligible for objects with plane surfaces. Moreover, this approach is much easier to be modified for cases where more than seven contacts are used. Unlike the previous one, there is no limit on the number of contacts in the condition applied, i.e. Equation 2.4. To be used in such cases, we just need to determine the extra b_i and c_i.

2.2.3 Design of form-closure grasps for large objects

2.2.3.1 Rectangular blocks

It is possible to design two fixed grippers to provide form-closure grasps for rectangular blocks with variable sizes. Consider the rectangular block in Example 2.1 again. If contact is on the line $y = f$, minimum distance (D_x) between the contact on the top surface and the seventh contact is:

$$\left| e - \frac{-\det(a_A) + fbc}{ad} \right| = \left| \frac{2\det(a_A)}{ad} \right|$$

Taking $a = l_y - A$ $b = l_z - A$

$$c = l_x - A \quad d = l_z - A$$

$$e = l_x - E \quad f = l_y - A$$

we have $D_x = |2(A - E)|$

Figure 2.4 shows the region for the last contact and the corresponding lines when $A > E$. As A and E are distances from the edges, two sets of point contacts, each with fixed configuration relative to a corner, can grasp a rectangular block with form-closure. There is no limit on the size of the block, except that the size of the block cannot be smaller than a certain limit such that a, b, c, d, e and f are all positive.

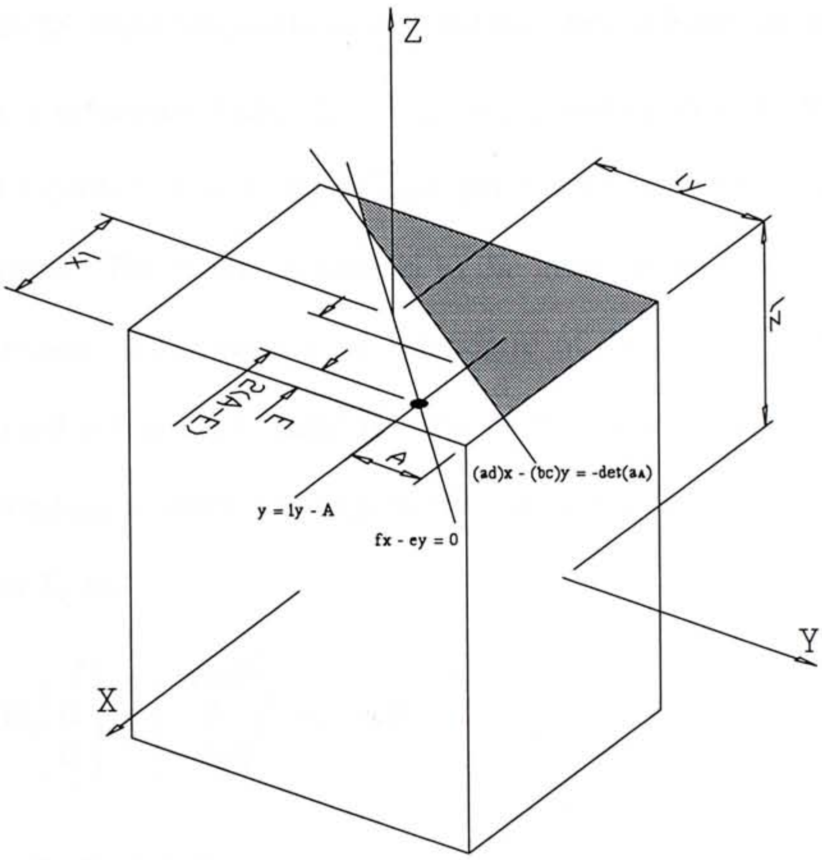

Figure 2.4 The possible region for the seventh contact location

2.2.3.2 *Object with three pairs of parallel planes only*

Similar to rectangular blocks, two fixed grippers can be designed to provide form-closure grasps for this type of object having a size within a certain range.

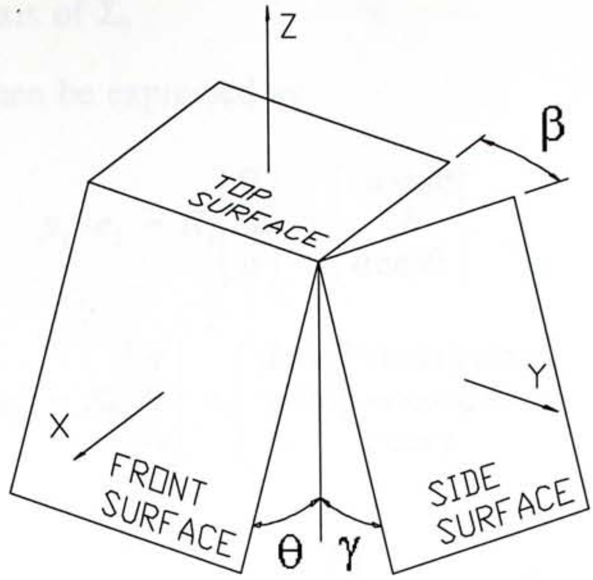

Figure 2.5 Three non-parallel surfaces and the reference frame considered

Consider a reference frame Σ_0. Figure 2.5 shows this reference frame and three surfaces at a corner. The z-axis of Σ_0 is perpendicular to and pointing outwards from the top surface. The y-axis is parallel to the intersecting line of the top surface and the front surface. The origin is at the centre of the object. This is a general approach to defined a reference frame relative to three non-parallel planes. Contact normals corresponding to three contacts on the surfaces at a corner can be expressed with reference to Σ_0 as

$$\boldsymbol{e}_1 = \boldsymbol{R}_1\begin{bmatrix}-1\\0\\0\end{bmatrix} = \begin{bmatrix}-\cos\theta\\0\\-\sin\theta\end{bmatrix},\ \boldsymbol{e}_2 = \boldsymbol{R}_2\begin{bmatrix}0\\-1\\0\end{bmatrix} = \begin{bmatrix}-\sin\beta\cos\gamma\\-\cos\beta\cos\gamma\\-\sin\gamma\end{bmatrix},\ \boldsymbol{e}_3 = \begin{bmatrix}0\\0\\-1\end{bmatrix}$$

where $\boldsymbol{R}_1 = \begin{bmatrix}\cos\theta & 0 & -\sin\theta\\0 & 1 & 0\\\sin\theta & 0 & \cos\theta\end{bmatrix}$ rotates the normal of the front surface parallel to the x-axis of Σ_0

$\boldsymbol{R}_2 = \begin{bmatrix}\cos\beta & \sin\beta\cos\gamma & -\sin\beta\sin\gamma\\-\sin\beta & \cos\beta\cos\gamma & -\cos\beta\sin\gamma\\0 & \sin\gamma & \cos\gamma\end{bmatrix}$ rotates the normal of the side surface

e_3 parallel to the y-axis of Σ_0

The values of $p_i \times e_i$ can then be expressed as

$$p_1 \times e_1 = R_1 \begin{bmatrix} 0 \\ -b \\ a \end{bmatrix} = \begin{bmatrix} -a\sin\theta \\ -b \\ a\cos\theta \end{bmatrix},$$

$$p_2 \times e_2 = R_2 \begin{bmatrix} d \\ 0 \\ -c \end{bmatrix} = \begin{bmatrix} d\cos\beta + c\sin\beta\sin\gamma \\ -d\sin\beta + c\cos\beta\sin\gamma \\ -c\cos\gamma \end{bmatrix},$$

$$p_3 \times e_3 = \begin{bmatrix} -f \\ e \\ 0 \end{bmatrix}$$

where $\begin{bmatrix} l_x \\ a \\ b \end{bmatrix}$, $\begin{bmatrix} c \\ l_y \\ d \end{bmatrix}$, and $\begin{bmatrix} e \\ f \\ l_z \end{bmatrix}$ are locations of the contact on the front surface, the side surface and the top surface respectively

l_x, l_y, and l_z are perpendicular distances from the origin to the corresponding surfaces

If the other three contacts are at the corresponding positions of the opposite corner such that

$$p_4 = -p_1 \quad p_5 = -p_2 \quad p_6 = -p_3$$

The approach for polyhedral objects can be applied to obtain the region for the seventh contact. In this case, we have one linearly independent b_i:

$$b_1 = \begin{bmatrix} -f \\ e \\ 0 \end{bmatrix}$$

and three linearly independent c_i:

$$c_1 = \begin{bmatrix} -a\sin\theta \\ -b \\ a\cos\theta \end{bmatrix}, \; c_2 = \begin{bmatrix} d\cos\beta + c\sin\beta\sin\gamma \\ -d\sin\beta + c\cos\beta\sin\gamma \\ -c\cos\gamma \end{bmatrix}, \; c_3 = \begin{bmatrix} -f \\ e \\ 0 \end{bmatrix}$$

Take e_7 equal to e_3 and apply Equation 2.3, we have

$$-C_1 a\cos\theta + C_2 c\cos\gamma = 0$$

$$C_2 = \frac{a\cos\theta}{c\cos\gamma} C_1$$

Therefore, $p_7 \times e_7$ is in the form:

$$D_1 \begin{bmatrix} -f \\ e \\ 0 \end{bmatrix} + D_2 \begin{bmatrix} -ac\cos\gamma\sin\theta + ad\cos\theta\cos\beta + ac\cos\theta\sin\beta\sin\gamma \\ -bc\cos\theta - ad\cos\theta\sin\beta + ac\cos\theta\cos\beta\sin\gamma \\ 0 \end{bmatrix}$$

where $D_1 < -1$

$D_2 < 0$

Then the required contact on the top surface must be in the form

$$D_1 \begin{bmatrix} e \\ f \\ 0 \end{bmatrix} + D_2 \begin{bmatrix} -bc\cos\theta - ad\cos\theta\sin\beta + ac\cos\theta\cos\beta\sin\gamma \\ ac\cos\gamma\sin\theta - ad\cos\theta\cos\beta - ac\cos\theta\sin\beta\sin\gamma \\ 0 \end{bmatrix} = D_1 \begin{bmatrix} e \\ f \\ 0 \end{bmatrix} + D_2 \begin{bmatrix} p_x \\ p_y \\ 0 \end{bmatrix}$$

The following example shows how this result can be applied to design grasp for this class of object.

Example 2.3

Consider the case when the object has a corner such that -90°< θ < 0°, -90°< β < 0°, -90°< γ < 0°. Take the coordinates of the corner as (L_x, L_y, L_z), which are all positive, and

$$\begin{aligned} a &= L_y - F & e &= L_x - E \\ b &= -(L_x - E)\sin\theta + (L_z - G_1)\cos\theta & f &= L_y - F \\ c &= (L_x - E)\cos\beta - (L_y - F)\sin\beta & g_1 &= L_z - G_1 \\ d &= -(L_x - E)\sin\beta\sin\gamma - (L_y - F)\cos\beta\sin\gamma + (L_z - G_2)\cos\gamma & g_2 &= L_z - G_2 \end{aligned}$$

where E, F, G_1 and G_2 are constants specifying the contact locations from the corner The values of E, F, G_1 and G_2 must be selected such that the corresponding contact locations are lie on the surfaces of the object. With such relations, p_y becomes:

$$ef\sin\theta\cos\beta\cos\gamma + f^2(\cos\theta\sin\gamma - \sin\theta\sin\beta\cos\gamma) - fg_2\cos\theta\cos\beta\cos\gamma$$

If the values of e, f, g_1 and g_2 are positive, the value of p_y is negative. This implies that the seventh contact can be selected such that the y-coordinate is f. In this case, the distance (D_x) between the seventh contact and the contact on the top surface must be greater than

$$\frac{e(g_1-g_2)\cos\theta\cos\beta\cos\gamma - f(g_1-g_2)\cos\theta\sin\beta\cos\gamma}{e\sin\theta\cos\beta\cos\gamma + f(\cos\theta\sin\gamma - \sin\theta\sin\beta\cos\gamma) - g_2\cos\theta\cos\beta\cos\gamma}$$

The absolute value of D_x decreases as g_2 (or L_z) increases and can be scaled by (g_1-g_2). Thus, this object can be grasp with form-closure using two groups of contacts with fixed configuration with respect to two opposite corners. The contacts may be provided using two fixed grippers. The object is specified by values of θ, β, γ, L_x and L_y. The value of L_z is required to be greater than a certain limit. ■

2.2.3.3 A class of polyhedron

Consider polyhedral objects which has two or more surfaces whose normals are negative combinations of the normals of three other surfaces. This class of polyhedral object includes all polyhedra with more than four surfaces.[1] For such kind of object, the following contact normals can be obtained:

e_1, e_2, and e_3 are linearly independent

e_4, e_5, e_6 and e_7 are negative combinations of e_1, e_2, and e_3

e_7 is equal to e_6

As e_4 and e_5 may be the same, only two different contact normals are required to be negative combinations of e_1, e_2, and e_3. If the contact normals e_1, e_2, and e_3 intersect at one point and the intersection is taken as the origin, there is only one linearly independent b_i:

$$b_1 = 0_{3\times 1}$$

and three linearly independent c_i:

[1] See Appendix A for a discussion on this class of polyhedra.

$$c_1 = p_4 \times e_4,\ c_2 = p_5 \times e_5,\ c_3 = p_6 \times e_6$$

Thus, results similar to those of the last section may be obtained. In this case, if the origin is the intersection of e_1, e_2, and e_3, and the axes of Σ_0 is selected according to e_4, e_5, and e_6 instead of e_1, e_2, and e_3 respectively, the expression for c_1, c_2 and c_3 is the same as before. As a result, a grasp may be designed as demonstrated in Example 2.3 if the angles between the surfaces corresponding to e_4, e_5, and e_6 satisfy the specified condition.

2.3 Configuration Stability of Form-closure Grasps

According to Montana [40], the stability of a grasp can be classified into two types, namely, spatial grasp stability and contact grasp stability. In this section, one more type of grasp stability is introduced and discussed. This type of stability is termed as configuration stability because it depends on the configuration of the contacts. The configuration stability is defined as the ability of a grasp to be maintained under disturbances on the contact point locations. This is an important quality measure of the grasp though it is neglected by most studies on optimal grasp.

The independent regions of contact[1] defined by Nguyen [3] can be used to measure the configuration stability. A grasp would have a better configuration stability if the distances of the contacts from the boundaries of the independent regions of contact is longer. For both polyhedral objects and polygons, it is not difficult to determine the independent regions of contact when frictional contacts are used in the grasp [3]. However, the problem is much more complicated when form-closure of 3D objects are considered as it involves at least 14 independent variables.

---

[1] In [3], independent regions of contact is defined as the regions within which the contacts can be located independently to provide force-closure grasp for the object. And this is completely different from those obtained in the previous section.

In this section, issues on how to test the configuration stability of form-closure grasps for polyhedral objects and to increase stability are discussed. Once again, the discussion is focused on grasps with seven point contacts.

As in the construction of a form-closure grasp, the analysis of configuration stability of such grasps can be divided into two parts. In the first part, the six independent contacts are considered. It is possible to determine a limit on the displacements of the contacts such that the contacts can be maintained as independent contacts. In the second part, by considering the region of the last contact, the grasp can be tested if it can be maintained after small displacements of the contacts. Although these approaches cannot be used to determine the exact displacements allowable for the contacts, they are useful in the analysis of the grasp and help to design better grasps.

2.3.1 Limit on the displacements of independent contacts

Consider six independent contacts. After a displacement of a contact, the corresponding value of $\boldsymbol{p}_i \times \boldsymbol{e}_i$ becomes

$$\boldsymbol{p}_i \times \boldsymbol{e}_i + \bar{\boldsymbol{p}}_i \times \boldsymbol{e}_i$$

where $\bar{\boldsymbol{p}}_i$ is the displacement

When all of the contacts displaced, the value of $\boldsymbol{a}$ becomes

$$(\boldsymbol{a}_A T_1 + \boldsymbol{a}_B) + (\bar{\boldsymbol{a}}_A T_1 + \bar{\boldsymbol{a}}_B)$$

where $\bar{\boldsymbol{a}}_A = [\bar{\boldsymbol{p}}_1 \times \boldsymbol{e}_1 \;\; \bar{\boldsymbol{p}}_2 \times \boldsymbol{e}_2 \;\; \bar{\boldsymbol{p}}_3 \times \boldsymbol{e}_3]$

$$\bar{\boldsymbol{a}}_B = [\bar{\boldsymbol{p}}_4 \times \boldsymbol{e}_4 \;\; \bar{\boldsymbol{p}}_5 \times \boldsymbol{e}_5 \;\; \bar{\boldsymbol{p}}_6 \times \boldsymbol{e}_6]$$

This shows that after the displacement, a 3×1 vector is added to each column of $\boldsymbol{a}$. The magnitude of these vectors depend on the amount of the displacements of the contacts. In terms of three dimensional geometry, the columns of the original $\boldsymbol{a}$ can be represented by three non-parallel vectors pointing from the origin because they are linearly independent. This is shown in Figure 2.6. If the vectors become linearly

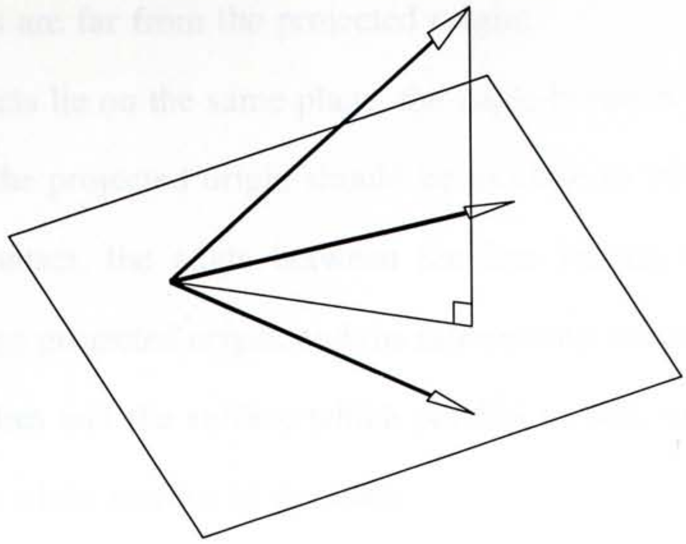

Figure 2.6 Three linearly independent 3D vectors

dependent, all of them must lie on the same plane or some of them become zero. Therefore, the minimum magnitudes of vectors required to be added to the vectors the columns of $\boldsymbol{a}$ depends on the angles between them and their magnitudes. The minimum magnitude required can be determined from the known values of the three non-parallel vectors. Thus, a maximum limit on the displacements of the contacts can be set. When the contacts move within this limit, $\boldsymbol{a}$ can be maintained nonsingular. Because polyhedral objects are considered, $\boldsymbol{e}_A$ will not change for any displacements of the contacts on the corresponding surfaces and nonsingularity of $\boldsymbol{a}$ implies the six contacts can provide six linearly independent contacts. Thus, the limit indicates the configuration stability. If the limit is small, the grasp must not have a good stability.

In order to increase the limit on allowable displacement, either the magnitudes of the vectors or the minimum angle between one of the three vectors and the plane formed by the other two may be increased. These properties depend on all six contact locations as well as the contact normals so it is difficult to give a general guideline to increase the values. However, for some special cases, as those discussed in Section 2.2.1, the rules are much simpler. If $\boldsymbol{a}$ is just depended on three contacts, the rules may be given as:

1. The contacts are far from the projected origin;
2. If two contacts lie on the same plane, the angle between the lines joining the contacts to the projected origin should be as close to 90° as possible;
3. For each contact, the angle between the line joining the contact and the corresponding projected origin, and the intersecting line of the plane on which the contact lies and the surface which parallel to both of the other two $\boldsymbol{p}_i \times \boldsymbol{e}_i$ should be as close to zero as possible.

The first rule ensures that the vectors are large in magnitude; and the second and third ensure that the angle between the vectors are as close to 90° as possible.

2.3.2 Change of the region for the last contact

The configuration stability can be reflected by the region for a contact obtained when all other contacts are fixed. This region can be determined for each contact using the approaches described in Section 2.2.2. However, this just considered the case when one contact is allowed to move at a time. The regions may change while other contacts move. Thus, an approach is proposed to test the change under small displacements of other contacts. The displacements may be considered as small if it is small as compared with the distances between the contacts and the boundaries of the corresponding region.

Applying the approaches described in Section 2.2.2.1, inequalities specifying the region for the last contact can be obtained. If one side of every inequality is made zero, the other sides of the inequalities can be used as measures to test if the last contact is inside the region. For each measure, we may then estimate the maximum change due to certain small finite displacements. This can be carried out using the partial differentiations of the measures and assuming the measures are linearized for small changes in the contact locations. If the estimated changes are all much smaller than the values of the corresponding measures computed from the contact locations of the grasp, we may conclude that the grasp can be maintained under the small displacements of the contacts. If the result is negative, it can be sure that the stability of the grasp is poor. In order to illustrate the procedure, an example is given below.

Example 2.4

From Example 2.1, two measure can be defined as

$$fx - ey$$
$$(ad)x - (bc)y + \det(ade - bcf)$$

where $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$, $\boldsymbol{d}$, $\boldsymbol{e}$ and $\boldsymbol{f}$ are different from the coordinates of the contacts in Example 2.1 and are the sums of the corresponding coordinates of the contacts on parallel surfaces

As a demonstration, only the first one is considered. This one depends only on the coordinates of the two contacts on the top and bottom surfaces. The estimated maximum change of this measure is then obtained by carrying out partial differentiation:

$$|y\,\delta e_t| + |x\delta f_t| + |y\delta e_b| + |x\delta f_b|$$

where δe_t and δf_t are changes of the coordinates of the contact on the top surface

δe_b and δf_b are changes of the coordinates of the contact on the bottom surface ■

In fact, all the conditions for the contacts required for form-closure are strict conditions. Any form-closure grasp can be maintained after any infinitesimal displacements of the contacts. This test assures that the contacts can displace with a finite amount. From the above discussions, it is clear that placing the last contact away from the boundaries of the region can increase the magnitude of the allowable displacements.

2.4 Determination of Object Frame from a Form-closure Grasp

One of the advantage of form-closure grasp is that the object frame can be determined from the end points of the grippers. This is because in form-closure grasp the object cannot move when the grippers are fixed. What is required, in addition to the contact points, is the information of the surfaces on which each contact lies.

For objects with three pairs of parallel surfaces, the determination may be quite simple and efficient. If only the three pairs parallel surfaces are used in the grasp, there must be at least one contact on each surface for a form-closure grasp.

Let $\boldsymbol{n_1}$, $\boldsymbol{n_2}$ and $\boldsymbol{n_3}$ be the normals, pointing out of the surfaces, of three adjacent surfaces such that there are two contacts on the surface specified by $\boldsymbol{n_1}$

$\boldsymbol{p_1}$, $\boldsymbol{p_2}$, and $\boldsymbol{p_3}$ be the position vectors of the any contact points on the surfaces with normals $\boldsymbol{n_1}$, $\boldsymbol{n_2}$ and $\boldsymbol{n_3}$ respectively

$\boldsymbol{p_4}$, $\boldsymbol{p_5}$ and $\boldsymbol{p_6}$ be the position vectors on the surfaces parallel to those for $\boldsymbol{p_1}$, $\boldsymbol{p_2}$, and $\boldsymbol{p_3}$ respectively

$\boldsymbol{p_7}$ be the position vector of another contact lie on the same surface as $\boldsymbol{p_1}$

$\boldsymbol{l_1}$, $\boldsymbol{l_2}$ and $\boldsymbol{l_3}$ be the distances between pairs of parallel surfaces with normals equal to $\boldsymbol{n_1}$, $\boldsymbol{n_2}$ and $\boldsymbol{n_3}$ respectively

Without loss of generality, take $\boldsymbol{n_1}$ as the x-axis of the object frame, the y-axis as $\boldsymbol{R_2 n_2}$ and the z-axis as $\boldsymbol{R_3 n_3}$, where $\boldsymbol{R_2}$ and $\boldsymbol{R_3}$ are constant orthogonal rotation matrices. Using the fact that the distance between $\boldsymbol{p_1}$ and $\boldsymbol{p_4}$ along $\boldsymbol{n_1}$ is $\boldsymbol{l_1}$, we have

$$(p_1 - p_4)^T n_1 = l_1 \tag{2.5}$$

Similarly, we have

$$(p_7 - p_4)^T n_1 = l_1 \tag{2.6}$$

$$(p_2 - p_5)^T n_2 = l_2 \tag{2.7}$$

$$(p_3 - p_6)^T n_3 = l_3 \tag{2.8}$$

As $\boldsymbol{n_1}$ is a unit vector

$$n_{1x}^2 + n_{1y}^2 + n_{1z}^2 = 1 \tag{2.9}$$

where $\boldsymbol{n_{1x}}$, $\boldsymbol{n_{1y}}$ and $\boldsymbol{n_{1z}}$ are the components of $\boldsymbol{n_1}$

From Equation 2.5 and Equation 2.6,

$$\begin{bmatrix} d_{1x} & d_{1y} & d_{1z} \\ d_{2x} & d_{2y} & d_{2z} \end{bmatrix} \begin{bmatrix} n_{1x} \\ n_{1y} \\ n_{1z} \end{bmatrix} = \begin{bmatrix} l_1 \\ l_1 \end{bmatrix}$$

where $\begin{bmatrix} d_{1x} \\ d_{1y} \\ d_{1z} \end{bmatrix} = \boldsymbol{p}_1 - \boldsymbol{p}_4$ and $\begin{bmatrix} d_{2x} \\ d_{2y} \\ d_{2z} \end{bmatrix} = \boldsymbol{p}_7 - \boldsymbol{p}_4$

Solving for $\boldsymbol{n_{1x}}$, we have

$$\begin{bmatrix} n_{1y} \\ n_{1z} \end{bmatrix} = \frac{1}{d_{1y}d_{2z} - d_{2y}d_{1z}} \begin{bmatrix} (d_{2z} - d_{1z})l_1 + (d_{1z}d_{2x} - d_{2z}d_{1x})n_{1x} \\ (d_{1y} - d_{2y})l_1 + (d_{1x}d_{2y} - d_{2x}d_{1y})n_{1x} \end{bmatrix} \tag{2.10}$$

$$= \frac{1}{E} \begin{bmatrix} A + Bn_{1x} \\ C + Dn_{1x} \end{bmatrix}$$

Substitute Equation 2.10 into Equation 2.9,

$$(E^2 + B^2 + D^2)n_{1x}^2 + (AB + DC)n_{1x} + (A^2 + C^2 - E^2) = 0$$

From this quadratic equation, we can obtain two solutions for $\boldsymbol{n_{1x}}$. Using Equation 2.10, two solution for $\boldsymbol{n_1}$ are obtained.

As $\boldsymbol{R_2 n_2}$ is orthogonal to $\boldsymbol{n_1}$,

$$n_{1i}^T R_2 n_2 = 0 \tag{2.11}$$

where $\boldsymbol{n_{1i}}$ is the ith solution of $\boldsymbol{n_1}$

From Equation 2.7,

$$\begin{bmatrix} d_{3x} & d_{3y} & d_{3z} \end{bmatrix} R_2^T R_2 n_2 = l_2 \tag{2.12}$$

where $\begin{bmatrix} d_{3x} \\ d_{3y} \\ d_{3z} \end{bmatrix} = p_2 - p_5$

In addition, $R_3 n_3$ is orthogonal to both n_1 and $R_2 n_2$ so we may rewrite Equation 2.8 as

$$(p_3 - p_6)^T R_3^T (n_{1_i} \times R_2 n_2) = l_3$$

or
$$(p_3 - p_6)^T R_3^T (n_{1_i} \times) R_2 n_2 = l_3 \tag{2.13}$$

Using Equations 2.11, 2.12 & 2.13, $R_2 n_2$ can be determined uniquely for each solution of n_1. Then we can check which solution is correct using the following relation:

$$n_{2x}^2 + n_{2y}^2 + n_{2z}^2 = 1 \tag{2.14}$$

where n_{2x}, n_{2y} and n_{2z} are the components of n_2

Finally, $R_3 n_3$ can be obtained as $n_1 \times R_2 n_2$.

Without loss of generality, we may take the origin p_o such that its distances from any two parallel surfaces are equal. Considering p_1, we have

$$n_1^T (p_1 - p_o) = \frac{l_1}{2}$$

or
$$n_1^T p_o = n_1^T p_1 - \frac{l_1}{2} \tag{2.15}$$

Similarly, we can have

$$n_2^T p_o = n_2^T p_2 - \frac{l_2}{2} \tag{2.16}$$

$$n_3^T p_o = n_3^T p_2 - \frac{l_3}{2} \tag{2.17}$$

Using Equations 2.15, 2.16 & 2.17, the origin p_o can be solved. Matlab M-files implements this algorithm are listed in Appendix B.

This approach is quite simple but its application is restricted to objects with three pairs of parallel plane surfaces. For other objects, much more complicated approach may be required.

CHAPTER THREE

DYNAMIC MODEL OF MULTI-ARM SYSTEMS HANDLING ONE OBJECT

This chapter describes a dynamic model of multi-arm systems manipulating one single object. The dynamic model assumes that all the arms are not at singular position. In this model, the arms do not necessarily grasp the object rigidly. The connections between the arms and the object are described by a general model. The matrices required in the description of the connection between the object and the arms are discussed in separate sections. In the last section, the general model together with general approaches for using the model are described in detail.

3.1 System Description

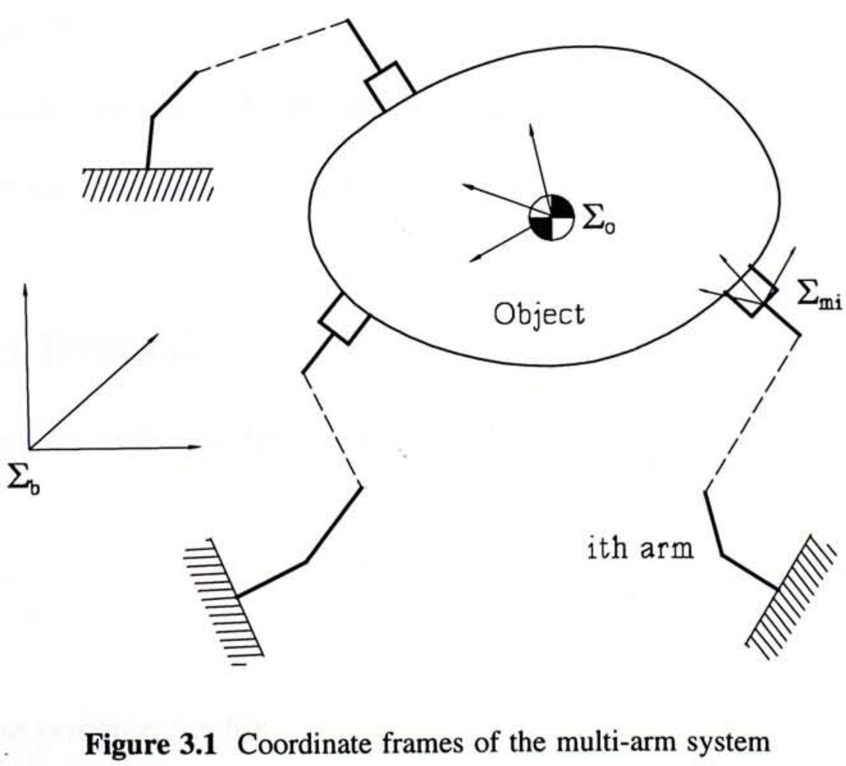

Figure 3.1 Coordinate frames of the multi-arm system

Consider a system with $\boldsymbol{n}$ robot arms. Let Σ_b be a fixed base coordinate frame. All variables are referred to and expressed in this frame if not otherwise specified. Also let Σ_o be the frame attached to the object with origin at the centre of mass; and Σ_{mi} be a frame attached to the gripper of the ith robot. Figure 3.1 shows the coordinate frames. It is assumed that each arms has six degrees of freedom with six joints and are not at singular positions.

3.2 Manipulator Dynamics

The dynamic equation of the ith manipulator can be written as

$$M_i(q_i)\dot{v}_{mi} + h_i(q_i, \dot{q}_i) = f_{mi} + f_{ci} + f_{ei} \tag{3.1}$$

where M_i is the operational space inertia matrix of the ith arm

v_{mi} is the velocity of the gripper frame

$h_i(q_i, \dot{q}_i)$ is the gravitational, Coriolis and centrifugal terms

q_i is the joint position vector

f_{mi}, f_{ci} and f_{ei} are the wrenches acted on the arm about Σ_{mi} and expressed in Σ_b due to reaction between object and arm, control torques and external forces respectively

A wrench due to friction may be included in f_{mi} if the gripper does not move relative to the object in the corresponding direction, otherwise, it must be considered in f_{ei}.

3.3 Object Dynamics

Denote the position and orientation of Σ_o by

$$r_o = \begin{bmatrix} x_o \\ \theta_o \end{bmatrix}$$

where x_o is the position vector

θ_o is the orientation represented using a set of Euler angles

The velocity of the object can then be expressed as

$$v_o = \begin{bmatrix} \dot{x}_o \\ \omega_o \end{bmatrix} = \begin{bmatrix} I_3 & 0_3 \\ 0_3 & \tilde{E}_o \end{bmatrix} \begin{bmatrix} \dot{x}_o \\ \dot{\theta}_o \end{bmatrix} = E_o \dot{r}_o$$

where ω_o is the angular velocity

$\tilde{E}_o(\theta_o)$ transforms the time derivatives of the Euler angles to angular velocities

The equation of motion of the object is

$$\begin{bmatrix} m_o I_3 & 0_3 \\ 0_3 & R_o^b I (R_o^b)^T \end{bmatrix} \dot{v}_o + \begin{bmatrix} -m_o(\omega_o \times \dot{x}_o + g) \\ \omega_o \times [R_o^b I (R_o^b)^T \omega_o] \end{bmatrix} = f_o + f_{eo}$$

$$I_o \dot{v}_o + h_o = f_o + f_{eo}$$

$$I_o E_o \ddot{r}_o + I_o \dot{E}_o \dot{r}_o + h_o = f_o + f_{eo} \qquad (3.2)$$

where m_o is the object mass

I_3 is a 3×3 identity matrix

I is the inertia tensor of the object about the axes of Σ_o, which is constant

R_o^b rotates vectors expressed in Σ_o to those expressed in Σ_b

g is the acceleration due to gravity

f_o and f_{eo} is the wrench about Σ_o and expressed in Σ_b acted on object due to reactions between object and arms and other external forces respectively

3.4 Contact Forces

3.4.1 Contact forces and moments applied to the manipulators

If the gripper of the ith arm can move relative to the object with (6-N_i) degree(s) of freedom, the force acted on the ith arm by the object can be written as:

$$f_{mi}^{mi} = \begin{bmatrix} w_{mi1} & w_{mi2} & \cdots & w_{mi(Ni)} \end{bmatrix} \begin{bmatrix} \mu_{i1} \\ \mu_{i2} \\ \vdots \\ \mu_{i(N_i)} \end{bmatrix}$$

$$f_{mi}^{mi} = W_{mi}\mu_i$$

$$f_{mi} = H_{mi}^{b} W_{mi}\mu_i \tag{3.3}$$

where f_{mi}^{mi} is f_{mi} expressed in Σ_{mi}

w_{mij} are linearly independent wrenches about Σ_{mi} and expressed in Σ_{mi}

μ_{ij} are the magnitudes of the corresponding wrenches

H_{mi}^{b} transforms screws expressed in Σ_{mi} to those expressed in Σ_b, and is equal to

$$\begin{bmatrix} R_{mi}^{b} & 0 \\ 0 & R_{mi}^{b} \end{bmatrix}$$

3.4.2 Contact forces and moments acted on object

The wrenches acted on the object about Σ_o by the arms are given as

$$f_o^o = \begin{bmatrix} W_{o1} & W_{o2} & \cdots & W_{on} \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_n \end{bmatrix}$$

$$H_b^o f_o = W_o\mu$$

where f_o^o is f_o expressed in Σ_o

W_{oi} are linearly independent wrenches about Σ_o and expressed in Σ_o, corresponding to W_{mi}'s

H_b^o transforms screws expressed in Σ_b to those expressed in Σ_o

Using generalized inverse, the vector μ can be expressed as

$$\mu = W_o^{-} H_b^o f_o + A_o\mu_I$$

where W_o^- is a generalized inverse of W_o

A_o is formed by a basis of the null space of W_o

μ_i can then be expressed as

$$\mu_i = W_{oi}^- H_b^o f_o + A_{oi}\mu_I \tag{3.4}$$

where W_{oi}^- and A_{oi} are the $(1+\sum_{j=1}^{i-1} N_j)th$ to $(N_i+\sum_{j=1}^{i-1} N_j)th$ rows of W_o^- and A_o

respectively

3.5 Kinematic Relations

The velocity of Σ_{mi} relative to Σ_o expressed in Σ_b can be written as

$${}_o v_{mi} = H_o^b T_i \alpha_i$$

where T_i is a $6\times(6\text{-}N_i)$ matrix with columns equal to the allowable twists of Σ_{mi}

relative to Σ_o and expressed in Σ_o

α_i is a vector containing the corresponding magnitudes of the twists

The velocity v_{mi} can then be expressed as

$$\begin{aligned} v_{mi} &= \begin{bmatrix} I_3 & {}_{mi}x_o \times \\ 0_3 & I_3 \end{bmatrix} v_o + H_o^b T_i \alpha_i \\ &= G_o^{mi} v_o + H_o^b T_i \alpha_i \\ &= G_o^{mi} E_o r_{\dot{o}} + H_o^b T_i \alpha_i \end{aligned} \tag{3.5}$$

where ${}_{mi}x_o$ is the displacement of the origin of Σ_o relative to Σ_{mi}

Differentiate the above equation with respect to time, the acceleration of Σ_{mi}, $\dot{v}_{mi}$, can be obtained as

$$\dot{v}_{mi} = \begin{bmatrix} I_3 & {}_{mi}x_o\times \\ 0_3 & I_3 \end{bmatrix} \dot{v}_o + \begin{bmatrix} 0_3 & (\omega_o\times)\, {}_{mi}x_o\times \\ 0_3 & 0_3 \end{bmatrix} v_o + H_o^b \frac{d}{dt}(T_i\alpha_i) + \begin{bmatrix} 2\omega_o\times & 0_3 \\ 0_3 & \omega_o\times \end{bmatrix} H_o^b T_i \alpha_i$$

$$= G_o^{mi}\dot{v}_o + G_1^{mi} v_o + H_o^b T_i \dot{\alpha}_i + H_o^b \dot{T}_i \alpha_i + G_2^{mi} H_o^b T_i \alpha_i$$

$$= G_o^{mi} E_o \ddot{r}_o + (G_o^{mi}\dot{E}_o + G_1^{mi} E_o)\dot{r}_o + H_o^b T_i \dot{\alpha}_i + (H_o^b \dot{T}_i + G_2^{mi} H_o^b T_i)\,\alpha_i \tag{3.6}$$

3.6 Overall System

Substitute Equation 3.2 into Equation 3.4, we have

$$\mu_i = W_{oi}^{-} H_b^o\,(I_o E_o \ddot{r}_o + I_o \dot{E}_o \dot{r}_o + Q_o - f_{eo}) + A_{oi}\mu_I \tag{3.7}$$

Substitute Equation 3.3 and Equation 3.6 into Equation 3.1, we have

$$M_i\Big[\, G_o^{mi} E_o \ddot{r}_o + (G_o^{mi}\dot{E}_o + G_1^{mi} E_o)\dot{r}_o + H_o^b T_i \dot{\alpha}_i + (H_o^b \dot{T}_i + G_2^{mi} H_o^b T_i)\,\alpha_i \Big] + h_i = H_{mi}^b W_{mi}\mu_i + f_{ci} + f_{ei} \tag{3.8}$$

Finally, put Equation 3.7 into Equation 3.8,

$$\begin{aligned} f_{ci} &= (M_i G_o^{mi} E_o - H_{mi}^b W_{mi} W_{oi}^{-} H_b^o\, I_o E_o)\ddot{r}_o + M_i H_o^b T_i \dot{\alpha}_i - H_{mi}^b W_{mi} A_{oi}\mu_I \\ &\quad + \Big[M_i (G_o^{mi}\dot{E}_o + G_1^{mi} E_o) - H_{mi}^b W_{mi} W_{oi}^{-} H_b^o\, I_o \dot{E}_o\Big]\dot{r}_o + M_i (H_o^b \dot{T}_i + G_2^{mi} H_o^b T_i)\,\alpha_i \\ &\quad + h_i - H_{mi}^b W_{mi} W_{oi}^{-} H_b^o\, Q_o + H_{mi}^b W_{mi} W_{oi}^{-} H_b^o f_{eo} - f_{ei} \\ &= (M_i G_o^{mi} E_o - H_{mi}^b W_{mi} W_{oi}^{-} H_b^o\, I_o E_o)\ddot{r}_o + M_i H_o^b T_i \dot{\alpha}_i - H_{mi}^b W_{mi} A_{oi}\mu_I + \tilde{h}_i + \tilde{f}_{ei} \end{aligned} \tag{3.9}$$

where $\tilde{h}_i$ is the terms due to the gravitational, Coriolis and centrifugal effects

$\tilde{f}_{ei}$ is the terms due to external forces and/or moments

Augmenting the matrix equations for i = 1, 2, ..., n, a set of equations describing the whole system can be obtained as follows:

$$f_c = (M G_o E_o - W_m^b W_o^{-} H_b^o\, I_o E_o)\ddot{r}_o + M\,T^b\dot{\alpha} - W_m^b A_o \mu_I + \tilde{h} + \tilde{f}_e \tag{3.10}$$

where $f_c = \begin{bmatrix} f_{c1} \\ \vdots \\ f_{cn} \end{bmatrix}$, $M = \begin{bmatrix} M_1 & & 0 \\ & \ddots & \\ 0 & & M_n \end{bmatrix}$, $G_o = \begin{bmatrix} G_o^{m1} \\ \vdots \\ G_o^{mn} \end{bmatrix}$,

$$W_m^b = \begin{bmatrix} H_{m1}^b W_{m1} & & 0 \\ & \ddots & \\ 0 & & H_{mn}^b W_{mn} \end{bmatrix}, \quad T^b = \begin{bmatrix} H_o^b T_1 & & 0 \\ & \ddots & \\ 0 & & H_o^b T_n \end{bmatrix},$$

$$\alpha = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{bmatrix}, \quad \tilde{h} = \begin{bmatrix} \tilde{h}_1 \\ \vdots \\ \tilde{h}_n \end{bmatrix}, \quad \tilde{f}_e = \begin{bmatrix} \tilde{f}_{e1} \\ \vdots \\ \tilde{f}_{en} \end{bmatrix}$$

This is a system of nonlinear differential equations with $6n$ independent variables, r_o, α_i, and μ_I.

3.7 Constraint Space Matrices

3.7.1 Determination of W_{mi} and W_{oi}

The matrices W_{mi} and W_{oi}[1] introduced in Section 3.4, together with μ_i, represent the reaction wrenches between two connected bodies. They are equivalent to the constraint space of the general joint model summarized by Lilly [30]. The main difference is that W_{mi} and W_{oi} are not normalized. There is no need to normalize them for use in the dynamic model.

Each column of these two matrices is a wrench which can be applied to the bodies being connected. Actually, these two matrices describe the same set of

---

[1] The subscripts $_{mi}$ and $_o$ are used to denote the ith manipulator and the object respectively in the previous sections. Though these notations are used here, the discussions that follow can be applied to any two objects being connected. These notations will also be used throughout the rest of this chapter to represent two connected bodies in general.

wrenches. The wrenches are just expressed in different frames and about different origins. The two matrices are related as follows:

$$W_{oi} = -\begin{bmatrix} R^o_{mi} & 0_3 \\ ({}_o x^o_{mi}\times)R^o_{mi} & R^o_{mi} \end{bmatrix} W_{mi} \tag{3.11}$$

where ${}_o x^o_{mi}$ is the displacement of the origin of Σ_{mi} relative to and expressed in Σ_o The columns of these matrices form a basis of the subspace spanned by all the possible reaction wrenches between the connected bodies. Therefore, to determine the matrices, a set of N_i linearly independent reaction wrenches should be determined first, where N_i is the number of degrees of freedom constrained by the joint.

For a joint formed by frictionless independent point contacts only, the determination is simple. In this case, W_{mi} is given by

$$\begin{bmatrix} e_{mi1} & e_{mi2} & \cdots & e_{mi(N_i)} \\ p_{mi1}\times e_{mi1} & p_{mi2}\times e_{mi2} & \cdots & p_{mi(N_i)}\times e_{mi(N_i)} \end{bmatrix}$$

where $p_{mi1}, p_{mi2}, \ldots, p_{mi(N_i)}$ are the contact locations relative to and expressed in Σ_{mi}

$e_{mi1}, e_{mi2}, \ldots, e_{mi(N_i)}$ which can be expressed in terms of the contact locations, are the corresponding unit contact normals pointing towards the surfaces

The corresponding W_{oi} is

$$\begin{bmatrix} e_{oi1} & e_{oi2} & \cdots & e_{oi(Ni)} \\ p_{oi1}\times e_{oi1} & p_{oi2}\times e_{oi2} & \cdots & p_{oi(Ni)}\times e_{oi(Ni)} \end{bmatrix}$$

where $p_{oi1}, p_{oi2}, \ldots, p_{oi(N_i)}$ are the contact locations relative to and expressed in Σ_o

e_{oij} is equal to $-R^o_{mi} e_{mij}$

R^o_{mi} rotates vectors expressed in Σ_{mi} to those expressed in Σ_o

When not all contacts are independent, each column of W_{mi} and W_{oi} would be a combination of zero-pitch wrenches instead of simply a zero-pitch wrench. In both cases, W_{mi} and W_{oi} can be expressed in terms of the contact locations only. In fact, it can be shown that any connection between two objects can be modelled as a joint with frictionless point contacts only. In Section 3.9, there will be further discussion on the general form of joint model.

Moreover, if only the relative position and orientation of the connected bodies are known, W_{mi} and W_{oi} can still be determined for these joints. The contact locations can be determined from the geometry of the corresponding contact surfaces. The surface normal or the corresponding contact normal can be expressed in terms of the contact locations, i.e.

$$e_{mij} = e_{mij}(p_{mij}) \qquad e_{oij} = e_{oij}(p_{oij})$$

From the relationship between the contact normals expressed in different reference frames, we have

$$e_{oij}(p_{oij}) = -R^{o}_{mi}\, e_{mij}(p_{mij}) \tag{3.12}$$

Also, from the relationship between p_{mij} and p_{oij}, we have

$$p_{oij} = {}_{o}x^{o}_{mi} + R^{o}_{mi} p_{mij} \tag{3.13}$$

Because ${}_{o}x^{o}_{mi}$ and R^{o}_{mi} are known, there are six equations with six unknowns which can be solved to obtain the contact locations, p_{mij} and p_{oij}. There should be a unique solution. Otherwise, a point contact does not exist between the contact surfaces. The complexity of the solution depends on the shapes of the contacting surfaces, i.e. $e_{mij}(p_{mij})$ and $e_{oij}(p_{oij})$. A special case is that one of the contact surface is a sharp point. While the contact surface attached to Σ_{mi} is a sharp point, e_{mij} is no longer a function of p_{mij} and p_{mij} is constant. Therefore, p_{oij} can be directly obtained from Equation 3.13.

3.7.2 Determination of $\dot{W}_{mi}$ and $\dot{W}_{oi}$

Although these two matrices are not in the dynamic model described in previous sections, they are necessary for describing a joint. For some common joints like revolute joints, prismatic joints, universal joints, etc, the values of W_{mi} and W_{oi} seem to be constant but this is not always true. For all joints, the value of R^o_{mi} and/or ${}_o x^o_{mi}$ change while the connected bodies move relative to one another. Therefore, W_{mi} and W_{oi} cannot be both constant. In case that one of them is constant, the computation of $\dot{W}_{mi}$ and $\dot{W}_{oi}$ is simple. If W_{mi} is constant, $\dot{W}_{mi}$ has all its elements equal to zero and the value of $\dot{W}_{oi}$ can be obtained as

$$-\begin{bmatrix} ({}_o\omega^o_{mi}\times)R^o_{mi} & 0_3 \\ [({}_o\dot{x}^o_{mi}\times)+({}_o x^o_{mi}\times)({}_o\omega^o_{mi}\times)]R^o_{mi} & ({}_o\omega^o_{mi}\times)R^o_{mi} \end{bmatrix} W_{mi}$$

where ${}_o\omega^o_{mi}$ is the angular velocity of Σ_{mi} relative to and expressed in Σ_o

${}_o\dot{x}^o_{mi}$ is the linear velocity of Σ_{mi} relative to and expressed in Σ_o

If W_{oi} is constant, the result is similar.

For a joint which is formed by only frictionless independent point contacts, W_{mi} is

$$\begin{bmatrix} e_{mi1} & e_{mi2} & \cdots & e_{mi(Ni)} \\ p_{mi1}\times e_{mi1} & p_{mi2}\times e_{mi2} & \cdots & p_{mi(Ni)}\times e_{mi(Ni)} \end{bmatrix}$$

By differentiating the variables in W_{mi}, $\dot{W}_{mi}$ can be obtained as

$$\begin{bmatrix} \dot{e}_{mi1} & \dot{e}_{mi2} & \cdots & \dot{e}_{mi(Ni)} \\ p_{mi1}\times\dot{e}_{mi1}+\dot{p}_{mi1}\times e_{mi1} & p_{mi2}\times\dot{e}_{mi2}+\dot{p}_{mi2}\times e_{mi2} & \cdots & p_{mi(Ni)}\times\dot{e}_{mi(Ni)}+\dot{p}_{mi(Ni)}\times e_{mi(Ni)} \end{bmatrix}$$

where $\dot{e}_{mi}$ is the time derivative of e_{mi} expressed in Σ_{mi}

$\dot{p}_{mi}$ is the time derivative of p_{mi} expressed in Σ_{mi}

Similarly, $\dot{W}_{oi}$ can be obtained as

$$\begin{bmatrix} \dot{e}_{oi1} & \dot{e}_{oi2} & \cdots & \dot{e}_{oi(Ni)} \\ p_{oi1}\times\dot{e}_{oi1}+\dot{p}_{oi1}\times e_{oi1} & p_{oi2}\times\dot{e}_{oi2}+\dot{p}_{oi2}\times e_{oi2} & \cdots & p_{oi(Ni)}\times\dot{e}_{oi(Ni)}+\dot{p}_{oi(Ni)}\times e_{oi(Ni)} \end{bmatrix}$$

where $\dot{e}_{oi}$ is the time derivative of e_{oi} expressed in Σ_o

$\dot{p}_{oi}$ is the time derivative of p_{oi} and expressed in Σ_o

As pointed out in the last sub-section, columns of W_{mi} and W_{oi} are combinations of zero-pitch wrenches if the contacts are not independent. Therefore, similar expressions for $\dot{W}_{mi}$ and $\dot{W}_{oi}$ can be obtained in this case. As a result, for joints with only frictionless point contacts, $\dot{W}_{mi}$ and $\dot{W}_{oi}$ can be expressed in terms of contact locations and their time derivatives.

To determine the velocities of the contact points, Equation 3.12 and Equation 3.13 can be differentiated with respect to time to obtain:

$$-R_o^{mi}\dot{e}_{oij} = \dot{e}_{mij} + {}_o\omega_{mi}^{mi}\times e_{mij} \tag{3.14}$$

$$\dot{p}_{oij} = R_{mi}^{o}\dot{p}_{mij} + {}_o\omega_{mi}^{o}\times R_{mi}^{o}p_{mij} + {}_o\dot{x}_{mi}^{o} \tag{3.15}$$

Therefore, the relative velocity between the bodies are also required in the determination of the contact point velocities. The time derivatives of e_{mi} and e_{oi}, can be obtained as:

$$\dot{e}_{mij} = \left(\frac{\partial e}{\partial p}\right)_{mij}\Bigg|_{p=p_{mij}} \dot{p}_{mij} = (e_p)_{mij}\dot{p}_{mij} \tag{3.16}$$

$$\dot{e}_{oij} = \left(\frac{\partial e}{\partial p}\right)_{oij}\Bigg|_{p=p_{oij}} \dot{p}_{oij} = (e_p)_{oij}\dot{p}_{oij} \tag{3.17}$$

where $(e_p)_{mij}$ and $(e_p)_{oij}$ are the partial differentiations of the contact normal, e_{mij} and e_{oij}, with respect to the contact point, p_{mij} and p_{oij}, respectively

Substitute Equation 3.16 and Equation 3.17 into Equation 3.14,

$$-R_o^{mi}(e_p)_{oij}\dot{p}_{oij} = (e_p)_{mij}\dot{p}_{mij} + {}_o\omega_{mi}^{mi}\times e_{mij} \tag{3.18}$$

Substitute Equation 3.15 into Equation 3.18 and rearrange, we have

$$\left[(e_p)_{mij} + R_o^{mi}(e_p)_{oij}R_{mi}^o\right]\dot{p}_{mij} = -R_o^{mi}(e_p)_{oij}\left[{}_o\omega_{mi}^o \times R_{mi}^o p_{mij} + {}_o\dot{x}_{mi}^o\right] - {}_o\omega_{mi}^{mi} \times e_{mij} \quad (3.19)$$

The coefficient matrix of $\dot{p}_{mij}$ in Equation 3.19 is nonsingular unless the contact surfaces of the objects do not have point contact at this location. This is because if the coefficient matrix is singular, the two surfaces must have the same gradient in a certain direction and a contact point is thus impossible. Therefore, $\dot{p}_{mij}$ can be obtained from Equation 3.19 as long as there is a point contact. When one of the contact surface is a sharp point, the computation is again different and simpler. Assume the sharp point is attached to Σ_{mi}. In this case, $(e_p)_{mij}$ are undefined, but p_{mi} is constant and $\dot{p}_{mi}$ is zero. Therefore, $\dot{p}_{oi}$ can be computed directly using Equation 3.15. $\dot{e}_{oij}$ and $\dot{e}_{mij}$ can then be obtained from Equation 3.17 and Equation 3.14 respectively.

3.7.3 Selection of A_o

The columns of matrix A_o in Equation 3.4 is a basis of the null space of matrix W_o which is composed by W_{oi}. Each column of A_o represents a combination of the magnitudes of the wrenches that produce no effect on the motion of the object. Therefore, if the object is form-closure grasped, all elements of A_o can be non-negative. When A_o is chosen such that all its elements are non-negative, the internal force and moment acting on the object increase with the increase of the value of μ_I. Furthermore, it is possible to choose an A_o such that each of the $(\sum_{i=1}^{n} N_i) - 6$ rows contains one element equal to one at different columns and all other elements equal to zero, i.e. in a row echelon form. With such an A_o, each element of μ_I is related to one of the contact force magnitude only, i.e.

$$\mu_{Ij} = \mu_j - W_{oj}^{-} H_b^o f_o$$

where μ_{Ij} is the jth element of μ_I

μ_j is the jth element of μ

W_{oj}^{-} is the jth row W_o^{-}

The μ_I corresponds to this selection is the difference between the actual contact force magnitude and the minimum norm solution. This selection allows the value of μ_I to be determined more efficiently as only $(\sum_{i=1}^{n} N_i) - 6$ contact force(s) are required to be determined.

3.8 Motion Space Matrices

3.8.1 Determination of T_i

The columns of T_i specifies a motion space of the general joint model [30]. Its columns form a basis of the sub-space spanned by all the twists that can be moved by one of the connected body relative to another. For some common joints such as revolute joint and prismatic joint, constant T_i may be obtained and the determination is simple. When T_i is not constant, one of the method to obtain T_i is to determine it from W_{mi} or W_{oi}, which may be obtained more easily. According to the definition of T_i given in Section 3.5, T_i is given as

$$T_i \alpha_i = {}_o v_{mi}^{o} \tag{3.20}$$

where ${}_o v_{mi}^{o}$ is the velocity of Σ_{mi} relative to and expressed in Σ_o

As any reaction wrenches acting on Σ_{mi} and any velocity of Σ_{mi} relative to Σ_o must be reciprocal, the following constraint for the relative velocity ${}_o v_{mi}^{o}$ can be obtained:

$$(W_{mi})^T H_o^{mi} {}_o v_{mi}^{o} = 0_{N_i \times 1} \tag{3.21}$$

where N_i is the number of degrees of freedom constrained by the joint

Thus, we have

$$(W_{mi})^T H_o^{mi} T_i = 0_{N_i \times (6 - N_i)}$$

In other words, T_i is a matrix whose columns form a basis of the null space of the matrix $W^o_{mi} = H^o_{mi} W_{mi}$. This shows how T_i can be obtained using the constraint space matrices. Using the relation between W_{mi} and W_{oi}, i.e. Equation 3.11, we may conclude that T_i is constant either when both W_{oi} and ${}_{mi}x_o$ are constant or when both W_{mi} and R^o_{mi} are constant.

According to Equation 3.20, for a certain ${}_o v^o_{mi}$, the value of α_i and the value of T_i depends on each other. α_i can have a more significant physical meaning by choosing a suitable T_i. To show this, first denote the position and orientation of Σ_{mi} relative to Σ_o as

$$
{}_o r^o_{mi} = \begin{bmatrix} {}_o x^o_{mi} \\ {}_o \theta_{mi} \end{bmatrix}
$$

where ${}_o\theta_{mi}$ is the orientation represented using a set of Euler angles

As the number of degree of freedom is $(6\text{-}N_i)$, ${}_o r^o_{mi}$ can be written as

$$
{}_o r^o_{mi} = \begin{bmatrix} f_{i1}({}_o r^o_{pmi}) & f_{i2}({}_o r^o_{pmi}) & f_{i3}({}_o r^o_{pmi}) & f_{i4}({}_o r^o_{pmi}) & f_{i5}({}_o r^o_{pmi}) & f_{i6}({}_o r^o_{pmi}) \end{bmatrix}^T
$$

where ${}_o r^o_{pmi}$ is a vector contains $(6\text{-}N_i)$ elements of ${}_o r^o_{mi}$

f_{ij} are functions of ${}_o r^o_{pmi}$

Thus, the time derivative of ${}_o r^o_{mi}$ is

$$
{}_o \dot{r}^o_{mi} = \frac{\partial}{\partial\, {}_o r^o_{pmi}} \begin{bmatrix} f_{i1} \\ f_{i2} \\ f_{i3} \\ f_{i4} \\ f_{i5} \\ f_{i6} \end{bmatrix} {}_o \dot{r}^o_{pmi}
$$

$$
= F_i \, {}_o \dot{r}^o_{pmi}
$$

where $\boldsymbol{F}_i$ is a $6\times(6\text{-}Ni)$ matrix containing the partial derivatives of $\boldsymbol{f}_{ij}$ with respect to

${}_o\boldsymbol{r}_{pmi}^{o}$

The velocity of the Σ_{mi} relative to Σ_o can then be written as

$$
{}_o v_{mi}^{o} = \begin{bmatrix} I_3 & 0 \\ 0 & {}_o\tilde{E}_{mi} \end{bmatrix} \boldsymbol{F}_i \, {}_o\dot{r}_{pmi} \tag{3.22}
$$

where ${}_o\tilde{\boldsymbol{E}}_{mi}({}_o\theta_{mi})$ transforms the time derivatives of the Euler angles ${}_o\dot{\theta}_{mi}$ to the relative angular velocity ${}_o\,\omega_{mi}^{o}$

Therefore, α_i is equal to ${}_o\boldsymbol{r}_{pmi}^{o}$ when $\boldsymbol{T}_i$ is in the following form:

$$
\begin{bmatrix} I_3 & 0 \\ 0 & {}_o\tilde{E}_{mi} \end{bmatrix} \boldsymbol{F}_i
$$

Conversely, based on Equation 3.22, $\boldsymbol{T}_i$ may be obtained directly from $\boldsymbol{W}_{mi}^{o}$ when α_i is equal to ${}_o\boldsymbol{r}_{pmi}^{o}$. In this case, the first three rows of $\boldsymbol{T}_i$ is equal to the first three rows of $\boldsymbol{F}_i$ and the other three rows depends on the fourth to the sixth rows of $\boldsymbol{F}_i$ only. Based on these two observations, α_i can be selected for joints with different degree(s) of freedom so that $(6\text{-}N_i)$ rows of $\boldsymbol{T}_i$ are known. The corresponding $\boldsymbol{T}_i$ can then be determined from $\boldsymbol{W}_{mi}^{o}$.

Because the elements of α_i are independent, $\boldsymbol{F}_i$ has a property that there are $(6\text{-}\boldsymbol{N}_i)$ rows which has one element equal to one and all others equal to zero. For example, if the joint has two degree of freedom and the first two elements of ${}_o\boldsymbol{r}_{mi}^{o}$ is selected as α_i, the corresponding $\boldsymbol{F}_i$ is

$$\begin{bmatrix} \dfrac{\partial ({}_o r^o_{mi})_1}{\partial ({}_o r^o_{mi})_1} & \dfrac{\partial ({}_o r^o_{mi})_1}{\partial ({}_o r^o_{mi})_2} \\ \dfrac{\partial ({}_o r^o_{mi})_2}{\partial ({}_o r^o_{mi})_1} & \dfrac{\partial ({}_o r^o_{mi})_2}{\partial ({}_o r^o_{mi})_2} \\ \vdots & \vdots \\ \dfrac{\partial ({}_o r^o_{mi})_6}{\partial ({}_o r^o_{mi})_1} & \dfrac{\partial ({}_o r^o_{mi})_6}{\partial ({}_o r^o_{mi})_2} \end{bmatrix}$$

where $({}_o\boldsymbol{r}^o_{mi})_j$ is the jth element of ${}_o\boldsymbol{r}^o_{mi}$

The values of the elements are unknown except the first two rows, they are given as

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

as $({}_o\boldsymbol{r}^o_{mi})_1$ and $({}_o\boldsymbol{r}^o_{mi})_2$ are independent variables. Thus, we have the following guidelines in selecting α_i:

1. If the jth (j = 1, 2 or 3) element of ${}_o\boldsymbol{x}^o_{mi}$ belongs to α_i , the jth row of $\boldsymbol{T}_i$ is simply a row with all elements equal to zero except the jth one.
2. If all the components of ${}_o\theta_{mi}$ are either an element of α_i or constant, the fourth to the sixth rows of $\boldsymbol{T}_i$ can be computed using ${}_o\theta_{mi}$.

The following example demonstrates how these guidelines can be applied.

Example 3.1

Consider a joint formed by a corner of a rectangular block and a gripper with four frictionless point contacts as shown in Figure 3.2. This joint has two degrees of freedom. If the gripper frame is selected as shown in Figure 3.2, the first two components of ${}_o\boldsymbol{x}^o_{mi}$ are independent variables and can be selected as α_i. With this

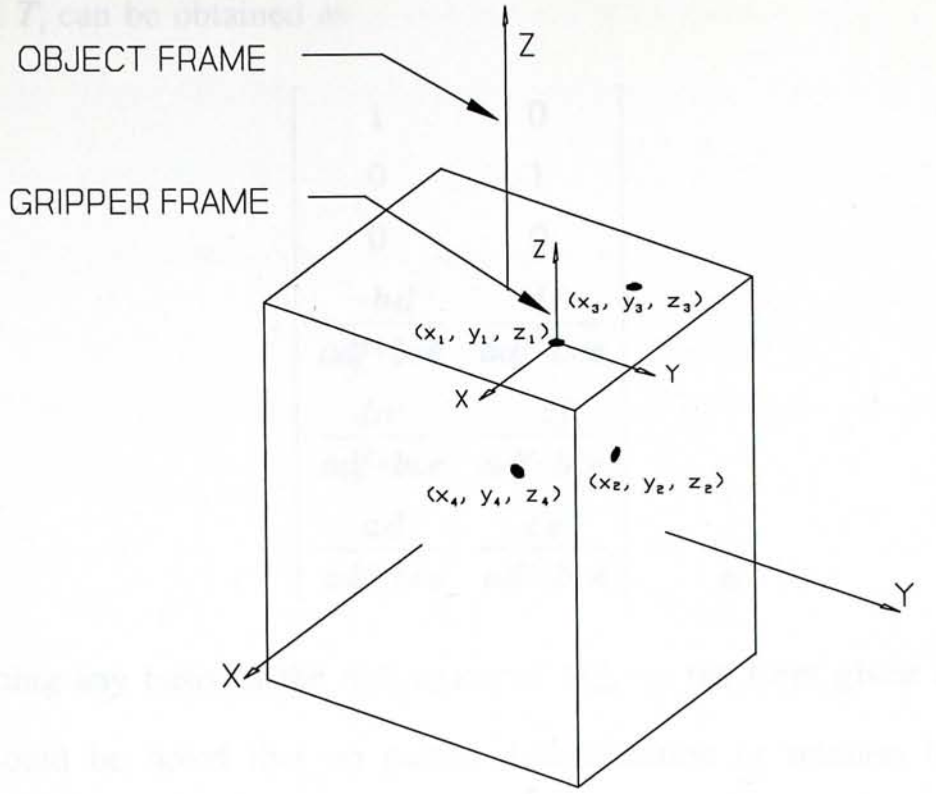

Figure 3.2 The configuration of the contacts and coordinate frames in Example 3.1

α_i, $\boldsymbol{T}_i$ is in the following form:

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ \times & \times \\ \times & \times \\ \times & \times \end{bmatrix} \tag{3.23}$$

where $\times$ denotes unknown expression

To obtain $\boldsymbol{T}_i$, $\boldsymbol{W}^o_{mi}$ may be first determined:

$$\begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & -1 & 0 & 0 \\ -1 & 0 & -1 & 0 \\ 0 & z_2-z_1 & y_1-y_3 & 0 \\ 0 & 0 & x_3-x_1 & z_1-z_4 \\ 0 & x_1-x_2 & 0 & y_4-y_1 \end{bmatrix}$$

The required T_i can be obtained as

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ \dfrac{-bd}{adf+bce} & \dfrac{df}{adf+bce} \\ \dfrac{bc}{adf+bce} & \dfrac{-cf}{adf+bce} \\ \dfrac{ad}{adf+bce} & \dfrac{ce}{adf+bce} \end{bmatrix}$$

by transforming any basis of the null space of W^o_{mi} to the form given in Equation 3.23. It should be noted that no partial differentiation or relation between the components of ${}_o r^o_{mi}$ are required in the determination. If only the value of T_i is required, it can be computed numerically without obtaining the analytical form.

As the gripper frame can only rotate relative to the block about two axes, the orientation components can also be selected as α_i. If z-x-y or z-x-z Euler angle set[1] is used in the representation of orientation, the first two components of ${}_o\theta_{mi}$ are independent and can be used as α_i. The corresponding T_i has the following form:

$$\begin{bmatrix} \times & \times \\ \times & \times \\ 0 & 0 \\ ({}_o\tilde{E}_{mi})_1 & ({}_o\tilde{E}_{mi})_2 \end{bmatrix}$$

where $({}_o\tilde{E}_{mi})_j$ is the jth column of ${}_o\tilde{E}_{mi}$, which is a function of ${}_o\theta_{mi}$

Using this result, the corresponding T_i can also be determined from W^o_{mt} ■

3.8.2 Determination of $\dot{T}_i$

Unless T_i is constant, T_i must first be expressed in terms of certain variables before $\dot{T}_i$ can be determined. The selection of these variables are critical in the

[1] This refers to the angle set conventions given in [41].

determination of $\dot{\boldsymbol{T}}_i$ as the complexity depends on the variables used.

Consider joints with frictionless point contacts only. Expression for $\boldsymbol{T}_i$ may also be obtained according to Equation 3.22. As f_{ij} may be highly nonlinear, it is very difficult to determine $\dot{\boldsymbol{T}}_i$ from this expression of $\boldsymbol{T}_i$. Even the expression for $\boldsymbol{F}_i$ can hardly be obtained in some situation. A comparably simple expression may be obtained for $\dot{\boldsymbol{T}}_i$ based on the use of contact locations. If the contacts are independent, $\boldsymbol{W}^o_{mi}$ can then be obtained as

$$\boldsymbol{H}^o_{mi}\boldsymbol{W}_{mi} = \begin{bmatrix} \boldsymbol{R}^o_{mi} & 0 \\ 0 & \boldsymbol{R}^o_{mi} \end{bmatrix} \begin{bmatrix} \boldsymbol{e}_{mi1} & \boldsymbol{e}_{mi2} & \cdots & \boldsymbol{e}_{mi(Ni)} \\ \boldsymbol{p}_{mi1}\times\boldsymbol{e}_{mi1} & \boldsymbol{p}_{mi2}\times\boldsymbol{e}_{mi2} & \cdots & \boldsymbol{p}_{mi(Ni)}\times\boldsymbol{e}_{mi(Ni)} \end{bmatrix}$$

$$= \begin{bmatrix} -\boldsymbol{e}_{oi1} & -\boldsymbol{e}_{oi2} & \cdots & -\boldsymbol{e}_{oi(Ni)} \\ \boldsymbol{e}_{oi1}\times(\boldsymbol{p}_{oi1} - {}_o\boldsymbol{x}^o_{mi}) & \boldsymbol{e}_{oi2}\times(\boldsymbol{p}_{oi2} - {}_o\boldsymbol{x}^o_{mi}) & \cdots & \boldsymbol{e}_{oi(Ni)}\times(\boldsymbol{p}_{oi(Ni)} - {}_o\boldsymbol{x}^o_{mi}) \end{bmatrix}$$

As discussed in the last section, $\boldsymbol{T}_i$ can be obtained directly from $\boldsymbol{W}^o_{mi}$ for certain α_i. Therefore, $\boldsymbol{T}_i$ can be expressed in terms of the relative position of the connected bodies and contact locations. As a result, $\dot{\boldsymbol{T}}_i$ can be expressed in terms of relative position, ${}_o\boldsymbol{x}^o_{mi}$, and velocity, ${}_o\dot{\boldsymbol{x}}^o_{mi}$, between the connected bodies, the contact locations, $\boldsymbol{p}_{oij}$, and velocities of contact points, $\dot{\boldsymbol{p}}_{oij}$. This result is also true for all joints formed by frictionless point contacts.

3.9 General Joint Model

As pointed out in Section 3.7.1 that any joint between two rigid bodies can be modelled by frictionless point contacts. Any wrenches can be expressed as a zero-pitch wrench or a difference between two zero-pitch wrenches[1]. Every zero-pitch wrench, in turn, can be considered as the wrench applied through a frictionless point contact. Thus, joints with frictionless point contacts only may be regarded as a

---

[1] See Appendix C for a proof of this.

general joint that can be used to model any type of joints. In addition, two contacts are sufficient to model one constraint.

In previous sections, it is shown that several matrices are required to describe a joint in a dynamic model. These include the constraint space matrices, i.e. $\boldsymbol{W}_{mi}$, $\boldsymbol{W}_{oi}$, $\dot{\boldsymbol{W}}_{mi}$, $\dot{\boldsymbol{W}}_{oi}$, and $\boldsymbol{A}_o$, and the motion space matrices, i.e. $\boldsymbol{T}_i$ and $\dot{\boldsymbol{T}}_i$. The determination of these matrices for joints formed by frictionless point contacts are all discussed in Section 3.7 and Section 3.8. Thus, if a joint is modelled with frictionless point contacts, the methods discussed before can be applied.

The contacts used to model a joint should satisfy a basic requirement. The contacts must be able to provide the same set of wrenches applied through the joint. In order to model a joint properly, this basic requirement must also be satisfied while the connected bodies move relatively to each other. In other words, the geometry of the contacting surfaces used in the model should match with the real situation. Otherwise, the time derivatives of the constraint space matrices and the motion space matrices cannot be computed using the general approaches.

The modelling joints with frictionless point contacts simplifies the analysis but does not mean that all types of joints can be designed with frictionless point contacts. Though it may be possible, the main objective is to model the joint at a certain configuration so that general approaches for kinematic analysis can be applied. The model may not be physically realizable. As an example, consider modelling a point contact with friction on a curved surface. Three frictionless coincident point contacts lying on three perpendicular surfaces may be used as a model of this joint. It is obvious that this model cannot be obtained in real life.

Moreover, in the model, it may require that contact forces can be applied in both directions of the contact normals. An example is a frictionless non-triangular surface contact. In this case, any three usual frictionless point contacts are not equivalent to the surface contact even when the force distribution is ignored. The contact forces may be required to apply in both directions of each contact normal.

3.9.1 Models for common joints

This section demonstrates how some common joints can be modelled using frictionless point contacts. In the models described here, one body (Body 1) has flat, cylindrical or/and spherical surfaces at the contact locations and the other (Body 2) has sharp points.

Revolute joint

Figure 3.3 shows a revolute joint and Figure 3.4 shows how five frictionless point contacts on a cylinder can be used to model the revolute joint. The model consists of four contacts on the curved surface of the cylinder and one at the end.

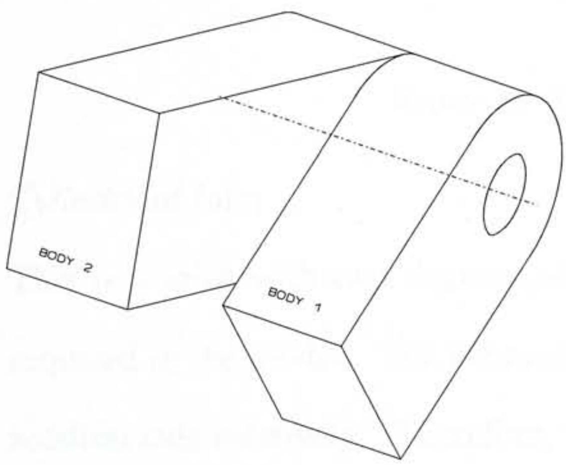

Figure 3.3 A revolute joint

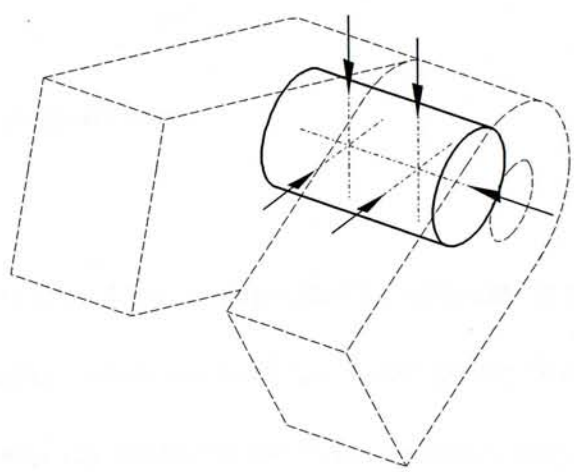

Figure 3.4 The model for revolute joint

Prismatic joint

Two flat surfaces are used to model this type of joint. An example of a prismatic joint and the corresponding model are shown in Figure 3.5 and Figure 3.6 respectively. A requirement for this model is that the three contacts on the same surface must be non-collinear.

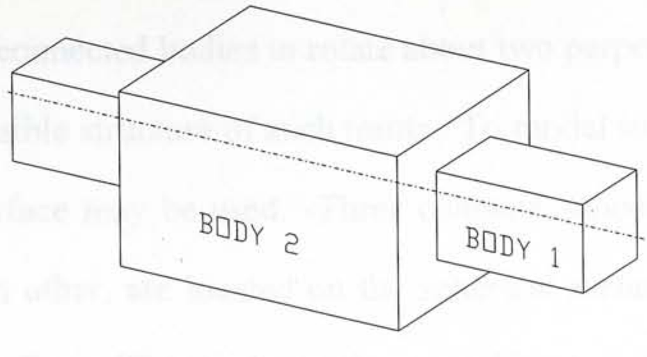

Figure 3.5 A prismatic joint

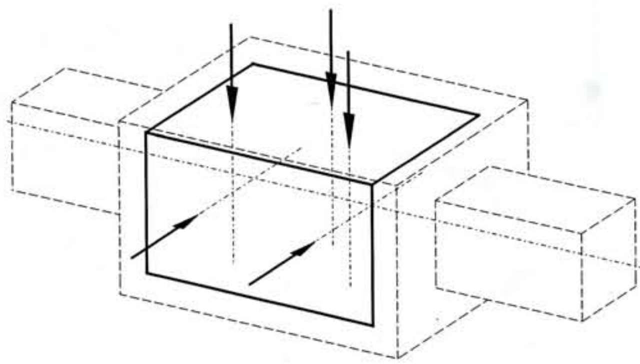

Figure 3.6 The model for prismatic joint

Cylindrical joint

This is a joint with two degrees of freedom so that four independent contacts are required in the model. The connected objects may rotate as well as move along the rotation axis relatively. Therefore, a model based on frictionless point contact may be obtained by removing the contact at the end from the model for the revolute joint. The structure and a model of the joint is shown in Figure 3.7 and Figure 3.8 respectively.

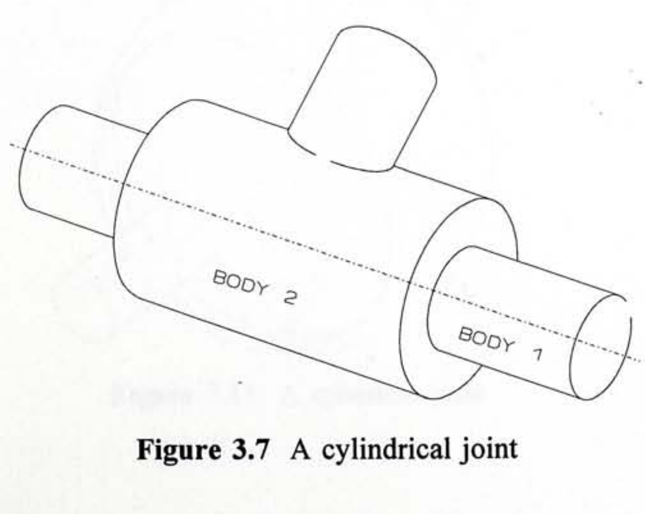

Figure 3.7 A cylindrical joint

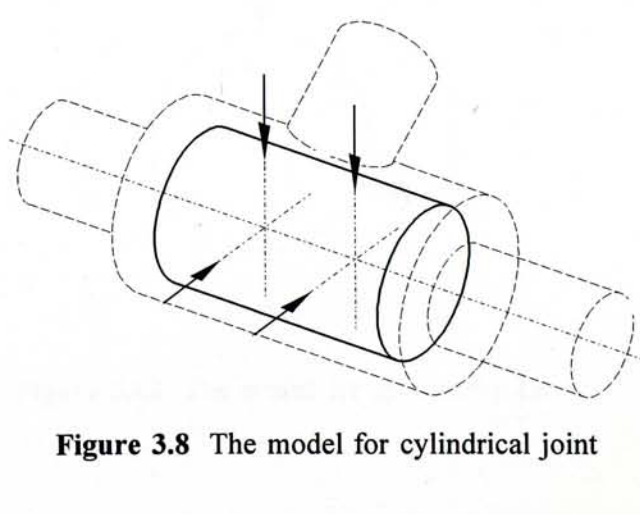

Figure 3.8 The model for cylindrical joint

Universal joint

This joint allows the connected bodies to rotate about two perpendicular axis. Shown in Figure 3.9 is a possible structure of such joints. To model such a joint, a spherical surface and a flat surface may be used. Three contacts, whose contact normals are perpendicular to each other, are located on the spherical surface and another one is located on the flat surface. The model is shown in Figure 3.10.

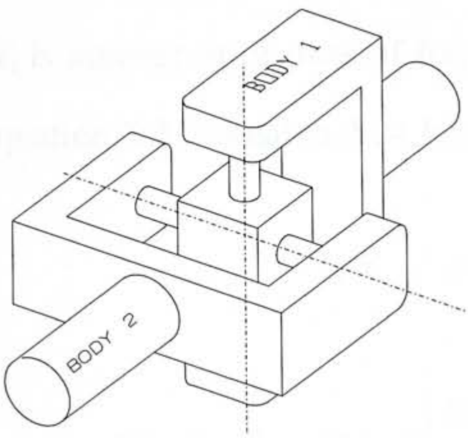

Figure 3.9 A universal joint

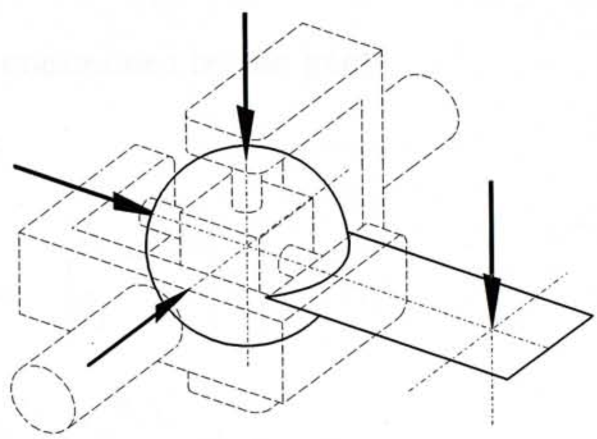

Figure 3.10 The model for universal joint

Spherical joint

This joint allows three-dimensional relative rotation. A possible structure is a socket joint as shown in Figure 3.11. A model similar to that for universal joint can be applied. In this case, only three contacts on the spherical surface are required. The resulted model is shown in Figure 3.12.

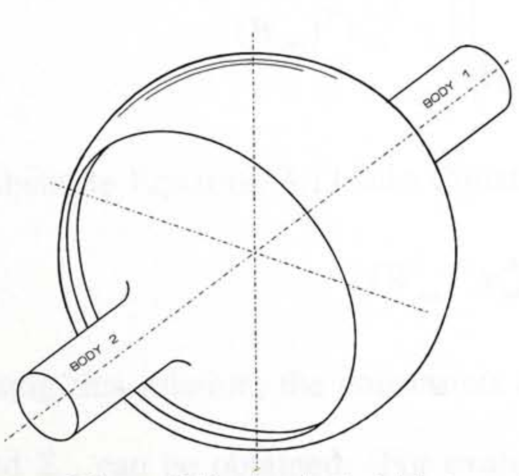

Figure 3.11 A spherical joint

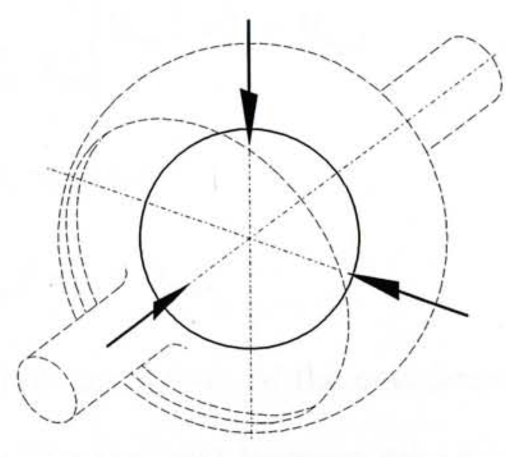

Figure 3.12 The model for spherical joint

3.9.2 Constraints provided by a joint

The matrices W_{mi} and W_{oi} give the constraints of the relative velocity of the connected bodies. This is discussed in Section 3.8.1. Alternatively, the constraints can be expressed in terms of velocities of the bodies instead of relative velocity. From Equation 3.21

$$(W_{mi})^T H_b^{mi} {}_o v_{mi} = 0_{N_i \times 1} \tag{3.24}$$

where N_i is number of degrees of freedom constrained by the joint

Using Equation 3.5, Equation 3.24 becomes

$$(W_{mi})^T H_b^{mi} \left(v_{mi} - \begin{bmatrix} I_3 & {}_{mi}x_o \times \\ 0_3 & I_3 \end{bmatrix} v_o \right) = 0_{N_i \times 1}$$

$$(W_{mi})^T v_{mi}^{mi} - (W_{mi})^T \begin{bmatrix} R_b^{mi} & R_b^{mi} ({}_{mi}x_o^{mi} \times) \\ 0_3 & R_b^{mi} \end{bmatrix} H_o^b v_o^o = 0_{N_i \times 1}$$

$$\because R_b^{mi} ({}_{mi}x_o \times) = ({}_{mi}x_o^{mi} \times) R_b^{mi} \text{ and } R_{mi}^o ({}_{mi}x_o^{mi} \times) = ({}_{mi}x_o^o \times) R_{mi}^o$$

$$(W_{mi})^T v_{mi}^{mi} - (W_{mi})^T \begin{bmatrix} R_o^{mi} & ({}_{mi}x_o^{mi} \times) R_o^{mi} \\ 0_3 & R_o^{mi} \end{bmatrix} v_o^o = 0_{N_i \times 1}$$

$$(W_{mi})^T v_{mi}^{mi} - \left(\begin{bmatrix} R_{mi}^o & 0_3 \\ ({}_{mi}x_o^o \times) R_{mi}^o & R_{mi}^o \end{bmatrix} W_{mi} \right)^T v_o^o = 0_{N_i \times 1} \tag{3.25}$$

Substitute Equation 3.11 into Equation 3.25, we have

$$(W_{mi})^T v_{mi}^{mi} + (W_{oi})^T v_o^o = 0_{Ni \times 1} \tag{3.26}$$

Using this relation, the constraints on the time derivatives of the coordinates of Σ_o and Σ_{mi} can be obtained. For example, consider the joint between the gripper of a robot and an object. If the joint variables q_i is used as the coordinates of Σ_{mi} and the coordinates of the object is taken as r_o, the constraints on the time derivatives of the coordinates can be obtained as

$$(W_{mi})^T J_i^{mi} q_i + (W_{oi})^T H_b^o E_o \dot{r}_o = 0_{Ni\times 1}$$

where J_i^{mi} is the Jacobian giving the velocity expressed in Σ_{mi}

Although the constraints on the time derivatives of the coordinates can be obtained easily from W_{mi} and W_{oi}, the corresponding constraints on the coordinates are also required in the analysis of the system. One of the application which requires the corresponding constraints on coordinates and their time derivatives is the constraint violation stabilization in numerical integration [33]. The corresponding coordinates constraints vary with the selection of W_{mi} and W_{oi} and therefore there is no general method to determine the constraints. Nevertheless, if W_{mi} and W_{oi} are selected such that each column is in the form:

$$\begin{bmatrix} e_i \\ p_i \times e_i \end{bmatrix} \text{ or } \begin{bmatrix} e_i \\ p_i \times e_i \end{bmatrix} - \begin{bmatrix} e_j \\ p_j \times e_j \end{bmatrix}$$

where e_i and e_j are contact normals

p_i and p_j are contact locations

the corresponding constraints on the coordinates can be determined.

Consider an object with a fixed point at p expressed in a frame Σ_o attached to the object. If the point is moved along a direction given by a unit vector e, the resulted velocity v_o of the frame Σ_o expressed in Σ_o is in the form

$$\alpha \begin{bmatrix} e \\ p \times e \end{bmatrix}$$

where α is a scalar specifying the magnitude of the motion

Therefore, the value

$$\begin{bmatrix} e \\ (p \times e) \end{bmatrix}^T v$$

is the magnitude of the linear velocity of the point p along the direction given by e

when the frame moves with a velocity $\boldsymbol{v}$. As a result, if there is a constraint

$$\begin{bmatrix} e \\ (p \times e) \end{bmatrix}^T v_o = 0$$

the point $\boldsymbol{p}$ cannot move along the direction of $\boldsymbol{e}$. Similarly, if there is a constraint

$$\left(\begin{bmatrix} e_i \\ (p_i \times e_i) \end{bmatrix} - \begin{bmatrix} e_j \\ (p_j \times e_j) \end{bmatrix} \right)^T v_o = 0$$

the linear velocities of the two points $\boldsymbol{p}_i$ and $\boldsymbol{p}_j$ along $\boldsymbol{e}_i$ and $\boldsymbol{e}_j$ respectively must have the same magnitude. Thus, under this constraint, any displacement of $\boldsymbol{p}_i$ along $\boldsymbol{e}_i$ must equal to that of $\boldsymbol{p}_j$ along $\boldsymbol{e}_j$. Using these two results, corresponding constraints on the coordinates and their time derivatives can be obtained from the structure of $\boldsymbol{W}_{mi}$.

To sum up, for a constraint on the time derivatives of the coordinates of two connected bodies derived from

$$(W_{mi})^T {}_o v_{mi}^{mi} = 0_{Ni \times 1} \tag{3.27}$$

the corresponding constraint on the coordinates can be obtained as:

1. For a column of $\boldsymbol{W}_{mi}$ in the form

$$\begin{bmatrix} e_{mij} \\ p_{mij} \times e_{mij} \end{bmatrix}$$

the corresponding coordinate constraint is that the displacement of the point or contact $\boldsymbol{p}_{mij}$ along $\boldsymbol{e}_{mij}$ is zero.

2. For a column of $\boldsymbol{W}_{mi}$ in the form

$$\begin{bmatrix} e_{mij} \\ p_{mij} \times e_{mij} \end{bmatrix} - \begin{bmatrix} e_{mik} \\ p_{mik} \times e_{mik} \end{bmatrix}$$

the corresponding coordinate constraint is that the difference between the displacement of $\boldsymbol{p}_{mij}$ along $\boldsymbol{e}_{mij}$ and that of $\boldsymbol{p}_{mik}$ along $\boldsymbol{e}_{mik}$ must be zero.

In order to obtain the equation for these constraints, the function ϕ_{mij} should first be defined:

$$\phi_{mij} = e_{mij}^T ({}_{mi}d_{mij}^{mi} - {}_{mi}d_{oij}^{mi})$$

where ${}_{mi}d_{mij}^{mi}$ is the position vector, expressed in and relative to Σ_{mi}, of the intersection of the contact normal and the contact surface of the object attached to Σ_{mi}

${}_{mi}d_{oij}^{mi}$ is the position vector, expressed in and relative to Σ_{mi}, of the intersection of the contact normal and the contact surface of the object attached to Σ_o

Therefore, for the first case above, the constraint equation can be given as

$$\phi_{mij} = 0$$

And in the other case, the constraint equation is

$$\phi_{mij} - \phi_{mik} = 0$$

If the contact locations are computed using the method described in Section 3.7.1, it may not required to compute the values of ${}_{mi}d_{mij}^{mi}$ and ${}_{mi}d_{oij}^{mi}$ directly. If the computed intersections of the contact normal and the contact surfaces are not the same point, the contact normal computed from the contact point obtained from this method is not a unit vector. This is because the equations used to solve the contact point only specify that the contact point lie on a line normal to both the contact surfaces but there is no constraint on the magnitude of the resulted contact normal. In other words, the computed contact point may not lie on the contact surfaces.

Let the computed contact point relative to Σ_{mi} and Σ_{oi} be $\tilde{p}_{mij}$ and $\tilde{p}_{oij}$ and the corresponding contact normals be $\tilde{e}_{mij}$ and $\tilde{e}_{oij}$. Using the functions giving the normalized contact normals e_{mij} and e_{oij}, we have

$$e_{mij}\left(\tilde{p}_{mij} + \bar{d}_{mij}\frac{\tilde{e}_{mij}}{\|\tilde{e}_{mij}\|}\right) = \frac{\tilde{e}_{mij}}{\|\tilde{e}_{mij}\|} \tag{3.28}$$

$$e_{oij}\left(\tilde{p}_{oij} + \bar{d}_{oij}\frac{\tilde{e}_{oij}}{\|\tilde{e}_{oij}\|}\right) = \frac{\tilde{e}_{oij}}{\|\tilde{e}_{oij}\|} \tag{3.29}$$

where $e_{mij}(p_{mij})$ and $e_{oij}(p_{oij})$ are functions of contact points with values equal to e_{mij} and e_{oij} respectively

$\bar{d}_{mij}$ and $\bar{d}_{oij}$ are distances, along e_{mij} and e_{oij} respectively, measured from the computed contact points to the intersections of the surfaces with the contact normals

From these two equations, the values of $\bar{d}_{mij}$ and $\bar{d}_{oij}$ can be determined. The values obtained must be checked such that $(\tilde{p}_{mij} + \bar{d}_{mij}e_{mij})$ and $(\tilde{p}_{oij} + \bar{d}_{oij}e_{oij})$ are points on the surfaces. These two values can also be given as

$$\bar{d}_{mij} = e_{mij}^{T}\left({}_{mi}d_{mij}^{mi} - p_{mij}\right)$$

$$\bar{d}_{oij} = -e_{mij}^{T}\left({}_{mi}d_{oij}^{mi} - p_{mij}\right)$$

Thus, the value of the corresponding constraint function ϕ_{mij} is equal to:

$$\phi_{mij} = \bar{d}_{mij} + \bar{d}_{oij}$$

If one of the contact surface is a sharp point, the computation is different. As discussed in Section 3.7.1, the determination of the contact point is also different in this situation. If the sharp point is attached to Σ_{mi}, p_{mij} is constant and p_{oij} is computed directly using Equation 3.13. Therefore, $\bar{d}_{mij}$ is not given by Equation 3.28 but equals to zero. The constraint function ϕ_{mij} is simply:

$$\phi_{mij} = \bar{d}_{oij}$$

However, it should be noted that the computation is simpler when one of the contact surfaces is a sharp point. This is also true in determining the contact locations and the velocities of the contacts as discussed in Section 3.7. As a result, it is advantageous to design grippers with spherical fingers. This is because the centre of the contact surface has a constant distance from the other contact surface and the contact may be modelled as a sharp point. This design would not scratch and damage the other contact surface like a real sharp point does.

Lastly, if some of the constraints are nonholonomic, the nonholonomic constraints should be modelled by separate contacts. This allows the constraints on the coordinates to be computed using the general approach.

CHAPTER FOUR

FORWARD DYNAMICS OF MULTI-ARM SYSTEMS HANDLING ONE OBJECT

This chapters presents an efficient algorithm to solve the forward dynamics of a multi-arm system handling one single object. This algorithm is a modification of two existing ones [38, 39]. The modified approach is more general and it can be applied to a larger class of problems. There are also several advantages of this algorithm over the previous ones. An algorithm used with this modified approach to solve the drift in numerical integration is also discussed at the end of this chapter.

4.1 Previous Works

4.1.1 Lilly and Orin's approach

Lilly and Orin [38] suggested a method to solve the forward dynamic problem of a multi-arm system connected in parallel to an object. The main idea of the approach is to solve for the acceleration of the object before computing the accelerations of the arms.

Consider the multi-arm system described in Section 3.1. The acceleration of Σ_{mi} can be expressed in terms of the object acceleration:

$$\begin{aligned} \dot{v}_{mi} &= G_o^{mi}\dot{v}_o + {}_o\dot{v}_{mi} \\ &= G_o^{mi}\dot{v}_o + (\hat{T}_i a_i + \hat{W}_i a_i^c) \end{aligned} \tag{4.1}$$

where $\hat{T}_i$ and $\hat{W}_i$ are matrices describing the motion space and the constraint space of the joint between the object and the ith arm which have the following properties:

$$\hat{W}_i^T \hat{T}_i = 0_{N_i \times (6-N_i)}$$
$$\hat{W}_i^T \hat{W}_i = I_{N_i}$$

The contact force then may be solved in terms of the object acceleration:

From Equation 3.1,

$$\begin{aligned} \dot{v}_{mi} - M_i^{-1} f_{mi} &= M_i^{-1}(f_{ci} + f_{ei} - h_i) \\ G_o^{mi} \dot{v}_o + \hat{T}_i a_i + \hat{W}_i a_i^c - M_i^{-1} f_{mi} &= \tilde{h}_i \end{aligned}$$

where M_i^{-1} is the inverse of the operational space inertia matrix and exists even when the arm is at singular position

As the value of a_i depends on the value of $\dot{v}_{mi}$, a_i must be eliminated. Pre-multiply the above equation with $\hat{W}_i^T$, we have

$$\hat{W}_i^T G_o^{mi} \dot{v}_o + a_i^c - \hat{W}_i^T M_i^{-1} f_{mi} = \hat{W}_i^T \tilde{h}_i \tag{4.2}$$

a_i^c, if not zero, can be computed from the velocities.

Express f_{mi} as

$$f_{mi} = \hat{T}_i \lambda_i + \hat{W}_i \lambda_i^c \tag{4.3}$$

and substitute the result into Equation 4.2, we have

$$\begin{aligned} \hat{W}_i^T G_o^{mi} \dot{v}_o + a_i^c - \hat{W}_i^T M_i^{-1} \hat{W}_i \lambda_i^c - \hat{W}_i^T M_i^{-1} \hat{T}_i \lambda_i &= \hat{W}_i^T \tilde{h}_i \\ \tilde{M}_i^{-1} \lambda_i^c &= -\hat{W}_i^T (\tilde{h}_i + M_i^{-1} \hat{T}_i \lambda_i) \\ &\quad + a_i^c + \hat{W}_i^T G_o^{mi} \dot{v}_o \end{aligned} \tag{4.4}$$

where $\tilde{M}_i^{-1} = \hat{W}_i^T M_i^{-1} \hat{W}_i$

Thus, f_{mi} can be expressed in terms of the object acceleration if $\tilde{M}_i^{-1}$ is nonsingular and λ_i is known.

$$\begin{aligned} f_{mi} &= \hat{W}_i \tilde{M}_i \left[-\hat{W}_i^T (\tilde{h}_i + M_i^{-1} \hat{T}_i \lambda_i) + a_i^c + \hat{W}_i^T G_o^{mi} \dot{v}_o \right] + \hat{T}_i \lambda_i \\ &= P_i + \hat{W}_i \tilde{M}_i \hat{W}_i^T G_o^{mi} \dot{v}_o \end{aligned} \tag{4.5}$$

$\tilde{M}_i^{-1}$ may be nonsingular even when the arm is in a singular position and/or has fewer than six degrees of freedom. This is different from the discussion by Lilly and Orin [38]. $\tilde{M}_i^{-1}$ will become singular only when the arm cannot move in the direction specified by $\hat{W}_i$. This fact allows the method to be used for cases where some of the arms have less than six degrees of freedom.

The next step is to eliminate the contact forces from the object dynamic equation:

$$I_o\dot{v}_o + Q_o = -\sum_{i=1}^{n}(G_o^{mi})^T f_{mi}$$

$$I_o\dot{v}_o + Q_o = -\sum_{i=1}^{n}(G_o^{mi})^T (P_i + \hat{W}_i\tilde{M}_i\hat{W}_i^T G_o^{mi}\dot{v}_o)$$

$$\left[I_o + \sum_{i=1}^{n}(G_o^{mi})^T \hat{W}_i\tilde{M}_i\hat{W}_i^T G_o^{mi}\right]\dot{v}_o = -\sum_{i=1}^{n}(G_o^{mi})^T P_i - Q_o \tag{4.6}$$

The acceleration of the object $\dot{v}_o$ can then be obtained by solving the above equation. Using the values of $\dot{v}_o$, f_{mi} and $\dot{v}_{mi}$ can be computed using Equation 4.5 and Equation 3.1.

For some cases, including manipulation with multiple point contacts, $\hat{W}_i$ and $\hat{T}_i$ are not constant. In such cases, a_i^c must not be zero and the determination of a_i^c may be quite difficult. Furthermore, λ_i and λ_i^c may not possess significant physical meaning and the values of λ_i may also be quite difficult to be computed. However, such problems do not present if $\hat{W}_i$ and $\hat{T}_i$ are constant.

4.1.2 McMillan, Sadayappan and Orin's approach

In order to consider of the problem of singularity, i.e. when some $\tilde{M}_i^{-1}$ are singular, McMillan, et al. [39] suggested a possibly less efficient[1] algorithm. The main idea is to solve for the wrenches acted about the end points of the manipulators with insufficient degrees of freedom. To do so, $\dot{v}_o$ is first expressed in terms of the λ_i^c of

[1] As discussed in [39], under certain conditions, this algorithm is more efficient when applied for a dual-arm system.

the arms with singular $\tilde{M}_i^{-1}$. Using Equation 4.5 and Equation 4.6, the dynamic equation of the object can be written as

$$\left[I_o + \sum_{i=s+1}^{n} (G_o^{mi})^T \hat{W}_i \tilde{M}_i \hat{W}_i^T G_o^{mi}\right] \dot{v}_o + Q_o + \sum_{i=s+1}^{n} (G_o^{mi})^T P_i = -\sum_{i=1}^{s} (G_o^{mi})^T f_{mi}$$

Substitute Equation 4.3 into the above equation and simplify, we have

$$\tilde{I}_o \dot{v}_o + \tilde{Q}_o = -\sum_{i=1}^{s} (G_o^{mi})^T (\hat{W}_i \lambda_i^c + \hat{T}_i \lambda_i)$$

$$\dot{v}_o = -\tilde{I}_o^{-1} \left[\sum_{i=1}^{s} (G_o^{mi})^T (\hat{W}_i \lambda_i^c + \hat{T}_i \lambda_i) + \tilde{Q}_o\right] \tag{4.7}$$

where $\tilde{I}_o = \left[I_o + \sum_{i=s+1}^{n} (G_o^{mi})^T \hat{W}_i \tilde{M}_i \hat{W}_i^T G_o^{mi}\right]$

s is the number of singular $\tilde{M}_i^{-1}$ and the corresponding arms are assigned with numbers from 1 to s

$\tilde{I}_o$ and $\tilde{Q}_o$ are the modified values of I_o and Q_o respectively including the effect of the $(s+1)$th to nth arms

Substitute Equation 4.7 into Equation 4.4, $\sum_{i=1}^{s} N_i$ equations relating λ_i^c can be obtained:

$$\left[\tilde{M}_i^{-1} + \hat{W}_i^T G_o^{mi} \tilde{I}_o^{-1} (G_o^{mi})^T \hat{W}_i\right] \lambda_i^c$$

$$+ \hat{W}_i^T G_o^{mi} \tilde{I}_o^{-1} \sum_{j=1, j\neq i}^{s} (G_o^{mj})^T \hat{W}_j \lambda_j^c = -\hat{W}_i^T (\tilde{h}_i + M_i^{-1} \hat{T}_i \lambda_i) + a_i^c$$

$$- \hat{W}_i^T G_o^{mi} \tilde{I}_o^{-1} \left[\sum_{j=1}^{s} (G_o^{mi})^T \hat{T}_j \lambda_j + \tilde{Q}_o\right] \tag{4.8}$$

where $i = 1, 2, \ldots, s$

These equations describe the force and moment balance of the whole system. Every set of λ_i^c satisfying these equations may lead to the same motion of the corresponding arms. Therefore, the motion of the system may be solved even when these equations have infinite number of solutions. After λ_i^c are computed, $\dot{\boldsymbol{v}}_o$ and $\dot{\boldsymbol{v}}_{mi}$ can be obtained using Equation 4.7, Equation 4.5 and Equation 3.1.

The above description generalizes the approach proposed by McMillan et al. [39]. They just discussed the case where all $\hat{\boldsymbol{W}}_i$ are equal to identity matrices. The approach described above show that solving λ_i instead of $\boldsymbol{f}_{mi}$ reduces the order of the system of equations. Nevertheless, the problem due to non-constant $\hat{\boldsymbol{W}}_i$ and $\hat{\boldsymbol{T}}_i$ cannot be solved using this approach.

4.2 Modified Approach

The approaches described above can be modified for use in a more general class of problems. In general, solving the forward dynamics of a multi-body system requires to solve a system of equations in the following form [32,33]:

$$\begin{bmatrix} \boldsymbol{M} & \tilde{\boldsymbol{W}} \\ \bar{\boldsymbol{W}}^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{q} \\ \mu \end{bmatrix} = \boldsymbol{c} \tag{4.9}$$

where $\boldsymbol{M}$ is the composite inertia matrix

$\bar{\boldsymbol{W}}$ is the composite constraint space matrix which describes the kinematic constraints

$\tilde{\boldsymbol{W}}$, which may not equal to $\bar{\boldsymbol{W}}$ [1], specifies the reactions acted among the bodies

$\boldsymbol{q}$ may be the velocities or accelerations of certain set of coordinates of the system

$\boldsymbol{\mu}$ is a set of Lagrangian multipliers

---

[1] This is different from descriptions in certain textbooks [32, 33]. An example is a multi-arm robotic system manipulating an object using grippers with frictional contacts. The frictional forces may give no kinematic constraint and its magnitude depend on the corresponding normal reactions. Thus, the frictional components may be combined with the components due to the corresponding normal reactions to form a $\tilde{\boldsymbol{W}}$ different from $\bar{\boldsymbol{W}}$.

c is a vector function involved in the constraint of the velocities or accelerations

The approach described by Lilly and Orin [38] considers systems described with a model in the following form:

$$\left[\begin{array}{ccccc|cccc} M_o & & & & & \tilde{W}_{o1} & \tilde{W}_{o2} & \cdots & \tilde{W}_{on} \\ & \bar{M}_1 & & & & \tilde{W}_1 & & & \\ & & \bar{M}_2 & & & & \tilde{W}_2 & & \\ & & & \ddots & & & & \ddots & \\ & & & & \bar{M}_n & & & & \tilde{W}_n \\ \hline \bar{W}_{o1}^T & \bar{W}_1^T & & & & & & & \\ \bar{W}_{o2}^T & & \bar{W}_2^T & & & & & & \\ \vdots & & & \ddots & & & & & \\ \bar{W}_{on}^T & & & & \bar{W}_n^T & & & & \end{array}\right] \begin{bmatrix} q_o \\ q_1 \\ q_2 \\ \vdots \\ q_n \\ \mu_1 \\ \mu_2 \\ \vdots \\ \mu_n \end{bmatrix} = \begin{bmatrix} \bar{h}_o \\ \bar{h}_1 \\ \bar{h}_2 \\ \vdots \\ \bar{h}_n \\ c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix}$$

where the un-shaded areas contain elements equal to zero

This is a system having n separate bodies connected to a single body which is not necessary a rigid body.

Example 4.1

For a multi-arm system as presented in the last chapter, the corresponding matrices may be obtained as:

$$M_o = I_o \qquad \bar{M}_i = M_i$$

$$\bar{W}_{oi} = \tilde{W}_{oi} = H_o^b W_{oi} \qquad \bar{W}_i = \tilde{W}_i = H_{mi}^b W_{mi}$$

$$q_o = \dot{v}_o \qquad q_i = \dot{v}_{mi}$$

$$\mu_i = -\mu_i \qquad \bar{h}_o = -Q_o$$

$$\bar{h}_i = f_{ci} + f_{ei} - h_i \qquad c_i = -\dot{W}_{oi}^T H_b^o v_o - W_{oi}^T H_b^o \begin{bmatrix} \omega_o \times & 0_3 \\ 0_3 & \omega_o \times \end{bmatrix}^T v_o$$

$$-\dot{W}_{mi}^T H_b^{mi} v_{mi} - W_{mi}^T H_b^{mi} \begin{bmatrix} \omega_{mi} \times & 0_3 \\ 0_3 & \omega_{mi} \times \end{bmatrix}^T v_{mi} \qquad \blacksquare$$

For this class of systems, an algorithm with the same efficiency as Lilly and Orin's approach [38] can be obtained by solving $\boldsymbol{q_o}$ first. Using the kinematic constraints, we may obtain

$$\bar{W}_i^T \bar{M}_i^{-1} \tilde{W}_i \mu_i = W_i^T \bar{M}_i^{-1} \bar{h}_i - c_i + \bar{W}_{oi}^T q_o \tag{4.10}$$

where $\boldsymbol{i} = 1, 2, \ldots, \boldsymbol{n}$

When $\bar{W}_i^T \bar{M}_i^{-1} \tilde{W}_i$ is nonsingular, $\boldsymbol{\mu_i}$ can be obtained as

$$\mu_i = (\bar{W}_i^T \bar{M}_i^{-1} \tilde{W}_i)^{-1} \left[\bar{W}_i^T \bar{M}_i^{-1} \bar{h}_i - c_i + \bar{W}_{oi}^T q_o \right] \tag{4.11}$$

By eliminating $\boldsymbol{\mu_i}$, $\boldsymbol{q_o}$ can then be obtained:

$$q_o = \left[M_o + \sum_{i=1}^{n} \tilde{W}_{oi} (\bar{W}_i^T \bar{M}_i^{-1} \tilde{W}_i)^{-1} \bar{W}_{oi}^T \right]^{-1} \left[\bar{h}_o - \sum_{i=1}^{n} \tilde{W}_{oi} (\bar{W}_i^T \bar{M}_i^{-1} \tilde{W}_i)^{-1} (\bar{W}_i^T \bar{M}_i^{-1} \bar{h}_i - c_i) \right] \tag{4.12}$$

$\boldsymbol{q_o}$ can be computed using Equation 4.12 as all terms in the R.H.S. are known. Using the value $\boldsymbol{q_o}$, $\boldsymbol{\mu_i}$ and $\boldsymbol{q_i}$ can then be calculated from Equation 4.11 and the dynamic equations respectively.

There are just some minor differences between this approach and the approach by Lilly and Orin [38]. In the former, the kinematic constraint is used instead of the relative values between q_i. This arrangement avoid the determination of a_i^c. a_i^c are dependent on $\dot{T}_i$ which is much more difficult to determine than $\dot{W}_{mi}$ or $\dot{W}_{oi}$ for a general joint as shown in the last chapter. However, when T_i is constant, a_i^c is zero and there is no difference in using these two approaches. The modified approach also shows that $\bar{W}_i$ do not have to possess the orthonormal property. It can be chosen such that μ_i, or λ_i^c, have more significant physical meanings and the corresponding constraint space matrices can be determined with less difficulty. In addition, dynamic models based on variables other than accelerations [42] can also be solved using this approach. To sum up, this approach provides a more systematic and general way to solve the problem while retaining the efficiency. It also avoids complicated computation when the joints involved are not simple ones.

The efficiency of this algorithm is affected by the computations of $\bar{M}_i$, $\bar{h}_i$, $\tilde{W}_i$, $\bar{W}_i$ and c_i. In general, $\bar{M}_i$ and $\bar{h}_i$ can be determined easily for serial mechanisms using recursive algorithms [30,36,43]. Therefore, it is desirable to divide the multi-body system into serial multi-body sub-systems to solve the forward dynamic problem. To obtain $\tilde{W}_i$, $\bar{W}_i$ and c_i efficiently, we may make use of the general joint model described in the last chapter. In addition, constraint violation stabilization method [32, 33] for stabilize numerical integrations can be adopted when the general joint model is used. This is discussed in the next section.

Moreover, McMillan, Sadayappan and Orin's approach [39] can also be easily modified in the same direction to obtain an algorithm for situations when there are singular $\bar{W}_i^T\bar{M}_i^{-1}\tilde{W}_i$. First, solve q_o in terms of μ_i,

$$q_o = \tilde{M}_o^{-1}(\tilde{h}_o - \sum_{i=1}^{s} \tilde{W}_i \mu_i) \tag{4.11}$$

where $\tilde{M}_o = M_o + \sum_{i=1+s}^{n} \tilde{W}_{oi}(\bar{W}_i^T\bar{M}_i^{-1}\tilde{W}_i)^{-1}\bar{W}_{oi}^T$

$$\tilde{h}_o = \bar{h}_o - \sum_{i=s+1}^{n} \tilde{W}_{oi}(\bar{W}_i^T\bar{M}_i^{-1}\tilde{W}_i)^{-1}(\bar{W}_i^T\bar{M}_i^{-1}\bar{h}_i - c_i)$$

s is the number of the singular $\bar{W}_i^T\bar{M}_i^{-1}\tilde{W}_i$ and the corresponding bodies are assigned with numbers from 1 to s

Substituting Equation 4.11 into Equation 4.8,

$$\bar{W}_i^T\bar{M}_i^{-1}\tilde{W}_i\mu_i = W_i^T\bar{M}_i^{-1}\bar{h}_i - c_i + \bar{W}_{oi}^T\tilde{M}_o^{-1}(\tilde{h}_o - \sum_{i=1}^{s}\tilde{W}_i\mu_i) \tag{4.12}$$

where $i = 1, 2, ..., s$

Combining the equations, we have

$$\begin{bmatrix} F_{11} & F_{12} & \cdots & F_{1s} \\ F_{21} & F_{22} & \cdots & F_{2s} \\ \cdots & \cdots & \ddots & \cdots \\ F_{s1} & F_{s2} & \cdots & F_{ss} \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_s \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_s \end{bmatrix}$$

where $F_{ij} = \bar{W}_{oi}^T\tilde{M}_o^{-1}\tilde{W}_{oj}$ *for* $i \neq j$

$$F_{ii} = \bar{W}_{oi}^T\tilde{M}_o^{-1}\tilde{W}_{oi} + \bar{W}_i^T\bar{M}_i^{-1}\tilde{W}_i$$

$$d_i = W_i^T\bar{M}_i^{-1}\bar{h}_i - c_i + \bar{W}_{oi}^T\tilde{M}_o^{-1}\tilde{h}_o$$

Similar to the original algorithm, any solution of $\boldsymbol{\mu}_i$ results in the same motion. Using the dynamic equations and/or Equation 4.11, the values of $\boldsymbol{q}_o$ and $\boldsymbol{q}_i$ can be obtained.

4.3 Constraint Violation Stabilization Method

If the constraints of the accelerations are used to solve for the variables used in the simulation of a kinematically constrained system, the resulted coordinates of the system obtained from numerical integration would not satisfy the constraints on the coordinates and the error would increase with time as well as the numerical error [32]. To deal with this drift due to numerical integration, several methods are available in the literature [32]. The constraint violation stabilization method or Baumgarte Stabilization is an efficient approach [32] though it may have problem when the system is close to a singular point [34].

The constraint violation stabilization method is actually an extension of feedback control theory [33]. To apply this method, the constraint on the accelerations

$$\ddot{\boldsymbol{\phi}} = \mathbf{0}$$

is modified as

$$\ddot{\boldsymbol{\phi}} + 2\alpha\dot{\boldsymbol{\phi}} + \beta^2\boldsymbol{\phi} = \mathbf{0}$$

where $\phi(\boldsymbol{q})$ is the corresponding constraint function for the coordinates $(\boldsymbol{q})$ of the system, i.e. $\boldsymbol{\phi} = \mathbf{0}$ is the constraint equation.

$\dot{\phi}$ is the corresponding constraint function for the time derivatives of the coordinates

α and β are positive constants

According to Nikravesh [33], a range of values between 1 and 10 for α and β is adequate for most practical problems. For nonholonomic constraints, the value of β may be simply taken as zero.

To apply this method, ϕ and $\dot{\phi}$ are required, which, unfortunately, are not easy to obtain for certain cases. García de Jalón and Bayo [32] discussed that the complexity of $\dot{\phi}$ and $\ddot{\phi}$ depends on the coordinates used to describe the configuration of the system. They also suggested to use the natural coordinates, which has a linear Jacobian $\frac{\partial \phi}{\partial \boldsymbol{q}}$ for ϕ; but using these coordinates may introduce more constraints to the system thus increase the complexity of the problem. To solve this problem, it is suggested here that the approaches described in Section 3.9.2 be used to compute ϕ and $\dot{\phi}$ numerically. Following is an example showing the application of the general model developed in the last chapter.

Example 4.2

Consider the system in Example 4.1 again. In order to apply the constraint violation stabilization method, only c_i is required to be modified as:

$$\tilde{c}_i = c_i - 2\alpha(W_{mi}v_{mi}^{mi} + W_{oi}v_o^o) - \beta^2\phi_i$$

where ϕ_i is the constraints function for the coordinates of Σ_{mi} which may be computed using the approach described in Section 3.9.2 ■

Lilly and Orin [38] also suggested a method to tackle the problem. The method is based on the same concept of the constraint violation stabilization and is suitable for use with the approach they developed earlier for solving the forward dynamics. The main problem of this method is that it requires the constrained relative positions, orientations and velocities. Unless the manipulators grasp the object rigidly, these variables may not be known or difficult to be determined using the parameters obtained from numerical integration which contain errors. Thus, the modified approach described in this chapter would be preferable when used with constraint violation stabilization.

4.4 Computational Requirement of the Algorithm

The algorithm proposed in this chapter has the same computational complexity as that proposed by Lilly and Orin [38]. Consider a multi-arm system with n arms as those discussed in the last chapter. If the arms have the same number of single degree-of-freedom joint and all arms are connected to the object with joint having the same degree(s) of freedom, the computational complexity depends only the number of arm n. To solve for q_o, Equation 4.12 is used. The computational complexity of the coefficient matrices is $O(n)$ and the number of operation required in solving the linear system of equation is fixed if the degree of freedom of the object is fixed. In solving for q_i, Equation 4.11 and the dynamic equations are used. This also requires a computational complexity of $\boldsymbol{O(n)}$. If $\boldsymbol{q_o}$ and $\boldsymbol{q_i}$ are solved directly from Equation 4.9, the computational complexity will be $\boldsymbol{O(k^3)}$. However, the actual efficiency of the algorithm also depends on many other factors, e.g. the programming language and the programming method. In order to test the efficiency of the proposed algorithm, tests are performed for systems in which each arm having 6 single degree-of-freedom joint and connected to the object with a joint with single degree of freedom.

In the test, the time required to compute the accelerations $\boldsymbol{q_o}$ and $\boldsymbol{q_i}$ are compared. The computations are performed using the Matlab M-files listed in Appendix D. To compute the accelerations, two steps are required. In the first step, the coefficient matrices in Equation 4.9 are computed. As serial robots are considered, the composite body method [30,43], which is the most efficient method for computing the joint space inertia matrix and the Jacobian together, is used. In the second step, the accelerations are solved from the equations. Two different functions, msolve.m which applies the proposed approach and msolve03.m which solve the equations directly, are used for comparison. The following table shows the result of the tests.

number of arms	total speed-up	speed-up in solving the accelerations	time saved (ms)
2	0.98	0.71	-6.5
3	0.99	0.77	-7.2
4	0.99	0.88	-4.9
5	1.00	1.03	1.3
6	1.01	1.22	12.3

number of arms	total speed-up	speed-up in solving the accelerations	time saved (ms)
7	1.02	1.41	27.1
8	1.03	1.64	47.7
9	1.05	1.88	73.7
10	1.06	2.13	104.6

In the above table, the results are average values and are defined as follows:

$$\text{total speed-up} = \frac{(t_2 + t)}{(t_1 + t)}$$

$$\text{speed-up in solving the accelerations} = \frac{t_2}{t_1}$$

$$\text{time saved} = t_2 - t_1$$

where t is average time to compute the coefficient matrices

t_1 is average time used by msolve.m

t_2 is average time used by msolve03.m

There is a significant speed-up when the number of arms is large. Therefore, it is advantageous to applied the proposed approach when the number of arms in the system is large.

CHAPTER FIVE

CONCLUSION

Grasping of large object using frictionless point contacts is studied in this thesis. Approaches that help select the required contact locations in a form-closure grasp are proposed. The selection process is divided into two steps. First, six contacts, which can provide six independent wrenches, are selected. Several special selections are suggested to simplify the problem. In the second step, one or more contacts are selected to satisfy the form-closure condition. To do so, methods are introduced to determine the possible region for the last contact. Methods are also suggested to test the quality of the designed grasp. The main objective of these tests is to ensure that the form-closure property can be maintained for small displacements of the contacts. Furthermore, an algorithm is proposed to determine the object frame from the contact points of a form-closure grasp. This algorithm, however, only works for cases where the contact points are on three pairs of parallel surfaces.

A dynamic model for multi-arm systems handling one single object is derived. This model utilizes a general joint model to describe the different joining condition between the end effectors and the object. The model is based on frictionless point contacts. With this model, any type of joints may be modelled and the analysis can be generalized. All the parameters, including the constraint functions, matrices describing the relative motion and the constraint forces as well as the time derivative of these matrices, can be determined with general approaches.

Simulation of multi-arm system handling one single object is also studied in this thesis. An efficient approach is proposed to solve the forward dynamics of these multi-arm systems. This approach, modified from an existing one, is a more general and can be applied to a larger class of systems. Using the approach, computation can be simplified when complicated joints are involved. Furthermore, the application of the general model with this approach to stabilize error in numerical integration is also discussed.

With these approaches, multi-arm systems for manipulation of large objects can be designed and analyzed. However, in order to build a working system, more work has to be done in the future, which includes force distribution, grasp planning, control, etc. Some of these topics are summarized in the next section.

5.1 Future Researches

In this study, approaches are only proposed to design form-closure grasps but there is no discussion on how to select an appropriate grasp. The optimization of the grasp should be further studied. As in most studies in grasp optimization, the contact force is the first factor to be considered. For form-closure grasp provided by multi-arm system, the configuration stability as well as the distances between the contacts should also be considered.

The force distribution among the contacts is also a very important issue in the study of manipulation of large objects. In addition to the determination of quality measure for contact forces, an appropriate force distribution algorithm is essential in the control of the system. This is because the magnitudes of contact forces must be maintained to be greater than zero.

In the design of grasps, it is assumed that contacts may be made at all surfaces of the object. In real life, some of the surfaces must be in contact with supporting surfaces and the grippers may not approach the object in certain directions before the object is picked up. Special fixtures may be applied to allow contacts to be made at the required locations but this restricts the use of the system. To made the system fully automatic, certain grasp planning algorithm must be derived to pick up the object. For polyhedral objects, it may be possible to pick up the object by first tilting it.

Only contact locations are considered in the design of grasps. The real designs are not considered. Point contacts, line contacts or surface contacts may be used in the design of the gripper. Even when the type of contact is decided, the geometry of the contact surfaces may affect the kinematic properties. The kinematic properties, the pressure distribution and the complexity of the structure must be considered in the design of grippers.

Besides grasping, the control of the system is also crucial in developing the multi-arm system. Two important points should be noted in the control of such multi-arm systems. As pointed out above, an appropriate force distribution algorithm must be used with the control algorithm. The relative motion between the arms and the object must be restricted such that the grasp is maintained. However, if possible the relative motion may also be applied to increase the working envelope of the system. Thus, the study of grasping and control must be considered together in designing multi-arm system for manipulating large objects.

APPENDIX A

PROOFS AND DISCUSSIONS RELATED TO CHAPTER 2

A.1 Expression of the Inverse of a Six by Six Matrix

Any nonsingular 6×6 matrix can be expressed as

$$\begin{bmatrix} e_A & e_B \\ a_A & a_B \end{bmatrix}$$

where $\begin{bmatrix} e_A \\ a_A \end{bmatrix} = \begin{bmatrix} e_1 & e_2 & e_3 \\ a_1 & a_2 & a_3 \end{bmatrix}$

$$\begin{bmatrix} e_B \\ a_B \end{bmatrix} = \begin{bmatrix} e_4 & e_5 & e_6 \\ a_4 & a_5 & a_6 \end{bmatrix}$$

$e_1, e_2, \cdots, e_6$ are 3×1 vectors

e_1, e_2, e_3 are linearly independent

$a_1, a_2, \cdots, a_6$ are 3×1 vectors

Take $\begin{bmatrix} e_A & e_B \\ a_A & a_B \end{bmatrix}\begin{bmatrix} I_3 & T_1 \\ 0 & I_3 \end{bmatrix} = \begin{bmatrix} e_A & 0 \\ a_A & a \end{bmatrix}$

then $\quad T_1 = -e_A^{-1} e_B$

$$a = a_A T_1 + a_B$$

Take $\begin{bmatrix} e_A & 0 \\ a_A & a \end{bmatrix}\begin{bmatrix} I_3 & 0 \\ T_2 & I_3 \end{bmatrix} = \begin{bmatrix} e_A & 0 \\ 0 & a \end{bmatrix}$

then $\quad T_2 = -a^{-1} a_A$

Using the above result, we may write

$$\begin{bmatrix} e_A & e_B \\ a_A & a_B \end{bmatrix}\begin{bmatrix} I_3 & T_1 \\ 0 & I_3 \end{bmatrix}\begin{bmatrix} I_3 & 0 \\ T_2 & I_3 \end{bmatrix} = \begin{bmatrix} e_A & 0 \\ 0 & a \end{bmatrix}$$

Therefore, we have

$$\begin{bmatrix} e_A & e_B \\ a_A & a_B \end{bmatrix}^{-1} = \begin{bmatrix} I_3 & T_1 \\ 0 & I_3 \end{bmatrix} \begin{bmatrix} I_3 & 0 \\ T_2 & I_3 \end{bmatrix} \begin{bmatrix} e_A^{-1} & 0 \\ 0 & a^{-1} \end{bmatrix}$$

$$= \begin{bmatrix} (I_3 + T_1 T_2) e_A^{-1} & T_1 a^{-1} \\ T_2 e_A^{-1} & a^{-1} \end{bmatrix}$$

A.2 Necessary Conditions for Six Frictionless Contacts Providing Linearly Independent Wrenches

In Section 2.2.1, necessary conditions for obtaining six linearly independent wrenches from six frictionless point contacts are listed without proof. In order to prove these conditions, it should first point out that though the values of the wrenches change as the origin changes, the property of being linearly independent is independent of the origin selected. In addition, three of the moments provided by the contacts must be linearly independent if the wrenches are linearly independent. Thus, in order to prove a necessary condition, it is sufficient to show that three linearly independent moments cannot be provided about a certain origin when the condition is not satisfied.

A.2.1 No more than 3 contact normals intersect at one point

If four of the contact normals intersect at one point and this point is selected as the origin, four of the moments provided by the contacts is zero. There are only two non-zero moments about this point under this situation. Therefore, there cannot be four or more contacts intersect at one point.

A.2.2 No more than 3 contacts normals are parallel

Consider vectors with the following form:

$$\begin{bmatrix} \pm c \\ d \end{bmatrix} \tag{A.1}$$

where c is a constant vector with dimension greater than or equal to one

$\boldsymbol{d}$ is a vector belong to a subspace spanned by $\boldsymbol{k}$ linearly independent vectors

let $\{\boldsymbol{d}_1, \boldsymbol{d}_2, ..., \boldsymbol{d}_k\}$ be a basis of the subspace

take $\boldsymbol{d}_{k+1}$ as

$$\sum_{i=1}^{k} \gamma_i d_i$$

where γ_i are constant scalars and $\sum_{i=1}^{k} \gamma_i \neq 1$

then, we have

$$\frac{1}{1-\sum_{i=1}^{k}\gamma_i}\left\{\begin{bmatrix} c \\ d_{k+1} \end{bmatrix} - \sum_{i=1}^{k}\gamma_i \begin{bmatrix} c \\ d_i \end{bmatrix}\right\} = \begin{bmatrix} c \\ 0 \end{bmatrix}$$

Therefore, the $\boldsymbol{k+1}$ vectors

$$\begin{bmatrix} c \\ d_1 \end{bmatrix}, \begin{bmatrix} c \\ d_2 \end{bmatrix}, ..., \begin{bmatrix} c \\ d_k \end{bmatrix}, \text{ and } \begin{bmatrix} c \\ d_{k+1} \end{bmatrix}$$

forms a basis for the subspace containing all the vectors in the form specified by Equation A.1. In other words, the subspace spanned by the vectors in this form has a dimension $\boldsymbol{k+1}$.

All the wrenches provided by contacts with parallel contact normals have the form given by Equation A.1. In this case, $\boldsymbol{c}$ equal to the common contact normals. This wrenches have the following form:

$$\begin{bmatrix} \pm e \\ p_i \times e \end{bmatrix}$$

where $\boldsymbol{e}$ is the common contact normal

$\boldsymbol{p}_i$ is the contact location

As $p_i \times e$ are perpendicular to e, they belong to a subspace spanned by two linearly independent vectors. Therefore, the wrenches provided by contacts with parallel contact normals belong to a subspace spanned by three wrenches. Thus, there cannot be four linearly independent wrenches provided by contacts lie on parallel surfaces.

A.2.3 No more than 3 contact normals lie on the same plane

Consider the case when the origin also lies on the same plane. The four moments about the origin provided by the four contacts must be parallel as they are all perpendicular to this plane. Because three of the contact normals must be independent contacts, there is at least one of the six contact normals which is not parallel to the plane. There must be an intersection made by this contact normal on the plane. If this intersection is taken as the origin, the moment applied through this contact is zero. There can only be two linearly independent moments applied about this origin. As a result, no more than three contact normals can lie on the same plane.

A.2.4 Conditions when there are 3 contact normals intersect at one point

If three contact normals intersect at one point and this intersection is taken as the origin, the moments provided by these three contacts are all zero. Therefore, the moments provided by the other three contacts about this point must be linearly independent. Three conditions are listed in Section 2.2.1 under this situation:

(i) the remaining contact normals must not be all parallel;

(ii) the rest contact normals must not intersect at one point;

(iii) the intersection must not lie on the planes formed by the remaining contact normals;

As pointed out in Section A.2.2, moments provided by contacts with parallel contact normals belong to a subspace spanned by two linearly independent vectors perpendicular to the normal of the surfaces. This proves (i).

Moments provided by contacts with contact normals intersecting at one point can be expressed as

$$p \times e_i$$

where e_i is the contact normal

p is the position of the intersection

This shows that these moments should perpendicular to the vector pointing from the origin to the intersection. Therefore, no more than two of such moments can be linearly independent. This proves (ii).

When the origin lies on the plane formed by two contact normals, the moments about the origin provided by the two contacts are parallel as they must be perpendicular to this plane and cannot be linearly independent. This proves (iii).

A.2.5 If 3 contact normals are parallel, the other 3 contact normals must not intersect at one point

The proof is the same as case (i) in Section A.2.4.

A.2.6 If 3 contact normals lie on the same plane, intersection(s) of the remaining contact normals must not lie on that plane.

This is similar to case (iii) in Section A.2.3. As before, if the origin lie on the plane formed by the three normals, the three corresponding moments are parallel. Therefore, if there is an intersection of the remaining contact normal lie on this plane, there can be at most two linearly independent moments when the intersection is taken as the origin.

A.3 Relationship between Contact Location on a Plane Surface and the Moment that can be Applied through the Contact

With this relationship, contacts can be selected to obtain the required moment. This is crucial in designing form-closure grasps for polyhedral objects. Figure A.1 shows a contact normal e_i and the corresponding position vector p_i of the contact. This

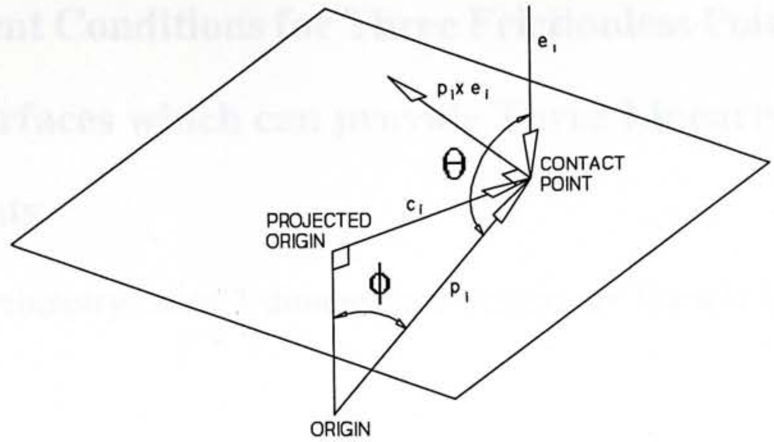

Figure A.1 Relationship between contact point on a flat surface and the corresponding moment about the origin

figure is general in the sense that no matter which point is taken as the origin, it is possible to obtain a view like this by looking from either side of the plane formed by the contact normal and the position vector. The moment that can be applied through this contact point about the origin is given by $\boldsymbol{p}_i \times \boldsymbol{e}_i$. $\boldsymbol{p}_i \times \boldsymbol{e}_i$ is a free vector whose magnitude is given as:

$$|\boldsymbol{p}_i| \sin\theta$$

where θ is the angle between the contact normal and the position vector as shown in Figure A.1

It can be noted that:

$$\sin\theta = \sin(180° - \theta) = \sin\phi$$

In fact, $|\boldsymbol{p}_i| \sin\phi$ is the length of the projection of $\boldsymbol{p}_i$ on the plane or the distance between the projected origin and the contact point. Therefore, the vector $\boldsymbol{c}_i$ stretching from the projected origin to the contact point has the same magnitude as that of $\boldsymbol{p}_i \times \boldsymbol{e}_i$. As $\boldsymbol{p}_i \times \boldsymbol{e}_i$ is perpendicular to the plane formed by $\boldsymbol{e}_i$ and $\boldsymbol{p}_i$ while $\boldsymbol{c}_i$ lies on this plane $\boldsymbol{p}_i \times \boldsymbol{e}_i$ can be obtained from $\boldsymbol{c}_i$ by rotating it through -90° about $\boldsymbol{e}_i$ as shown in Figure A.1.

A.4 Sufficient Conditions for Three Frictionless Point Contacts on Flat Surfaces which can provide Three Linearly Independent Moments

In terms of 3D geometry, three 3-dimensional vectors are linearly independent if and only if:

(i) none of the vectors has all of the elements equal to zero

(ii) none of the vectors are parallel

(iii) not all of the vectors are parallel to the same plane

In order to obtain the sufficient conditions for three frictionless contacts on flat surfaces which provide three linearly independent moments, these conditions together with the relationship discussed in the last section can be applied. The sufficient conditions are listed in Section 2.2.1 as:

1. The contacts are not coincide with the orthogonal projection of the origin on the corresponding surfaces;
2. For any two contacts on two non-parallel surfaces, the two lines joining the contact points and the projected origins on the corresponding surfaces must not both perpendicular to the intersecting line of the surfaces.
3. If two contacts lie on the same plane, the projected origin must not lie on the line joining the contact points.
4. The contact normals are not intersect at one point.
5. The contacts are not all lie on parallel surfaces.
6. For each contact, the line joining the contact and the projected origin must not perpendicular to the intersecting line (CD) of the plane on which the contact lies and a surface (PLANE A) parallel to both of the other two $\boldsymbol{p}_i \times \boldsymbol{e}_i$. (The planes and the lines are shown in Figure A.2.)

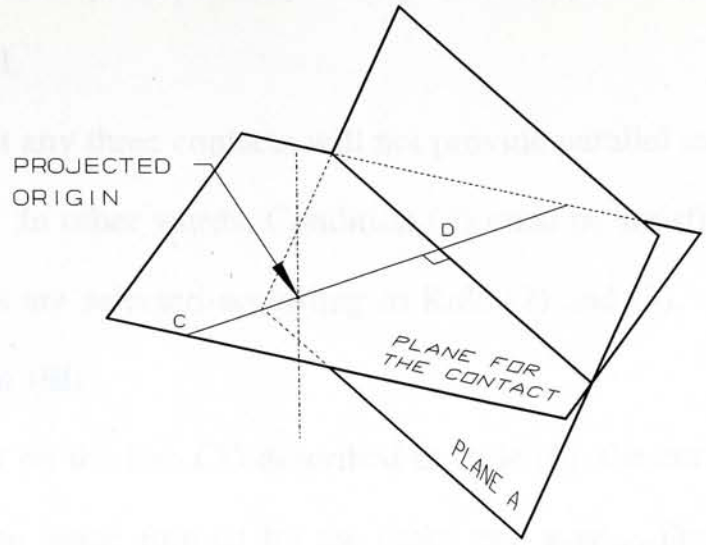

Figure A.2 Lines and planes described in Rule (6)

Rule (1) is required to satisfy Condition (i) above, Rules (2) and (3) are required to satisfy Condition (ii) and Rules (4) to (6) are required to satisfy Condition (iii). The following sub-sections show that why the conditions are satisfied when the rules are obeyed.

A.4.1 Condition (i)

It is simple that if all contacts are not coincide with the corresponding projected origins, none of the moments will be zero vector. Therefore, Condition (1) must be satisfied by the moments if the contacts are selected according to Rule (1).

A.4.2 Condition (ii)

The moments provided by two contacts lie on two non-parallel flat surfaces can only be parallel to the each other if they both parallel to the intersecting line of the surfaces. This is because they should be parallel to the surface on which the corresponding contacts lie. Thus, according to the relationship discussed in last section, any two moments provided by two contacts on two non-parallel flat surfaces cannot be parallel if they satisfy Rule (2).

Consider again the relationship between a contact point and the corresponding $p_i \times e_i$. It is obvious that two moments provided by two contacts on the same plane are parallel if the origin lie on the line joining the two contacts. Therefore, for two contacts on the same plane, if they satisfy Rule (3), the two corresponding $\boldsymbol{p}_i \times \boldsymbol{e}_i$ cannot be parallel.

As a result any three contacts will not provide parallel moments if they satisfy Rule (2) and (3). In other words, Condition (ii) must be satisfied by the moments if the three contacts are selected according to Rule (2) and (3).

A.4.3 Condition (iii)

If a contact is not on the line CD described in Rule (6), the corresponding $\boldsymbol{p}_i \times \boldsymbol{e}_i$ must not parallel to the plane formed by the other two $\boldsymbol{p}_i \times \boldsymbol{e}_i$. Thus, if the contacts are selected according to Rule (6), Condition (iii) must be satisfied by the three moments. Rules (4) and (5) corresponds to (ii) and (i) in Section A.2.4 respectively. Actually, they are special case of Rule (6) but these rules give more obvious physical interpretation.

As a conclusion, the moments provided by three contacts selected according Rules (1) - (6) are linearly independent because they must satisfy the sufficient conditions for three 3-dimensional vectors to be linearly independent, i.e. Conditions (i), (ii) and (iii).

A.5 Six Contacts Selected According to Case III in Section 2.2.1 are Independent Contacts

Take $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ as the contact normals which are intersecting with each other

$\boldsymbol{e}_3$, $\boldsymbol{e}_4$ and $\boldsymbol{e}_5$ are contact normals of contacts lying on the same plane

$\boldsymbol{e}_6$ is the contact normal of the last contact

There are three linearly independent contact normals. Therefore, the six contacts are independent ones if $\boldsymbol{a}$ is nonsingular. There two different cases to be considered. First, if $\boldsymbol{e}_1$, $\boldsymbol{e}_2$ and $\boldsymbol{e}_6$ are linearly independent and $\boldsymbol{e}_A$ is selected as $[\boldsymbol{e}_1\ \boldsymbol{e}_2\ \boldsymbol{e}_6]$, $\boldsymbol{a}$ is given as

$$\left[p_3{\times}e_3-t(p_6{\times}e_6)\;\; p_4{\times}e_4-t(p_6{\times}e_6)\;\; p_5{\times}e_5-t(p_6{\times}e_6)\right]$$

where t is the third element of $e_A^{-1}e_3$

Because $p_3{\times}e_3$, $p_4{\times}e_4$ and $p_5{\times}e_5$ lie on the same plane, this matrix can be reduced to the following form using elementary column operations:

$$\left[p_3{\times}e_3-p_5{\times}e_5\;\; p_4{\times}e_4-p_5{\times}e_5\;\; p_6{\times}e_6\right]$$

Because of the condition in the selection of the last contact, $p_6{\times}e_6$ is not parallel to the plane parallel to $p_3{\times}e_3$, $p_4{\times}e_4$ and $p_5{\times}e_5$. Its elements are also not all equal to zero. Thus, a is nonsingular and the six corresponding wrenches are linearly independent.

If e_1, e_2 and e_6 are not linearly independent, e_A may be selected as $[e_1\ e_2\ e_3]$. In this case, a can be obtained as

$$\left[p_4{\times}e_4-(p_3{\times}e_3)\;\; p_5{\times}e_5-(p_3{\times}e_3)\;\; p_6{\times}e_6-t_1(p_3{\times}e_3)\right]$$

where t_1 is the third element of $e_A^{-1}e_6$

Again, this matrix can be reduce using elementary column operations as follows:

$$\left[p_4{\times}e_4-(p_3{\times}e_3)\;\; p_5{\times}e_5-(p_3{\times}e_3)\;\; p_6{\times}e_6\right]$$

With the same argument as above, a in this case is also nonsingular and the six corresponding wrenches are linearly independent. Thus, contacts selected according to Case III in Section 2.2.1 are independent contacts.

A.6 Region for a Contact on a Curved Surface in a Form-closure Grasp

Nguyen [3] pointed out that the region for a contact in a form-closure grasp must:

1. be either flat or spherical, and
2. have a convex boundary.

These restricted that the regions must on a flat or spherical surface. However, this is not true and regions of contact in a form-closure grasp may be on a curved surface.

To prove the above conditions, two conditions for the wrenches applied through frictionless contacts are used in [3]. First, the affine combination of two wrenches applied through two contacts inside the same region is also a wrench that can be applied through a point inside the same region, i.e.

$$\gamma\begin{bmatrix} e_1 \\ p_1 \times e_1 \end{bmatrix} + (1-\gamma)\begin{bmatrix} e_1 \\ p_2 \times e_2 \end{bmatrix} = \begin{bmatrix} e \\ p \times e \end{bmatrix} \tag{A.3}$$

where $\boldsymbol{p}$, $\boldsymbol{p_1}$ and $\boldsymbol{p_2}$ are contacts in the same contact region

$\boldsymbol{e}$, $\boldsymbol{e_1}$ and $\boldsymbol{e_2}$ are contact normals

$0 \leq \gamma \leq 1$

Secondly, the force component and the moment component of each of these zero-pitch wrenches are perpendicular, therefore

$$(p_1 \times e_1)^T e_1 = 0, \quad (p_2 \times e_2)^T e_2 = 0, \quad (p \times e)^T e = 0 \tag{A.4}$$

Using these two conditions, the following was obtained:

$$(p_1 \times e_1)^T e_2 + (p_2 \times e_2)^T e_1 = 0 \tag{A.5}$$

$\boldsymbol{e_1}$ and $\boldsymbol{e_2}$ must intersect or be parallel in order to satisfy Equation A.5. By extrapolate this condition to other points in the region, the conditions for the region of contact were concluded. The main problem of this proof is that Equation A.3 is not always true. Though any affine combination of two wrenches applied through contacts in the same contact region is a wrench which satisfy the form-closure condition with the other wrenches in the grasp, its pitch may not be zero. Therefore, there is no requirement on the contact region that it must be either flat or spherical. It may lie on certain curved surfaces which is not spherical. As contacts on a flat or spherical surface satisfy Equation A.3, it may just be concluded that contact regions on flat and spherical surfaces have convex boundaries.

A.7 The Class of Polyhedral Object Discussed in Section 2.2.3.3

This section discuss the class of polyhedron which has the property that there are two or more surfaces whose normals are negative combinations of the normals of three other surfaces.

For any object, there should be four to six surfaces whose contact normals are sufficient to satisfy Equation 2.1, i.e.

$$\sum_{i=1}^{n} \alpha_i e_i = 0_{3\times1} \tag{A.2}$$

where e_i are contact normals

$\alpha_i > 0$

$n = 4, 5, \text{ or } 6$

Therefore, for polyhedral objects, there should be four to six surfaces whose normals satisfy Equation A.2, If there is no parallel surfaces, four normals must be enough to satisfy Equation A.2. Otherwise, one or three pairs of parallel normals may be involved.

Consider polyhedral objects with more than four surfaces. When four normals are sufficient to satisfy Equation A.2, it is obvious that any three of these four are linearly independent. Therefore, any normal e may be written in the form:

$$\sum_{i=1}^{3} \gamma_i e_i - \beta \sum_{i=1}^{4} \alpha_i e_i$$

where e_1, e_2, e_3, and e_4 are four contact normals satisfies Equation A.2

γ_i are real numbers and is given as

$$\begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{bmatrix} = \begin{bmatrix} e_1 & e_2 & e_3 \end{bmatrix}^{-1} e$$

β is a positive real number

If β equals the largest γ_i/α_i, at least one of the coefficients of e_i is zero and the other three are negative. This demonstrates that e can be equal to negative combination of three of the four vectors satisfy Equation A.2. The contact normal e_i corresponding to the largest γ_i/α_i can also be expressed as a negative combination of the same three contact normals. It is because any one of the four normals satisfying Equation A.2 can be expressed as strictly negative combination of the other three. Thus, for this kind of polyhedron there must be at least two surfaces whose normals are negative combinations of normals of another three surfaces.

When at least five normals are required to satisfy Equation A.2, two of these normals must be parallel and opposite. The other three must be perpendicular to these two and can be added to zero with positive coefficients. Therefore,

$$e_1 = -e_2$$

$$\sum_{i=3}^{5} \gamma_i e_i = 0_{3\times 1}$$

where e_1 and e_2 are the two parallel and opposite normals

e_3, e_4 and e_5 are the rest normals

Any normal can be divided into two parts such that one is parallel to the two parallel ones and the other is perpendicular to them. The first part can then be expressed as a negative multiple of either one of the two parallel normals. Using similar argument as in the last case, it can be concluded that the other part can be expressed as negative combination of two of the other three normals. As a result, any normal e can be expressed as negative combination of three e_i including one of the two parallel ones. Moreover, the two remaining e_i can both be expressed as negative combinations of other three. Thus, there must be at least two surfaces whose normals are negative combinations of normals of three other surfaces; and there must be at least three if the object have more than five surfaces.

In the case where at least six normals are required to satisfy Equation A.2, there must be only three pairs of surfaces with parallel and opposite normals. Assume $\boldsymbol{e}_1$ is parallel to $\boldsymbol{e}_2$, $\boldsymbol{e}_3$ is parallel to $\boldsymbol{e}_4$, and $\boldsymbol{e}_5$ is parallel to $\boldsymbol{e}_6$. Then $\boldsymbol{e}_1$, $\boldsymbol{e}_3$ and $\boldsymbol{e}_4$ must be linearly independent. As $\boldsymbol{e}_2 = -\boldsymbol{e}_1$, $\boldsymbol{e}_4 = -\boldsymbol{e}_3$ and $\boldsymbol{e}_6 = -\boldsymbol{e}_5$, it is obvious that any normal can be expressed as negative combination of three of the five normals. Therefore, there are three surfaces whose normals can be expressed as negative combination of normals of three other surfaces.

In conclusion, any polyhedral object with more than four surfaces have the property that it has at least two surfaces whose normals are negative combinations of the normals of three other surfaces.

APPENDIX B

IMPLEMENTATION OF THE ALGORITHM FOR DETERMINING THE OBJECT FRAME FROM A FORM-CLOSURE GRASP

The following are Matlab M-files implement the algorithm described in Section 2.4:

```
% function [n,po] = objfrm(p,R2,R3,l1,l2,l3)
%
% This function computes the object frame of an object with three
% parallel planes from the contact points of a form-closure grasp
%
% p            a vector containing all the contact points
% R2, R3       rotation matrices which rotate the surface normals to
%              align with the axes of the reference frame
% l1,l2,l3     distances between parallel planes

% The variables are described with more details in Section 2.4
% of the thesis.

function [n,po] = objfrm(p,R2,R3,l1,l2,l3)

% start computation
d1 = p(:,1)-p(:,4);     d2 = p(:,7)-p(:,4);
d3 = p(:,2)-p(:,5);     d4 = p(:,3)-p(:,6);
A = (d2(3)-d1(3))*l1;  B = d2(1)*d1(3) - d1(1)*d2(3);
C = (d1(2)-d2(2))*l1;          D = d1(1)*d2(2) - d2(1)*d1(2);
E = d1(2)*d2(3) - d1(3)*d2(2);

[n1x1 n1x2] = quadrat(B^2+D^2+E^2, 2*(A*B+C*D),A^2+C^2-E^2);
n1=[n1x1 n1x2; (n1x1*B+A)/E (n1x2*B+A)/E; (n1x1*D+C)/E (n1x2*D+C)/E];

R2n21 =[d3'*R2';n1(:,1)';d4'*R3'*cross(n1(:,1)')]\[l2;0;l3];
R2n22 =[d3'*R2';n1(:,2)';d4'*R3'*cross(n1(:,2)')]\[l2;0;l3];

if abs(R2n21'*R2n21-1)<1e-10
       n = [n1(:,1) R2n21 cross(n1(:,1),R2n21)];
elseif abs(R2n22'*R2n22-1)<1e-10
       n = [n1(:,2) R2n22 cross(n1(:,2),R2n22)];
elseif abs(R2n21'*R2n21-1) < abs(R2n22'*R2n22-1)
       n = [n1(:,1) R2n21 cross(n1(:,1),R2n21)];
else
       n = [n1(:,2) R2n22 cross(n1(:,2),R2n22)];
end

po = [n(:,1)';n(:,2)'*R2;n(:,3)'*R3]\[n(:,1)'*p(:,1)-l1/2; n(:,2)'*R2*p(:,2)-l2/2;...
       n(:,3)'*R3*p(:,3)-l3/2];



% function [x1, x2]=quadratic (a, b, c)
%
% This function solves quadratic equations
%

function [x1, x2]=quadratic (a, b, c)
d =sqrt( b^2-4*a*c);
x1 = (-b+d)/2/a;
x2 = (-b-d)/2/a;
```

APPENDIX C

EXPRESSING WRENCHES WITH ZERO-PITCH WRENCHES

In Section 3.9, it is pointed out that any wrench can be expressed as a zero-pitch wrenches or the difference of two zero-pitch wrenches. Consider the force component of a wrench. It is obvious that any non-zero force component $\tilde{f}$ of a wrench can be expressed as the difference of two unit vectors as follows:

$$\tilde{f} = \gamma(e_1 - e_2)$$

where e_1 and e_2 are two different unit vectors

γ is a real number

Zero-pitch wrenches with e_i's as force components are in the following form:

$$\begin{bmatrix} e_i \\ p_i \times e_i \end{bmatrix}$$

where p_i is a point on the line of action of the force

i = 1 or 2

The difference of these two zero-pitch wrenches is equal to

$$\begin{bmatrix} e_1 - e_2 \\ p_1 \times e_1 - p_2 \times e_2 \end{bmatrix}$$

As $p_i \times e_i$ is a vector perpendicular to e_i and the two e_i are non-parallel, $p_1 \times e_1$ and $p_2 \times e_2$ lie on two non-parallel planes and their difference can be equal to any 3×1 vector. Therefore,

$$\gamma \left\{ \begin{bmatrix} e_1 \\ p_1 \times e_1 \end{bmatrix} - \begin{bmatrix} e_2 \\ p_2 \times e_2 \end{bmatrix} \right\} = \begin{bmatrix} \tilde{f} \\ \tilde{m} \end{bmatrix}$$

where $\tilde{m}$ is any moment about the origin

When $\tilde{f}$ is equal to $\mathbf{0}_{3\times 1}$, e_1 and e_2 must be the same. In this case, it is still possible to express the wrench as the difference of two zero-pitch wrenches but $\boldsymbol{e_1}$ and $\boldsymbol{e_2}$ must be selected such that they are perpendicular to $\tilde{\boldsymbol{m}}$. This completes the proof for the fact that any wrenches can be expressed as a difference of two zero-pitch wrenches.

APPENDIX D

IMPLEMENTATION OF THE PROPOSED SIMULATION ALGORITHM

This appendix presents an implementation of the simulation algorithm proposed in Chapter 4 using Matlab M-files. The functions are tested using Matlab version 4.2a run on SUN workstation.

D.1 massmat.m

```
% function [M,J,sd,h,GT0m] = massmat (dyn, I, q, qd, g)
%
% This function compute the inertia matrix (M), the Jacobian with respect to
% end effector frame (J), product of derivative of Jacobian and joint velocity (sd),
% the sum of coriolis terms and gravitational terms (h), and the transformation
% matrix which transform wrenches about end effector frame to base frame (GT0m).
% Composite body method is used in computing the inertia matrix.
%
% dyn           arm description matrix
% I             inertias of the links with respect to the corresponding frames
% q             joint variables
% qd            joint velocities
% g             unit vector expressed in frame 0 pointing vertically upward
%               (gravity is taken as 9.81 m^2/s)

function [M,J,sd,h,GT0m] = massmat (dyn, I, q, qd, g)

n=size(dyn,1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% update the current pose of the system %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=1:n
        if dyn(i,5)==0 % revolute joint
                dyn(i,3) = q(i);
        else
                dyn(i,4) = q(i);
        end
end

%%%%%%%%%%%%%%%%%%%%%
% forward recursion %
%%%%%%%%%%%%%%%%%%%%%

% initialize variables
G = zeros(6*n,6);
w = zeros(3,n);
z = [0 0 1]';
hj = zeros(6,n);

G(1:6,:) = dhstran(dyn(1,1),dyn(1,2),dyn(1,3),dyn(1,4));
if dyn(1,5)==0  % revolute joint
        w(:,1) = G(4:6,4:6)*z*qd(1);

else            % prismatic joint
        w(:,1) = zeros(3,1);
end
sdtemp = G(1:6,:)*[9.81*g;zeros(3,1)];     % acceleration corresponding to gravity
sd = sdtemp;
wo = w(:,1);
hj(:,1) = [I(1,1)*cross(wo,cross(wo,dyn(1,7:9)));cross(wo,I(4:6,4:6)*wo)];
```

```
for i = 2:n
        G(6*(i-1)+1:6*i,:) = dhstran(dyn(i,1),dyn(i,2),dyn(i,3),dyn(i,4));
        if dyn(1,5)==0  % revolute joint
                % compute accelerations
                w(:,i) = G(6*(i-1)+4:6*i,4:6)*(wo+z*qd(i));
                sd = G(6*(i-1)+1:6*i,:)*(sd+[zeros(3,1);cross(wo, z*qd(i))])+...
                        [cross(w(:,i),G(6*(i-1)+1:6*(i-1)+3,4:6)*(wo+z*qd(i)));...
                        zeros(3,1)];
                sdtemp = G(6*(i-1)+1:6*i,:)*sdtemp;
                wo = w(:,i);

        else    % prismatic joint
                %compute accelerations
                w(:,i) = G(6*(i-1)+4:6*i,4:6)*wo;
                sd = G(6*(i-1)+1:6*i,:)*(sd + [2*cross(wo,z*qd(i));zeros(3,1)])+...
                        [cross(w(:,i),G(6*(i-1)+1:6*(i-1)+3,4:6)*wo);zeros(3,1)];
                sdtemp = G(6*(i-1)+1:6*i,:)*sdtemp;
                wo = w(:,i);
        end
        hj(:,i) = [I(6*(i-1)+1,1)*cross(wo,cross(wo,dyn(i,7:9)));cross(wo,...
                        I(6*(i-1)+4:6*i,4:6)*wo)]+I(6*(i-1)+1:6*i,:)*sd;
end
%%%%%%%%%%%%%%%%%%%%%%
% backward recursion %
%%%%%%%%%%%%%%%%%%%%%%

% initialize variables
J = zeros(6,n);
Kphi = zeros(6,n);
h = zeros(6,1);

% compute h and K for the nth link
K = I(6*(n-1)+1:6*n,:);
htemp = hj(:,n);
G1 = G(6*(n-1)+1:6*n,:);
GT0m = G1';
if dyn(i,5)==0  % revolute joint
        J(:,n) = G1(:,6);
else            % prismatic joint
        J(:,n) = G1(:,3);
end
Kphi(:,n) = K*J(:,n);
h(n) = J(:,n)'*htemp;

% compute h and K for the rest links
for i=n-1:-1:1
        K = G1'*K*G1 + I(6*(i-1)+1:6*i,:);
        htemp = G1'*htemp + hj(:,i);
        G1 = G(6*(i-1)+1:6*i,:);
        GT0m = G1'*GT0m;
        % compute the allowed twist corresponding to the ith link with respect
        % to the ith frame
        if dyn(i,5)==0  % revolute joint
                phi = G1(:,6);
        else            % prismatic joint
                phi = G1(:,3);
        end
        Kphi(:,i) = K*phi;
        h(i) = phi'*htemp;
        for j = i:n-1
                M(i,j) = phi'*Kphi(:,j);
                M(j,i) = M(i,j);
                phi = G(6*j+1:6*(j+1),:)*phi;
        end
        M(i,n) = phi'*Kphi(:,n);
        M(n,i) = M(i,n);
        J(:,i) = phi;
end
% remove the acceleration corresponding to gravity introduced for simpler
% computation
sd = sd - sdtemp;
```

D.2 msys01i.m

```
% function [t, r, q, rd, qd] = MSys01I(m, n, N, SYSTEM, ARM, I, JOINT, OBJECT, x0, ...
%                               p0, q0, v0, qd0, Control, Alpha, Beta, tspan)
%
% This function Integrate the forward dynamic equations of a Multi-body SYStem
% with one single node.  This function is designed to be used with functions
```

```
% MModel01 & MSys01.
%
% m             number of arms
% n             number of joint of the each arm
% N             number of constraints between the arm and the object
% SYSTEM        system description matrix
% ARM           arm configuration description matrix
% I             arm inertia properties matrix
% JOINT         description matrix of the joints at which the arms connected
%               to the object
% OBJECT        object description matrix (equal to object inertia matrix
%               about object frame)
% x             object position with respect to system base frame
% p             Euler parameters describing orientation of object frame with
%               respect to
%               system base frame
% q             joint variables of the arms
% v             twist of the object frame relative to and expressed in system
%               base frame
% qd            joint velocities of the arms
% Control       a string contains the function computes the control torque
% Alpha         gain of feedback of constraints of coordinates
% Beta          gain of feedback of velocity constraint
% tspan         time span of the integration ([t0 Tfinal])
%

function [t, r, q, rd, qd] = MSys01I(m, n, N, SYSTEM, ARM, I, JOINT, OBJECT, x0, ...
                              p0, q0, v0, qd0, Control, Alpha, Beta, tspan)


totaln = sum(n);
totalN = sum(N);

global SYSTEM ARM I JOINT OBJECT m n N totaln totalN Alpha Beta Control

%%%%%%%%%%%%%%%%%%%%%%%%%
% perform integration %
%%%%%%%%%%%%%%%%%%%%%%%%%

% In this step, other integration function may/should be used.  It depends
% the problem to be solved.

[t y] = ode45 ('msys01', tspan, [x0; p0; q0; v0; qd0]);

r = y(:,1:6);
q = y(:,7:6+totaln);
v = y(:,7+totaln:12+totaln);
qd = y(:,13+totaln:12+2*totaln);
```

D.3 msys01.m

```
% function yd = MSys01(t,y)
%
% This function returns the derivatives of the coordinates and velocities (yd) of
% a Multi-body SYStem with one single node.  This function is designed to be used
% with functions MModel01 & MSys01I.
%

function yd = msys01(t,y)

global SYSTEM ARM I JOINT OBJECT m n N totaln totalN Control

p = y(4:6);
p = [sqrt(1-p'*p);p];
v = y(7+totaln:12+totaln);
qd = y(13+totaln:12+2*totaln);

[Mo, Mi, ho, hi, Woibar, Wmibar, ci, G] = mmodel01 (y(1:3), p, y(7:6+totaln),v, qd);

[vd, qdd] = msolve(m, n, N, Mo, Mi, -ho, feval(Control,t,y)-hi, Woibar, Wmibar, ...
                  Woibar, Wmibar, ci);

pd =0.5*G'*y(10+totaln:12+totaln);

yd = [v(1:3);pd(2:4);qd;vd;qdd];
```

D.4 mmodel01.m

```
% function [Mo, Mi, ho, hi, Woibar, Wmibar, ci, G] = MModel01 (x, p, q, v, qd)
%
% This function compute the matrices in the dynamic MODEL of a Multi-arm system
% manipulating a single object.  In this model, the joints between the object
% and the arms have constant normals relative to either the object or the arms.
%
%
% note: gravity is taken as 9.81 m/s^2
%

% This function assume either emij or eoij is constant.  If emij is constant, the
% corresponding eoij is selected so that it is constant at every time step.  This
% is only suitable for a model which has one circular surface, e.g. the model for
% revolute joint or spherical joint.

function [Mo, Mi, ho, hi, Woibar, Wmibar, ci, G] = MModel01 (x, p, q, v, qd)

global SYSTEM ARM I JOINT OBJECT m n N totaln totalN Alpha Beta

% initialize the variables
Mi = zeros(totaln, max(n));             % inertia matrices of arms
hi = zeros(totaln,1);                   % Coriolus and gravitational terms of arms
GT0m = zeros(6*m,6);                    % transformation matrices
Wmibar = zeros(max(n), totalN);         % constraint spaces
Woibar = zeros(6, totalN);
ci = zeros(totalN,1);


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Compute object inertia matrix (Mo) and Coriolus and gravitational terms (ho) %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% compute object position and orientation
[Rob, G, Gob] = eulerpar(x,p);

Hob = [Rob zeros(3); zeros(3) Rob];
Mo =  Hob * OBJECT * Hob';
ho = [-OBJECT(1,1)*(cross(v(4:6),v(1:3)) + 9.81*Rob(1,:)');
       cross(v(4:6),Mo(4:6,4:6)*v(4:6))];                       % g taken as 9.81


% By-products: G0mi, J and Jdqd)

nisum = 0;
Nisum = 0;

for i = 1:m
        ni = n(i);
        Ni = N(i);
        cemi = JOINT(i,8);     % number of constant contact normals relative to gripper

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% compute arm inertia matrices (Mi) and coriolus and gravitational terms (hi) %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        [Mi(nisum+1:nisum+ni,1:ni), J, Jdqd, hi(nisum+1:nisum+ni), ...
        GT0m(6*(i-1)+1:6*i,:)] = ...
                massmat (ARM(nisum+1:nisum+ni,:), I(6*nisum+1:6*(nisum+ni),:), ...
                q(nisum+1:nisum+ni), qd(nisum+1:nisum+ni),SYSTEM(6*(i-1)+1,1:3)');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Compute the constraint space of the joints (Wmi & Woi) %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        % compute transformation matrices
        Gom = GT0m(6*(i-1)+1:6*i,:)'*inv(SYSTEM(6*(i-1)+1:6*i,1:6))*Gob;
        Rom = Gom(1:3,1:3);
        mxomcross = Gom(1:3,4:6)*Rom';
        mxom = mxomcross([6 7 2])';

        % compute contact position and contact normal
        pmij = JOINT(Nisum+1:Nisum+Ni,1:3)';
        poij = Rom'*(pmij-mxom(:,ones(Ni,1)));
        emij = [JOINT(Nisum+1:Nisum+cemi,4:6)'-Rom*JOINT(Nisum+cemi+1:Nisum+Ni,4:6)'];
        eoij = -Rom'*emij;

        Wmi = [emi; cross(pmi,emi)];
        Woi = [eoi; cross(poi,eoi)];
        Wmibar(1:ni,Nisum+1:Nisum+Ni) = J' * Wmi;
        Woibar(:,Nisum+1:Nisum+Ni) = Hob * Woi;
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Compute the terms in the acceleration constraints (ci) %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        % compute the relative velocities
        ovmimi = J*qd(nisum+1:nisum+ni) - Gom*inv(Gob)*v;

        % compute derivative of contact points and contact normals
        % constant contact point of grippers and constant contact normals of
        % object are assumed
        poijdot = Rom'*(cross(ovmimi(4:6,ones(Ni,1)),poij) + ovmimi(1:3,ones(Ni,1)));
        pmijdot = zeros(3,Ni);
        if cemi == Ni
                emijdot = zeros(3,cemi);
        else
                emijdot = [zeros(3,cemi),...
                                -cross(ovmimi(4:6,ones(Ni-cemi,1)),emij(:,cemi+1:Ni))];
        end

        if cemi == 0
                eoijdot = zeros(3,Ni);
        else
                eoijdot = [Rom'*cross(ovmimi(4:6,ones(cemi,1)),emij(:,1:cemi)),...
                                zeros(3,Ni-cemi)];
        end

        % compute derivative of Wmi and Woi
        Wmidot = [emijdot; cross(pmijdot, emij) + cross(pmij, emijdot)];
        Woidot = [eoijdot; cross(poijdot, eoij) + cross(poij, eoijdot)];

        % compute ci
        doo = [diag(eoij(:,1:cemi)'*JOINT(Nisum+1:Nisum+cemi,9:11)'); ...
                JOINT(Nisum+cemi+1:Nisum+Ni,9)];

        ci(Nisum+1:Nisum+Ni)= - Wmidot'*J*qd(nisum+1:nisum+ni) - Wmi'*Jdqd - ...
                                Woidot'*Hob'*v - Woibar(:,Nisum+1:Nisum+Ni)'*...
                                [cross(v(4:6),v(1:3)); cross(v(4:6),v(4:6))]-...
                                2*Alpha*(Woibar(:,Nisum+1:Nisum+Ni)'*v+Wmibar(1:ni,...
                                        Nisum+1:Nisum+Ni)'*qd(nisum+1:nisum+ni))-...
                                Beta^2*(diag(eoij'*poij)-doo);


        nisum = nisum + ni;
        Nisum = Nisum + Ni;

end
```

D.5 msolve.m

```
% function [qo,q] = MSolve(m, n, N, Mo, Mi, hobar, hibar,...
%                               Woibar, Wibar, Woitilde, Witilde, ci)
%
% This function SOLVE the variables (velocities or accelerations) for simulation of
% a Multi-body system with one single node using the proposed approach.
%
% m     number of arms
% n     number of joint of the each arm
% N     number of constraints between the arm and the object
%

function [qo,q] = MSolve(m, n, N, Mo, Mi, hobar, hibar,...
                                Woibar, Wibar, Woitilde, Witilde, ci)

totaln = sum(n);
totalN = sum(N);

% initialize the variables
Minv = zeros(totaln, max(n));
WMW = zeros(totalN, max(N));            % (Wibar^T Mi^-1 Witilde)^-1
WMhc = zeros(max(N), m);                % Wibar^T Mi^-1 hibar - ci
a = zeros(6,1);
b = zeros(6);

nisum = 0;
Nisum = 0;

for i = 1:m
        ni = n(i);
        Ni = N(i);

        Minv(nisum+1:nisum+ni,1:ni) = inv(Mi(nisum+1:nisum+ni,1:ni));
```

```
        WM = Wibar(1:ni, Nisum+1:Nisum+Ni)'*Minv(nisum+1:nisum+ni,1:ni);
        WMW(Nisum+1:Nisum+Ni, 1:Ni) = inv(WM*Witilde(1:ni, Nisum+1:Nisum+Ni));
        WoWMW = Woitilde(:,Nisum+1:Nisum+Ni)*WMW(Nisum+1:Nisum+Ni, 1:Ni);
        WMhc(1:Ni,i) = WM * hibar(nisum+1:nisum+ni) - ci(Nisum+1:Nisum+Ni);
        a = a + WoWMW * WMhc(1:Ni,i);
        b = b + WoWMW * Woibar(:,Nisum+1:Nisum+Ni)';

        nisum = nisum+ni;
        Nisum = Nisum+Ni;
end

qo = (Mo + b)\(hobar - a);

q = zeros(totaln,1);

nisum = 0;
Nisum = 0;

for i = 1:m
        ni = n(i);
        Ni = N(i);

        mu = WMW(Nisum+1:Nisum+Ni,1:Ni)*...
                (WMhc(1:Ni,i)+Woibar(:,Nisum+1:Nisum+Ni)'*qo);
        q(nisum+1:nisum+ni) = Minv(nisum+1:nisum+ni,1:ni)*(hibar(nisum+1:nisum+ni)-...
                                        Witilde(1:ni,Nisum+1:Nisum+Ni)*mu);
        nisum = nisum + ni;
        Nisum = Nisum + Ni;
end
```

D.6 msolve03.m

```
% function [qo,q] = MSolve03(Mo, Mi, hobar, hibar,...
%                               Woibar, Wibar, Woitilde, Witilde, ci)
%
% This function SOLVE the variables (velocities or accelerations) for simulation of
% a Multi-body system with one single node using standard elimination technique.
%

function [qo,q] = MSolve03(Mo, Mi, hobar, hibar,...
                                Woibar, Wibar, Woitilde, Witilde, ci)

global m n N totaln totalN

M = zeros(6+totaln+totalN);
M(1:6,1:6) = Mo;M(6+totaln+1:6+totaln+totalN,1:6) = Woibar';
M(1:6,6+totaln+1:6+totaln+totalN) = Woitilde;
nisum = 0;
Nisum = 0;
for i = 1:m
        ni = n(i);
        Ni = N(i);
        M(6+nisum+1:6+nisum+ni,6+nisum+1:6+nisum+ni) = Mi(nisum+1:nisum+ni,1:ni);
        M(6+totaln+Nisum+1:6+totaln+Nisum+Ni,6+nisum+1:6+nisum+ni)    =
Wibar(1:ni,Nisum+1:Nisum+Ni)';
        M(6+nisum+1:6+nisum+ni,6+totaln+Nisum+1:6+totaln+Nisum+Ni)    =
Witilde(1:ni,Nisum+1:Nisum+Ni);
nisum = nisum+ni;
Nisum = Nisum+Ni;
end
q = M\[hobar;hibar;ci];
qo=q(1:6);
q=q(7:6+totaln);
```

D.7 Others

```
% CROSS          compute the vector cross product
%
% c = cross (a, b)     or
% A = cross (a)
%
% a,b   vectors or 3xn matrices
% c     cross product(s) of a and b or corresponding columns of a and b
% A     a skew symmetric matrix which give the cross product of
%       a and other vector such that Ab = a x b
```

```
function c = cross (a, b)

if nargin == 1                    % just one input
        c = zeros(3);
        c(1,2) = -a(3);
        c(2,1) =  a(3);
        c(1,3) =  a(2);
        c(3,1) = -a(2);
        c(2,3) = -a(1);
        c(3,2) =  a(1);
else                              % two input
        if sum(size(a)) == 4   % a is a vector
                c = [  a(2)*b(3) - a(3)*b(2);
                       a(3)*b(1) - a(1)*b(3);
                       a(1)*b(2) - a(2)*b(1) ];
        else                      % a is a set of vectors
                c = [  a(2,:).*b(3,:)-a(3,:).*b(2,:)
                       a(3,:).*b(1,:)-a(1,:).*b(3,:)
                       a(1,:).*b(2,:)-a(2,:).*b(1,:)];
        end
end


% function ts = DHsTran(alpha, a, theta, d)
%
% This function compute the transformation matrix (G) for a twist between two
% links specified by Standard D-H convention
%
% theta, alpha, d, a    parameters of the link
%


function G = DHsTran(alpha, a, theta, d)

ct=cos(theta); st=sin(theta);
ca=cos(alpha); sa=sin(alpha);

R1=[1 0 0;0 ca sa; 0 -sa ca];
R2=[ct st 0; -st ct 0; 0 0 1];
G=[R1 cross([-a 0 0])*R1;zeros(3) R1]*[R2 cross([0 0 -d])*R2; zeros(3) R2];


% function [Rob, G, Gob] = EulerPar(r, [e0, e1, e2, e3])
%
% This function calculate the rotation matrix from object frame to base frame (Rob);
% the matrix for transforming derivative of the Euler parameters (p) to and from the
% angular velocity (G); and the transformation matrix (Gob) transforms twists of
% object frame to those of system base frame
%
% r     position vector of object frame with respect to base frame
% p     Euler parameters describing orientation of object frame

% where                p_dot = 1/2 * G' * w
%               w = 2 * G * p_dot

function [Rob, G, Gob] = EulerPar(x, p)

e = p(2:4);
e0=p(1);
G = [-e cross(e)+e0*eye(3)];
Rob = (2*e0^2-1)*eye(3)+2*(e'*e-e0*cross(e));
Gob = [Rob cross(x)*Rob;zeros(3) Rob];
```

REFERENCES

[1] Cutkosky, M.R., 1985, *Robotic grasping and fine manipulation*, Kluwer Academic Publishers.

[2] Markenscoff, X., Ni, L, and Papadimitriou, C.H., 1990, *The geometry of grasping*, Int. J. of Robotics Research, 9(1), pp. 61-74.

[3] Nguyen, V.D., 1988, *Constructing force-closure grasps*, Int. J. of Robotics Research, 7(3), pp. 3-16.

[4] Markenscoff, X. and Papadimitriou, C.H., 1989, *Optimal grip of a polygon*, Int. J. of Robotics Research, 8(2), pp. 17-29.

[5] Xiong, Y.L., Sanger, D.J., & Kerr, D.R., 1993, *Geometric Modelling of boundless grasps*, Robotica vol.11 pp. 19-26.

[6] Trinkle, J.C., Abel, J.M., and Paul, R.P., 1988, *An investigation of frictionless enveloping grasping in the plane*, Int. J. of Robotics Research, 7(3), pp. 33-51.

[7] Hsu, P. and Su, S., 1992, *Coordinated control of multiple manipulator systems -- experimental results*, Proc. IEEE Int. Conf. on Robotics and Automation, pp. 2199-2204.

[8] Hu, Y.R. and Goldenberg, A.A., 1994, *An approach to motion and force

control of coordinated robot arms in the presence of joint flexibility, ASME Journal of Dynamic Systems, Measurement, and Control, vol. 116, pp. 326-335.

[9] Osumi, H. and Arai, T., 1994, *Cooperative control between two position-controlled manipulators*, Proc. IEEE Int. Conf. on Robotics and Automation, pp. 1509-1514.

[10] Uchiyama, M., 1990, *A unified approach to load sharing, motion decomposing, force sensing of dual arm robotics*, Robotics Research: the Fifth Int. Sym., pp. 225-232.

[11] Uchiyama, M. and Dauchez, P., 1988, *A symmetric hybrid position/force control scheme for the coordination of two robots*, Proc. IEEE Int. Conf. on Robotics and Automation, pp. 350-356.

[12] Carignan, C.R. and Akin, D.L., 1988, *Cooperative control of two arms in the transsport of an inertial load in zero gravity*, IEEE J. of Robotics and Automation, vol. 4, no. 4, pp. 414-419.

[13] Walker, I.D. and Freeman, R.A., 1989, *Internal object loading for multiple cooperating robot manipulators*, Proc. IEEE Int. Conf. on Robotics and Automation, pp. 606-611.

[14] Lu, W.S. and Meng, Q.H., 1992, *On optimal force distribution of coordinating manipulators*, Int. J. of Robotics and Automation, vol. 7, no. 2, pp. 70-79.

[15] Uchiyama, M. and Yamashita, T., 1992, *Adaptive load-sharing for two cooperative manipulators to hold an object robustly*, Int. J. of Robotics and Automation, vol. 7, no. 1, pp. 3-9.

[16] Yun, X., 1993, *Object handling using two arms without grasping*, Int. J. of Robotics Research, 12(1), pp. 99-106.

[17] Zheng, Y.F., 1989, *Kinematics and dynamics of two industrial robots in assembly*, Proc. IEEE Int. Conf. on Robotics and Automation, pp. 1360-1365.

[18] Dellinger, W.F. and Anderson, J.N., 1992, *Interactive force dynamics of two robotic manipulators grasping a non-rigid object*, Proc. IEEE Int. Conf. on Robotics and Automation, pp. 2205-2210.

[19] Tarn, T.J., Bejczy, A.K., and Yun, X., 1987, *Design of dynamic control of two cooperating robot arms: closed chain formulation*, Proc. IEEE Int. Conf. on Robotics and Automation, pp. 7-13.

[20] Yun, X. and Kumar, V.R., 1991, *An approach to simultaneous control of trajectory and interaction forces in dual-arm configurations*, IEEE Trans. on Robotics and Automation, vol. 7, no. 5, pp. 618-625.

[21] Yun, X., Kumar, V., Sarkar, N., and Paljug, E., 1992, *Control of multiple arms with rolling constraints*, Proc. IEEE Int. Conf. on Robotics and Automation, pp. 2193-2189.

[22] Nilanjan, S., Yun, X., and Kumar, V., 1993, *Dynamic control of 3-D rolling*

contacts in two-arm manipulation, Proc. IEEE Conf. on Robotics and Automation, vol. 3, pp. 978-983.

[23] Sarkar, N., Yun, X., and Kumar, V., 1994, *Control of a single robot in a decentralized multi-robot system*, Proc. IEEE Int. Conf. on Robotics and Automation, pp. 896-900.

[24] Kumar, V., Yun, X., Paljug, E. & Sarkar, N., 1991, *Control of contact conditions for manipulation with multiple robotic systems*, Proc. IEEE Int. Conf. on Robotics and Automation, pp. 170-175.

[25] Li, Z., Hsu, P., and Sastry, S., 1989, *Grasping and coordinated manipulation by a multifingered robot hand*, Int. J. of Robotics Research, 8(4), pp. 33-50.

[26] Jankowski, K.P., ElMaraghy, H.A., and ElMaraghy, W.H., 1993, *Dynamic coordination of multiple robot arms with flexible joints*, Int. J. of Robotics Research, 12(6), pp. 505-528.

[27] Laroussi, K., Hemami, H., and Goddard, R.E., 1988, *Coordination of two planar robots in lifting*, IEEE J. of Robotics and Automation, vol. 4, no. 1, pp. 77-85.

[28] Mason, M.T. and Salisbury, J.K., 1985, *Robot hands and the mechanics of manipulation*, The MIT Press.

[29] Yoshikawa, T. & Zheng, X.Z., 1993, *Coordinated dynamic hybrid position/force control for multiple robot manipulators handling one*

constrained object, Int. J. of Robotics Research, 12(3), pp. 219-230.

[30] Lilly, K.W., 1993, *Efficient Dynamic Simulation of Robotic Mechanisms*, Kluwer Academic Publishers.

[31] Jain, A., Rodriguez, G., and Kreutz, K., 1990, *Multi-arm grasp and manipulation of objects with internal degrees of freedom*, Proc. 29th Conf. on Decision and Control, pp. 3110-3111.

[32] García de Jalón, J. and Bayo, E., 1994, *Kinematic and dynamic simulation of multibody systems - the real-time challenge*, Springer-Verlag.

[33] Nikravesh, P.E., 1988, *Computer-aided analysis of mechanical systems*, Prentice Hall.

[34] Haug, E.J., 1989, *Computer-aided kinematics and dynamics of mechanical systems , Volume I: basic Methods*, Allyn and Bacon.

[35] G. Rodriguez, Kreutz, K., and Jain, A., 1989, *A spatial operator algebra for manipulator modeling and control*, Int. J. of Robotics Research, 10(4), pp. 371-381.

[36] Featherstone, R., 1987, *Robot dynamics algorithm*, Dluwer Acadmic Publishers.

[37] Oh, S.Y. and Orin, D.E., 1986, *Dynamic computer simulation of multiple closed-chain robotic mechanisms*, Proc. IEEE Conf. on Robotics and

Automation, pp. 299-304.

[38] Lilly, K.W. and Orin, D.E., 1994, *Efficient dynamic simulation of multiple chain robotic mechanisms*, ASME Journal of Dynamic Systems, Measurement, and Control, vol. 116, pp. 223-231.

[39] McMillan, S., Sadayappan, P., and Orin, D.E., 1992, *Efficient dynamic simulation of multiple manipulator system with singularities*, Proc. IEEE Int. Conf. on Robotics and Automation, pp. 299-304.

[40] Montana, D.J., 1992, *Contact stability for two-fingered grasps*, IEEE Trans. on Robotics and Automation, vol. 8, no. 4, pp. 421-430.

[41] Craig, J.J., 1989, *Introduction to robotics -- mechanics and control*, Second Edition, Addison-Wesley Publishing Company, Inc.

[42] Lankarani, H.M. and Nikravesh, P.E., 1988, *Application of the canonical equations of motion in problems of constrained multibody systems with intermittent motion*, Advances in Design Automation, vol. 14, pp. 417-423.

[43] Walker, M.W. and Orin D.E., 1982, *Efficient Dynamic computer Simulation of Robotic Mechanisms*, ASME Journal of Dynamic Systems, Measurement, and Control, vol. 104, pp. 205-211.