REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
3/3/95

3. REPORT TYPE AND DATES COVERED
Final Report

4. TITLE AND SUBTITLE
Synthesis and Simulation of Robotic Motions

5. FUNDING NUMBERS
DAAL03-90-G-0005

6. AUTHOR(S)
Bahram Ravani

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Department of Mechanical & Aeronautical Engineering
University of California, Davis
Davis, CA 95616

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
ARO 27075.4-EG

11. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This research has dealt with scientific studies in design synthesis and simulation of robotic motions. Methods are developed for design of robotic systems components as well as for computer graphics based simulation of robotic motions. The results developed make fundamental contributions to the fields of Mechanical Design, Kinematics and Multi-Body Dynamics, Computer Aided Geometric Design, and Computer Graphics.

14. SUBJECT TERMS
Robotics, Kinematics, Dynamics, Computer Aided Geometric Design, Computer Graphics

15. NUMBER OF PAGES
6 pages w/o appendix

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

27075-EG

Synthesis and Simulation of Robotic Motions

ARO Proposal Number 27075-EG
ARO Grant Number DAAL03-90-G-0005

Final Report

Grant Number DAAL03-90-G-0005

Principal Investigator: Professor Bahram Ravani

Revised and Submitted March 1995

Problem Statement

This research project has involved studying methods for Synthesis and Simulation of robotic motions. The emphasis has been on the development of a scientific foundation for design synthesis as well as computer graphics based high performance simulation for multi-degree of freedom motions in general and robotic systems and components in particular.

Background

Military applications of robot manipulators are usually different from the commercial manufacturing applications in that robots are used in the unstructured field operations where there is a need for long reach, redundant degrees of freedom, high payloads and on site calibration and targeting. Commercially available robots are usually designed for the more structured manufacturing applications and therefore are not suitable for such field operations. Military robots are therefore custom designed for each application. This means that there is a need for a set of scientific design and planning tools that can be used in the initial design and the subsequent field testing of military robots. This research deals with development of scientific fundamentals that can provide the foundation for the development of such tools.

Summary of Results

The main results of this research has been published (or is being published) in the technical literature. A summary of these publications is provided in the next section. Here, we provide a summary of some of the main results.

The work performed under this project can be divided into two parts: one dealing with issues associated with motion synthesis meaning design synthesis of robotic motions, sub-components, and systems and one dealing with high performance computer graphics simulation of robotic and general multi-degree-of-freedom motions. Several results have been derived and development in each of these broad areas that have not only extended the state of knowledge in robotics and mechanical system design but have also provided impact in other areas such as computer graphics and computational geometry. Each of these two parts are described in separate sub-sections below.

Results in Design Synthesis

In the area of design synthesis we have published seven technical papers (Papers 1-7 in the next section). We have developed a general computational geometric structure for mechanical motion synthesis that can be used to design mechanical linkages (see publication No. 1). Since spherical linkages can be used to model robotic wrists, we have applied this technique to the design of spherical linkages representing robotic wrists mechanisms.

In the case of redundant manipulators, we have developed a set of kinematics design criterion that can be used to generate singularity free motion trajectories. Kinematic design arrangements have been studied for redundant manipulators that would produce singularity free trajectories within the workspace. Redundant manipulators have extra degrees of freedom that can be used in reaching around objects or in avoiding singularities. These manipulators are specially useful in Army applications involving, for example, camouflage painting of Army vehicles or in rapid repair of Army facilities. In the past there exists very little knowledge on how to globally design such manipulators to assure that singularity free trajectories will exist for any set of path requirements within the workspace of the manipulator. In this research an approach based on Lie algebra is used and the kinematic problem for a redundant robot is formulated as a control problem. This problem is then used to define a set of kinematic design criterion that can be used to check suitability of a design for singularity avoidance. The results are summarized in publication No. 2.

Another, general set of results in design synthesis has dealt with motion design for instantaneous performance. A powerful tool used in instantaneous kinematic design is the set of so-called "instantaneous invariants". These quantities are used to either study differential properties of motions or to design for certain differential kinematic performance. The problem for applying instantaneous invariants to three dimensional linkages such as robot manipulators is that their derivation is based on the use of the so-called canonical coordinate system which is difficult to derive for spatial linkages. In this research we have developed a method for the derivation of instantaneous invariants which are independent of the use of canonical coordinate system. This is described in publication No. 3.

In terms of more specialized research in design synthesis, our emphasis in this work has been on the design of drive systems and their components for robotic systems. Several applications of robotic for military field operations require design of specialized drive systems. This includes for example design and development of specialized automatic weapon loading systems. Here we have developed a set of scientific techniques for design and sizing of the drive system of robot manipulators including the joint actuators and ball screw type drive mechanisms.

In this area, we have developed a method for sizing the joint actuators of a robot manipulator. The problem is formulated as an optimization problem where the actuators are selected to achieve a desired dynamic load carrying capacity of the manipulator over an entire trajectory. Worst case trajectories in the robot workspace are then used to generalize the results to joint actuator selection over the entire robot workspace. Appropriate constraints are included in the problem formulation to prevent actuators from being overloaded and to limit their sizes. The results are summarized in publication No. 4.

In most robotic systems, electric actuators are usually coupled with some form of geared transmission mechanism. There has been some previous work on gearing systems but very little work exist in understanding of ball screw mechanisms which are also commonly used in several robotic systems. These mechanisms are also used in other one or two degree of freedom automation systems used by the US Army such as the tilt table ammunition loading system. We have developed the basic kinematic model of the ball screw mechanism and have shown that much of the existing results in the literature are based on false assumptions. We have also included in our model the effect of contact deformation and have identified the pattern of sliding lines of contact for wear and finite element analyses. We have used the results of our kinematic analysis to develop methods for design and efficiency analysis of such mechanisms. The work developed under this project represents the first comprehensive and mathematically correct treatment of the subject. The results are summarized in two separate publications No. 5 and 6.

Since every drive system for a robot manipulator has to use some sort of a ball bearing, this research has also looked into some aspects of the analysis of the high speed thrust Ball Bearings. A method is developed for design of Ball Bearings that would results in optimum contact angle and minimum friction. The method is specially useful for high speed applications and is described in detail in publication No. 7.

Results in High Performance Computer Graphics Simulation

In the area of simulation of robotic motions, this research has made several fundamental contributions in both kinematics and dynamics simulation as well as in the area of calibration and referencing for simulation purposes. The results in addition to their applications in robotics are considered as fundamental contributions to the fields of Computer Aided Geometric Design, Computer Graphics, Kinematics and Dynamics.

IN the area of kinematic simulation of robotic motions, a scientific foundation is developed for motion interpolation based on the use of a kinematic mapping for spatial kinematics. Both analytical (publication No. 8) as well as discrete computational geometric methods (publication No. 9) are developed for design of Bezier type motions that can interpolate a series of control configurations of an end-effector. In robotics, generation of Cartesian trajectories requires interpolation of specified target configurations of the end-effector. In computer simulation of large mechanical systems, generation of tightly spaced displacements of a moving body along its trajectory may not be cost effective. Instead, a series of displacements are generated using the dynamic equations of motion and then the in-between displacements can be constructed using appropriate motion inteprolants. In computer graphics, motion interpolation is a fundamental problem in animation. In this research we have development the first very fundamental analysis of the motion interpolation problem (publications No. 8 and 9) and for the first time have developed a completely coordinate independent method for interpolation of rotations (publications No. 10 and 11). We have also been able to develop the first generation of the so-called Bezier curves in nonlinear or curved spaces of manifolds of rigid body motion (publications No. 10 and 11). In dealing with simulation of robotic motions, the kinematic inteprolation techniques developed would allow smooth animation of robot motions without the need for solving the differential equations of motions at all display sampling intervals. This results in a simulation capability that would require much less computer power as the more traditional approaches.

Any CAD (Computer Aided Design) based robotic simulation system requires some level of calibration capability before it can be utilized for military field applications. As part of this research, a method is developed for field calibration and targeting of robot manipulators. The calibration approach would allow a robot to use targets in a site to reference itself to its operational field environment. The algorithm developed uses Clifford algebra to exploit the geometry of the CAD model and uniquely assigns a body fixed coordinate system to the computer model of the object. This is important in World Model calibration for off-line robot programming and simulation (see publication No. 12) where the computer model has to update its object models automatically from sensor measurements. The results, in addition to their robotic applications, can also be useful in artillery aiming and target acquisition problems in smart Weapon systems. In addition, we have studied the mechanics of automatic targeting using different sensor arrangements and have developed a method for reconstructing object or target locations based on redundant sensor measurements of different geometric features (see publication No. 13). Although the problem lends itself to a non-linear least squares problem, we have been able to develop a formulation that reduces the problem into a linear problem suitable for fast computations. The results, in addition to their application in calibration of robotic and mechanical system simulation software packages have applications in automatic targeting for smart weapon systems.

The method, in a specialized form, has been tested by a specialized medical instrument manufacturer (Accuray in Silicon Valley, California) for a robotic system for Stereolaxic Radiosurgery. The results have been promising and has the potential in improving the accuracy of their system and reducing the calibration requirements for their device.

Al l of the work described so far in this section has dealt with kinematic simulation. In the area of dynamic simulation, our activities have focused on development of methods for the formulation of dynamical equations of motion for computational purposes and dynamic interpolation of motion trajectories. In terms of formulation of equations of motion for dynamical systems, we have developed a method that would result in first order decoupled equations of motion for multi body mechanical systems. Such equations can then be more easily integrated resulting in the computational performance needed for high performance computer graphic simulation. The results developed apply to not only robotic systems but also to the more general class of mechanical systems with multiple degrees of freedom, and made of multiple bodies. The decoupling method involves an algorithm for selecting the generalized coordinates used to define the configuration of a dynamical system. This choice of the so-called generalized coordinates guarantee that the resulting dynamical equations of motion become decoupled in the highest derivative terms. The algorithm uses congruency transformation and constraint relaxation to achieve the first order decoupling of the resulting equations of motion. In the case of the more complex dynamical systems where the appropriate congruency transformation may be difficult to obtain, a method is developed based on the use of orthogonal complements that still achieves the needed computational efficiency. The results have been used to simulate the dynamic equations of motion for a robotic system on a moving base. This is the kind of robotic system useful in Army field applications where a robot for ammunition loading for example can be mounted on the back of an Army truck. The work is published in publication No. 14.

In the area of dynamic interpolation, a method is developed for generation of dynamic equations of motion that interpolates two end positions of an end-effector. The problem is formulated as a two pointy boundary value problem and is solved using a numerical method. The results are summarized in publication No. 15 and have applications again in robotic system simulation and computer graphics animation of rigid body motion.

List of Publications

1. Ge, Q. J. and B. Ravani, "A Computational Geometric Approach to Motion Synthesis", in Artificial Intelligence, Expert Systems and Symbolic Computing, E. N. Housus and J. R. Rice editors, Elsevier Publishers, 1992, pp. 193-202.

2. Sardis, R. M., B. Ravani, and R. C. M. Bodduluri, "A Kinematic Design Criterion for Singularity Avoidance in Redundant Manipulators", Proc. of the Third International Workshop on Advances in Robot Kinematics, Ferrara Italy, Sept. 1992.

3. Lee, C. , A. T. Yang, and B. Ravani, "Coordinate System Independent Form of Instantaneous Invariants in Spatial Kinematics", ASME Trans., Journal of Mechanical Design, Vol. 115, December 1993, pp. 946-952. Also in the Proc. of 1990 ASME Mechanism Conference, Chicago, Il, Sept. 1990.

4. R. Kashani and B. Ravani, "Optimal Sizing of Robot Actuators Based on Dynamic Load Carrying Capacity ", Proc. of the 1990 ASME Mechanism Conference, Chicago, IL, Sept. 1990, pp. 193-197.

5. Lin, M. C., B. Ravani, and S. A. Velinsky, "Kinematics of the Ball Screw Mechanism", ASME Trans. , Journal of Mechanical Design, Vol. 116, Sept. 1994, pp. 849-855.

6. Lin, M. C., S. A. Velinsky, and B. Ravani, "Design of the Ball Screw Mechanism for Optimum Efficiency", ASME Trans. , Journal of Mechanical Design, Vol. 116, Sept. 1994, pp. 856-861.

7. Lin, M. C., S. A. Velinsky, and B. Ravani, "On the Analysis and Design of High Speed Thrust Ball Bearing", Proc. of the Eight World Congress on the Theory of machines and Mechanisms, Prague, Czechoslovakia, August 1991.

8. Ge, Q. J. , and B. Ravani, "Computer Aided Geometric Design of Motion Interpolants", ASME Trans. Journal of Mechanical Design, Vol. 116, Sept. 1994, pp. 756-762.

9. Ge, Q. J. , and B. Ravani, "Geometric Construction of Bezier Motion Interpolants", ASME Trans. Journal of Mechanical Design, Vol. 116, Sept. 1994, pp. 749-755.

10. Park, F. C., and B. Ravani, "Bezier Curves on Riemanian Manifolds and Lie Groups with Kinematics Applications", ASME Trans. Journal of Mechanical Design, March 1995, to appear.

11. Park, F. C., and B. Ravani, "Smooth Interpolation of Rotations", Submitted for Publication to ACM Trans. on Computer Graphics, 1994.

12. Ge, Q. J. and B. Ravani, "Kinematic Localization for World Model Calibration in Off-Line Robot Programming Using Clifford Algebra", Proc. of IEEE Int'l Conf. on Robotics and Automation, Sacramento, CA, April 1991, pp. 584-589.

13. Ge, Q. J. and B. Ravani, "Computation of Spatial Displacements From Redundant Feature Information", ASME Trans. Journal of Mechanical Design, Vol. 116, December 1994, pp. 1073-1080.

14. Loduha, T. A. and B. Ravani, "On the First Order Decoupling of Equations of Motion for Constrained Dynamical Systems", ASME Trans. Journal of Applied Mechanics, March 1995, to appear.

15. Loduha, T. A. and B. Ravani, "Motion Interpolation Using Dynamics", Proc. of the Design Automation Conf., Minneapolis, MN, Sept. 1994, pp. 175-181.

Participating Scientific Personnel

The following is a list of technical personnel participating at various stages of this project:

* Professor Bahram Ravani, Principal Investigator
* Q. Jeffrey GE, Research Associate (post doctoral student)
* Timothy A. Loduha, Research Assistant. ***Received PhD degree*** in Mechanical Engineering for his research on this project.
* Frank C. Park, visiting faculty.
* R. M. C. Bodduluri, Research Associate

Report of Inventions:

This research was for basic fundamental developments and therefore there was no inventions in this project. The results were a set of new theories and fundamental knowledge.

Appendix A

Copies of Publications

Artificial Intelligence, Expert Systems and Symbolic Computing

Selected and revised papers from the
IMACS 13th World Congress,
Dublin, Ireland, July 1991

edited by

Elias N. HOUSTIS
John R. RICE
Department of Computer Science
Purdue University
West Lafayette, IN, U.S.A.

1992

NORTH-HOLLAND
AMSTERDAM • LONDON • NEW YORK • TOKYO

ION
tion)

igation of
e systems
·rovement
symbolic
xpression
[5]. This

√ Mexico,

·a system
ics/ Eds.
:rg, 1989,

: stability
[.Nagata,

ng three-
Rodrigue,

·, A code
t, ACM–

Artificial Intelligence, Expert Systems and Symbolic Computing
E.N. Houstis and J.R. Rice (Editors)
Elsevier Science Publishers B.V. (North-Holland)

A Computational Geometric Approach to Motion Synthesis

Q. J. Ge[a] and B. Ravani[b]

[a]Department of Mechanical Engineering, University of New Orleans,
New Orleans, Louisiana 70148, U.S.A.

[b]Computer Integrated Design and Manufacturing Laboratory,
Department of Mechanical, Aeronautical and Materials Engineering,
University of California-Davis, Davis, California 95616, U.S.A.

Abstract

This paper studies computational geometry of motions and develop a computational geometric structure that can be used for mechanical motion synthesis. This allows development of computational algorithms and software systems to support the mechanical design activity. The approach uses an orientable kinematic mapping to transform the mechanical design problem into a curve design problem in the space of the mapping. The curve design problem for synthesis of an analytic motion is carried out by Hermite interpolation. In case of a mechanical linkage, however, the Hermite interpolation is combined with a curve fitting procedure for synthesizing the motion.

1 INTRODUCTION

This paper combines concepts from the field of computational geometry and kinematics. It develops a computational geometric structure that forms the basis for solving a class of mechanical design synthesis problems. The geometric structure presented allows formulation and application of various computational algorithms and software systems to solve several problems in mechanical design. The domain of such design problems considered is that of mechanical motion synthesis (see Ravani and Roth 1983). This involves design and sizing a mechanism (mechanical linkage) for a prescribed motion of one of its links or an analytic motion program, for example, for joint actuators of a robot manipulator for a prescribed end-effector trajectory.

The basic idea is to transform, the mechanical design synthesis problem into a geometric problem with a computational structure. Software systems and algorithms can then be more readily developed for the mechanical problem taking advantage of existing algorithms and methods in the fields of computational geometry and computer algorithms. The ideas presented here are extensions of the ideas in Ravani and Roth (1983) and (1984) taking advantage of more recent computational geometric results in Ge and Ravani (1991). The basic approach involves transforming the mechanical design prob-

Preface

This volume contains a selection of papers presented at the 13th World Congress on Computation and Applied Mathematics. This Congress was organized by IMACS in Dublin, Ireland on July 22-26, 1991. The best papers in the areas of artificial intelligence, expert systems, and symbolic computing were selected along with applications to scientific computing. About one-third of the papers presented in these areas at the World Congress were selected.

These 48 papers provide an excellent overview of the dynamic state of these closely related fields. We foresee that scientific computation will involve symbolic and artificial intelligence tools more and more as these software systems become more and more sophisticated. The future systems of computational science and engineering will be problem solving environments created with components from numerical analysis, computational geometry, symbolic computing, and artificial intelligence. The historical separation of these fields will gradually blur as they come together to create the high level, natural systems of the future.

We thank Georgia Connaroe and Connie Wilson for their excellent help in organizing the correspondence for the papers in this book. We also thank Robert Vichnevetsky for his encouragement to us in editing this volume.

Elias N. Houstis and John R. Rice

lem into the problem of geometric curve design problem. This is made possible using a geometric mapping introduced by Ravani and Roth (1984). The mapping transforms a mechanical motion in the Euclidean space into a curve in a non-Euclidean space referred to as the image space of the mapping. Ravani and Roth used the mapping and solved the infinite dimensional form of the linkage design problem by a curve fitting procedure. In practice, the number of specified points for the curve fitting procedure is usually small not allowing global control on the entire motion of the designed linkage. In this paper, the geometric interpolation method developed for synthesis of analytic motions is used to generate additional desired points for the curve fitting procedure. Research is presently underway to fully eliminate the curve fitting procedure. This is not presently tractable since many mechanism motions can not be easily described by piecewise parametric representations used in Computer Aided Geometric Design (CAGD).

In this paper the kinematic mapping introduced by Ravani and Roth (1984) is made orientable making it more suitable for computational geometric algorithms. Using this oriented form of the mapping and generalizing techniques from the field of CAGD, methods are developed for geometric design of piecewise parametric curves that are approximations of analytic motions. In this manner, controlling the shape of a curve allows dimensional synthesis of analytic motions. In the case of dimensional synthesis for a mechanical linkage, the problem is considered in two stages. The first stage uses the geometric interpolation to generate enough specifications for a desired motion to have more global control on the design of the linkage. The second stage uses a curve fitting procedure to determine dimensional parameters of the desired linkage. The approach provides for a computational geometric method for mechanical design synthesis with global control over the entire range of the motion of the linkage.

The purpose of this paper is only to introduce the idea and develop some of the required mathematics. The paper does not attempt to provide a complete solution of the problem or work out general cases for every kind of motion approximation. In the case of synthesis of analytic motions only Hermite interpolation is discussed. In relation to dimensional synthesis of mechanical linkages, only a simple example involving dimensional synthesis of special spherical linkage is presented.

The outline of the paper is as follows. We first provide a brief description of the oriented image space of kinematic mapping. This provides for a geometric representation for displacements and motions in terms of points and curve segments, respectively, in the image space. Then we develop a method for Hermite interpolation for spherical motions. This can be used to synthesize analytic spherical motions. The approach is then combined with a curve fitting procedure for design of spherical linkages. The results are only presented for a special spherical four bar linkage.

2 THE ORIENTED MAPPING

A screw displacement in Euclidean three-space is a rotation about and a translation along a line in the space called the screw axis. A general displacement of an object from one position to another is equivalent to two oppositely oriented (namely "forward" and "backward") screw displacements about two coincident but oppositely oriented screw axes. The forward and backward screw displacements define two sets of oppositely signed *dual Euler parameters* $\hat{\mathbf{X}} = (\hat{X}_1, \hat{X}_2, \hat{X}_3, \hat{X}_4)$ and $-\hat{\mathbf{X}} = (-\hat{X}_1, -\hat{X}_2, -\hat{X}_3, -\hat{X}_4)$ (see Ge and Ravani 1991) where

$$\begin{aligned} \hat{X}_1 &= \hat{s}_1 \sin(\hat{\theta}/2), \\ \hat{X}_2 &= \hat{s}_2 \sin(\hat{\theta}/2), \\ \hat{X}_3 &= \hat{s}_3 \sin(\hat{\theta}/2), \\ \hat{X}_4 &= \cos(\hat{\theta}/2). \end{aligned} \tag{1}$$

The symbol "$\hat{}$" denotes numbers of the form $\hat{a} = a + \epsilon a^0$, called *dual numbers*, where ϵ has the property $\epsilon^2 = 0$. The dual numbers $\hat{s}_i = s_i + \epsilon s_i^0$ $(i = 1, 2, 3)$ are *Plücker line coordinates* representing the screw axis; the dual number $\hat{\theta} = \theta + \epsilon d$ is the dual angle representing the amount of a screw displacement. The sine and cosine functions of a dual angle are defined as

$$\begin{aligned} \sin(\hat{\theta}/2) &= \sin(\theta/2) + \epsilon(d/2)\cos(\theta/2), \\ \cos(\hat{\theta}/2) &= \cos(\theta/2) - \epsilon(d/2)\sin(\theta/2). \end{aligned}$$

More details on dual numbers, Plücker coordinates and dual Euler parameters can be found in Bottema and Roth (1979).

The dual Euler parameters (1) can also be written in the form of $\hat{X}_i = X_i + \epsilon X_i^0$, $i = 1, 2, 3, 4$, where

$$\begin{aligned} X_1 &= s_1 \sin(\theta/2), \\ X_2 &= s_2 \sin(\theta/2), \\ X_3 &= s_3 \sin(\theta/2), \\ X_4 &= \cos(\theta/2) \end{aligned} \tag{2}$$

and

$$\begin{aligned} X_1^0 &= s_1(d/2)\cos(\theta/2) + s_1^0 \sin(\theta/2), \\ X_2^0 &= s_2(d/2)\cos(\theta/2) + s_2^0 \sin(\theta/2), \\ X_3^0 &= s_3(d/2)\cos(\theta/2) + s_3^0 \sin(\theta/2), \\ X_4^0 &= -(d/2)\sin(\theta/2). \end{aligned}$$

The parameters $\mathbf{X} = (X_1, X_2, X_3, X_4)$ are the Euler parameters of a rotation and the parameters $\mathbf{X}^0 = (X_1^0, X_2^0, X_3^0, X_4^0)$ represents a translation after the rotation $\mathbf{X}$.

Study (1891) used the dual Euler parameters in the form of $(\mathbf{X}, \mathbf{X}^0)$ and developed a geometric representation for a spatial displacement in terms of Study vectors which represent oriented lines in a space of four dimensions called the *Soma space.* A recent account of the Study vectors can be found in Bottema and Roth (1979). Ravani and Roth (1984) used the dual Euler parameters in the form of (1) as a set of four homogeneous coordinates to define a geometric mapping of displacements into points of a three dimensional projective space with a dual metric called the *image space.* Thus a one-degree-of-freedom mechanical motion is mapped into an image space curve; and a two-degree-of-freedom motion is mapped into a two dimensional surface in the image space. They then solved the linkage dimensional synthesis problem by a curve-fitting procedure for an algebraic curve.

Ge and Ravani (1991) made the geometric mapping of Ravani and Roth orientable by considering the *signed* dual Euler parameters as the *signed* homogeneous coordinates of an image point. Instead of identifying the points $\hat{\mathbf{X}}$ and $\hat{w}\hat{\mathbf{X}}$ (where $\hat{w} = w + \epsilon w^0$ is a nonpure dual number) completely, Ge and Ravani distinguished them according to the sign of w. They are considered to be identical if $w > 0$; and they are two antipodal points if $w < 0$. Two antipodal points are two distinct points occupying the same position in the image space but with "opposite orientations", namely "forward" and "backward". In this way a general displacement, which can be performed either by a forward or a backward screw displacement, corresponds not to *one* image point but *two* antipodal points. The image space consists of such antipodal points is called the *oriented image space*. It is a double covering of the original image space.

The introduction of orientation into the image space makes it possible to unambiguously define the distance between two image points, and the directed line joining from one point to another. Let $\hat{\mathbf{X}} = (\hat{X}_1, \hat{X}_2, \hat{X}_3, \hat{X}_4)$ and $\hat{\mathbf{Y}} = (\hat{Y}_1, \hat{Y}_2, \hat{Y}_3, \hat{Y}_4)$ denote two oriented points. The distance between them is the dual angle $\hat{\alpha} = \alpha + \epsilon l$ (where $0 \leq \alpha \leq \pi$) obtained from:

$$\cos\hat{\alpha} = \frac{\hat{\mathbf{X}} \cdot \hat{\mathbf{Y}}}{(\hat{\mathbf{X}} \cdot \hat{\mathbf{X}})^{1/2}(\hat{\mathbf{Y}} \cdot \hat{\mathbf{Y}})^{1/2}}, \tag{3}$$

where the inner product, $\hat{\mathbf{X}} \cdot \hat{\mathbf{Y}} = \hat{X}_1\hat{Y}_1 + \hat{X}_2\hat{Y}_2 + \hat{X}_3\hat{Y}_3 + \hat{X}_4\hat{Y}_4$, is a signed dual number. The norm of $\hat{\mathbf{X}}$ is $\sqrt{\hat{\mathbf{X}} \cdot \hat{\mathbf{X}}}$ and $\hat{\mathbf{X}}$ is said to have normalized coordinates when $\hat{\mathbf{X}} \cdot \hat{\mathbf{X}} = 1$.

The line joining from $\hat{\mathbf{X}}$ to $\hat{\mathbf{Y}}$ is given by the wedge product $\hat{\mathbf{X}} \wedge \hat{\mathbf{Y}}$. The symbol "$\wedge$" denotes the vector wedge product which generalizes to higher dimensions the vector cross product, see Flanders (1963). The components of $\hat{\mathbf{X}} \wedge \hat{\mathbf{Y}}$ are the *signed* Plücker coordinates of a line. Here the sign is significant because $-\hat{\mathbf{X}} \wedge \hat{\mathbf{Y}} = \hat{\mathbf{Y}} \wedge \hat{\mathbf{X}}$ represents a line joining from $\hat{\mathbf{Y}}$ to $\hat{\mathbf{X}}$. They define two coincident but oppositely oriented lines. Similarly, the orientations of a plane defined by three oriented points and a tetrahedron defined by four points can also be unambiguously defined.

3 INTERPOLATING ROTATIONS

Recent works on rotation interpolation that are related to our approach are Shoemake (1985, 1987), Duff (1986), and Pletinckx (1989). They developed subdivision methods for interpolating rotations using unit quaternions. Ge and Ravani (1991) built on their works and developed analytical methods for interpolating general displacements (including both rotations and translations) with higher order continuities. This section applies the results in Ge and Ravani (1991) to Hermite interpolation of rotations which correspond to spherical motions.

Given two distinct orientations of an object in space together with two instantaneous rotation axes at these orientations, a cubic parametric motion can be designed using Hermite interpolation. Essential to Hermite interpolation is the ability to define the "tangent direction" of an orientation. In this paper, we define the tangent direction using the notion of *absolute polarity* since the image space is a projective three space with an elliptic metric. Let an orientation be represented by an image point $\mathbf{X} = (X_1, X_2, X_3, X_4)$ where X_i $(i = 1, 2, 3, 4)$ are given by (2). Its tangent direction is then defined by a point $\mathbf{X}'$ on the polar plane of $\mathbf{X}$ with respect to the absolution $\Omega = \mathbf{X} \cdot \mathbf{X} = 0$, that is $\mathbf{X}'$ is defined such that $\mathbf{X}' \cdot \mathbf{X} = 0$. Let $\mathbf{p} = (p_1, p_2, p_3)$ denote the coordinates of the instantaneous axis at the orientation $\mathbf{X}$. Then the corresponding polar point $\mathbf{X}' = (X'_1, X'_2, X'_3, X'_4)$ is given by

$$\begin{bmatrix} X_1' \\ X_2' \\ X_3' \\ X_4' \end{bmatrix} = \begin{bmatrix} 0 & -p_3 & p_2 & p_1 \\ p_3 & 0 & -p_1 & p_2 \\ -p_2 & p_1 & 0 & p_3 \\ -p_1 & -p_2 & -p_3 & 0 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix} \tag{4}$$

The skewsymmetry of the matrix in (4) results in $\mathbf{X}' \cdot \mathbf{X} = 0$ for an arbitrary $\mathbf{p}$. Readers are referred to Coxeter (1961) for more explanations on the absolute polarity of the elliptic geometry.

We now turn to the Hermite interpolation of orientations. Let $\mathbf{X}_1$ and $\mathbf{X}_2$ denote the coordinates of the image points for two given orientations, respectively. In order for the interpolation to be unique, these coordinates are required to satisfy $\mathbf{X}_1 \cdot \mathbf{X}_1 = 1$, $\mathbf{X}_2 \cdot \mathbf{X}_2 = 1$, and $\mathbf{X}_1 \cdot \mathbf{X}_2 \geq 0$. Geometrically, this means that the coordinates are normalized and the angular distance between $\mathbf{X}_1$ and $\mathbf{X}_2$ is less than $\pi/2$. Let $\mathbf{p}_1$ and $\mathbf{p}_2$ denote, respectively, the coordinates of two given instantaneous axes. Then the two corresponding points $\mathbf{X}_1'$ and $\mathbf{X}_2'$ polar to $\mathbf{X}_1$ and $\mathbf{X}_2$, respectively, with respect to $\Omega = 0$, are can be obtained using (4). The image curve for the cubic motion that starts from the orientation $\mathbf{X}_1$ with the initial instantaneous axis $\mathbf{p}_1$ and ends at the orientation $\mathbf{X}_2$ with the final instantaneous axis $\mathbf{p}_2$ is obtained by the following Hermite interpolation

$$\mathbf{P}(t) = [t^3 \; t^2 \; t \; 1][M_h] \begin{bmatrix} w_1\mathbf{X}_1 \\ w_1'\mathbf{X}_1' \\ w_2'\mathbf{X}_2' \\ w_2\mathbf{X}_2 \end{bmatrix}, \quad t = [0, 1]$$

where $[M_h]$ is the Hermite basis matrix:

$$[M_h] = \begin{bmatrix} 2 & 1 & 1 & -2 \\ -3 & -2 & -1 & 3 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$$

The Hermite cubic image curve in normalized coordinates is given by

$$\mathbf{X}(t) = \mathbf{P}(t)/w(t) \tag{5}$$

where $w(t)$ is the norm of $\mathbf{P}(t)$. The weighting factors w_i, w_i' $(i = 1, 2)$ provides additional adjustments to the shape of the cubic curve.

In this manner an analytic spherical motion can be designed using Hermite interpolation based on specifications of a set of discrete orientations together with an instantaneous axis at each orientation. The resulting piecewise Hermite interpolation is automatically C^1 continuous. Using the differential geometry of the image space (McCarthy and Ravani 1986), the instantaneous poles can be chosen such that the interpolated motion possess higher order continuities (Ge and Ravani 1991).

This approach is not restricted to Hermite interpolation. Other forms of interpolation can also be utilized. For example, rather than specifying two displacements together with two instantaneous poles, four displacements can be specified and interpolated using Bézier interpolation. Ge and Ravani (1991) developed a general mathematical framework for designing parametric motions by generalizing techniques in the field of CAGD.

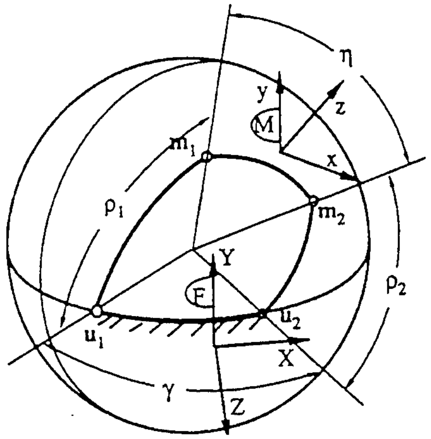


Figure 1: A spherical kite linkage

4 SPHERICAL KITE MOTION

The spherical kite linkage is a spherical four bar mechanism with two pairs of neighboring links having the same angular length, see Figure 1. Let the angular length of each link be denoted by γ for the ground link, η for the coupler and ρ_1, ρ_2 for the two cranks. In this paper it is assumed that $\gamma = \rho_1$ and $\eta = \rho_2$. The kite linkage has two folding positions, the left and the right.

It is known in the theory of kinematic mapping that the image curve for the coupler motion of a spherical kite linkage consists of a line and a cubic curve, see Ge and McCarthy (1991). They correspond to two "modes" of the coupler motion: the line represents the continuous rotation of the coupler about a fixed pivot; and the cubic represents the cubic rotational motion of the coupler. The line and the cubic intersect at two points which correspond to the left and right folding positions.

To derive the constraint equations needed for the design of a kite linkage, we consider the coupler η to be the link common to the two two-link open chains on the left and on the right. We assume the fixed and moving pivots of the linkage to lie on a unit sphere and we attach a moving frame M to the coupler and a fixed frame F to the ground. Let the unit vectors $\mathbf{u}_i = (u_i, v_i, w_i)$ $(i = 1,2)$ define the positions of two fixed pivots in F, and $\mathbf{m}_i = (m_i, n_i, l_i)$ $(i = 1,2)$ define the positions of two moving pivots in M. Let $\mathbf{X} = (X_1, X_2, X_3, X_4)$ denote the coordinates of the image point for a rotation of the coupler link. Then the coupler as the link common to the two open chains is constrained by the following two equations quadratic in $\mathbf{X}$:

$$Q_i \ : \ \mathbf{X}^T[Q_i]\mathbf{X} = 0, \quad i = 1,2 \tag{6}$$

where the coefficient matrices $[Q_i]$ $(i = 1,2)$ are given by

$$[Q_i] = \begin{bmatrix} m_iu_i - n_iv_i - l_iw_i - \cos\rho_i & n_iu_i + m_iv_i & l_iu_i + m_iw_i & n_iw_i - l_iv_i \\ n_iu_i + m_iv_i & -m_iu_i + n_iv_i - l_iw_i - \cos\rho_i & l_iv_i + n_iw_i & l_iu_i - m_iw_i \\ l_iu_i + m_iw_i & l_iv_i + n_iw_i & -m_iu_i - n_iv_i + l_iw_i - \cos\rho_i & m_iv_i - n_iu_i \\ n_iw_i - l_iv_i & l_iu_i - m_iw_i & m_iv_i - n_iu_i & m_iu_i + n_iv_i + l_iw_i - \cos\rho_i \end{bmatrix}. \tag{7}$$

The set of image points $\mathbf{X}$ satisfying (6) are said to define a quadric surface in the image space, see Ravani and Roth (1984). The quadric surface can be further interpreted as a right circular real hyperboloid of one sheet (Ge and McCarthy 1991). The size of the hyperboloid is determined by the crank length ρ_i; the position of the hyperboloid is determined by the locations of the fixed and moving pivots $\mathbf{u}_i$ and $\mathbf{m}_i$.

The intersection of the two hyperboloids defined by equations (6) is the image curve of a general four-bar motion. It is a quartic curve of the first kind (A quartic curve of the second kind lies only on one quadric), see Snyder and Sisam (1941). The problem of dimensional synthesis of a spherical four bar linkage is then translated into a geometric problem of adjusting the sizes and the relative positions of the two hyperboloids such that their intersection approximates the image curve of a desired motion. In the case of a spherical kite linkage, the size of each hyperboloid is not an independent design variable. This is due to the two special conditions among its link dimensions: $\gamma = \rho_1$ and $\eta = \rho_2$. These conditions result in the following relationships among the design variables:

$$\begin{aligned} \cos\rho_1 &= u_1u_2 + v_1v_2 + w_1w_2, \\ \cos\rho_2 &= m_1m_2 + n_1n_2 + l_1l_2. \end{aligned} \tag{8}$$

Therefore the shape and location of the two hyperboloids and thus their intersection curve is determined by eight independent variables from the unit vectors $\mathbf{u}_1$, $\mathbf{u}_2$, $\mathbf{m}_1$ and $\mathbf{m}_2$. The substitution of (8) into (7) results in the equations for two special hyperboloids, denoted by $Q_1(\mathbf{u}_1, \mathbf{m}_1, \mathbf{u}_2)$ and $Q_2(\mathbf{u}_2, \mathbf{m}_2, \mathbf{m}_1)$, respectively. These two special hyperboloids always have a line in common. The rest of their intersection is a twisted cubic curve, denoted as $K(\mathrm{X}; \mathbf{u}_1, \mathbf{u}_2, \mathbf{m}_1, \mathbf{m}_2)$. The problem of dimensional synthesis for a kite linkage is then reduced to that of selecting the unit vectors $\mathbf{u}_i$, $\mathbf{m}_i$ $(i = 1,2)$ such that $K(\mathrm{X}; \mathbf{u}_1, \mathbf{u}_2, \mathbf{m}_1, \mathbf{m}_2)$ approximates the image curve of a desired motion.

5 DESIGN SYNTHESIS OF A KITE LINKAGE

Traditionally, a desired motion is specified by a set of discrete positions so that the actual coupler motion may best guide through them. This specification, however, does not have much global control over the coupler motion. In this section, we apply the Hermite interpolation method for design of an analytic cubic motion. We then use this analytic motion as a desired motion and use a curve fitting procedure to design the linkage parameters. We have chosen the kite spherical four-bar linkage since its image curve is a cubic curve and the Hermite interpolation method that we have developed is also for cubic curves. In theory, the results can be generalized to any linkage, for example, by using Bézier , B-spline or Cardinal spline approximation/interpolation.

We start the specification of a desired motion by a set of image points representing discrcte positions. We then design a cubic curve of desired shape by piecewise Hermite interpolation of these points. The resulting polar points at each given point is then used as additional design specifications for the desired coupler motion.

Let $\mathbf{Y}_i$ $(i = 1, 2, \cdots, n)$ denote a set of image points representing desired orientations, and let $\mathbf{Y}_i'$ denote the polar points representing the desired tangent directions. Then the problem of designing a kite linkage is to choose the unit vectors $\mathbf{u}_1$, $\mathbf{u}_2$, $\mathbf{m}_1$, and $\mathbf{m}_2$ such that at least one measure of the approximation error of $\mathbf{Y}_i$ and $\mathbf{Y}_i'$ from the twisted cubic $K(\mathbf{X}; \mathbf{u}_1, \mathbf{u}_2, \mathbf{m}_1, \mathbf{m}_2)$ is minimized.

Ravani and Roth (1983) defined the approximation error at a desired image point as the normal deviation of the point from the actual image curve and developed a normal curve-fitting technique for motion approximation. In what follows we modify the approximation error defined by Ravani and Roth to include the prescribed tangent requirements.

Let $\mathbf{X}_i$, $\mathbf{X}_i'$ denote, respectively, a point of the twisted cubic K and its polar point with respect to K. Let $\Delta\mathbf{X}_i = \mathbf{X}_i - \mathbf{Y}_i$ and $\Delta\mathbf{X}_i' = \mathbf{X}_i' - \mathbf{Y}_i'$ denote the deviation vectors where $\mathbf{Y}_i$ and $\mathbf{Y_i}'$ are the desired image point and its polar point. The normal deviation defined by Ravani and Roth (1983) satisfy the following linear equation:

$$[A]\Delta\mathbf{X}_i = \mathbf{q}_i \tag{9}$$

where

$$[A] = \begin{bmatrix} 2\mathbf{Y}_i^T[Q_1] \\ 2\mathbf{Y}_i^T[Q_2] \\ 2\mathbf{Y}_i^T \end{bmatrix}$$

and

$$\mathbf{q}_i = [-\mathbf{Y}_i^T[Q_1]\mathbf{Y}_i \ \ -\mathbf{Y}_i^T[Q_2]\mathbf{Y}_i \ 0]^T.$$

Eq.(9) is a linear approximation of K near the desired image point $\mathbf{Y}_i$.

The polar point $\mathbf{X}_i'$ of $\mathbf{X}_i$ with respect to the cubic K is defined by the four dimensional unit vector:

$$\mathbf{X}' = \frac{{}^*([Q_1]\mathbf{X}_i \wedge [Q_2]\mathbf{X}_i \wedge \mathbf{X}_i)}{|{}^*([Q_1]\mathbf{X}_i \wedge [Q_2]\mathbf{X}_i \wedge \mathbf{X}_i)|}. \tag{10}$$

The symbol "*" denotes the star operator for multivectors which transforms, in this case, a multivector of rank 3 into a vector, see Flanders (1963). The symbol $|\cdot|$ denote the Euclidean norm of a vector. By substituting $\mathbf{X}_i' = \mathbf{Y}_i' + \Delta\mathbf{X}_i'$ and $\mathbf{X}_i = \mathbf{Y}_i + \Delta\mathbf{X}_i$ into (10), we expand (10) about the polar point $\mathbf{Y}_i'$ at $\mathbf{Y}_i$. This yields, after discarding higher order terms involving $\Delta\mathbf{X}_i$ and $\Delta\mathbf{X}_i'$,

$${}^*([Q_1]\mathbf{Y}_i \wedge [Q_2]\mathbf{Y}_i \wedge \Delta\mathbf{X}_i) + {}^*(\mathbf{Y}_i \wedge [Q_1]\mathbf{Y}_i \wedge [Q_2]\Delta\mathbf{X}_i) + {}^*([Q_2]\mathbf{Y}_i \wedge \mathbf{Y}_i \wedge [Q_1]\Delta\mathbf{X}_i) + C\Delta\mathbf{X}_i' = \mathbf{p}_i, \tag{11}$$

where

$$\mathbf{p}_i = C\mathbf{Y}_i' - {}^*([Q_1]\mathbf{Y}_i \wedge [Q_2]\mathbf{Y}_i \wedge \mathbf{Y}_i)$$

and

$$C = |^*([Q_1]\mathrm{Y}_i \wedge [Q_2]\mathrm{Y}_i \wedge \mathrm{Y}_i)|.$$

Eq.(11) which approximates the specified polar point can also be rewritten as

$$[B]\Delta\mathrm{X}_i + C\Delta\mathrm{X}'_i = \mathrm{p}_i. \tag{12}$$

Eq.(9) together with (12) is a system of seven linear equations with eight unknowns. The minimum norm solution, $(\Delta\mathrm{X}^*_i, \Delta\mathrm{X}'^*_i)$, of this system is the one that minimizes the following Lagrangian function:

$$L = \Delta\mathrm{X}_i^T\Delta\mathrm{X}_i + \Delta\mathrm{X}'^T_i\Delta\mathrm{X}'_i + \lambda_1^T([A]\Delta\mathrm{X}_i - \mathrm{q}_i) + \lambda_2^T([B]\Delta\mathrm{X}_i + C\Delta\mathrm{X}'_i - \mathrm{p}_i),$$

where λ_1 and λ_2 are vectors of Lagrange's multipliers. The approximation error function at each point Y_i that includes both orientation and tangent requirements is thus defined as

$$e_i = \Delta\mathrm{X}^{*T}_i\Delta\mathrm{X}^*_i + \Delta\mathrm{X}'^{*T}_i\Delta\mathrm{X}'^*_i.$$

The total error is obtained by summing over all points Y_i $(i = 1, 2, \cdots, n)$:

$$E = \sum_{i=1}^{n}(\Delta\mathrm{X}^{*T}_i\Delta\mathrm{X}^*_i + \Delta\mathrm{X}'^{*T}_i\Delta\mathrm{X}'^*_i).$$

Standard routines for nonlinear least squares optimization can then be applied to obtain the design parameters u_i, m_i $(i = 1, 2)$.

CONCLUSIONS

This paper demonstrated that methods from continuous computational geometry can be used with the aid of an orientable kinematic mapping for mechanical motion synthesis. The initial ideas presented form the basis for more research that would bring together the fields of computational geometry and kinematic design.

Acknowledgement

This research was supported by US ARO grant number DAAL03-90-G-0005.

References

1. Bottema, O. and Roth B., 1979, *Theoretical Kinematics*, North Holland Publ., Amsterdam, 558pp.
2. Coxeter, H. S. M., 1961, *Non-Euclidean Geometry*, University of Toronto Press, 309pp.
3. Duff, T., 1986, Splines in animation and modeling, *SIGGRAPH'86 Course #15: State of the art in image synthesis.*

4. Flanders, H., 1963, *Differential Forms with Applications to the Physical Sciences*, Academic Press, New York, 205pp.

5. Ge, Q. J. and McCarthy, J. M., 1991, The algebraic classification of the image curves of spherical four-bar Motion, *ASME J. of Mechanical Design*, 113(3):227-231.

6. Ge, Q. J. and Ravani, B., 1991, Computer aided geometric design of motion interpolants, *Proc. ASME Design Automation Conf.*, DE-Vol. 32-2:33-41, Miami, FL.

7. McCarthy, J. M. and Ravani, B., 1986, Differential kinematics of spherical and spatial motions using kinematic mapping, *ASME J. of Appl. Mech.*, 53:15-22.

8. Pletinckx, D., 1989, Quaternion calculus as a basic tool in computer graphics, *The Visual Computer*, 5:2-13.

9. Ravani, B. and Roth, B., 1983, Motion synthesis using kinematic mappings, *ASME J. of Mech., Transmissions., and Auto. in Design*, 105(3):460-467.

10. Ravani, B. and Roth, B., 1984, Mappings of spatial kinematics, *ASME J. of Mech., Transmissions., and Auto. in Design*, 106(3):341-347.

11. Shoemake, K., 1985, Animating rotation with quaternion curves, *ACM Siggraph*, 19(3):245-254.

12. Shoemake, K., 1987, Quaternion calculus and fast animation, *SIGGRAPH'87 Course #10: Computer Animation: 3D Motion specification and control*, pp. 101-121.

13. Snyder, V. and Sisam, C. H., 1941, *Analytic Geometry of Space*, Henry Holt and co., New York.

14. Study, E., 1891, Von den bewegungen and umlegungen, *Math. Ann.*, 39:441-566.

A Kinematic Design Criterion for Singularity–Avoidance in Redundant Manipulators

R.M. Sardis
B. Ravani
R.M.C. Bodduluri

University of California, Davis

Abstract

In this paper, we study a set of conditions that defines a kinematic design criterion for a redundant robotic manipulator in order that it can generate singularity-free trajectories. The design criterion is enforced on a scalar function on the joint space that is positive everywhere except at singular configurations where it vanishes, and a normal vector function that lifts the paths away from the singular configurations. A theorem is stated specifying the design criterion on the functions and is proved for a general m degree of freedom robot in an n dimensional space. The conditions are global in the sense that when these conditions are met for a given robot, any path in its workspace can be generated avoiding singularities. An example is presented that demonstrates how to identify the functions to evaluate a design.

Keywords: Redundant Robot Manipulators, Design of Robots, Singularities, Singularity–Free Control, Kinematic Control.

1 Introduction

The control of redundant manipulators has become a popular topic of research in the field of robotics. Redundant robot manipulators have been considered the solutions to the problems of trajectory planning which is often plagued by singularities in case of nonredundant robots. The idea is that the available extra degrees of freedom can be used to avoid singularities. In this paper, we develop a kinematic criterion to evaluate the design of a robot in a global sense in terms of its ability to generate singularity-free paths.

Past research in robotics has focussed in developing algorithms to avoid singularities for redundant robots. Whitney (1969) first proposed resolving redundancies in robots using pseudoinverse techniques which were extended by many researchers such as Klein and Huang (1983), Klein (1984), Hollerbach and Suh (1985). In particular, Yoshikawa (1984, 85) defines a 'manipulability index' based on the jacobian of the robot. This index vanishes at the singularities and assumes large values away from them. By maximizing the mainpulability index at every point of the trajectory, his algorithm keeps the robot as far as possible from the singular configurations. Long and Paul (1992) develop an algorithm that treats the redundant robot as a nonredundant robot by locking the extra joints. When the equivalent nonredundant robot reaches its singular configurations, the extra degrees-of-freedom are used to move away from the singularities in an algorithmic fashion. In both these cases, the singularity-avoidance is guaranteed locally if there exists such a path. However, there is no *a priori* information on the existence or nonexistence of such paths, nor there is any singularity-avoidance in a global sense. In this paper, we focus our attention to the conditions that identify robot designs that can always provide singularity-free movements. We show that in our general and global treatment of the problem, we encompass Yoshikawa's result as well.

A designer's approach to avoiding singularities is to design a singularity-free robot. There have been many attempts in the past in designing robots with minimum number of singularities. Soon it became obvious that singularity-free workspaces are nonexistant in nonredundant as well as redundant robots (which is proved using topological arguments by Gottlieb 1986, and followed by Wampler 1987, 89, and Baker and Wampler 1987, 88). However, one can expect that certain designs have the capability of avoiding singularities for any path in the workspace. This paper presents conditions that would allow testing a kinematic design to see if it has the capability to produce, globally, singularity-free trajectories.

In what follows, we first formulate the problem of inverse kinematics of a general m degree of freedom robot as a control problem in the velocity domain. Then we present the functions that play the key role in our result followed by some examples on how to choose these functions. A theorem is then stated and proved that enumerates the conditions these functions have to meet that guarantees the existence of nonsingular paths. A second theorem follows as a modification of the first theorem to ease the computational aspects. Finally, we consider the example of a four jointed spherical wrist (considered by Yoshikawa 1984, 85 and Long and Paul 1992, Long et al 1992) and evalute its design using our kinematic design criterion.

2 The Inverse Kinematics Problem

A robot manipulator consisits of several rigid bodies connected by various types of joints. Several representations have been developed in the robotics literature to represent the interconnections of these rigid bodies, and the relationship of the end-effector to the base of the robot. One of them seemed to have emerged as more popular as well as geometric, 4 × 4 matrices using the Denavit-Hartenberg parameters. We use this representation in what follows, however, the theory is very general and independent of the representation scheme.

Let us consider a robot with m degrees of freedom. Each degree of freedom denotes the capability of one degree of freedom joint. In general, each joint can either be revolute or prismatic. If there are r revolute joints and p prismatic joints where $m = p + r$, and the joint variables are denoted by $q_i, i = 1, 2, \ldots, m$, then the 4×4 transformation matrix that defines the location and orientation of the end-effector to the frame is given by the kinematic equation:

$$T = T_1(q_1)T_2(q_2)\cdots T_m(q_m) \tag{1}$$

where T_i is a 4×4 transformation matrix describing the kinematic relationship between the ith link and its next one, the $(i+1)$th. Its actual form depends on the geometry of the robot, and is given by

$$T_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2}$$

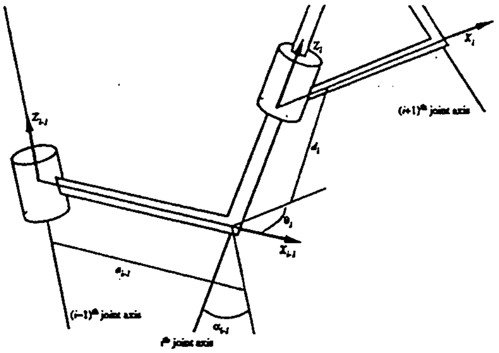


Figure 1: Denavit-Hartenberg parameters associated with a robot.

where a_i, α_i are the link parameters, and d_i, θ_i are the joint parameters, together known as Denavit-Hartenberg Parameters (Figure 1 shows these parameters). The degree of freedom q_i will be equal to θ_i if the ith joint is revolute, or to d_i if it is prismatic.

The matrix T defines the location and orientation of the end-effector of the robot with respect to its base, that is, it involves a 3×3 rotation matrix, and a translation vector. Therefore, this matrix, in general, spans the space of three dimensional displacements, known as *Euclidean Group* $E(3)$, which is a composition of the group of rotations $SO(3)$ and the group of translations $T(3)$, therefore

$$E(3) = SO(3) \times T(3). \tag{3}$$

This group of spatial displacements can be parameterized by six independent parameters, typically by three angles of rotation about independent axes and three translation parameters, and hence represent a 6-dimensional manifold, known as *configuration manifold of E(3)*, or in this case *configuration manifold of the end-effector*. The functions that define the displacement matrix are infinitely differentiable with respect to the parameters, therefore the manifold is smooth.

A set that is both a group and a smooth manifold on which composition and inversion appear as smooth maps is a *Lie group*. The configuration manifold of the end-effector is a group and a smooth manifold, moreover, the composition and inversion are defined in terms of the matrix multiplication and matrix inversion as smooth maps, hence it forms a *compact Lie group*, see Samuel et al (1991) for more details. To be general and to include the cases of planar, spherical, and spatial robots, we note that the configuration manifold can be of any dimension, not necessarily of 6. Let the configuration manifold be n–dimensional and be denoted by N. Similarly, the joint space forms an m–dimensional manifold denoted by M, also a compact Lie group.

Therefore, the kinematic equations (Eq. 1) is a mapping from M to N, or symbolically,

$$y = f(x) \tag{4}$$

where y denotes the position of the end-effector, and x denotes the values of the joint degrees of freedom. The mapping f, known as forward kinematics, relates the joint variables to the end-effector position, and its inverse f^{-1}, inverse kinematics, maps the position of the end-effector to the joint variables. As is very well-known in the robotics literature, the forward kinematics is single-valued, whereas the inverse kinematics is not, due to the non-linearity of the equations, therefore, $f : M \to N$ is an onto mapping.

We differentiate Eq. (4) with respect to time t to obtain

$$\dot{y} = \mathrm{d}f|_x \dot{x} \tag{5}$$

where the $\dot{()}$ denotes the derivative with respect to t and $\mathrm{d}f|_x = \mathrm{d}f/\mathrm{d}x$. Note that now we have transformed the problem into the velocity domain, that is, the mapping $\mathrm{d}f|_x$ relates the points in the tangent space TM of the joint space to the tangent space TN of the configuration space of the end-effector, $\mathrm{d}f|_x : TM \to TN$.

Both Eqs. (4) and (5) represent the inverse kinematics problem. Given a path in $y(t)$ in the configuration space, the goal of inverse kinematics problem is to find $x(t)$ for all t. However, at some locations, the mapping f becomes singular, that is $(\dot{y})$ may not be in the image of $\mathrm{d}f$ as it is not of full rank at singularities. Our goal is to find $x(t)$ such that it does not pass through any of the singularities.

Therefore, we state the problem of inverse kinematics as follows:
Given $y(t) \in N$, find $x(t) = f^{-1}(y(t)) \in M \setminus S$ for all t, where $M \setminus S$ is the joint space excluding the singular configurations S defined as

$$S = \{x \in M \mid \text{rank } (\mathrm{d}f|_x) < n\}.$$

3 Inverse Kinematics as a Control Problem

We transform the above mentioned inverse kinematics problem into a control problem which eventually yields the design criterion we have been looking for. Moreover, this approach also results in a control that avoids singularities, though the topic of this paper is just to present the design criterion.

We first define three functions that are needed in the following theory:

1. a scalar function or a parameter $p : M \to [0, \infty)$ with $p(x) = 0$ if and only if $x \in S$;
2. a vector lifting function $\lambda : (M \setminus S) \times TN \to TM$ such that
   (a) $\lambda(x(t), \dot{y}(t))$ is a right-invariant vector field on M, and
   (b)
   $$\mathrm{d}f|_x(\lambda(x, \dot{y})) = \dot{y}|_{f(x)}; \tag{6}$$
3. $B(t) : [0, \infty) \to TM$ such that
   (a) $B(t)$ is a right-invariant vector field on M, and
   (b)
   $$B(t)|_{x(t)} \in \ker(\mathrm{d}f|_{x(t)}); \tag{7}$$

The right-invariance of the vector fields λ and B is not necessary for the proofs of the theorems presented in this paper. However, they are useful in developing the actual control that tracks singularity-free trajectories, which is the subject of a future paper.

The inverse kinematics problem is to be able to identify $\dot{x}(t)$ at every instant, therefore, $\dot{x}(t)$ can be expressed as

$$\dot{x}(t) = \lambda(x(t), \dot{y}(t)) + B(t). \tag{8}$$

Substituting for $\dot{x}$ from this equation in the right hand side of Eq. (5), we see that

$$\mathrm{d}f|_x \dot{x}(t) = \mathrm{d}f|_x \lambda(x(t), \dot{y}(t)) + \mathrm{d}f|_x B(t)$$

which can be simplified using the definitions of the functions (Eqs. 6 and 7) to

$$\mathrm{d}f|_x \dot{x}(t) = \dot{y}(t) + 0.$$

Therefore, the solution $x(t)$ of the differential equation (8) is in fact the solution of the inverse kinematics problem specified in Eq. (5).

Our goal in solving the inverse kinematics problem is to find the control $B(t)$ such that the solution $x(t) \in M \setminus S$ starting from $x(0) = f^{-1}(y(0))$ for all t.

Examples of Choice of p and λ

Example 1: Define

$$\begin{aligned} p(x) &= \det(J(x)J^T(x)), \\ \lambda(x,\dot{y}) &= J^+\dot{y}, \\ B(t) &= b(t)(I - J^+J)\frac{\mathrm{d}p}{\mathrm{d}x}, \end{aligned}$$

where $J^+ = J^T(JJ^T)^{-1}$ and J is the jacobian of the robot. Note that the scalar function p is zero only when $x \in S$ and otherwise is positive. The vector function λ satisfies its properties as well. This, as a matter of fact, is Yoshikawa's algorithm.
Example 2: The following gives a more computationally efficient method of finding p and λ:

- define $p(x) = |\nu_x|$ where ν_x is the smallest eigenvalue of JJ^T;
- define λ by solving the equation $\mathrm{d}f|_x(\lambda(x,\dot{y})) = \dot{y}$ by singular value decomposition.

p is now only continuous, not smooth. λ is the same as before, but however this definition does not require the computation of the matrix J^+, and is defined even when $x \in S$.

4 Conditions for the Existence of Singularity-Free Paths

The following theorem lists the conditons on the existence of singularity-free paths for a given design of robot:

Theorem 1 *Let $f : M \to N$, $p : M \to [0,\infty)$, and λ as before, and assume that f satisfies the following three conditions:*

1. *$\sigma > 0$;*
2. *there is an integer μ such that $\pi_0 M(y) = \mu$ for $y \in N$;*
3. *there is an open neighborhood W of the identity element of M such that for any $y \in N$ and any $x \in M$, the translate xW of W intersects $M(y)$ in at most one point;*

where σ and $M(y)$ are defined by

$$\sigma := \inf_{y\in N}\{\sup_{\xi\in f^{-1}(y)} p(\xi)\}$$

and

$$M(y) := \{x \in f^{-1}(y) \mid p(x) = \sup_{\xi\in f^{-1}(y)} p(\xi)\}$$

for each $y \in N$, and where $\pi_0(M(y))$ is the number of path componants of the set $M(y)$.

Let $\dot{y} : [0,\infty) \to TN$ be a continuous function. Then, given any $\epsilon > 0$, there is a control vector $B(t)$ such that if x_0 is in $M(f(x_0))$ then the solution to Eq. (8) satisfies $p(x(t)) > (1-\epsilon)\sigma$, for all t.

The first of the conditions is a necessary condition for the existence of a singularity-free solution. σ will be strictly greater than 0 if and only if, for every $y \in N$, $f^{-1}(y)$ contains points lying outside the sigular set S, as at the singularities p is identically equal to zero as per its definition. The second condition requires that the number of maximal points of p restricted to $f^{-1}(y)$ be constant for all $y \in N$. This number is the number of paths available as the nonsingular paths. Finally, the third condition guarantees that these different nonsingular paths are nonintersecting which guarantees that there is tubular range around each of these paths that are nonsingular.

When all these conditions are met, the theorem states that starting from $x_0 \in f^{-1}(y_0)$ one can follow a path arbitraily close to $p(x) = \sigma$ for all $y \in N$. A brief proof is presented in the Appendix A (see Sardis 1988 for a more rigorous proof).

Note that when a robot satisfies these conditions for any arbitrary choice of the functions p and λ, the robot is guaranteed to follow *any* trajectory in its workspace avoiding singularities.

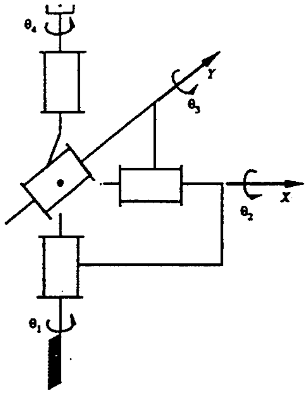


Figure 2: Schematic of a four jointed spherical wrist.

Therefore, these conditions can be considered as the criterion to evaluate the design of an existing robot or as the design criterion to be used in the process of designing a new robot.

In practice, the hypotheses of Theorem 1 are too restrictive for typical real robots; they can, however, be relaxed somewhat as we show in the following Theorem.

Theorem 2 *Suppose $M = M_1 \cup M_2$ with p constant on M_2 and $f(M_2)$ contained in a submanifold of N of dimension at most (m-2). Write $N_1 = f(M_1)$. Define σ as before and assume*

1. *$\sigma > 0$;*
2. *$\pi_0 M(y) = \mu$ for all $y \in N_1$;*
3. *there is an open neighborhood W of the idenitity element of M such that for any $y \in N_1$ and $x \in M$, the translate xW of W intersects $M(y)$ in at most one point.*

Then, given $y(t)$ and $\epsilon > 0$, there is a control vector $B(t)$ such that if $x_0 \in M(f(x))$, the solution to Eq. (8) satisfies $p(x(t)) > (1-\epsilon)\sigma$ for all t.

Notice that this theorem is a slight modification of the earlier one. The second and third conditions deal with only part of the configuration manifold of the end-effector instead of the whole manifold, that is, N_1 instead of N. This is possible because the parameter p is constant on M_2. We do not present the proof for this theorem in this paper, however it is similar to the proof of Theorem 1. The following examples demonstrate how this theorem can be used.

5 An Example

Figure 2 shows a four jointed spherical wrist. Note that the four axes intersect at one point which is known as the wrist center point (WCP). Therefore, the last link (the end-effector) only rotates (does not translate) with respect to the first link (the base). The configuration manifold is hence $SO(3)$ and is of 3 dimensions. The transformation that denotes the position of the end-effector relative to the base is given by

$$\begin{aligned} y = &\begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 \\ \sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_2 & -\sin\theta_2 \\ 0 & \sin\theta_2 & \cos\theta_2 \end{bmatrix} \\ &\begin{bmatrix} \cos\theta_3 & 0 & \sin\theta_3 \\ 0 & 1 & 0 \\ -\sin\theta_3 & 0 & \cos\theta_3 \end{bmatrix} \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 \\ \sin\theta_4 & \cos\theta_4 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (9) \end{aligned}$$

The right hand side of the above equation denotes the mapping $f : M \to N$ where $x = (\theta_1, \theta_2, \theta_3, \theta_4)$.

We now need to define the parameter p, it has to be positive valued everywhere in M except at the singularities where it vanishes. Singularities occur for this robot when two pairs of joint axes line up, that is when TN loses its rank from 3 to 2.

We will use a different parameter than those given by earlier examples. Choose unit vectors a_1, a_2, a_3, and a_4 parallel to the joints of the wrist, and define

$$p(x) = \max(|a_1 \times a_3|, |a_2 \times a_4|) \qquad (10)$$

where $a_i, i = 1,2,3,4$ are funtions of x. Note that this expression becomes zero only when a_1 is parallel to a_3 and a_2 to a_4, otherwise it is equal to the maximum of the absolute values of the sines of the angles between the two pairs of the axes. This is the same condition for the singularities as the four axes becoming coplanar.

For a given y and a value of θ_1, the inverse kinematics problem yields two solutions, analogous to elbow-up and elbow-down configurations of a three link manipulator. Therefore, as we vary θ_1 through its range, the inverse kinematics solution of the four degree of freedom spherical wrist are obtained as two curves in the joint space, both parameterized by θ_1. These are also known as fibers associated with the mapping $f^{-1} : N \to M$. Note that $p(x)$ attains maximum when the angles between a_1, a_3, and a_2, a_4 become $\pm 90°$, which implies that $\sigma = 1$. Hence, the function $p(x)$ remains constant and equal to 1 if a_1 is parallel to a_4, and otherwise attains four local maxima on each fiber.

Now, if y is an orientation having a_1 parallel to a_4 then $p(f^{-1}(y))$ is constant and non-zero, therefore we can define

$$\begin{aligned} M_2 &= \{x \mid a_1 \times a_4 = 0\}, \\ M_1 &= M \setminus M_2. \end{aligned}$$

It can be seen that p has a constant value 1 on all of M_2, and $f(M_2)$ has the dimension 1 as it is the set of all rotations about a_1 or a_4.

$\mu = \pi_0 M(y)$ is the number of local maxima for all $y \in N_1$ which is four on each of the two fibers, that is $\mu = 8$. Finally, if we define

$$W = \{x \mid |\theta_1| < 45°, |\theta_2| < 90°\}$$

the third condition of Thereom 2 is satisfied as well. Therefore, according to Theorem 2, given any smooth trajectory $y(t)$, we can track $y(t)$ by a continuous change of configuration $x(t)$ which will keep $p(x)$ arbitraily close to 1, that is, the wrist will never encounter a singular configuration.

6 Conclusion

In this paper, we have formulated the inverse kinematics problem in a controls setting. In this setting, our goal is to investigate if the redundant robot can track any trajectory in its workspace avoiding the singularities. Two theorems specify the conditions that globally guarantee the existence of such paths regardless of the path of the end-effector under consideration. We have also applied the theorems to the four-jointed spherical wrist that has appeared in the literature before.

The existence of the control vector $B(t)$ is guaranteed to avoid the singularities, however, we haven't determined the control $B(t)$ that actually guides the robot on the desired trajectory. This will be the topic of a future paper.

Acknowledgement

This work was supported in part by US Army Grant No. DAAL03-90-G0005, and in part by California Department of Transportation (CALTRANS), Advanced Highway Maintenance and Construction Technology (AHMCT) program.

A Proof of Theorem 1

First we consider the following two Lemmas.

Lemma 1 *Suppose F is a smooth manifold, $X_0(t), \ldots, X_k(t)$ smooth vector fields on F, and assume that $X_0(t), \ldots, X_k(t)$ span the tangent space $T_x F$ for all $x \in F$ and for all values of t. Let U be an open set in R^n and let $\phi_t : U \to F$ be a continuous one-parameter family of open maps. Assume $x_0 \in \phi_0(U)$. Then there are continuous control parameters $u_j(t)$ such that the solution to*

$$\begin{aligned} \dot{x} &= X_0(t) + u_1(t)X_1(t) + \cdots + u_k(t)X_k(t) \\ x(0) &= x_0 \end{aligned}$$

satisfies $x(t) \in \phi_t(U)$ for all $t > 0$.

Proof: Without loss of generality, $X_0(t) = 0$ since the X_j span the tangent space of F. Take $z \in U$ such that $\phi_0(z) = x_0$ and define a curve $\alpha(t) := \phi_t(z)$. Then $\alpha(t)$ is a continuous curve, with $\alpha(t) \in \phi_t(U)$. Since $\phi_t(U)$ is open and $\phi_t(z)$ is continuous in t, α can be approximated by a smooth curve $x(t)$ with $x(0) = x_0$ and $x(t) \in \phi_t(U)$. Define $u_i(t) := X_i(t)(x(t))$; then $\dot{x}(t) = \sum u_i(t)X_i(t)$, since X_i span $T_x F$.

The connection between Lemma 1 and Theorem 1 is given by the following Lemma and Definition.

Lemma 2 *For any $\delta > 0$ there is an open neighborhood V of 1 in M, such that for any x in M and any $z \in xV$, $|p(x) - p(z)| < \delta$.*

Proof: First, for every $x \in M$ take a neighborhood W_x of 1 with $|p(x) - p(z)| < \delta/2$ whenever $z \in x(W_x)^2$; this is possible since p is a continouous function. Then $\{xW_x\}_{x \in M}$ is an open cover of M, and M is compact; so there is a finite subcover $\{x_1W_1, \ldots, x_kW_k\}$. Define $V := W_1 \bigcap \ldots \bigcap W_k$. Now take an x in M and a point z in xV. $x \in x_jW_j$ for some j; and so $z \in (x_jW_j)V \leq x_j(W_j)^2$. Then

$$|p(x) - p(z) \leq |p(x) - p(x_j)| + |p(x_j) - p(z)| \leq \delta/2 + \delta/2 = \delta.$$

Let $f : M \to N$ and λ be as before with γ a smooth path in N. Pick a time $t \in [0, \infty)$ and let F be a subset of $f^{-1}(t)$.

Assume that F does not contain any singular points. Let I_t be a subinterval of $[0, \infty)$ containing t. Then if I_t is small enough, for any $x \in F$ we can lift $\gamma[I_t]$ to a path

$$\begin{aligned} \bar{\gamma}_x(0) &= x \\ \dot{\bar{\gamma}}_x(0) &= \lambda(\bar{\gamma}_x(t), \dot{\gamma}(t)). \end{aligned}$$

This gives a map $\psi : F \times I_t \to M$ by $\psi(x,s) = \bar{\gamma}_x(s)$. Denote the image of ψ by T_t.

Definition 1 *T_t is said to be a* tubular interval *about F (with respect to f, λ, and γ) if $\psi : F \times I_t \to M$ is a homeomorphism.*

Proof of Theorem 1: It is sufficient to prove the result for t in an arbitrary compact interval $[0, \tau]$.

Let $K = \{(x,t) \mid t \in [0,\tau], x \in M(y(t))$, that is K is the set of all maximal points of $p(x)$ in the fibers over y in the space $M \times [0, \tau]$. K is a closed set. Define $\mathrm{proj}_2 : M \times [0,\tau] \to [0,\tau]$ by projection.

Suppose K_i is a connected component of K; by the third condition of the theorem, proj_2 is one-to-one on K_i, since $\bigcup_{x \in K_i} xU$ covers K_i. Also, K_i is closed since K is. So proj_2 gives a μ-fold cover of $[0, \tau$ by closed sets.

Now let K_0 be the componant of K containing the point $(x_0, 0)$; we want to show that $\mathrm{proj}_2(K_0)$ is all of $[0,\tau]$. So, suppose that $\mathrm{proj}_2(K_0) = [0, \tau_0]$ for some $\tau_0 \leq \tau$. Define K_1 to be the union of all other path componants of K containing a point in $f^{-1}(\tau_0)x\{\tau_0\}$. But then

$$\begin{aligned} R_1 &:= \mathrm{proj}_2(K_0) \\ R_2 &:= \mathrm{proj}_2(K \setminus K_1) \setminus (\tau_0, \tau) \end{aligned}$$

are disjoint closed sets whose union is all of $[0,\tau]$; so since $[0,\tau]$ is connected, $\mathrm{proj}_2(K_0)$ is all of $[0,\tau]$.

Now, since $\mathrm{proj}_2(K_0) = [0,\tau]$ we can define a continuous function $\alpha : [0,\tau] \to M$ by the relation $(\alpha(t), t) \in K_0$ for $t \in [0,\tau]$.

We now define a set U and collections of smooth manifolds F_i and open maps $\phi^i_t : U \to F_i$, and apply Lemma 1.

First, by Lemma 2, choose a neighborhood V of 1 in M such that for any $x \in M$, $|p(x) - p(z)| < \sigma\epsilon$ whenever $z \in xV$. In particular, if $z \in \alpha(t)V$ for any t then

$$\begin{aligned} p(z) &\geq p(\alpha(t)) - \epsilon\sigma \\ &\geq \sigma - \epsilon\sigma \\ &= (1-\epsilon)\sigma. \end{aligned}$$

Let U be an open set with $1 \in U$ and the closure of U contained in V.

For every $t \in [0,t]$ choose an open interval I_t containing t such that I_t defines a tubular interval T_t about $\alpha(t)V \bigcap f^{-1}(y(t))$, with I_t sufficiently small that $\alpha(s)U \bigcap f^{-1}(y(s))$ is in T_t for all $s \in I_t$. Then I_t is an open cover of the compact interval $[0,\tau$; take a finite subcover $I_{t1}, I_{t2}, \ldots, I_{tk}$, with $0 = t_0 < t_1 \cdots < t_k = \tau$.

Define $F_i := \alpha(t_i)V \bigcap f^{-1}(y(t_i))$ for $i = 0, 1, \ldots, k$. Also, define $\phi^i_t : U \to F_i$ for $t \in [t_i, t_{i+1}]$ by $\phi^i_t(w) := g_i(\alpha(t)w)$ where $g_i(t) : T_{ti} \to f^{-1}(y(t))$ by projection in T_{ti}, for $t \in [t_i, t_{i+1}]$.

For $t \in [t_i, t_{i+1}]$ define $X_0(t)$ on F_i to be the projection of $\mathrm{d}g_i(\lambda)$. Pick vector fields $X_1, \ldots, X_k$ spanning F_i.

Now define $z(t) := g_i(ti)x(t)$; then z(t) is a function with image lying in F_i for $t \in [t_i, t_{i+1}]$. By Lemma 1 there are scalaras $u_1, \ldots, u_k$ such that the solution to $\dot{z} = X_0 + \sum u_i X_i$ lies inside $\phi^i_t(U)$ for all $t \in [t_i, t_{i+1}]$. Define $B(t) \in T(\alpha(t)V \bigcap f^{-1}(y(t)))$ by

$$B(t) := \mathrm{d}g_i(t)(\sum u_i X_i).$$

Then the solution $x(t)$ to the equation $\dot{x}(t) = \lambda(x(t), \dot{y}(t)) + B(t)$ satisfies $g_i(x(t)) \in \phi^i_t(U)$ for all $t \in [t_i, t_{i+1}]$, or equivalently,

$$x(t) \in \alpha(t)U$$

for all $t \in [t_i, t_{i+1}]$. Thus

$$p(x(t)) \geq (1-\epsilon)\sigma$$

for all $t \in [t_i, t_{i+1}]$. Since this can be done for each $i = 0, 1, \ldots, k$, we have $p(x(t)) \geq (1-\epsilon)\sigma$ for all t.

References

Baker, D.R. and Wampler II, C.W., 1987, "Some Facts Concerning the Inverse Kinematics of Redundant Manipulators," *Proceedings of 1987 IEEE International Conference on Robotics and Automation*, pages 604–609.

Baker, D.R. and Wampler II, C.W., 1988, "On the Inverse Kinematics of Redundant Manipulators," *The International Journal of Robotics Research*, 7(2):3–21.

Forsythe, G., Malcolm, M., and Moler, C., 1977, *Computer Methods for Mathematical Computations*, Prentice-Hall.

Gottlieb, D., "Robots and Topology," *Proceedings of 1986 IEEE International Conference on robotics and Automation*, 1689-91.

Hollerbach, J.M. and Suh, K.C., 1985, "Redundancy Resolution of Manipulators Through Torque Optimization," *Proceedings of 1985 IEEE International Conference on Robotics and Automation*, pages 1016–1021.

Klein, C.A., 1984, "Use of Redundancy in the Design of Robotic Systems," *Robotics Research: The Second International Symposium*, pages 207–214.

Klein, C.A. and Huang, C.H., 1983, "Reveiew of Pseudoinverse Control for use with kinematically redundant manipulators," *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13(3):245-250.

Long, G.L. and Paul, R.P., 1992, "Singularity Avoidance and the Control of an Eight Revolute Jointed Manipulator," *International Journal of Robotics Research*, to appear.

Long G.L., McCarthy, J.M., and Paul, R.P., 1992, "Kinematic Modules for Singularity Free Movement with Three Cartesian Freedoms," *ASME Journal of Mechanical Design*, to appear.

Samuel, A.E., McAree, P.R., and Hunt, K.H., 1991, "Unifying Screw Geometry and Matrix Transformations," *International Journal of Robtoics Research*, 10(5):454-472.

Sardis, R.M., 1988, *Robotic Singularities and Control*, PhD Dissertation, University of Wisconsin-Madison.

Varadarajan, V.S., 1974, *Lie Groups, Lie Algebras, and Their Representations*, Prentice-Hall.

Wampler II, C.W., 1987, "Inverse Kinematic Functions for Redundant Manipulators," *Proceedings of 1987 IEEE International Conference on Robotics and Automation*, pages 610-617.

Wampler II, C.W., 1989, "Inverse Kinematic Functions for Redundant Spherical Wrists," *IEEE Transactions on Robotics and Automation*, 5(1):106-111.

Yoshikawa, Tsueno, 1984, "Analysis and Control of Robot Manipulators with Redundancy," Robotics Research, Brady and Paul, Editors, MIT Press, Cambridge, Mass.

Yoshikawa, Tsueno, 1985, "Manipulability and Redundancy Control of Robotic Mechanisms," *IEEE Conference on Robotics.*

V. Parenti-Castelli and J. Lenarcic (eds.)

Third International Workshop on

Advances in Robot Kinematics

September 7-9, 1992, Ferrara, Italy

Coordinate System Independent Form of Instantaneous Invariants in Spatial Kinematics

C. Lee

A. T. Yang

B. Ravani

Computer Integrated Design and
Manufacturing Laboratory,
Department of Mechanical Engineering,
University of California, Davis,
Davis, CA 95616

This paper presents explicit equations for the instantaneous invariants of spatial kinematics that are coordinate systems independent. This eliminates the need for laborious coordinate transformations necessary in the determination of the canonical coordinate systems. It provides for a direct method for the calculation of instantaneous invariants based on information specifying the rigid body motion in any arbitrary task coordinate systems. Explicit equations are presented for instantaneous invariants up to the third order for spatial and spherical motions and up to the fourth order for planar motions. The results provide for a useful tool in design and analysis of mechanisms and motions based on instantaneous invariants. Examples are presented to illustrate the theory.

Introduction

Instantaneous invariants and canonical coordinate systems, since their introduction by Bottema (1961) and Veldkamp (1967 and 1976), have become a powerful tool in differential kinematic analysis and synthesis. Roth and Yang (1977) and Gupta (1978), among others, have applied them to analysis and synthesis of planar mechanisms. Kirson and Yang (1978) as well as McCarthy and Roth (1982), Nayark and Roth (1981) have extended the concept of instantaneous invariants and canonical coordinate systems to a study of spatial rigid body motions and mechanisms. A detailed account of the concept of instantaneous invariants and the canonical coordinate systems for planar, spherical and spatial motions can be found in Bottema and Roth (1979). In robotics literature, Stanišić and Pennock (1986) have applied the concept to the study of differential motions of robot manipulators.

All existing work on determination of instantaneous invariants require the use of the so-called canonical coordinate systems. This usually involves tedious coordinate transformations necessary to represent the specified quantities in the canonical coordinate systems. In this paper, we present concise expressions for the instantaneous invariants using an arbitrary set of moving and fixed coordinate systems. This eliminates the need for the computations associated with the use of the canonical coordinate systems. The derivations are based upon the concept of kinematic mapping introduced by Ravani and Roth (1984). A rigid body motion,[1] specified in any coordinate system, can be mapped into an image curve in the space of the mapping. The mapping is defined such that the intrinsic properties of the image curve are invariant with respect to coordinate transformations. This property then enables us to calculate the instantaneous invariants directly from the intrinsic properties of the resulting image curve. Although the intrinsic properties of an image curve can be used to characterize the corresponding rigid body motion instantaneously (see McCarthy and Ravani, 1986), the use of the instantaneous invariants has been (at least up to the present) more familiar to kinematicians. This is why we are presenting the explicit expressions for the instantaneous invariants. The main contribution of this work is therefore in providing a set of explicit expressions for computations of the instantaneous invariants of a given rigid body motion without the need for the use of the canonical coordinate systems. This is useful for design and analysis of mechanisms as well as comparing rigid body motions. Expressions are presented for the instantaneous invariants of spatial and spherical kinematics up to the third order and for the planar motions up to the fourth order. Numerical examples are used to illustrate the results.

The organization of the paper is as follows: first, we give a very brief overview of kinematic mapping and derive the governing equations in terms of the 3×3 dual matrix representation of spatial displacements. We then derive the differential properties of the image curve of a motion in terms of the differential values of the elements of the 3×3 dual matrix representation of spatial displacements. This enables us, in a subsequent section, to directly arrive at the expressions for the dual instantaneous invariants (up to the third order) that are independent of the canonical coordinate systems. We then provide two examples, an epicyclic hypoid gear train and a RCCC mechanism, to illustrate the theory for spatial motions.

[1] In this work we are only considering single degree of freedom motions.

Contributed by the Mechanisms Committee and presented at the Design Technical Conference, Chicago, IL, Sept. 16-19, 1990, of THE AMERICAN SOCIETY OF MECHANICAL ENGINEERS. Manuscript received March 1990.

The spherical results can fall directly from our spatial results using the principle of transference (Kotelnikov, 1895, and Study, 1903). We have therefore eliminated any further discussions for spherical motions. In the case of planar motions, we specialize the mapping and develop the results with the same detail as in the general spatial case. Since the mathematics is less tedious in this case, we develop the instantaneous invariants up to the fourth order and use a numerical example for illustrative purpose.

Notations

A *dual number* is denoted by an angle symbol on top of a letter. For example, $\hat{\beta} = b + \epsilon \mathring{b}$ in which b and $\mathring{b}$ are referred to, respectively, as the real part and dual part of $\hat{\beta}$. The symbol ϵ designates the dual unit with the property $\epsilon^2 = 0$. In the study of instantaneous kinematics we are interested in the motion of a rigid body at a particular instant, referred to as the zero-position (Veldkamp, 1967). The order of the time derivative of functions evaluated at zero position (t = 0), is denoted by the last subscript. For example $\mathbf{R}_2 = \ddot{\mathbf{R}}(0) = [X_{12}, X_{22}, X_{32}, X_{42}]^T$, $\hat{\beta}_{32} = \ddot{\hat{\beta}}_3(0)$, $\alpha_{21} = \dot{\alpha}_2(0)$, etc. The value specified in canonical system is denoted by a tilde symbol under a letter such as $\underset{\sim}{\hat{\alpha}}_{23}$.

Kinematic Mapping

According to Chasles theorem, the displacement of a rigid body can be uniquely described by rotating a dual angle $\hat{\phi}$ about a screw axis $\hat{\mathbf{S}} = (\hat{S}_x, \hat{S}_y, \hat{S}_z)$. The rigid body displacement can then be mapped into a point in the dual image space $\hat{\Sigma}$ (see Ravani and Roth, 1984). The coordinates of the point are defined by

$$\hat{\mathbf{R}} = (\hat{X}_1, \hat{X}_2, \hat{X}_3, \hat{X}_4) = \left(\hat{S}_x \mathrm{Sin}\,\frac{\hat{\phi}}{2}, \hat{S}_y \mathrm{Sin}\,\frac{\hat{\phi}}{2}, \hat{S}_z \mathrm{Sin}\,\frac{\hat{\phi}}{2}, \mathrm{Cos}\,\frac{\hat{\phi}}{2}\right) \qquad (1)$$

Consider a rigid body in continuous motion in Euclidean space. In such a case, the screw axis $\hat{\mathbf{S}}$ and the dual angle $\hat{\phi}$ are functions of time t. Equation (1) will then represent a dual curve in terms of the real parameter t in the dual image space $\hat{\Sigma}$. We refer to the image curve $\hat{\mathbf{R}}(t)$ as a dual image curve in the dual image space $\hat{\Sigma}$.

Let $\mathfrak{M}$ be an arbitrary chosen coordinate system attached to the moving body and $\mathfrak{F}$ be a system fixed in Euclidean space. The transformation from $\mathfrak{M}$ to $\mathfrak{F}$, which describes the rigid body motion, may be expressed as a dual matrix (Veldkamp, 1976)

$$\hat{\mathbf{A}} = \begin{bmatrix} \hat{\alpha}_1 & \hat{\beta}_1 & \hat{\gamma}_1 \\ \hat{\alpha}_2 & \hat{\beta}_2 & \hat{\gamma}_2 \\ \hat{\alpha}_3 & \hat{\beta}_3 & \hat{\gamma}_3 \end{bmatrix} \qquad (2)$$

where the dual-elements are continuous functions of the real parameter t.

Using Rodrigues formula, we may show that the dual angle $\hat{\phi}$ and the screw axis $\hat{\mathbf{S}} = (\hat{S}_x, \hat{S}_y, \hat{S}_z)$ may be expressed as the elements of the dual matrix $\hat{\mathbf{A}}$:

$$\mathrm{Cos}\hat{\phi} = \frac{1}{2}(\hat{\alpha}_1 + \hat{\beta}_2 + \hat{\gamma}_3 - 1), \quad \hat{S}_x = \frac{1}{2\mathrm{Sin}\hat{\phi}}(\hat{\beta}_3 - \hat{\gamma}_2),$$
$$\hat{S}_y = \frac{1}{2\mathrm{Sin}\hat{\phi}}(\hat{\gamma}_1 - \hat{\alpha}_3), \quad \hat{S}_z = \frac{1}{2\mathrm{Sin}\hat{\phi}}(\hat{\alpha}_2 - \hat{\beta}_1) \qquad (3)$$

Substituting Eq. (3) into Eq. (1), we have the parametric form of the image curve $\hat{\mathbf{R}}(t)$ written in terms of the elements of the dual matrix $\hat{\mathbf{A}}$:

$$\hat{\mathbf{R}} = \begin{bmatrix} \hat{X}_1 \\ \hat{X}_2 \\ \hat{X}_3 \\ \hat{X}_4 \end{bmatrix} = \begin{bmatrix} \dfrac{\hat{\beta}_3 - \hat{\gamma}_2}{4\hat{X}_4} \\ \dfrac{\hat{\gamma}_1 - \hat{\alpha}_3}{4\hat{X}_4} \\ \dfrac{\hat{\alpha}_2 - \hat{\beta}_1}{4\hat{X}_4} \\ \dfrac{1}{2}\sqrt{\hat{\alpha}_1 + \hat{\beta}_2 + \hat{\gamma}_3 + 1} \end{bmatrix} \qquad (4)$$

Differential Properties of Image Curves

In this section we review the differential properties of an image curve which are independent of the coordinate systems in Euclidean space. These geometric quantities are then used to derive the explicit expressions of the instantaneous invariants. The formal proof of the independent coordinate properties of the kinematic mapping is given in Ravani and Roth (1984). For a description of the differential geometry of the image space, readers may refer to McCarthy and Ravani (1986) and McCarthy (1986 and 1987).

Using prime to denote differentiation with respect to the dual arc-length $\hat{q}$, we may write the Frenet formulas for the base frame $(\hat{\mathbf{T}}, \hat{\mathbf{N}}, \hat{\mathbf{B}}, \hat{\mathbf{E}})$ as

$$\begin{bmatrix} \hat{\mathbf{T}}' \\ \hat{\mathbf{N}}' \\ \hat{\mathbf{B}}' \\ \hat{\mathbf{E}}' \end{bmatrix} = \begin{bmatrix} 0 & \hat{\kappa} & 0 & -1 \\ -\hat{\kappa} & 0 & \hat{\tau} & 0 \\ 0 & -\hat{\tau} & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \hat{\mathbf{T}} \\ \hat{\mathbf{N}} \\ \hat{\mathbf{B}} \\ \hat{\mathbf{E}} \end{bmatrix} \qquad (5)$$

With the aid of Eq. (5), we obtain the derivatives of the image curve $\hat{\mathbf{R}}$ with respect to t, namely:

$$\hat{\mathbf{R}} = \hat{\mathbf{E}}$$
$$\dot{\hat{\mathbf{R}}} = \hat{\upsilon}\hat{\mathbf{T}}$$
$$\ddot{\hat{\mathbf{R}}} = \dot{\hat{\upsilon}}\hat{\mathbf{T}} + \hat{\upsilon}^2\hat{\kappa}\hat{\mathbf{N}} - \hat{\upsilon}^2\hat{\mathbf{E}} \qquad (6)$$
$$\dddot{\hat{\mathbf{R}}} = [\ddot{\hat{\upsilon}} - (1 + \hat{\kappa}^2)\hat{\upsilon}^3]\hat{\mathbf{T}} + (3\hat{\upsilon}\dot{\hat{\upsilon}}\hat{\kappa} + \hat{\upsilon}^3\dot{\hat{\kappa}})\hat{\mathbf{N}} + \hat{\upsilon}^3\hat{\kappa}\hat{\tau}\hat{\mathbf{B}} - 3\hat{\upsilon}\dot{\hat{\upsilon}}\hat{\mathbf{E}}$$

where dot denotes differentiation with respect to the parameter t and $\hat{\upsilon} = \dot{\hat{q}} = \left|\dfrac{d\hat{\mathbf{R}}}{dt}\right|$.

From Eq. (6) we observe that the dual functions $\hat{\upsilon}$, $\hat{\kappa}$ and $\hat{\tau}$ characterize the differential geometry of $\hat{\mathbf{R}}(t)$. Using the notations

$$\hat{\mathbf{R}}(0) = \hat{\mathbf{R}}_0, \quad \left.\frac{d^n\hat{\mathbf{R}}}{dt^n}\right|_{t=0} = \hat{\mathbf{R}}_n$$
$$\hat{\upsilon}(0) = \hat{\upsilon}_0, \quad \left.\frac{d^n\hat{\upsilon}}{dt^n}\right|_{t=0} = \hat{\upsilon}_n \quad \text{etc.}$$

We may write the Taylor expansion of the column vector $\hat{\mathbf{R}}(t)$ at a reference point (t = 0) as

$$\hat{\mathbf{R}}(t) = \hat{\mathbf{R}}_0 + \hat{\mathbf{R}}_1 t + \hat{\mathbf{R}}_2\frac{t^2}{2} + \hat{\mathbf{R}}_3\frac{t^3}{6} + \ldots..$$

Using Eq. (6) the above equation may be written as

$$\hat{\mathbf{R}}(t) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} \hat{\upsilon}_0 \\ 0 \\ 0 \\ 0 \end{bmatrix} t + \begin{bmatrix} \hat{\upsilon}_1 \\ \hat{\upsilon}_0^2\hat{\kappa}_0 \\ 0 \\ -\hat{\upsilon}_0^2 \end{bmatrix} \frac{t^2}{2} + \begin{bmatrix} \hat{\upsilon}_2 - \hat{\upsilon}_0^3(1 + \hat{\kappa}_0^2) \\ 3\hat{\upsilon}_0\hat{\kappa}_0\hat{\upsilon}_1 + \hat{\upsilon}_0^3\hat{\kappa}_1 \\ \hat{\upsilon}_0^3\hat{\kappa}_0\hat{\tau}_0 \\ -3\hat{\upsilon}_0\hat{\upsilon}_1 \end{bmatrix} \frac{t^3}{6} \qquad (7)$$

The above equation is an intrinsic representation of the image curve up to the third order. The values of $\hat{\upsilon}_0$, $\hat{\upsilon}_1$, $\hat{\upsilon}_2$, $\hat{\kappa}_0$, $\hat{\tau}_0$, $\hat{\kappa}_1$ in the above equation can be written in terms of the dot and the wedge products of $\hat{\mathbf{R}}_j$(j = 0, 1, 2, 3), namely

$$\begin{cases} \hat{v}_0 = (\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_1)^{1/2} \\ \hat{v}_1 = \dfrac{\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2}{\hat{v}_0} \\ \hat{v}_2 = \dfrac{\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_3 + \hat{\mathbf{R}}_2 \cdot \hat{\mathbf{R}}_2}{\hat{v}_0} - \dfrac{(\hat{\mathbf{R}}_1 \cdot \hat{\mathbf{R}}_2)^2}{\hat{v}_0^3} \\ \hat{\kappa}_0 = \dfrac{[(\hat{\mathbf{R}}_1 \wedge \hat{\mathbf{R}}_2 \wedge \hat{\mathbf{R}}_0) \cdot (\hat{\mathbf{R}}_1 \wedge \hat{\mathbf{R}}_2 \wedge \hat{\mathbf{R}}_0)]^{1/2}}{\hat{v}_0^3} \\ \hat{\tau}_0 = \dfrac{\det(\hat{\mathbf{R}}_1, \hat{\mathbf{R}}_2, \hat{\mathbf{R}}_3, \hat{\mathbf{R}}_0)}{\hat{v}_0^6 \hat{\kappa}_0^2} \\ \hat{\kappa}_1 = \dfrac{(\hat{\mathbf{R}}_1 \wedge \hat{\mathbf{R}}_2 \wedge \hat{\mathbf{R}}_0) \cdot (\hat{\mathbf{R}}_1 \wedge \hat{\mathbf{R}}_3 \wedge \hat{\mathbf{R}}_0)}{\hat{v}_0^7 \hat{\kappa}_0} - 3 \dfrac{\hat{v}_1}{\hat{v}_0^2} \hat{\kappa}_0 \end{cases} \tag{8}$$

where

$$(\hat{\mathbf{R}}_1 \wedge \hat{\mathbf{R}}_2 \wedge \hat{\mathbf{R}}_0)_m \cdot (\hat{\mathbf{R}}_1 \wedge \hat{\mathbf{R}}_3 \wedge \hat{\mathbf{R}}_0)_n = \hat{M}^m_{123} \hat{M}^n_{123} + \hat{M}^m_{234} \hat{M}^n_{234} + \hat{M}^m_{134} \hat{M}^n_{134} + \hat{M}^m_{124} \hat{M}^n_{124}$$

$\hat{M}^m_{ijk} = \begin{vmatrix} \hat{X}_{i1} & \hat{X}_{j1} & \hat{X}_{k1} \\ \hat{X}_{i2} & \hat{X}_{j2} & \hat{X}_{k2} \\ \hat{X}_{i0} & \hat{X}_{j0} & \hat{X}_{k0} \end{vmatrix}$ and $\hat{M}^n_{ijk} = \begin{vmatrix} \hat{X}_{i1} & \hat{X}_{j1} & \hat{X}_{k1} \\ \hat{X}_{i3} & \hat{X}_{j3} & \hat{X}_{k3} \\ \hat{X}_{i0} & \hat{X}_{j0} & \hat{X}_{k0} \end{vmatrix}$ are the minors of the matrix $[\hat{\mathbf{R}}_1, \hat{\mathbf{R}}_2, \hat{\mathbf{R}}_0]^T$ and matrix $[\hat{\mathbf{R}}_1, \hat{\mathbf{R}}_3, \hat{\mathbf{R}}_0]^T$ respectively, and det $(\hat{\mathbf{R}}_1, \hat{\mathbf{R}}_2, \hat{\mathbf{R}}_3, \hat{\mathbf{R}}_0)$ is the determinant of the matrix $[\hat{\mathbf{R}}_1, \hat{\mathbf{R}}_2, \hat{\mathbf{R}}_3, \hat{\mathbf{R}}_0]^T$.

The detail of operations of wedge product in general form can be found in Flanders (1963) and McCarthy (1987).

The values of $\hat{\mathbf{R}}_n$ can be determined by differentiating Eq. (4) with respect to t and evaluating them at t = 0. We show the formulas for n = 0, 1, 2, 3, 4.

$$\begin{cases} \hat{X}_{i1} = \dfrac{1}{\hat{X}_{40}} \left(\dfrac{\hat{\eta}_{i1}}{4} - \hat{X}_{41} \hat{X}_{i0} \right) \\ \hat{X}_{i2} = \dfrac{1}{\hat{X}_{40}} \left(\dfrac{\hat{\eta}_{i2}}{4} - 2\hat{X}_{41} \hat{X}_{i1} - \hat{X}_{42} \hat{X}_{i0} \right) \\ \hat{X}_{i3} = \dfrac{1}{\hat{X}_{40}} \left(\dfrac{\hat{\eta}_{i3}}{4} - 3\hat{X}_{41} \hat{X}_{i2} - 3\hat{X}_{42} \hat{X}_{i1} - \hat{X}_{43} \hat{X}_{i0} \right) \\ \hat{X}_{i4} = \dfrac{1}{\hat{X}_{40}} \left(\dfrac{\hat{\eta}_{i4}}{4} - 4\hat{X}_{41} \hat{X}_{i3} - 6\hat{X}_{42} \hat{X}_{i2} - 4\hat{X}_{43} \hat{X}_{i1} - \hat{X}_{44} \hat{X}_{i0} \right) \end{cases} \tag{9}$$

where i = 1, 2, 3 and

$$\hat{\eta}_{1j} = \hat{\beta}_{3j} - \hat{\gamma}_{2j},$$

$$\hat{\eta}_{2j} = \hat{\gamma}_{1j} - \hat{\alpha}_{3j},$$

$$\hat{\eta}_{3j} = \hat{\alpha}_{2j} - \hat{\beta}_{1j}$$

j = 1, 2, 3, 4

and

$$\begin{cases} \hat{X}_{41} = \dfrac{1}{8\hat{X}_{40}} (\hat{\alpha}_{11} + \hat{\beta}_{21} + \hat{\gamma}_{31}) \\ \hat{X}_{42} = \dfrac{1}{8\hat{X}_{40}} (\hat{\alpha}_{12} + \hat{\beta}_{22} + \hat{\gamma}_{32} - 8\hat{X}_{41}^2) \\ \hat{X}_{43} = \dfrac{1}{8\hat{X}_{40}} (\hat{\alpha}_{13} + \hat{\beta}_{23} + \hat{\gamma}_{33} - 24\hat{X}_{41}\hat{X}_{42}) \\ \hat{X}_{44} = \dfrac{1}{8\hat{X}_{40}} (\hat{\alpha}_{14} + \hat{\beta}_{24} + \hat{\gamma}_{34} - 24\hat{X}_{42}^2 - 32\hat{X}_{41}\hat{X}_{43}) \end{cases} \tag{10}$$

Determination of Instantaneous Invariants

The dual matrices $\underset{\sim}{\hat{\mathbf{B}}}_n$ of the canonical systems in spatial kinematics were derived by Kirson and Yang (1978). They are the transformation matrices between the moving and fixed canonical systems.

$$\underset{\sim}{\hat{\mathbf{B}}}_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \underset{\sim}{\hat{\mathbf{B}}}_1 = \begin{bmatrix} 0 & -\underset{\sim}{\hat{\alpha}}_{21} & 0 \\ \underset{\sim}{\hat{\alpha}}_{21} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{11}$$

$$\begin{gathered} \underset{\sim}{\hat{\mathbf{B}}}_2 = \begin{bmatrix} -\underset{\sim}{\hat{\alpha}}_{21}^2 & -\underset{\sim}{\hat{\alpha}}_{22} & 0 \\ \underset{\sim}{\hat{\alpha}}_{22} & -\underset{\sim}{\hat{\alpha}}_{21}^2 & -\underset{\sim}{\hat{\beta}}_{32} \\ 0 & \underset{\sim}{\hat{\beta}}_{32} & 0 \end{bmatrix} \\ \underset{\sim}{\hat{\mathbf{B}}}_3 = \begin{bmatrix} -3\underset{\sim}{\hat{\alpha}}_{21}\underset{\sim}{\hat{\alpha}}_{22} & -\underset{\sim}{\hat{\alpha}}_{23} & \underset{\sim}{\hat{\gamma}}_{13} \\ \underset{\sim}{\hat{\alpha}}_{23} & -3\underset{\sim}{\hat{\alpha}}_{21}\underset{\sim}{\hat{\alpha}}_{22} & -\underset{\sim}{\hat{\beta}}_{33} \\ -\underset{\sim}{\hat{\gamma}}_{13} + 3\underset{\sim}{\hat{\alpha}}_{21}\underset{\sim}{\hat{\beta}}_{32} & \underset{\sim}{\hat{\beta}}_{33} & 0 \end{bmatrix} \end{gathered} \tag{12}$$

where $\underset{\sim}{\hat{\alpha}}_{21}$, $\underset{\sim}{\hat{\alpha}}_{22}$, $\underset{\sim}{\hat{\alpha}}_{23}$, $\underset{\sim}{\hat{\beta}}_{32}$, $\underset{\sim}{\hat{\beta}}_{33}$, $\underset{\sim}{\hat{\gamma}}_{13}$ are the dual instantaneous invariants of a given spatial rigid body motion at the zero position and up to the third order.

Substituting Eqs. (11) and (12) into (9) and (10), we have $\underset{\sim}{\hat{\mathbf{R}}}_0 = [0, 0, 0, 1]^T$ and

$$\underset{\sim}{\hat{\mathbf{R}}}_1 = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{2}\underset{\sim}{\hat{\alpha}}_{21} \\ 0 \end{bmatrix}, \quad \underset{\sim}{\hat{\mathbf{R}}}_2 = \begin{bmatrix} \frac{1}{2}\underset{\sim}{\hat{\beta}}_{32} \\ 0 \\ \frac{1}{2}\underset{\sim}{\hat{\alpha}}_{22} \\ -\frac{1}{4}\underset{\sim}{\hat{\alpha}}_{21}^2 \end{bmatrix}, \quad \underset{\sim}{\hat{\mathbf{R}}}_3 = \begin{bmatrix} \frac{1}{2}\underset{\sim}{\hat{\beta}}_{33} \\ \frac{1}{2}\underset{\sim}{\hat{\gamma}}_{13} - \frac{3}{4}\underset{\sim}{\hat{\alpha}}_{21}\underset{\sim}{\hat{\beta}}_{32} \\ \frac{1}{2}\underset{\sim}{\hat{\alpha}}_{23} + \frac{3}{8}\underset{\sim}{\hat{\alpha}}_{21}^3 \\ -\frac{3}{4}\underset{\sim}{\hat{\alpha}}_{21}\underset{\sim}{\hat{\alpha}}_{22} \end{bmatrix} \tag{13}$$

Substituting the above $\underset{\sim}{\hat{\mathbf{R}}}_0$, $\underset{\sim}{\hat{\mathbf{R}}}_1$, $\underset{\sim}{\hat{\mathbf{R}}}_2$ and $\underset{\sim}{\hat{\mathbf{R}}}_3$ into Eq. (8), we have the dual instantaneous invariants in terms of functions of $\hat{\mathbf{R}}_n$ $(n = 0, 1, 2, 3)$ in an arbitrary pair of coordinate systems. These equations are:

$$\underset{\sim}{\hat{\alpha}}_{21} = 2\left(\sum_{i=1}^{4} \hat{X}_{i1}^2 \right)^{1/2} \tag{14}$$

$$\underset{\sim}{\hat{\alpha}}_{22} = \frac{4}{\underset{\sim}{\hat{\alpha}}_{21}} \sum_{i=1}^{4} (\hat{X}_{i1}\hat{X}_{i2}) \tag{15}$$

$$\underset{\sim}{\hat{\beta}}_{32}=\frac{4}{\underset{\sim}{\hat{\alpha}}_{21}}\left[(\hat{M}^m_{123})^2+(\hat{M}^m_{234})^2+(\hat{M}^m_{134})^2+(\hat{M}^m_{124})^2\right]^{1/2} \quad (16)$$

$$\underset{\sim}{\hat{\alpha}}_{23}=\frac{4}{\underset{\sim}{\hat{\alpha}}_{21}}\left[\sum_{i=1}^{4}\hat{X}_{i1}\hat{X}_{i3}+\sum_{i=1}^{4}\hat{X}^2_{i2}\right]-\underset{\sim}{\hat{\alpha}}^3_{21}-\frac{\underset{\sim}{\hat{\beta}}^2_{32}}{\underset{\sim}{\hat{\alpha}}_{21}}-\frac{\underset{\sim}{\hat{\alpha}}^2_{22}}{\underset{\sim}{\hat{\alpha}}_{21}} \quad (17)$$

$$\underset{\sim}{\hat{\gamma}}_{13}=\frac{-8}{\underset{\sim}{\hat{\alpha}}_{21}\underset{\sim}{\hat{\beta}}_{32}}\begin{vmatrix}\hat{X}_{11} & \hat{X}_{21} & \hat{X}_{31} & \hat{X}_{41}\\ \hat{X}_{12} & \hat{X}_{22} & \hat{X}_{32} & \hat{X}_{42}\\ \hat{X}_{13} & \hat{X}_{23} & \hat{X}_{33} & \hat{X}_{43}\\ \hat{X}_{10} & \hat{X}_{20} & \hat{X}_{30} & \hat{X}_{40}\end{vmatrix}+\frac{3}{2}\underset{\sim}{\hat{\alpha}}_{21}\underset{\sim}{\hat{\beta}}_{32} \quad (18)$$

$$\underset{\sim}{\hat{\beta}}_{33}=\frac{16}{\underset{\sim}{\hat{\alpha}}^2_{21}\underset{\sim}{\hat{\beta}}_{32}}\left(\hat{M}^m_{123}\hat{M}^n_{123}+\hat{M}^m_{234}\hat{M}^n_{234}\right.$$

$$\left.+\hat{M}^m_{134}\hat{M}^n_{134}+\hat{M}^m_{124}\hat{M}^n_{124}\right) \quad (19)$$

where $\hat{M}^m_{ijk}=\begin{vmatrix}\hat{X}_{i1} & \hat{X}_{j1} & \hat{X}_{k1}\\ \hat{X}_{i2} & \hat{X}_{j2} & \hat{X}_{k2}\\ \hat{X}_{i0} & \hat{X}_{j0} & \hat{X}_{k0}\end{vmatrix}$ and $\hat{M}^n_{ijk}=\begin{vmatrix}\hat{X}_{i1} & \hat{X}_{j1} & \hat{X}_{k1}\\ \hat{X}_{i3} & \hat{X}_{j3} & \hat{X}_{k3}\\ \hat{X}_{i0} & \hat{X}_{j0} & \hat{X}_{k0}\end{vmatrix}$

Equations (14–19) show the expressions of dual instantaneous invariants up to the third order for a given rigid body motion in an arbitrary coordinate system. It is clear that if the matrix $\hat{\mathbf{A}}$ in Eq. (2) is given then $\hat{\mathbf{R}}_n$ (n = 0, 1, 2, 3) can be calculated from Eqs. (9 and 10). The instantaneous invariants, up to the third order, can then be obtained directly from Eqs. (14–19).

For the case of a spherical motion, the image curve becomes a real image curve. The procedure to derive the expressions of instantaneous invariants for spherical motions can be obtained by removing the dual components as stated by the principle of Transference (Kotelnikoṽ, 1895, and Study, 1903) and the results are as follows:

$$\underset{\sim}{\beta}_{32}=\frac{1}{2\left(\sum_{i=1}^{4}X^2_{i1}\right)^{3/2}}\left[(M^m_{123})^2+(M^m_{234})^2\right.$$

$$\left.+(M^m_{134})^2+(M^m_{124})^2\right]^{1/2} \quad (20)$$

$$\underset{\sim}{\gamma}_{13}=\frac{3}{2}\underset{\sim}{\beta}_{32}-\frac{2\det(\mathbf{R}_1,\mathbf{R}_2,\mathbf{R}_3,\mathbf{R}_0)}{\underset{\sim}{\beta}^3_{32}\left(\sum_{i=1}^{4}X^2_{i1}\right)^3} \quad (21)$$

$$\underset{\sim}{\beta}_{33}=\frac{(M^m_{123}M^n_{123}+M^m_{234}M^n_{234}+M^m_{134}M^n_{134}+M^m_{124}M^n_{124})}{8\underset{\sim}{\beta}_{32}\left(\sum_{i=1}^{4}X^2_{i1}\right)^{7/2}}$$

$$-\frac{3\underset{\sim}{\beta}_{32}\sum_{i=1}^{4}(X_{i1}X_{i2})}{2\left(\sum_{i=1}^{4}X^2_{i1}\right)^{3/2}} \quad (22)$$

Numerical Examples

Example 1. Figure 1 shows an epicyclic hypoid gear train consisting of a pair of meshing hypoid gears (fixed sun gears 1, and planet gear 2) connected by a twisted rigid member (carrier 3). Determine the instantaneous invariants which characterize the geometric properties of the motion of the planet gear 2 up to the third order.

As shown in Fig. 1, we use a moving coordinate system {**b**} which is attached to the planetary gear 2 and a coordinate system {**f**} fixed to the sun gear 1. The dimensions of the hypoid gear train is given by the dual angle $\hat{\delta}=\delta+\epsilon\,\mathring{\delta}$ between the two gear axes $\hat{\mathbf{f}}_3$ and $\hat{\mathbf{b}}_3$. The line of contact of the two hypoid gears forms the dual angle $\hat{\zeta}=\zeta+\epsilon\,\mathring{\zeta}$ with $\hat{\mathbf{f}}_3$ and dual angle $\hat{\eta}=\eta+\epsilon\,\mathring{\eta}$ with $\hat{\mathbf{b}}_3$. $\dot{\theta}$ and $\dot{\phi}$ are the constant angular velocities of the planet gear and the carrier, respectively.

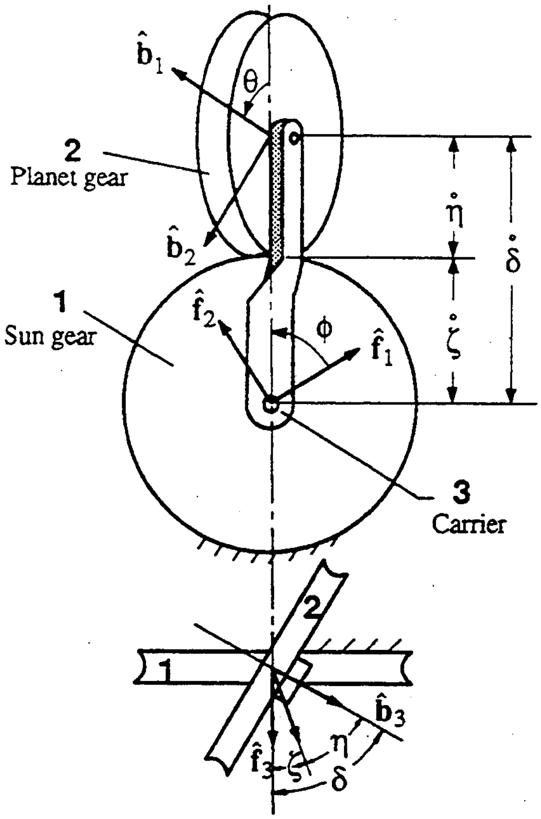


Fig. 1 Epicyclic hypoid gear train

If the gear ratio $\frac{\text{Sin}\zeta}{\text{Sin}\eta}=2$, $\hat{\delta}$ = 60 deg. + ϵ 3, the zero position $\theta_0=\phi_0=0$, and the angular velocities $\phi_1=\frac{\text{Sin}\zeta}{\text{Sin}\delta}=$ 0.37796 and $\theta_1=2\,\phi_1$ (see Kirson, 1975, and Hsia, 1979). The dual matrix $\hat{\mathbf{A}}$ defining {**b**} relative to {**f**} is

$$\hat{\mathbf{A}}=\begin{bmatrix}C\phi & -S\phi & 0\\ S\phi & C\phi & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}1 & 0 & 0\\ 0 & C\hat{\delta} & -S\hat{\delta}\\ 0 & S\hat{\delta} & C\hat{\delta}\end{bmatrix}\begin{bmatrix}C\theta & -S\theta & 0\\ S\theta & C\theta & 0\\ 0 & 0 & 1\end{bmatrix}$$

$$=\begin{bmatrix}C\phi C\theta-S\phi S\theta C\hat{\delta} & -C\phi S\theta-S\phi C\theta C\hat{\delta} & S\phi S\hat{\delta}\\ S\phi C\theta+C\phi S\theta C\hat{\delta} & -S\phi S\theta+C\phi C\theta C\hat{\delta} & -C\phi S\hat{\delta}\\ S\theta S\hat{\delta} & C\theta S\hat{\delta} & C\hat{\delta}\end{bmatrix} \quad (23)$$

where Cθ = Cosθ, Sφ = Sinφ, etc.

From the dual matrix $\hat{\mathbf{A}}$ in Eq. (23) and the given data, we can calculate the $\hat{\mathbf{R}}_n$ (n = 0, 1, 2, 3) from Eqs. (9 and 10). Substituting the values of $\hat{\mathbf{R}}_n$(n = 0, 1, 2, 3) into Eqs. (14–19), we obtain the instantaneous invariants of the planet gear as follows:

$\underset{\sim}{\hat{\alpha}}_{21}=1-\epsilon\,0.74231$ $\quad$ $\underset{\sim}{\hat{\beta}}_{32}=0.24744+\epsilon\,0.42861$

$\underset{\sim}{\hat{\alpha}}_{22}=0$ $\quad$ $\underset{\sim}{\hat{\gamma}}_{13}=0.31813+\epsilon\,0.23622$

$\underset{\sim}{\hat{\alpha}}_{23}=-1.06122+\epsilon\,1.96931$ $\quad$ $\underset{\sim}{\hat{\beta}}_{33}=0$

The above values are in agreement with those given in Kirson (1975) and Hsia (1979).

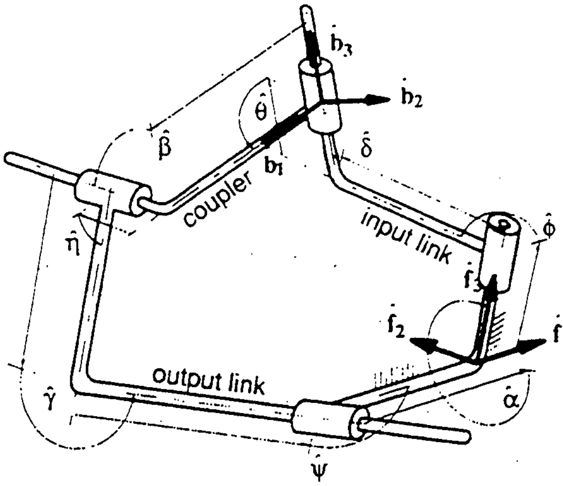


Fig. 2 RCCC mechanism

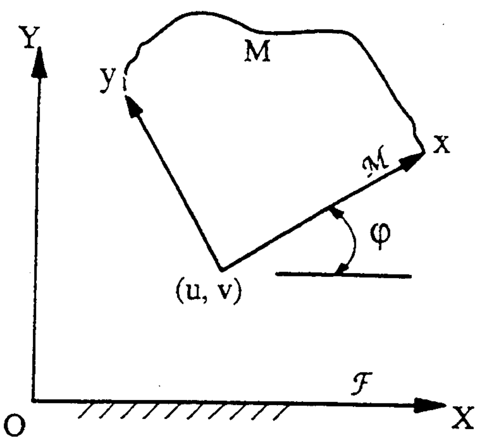


Fig. 3 Arbitrary fixed and moving coordinates $\mathfrak{F}$ and $\mathfrak{M}$

Example 2. The RCCC mechanism (one revolute pair and three cylindrical pairs) is shown in Fig. 2. The dimensions of the links are specified by four constant dual angles $\hat{\alpha}$, $\hat{\beta}$, $\hat{\gamma}$, and $\hat{\delta}$. The relative positions of the links are defined by four time dependent dual angles: $\hat{\phi}$ (dual part is constant since the input member is a revolute pair), $\hat{\theta}$, $\hat{\eta}$, and $\hat{\psi}$. The angle ϕ is the input and $\hat{\psi}$ the output of the mechanism. Determine the third order instantaneous invariants of the coupler link corresponding to a given value of input angle ϕ_0.

We choose the body system $\{\mathbf{b}\}$ attached to the coupler link and the fixed system $\{\mathbf{f}\}$ attached to the frame as shown in Fig. 2. The displacement, velocity, and acceleration of the coupler link as functions of input angle ϕ are derived in Hsia (1979). The numerical zero-position values of the dual parameters are given as follows:

$$\hat{\delta}=30^\circ+\epsilon 2,$$

$$\hat{\theta}_0=86.6^\circ-\epsilon\ 2.9591 \qquad \phi_0=0.$$

$$\hat{\theta}_1=-1.9917-\epsilon\ 2.3066 \qquad \phi_1=1.8155.$$

$$\hat{\theta}_2=0.05274+\epsilon\ 11.8924 \qquad \phi_2=0.2443.$$

$$\hat{\theta}_3=-7.51-\epsilon\ 13.77 \qquad \phi_3=7.99.$$

The dual matrix $\hat{\mathbf{A}}$ **defining** $\{\mathbf{b}\}$ relative to $\{\mathbf{f}\}$ is

$$\hat{\mathbf{A}}=\begin{bmatrix} C\phi & -S\phi & 0 \\ S\phi & C\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & C\hat{\delta} & -S\hat{\delta} \\ 0 & S\hat{\delta} & C\hat{\delta} \end{bmatrix}\begin{bmatrix} C\hat{\theta} & -S\hat{\theta} & 0 \\ S\hat{\theta} & C\hat{\theta} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$=\begin{bmatrix} C\phi C\hat{\theta}-S\phi S\hat{\theta}C\hat{\delta} & -C\phi S\hat{\theta}-S\phi C\hat{\theta}C\hat{\delta} & S\phi S\hat{\delta} \\ S\phi C\hat{\theta}+C\phi S\hat{\theta}C\hat{\delta} & -S\phi S\hat{\theta}+C\phi C\hat{\theta}C\hat{\delta} & -C\phi S\hat{\delta} \\ S\hat{\theta}S\hat{\delta} & C\hat{\theta}S\hat{\delta} & C\hat{\delta} \end{bmatrix} \qquad (24)$$

From the dual matrix $\hat{\mathbf{A}}$ in Eq. (24) and the given data, we can calculate the $\hat{\mathbf{R}}_n$ (n = 0, 1, 2, 3) from Eqs. (9, 10). Substituting the values of $\hat{\mathbf{R}}_n$ (n = 0, 1, 2, 3) into Eqs. (14–19), we obtain the dual instantaneous invariants of the coupler link, up to the third order, as follows:

$$\underline{\hat{\alpha}}_{21}=1+\epsilon\ 4.58358 \qquad \underline{\hat{\beta}}_{32}=1.83126+\epsilon\ 9.96317$$

$$\underline{\hat{\alpha}}_{22}=-\epsilon\ 5.21866 \qquad \underline{\hat{\gamma}}_{13}=3.22836+\epsilon\ 15.25479$$

$$\underline{\hat{\alpha}}_{23}=0.3954-\epsilon\ 10.0621 \qquad \underline{\hat{\beta}}_{33}=0.37465-\epsilon\ 14.20382$$

The above results are verified by using different coordinate systems in describing the dual matrix $\hat{\mathbf{A}}$.

Instantaneous Invariants for Planar Motion

For the case of planar motions, the components of $\hat{\mathbf{R}}(\hat{X}_1, \hat{X}_2)$ will become pure dual numbers. The determination of the explicit expressions for instantaneous invariants needs some special treatment. It means that we have to deal with the real part and the dual part of the curvature and the torsion separately. Furthermore, the value of the det $(\hat{\mathbf{R}}_1, \hat{\mathbf{R}}_2, \hat{\mathbf{R}}_3, \hat{\mathbf{R}}_0)$ will vanish due to the existence of ϵ^2 term. The torsion of the image curve is therefore obtained by a modified method. We describe the derivation of instantaneous invariants for planar motion as follows:

Consider a plane M in continuous motion relative to a fixed plane F. Let us choose, arbitrarily, two coordinate systems; $\mathfrak{M}$ (attached to M) and $\mathfrak{F}$ (fixed in F), as shown in Fig. 3. The transformation matrix of $\mathfrak{M}$ relative to $\mathfrak{F}$ is given by the matrix:

$$\hat{\mathbf{A}}=\begin{bmatrix} C\varphi & -S\varphi & 0 \\ S\varphi & C\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}+\epsilon\begin{bmatrix} 0 & 0 & v \\ 0 & 0 & -u \\ -v & u & 0 \end{bmatrix}\begin{bmatrix} C\varphi & -S\varphi & 0 \\ S\varphi & C\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$=\begin{bmatrix} C\varphi & -S\varphi & \epsilon v \\ S\varphi & C\varphi & -\epsilon u \\ \epsilon(uS\varphi-vC\varphi) & \epsilon(uC\varphi+vS\varphi) & 1 \end{bmatrix} \qquad (25)$$

where $\mathbf{d}$ = $[u, v, 0]^T$ is the position vector of the origin of $\mathfrak{M}$ with respect to $\mathfrak{F}$ and φ is the rotation angle of $\mathfrak{M}$ relative to $\mathfrak{F}$.

We may express $\mathbf{d}$ as a function of the angle φ. Consider a planar motion given as functions of $u(\varphi)$ and $v(\varphi)$ in an arbitrary chosen coordinate system up to the fourth order.

$$u(\varphi)=u_0+u_1\varphi+\frac{1}{2}u_2\varphi^2+\frac{1}{6}u_3\varphi^3+\frac{1}{24}u_4\varphi^4+o(\varphi^5)$$

$$v(\varphi)=v_0+v_1\varphi+\frac{1}{2}v_2\varphi^2+\frac{1}{6}v_3\varphi^3+\frac{1}{24}v_4\varphi^4+o(\varphi^5)$$

where $\qquad u_i=\left.\frac{d^iu}{d\varphi^i}\right|_{\varphi=o}$, and $v_i=\left.\frac{d^iv}{d\varphi^i}\right|_{\varphi=o}$

From Eqs. (4) and (25), we have the representation of the image curve as

$$\hat{\mathbf{R}} = \begin{bmatrix} \hat{X}_1 \\ \hat{X}_2 \\ \hat{X}_3 \\ \hat{X}_4 \end{bmatrix} = \begin{bmatrix} \frac{1}{2}\epsilon\left(\mathrm{u}\,\mathrm{Cos}\,\frac{\varphi}{2} + \mathrm{v}\,\mathrm{Sin}\,\frac{\varphi}{2}\right) \\ \frac{1}{2}\epsilon\left(\mathrm{v}\,\mathrm{Cos}\,\frac{\varphi}{2} - \mathrm{u}\,\mathrm{Sin}\,\frac{\varphi}{2}\right) \\ \mathrm{Sin}\,\frac{\varphi}{2} \\ \mathrm{Cos}\,\frac{\varphi}{2} \end{bmatrix} \tag{26}$$

Using Eqs. (9, 10, and 26), we can determine the values of $\hat{\mathbf{R}}_n$ (n = 0, 1, 2, 3, 4) at the reference position. Let $\hat{v} = v + \epsilon\,\mathring{v}$, $\hat{\kappa} = \kappa + \epsilon\,\mathring{\kappa}$ and $\hat{\tau} = \tau + \epsilon\,\mathring{\tau}$, the instantaneous invariant b_2 can be obtained from the dual part of the curvature $\hat{\kappa}$.

From the Eq. (8), it follows that

$$[(\hat{\mathbf{R}}_1 \wedge \hat{\mathbf{R}}_2 \wedge \hat{\mathbf{R}}_0) \bullet (\hat{\mathbf{R}}_1 \wedge \hat{\mathbf{R}}_2 \wedge \hat{\mathbf{R}}_0)]^{1/2} = [v_0^3\kappa_0 + \epsilon(3v_0^2\mathring{v}_0\kappa_0 + v_0^3\mathring{\kappa}_0)]$$

Now substituting the values of $\hat{\mathbf{R}}_n$ at the reference position in the above equation, we have $v_0^3\,\kappa_o = 0$ and

$$(v_0^3\mathring{\kappa}_0)^2 = \frac{1}{16}\,[(v_1 + u_2)^2 + (u_1 - v_2)^2]$$

But $v_0 = \frac{1}{2}$, we have $\kappa_0 = 0$ and

$$\mathring{\kappa}_0^2 = 4[(v_1 + u_2)^2 + (u_1 - v_2)^2]$$

If we substitute the instantaneous invariants of the canonical system ($a_0 = b_0 = a_1 = b_1 = a_2 = 0$) into this last equation, we have the formula for the instantaneous invariants b_2.

$$b_2 = \sqrt{(v_1 + u_2)^2 + (u_1 - v_2)^2} \tag{27}$$

From Eq. (26), we observe that $\hat{X}_1$ and $\hat{X}_2$ are pure dual numbers. The evaluation of det ($\hat{\mathbf{R}}_1$, $\hat{\mathbf{R}}_2$, $\hat{\mathbf{R}}_3$, $\hat{\mathbf{R}}_0$) involves only the ϵ^2 term. If we remove the dual of $\hat{X}_1$ and $\hat{X}_2$, det ($\hat{\mathbf{R}}_1$, $\hat{\mathbf{R}}_2$, $\hat{\mathbf{R}}_3$, $\hat{\mathbf{R}}_0$) will be a real number only. The real part of torsion can be obtained from the ϵ^2 term of the $\hat{v}_0^6\hat{\kappa}_0^2\hat{\tau}_0$, which is $v_0^6\mathring{\kappa}_0^2\tau_0$, and we have

$$\tau_0 = \frac{\det\left(\frac{\hat{\mathbf{R}}_1}{\epsilon}, \frac{\hat{\mathbf{R}}_2}{\epsilon}, \hat{\mathbf{R}}_3, \hat{\mathbf{R}}_0\right)}{v_0^6\mathring{\kappa}_0^2} \tag{28}$$

From the values of $\hat{\mathbf{R}}_n$ that we have already determined, we have

$$\det\left(\frac{\hat{\mathbf{R}}_1}{\epsilon}, \frac{\hat{\mathbf{R}}_2}{\epsilon}, \hat{\mathbf{R}}_3, \hat{\mathbf{R}}_0\right) = -\frac{1}{16}\,[v_1^2 + u_1^2 + 4u_2v_1 - 4u_1v_2 - 2v_1v_3 + 3v_2^2 - 2u_1u_3 + 3u_2^2 + 2u_3v_2 - 2u_2v_3] \tag{29}$$

Similarly, we substitute the instantaneous invariants of the canonical system ($a_0 = b_0 = a_1 = b_1 = a_2 = 0$) into Eqs. (28) and (29) to obtain

$$\tau_0 = -3 - 2\,\frac{a_3}{b_2} \quad \text{or} \quad a_3 = -\frac{1}{2}\,\tau_0 b_2 - \frac{3}{2}\,b_2 \tag{30}$$

From Eqs. (28, 29, and 30), we have the formula for the instantaneous invariant a_3.

$$a_3 = \frac{1}{2b_2}\,[v_1^2 + u_1^2 + 4u_2v_1 - 4u_1v_2 - 2v_1v_3 + 3v_2^2 - 2u_1u_3 + 3u_2^2 + 2u_3v_2 - 2u_2v_3] - \frac{3}{2}\,b_2 \tag{31}$$

Using the same method, we can determine the instantaneous invariant b_3 by evaluating the change of curvature and we obtain $\hat{\kappa}_1 = \epsilon 4b_3$ and

$$b_3 = \frac{1}{b_2}\,(v_1v_2 + u_1u_2 + u_3v_1 - u_1v_3 + v_2v_3 + u_2u_3) \tag{32}$$

In a similar fashion, we determine the instantaneous invariants a_4 and b_4 by evaluating the change of torsion and change of change of curvature, respectively. We obtain the explicit formulas for a_4 and b_4 as follows:

$$a_4 = \frac{1}{b_2}\,[v_1v_2 + u_1u_2 + 2(u_3v_1 - u_1v_3 + v_2v_3 + u_2u_3) - u_1u_4 - v_1v_4 + u_4v_2 - u_2v_4] - 2b_3 \tag{33}$$

$$b_4 = \frac{1}{b_2}\,[u_2^2 + v_2^2 + v_1v_3 + u_1u_3 + u_3v_2 - u_2v_3 + u_4v_1 - u_1v_4 + u_2u_4 + v_2v_4 + u_3^2 + v_3^2] - \left(\frac{b_3^2}{2b_2} + b_2 + a_3 + \frac{a_3^2}{b_2}\right) \tag{34}$$

Numerical Example

In the following example we use the notations $\dot{u}(0) = u_1^*$, $\dddot{v}(0) = v_3^*$, $\ddot{\varphi}(0) = \varphi_2$ etc. **Example:** Design a four-bar linkage so that the input crank O_bB rotates at a constant angular velocity of 2 rad/sec (Clockwise) while the output crank O_cC rotates with an angular velocity of 5 rad/sec (Counterclockwise) and angular acceleration of 21 rad/s^2, the angular acceleration decreases at a constant rate of 15 rad/s^3 (This example is taken from Roth and Yang, 1977 and Gupta, 1978). For convenience we choose the fixed frame $\mathfrak{F}$ with origin at O_b, and x-axis along O_bO_c and the moving frame $\mathfrak{M}$ with origin at O_c and x-axis along O_cC (see Fig. 6 of Roth and Yang, 1977). The motion of link O_cC relative to the input link O_bB may be expressed as

$$u(t) = \mathrm{Cos}(2t) = 1 - 4\,\frac{t^2}{2} + 16\,\frac{t^4}{24} + o(t^5)$$

$$v(t) = \mathrm{Sin}(2t) = 2t - 8\,\frac{t^3}{6} + o(t^5)$$

$$\varphi(t) = 7t + 21\,\frac{t^2}{2} - 15\,\frac{t^3}{6} + o(t^5)$$

From the above expressions we obtain the values of time derivatives at t = 0:

$$\begin{cases} u_1^* = 0 \\ u_2^* = -4 \\ u_3^* = 0 \\ u_4^* = 16 \end{cases} \quad \begin{cases} v_1^* = 2 \\ v_2^* = 0 \\ v_3^* = -8 \\ v_4^* = 0 \end{cases} \quad \begin{cases} \phi_1 = 7 \\ \phi_2 = 21 \\ \phi_3 = -15 \\ \phi_4 = 0 \end{cases}$$

After changing the parameter t to φ, we obtain

$$\begin{cases} u_1 = 0 \\ u_2 = -0.0816 \\ u_3 = 0.104956 \\ u_4 = -0.2325 \end{cases} \quad \begin{cases} v_1 = 0.2857 \\ v_2 = -0.12245 \\ v_3 = 0.1466 \\ v_4 = -0.3309 \end{cases}$$

Substituting the above values into Eqs. (27), (31), and (32), we have the instantaneous invariants

$$b_2 = 0.2380 \qquad a_3 = -0.4247 \qquad b_3 = -0.1324$$

$$a_4 = 0.5504 \qquad b_4 = -0.2373$$

The above results are identical to those given in Roth and Yang (1977) and Gupta (1978).

Conclusion

We have derived explicit expressions for the instantaneous invariants of rigid body motion in terms of arbitrary rather than the canonical coordinate systems. Equations (14–22) give the expressions of the instantaneous invariants for spatial and spherical motions up to the third order. Equations (27 and 31-34) give the expressions of instantaneous invariants for planar motions up to the fourth order. A few examples are used to illustrate the results. These expressions are more explicit than any existing methods in determining the instantaneous invariants and they are useful in design and analysis of mechanisms as well as in comparing rigid body motions. It is also hoped that these expressions would facilitate the use of the instantaneous invariants in more practical mechanism design and analysis problems.

Acknowledgment

The financial support of this work by US Army Research Office grant number DAAL03-90-G-0005 is gratefully acknowledged.

References

Bottema, O., 1961, "Some Remarks on Theoretical Kinematics," *Proceedings of International Conference for Teachers of Mechanisms*, The Shoe String Press Inc., pp. 157-168.

Bottema, O., and Roth, B., 1979, *Theoretical Kinematics*, North Holland.

Brand, L., 1962, *Vector and Tensor Analysis*, John Wiley, New York, NY.

Flanders, H., 1963, *Differential Forms with Applications to the Physical Sciences*, Academic Press, New York.

Gupta, K. C., 1978, "A Direct Method for the Evaluation of Instantaneous Invariants of a Given Motion," *Mechanism and Machine Theory*, Vol. 13, pp. 567-576.

Hsia, L. M., 1979, "Curvature Theory of Point Trajectories in Three-Dimensional Kinematics," Ph.D. Dissertation, University of California, Davis.

Hsia, L. M., and Yang, A. T., 1981, "On the Principle of Transference in Three-Dimensional Kinematics," ASME Journal of Mechanical Design, Vol. 103, No. 3, July, pp. 652-656.

Kirson, Y., and Yang, A. T., 1978, "Instantaneous Invariants in Three-Dimensional Kinematics," ASME *Journal of Applied Mechanics*, Vol. 45, June, pp. 409-414.

Kirson, Y., 1975, "Curvature Theory in Space Kinematics," Ph.D. Dissertation, University of California, Berkeley.

Kotelnikov, A. P., 1895, "Screw Calculus and Some Applications to Geometry and Mechanics," *Annals of the Imperial University of Kazan*.

McCarthy, J. M., and Roth, B., 1982, "Instantaneous Properties of Trajectories Generated by Planar, Spherical and Spatial Rigid Body Motions," ASME Journal of Mechanical Design, Vol. 104, January, pp. 39-51.

McCarthy, J. M., and Ravani, B., 1986, "Differential Kinematics of Spherical and Spatial Motions Using Kinematic Mapping," ASME *Journal of Applied Mechanics*, Vol. 53, March, pp. 60-64.

McCarthy, J. M., 1986, "On the Relation Between Kinematic Mappings of Planar and Spherical Displacements," ASME *Journal of Applied Mechanics*, Vol. 53, June, pp. 457-459.

McCarthy, J. M., 1986, "The Instantaneous Kinematics of Line Trajectories in Terms of a Kinematic Mapping of Spatial Rigid Motion," ASME Journal of Mechanisms, Transmissions, and Automation in Design, Vol. 109, March, pp. 95-100.

McCarthy, J. M., 1987, "The Differential Geometry of Curves in an Image Space of Spherical Kinematics," *Mechanisms and Machine Theory*, Vol. 22, No. 3, pp. 205-211.

Nayak, J. H., 1981, "Instantaneous Kinematics of Multiple-Degree-of-Freedom Motions," ASME Journal of Mechanical Design, Vol. 103, July, pp. 608-620.

Ravani, B., 1982, "Kinematic Mapping as Applied to Motion Approximation and Mechanism Synthesis," Ph.D. Dissertation, Stanford University, June.

Ravani, B., and Roth, B., 1983, "Motion Synthesis Using Kinematic Mappings," ASME Journal of Mechanisms, Transmissions, and Automation in Design, Vol. 105, No. 3, September, pp. 460-467.

Ravani, B., and Roth, B., 1984, "Mappings of Spatial Kinematics," ASME Journal of Mechanisms, Transmissions, and Automation in Design, Vol. 106, September, pp. 341-347.

Roth, B., and Yang, A. T., 1977, "Application of Instantaneous Invariants to the Analysis and Synthesis of Mechanisms," ASME *Journal of Engineering for Industry*, Series B, Vol. 99, February, pp. 97-103.

Stanišić, M. M., and Pennock, G. R., 1986, "The Canonical Inverse Velocity and Acceleration Solutions of a Planar Two-Link Open Chain," *The International Journal of Robotics Research*, Vol. 5, No. 2, Summer, pp. 82-90.

Study, E., 1903, *Die Geometrie der Dynamen*, Leipzig, 437 pp.

Veldkamp, G. R., 1967, "Canonical Systems and Instantaneous Invariants in Spatial Kinematics," *Journal of Mechanisms*, Vol. 3, pp. 329-388.

Veldkamp, G. R., 1976, "On the Use of Dual Numbers, Vectors, and Matrices in Instantaneous, Spatial Kinematics," *Mechanism and Machine Theory*, Vol. 11, pp. 141-156.

Yang, A. T., 1974, "Calculus of Screws," *Basic Question of Design Theory*, W. R. Spillers, ed., American Elsevier Publishing Co., Inc., New York, NY, pp. 265-281.

OPTIMAL SIZING OF ROBOT ACTUATORS BASED ON DYNAMIC LOAD CARRYING CAPACITY

R. Kashani
Department of Mechanical Engineering and Engineering Mechanics
Michigan Technological University
Houghton, Michigan

B. Ravani
Computer Integrated Design and Manufacturing Laboratory
Department of Mechanical Engineering
University of California–Davis
Davis, California

NOMENCLATURE

τ^k total torque at joint k (motor torque)

$\tau^k{}_{nla}$ torque at joint k due to the dynamics of the robot linkage

$\tau^k{}_l$ torque at joint k due to the dynamics of the load

$\tau^k{}_{ai}$ torque at joint k due to the dynamics of the mass of actuator i

m number of discretized points

n number of degrees of freedom of the robot manipulator

k joint number

i index on joint number

j index on the discretized point

w_i weighting factor

T_{rated} rated torque of the actuator

I_{rated} D.C. motor rated current

K constant

v volume of the actuator

τ ((mxn) x 1) total joint torque vector

τ_a ((mxn) x 1) joint torque vector due to the dynamics of mass of the actuators

τ_{nla} ((mxn) x 1) joint torque vector due to the linkage dynamics

τ_l ((mxn) x 1) joint torque vector due to the load dynamics

τ_e *extra torque*; ((mxn) x 1) vector of deviation of the portion of joint torque allocated to load τ_l, from the load carrying capacity, "load_{goal}", resolved into joint space

ABSTRACT

A procedure for sizing the joint actuators of a robot manipulator in the design stage is developed. The problem is formulated as an optimization problem where the actuators are selected to achieve a desired dynamic load carrying capacity of the manipulator over an entire discretized trajectory. Appropriate constraints on this optimization prevent the actuators from being overloaded and limits their sizes. Using worst case trajectories within the workspace, the problem is generalized to actuator selection over the entire workspace. The procedure proposed is illustrated by the choice of actuators for the Carnegie Mellon University (CMU) direct drive robot manipulator.

1 INTRODUCTION

An important step in the mechanical design of robot manipulators is sizing the joint actuators. This is usually done after the design of the kinematic configuration of the manipulator as well as structural considerations. Workspace considerations and solvability of kinematic equations are usually used to choose an appropriate kinematic configuration and structural characteristics of a manipulator are selected using strength and stiffness requirements. Once these two steps in the mechanical design of a manipulator are completed, then the joint actuators are selected.

Traditionally, robot actuator sizing have been based largely on use of simple design methods, e.g. they are sized to meet worst-case or average-case gravity and acceleration torques or loads. Such procedures are too simplistic and usually result in actuators which are far from optimal. A formalized procedure for optimal selection of robot actuators was developed by Vukobratovic, Potkonjak and Datic 1984. Their approach, however, minimized the energy which is a non-task oriented objective function. In addition, their procedure was directed toward selection of hydraulic actuators. In applications such as material handling robots are used to carry a payload along a trajectory. In such a situation, the robot actuators should be selected based on a task oriented specification such as a desired maximum payload. Thomas, Yuan-Chou and Tesar 1985 have considered such a design selection strategy. Their approach, however, sizes the actuators only locally in the neighborhood of a configuration. Ideally, the actuators should be sized such that the robot will be able to carry the maximum desired payload throughout its workspace. This is the problem considered in this technical note. First, selection of robot actuators for a given trajectory and a maximum desired payload is considered. Then, using worst case trajectories in the workspace, the method is generalized to actuator sizing for a manipulator based on a desired dynamic payload and on a global basis (i.e. over the entire workspace).

The method presented uses the Dynamic Load Carrying Capacity (DLCC) of a robot (see Wang and Ravani 1988a) to formulate an optimization problem for the selection of the joint actuators. All joint actuators are sized simultaneously and worst case trajectories introduced in Wang and Ravani 1988b are used to achieve actuator designs that can dynamically carry the desired payload over the entire workspace of the robot.

2 Dynamic Load Carrying Capacity for a Given Trajectory

The dynamic load carrying capacity of a robot manipulator depends on the end-effector trajectory (position, velocity and acceleration), and can be defined as the maximum load that the manipulator can carry in executing the trajectory, without exceeding the torque limits of the joint actuators (Wang and Ravani 1988a).

If the continuous trajectory of the end-effector is discretized into m points along the trajectory, then the total torque/force for each joint at every grid point will be obtained by linear superposition of the joint torques/forces due to the dynamics of the robot linkages (excluding the masses of the actuators), of the load, and the dynamics of the masses of the joint actuators, i.e.

$$\{\tau^k\}_j=\{\tau^k_{nla}\}_j+\{\tau^k_l\}_j+\sum_{i=k+1}^{n}\{\tau^k_{ai}\}_j \qquad (1)$$

$$j=1,2,\ldots,m.$$

The minimum value of $\{\tau^k_l\}_j$ over all joints (k), and entire discretized trajectory (j) is the dynamic load carrying capacity of the robot for that trajectory.

The concept of DLCC defined by Equation (1) will be used formulating optimal actuator sizing problem in Section 3.

3 Optimal Actuator Sizing Formulation

Since the most popular choice of the actuators for medium and small size manipulators is d.c. motors (either permanent magnet or shunt type), we limit our study to this type of actuators. The approach, however, can easily be extended to other kinds of drives.

Equation (1), gives the total joint torques which should be supplied by the joint drives. These torques are bounded by the rated torques of the motors. The torque that actuator can supply for carrying the load at each grid point,using equation (1), is

$$\{\tau^k_l\}_j=\{\tau^k\}_j-\{\tau^k_{nla}\}_j-\sum_{i=k+1}^{n}\{\tau^k_{ai}\}_j. \qquad (2)$$

To evaluate τ^k_{ai}, the mass of the actuators are needed. Archer and Blenkinsop 1986, have developed the relationship between the volume and rated torques of various electric and hydraulic actuators. The general form of this relationship is

$$\log T_{rated}=\alpha \log V+\beta,$$

where α and β are the coefficients given in Archer and Blenkinsop 1986. This last relationship will be used for calculation of τ^k_{ai}. The mass is directly related to volume with electric-motor density of 3 to 4 gr/cm^3, (and hydraulic motor density of slightly higher).

D.C. motors in control applications, e.g. robotics, are mostly current-driven (see Koren 1986). In such a case, the actuator can be idealized as a *torque source.* The output torque of the actuator,T_{output}, should be limited as

$$-T_{rated}=-K\,I_{rated}\le T_{output}\le K\,I_{rated}=T_{rated}. \qquad (3)$$

When voltage-driven d.c. motors are used for the joint actuators, the electrical dynamics of the unit should also be considered in the constraint formulation.

Equations (2) and (3) can be written in vector form over all the joints,

$$\tau_l=\tau-\tau_{nla}-\tau_a \qquad (4)$$

$$-T_{rated}\le\tau\le T_{rated}.$$

We consider the selection of joint actuators for a desired load or goal carrying capacity ("load$_{goal}$"). In order to formulate the objective function, we define the *extra torque* vector, τ_e,. This torque vector is defined as the deviation of the portion of joint torque vector, which is allocated to load (τ_l), from the vector of load carrying capacity, "load$_{goal}$", resolved into joint space, (τ_{goal}), i.e.

$\tau_e = \tau_l - \tau_{goal} = (\tau - \tau_{nla} - \tau_a) - \tau_{goal}$
where τ_{goal} is a vector of (m x n) x 1, resulting from resolving the desired load to be carried , ("load$_{goal}$") into joint space, over all the discretized points along the trajectory.

The optimization problem is then defined as the minimization of the *extra torque* vector, τ_e, i.e.

$$\min\ (\tau_e) = \min\ (\tau_l - \tau_{goal}) = \min\ \{(\tau - \tau_{nla} - \tau_a) - \tau_{goal}\} \quad (5)$$

constrained by

$$-T_{rated} \leq \tau \leq T_{rated} . \quad (6)$$

The constraint set (6) indicates that the absolute value of each joint torque should be smaller than or equal to the rated torque of the actuator at that joint. The design parameters are τ and T_{rated}.

The vector objective function (5), is converted to a scalar objective function as shown in Equation (7), using sum of L_2 norms of weighted *extra torque* vector, τ_e, over the entire discretized trajectory, namely

$$\min\ (\sum_{j=1}^{m} \left(\sum_{i=1}^{n} (w_i [\tau_e (i)]^2) \right)^{1/2}) =$$

$$\min\ (\sum_{j=1}^{m} \left(\sum_{i=1}^{n} (w_i [(\tau^i - \tau^{ij}_{nla} - \tau^{ij}_a) - \tau^{ij}_{goal}]^2) \right)^{1/2}) \quad (7)$$

Now in order to solve the optimization problem defined by the objective function (7) and the set of inequality constraints (6), we use exterior inequality quadratic penalty function. A computer program based on Quasi-Newton method is developed to solve the resulting unconstrained problem.

3.1 Trajectory Selection

The optimal actuator sizing methodology developed in this work, depends on the end-effector trajectory. In sizing the actuators, it is desired to select smallest actuators that would allow the robot to carry maximum payload in a desired cycle time throughout its entire workspace. This means that the actuators can be sized based on the worst-case trajectories within the workspace. For each worst-case trajectory, a corresponding set of actuators will be sized. Amongst these sets of actuators, the one which provides the maximum dynamic load carrying capacity[7], will be selected as the optimal set of joint actuators. The worst-case trajectories defined by Wang and Ravani [7], is used in this work.

4 Numerical Example

An example of optimal selection of robot actuators, based on maximizing the dynamic load carrying capacity for worst-case trajectory, is presented for a robot manipulator with the structure of the Carnegie Mellon University (CMU) direct drive arm Asada and Kanade 1983, as shown in Fig. 1.

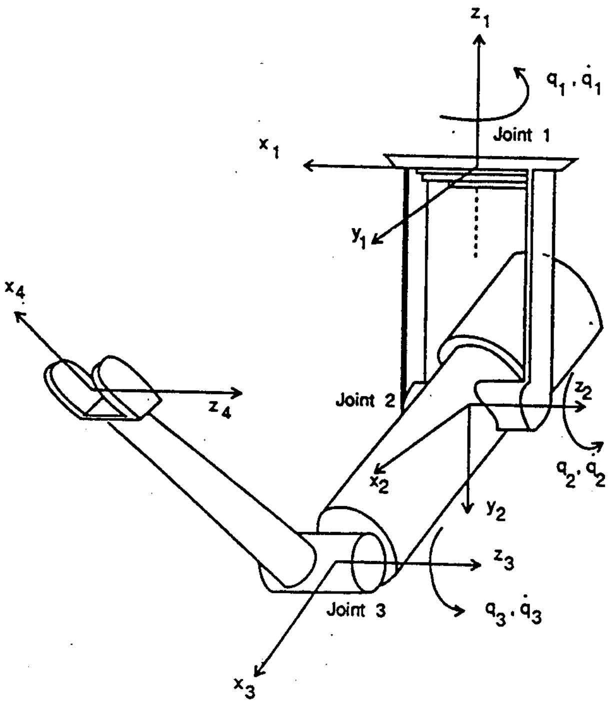


Fig. 1. A three joint direct drive robot (Reprint from [7]).

The initial and final joint configurations of the robot for this example are:

$[0°, 0°, 0°]^T$ and $[0°, 90°, 90°]^T$,

A cubic polynomial position trajectory is used between these two configurations. The two end-points correspond to minimum vertical and maximum horizontal positions of the end-effector. The maximum horizontal position also happens to be the terminal point (see Wang and Ravani 1988b). This trajectory satisfies the requirements stated in Wang and Ravani 1988b for the worst-case trajectory. The total desired cycle time for this motion is 1.5 second. This time period is based on average speed of 50 in/sec, recommended for industrial robots Asfahl 1985. The payload ("load$_{goal}$") is taken to be 14 kg.

In order to guarantee the existence of a minima, rated torque of the actuator of the last joint has been selected apriori to be 40 Nm.

The trajectory is discretized into 10 equally spaced points. Joint torques due to link dynamics, τ_{nla}, and due to carrying the desired load, τ_{goal}, are evaluated and shown in Fig.'s 2 and 3.

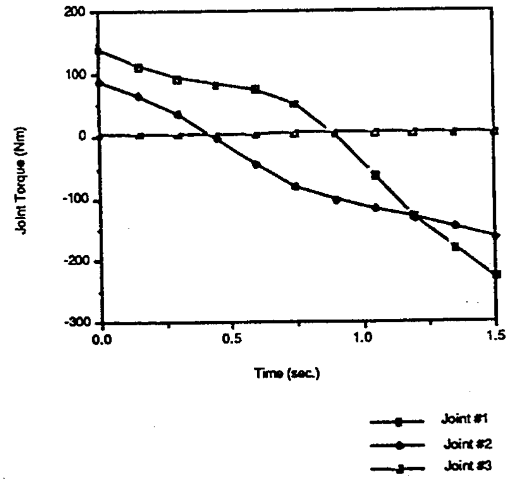


Fig. 2. Joint torques due to the robot's linkage dynamics.

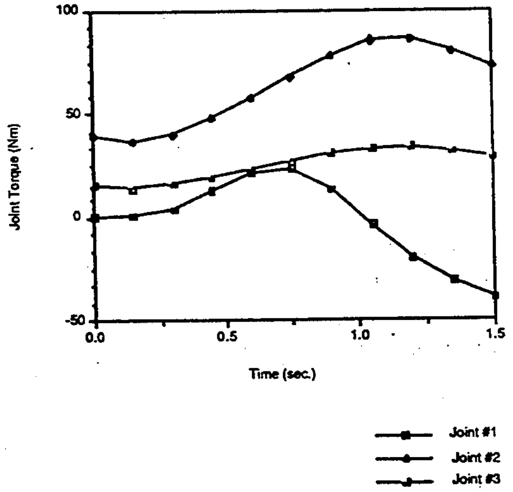


Fig. 3. Joint torques due to dynamics of carrying the desired load.

The rated torque of the actuators of joints 1 and 2, optimized over the entire trajectory, is found to be 140 and 80 Nm. Moreover optimal rated torques of the first and the second joint actuators at each discrete point along the trajectory are calculated and shown in Fig. 4.

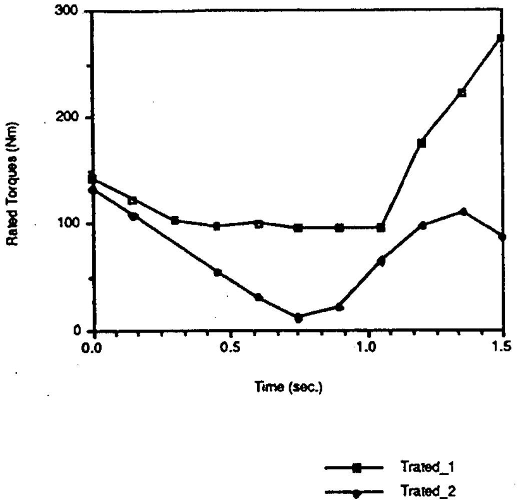


Figure 4. Optimal actuators' rated torques at discrete points along the trajectory.

Comparison of Fig. 4 with the result of optimization over the entire trajectory (actuator rated torques of 140 and 80 Nm), indicates that this actuators would be overloaded at some points along the trajectory. This is acceptable, considering the fact that actuators, especially d.c. motors, are made to take transient overloading.

If the transient overloading of the actuators is not desirable, one could select the actuators with maximum T_{rated}, using Fig. 4. This would result in actuator rated torques of 270 and 130 Nm and keeps the rated torque of the selected actuators always higher than the maximum value of total joint torques.

The original actuators of joints 1 and 2 of CMU arm have the rated torques of 204 and 136 Nm.

5 CONCLUSIONS

A method is presented for sizing of robot joint actuators based on meeting a functional requirement of a desired dynamic payload over the entire robot workspace. The problem is formulated as an optimization problem minimizing the sum of L_2 norms of weighted *extra torque* vector over the entire discretized worst case trajectories. The constraints are based on prevention of joint actuators from excessive overloading.

Actuators are selected for worst case trajectories within the workspace resulting in manipulator designs that can meet a desired specification of dynamic payload. The algorithm presented provides the basis for rational selection of joint actuators in mechanical design of robot manipulators.

6 ACKNOWLEDGEMENT

The financial support of this work by US Army Research Office grant number DAAL03-90-G-0005 is gratefully acknowledged.

7 REFERENCES

1. Archer, J. R., and Blenkinsop, P. T., 1986, "Actuation for industrial Robots", *Proc. Instn. Mech. Engrs.* vol. 200 No. B2.

2. Asada, H., Kanade, T., 1983, "Control of Direct-Drive Arm", *ASME Transaction, J. Dyn. Sys. Meas. Contr.* vol. 105, Sept, p. 136-142.

3. Asfahl, C. Ray, 1985, Robots and Manufacturing Automation, p. 170 , Wiley.

4. Gill, P. E., Murray, W. and Wright, M., H., 1981, "Practical Optimization",*Academic Press.*

5. Koren, Y., 1986, "Control of machine tools and robots" *Appl Mech Rev* Vol. 39, No. 9, Sept.

6. Wang, L. T., and Ravani, B., 1988a,"Dynamic Load Carrying Capacity of Mechanical Manipulators; Part I: Problem Formulation", *ASME Transaction, J. Dyn. Sys. Meas. Contr.* Vol. 110, pp. 46-52.

7. Wang, L. T., and Ravani, B. 1988b,"Dynamic Load Carrying Capacity of Mechanical Manipulators; Part II: Numerical Solution", *ASME Transaction, J. Dyn. Sys. Meas. Contr.* 110, pp. 53-61.

DE-Vol. 26

CAMS, GEARS, ROBOT AND MECHANISM DESIGN

presented at

THE 1990 ASME DESIGN TECHNICAL CONFERENCES —
21st BIENNIAL MECHANISMS CONFERENCE
CHICAGO, ILLINOIS
SEPTEMBER 16–19, 1990

sponsored by

THE MECHANISMS COMMITTEE OF THE
DESIGN ENGINEERING DIVISION, ASME

edited by

A. PISANO
UNIVERSITY OF CALIFORNIA–BERKELEY

M. McCARTHY
UNIVERSITY OF CALIFORNIA–IRVINE

S. DERBY
RENSSELAER POLYTECHNIC INSTITUTE

THE AMERICAN SOCIETY OF MECHANICAL ENGINEERS
345 East 47th Street □ United Engineering Center □ New York, N.Y. 10017

M. C. Lin

B. Ravani

S. A. Velinsky

Department of Mechanical and
Aeronautical Engineering,
University of California-Davis,
Davis, CA 95616

Kinematics of the Ball Screw Mechanism

This paper studies the kinematics of the Ball Screw Mechanism (BSM) with the aim of developing a foundation for understanding the motion of the balls and their contact patterns with the contacting elements. It is shown that there is always slip between the balls and the nut or screw, and therefore, the no-slip condition assumed in the BSM literature is not attainable. The effect of contact deformation on the motion of the balls is also studied and is used to develop the pattern of the constant sliding lines of contact between the ball and the screw or the nut. The results have applications in efficiency analysis, design, wear evaluation and finite element modeling of the BSM.

1 Introduction

The reciprocating ball screw mechanism is a force and motion transfer device belonging to the family of power transmission screws (Fig. 1). Two of the most important features of the mechanism are its positional accuracy and load carrying capacity making it suitable as the drive mechanism for robot manipulators or the feed-drive mechanism of machine tools. The utilization of bearing balls in the mechanism replaces the sliding friction of the conventional power screw with the rolling friction of the balls. This results in minimal friction during force and motion transmission and eliminates slipstick with minimal wear.

This paper provides a theoretical study of the kinematics of the ball screw mechanism. It derives relationships describing the motion of the ball and shows that slipping takes place between the ball and the nut (or the screw) at all times. This means that the no-slip condition assumed in the literature (Levit, 1963; Drozdov, 1984) is unattainable. The proper slip conditions are derived and the ball motion is studied.

In addition, the effect of elastic deformation at contact areas between the ball and the nut (or the screw) on the kinematics of the mechanism is analyzed. The analysis is used to determine the pattern of constant sliding lines of the ball in the contact area. The contact line pattern is useful in wear and finite element analyses of the ball screw mechanism. The work presented provides a theoretical framework for understanding the motion of the ball in the BSM and sets the basis for efficiency and design analysis of the mechanism. The application of the results of this paper to efficiency and friction analysis of the BSM is given in a companion paper (Lin, Velinsky, and Ravani, 1994).

In all the analyses presented right-hand screw threads and a single nut are assumed.

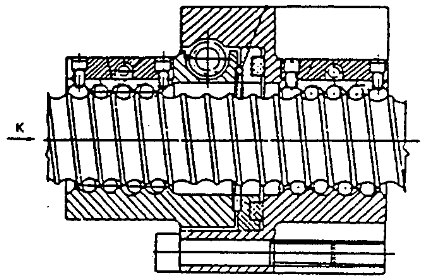


Fig. 1 A ball screw and nut mechanism

2 Motion of the Ball

In this section, we study the motion of the ball by affixing a Frenet coordinate system to the path of the center of the ball. This will enable us to study the kinematics of the ball motion and derive the slip conditions.

In a BSM, the center of the ball moves along the helical groove of the screw. We introduce three sets of coordinate systems. The first (world) coordinate system, $ox'y'z'$, is fixed with its z' axis coincident with the axis of the screw. The second (rotating) coordinate system, $oxyz$, also has its z axis coincident with the screw axis (Fig. 2) but it rotates with the screw. The third coordinate system is the Frenet frame moving with the center of the ball along the trajectory of the ball center. This trajectory, with respect to the frame $oxyz$, is a circular helical line along a circular cylindrical surface with a mean radius r_m (Fig. 2). The coordinate transformation between the first two coordinate systems can be written as

$$\mathbf{X}'=\mathbf{T}_1\mathbf{X} \tag{1}$$

where $\mathbf{X}' = [i'\ j'\ k']^T$, $\mathbf{X} = [i\ j\ k]^T$,

$$\mathbf{T}_1 \equiv \begin{bmatrix} C_\Omega & -S_\Omega & 0 \\ S_\Omega & C_\Omega & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Contributed by the Mechanisms Committee for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received March 1991; revised Feb. 1994. Associate Technical Editor: G. L. Kinzel.

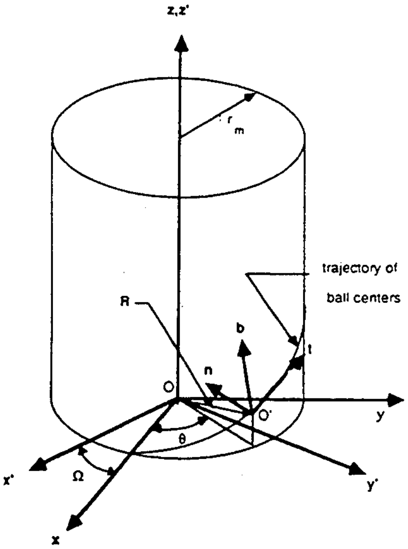


Fig. 2 The position of the ball center, O′, in Cartesian coordinates and Frenet coordinates

and $C_\Omega \equiv \cos(\Omega)$, $S_\Omega \equiv \sin(\Omega)$, and Ω denotes the angular displacement between the two coordinate systems.

Assume that a ball has moved through an angle $\theta\mathbf{k}$ along the helical groove of a screw with lead L. The position vector of the ball center, $\mathbf{R}(\theta)$, can be expressed as

$$ {}^{r}\mathbf{R}(\theta)=\mathbf{R}^T\mathbf{X} \tag{2}$$

where $\mathbf{R}^T = [r_mC_\theta\ r_mS_\theta\ r_m\theta t_\alpha]$, $C_\theta = \cos(\theta)$, $S_\theta = \sin(\theta)$, and the helix angle, α, is defined as $t_\alpha = \tan(\alpha) = L/2\pi r_m$. The superscripts "$r$" and "$W$" are used here to distinguish a vector with respect to the rotational and the world coordinate systems, respectively.

By substituting Eq. (1) into Eq. (2), the position of the ball center can be expressed with respect to the world coordinate system as

$$ {}^{W}\mathbf{R}=\mathbf{R}^T\mathbf{T}_1^{-1}\mathbf{X}'. \tag{3}$$

By definition (Kreyszig, 1983), the triad of unit vectors describing the Frenet Coordinate system of the ball center with respect to the rotating Cartesian system, *oxyz*, can be expressed as follows (Fig. 2):

(*a*) Unit tangent vector

$$\mathbf{t}=[-kdS_\theta\quad kdC_\theta\quad \tau d]\mathbf{X} \tag{4}$$

(*b*) Unit normal vector

$$\mathbf{n}=[-C_\theta\quad -S_\theta\quad 0]\mathbf{X} \tag{5}$$

(*c*) Unit binormal vector

$$\mathbf{b}=[\tau dS_\theta\quad -\tau dC_\theta\quad kd]\mathbf{X} \tag{6}$$

where

$$d=\frac{r_m}{C_\alpha},$$

curvature

$$k=\frac{C_\alpha^2}{r_m},$$

and torsion

$$\tau=\frac{S_\alpha\ \ C_\alpha}{r_m} \tag{7}$$

The coordinate transformation between the Frenet frame of the ball center and the rotational Coordinate system *oxyz* can be expressed as

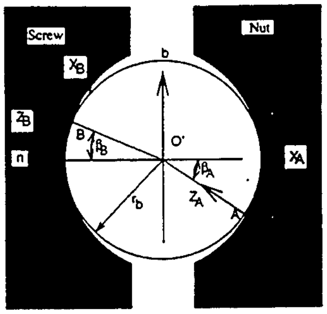


Fig. 3 Location of contact points on the normal plane

$$\mathbf{X}=\mathbf{T}_2\mathbf{Y} \tag{8}$$

where

$$\mathbf{Y}\equiv[\mathbf{t}\ \mathbf{n}\ \mathbf{b}]^T,\quad \text{and}\quad \mathbf{T}_2\equiv\begin{bmatrix} -C_\alpha S_\theta & -C_\theta & S_\alpha S_\theta \\ C_\alpha C_\theta & -S_\theta & -S_\alpha C_\theta \\ S_\alpha & 0 & C_\alpha \end{bmatrix}.$$

The terms S_α and C_α denote $\sin(\alpha)$ and $\cos(\alpha)$, respectively. Now, Eq. (2) can be rewritten in terms of the Frenet frame of the ball center as

$$ {}^{r}\mathbf{R}=\mathbf{R}^T\mathbf{T}_2\mathbf{Y}=r_m[S_\alpha t_\alpha\theta\quad -1\quad C_\alpha t_\alpha\theta]\mathbf{Y}. \tag{9}$$

It will be shown in the next section, that the ball can only move relative to the screw in the tangential direction of the Frenet frame of the ball center trajectory. This is because the ball is confined along the helical groove in directions parallel to the normal plane of the trajectory of the ball center. Physically, this means that the contact points between the ball and the screw, as well as between the ball and the nut, must be located on this normal plane.

In order to locate these contact points, the contact angle, β, is defined as the angle between the unit normal vector and the contact vector. The contact vector is oriented from the ball center toward the contact point, as shown in Fig. 3. Points A and B represent the instantaneous contact points between the ball and the nut and between the ball and the screw, respectively. Note that β_A (Fig. 3) is considered positive when measured clockwise from the negative side of the normal axis; whereas β_B (Fig. 3) is considered positive when measured clockwise from the positive side of the normal axis. The angle β_A and β_B are always positive for a counterclockwise rotation of the screw (viewed from the end) and negative for a clockwise screw rotation.

We now introduce a pair of new coordinate systems $iX_iY_iZ_i$, with $i = A, B$, between the ball and the raceway such that the X_iY_i plane lies on the plane of contact and the Z_i-axis lies along the common normal of the two contacting bodies (Fig. 3). These coordinate systems are used to describe the position of the contact point between the two contacting bodies. We also assume point contacts along a diagonal line between the ball and the screw and nut. The coordinate transformation between the Frenet frame of the ball center trajectory and the $iX_iY_iZ_i$ coordinate system is

$$\mathbf{X}_i=\mathbf{T}_3\mathbf{Y} \tag{10}$$

where

$$\mathbf{X}_i=[\mathbf{i}_i\quad \mathbf{j}_i\quad \mathbf{k}_i]^T\quad \text{and}\quad \mathbf{T}_3=\begin{bmatrix} 0 & -S_{\beta i} & C_{\beta i} \\ 1 & 0 & 0 \\ 0 & C_{\beta i} & S_{\beta i} \end{bmatrix}.$$

The position vector of the contact point B with respect to the ball center can thus be expressed as

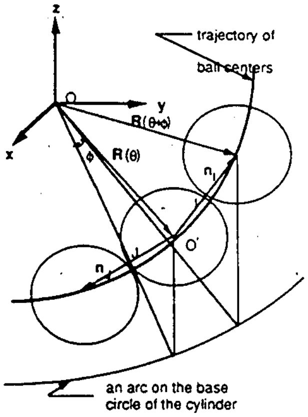
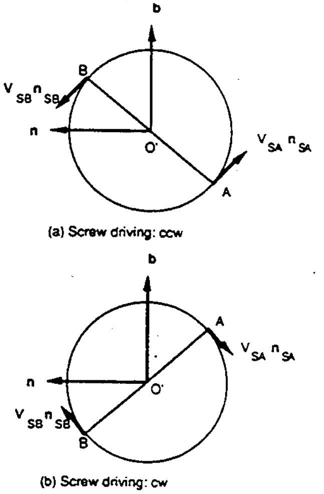

z
trajectory of ball centers
O
y
B(θ+φ)
x
φ
R(θ)
n_I
O'
n
an arc on the base circle of the cylinder

Fig. 4 Phase angle between two consecutive balls



(a) Screw driving: ccw

(b) Screw driving: cw

Fig. 5 Slip velocities at steady state with no-slip along the tangential direction, for the conversion of rotary into linear motion

$$\mathbf{R}_{BO}' = [0 \quad 0 \quad r_b]\mathbf{X}_B \tag{11}$$

where r_b denote the ball radius. The position vectors of contact points A and B with respect to the rotational coordinate system can be expressed as

$$^r\mathbf{R}_A = {}^r\mathbf{R} + \mathbf{R}_{AO}' \tag{12}$$

and

$$^r\mathbf{R}_B = {}^r\mathbf{R} + \mathbf{R}_{BO}'. \tag{13}$$

There is only one independent variable, θ, which describes the relative position of the center of the ball with respect to the screw in the equation of the circular helical line [Eq. (2)]. The relative position of the two consecutive ball centers can therefore be expressed as a function of a phase angle, ϕ, shown in Fig. 4, as

$$\begin{aligned}\delta\mathbf{R}_I &= {}^r\mathbf{R}(\theta+\phi) - {}^r\mathbf{R}(\theta)\\ &= r_m[C_\alpha S_\phi + S_\alpha t_\alpha \phi \quad 1 - C_\phi \quad S_\alpha(\phi - S_\phi)]\mathbf{Y}.\end{aligned} \tag{14}$$

Furthermore, the minimal central distance between two consecutive ball centers must be equal to the diameter, $2r_b$, of the balls; that is

$$\sqrt{\delta\mathbf{R}_I^T \cdot \delta\mathbf{R}_I} = 2r_b$$

or

$$(\phi t_\alpha)^2 + 2(1 - C_\phi) = 4a^2 \tag{15}$$

where

$$a \equiv \frac{r_b}{r_m}.$$

By solving the above equation for the phase angle, ϕ, the maximum number of balls per revolution, N, can be obtained:

$$N = \frac{2\pi}{\phi}. \tag{16}$$

Furthermore, Eq. (14) can be expressed as

$$\delta\mathbf{R}_I = 2r_b\mathbf{n}_I. \tag{17}$$

Similarly,

$$\delta\mathbf{R}_J = 2r_b\mathbf{n}_J. \tag{18}$$

In these equations, $\mathbf{n}_I$ and $\mathbf{n}_J$ are unit vectors between two consecutive ball centers. Using the Frenet frame, the relative motion between the centers of the balls in contact with each other can be written as:

$$\mathbf{Y}(\theta+\phi) = \begin{bmatrix} C_\alpha^2 C_\phi + S_\alpha^2 & -C_\alpha S_\phi & C_\alpha S_\alpha(1 - C_\phi) \\ C_\alpha S_\phi & C_\phi & -S_\alpha S_\phi \\ C_\alpha S_\alpha(1 - C_\alpha) & S_\alpha S_\phi & S_\alpha^2 C_\phi + C_\alpha^2 \end{bmatrix} \mathbf{Y}(\theta). \tag{19}$$

The above equation is valid for balls which are both ahead (positive ϕ values) and behind (negative ϕ values).

3 Slip Analysis

Determination of the slip conditions between the reciprocating balls and the nut or the screw is important in understanding the motion of the ball in the ball screw mechanism. These conditions are also necessary for efficiency analysis and the design considerations (see Lin, Velinsky, and Ravani, 1994). A complete velocity analysis is necessary for determining slip directions and velocities. This section provides such an analysis and applies the results to a characterization of the slip conditions for the BSM. Previous works (see Levit, 1963; and Drozdov, 1984) have treated the kinematics of the BSM using the previous results from ball bearings (see, e.g., Harris, 1971; and Jones, 1959). This has resulted in incorrect results for the BSM since the angular velocities of different elements, namely the ball, the screw, and the nut are not additive as used in ball bearing analysis.

The velocity of the ball center with respect to the rotational Cartesian Coordinates, *oxyz*, can be obtained by differentiating Eq. (9) with respect to time; i.e.,

$$^r\dot{\mathbf{R}} = [d\dot{\theta} \quad 0 \quad 0]\mathbf{Y}. \tag{20}$$

Note that in the above equation, the velocity of the ball center relative to the rotating coordinate system has only a tangential component. Physically, the ball cannot move in the normal plane since contacting surfaces would have to separate or crush together for motion in the normal plane to exist. Furthermore, the radius of motion of the ball center is d, which includes both the curvature and the torsion of the helix.

The velocity of the ball center with respect to the world coordinate system, *ox' y' z'*, can be obtained by differentiating Eq. (3) with respect to time; i.e.,

$$^W\dot{\mathbf{R}} = [d\dot{\theta} + r_m C_\alpha \dot{\Omega} \quad 0 \quad -r_m S_\alpha \dot{\Omega}]\mathbf{Y}. \tag{21}$$

The above equation can also be obtained from $^W\dot{\mathbf{R}} = {}^r\dot{\mathbf{R}} + \dot{\Omega} \times {}^r\mathbf{R}$.

If we let $\omega = [\omega_t \quad \omega_n \quad \omega_b]$ $\mathbf{Y}$ be the angular velocity of the ball, then the instantaneous velocity of the two points A and B (namely V_{Ab} and V_{Bb}) on the ball (which are coincident with the two contact points between the nut and the screw, respectively) can be expressed as

$$\mathbf{V}_{Ab} = {}^{W}\dot{\mathbf{R}} + \omega \times \mathbf{R}_{AO}'$$

$$= \begin{bmatrix} d\dot{\theta} + r_m C_\alpha \dot{\Omega} + r_b(\omega_b C_{\beta A} - \omega_n S_{\beta A}) \\ r_b \omega_t S_{\beta A} \\ -r_m S_\alpha \dot{\Omega} - r_b \omega_t C_{\beta A} \end{bmatrix}^T \mathbf{Y} \tag{22}$$

and

$$\mathbf{V}_{Bb} = {}^{W}\dot{\mathbf{R}} + \omega \times \mathbf{R}_{BO}$$

$$= \begin{bmatrix} d\dot{\theta} + r_m C_\alpha \dot{\Omega} + r_b(\omega_b C_{\beta B} - \omega_n S_{\beta B}) \\ -r_b \omega_t S_{\beta B} \\ -r_m S_\alpha \dot{\Omega} + r_b \omega_t C_{\beta B} \end{bmatrix}^T \mathbf{Y} \tag{23}$$

We will now determine the velocity of the point on the nut which is coincident with the contact point for the case when the screw is driving. If the nut is driving, similar results can easily be obtained. We will only consider the conversion of rotary into linear motion. The cases involving the conversion of linear into rotary motion are merely the kinematic inversions of the cases presented here.

As in the conventional power screw unit, if the nut is restrained from rotating, it will move axially a distance $\Omega L/2\pi$ along the screw for a screw rotation of angle Ω. Hence, if the angular velocity of the screw is $\dot{\Omega}\mathbf{k}$, then the velocity of any point on the nut with respect to the world coordinate system, $ox'y'z'$ can be represented as

$$\mathbf{V}_{An} = \begin{bmatrix} 0 & 0 & \dfrac{-\dot{\Omega}L}{2\pi} \end{bmatrix} \mathbf{X}'$$

$$= -r_m S_\alpha \dot{\Omega}[t_\alpha \quad 0 \quad 1]\,\mathbf{Y}. \tag{24}$$

The velocity of the contact point on the screw can similarly be obtained as follows.

If the screw rotates with an angular velocity $\dot{\Omega}\mathbf{k}$, the velocity of point B on the screw (namely V_{BS}) coincident with the contact point will be

$$\mathbf{V}_{BS} = \dot{\Omega}\mathbf{k} \times {}^{r}\mathbf{R}_B$$

$$= \dot{\Omega} \begin{bmatrix} (r_m - r_b C_{\beta B}) C_\alpha \\ -r_b S_{\beta B} S_\alpha \\ -(r_m - r_b C_{\beta B}) S_\alpha \end{bmatrix}^T \mathbf{Y}. \tag{25}$$

This velocity is with respect to the world coordinate system, $ox'y'z'$.

The slip velocities at contact points A and B (V_{SA} and V_{SB}, respectively) can now be determined as follows.

From Eqs. (22), (23), (24), and (25), the slip velocities at points A and B are, respectively:

$$\mathbf{V}_{SA} = \mathbf{V}_{Ab} - \mathbf{V}_{An}$$

$$= \begin{bmatrix} d(\dot{\theta} + \dot{\Omega}) + r_b(\omega_b C_{\beta A} - \omega_n S_{\beta A}) \\ r_b \omega_t S_{\beta A} \\ -r_b \omega_t C_{\beta A} \end{bmatrix}^T \mathbf{Y}$$

$$= \begin{bmatrix} -r_b \omega_t \\ d(\dot{\theta} + \dot{\Omega}) + r_b(\omega_b C_{\beta A} - \omega_n S_{\beta A}) \\ 0 \end{bmatrix}^T \mathbf{X}_A$$

$$= V_{SA}\mathbf{n}_{SA} \tag{26}$$

and

$$\mathbf{V}_{SB} = \mathbf{V}_{Bb} - \mathbf{V}_{Bn}$$

$$= \begin{bmatrix} d\dot{\theta} - r_b[(\omega_b - \dot{\Omega}C_\alpha) C_{\beta B} - \omega_n S_{\beta B}] \\ -r_b(\omega_t - \dot{\Omega}S_\alpha) S_{\beta B} \\ r_b(\omega_t - \dot{\Omega}S_\alpha) C_{\beta B} \end{bmatrix}^T \mathbf{Y}$$

$$= \begin{bmatrix} r_b(\omega_t - \dot{\Omega}S_\alpha) \\ d\dot{\theta} - r_b[(\omega_b - \dot{\Omega}C_\alpha) C_{\beta B} - \omega_n S_{\beta B}] \\ 0 \end{bmatrix}^T \mathbf{X}_B$$

$$= V_{SB}\mathbf{n}_{SB} \tag{27}$$

where $\mathbf{n}_{SA}$ and $\mathbf{n}_{SB}$ denote the unit vectors in the direction of the slip velocities at the instantaneous contact points A and B, respectively, and $\mathbf{V}_{SA}$ and $\mathbf{V}_{SB}$ are the magnitudes of these velocities (Fig. 5).

Physically, contacting surfaces should have common normals at their instantaneous points of contact. This means that the slip velocities should be perpendicular to the common normal to the two surfaces, namely $\mathbf{R}_{AO'} \cdot \mathbf{V}_{SA} = 0$ and $\mathbf{R}_{BO'} \cdot \mathbf{V}_{SB} = 0$. These conditions on slip velocities represent the physical contact requirement for the contacting surfaces not to separate or crush together. We should note that the unit vectors $\mathbf{n}_{SA}$ and $\mathbf{n}_{SB}$ are opposite in direction to the frictional forces on the ball at the contact points.

The location of all contact points can be determined using these physical contact conditions. Suppose that the contact points are not located in the central normal plane. Since the balls are spherical, only two angles are necessary to define the location of a contact point on a ball. Suppose these two angles are β_B and an angle Ψ off the central normal plane. Consider point B as an example to simplify the analysis. The position vector $\mathbf{R}_{BO}'$ can be determined from geometry as

$$\mathbf{R}_{BO}' = [-r_b S_\Psi \quad r_b C_\Psi C_{\beta B} \quad r_b C_\Psi S_{\beta B}]\mathbf{Y}.$$

Substituting this last equation into Eqs. (11), (13), (23), and (25), we can obtain $\mathbf{V}_{SB}$, which when substituted into the contact conditions result in:

$$\mathbf{R}_{BO}' \cdot \mathbf{V}_{SB} = -r_b d\dot{\theta} S_\Psi = 0.$$

Since $r_b d$ cannot be zero and $\dot{\theta}$ is not a geometric parameter, the offset angle, Ψ, has to be zero to satisfy the contact condition. In other words, all the contact points must lie on the normal plane.

Let us decompose the slip-velocity into two parts: (*1*) on the central normal plane, and (*2*) along the tangential direction of the Frenet frame of the ball center trajectory. Then, as we can see from the Eqs. (26) and (27), the magnitudes of the resultant slip-velocities on the central normal plane at the ball/nut and the ball/screw contact points are $r_b(\omega_t - S_\alpha\Omega)$ and $r_b\omega_t$, respectively. Accordingly, no ω_t value exists which causes the slip-velocities on the central normal plane at both contact points to vanish simultaneously. In other words, friction can never vanish at both contact points on the central normal plane unless the helix angle, α, equals zero. A method for calculating the frictional losses, which has been used as the basis for many subsequent studies, was given by Levit (1963) who defined the BSM friction angle to lie along the tangential direction of the Frenet frame of the ball center trajectory. This produces an inaccuracy due to the fact that it does not account for the effects of the torsion of the helix.

4 Acceleration Analysis

The acceleration of the ball center with respect to the rotational Cartesian coordinate system, $oxyz$, can be expressed as

$$^{r}\ddot{\mathbf{R}}=[d\ddot{\theta} \quad r_m\dot{\theta}^2 \quad 0]\mathbf{Y} \tag{28}$$

where $d\ddot{\theta}$ and $r_m\dot{\theta}^2$ represent components of tangential and centripetal acceleration of the ball center with respect to the rotational Coordinate System *oxyz*, respectively. The acceleration of the ball center in the world coordinate system is expressed as

$$^{W}\ddot{\mathbf{R}}={}^{r}\ddot{\mathbf{R}}+\ddot{\Omega}\times{}^{r}\mathbf{R}+\dot{\Omega}\times\dot{\Omega}\times{}^{r}\mathbf{R}+2\dot{\Omega}\times{}^{r}\dot{\mathbf{R}}$$

$$=[d(\ddot{\theta}+C_\alpha^2\ddot{\Omega}) \quad r_m(\dot{\theta}+\dot{\Omega})^2 \quad -r_mS_\alpha\ddot{\Omega}]\mathbf{Y} \tag{29}$$

where the component along the normal direction, $r_m(\dot{\theta}+\dot{\Omega})^2$ includes both the centripetal and Coriolis effects. The angular acceleration of the ball is given by the symbol $\mathbf{a}$:

$$\mathbf{a}=\begin{bmatrix} \dot{\omega}_t-\omega_nC_\alpha(\dot{\theta}+\dot{\Omega})+S_\alpha(\ddot{\theta}+\ddot{\Omega}) \\ \dot{\omega}_n+(\omega_tC_\alpha-\omega_bS_\alpha)(\dot{\theta}+\dot{\Omega}) \\ \dot{\omega}_b+\omega_n+S_\alpha(\dot{\theta}+\dot{\Omega})+C_\alpha(\ddot{\theta}+\ddot{\Omega}) \end{bmatrix}^T\cdot\mathbf{Y} \tag{30}$$

where the first term in each component represents the change in magnitude and the second term represents the gyroscopic motion.

5 The Kinematics of the BSM with Elastic Deformation

Loads acting between the balls and raceways in the BSM develop only small areas of contact between the mating members. Consequently, although the elemental loading may only be moderate, stresses induced on the surfaces of the balls and raceways are usually large. Contact deformations are caused by contact stresses. Because of the rigid nature of the balls, these deformations are generally of a low order of magnitude. The classical solution for the local stress and deformation of two elastic bodies contacting at a single point is that of Hertz (1881) which has been applied to ball-bearing problems. Levit (1963) introduced the theory to the BSM. However, the kinematics of the BSM regarding elastic deformations has never been solved completely. It is the purpose of this section to develop the internal motions of the BSM, the relative slip-velocity between the ball and the nut/screw and the pattern of sliding lines of contact and thus set the foundation for further investigations on friction, wear and finite element analyses.

5.1 Position of the Contact Point. To specify the position of a contact point between two deformed bodies, a coordinate system, $iX_iY_iZ_i, i=A, B$, is introduced between the ball and the raceway (at each contact point as before). In this coordinate system, the X_iY_i plane lies on the plane of contact and the Z_i-axis is the common normal of the two contacting bodies. Figure 6 shows a ball contacting the screw raceway such that the normal force between the ball and the screw is distributed over an elliptical surface defined by the projected major and minor semi-axes, a_B and b_B, respectively. The coordinate transformation between the Frenet frame and the $iX_iY_iZ_i$ system is

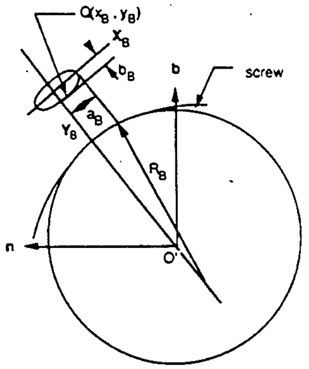


Fig. 6 Ball-screw contact

$$\mathbf{X}_i=\mathbf{T}_3\mathbf{Y} \tag{31}$$

where

$$\mathbf{X}_i=[i_i \quad j_i \quad k_i]^T \quad \text{and} \quad \mathbf{T}_3=\begin{bmatrix} 0 & -S_{\beta i} & C_{\beta i} \\ 1 & 0 & 0 \\ 0 & C_{\beta i} & S_{\beta i} \end{bmatrix}.$$

The position vector of an arbitrary point Q, on the X_BY_B plane with respect to the origin B, can be expressed as $\mathbf{R}_{QB}=[x_B \quad y_B \quad 0]\,\mathbf{X}_B$ and the position vector of point B with respect to the ball center is $\mathbf{R}_{BO'}=[0 \quad 0 \quad r_b]\,\mathbf{X}_B$. The radius of curvature of the screw raceway groove is

$$r_B=\sqrt{R_B^2-x_B^2}-\sqrt{R_B^2-a_B^2}+\sqrt{r_b^2-a_B^2}, \tag{32}$$

and the radius of curvature of the deformed surface as defined by Hertz is

$$R_i=\frac{2f_ir_b}{f_i+1} \tag{33}$$

where

$$f_i=\frac{r_i}{r_b},\ i=A,\ B.$$

Therefore, the position vector of point Q with respect to the world coordinates is

$$\mathbf{R}_{QO'}=\mathbf{R}_{QB}+\mathbf{R}_{BO'}+{}^{W}\mathbf{R}. \tag{34}$$

5.2 Velocity of BSM with Deformation. The derivation of the velocity of the BSM with deformation is similar to the procedure in Section 3, from which, the following equations can be determined:

5.2.1 Velocity of Any Contact Point P/Q on the Ball

$$\mathbf{V}_{Pb}={}^{W}\dot{\mathbf{R}}+\omega\times\mathbf{R}_{PO'}$$

$$=\begin{bmatrix} d(\dot{\theta}+C_\alpha^2\dot{\Omega})+(x_AS_{\beta A}+r_AC_{\beta A})\omega_b+(x_AC_{\beta A}-r_AS_{\beta A})\omega_n \\ -(x_AC_{\beta A}-r_AS_{\beta A})\omega_t+y_A\omega_b \\ -r_mS_\alpha\dot{\Omega}-(x_AS_{\beta A}+r_AC_{\beta A})\omega_t-y_A\omega_n \end{bmatrix}^T\mathbf{Y} \tag{35}$$

$$\mathbf{V}_{Qb}={}^{W}\dot{\mathbf{R}}+\omega\times\mathbf{R}_{QO'}$$

$$=\begin{bmatrix} d(\dot{\theta}+C_\alpha^2\dot{\Omega})+(x_BS_{\beta B}-r_BC_{\beta B})\omega_b+(x_BC_{\beta B}+r_BS_{\beta B})\omega_n \\ -(x_BC_{\beta B}+r_BS_{\beta B})\omega_t+y_B\omega_b \\ -r_mS_\alpha\dot{\Omega}-(x_BS_{\beta B}+r_BC_{\beta B})\omega_t+y_B\omega_n \end{bmatrix}^T\mathbf{Y} \tag{36}$$

5.2.2 Velocity of Any Contact Point P on the Nut

$$\mathbf{V}_{Pn}=-r_mS_\alpha\dot{\Omega}\,[t_\alpha \quad 0 \quad 1]\mathbf{Y} \tag{37}$$

5.2.3 Velocity of Any Contact Point Q on the Screw

$$\mathbf{V}_{QS}=\dot{\Omega}\mathbf{K}\times\mathbf{R}_{QO}$$

$$=\begin{bmatrix}(r_m-r_B C_{\beta B}+x_B S_{\beta B})\,C_\alpha\dot{\Omega}\\ y_B C_\alpha\dot{\Omega}-(r_B S_{\beta B}+x_B C_{\beta B})S_\alpha\dot{\Omega}\\ -(r_m-r_B C_{\beta B}+x_B S_{\beta B})\,S_\alpha\dot{\Omega}\end{bmatrix}^T\mathbf{Y} \qquad (38)$$

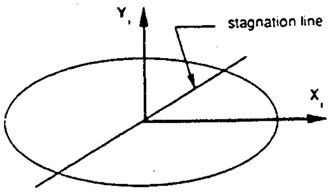


Fig. 7 Stagnation line, line of no-slip along the common normal of the contact surfaces

5.3 Slip Velocities at Contact Points P and Q

$$\mathbf{V}_{SP}=\mathbf{V}_{Pb}-\mathbf{V}_{Pn}$$

$$=\begin{bmatrix}d(\dot{\theta}+\dot{\Omega})+(x_A S_{\beta A}+r_A C_{\beta A})\omega_b+(x_A C_{\beta A}-r_A S_{\beta A})\omega_n\\ -(x_A C_{\beta A}-r_A S_{\beta A})\omega_t+y_A\omega_b\\ -(x_A S_{\beta A}+r_A C_{\beta A})\omega_t-y_A\omega_n\end{bmatrix}^T\mathbf{Y}$$

$$=\begin{bmatrix}-y_A(S_{\beta A}\omega_b+C_{\beta A}\omega_n)-r_A\omega_t\\ x_A(S_{\beta A}\omega_b+C_{\beta A}\omega_n)+d(\dot{\theta}+\dot{\Omega})+r_A(C_{\beta A}\omega_b-S_{\beta A}\omega_n)\\ -x_A\omega_t+y_A(C_{\beta A}\omega_b-S_{\beta A}\omega_n)\end{bmatrix}^T\mathbf{X}_A \qquad (39)$$

$$\mathbf{V}_{SQ}=\mathbf{V}_{Qb}-\mathbf{V}_{QS}$$

$$=\begin{bmatrix}d\dot{\theta}+(x_B S_{\beta B}-r_B C_{\beta B})(\omega_b-C_\alpha\dot{\Omega})+(x_B C_{\beta B}+r_B S_{\beta B})\omega_n\\ (x_B C_{\beta B}+r_B S_{\beta B})(S_\alpha\dot{\Omega}-\omega_t)+y_B(\omega_b-C_\alpha\dot{\Omega})\\ (x_B S_{\beta B}-r_B C_{\beta B})(S_\alpha\dot{\Omega}-\omega_t)-y_B\omega_n\end{bmatrix}^T\mathbf{Y}$$

$$=\begin{bmatrix}-y_B[S_{\beta B}(\omega_b-C_\alpha\dot{\Omega})+C_{\beta B}\omega_n]-r_B(S_\alpha\dot{\Omega}-\omega_t)\\ x_B[S_{\beta B}(\omega_b-C_\alpha\dot{\Omega})+C_{\beta B}\omega_n]+d\dot{\theta}-r_B[C_{\beta B}(\omega_b-C_\alpha\dot{\Omega})-S_{\beta B}\omega_n]\\ x_B(S_\alpha\dot{\omega}-\omega_t)+y_B[C_{\beta B}(\omega_b-C_\alpha\dot{\Omega})-S_{\beta B}\omega_n\end{bmatrix}^T\mathbf{X}_B \qquad .(40)$$

Note that the equations derived in this section can be reduced to the corresponding equations in Section 3, without considering elastic deformations, where $r_i = r_b$ and $x_i = y_i = 0$.

5.4 Slip Velocity and Pattern of Constant Sliding Lines. The pattern of constant sliding lines of ball bearings in the elliptical contact area was investigated by Lundberg (1954) and presented by Harris (1984) in his book. Here a pattern of sliding lines are derived for the BSM directly from the equations of slip-velocity obtained from the previous section. Since the pattern of constant sliding lines are similar for different types of motions, only the case with screw driving and conversion of rotary into linear motion is shown as an example. In accordance with the Hertzian radius of contact in the direction transverse to the motion, the contact surface has a harmonic mean profile radius. This implies that the contact surface is not straight but generally curved. The no-slip condition along the common normal of the contact surfaces between the ball and the nut is obtained from the Z_A-component of Eq. (39) as follows:

$$-x_A\omega_t+y_A\,(\omega_b C_{\beta A}-\omega_n S_{\beta A})=0. \qquad (41)$$

This equation represents a straight line, referred to as the stagnation line, on the $X_A\,Y_A$ plane. This line passes through the origin A, with a slip of $\omega_t/(\omega_b C_{\beta A}-\omega_n S_{\beta A})$, as shown in Fig. 7.

We first assume that the nut is fixed in space. Thus, it tends to "cut-in" the contact surface for every contact point on the ball lying on one side of the stagnation line in the contact ellipse and "leave-from" the contact surface for every point lying on the opposite side of the line. There is a special case, in which $\omega_b C_{\beta A}-\omega_n S_{\beta A}=0$, which indicates that the resultant spinning velocity on the normal plane is along the common normal of the contact surfaces. In this case, the common normal has been referred to as the "spinning axis" in the ball-bearing literature. Additionally, the "stagnation line" becomes the Y_A-axis ($x_A = 0$) which is true because pure spin will not change the depth of "cut-in" of a certain point. Similarly, if $\omega_t = 0$, the "stagnation line" becomes the X_A-axis ($y_A = 0$) because the rolling motion changes the depth of "cut-in."

The equation of constant sliding lines can now be obtained from the X_A- and Y_A-components of Eq. (39); that is

$$(x_A-p_A)^2+(y_A-q_A)^2=C \qquad (42)$$

where

$$p_A=-\frac{d(\dot{\theta}+\dot{\Omega})+r_A(\omega_b C_{\beta A}-\omega_n S_{\beta A})}{\omega_b S_{\beta A}+\omega_n C_{\beta A}},$$

$$q_A=-\frac{r_A\omega_t}{\omega_b S_{\beta A}+\omega_n C_{\beta A}},$$

and

$$C=\frac{\text{constant}}{(\omega_b S_{\beta A}+\dot{\omega}_n C_{\beta A})^2}.$$

Equation (42) represents a family of circles on the $X_A\,Y_A$-plane with center located at (p_A, q_A), where the no-slip condition on the $X_A\,Y_A$-plane occurs, as shown in Fig. 8. It is interesting that the point, (p_A, q_A), will not lie on the stagnation line [Eq. (41)] unless the term, $\dot{\theta}+\dot{\Omega}$, vanishes which is impossible except for the static situation. In other words, a point with no-slip is not possible within the contact ellipse. Note that r_A can also be defined as a variable from Hertzian contact theory and Eq. (42) remains valid. In such a case, Eq.

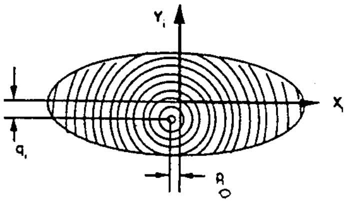


Fig. 8 Pattern of constant sliding lines at constant surfaces

(42) results in a family of non-coplanar circles representative of the true contact surface shape.

As one can see from Eq. (42), under the special case where

$$\omega_b S_{\beta A} + \omega_n C_{\beta A} = 0, \tag{43}$$

the slip-velocity on the X_A Y_A-plane is an invariant which is exactly the same as the slip-velocity at point A without elastic deformation if $r_i = r_b$. The line defined by Eq. (43), which is a line on the normal plane perpendicular to the common normal of the contact surfaces, has been called the "rolling axis" in the case of ball-bearings. It is obvious from Eq. (42) that the no-slip condition is not possible even when spinning is absent.

A similar analysis can be applied to the contact area between the ball and the screw, and the corresponding resultant equations are:

Equation of stagnation line:

$$x_B(S_\alpha\dot{\Omega} - \omega_t) + y_B[(\omega_b - C_\alpha\dot{\Omega})C_{\beta B} - \omega_n S_{\beta B}] = 0 \tag{44}$$

Equation of constant sliding lines:

$$(x_B - p_B)^2 + (y_B - q_B)^2 = C \tag{45}$$

where

$$p_B = -\frac{d\dot{\theta} - r_b[(\theta_b - C_\alpha\dot{\Omega})C_{\beta B} - \omega_n S_{\beta B}]}{(\omega_B - C_\alpha\dot{\Omega})\, S_{\beta B} + \omega_n C_{\beta B}},$$

$$q_B = -\frac{r_B(s_\alpha\dot{\Omega} - \omega_t)}{(\omega_b - C_\alpha\dot{\Omega})S_{\beta B} + \omega_n C_{\beta B}},$$

and

$$C = \frac{\text{constant}}{[(\omega_b - C_\alpha\dot{\Omega})S_{\beta B} + \omega_n C_{\beta B}]^2}.$$

Equation of "pure rolling":

$$(\omega_b - C_\alpha\dot{\Omega})\, S_{\beta B} + \omega_n C_{\beta B} = 0 \tag{46}$$

Equation of "pure spinning":

$$(\omega_b - C_\alpha\dot{\Omega})\, C_{\beta B} - \omega_n S_{\beta B} = 0 \tag{47}$$

Note that the corresponding equations at the ball-nut contact area are independent of the helix angle of the BSM, whereas those at the ball-screw contact area are dependent on the helix angle.

Conclusions

In this paper, we have studied the kinematics of the ball screw mechanism with the aim of understanding the slip conditions and pattern of contact points between the elements. We have derived the general slip conditions and have shown that the condition of no-slip in the central normal plane assumed in the previous literature is theoretically unattainable. The effect of contact deformation on the motion of the balls was also studied and used to determine the pattern of sliding lines of the balls in contact areas.

The results, in addition to their theoretical interest, provide the basis for efficiency analysis, design, wear analysis, and finite element modeling of the ball screw mechanism.

Acknowledgment

This research was supported in part by US Army Research Office grant number DAAL03-90-G-0005 and California Department of Transportation through the AHMCT program.

References

1 Belyaev, V. G., and Drobashevskii, G. S., 1974(a), "Recirculating Ball Nut and Screw Transmissions with Arched and Semi-Circular Thread Profiles," *Russian Engineering Journal*, LIV, No. 9, pp. 18–21.

2 Belyaev, V. G., and Malyuga, V. S., 1983, "The Force Transfer Factor in the Return Channel of a Ball Screw Mechanism," *Soviet Engineering Research*, Vol. 3, No. 2, pp. 78–80.

3 Belyaev, V. G., and Turavinov, V. P., 1981, "Positioning Accuracy of Ball-Screw Mechanism," *Soviet Engineering Research*, Vol. 1, No. 5, pp. 34–36.

4 Drozdov, Y. N., 1984, "Calculating the Wear of a Screw and Nut Transmission with Sliding Friction," *Soviet Engineering Research*, Vol. 4, No. 5, pp. 6-8.

5 Harris, T. A., 1984, *Rolling Bearing Analysis*, John Wiley and Sons, New York, 2nd edition.

6 Jones, A. B., 1959, "Ball Motion and Sliding Friction in Ball Bearings," *Journal of Basic Engineering, Transactions of the ASME*, Vol. 81, No. 1, pp. 1-12.

7 Levit, G. A., 1963a, "Recirculating Ball Screw and Nut Units," *Machines and Tooling*, XXXIV, No. 4, pp. 3-8.

8 Lin, M. C., Velinsky, S. A., and Ravani, B., 1994, "Design of Ball Screw Mechanism for Optimal Efficiency," ASME JOURNAL OF MECHANICAL DESIGN.

9 Mukhortov, V. N., 1982, "Increasing the Life of Anti-Friction Screw-and-Nut Transmissions," *Soviet Engineering Research*, Vol. 2, No. 10, pp. 86–87.

Design of the Ball Screw Mechanism for Optimal Efficiency

M. C. Lin

S. A. Velinsky

B. Ravani

Department of Mechanical
& Aeronautical Engineering,
University of California—Davis,
Davis, CA 95616

This paper develops theories for evaluating the efficiency of the ball screw mechanism and additionally, for designing this mechanism. Initially, a quasi-static analysis, which is similar to that of the early work in this area, is employed to evaluate efficiency. Dynamic forces, which are neglected by the quasi-static analysis, will have an effect on efficiency. Thus, an exact theory based on the simultaneous solution of both the Newton-Euler equations of motion and the relevant kinematic equations is employed to determine mechanism efficiency, as well as the steady-state motion of all components within the ball screw. However, the development of design methods based on this exact theory is difficult due to the extensive computation necessary and thus, an approximate closed-form representation, that still accounts for the ball screw dynamics, is derived. The validity of this closed-form solution is proven and it is then used in developing an optimum design methodology for the ball screw mechanism based on efficiency. Additionally, the self-braking condition is examined, as are load capacity considerations.

Introduction

The ball screw mechanism (BSM) has been used for many years in a wide variety of applications. The most referred to work on the BSM is due to Levit (1963a, 1963b). In his work, Levit has both reviewed the literature prior to his, as well as providing a series of calculations for designing this mechanism. Levit's work has provided the foundation for most of the subsequent work on the design and manufacture of this mechanism including that of Belyaev and associates (1971, 1973, 1974a, 1974b, 1981, 1983), Drozdov (1984) and Mukhortov (1982), to name a few. Unfortunately, Levit includes several improper assumptions causing his results, as well as those based on his results, to be questionable. These errors are noted by Lin et al. (1994), who have taken a fundamental approach in examining the kinematics of the BSM. Basically, Levit does not account for the torsion of the helical path of the individual ball center.

In this paper, we will first examine the efficiency of the BSM using a quasi-static approach much like Levit's, but with the proper kinematics. Since dynamic forces are neglected by the quasi-static analysis, a more exact approach will be taken. In this approach, we will consider the steady-state motion of the ball for a three-point-contact profile (Gothic profile) without deformation by numerically solving the Newton-Euler equations and the relevant kinematic equations simultaneously. While the efficiency of the BSM actually varies as a function of time, the steady-state simplification is valid for many applications and additionally, it can be more easily applied for developing design methods. Closed-form solutions are even more easily applied to the development of design methods and such solutions are derived in this paper. These closed-form solutions, which account for the dynamic forces, are shown to be valid through close agreement with the results of the steady-state theory. A detailed design methodology for the BSM based on the closed-form theory is then presented.

In general, the analysis of the ball screw motion can be divided into two different categories according to the driving component; i.e., nut driving or screw driving. Additionally, each of these categories can be further divided according to the type of input motion; i.e., conversion of rotary into linear motion or conversion of linear into rotary motion. Since the procedures for analyzing the different types of motion are similar and all types of motion are kinematic inversions of each other, in this paper, we will only examine the cases with the screw as the driving component.

Quasi-Static Efficiency Analysis

Levit, in his classic work on the BSM, assumes that there is no-slip between the balls and the nut and screw. Thus, he derives the quasi-static ball screw efficiency by assuming that there is only rolling resistance at the ball/screw and ball/nut contact points. Levit's results imply that the upper bound on ball screw efficiency should be unity. However, recent detailed examination of the kinematics of the BSM proves that the no-slip condition is not attainable and that there must be slip on the normal plane of at least one contact point; see Lin et al. (1994). Therefore, in this work, we have used the correct kinematics and have assumed both slip on the normal plane and rolling resistance in the tangent direction at the contact points in order to determine the quasi-static efficiency of the BSM. Additionally, we assume that there are two contact points between an individual ball and the screw and nut.

For the case of screw driving, suppose that a moment, $M\mathbf{k}$, is acting on the screw in order to overcome an axial load $F_a\mathbf{k}$

Contributed by the Mechanisms Committee for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received March 1990; revised Feb. 1994. Associate Technical Editor: G. L. Kinzel.

applied on the nut. By considering static equilibrium of the screw, the ball and the nut, respectively, we arrive at the following relations:

$$M = r_m Q_n \{ S_\alpha C_\rho [S_\beta + f(a - C_\beta)] + C_\alpha S_\rho \},$$

$$Q_A \approx Q_B \equiv Q_n, \text{ and}$$

$$F_a = -Q_n [C_\alpha C_\rho (S_\beta - f C_\beta) - S_\alpha S_\rho] \qquad (1)$$

where

M = the torque applied on the screw or the nut,
F_a = the axial force applied on the screw or the nut,
Q_i = the normal force at the contact points between the ball and nut and screw, respectively, and the corresponding subscripts:
A and B denote the points between the ball and the nut and between the ball and the screw, respectively,
f = the Coulomb coefficient of friction at the contact points,
r_m = mean radius of the helical path of the ball centerline,
r_b = ball radius,
a = r_b/r_m,
α = helix angle of the path of the ball centerline,
C_i, S_i = cosine and sine functions, respectively, with the angle denoted by the subscript, and we will later use
t_i = tangent function with the angle denoted by the subscript.

The friction angle, ρ, is used to represent the dissipated energy due to rolling friction between the contact surfaces and can be represented by the following expression, which was derived by Levit:

$$\rho = \tan^{-1}\left[\frac{f_r}{r_b S_\beta}\right] \qquad (2)$$

where f_r denotes the rolling coefficient of friction. We additionally assume that the contact angles between the ball and the screw and nut, β_A and β_B, respectively, are equal; i.e., $\beta_A = \beta_B = \beta$.

The efficiency is equal to the ratio of the work done by the output forces to the work done by the input and can be represented as

$$\eta = 1 - \frac{fa + t_\rho/(C_\alpha S_\alpha)}{S_\beta + f(a - C_\beta) + t_\rho/t_\alpha}. \qquad (3)$$

Using a similar approach for the case of nut driving, the efficiency can be simply expressed as

$$\eta = 1 - \frac{f(a + 2C_\beta) + t_\rho/(C_\alpha S_\alpha)}{S_\beta + f(a + C_\beta) + t_\rho/t_\alpha}. \qquad (4)$$

A More Exact Efficiency Analysis

The quasi-static analysis of the preceding section provides an approximation to the ball screw's efficiency since dynamic forces, which will have an effect, have been neglected. In actuality, ball screw efficiency will vary as a function of time due to the system dynamics and the time-dependent nature of the input torque. In order to design with utmost care, one would need to solve the general equations of motion in order to calculate the time varying efficiency, and corresponding design methods could be developed that consider either a maximum or a mean value of efficiency. However, such an approach would be quite tedious and results may be difficult to use for design purposes. By considering steady-state ball screw motion, an approximation of the equations of motion is obtained that still accounts for some of the dynamic forces. This approximation provides a basis for developing a simplified design method and it is valid for many ball screw applications. Since the steady-state theory results in an approximate efficiency, in certain instances, the designer may elect to integrate the general equations of motion in order to monitor the time varying efficiency.

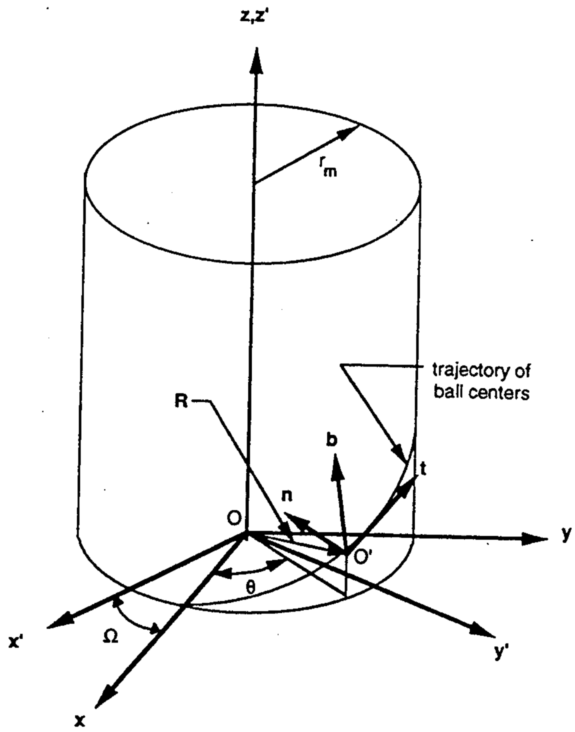


Fig. 1 The position of the ball center, O′, in Cartesian coordinates and Frenet-Serret coordinates

To develop the equations of motion, we first consider the position of the ball center and we employ the following three coordinate frames: a fixed Cartesian frame, $ox'y'z'$, a Cartesian frame rotating with the screw, $oxyz$, and a Frenet-Serret coordinate frame along the ball trajectory, $o'nbt$, as shown in Fig. 1. We then arrive at the Newton-Euler equations which represent the ball motion. The steady-state form of these equations of motion, for the case of converting rotary into linear motion, follow.

$$f(Q_B S_{\psi B} + Q_A S_{\psi A} + Q_{A'} S_{\psi A'}) = 0 \qquad (5a)$$

$$Q_A (C_\beta - f S_\beta C_{\psi A}) - Q_B (C_\beta + f S_\beta C_{\psi B}) + Q_{A'} (C_\beta + f S_\beta C_{\psi A'}) = m r_m (\dot\theta + \dot\Omega)^2 \qquad (5b)$$

$$Q_A (S_\beta + f C_\beta C_{\psi A}) - Q_B (S_\beta - f C_\beta C_{\psi B}) - Q_{A'} (S_\beta - f C_\beta C_{\psi A'}) = 0 \qquad (5c)$$

$$f r_b (Q_B C_{\psi B} - Q_A C_{\psi A} - Q_{A'} C_{\psi A'}) = -I(\dot\theta + \dot\Omega)\omega_n C_\alpha \qquad (5d)$$

$$f r_b S_\beta (Q_B S_{\psi B} - Q_A S_{\psi A} + Q_{A'} S_{\psi A'}) = I(\dot\theta + \dot\Omega)(\omega_t C_\alpha - \omega_b S_\alpha) \qquad (5e)$$

$$-f r_b C_\beta (Q_B S_{\psi B} - Q_A S_{\psi A} - Q_{A'} S_{\psi A'}) = I(\dot\theta + \dot\Omega)\omega_n S_\alpha \qquad (5f)$$

where:

ψ_i = the angle between the direction of the friction force and the normal plane at the contact points, and the corresponding subscripts:
A and A' denote the major and minor contact point between the ball and the nut, respectively, and B denotes the major contact point between the ball and the screw,
m = mass of an individual ball,
I = mass moment of inertia of the ball relative to its mass center,
$\dot\theta$ = the angular velocity of the ball relative to the screw along the helical path,

$\dot{\Omega}$ = the angular velocity of the screw, and

$\omega_n, \omega_t, \omega_b$ = the angular velocity of the ball with respect to its center of mass in the normal, tangential and binormal directions of the helical path.

Note that these equations assume a three-point contact profile between the ball and the nut and screw. With the three-point contact, there are two major contact points, A and B, through which the majority of the load is transferred. Additionally, a minor contact point, A', must exist in order to equilibrate the centripetal force and adequately constrain the ball. We additionally assume that the contact angles between the ball and the screw and nut, β_A, $\beta_{A'}$, and β_B, respectively, in this case, are equal; i.e., $\beta_A = \beta_{A'} = \beta_B = \beta$. From a design perspective, equal contact angles result in consistent loading over all of the contact points helping prevent local failure and premature fatigue. Theoretically, the contact angle can be between 0 deg. and 90 deg. In actuality, the contact angle is limited by geometry and manufacturing techniques and is most often between 45 deg. and 60 deg. The interested reader is again referred to Lin et al. (1994) for a more detailed derivation of these equations, as well as those to follow.

The slip angles, the angles between the friction force directions and the normal plane at the contact points, are better defined as

$$\psi_i = \pi + \tan^{-1}\left(\frac{V_{yi}}{V_{xi}}\right) \qquad (6)$$

where V_{xi} and V_{yi} denote the magnitude of the velocity in the normal plane and the tangential direction, respectively, with respect to the local coordinate systems at the contact points; see Fig. 2. Through a study of the kinematics (see Lin et al., 1994), the following slip velocities result:

For the ball/nut contact point:

$$V_{xA} = -r_b\omega_t \text{ and } V_{yA} = d(\dot{\theta} + \dot{\Omega}) + r_b(\omega_b C_\beta - \omega_n S_\beta) \qquad (7)$$

and for the ball/screw contact point:

$$V_{xB} = r_b(\omega_t - \dot{\Omega} S_\alpha) \text{ and}$$

$$V_{yB} = d\dot{\theta} - r_b[(\omega_b - \dot{\Omega} C_\alpha)C_\beta - \omega_n S_\beta] \qquad (8)$$

where $d = r_m/C_\alpha$.

By imposing static equilibrium on the screw and nut, we have the following expressions which can be used to relate the torque applied to the screw (or nut) to the axial load acting on the nut (or screw):

$$M = ({}^{r}\mathbf{R}_i \times \mathbf{F}_i + {}^{r}\mathbf{R}_{i'} \times \mathbf{F}_{i'}) \cdot \mathbf{k}$$

$$= r_m\{ w_l(Q_{i'} - Q_i) S_\alpha S_\beta + f[-S_\alpha(Q_i C_{\psi i} + Q_{i'} C_{\psi i'})(w_l a + C_\beta)$$

$$+ C_\alpha(Q_i S_{\psi i} + Q_{i'} S_{\psi i'})(1 + w_l a C_\beta)]\} \text{ and}$$

$$F_a = (\mathbf{F}_i + \mathbf{F}_{i'}) \cdot \mathbf{k}$$

$$= w_l(Q_i - Q_{i'}) C_\alpha S_\beta + f[C_\alpha C_\beta(Q_i C_{\psi i} + Q_{i'} C_{\psi i'})$$

$$+ S_\alpha(Q_i S_{\psi i} + Q_{i'} S_{\psi i'})]; \quad i = A, B. \qquad (9)$$

where

${}^{r}\mathbf{R}_i$ = the position vector of contact point i with respect to the rotational Cartesian coordinate system (see Fig. 1),

$\mathbf{F}_i$ = the force vector applied on the ball at contact point i, and

the weighting function, w_l, allows distinction between the various contact points and is defined as

$w_l = 1$ for $i = A$ (A denotes the ball/nut contact)

$= -1$ for $i = B$ (B denotes the ball/screw contact).

Equation (9) provides two simultaneous equations for either the screw driving or the nut driving case. When a torque is applied to the nut, the subscript $i = A$ will apply for the moment equation and the subscript $i = B$ will apply for the force equation since an axial force will be acting on the screw. When the torque is applied to the screw, $i = B$ applies for the moment equation and $i = A$ applies for the force equation.

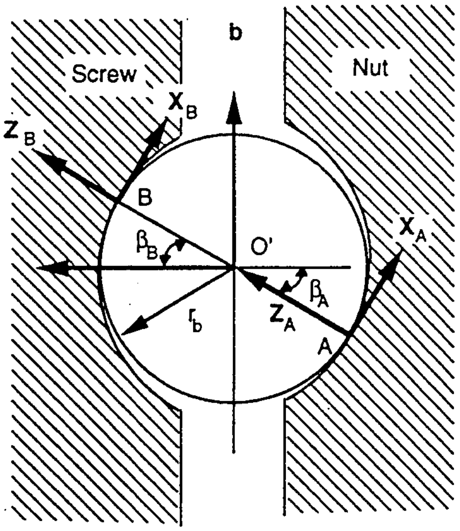


Fig. 2 Coordinate systems at the contact points

Equations (5)–(9) have been solved numerically for the screw driving case with the following ball screw parameter values: r_b = 4.37 mm (0.172 in.), r_m = 24.3 mm (0.956 in.), $\dot{\Omega}$ = 2000 rpm, and α = 10 deg. for varying contact angles or β = 45 deg. for varying helix angles. The coefficient of friction, f, is assumed equal to 0.075, a value representative of the contact conditions normally found in the BSM and this value will be used for the remainder of the paper. Additionally, it is assumed that an axial load, F_a, equal to 2113 Nt (475 lb) is resisting motion of the nut. Figures 3 and 4 depict the results. We show the relationships between efficiency and contact angle and helix angle in Fig. 3. Figure 4 displays the normal loads as a function of contact angle. In this figure, the curve represents the load at the major contact points, A and B, which, for all practical purposes, are equal, and the load at the minor contact point, A', which is orders of magnitude smaller, is not distinguishable from the abscissa.

The Approximate Closed-Form Solution

The theory developed above involves the numerical solution of the set of equations representing the steady-state motion of the ball screw and designing ball screw mechanisms using this theory is obviously quite cumbersome. The current section derives an approximate closed-form solution for the ball screw motion and considerable insight is gained into the sensitive design parameters. This solution also allows us to investigate the optimal design of the BSM.

For the mechanism operating at relatively low speed, which is true for most BSM currently used in industry, it is reasonable to assume that the centripetal force is very small compared with the normal loads. Additionally, from the numerical results above, we recognize that the normal load at the minor contact point is very small compared to the normal loads at the two major contact points. The third contact point is necessary to provide a stable system, but has only little effect on the resultant motion at these relatively low speeds. Thus, $Q_A \approx Q_B$, indicating that a two-point-contact model is a reasonable approximation. For the particular size of balls used in this model, we find that this approximation is valid for screw speeds as high as 2,000 rpm.

With the assumption that $Q_{A'} = 0$, from Eq. (5a), we obtain

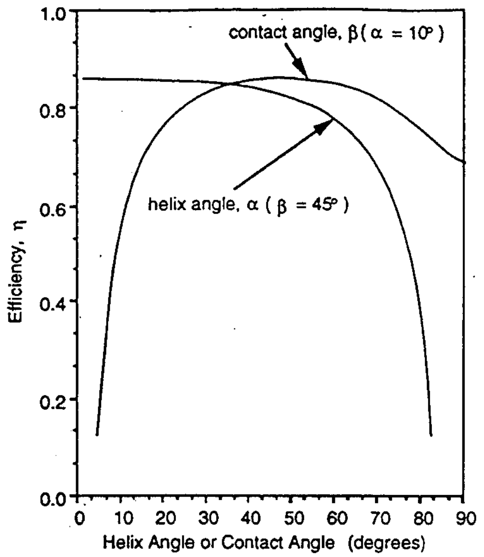


Fig. 3 Efficiency as a function of helix angle or contact angle—conversion of rotary into linear motion

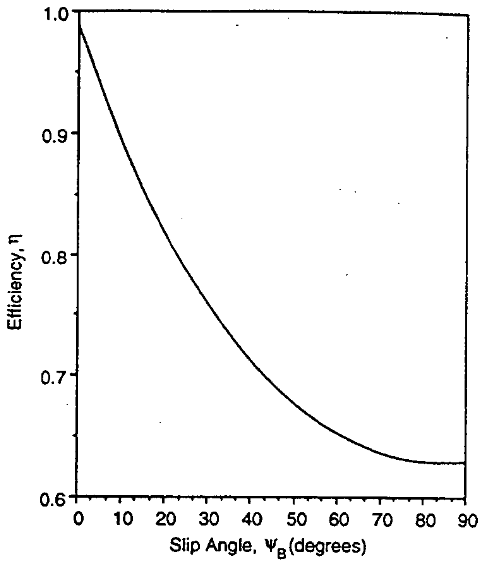


Fig. 5 Efficiency as a function of the slip angle between the ball and the screw

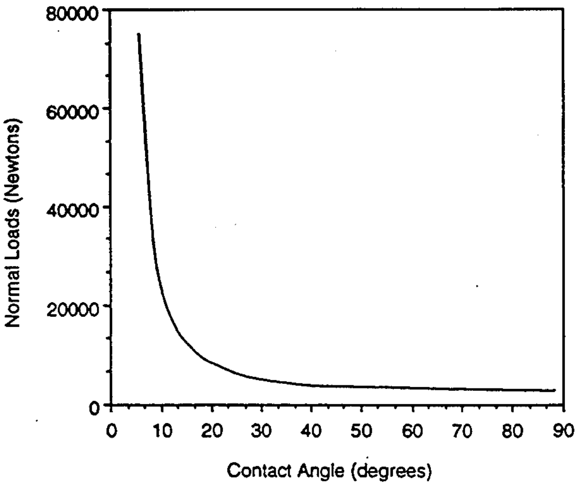


Fig. 4 Normal loads as a function of contact angle—conversion of rotary into linear motion, screw driving

$$\frac{Q_A}{Q_B} = -\frac{S_{\psi B}}{S_{\psi A}} \cong 1 \tag{10}$$

which implies that $\psi_A \approx \pi + \psi_B$. Note that this result also agrees well with the numerical results.

We note that the nut moves a distance equal to $-\Omega L/2\pi = -\Omega r_m t_\alpha$ (where L denotes the lead of the screw) as the screw rotates through an angle Ω. Thus, from the above assumptions and Eq. (9), the efficiency of the BSM, η, can now be written in closed-form as

$$\eta = \frac{S_\beta - f(C_\beta C_{\psi A} + S_{\psi A} t_\alpha)}{S_\beta + f[C_{\psi B}(a - C_\beta) + S_{\psi B}(1 - aC_\beta)/t_\alpha]}. \tag{11}$$

It is now simple to observe the effects of the various parameters on efficiency.

Figure 5 shows the variation in efficiency with respect to the slip angle at the screw/ball contact point, ψ_B. In this figure, the following parameter values are used: $a = 0.1$ and $\alpha = 10$ deg. The curve indicates that the friction along the tangential direction dissipates more energy than that on the normal plane. In other words, the component of frictional force along the tangential direction plays a more important role, from the efficiency point of view, than that on the normal plane. Thus, the upper and lower limits of efficiency are at $\psi_B = 0$ deg. (normal plane) and $\psi_B = 90$ deg. (tangential direction), respectively. We note that we could arrive at the same conclusions by using the slip angle at the screw/ball contact point, ψ_A, since $\psi_A \approx \pi + \psi_B$.

Furthermore, the following equation is obtained by dividing Eq. (5*f*) by Eq. (5*d*):

$$\cot(\psi_A) + \cot(\psi_B) = \frac{2C_\beta}{t_\alpha}. \tag{12}$$

By combining Eqs. (10) and (12), we have

$$\psi_B = \psi_A - \pi = \tan^{-1}\left(\frac{t_\alpha}{C_\beta}\right). \tag{13}$$

Therefore, Eq. (11) can be further simplified to

$$\eta = 1 - \frac{f}{C_\alpha^2 S_\beta\left[\sqrt{C_\beta^2 + t_\alpha^2} + fS_\beta\right]} \tag{14}$$

and we note that the only parameters affecting efficiency from this equation are the coefficient of friction, the helix angle and the contact angle.

Using Eq. (14), we find that the efficiencies generated are nearly identical to the numerical results obtained from the steady-state solution of the Newton-Euler equations shown in Fig. 3 where the maximum difference between the theories for the efficiency versus helix angle relationship is 0.33 percent and between the theories for the efficiency versus contact angle relationship is 0.86 percent. Thus, Eq. (14) is a valid representation of the efficiency of the BSM for this type of motion. Further investigation of Fig. 3 reveals that an optimum contact angle exists for peak efficiency. This existence of an optimum can be explained as follows. If one considers only the friction on the normal plane, the efficiency is proportional to the contact angle, which can be observed through Eq. (11). However, from Fig. 5, the component of frictional force along the tangential direction, which dissipates more energy than that on the normal plane, becomes more important at higher contact angles. Hence, a peak efficiency exists at a contact angle at which the two frictional components reach some particular ratio. To obtain the optimum contact angle mathematically, we differentiate the denominator of Eq. (14) with respect to

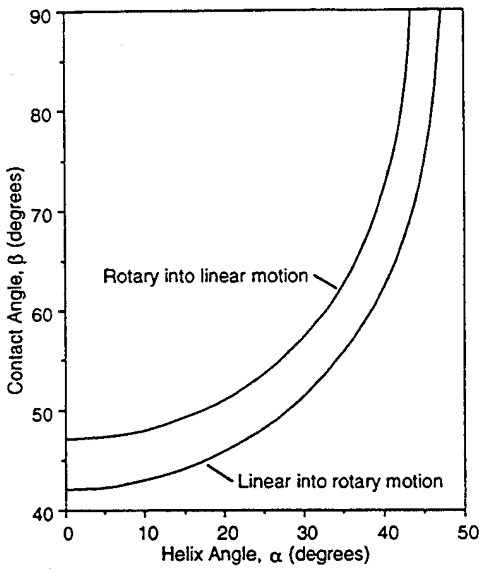


Fig. 6 Relationship between contact angle and helix angle to achieve optimal efficiency

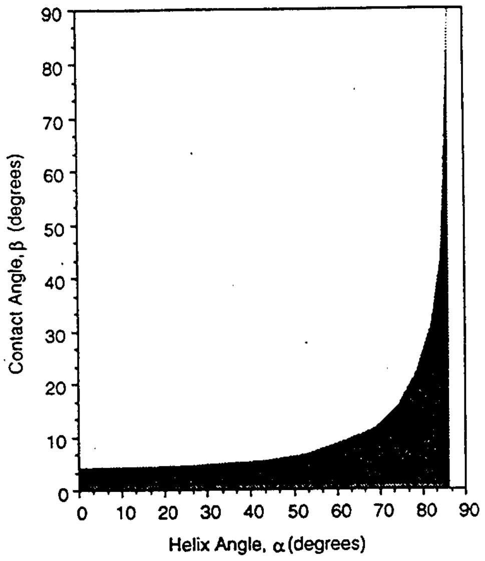


Fig. 7(*a*) The self-braking design space—conversion of rotary into linear motion

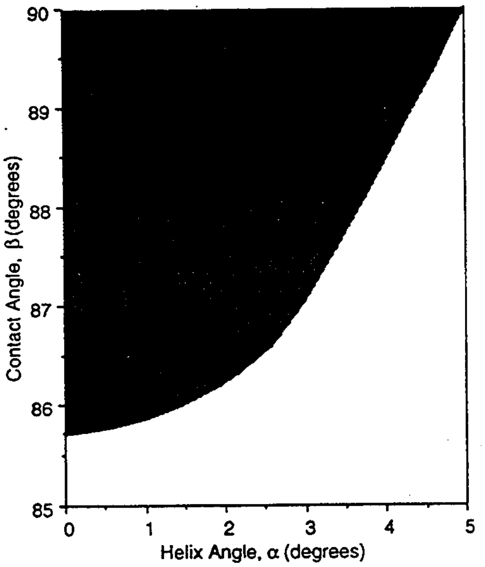


Fig. 7(*b*) The self-braking design space—conversion of linear into rotary motion

the contact angle and set the derivative equal to zero, resulting in the following expression.

$$C_\beta^2 + t_\alpha^2 + 2fS_\beta\sqrt{C_\beta^2 + t_\alpha^2} = S_\beta^2 \tag{15}$$

One notes, from this equation, that the optimum contact angle can be expressed as a function of helix angle. The resultant relationship is depicted in Fig. 6. This curve indicates that the peak efficiency exists only at lower helix angles and higher contact angles. The optimum efficiency is arrived at by merely solving Eqs. (14) and (15) simultaneously. The resulting optimal efficiency value is 86.1 percent for all helix angles and the corresponding contact angle from Eq. (15), and the coefficient of friction equal to 0.075.

Self-braking is that condition in which it is kinematically impossible for the mechanism to perform useful work and is, of course, highly undesirable. The self-braking condition is determined by combining Eqs. (11) and (13) and results when the numerator of Eq. (11) becomes negative; i.e.,

$$S_\beta \leq f\sqrt{C_\beta^2 + t_\alpha^2}. \tag{16}$$

The shaded region below the curve in Fig. 7a represents the design space in which self-braking occurs. We note that self-braking takes place for designs with low contact angles for the helix angles normally used and additionally, that the self-braking condition always takes place at the driven side of the mechanism.

Conversion of Linear into Rotary Motion. Employing a similar procedure as above, we obtain the efficiency for the case of converting linear into rotary motion as

$$\eta = 1 - \frac{f}{C_\alpha^2\left[S_\beta\sqrt{C_\beta^2 + t_\alpha^2} + f(C_\beta^2 + t_\alpha^2)\right]}. \tag{17}$$

It is again easy to develop relationships between efficiency and contact angle and helix angle. For this case, the trends are the same as the case of converting rotary into linear motion, but with slightly different values. Now, the peak efficiency occurs when the following expression holds:

$$C_\beta^2 + t_\alpha^2 = S_\beta^2 + 2fS_\beta \tag{18}$$

and the resulting relationship between contact angle and helix angle is additionally shown in Fig. 6. This curve also indicates that the peak efficiency exists only at lower helix angles and higher contact angles. The optimum efficiency is arrived at by merely solving Eqs. (17) and (18) simultaneously. As in the case of conversion of rotary to linear motion, the optimal efficiency value is 86.1 percent for all helix angles and the corresponding contact angle from Eq. (18), and the coefficient of friction equal to 0.075. In this case, the self-braking condition occurs when

$$fS_\beta \geq \sqrt{C_\beta^2 + t_\alpha^2}. \tag{19}$$

The shaded region above the curve in Fig. 7(*b*) depicts the design space for self-braking and this takes place only for designs with low helix angles and high contact angles.

Load Capacity Considerations

The normal load transmitted through the contact surfaces should be designed to be as low as possible for a given loading condition to attain the highest possible load capacity. For the case of conversion of rotary into linear motion, we consider a constant load F_a. Then, by neglecting the friction force terms which result in surface tractions as opposed to normal loads, from Eq. (9), we notice that causing $C_\alpha S_\beta$ to be as high as possible results in the lowest normal loads. This implies that designs with small helix angles and large contact angles give the BSM the maximum load capacity. With the ball-screw designed such that it operates at its maximum efficiency, we find that the term $C_\alpha S_\beta$ is constant for all helix angles and the corresponding contact angle from Eq. (15), and thus, the load carrying capacity cannot be improved for this optimum case.

For the case of conversion of linear into rotary motion, assuming constant load M, Eq. (9) reveals that the lowest normal loads occur for a high $r_m S_\alpha S_\beta$ term. This indicates that with large mean radius, helix angle and contact angle, the BSM has the highest load capacity. Additionally, we note that the parameter a (ratio of ball radius to mean radius) is not a critical parameter within the range found in most industry applications; i.e., usually, $0.07 \leq a \leq 0.2$. Thus, for this case, one would like to choose the highest contact angle that is possible given manufacturing constraints and select the corresponding helix angle from Eq. (18) for optimum efficiency in order to design the most efficient, highest load carrying ball screw mechanism.

Design Procedure

Based on the results arrived at in the preceding analysis, the following methodology is recommended for the design of the BSM:

(*1*) Select a preliminary helix angle based on the discussion related to load capacity and the corresponding motion.

(*2*) Determine the contact angle for optimal efficiency from either Eq. (15) or (18).

(*3*) Choose the screw length according to the application requirements.

(*4*) Determine the mean radius such that $r_m \geq$ screw length/60, as noted by Levit (1963b).

(*5*) Select the ball size such that $0.07 \leq a \leq 0.2$.

(*6*) Determine the thread profiles such that $0.95 \leq r_b/r_2 \leq 0.97$, where r_2 = the radius of curvature of the thread profile for the nut or screw.

(*7*) Determine the approximate contact force from the specified resistive force or moment and the equations shown below, which were derived from Eq. (9) with friction neglected.

For the conversion of rotary into linear motion:

$$Q_i = \frac{F_a}{C_\alpha S_\beta} \tag{20}$$

For the conversion of linear into rotary motion:

$$Q_i = \frac{M}{r_m S_\alpha S_\beta} \tag{21}$$

(*8*) Determine the relative motion between the screw and the nut from the following relationship:

advance distance = $(r_m t_\alpha)$ (angular displacement).

(*9*) Calculate the maximum contact stress using the approximate contact load, the ball radius, the thread radius of curvature and Hertz' theory.

(*10*) Determine the efficiency of the BSM from Eqs. (14) or (17).

This procedure should be an iterative process with a new set of ball screw dimensions selected if any of the criteria are violated. For instance, if the contact stresses determined from the Hertzian theory are too high, most likely larger dimensions are necessary. Similarly, if the relative motion is not appropriate (size constraints are violated), a new helix angle, as well as other dimensions, may be necessary. Additionally, this procedure has not discussed the speed requirements nor the available motor power of the application. Obviously, these factors will also affect the final design.

Conclusions

This paper provides a powerful set of tools for analyzing and designing the ball screw mechanism. The exact steady-state motion within the ball screw mechanism has been determined by numerically solving the Newton-Euler equations of motion and the kinematic equations simultaneously. This paper has developed a simplified closed-form solution for ball screw motion and has employed this theory in considering the optimum design of this mechanism. In terms of efficiency, the driving component does not have an effect. That is, only the type of motion conversion is important. Thus, the necessary relationship between contact angle and helix angle for optimum efficiency has been formulated for all types of ball screw motion. Additionally, the conditions for self-braking have been discussed as well as the load capacity of the BSM. The results of the closed-form solution have been presented in a dimensionless manner allowing this work to be applied to the design of any ball screw mechanism. Finally, a complete design procedure for the BSM has been developed.

Acknowledgment

This research was supported in part by US Army Research Office grant number DAAL03-90-G-005 and the California Department of Transportation through the AHMCT program.

References

Belyaev, V. G., 1971, "Re-entry of Balls in Recirculating Ball-screw-and-nut Mechanisms," *Russian Engineering Journal*, Vol. LI, No. 11, pp. 30–34.

Belyaev, V. G., and Kogan, A. I., 1973, "Effect of Geometrical Errors of Ball Contact Angle in Ball-Screw Transmissions," *Machines & Tooling*, No. 5, pp. 25–29.

Belyaev, V. G., and Drobashevskii, G. S., 1974a, "Recirculating Ball Nut and Screw Transmissions with Arched and Semi-circular Thread Profiles," *Russian Engineering Journal*, Vol. LIV, No. 9, pp. 18–21.

Belyaev, V. G., and Kogan, A. I., 1974b, "Effect of Screw Diameter Variations on Accuracy and Stiffness of Recirculating Ball Nut-Screw Pairs," *Machines & Tooling*, Vol. XLV, No. 9, pp. 16–18.

Belyaev, V. G., and Turavinov, V. P., 1981, "Positioning Accuracy of Ball-Screw Mechanism," *Soviet Engineering Research*, Vol. 1, No. 5, pp. 34–36.

Belyaev, V. G., and Malyuga, V. S., 1983, "The Force Transfer Factor in the Return Channel of a Ball and Screw Mechanism," *Soviet Engineering Research*, Vol. 3, No. 2, pp. 78–80.

Drozdov, Y. N., 1984, "Calculating the Wear of a Screw and Nut Transmission with Sliding Friction," *Soviet Engineering Research*, Vol. 4, No. 5, pp. 6–8.

Levit, G. A., 1963a, "Recirculating Ball Screw and Nut Units," *Machines and Tooling*, Vol. XXXIV, No. 4, pp. 3–8.

Levit, G. A., 1963b, "Calculations of Recirculating Ball Screw and Nut Transmissions," *Machines and Tooling*, Vol. XXXIV, No. 5, pp. 9–16.

Lin, M. C., Ravani, B., and Velinsky, S. A., 1994, "Kinematics of the Ball Screw Mechanism," ASME JOURNAL OF MECHANICAL DESIGN, *in press*.

Mukhortov, V. N., 1982, "Increasing the Life of Anti-Friction Screw-and-Nut Transmissions," *Soviet Engineering Research*, Vol. 2, No. 10, pp. 86–87.

ON THE ANALYSIS AND DESIGN OF HIGH SPEED THRUST BALL BEARINGS

M.C. LIN, S.A. VELINSKY, B. RAVANI
Department of Mechanical Engineering, University of California-Davis,
Davis, CA, USA

Abstract. In this paper, a complete solution of the steady-state Newton-Euler equations describing the motion of the balls in high speed thrust ball bearings is presented. Kinematic equations are derived and used as ancillary equations to obtain the numerical solution of the Newton-Euler equations. This differs from previously reported work where extraneous assumptions (instead of kinematic analysis) are used to allow solutions of these equations. It is shown that the no-slip condition for the ball motion reported in the literature is based on an unnecessary assumption. The correct form of the condition is developed and used in the analysis of the ball motion. The analysis shows the importance of contact angle on both ball motion and contact force. An approximate theory is then developed for determining the optimum contact angle that would result in minimum contact force in high speed ball bearings. The results are compared with experimental data, and theory is illustrated by an example.

Keywords. Thrust ball bearings; high speed; kinematics; design.

INTRODUCTION

The ball motion within ball bearings was first investigated by Jones (1959) where he derived equations governing the steady-state motion of the ball. Jones did not solve the equations that he derived and only used them to investigate frictional effects in ball bearings. Subsequently, Harris (1971) made several simplifying assumptions that made it possible to solve Jones' equations obtaining numerical results for the motion of the ball. Harris' technique, also presented in Harris (1984), includes an assumption that there is a sufficiently large static coefficient of friction between the ball and the raceway to prevent ball rotation due to the gyroscopic moment. Additionally, he assumed that there is no-slip at the contact points in certain directions. Later, Gupta (1975, 1979) determined the complete solution to the differential equations of motion of the ball by including contact deformations. He also considered full elastohydrodynamic lubrication.

In this paper, we show that neither the so called race control assumption of Jones and Harris nor the inclusion of the contact deformation are necessary for obtaining solutions to the equations representing ball motion in high speed thrust ball bearings. For such devices, one can assume three points of contact between the ball and the raceways and solve the equations of motion. This is accomplished through the use of kinematic analysis to develop ancillary equations, which when combined with the steady-state equations, allow for numerical solutions describing the motion of the ball. In performing the kinematic analysis, we develop theoretically correct no-slip conditions and show that they differ from those in the literature; see, e.g., Palmgren (1945), Allen (1964), Houghton (1976), Hamrock and Dowson (1981), and Harris (1984). We then apply the results and develop an approximate, but closed-form, theory suitable for design purposes. This approximate theory is used to develop a design strategy for optimum contact angle.

KINEMATICS OF THRUST BALL BEARINGS

Classical ball bearing kinematics assumes that the velocity of the ball center can be obtained as the mean of the inner and outer raceway velocities, and additionally, the relative angular velocity between the inner raceway and the ball is defined as $(\omega_m - \omega_i)$. Accordingly, previous researchers have developed the following relationship between angular velocities for the no-slip condition:

$$(\omega_m - \omega_i)(r_m - r_b) = r_b \omega_b \tag{1}$$

where

r_m = mean radius of the path of the ball center,

r_b = ball radius, and

ω_b = the angular velocity of the ball with respect to its center.

However, the arms of rotation of the ball and the raceways are different such that the angular velocities of the various elements of the bearing are not additive. Hence, the correct equation for angular velocities with the no-slip condition is actually

$$r_m \omega_m - r_b \omega_b = (r_m - r_b)\,\omega_i \tag{2}$$

where, here, we define

ω_m = the orbital angular velocity, and

ω_i = the angular velocity of the inner raceway with respect to the ground.

By comparing the above two equations, we recognize that the arm of rotation of the ball center from Eqn. (2) is equal to r_m as opposed to $(r_m - r_b)$ indicated by Eqn. (1). Depending on the relative size of the balls and the raceway, the error associated with use of Eqn. (1) could be considerable.

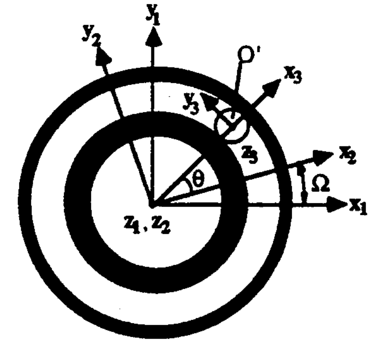


Fig. 1. The coordinate systems employed for the kinematics of thrust ball bearings.

We now study the kinematics of the thrust ball bearing for the case in which the outer raceway is fixed and the inner raceway is rotating. The case with the inner raceway fixed is merely a kinematic inversion of this case. We employ the following three coordinate frames: a fixed Cartesian frame, $ox_1y_1z_1$, a Cartesian frame rotating with the inner raceway,

$ox_2y_2z_2$, and a ball centered coordinate frame moving along the ball trajectory, $o'x_3y_3z_3$, as shown in Fig. 1. Fig. 2 shows the local coordinate system, $Bx_4y_4z_4$, at the contact point between the ball and the inner raceway which will be used in developing the slip velocities at this point.

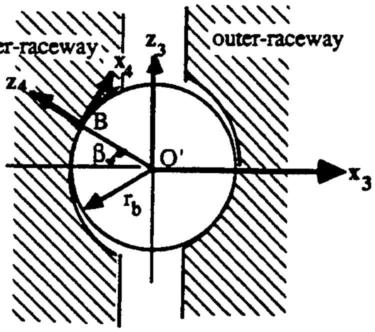


Fig. 2. Local coordinate systems at the contact point.

With the assumption of rigid body motion, we arrive at the following velocity of the contact point on the ball:

$$V_{Bb} = \dot{R}_{O'} + \omega \times r_b k_4$$

$$= \begin{bmatrix} r_b\omega_y \\ r_m(\dot\theta + \Omega) - r_b\left(\omega_x S_\beta + \omega_z C_\beta\right) \\ 0 \end{bmatrix}^T X_4 \qquad (3)$$

where

$$\omega = [\omega_x \;\omega_y \;\omega_z]X_3,\; X_3 = [i_3 \;j_3 \;k_3]^T,\; X_4 = [i_4 \;j_4 \;k_4]^T,$$

$\dot{R}_{o'}$ = the absolute velocity of the origin of the ball centered coordinate system,

$\dot\theta$ = the angular velocity of the ball along its circular path relative to the inner-raceway,

$\dot\Omega$ = the angular velocity of the inner-raceway,

ω_x, ω_y, ω_z = the angular velocity of the ball with respect to its center of mass in the x_3, y_3, z_3 directions, respectively, of the ball centered coordinate system,

C_i, S_i = cosine and sine functions, respectively, with the angle denoted by the subscript, and

β = the contact angle at the ball/inner raceway contact point.

Likewise, the velocity of the contact point on the inner raceway can be expressed as

$$V_{Bi} = \dot\Omega k_2 \times R_B = \begin{bmatrix} 0 \\ \left(r_m - r_b C_\beta\right)\dot\Omega \\ 0 \end{bmatrix}^T X_2 \qquad (4)$$

where

$$X_2 = [i_2 \;j_2 \;k_2]^T,$$

and R_B denotes the position of the contact point with respect to the fixed coordinate system.

Thus, the slip velocity at contact point B between the ball and the inner raceway can be expressed as

$$V_{SB} = V_{Bb} - V_{Bi} = \left[V_{xB} \;V_{yB} \;V_{zB}\right]^T$$

$$= \begin{bmatrix} r_m\omega_y \\ r_m\dot\theta + r_b C_\beta\dot\Omega - r_b(\omega_x S_\beta + \omega_z C_\beta) \\ 0 \end{bmatrix}^T X_4. \qquad (5)$$

In a similar manner, the slip velocity at the ball/outer raceway contact point, A, can be represented as

$$V_{SA} = \left[V_{xA} \;V_{yA} \;V_{zA}\right]^T$$

$$= \begin{bmatrix} -r_b\omega_y \\ r_m(\dot\theta + \dot\Omega) - r_b(\omega_x S_\beta + \omega_z C_\beta) \\ 0 \end{bmatrix}^T X_4. \qquad (6)$$

No-slip conditions can now be derived from these last two equations. From the x-direction slip velocities:

$$\omega_y = 0 \qquad (7)$$

and from the y-direction slip velocities:

$$\omega_x = 0 \qquad (8)$$

$$\frac{\dot\theta + \dot\Omega}{\dot\Omega} = \frac{1 \pm aC_\beta}{2} \qquad (9)$$

$$\frac{\omega_z}{\dot\Omega} = \frac{aC_\beta \pm 1}{2aC_\beta} \qquad (10)$$

where $(\dot\theta+\dot\Omega)/\dot\Omega$ can be thought of as the orbital velocity of the ball, $\omega_z/\dot\Omega$ as the spinning velocity of the ball, $a = r_b/r_m$, the ratio of the ball size to the mean radius of its path, and the "+" sign applies for outer-raceway driving and the "-" sign applies for inner-raceway driving.

DYNAMICS OF BALL MOTION

By considering equilibrium of the individual ball, we write the steady-state Newton-Euler equations governing ball motion in thrust ball bearings as follows:

$$f(Q_B S_{\psi B} + Q_A S_{\psi A} + Q_A S_{\psi A'}) = 0 \qquad (11a)$$

$$Q_A(C_\beta - fS_\beta C_{\psi A}) - Q_B(C_\beta - fS_\beta C_{\psi B}) + Q_{A'}(C_\beta - fS_\beta C_{\psi A'})$$
$$= mr_m(\dot\theta + \dot\Omega)^2 \qquad (b)$$

$$Q_A(S_\beta + fC_\beta C_{\psi A}) - Q_B(S_\beta - fC_\beta C_{\psi B}) - Q_{A'}(S_\beta - fC_\beta C_{\psi A'}) = 0 \qquad (c)$$

$$fr_b(Q_B C_{\psi B} - Q_A C_{\psi A} - Q_{A'} C_{\psi A'}) = I(\dot\theta + \dot\Omega)\omega_x \qquad (d)$$

$$fr_b S_\beta(Q_B S_{\psi B} - Q_A S_{\psi A} + Q_{A'} S_{\psi A'}) = I(\dot\theta + \dot\Omega)\omega_y \qquad (e)$$

$$-fr_b C_\beta(Q_B S_{\psi B} - Q_A S_{\psi A} - Q_A S_{\psi A'}) = 0 \qquad (f)$$

where:

Q_i = the normal force at the contact points between the ball and outer- and inner-raceway, respectively,

ψ_i = the angle between the direction of the friction force and the normal plane at the contact points, and the corresponding subscripts:
A and A' denote the major and minor contact point between the ball and the outer-raceway, respectively, and B denotes the major contact point between the ball and the inner-raceway,

f = the Coulomb coefficient of friction at the contact points,

m = mass of an individual ball, and

I = mass moment of inertia of the ball relative to its mass center.

Note that we assume that there are three points of contact between the ball and the inner and outer raceways, and that the contact angles are equal at these contact points; i.e., $\beta_A = \beta_{A'} = \beta_B = \beta$. Additionally, we note that these governing equations are identical to those of Jones (1959). Also, the above equations are valid for the case of the inner raceway fixed by allowing $\dot\Omega$ to represent the outer raceway angular velocity.

The slip angles, the angles between the friction force directions and the normal plane at the contact points, are better defined as

$$\psi_i = \pi + \tan^{-1}\left(\frac{V_{yi}}{V_{xi}}\right) \tag{12}$$

where V_{xi} and V_{yi} denote the magnitude of the slip velocity in the normal plane and the tangential direction, respectively, with respect to the local coordinates at the contact points; see Fig. 2. These slip velocities have been derived in the preceding section and appear in Eqns. (5) and (6). Equation (12), which is based on the kinematic analysis, dictates the directions of the friction forces at the contact points. These equations are the ancillary conditions which allow the solution of the Newton-Euler equations without additonal simplifying assumptions. We note that the kinematics employed assumes rigid body motion, but that it would be reasonably simple to extend this to include contact deformation.

Now, assume that the thrust load, F_a, is applied by the shaft. By imposing static equilibrium on the inner-raceway, one obtains the following equation:

$$F_a = Q_B(S_\beta - f\, C_\beta C_{\psi B}). \tag{13}$$

Steady-state ball motion is then determined by the simultaneous solution of the ten nonlinear algebraic equations (11-13) which determines the ten unknowns: $\dot{\theta}$, ω_x, ω_y, ω_z, Q_A, $Q_{A'}$, Q_B, ψ_A, $\psi_{A'}$ and ψ_B. We have obtained solutions of these equations using a modified Newton's method; see Forsythe, Malcolm and Moler (1977). Results of the numerical solution are shown in Fig. 3 in which we compare ball velocities to the experimental results of Poplawski and Mauriello (1969). We employ the following parameter values: r_b = 4.37 mm (11/64 in.), r_m = 24.27 mm (0.9555 in.), β = 24.5°, f = 0.075, and $\dot{\Omega}$ = 35000 rpm.

Recognizing that the contact angle plays a critical role in the resultant characteristics of high-speed thrust ball-bearings, we employ the above equations and show the variation in ball motion with respect to contact angle in Fig. 4. We have additionally shown the effect of contact angle on the normal load at the ball/outer-raceway contact point in Fig. 5. In addition to the noted parameter values, Figs. 4 and 5 are based upon: F_a = 1780 N (400 lb) and shaft speed, $\dot{\Omega}$ = 27,500 rpm. Of considerable interest in Fig. 5 is that a minimum of the normal load Q_A exists at some contact angle. Should the thrust bearing be designed at such a contact angle, its load capacity and life could be maximized.

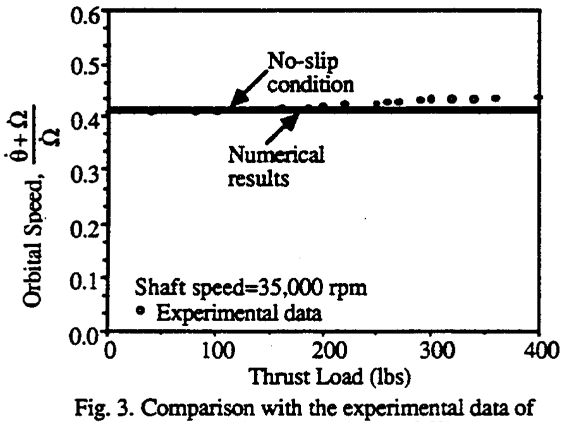


Fig. 3. Comparison with the experimental data of Poplawski and Mauriello (1969).

APPROXIMATE THEORY FOR DESIGNING HIGH-SPEED THRUST BEARINGS

The theory developed above involves the numerical solution of a set of equations and the design of ball bearings using this theory is obviously quite cumbersome. In this section, we provide an approximation to the theory that allows a closed-form solution and the development of a simple methodology for determining the optimum contact angle.

First, consider the no-slip conditions, Eqns. (7)-(10), derived earlier in this paper. These conditions provide a closed-form representation of the ball motion and results are additionally shown in Fig. 3. Evidently, with the low coefficient of friction normally found in ball bearings, the no-slip assumption seems to provide accurate results and thus, it can be quite useful for developing a design methodology.

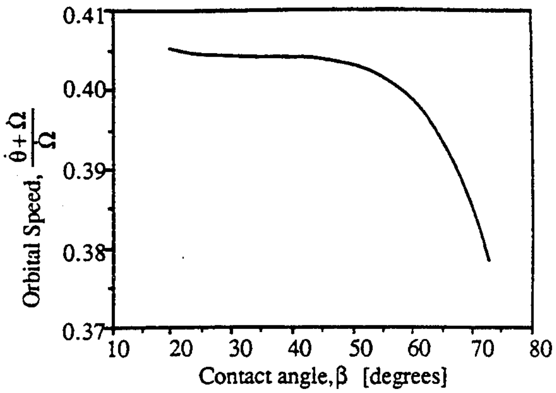


Fig. 4. Orbital speed vs contact angle.

We now recognize that the normal load consists of two major components: one from the thrust force and the other from the centripetal force. Therefore,

$$Q_A \equiv \frac{F_a}{S_\beta} + \frac{CF}{2C_\beta} \tag{14}$$

where the centripetal force, $CF = mr_m(\dot{\theta}+\dot{\Omega})^2$. The first term of the above equation is obtained from equation (13) by neglecting the friction effect (which is small) and the second term is a result of the centripetal force which is shared by the two contact points between the ball and the outer-raceway (see Fig. 6). As we can see from Eqn. (14), the thrust load plays a greater role at lower contact angles, whereas, the centripetal force is more important at higher contact angles. Thus, the optimal contact angle depends on the ratio between thrust load and centripetal force.

Since, the no-slip condition along the tangential direction is a reasonable approximation for determining the motion in ball bearings, we substitute Eqn. (9) into (14) and set the first derivative of Q_A with respect to the contact angle equal to zero. Thus, we arrive at this condition for minimizing the normal load:

$$\frac{8\,F_a}{m\,r_m\Omega^2} = \left(1 - a^2C_\beta^2\right)t_\beta^3 \tag{15}$$

where t_β denotes the tangent of the contact angle, β. By recognizing that $a^2C_\beta^2 \ll 1$, we arrive at the following expression for the optimum contact angle:

$$\beta = \tan^{-1}\left[\sqrt[3]{\frac{8\,F_a}{m\,r_m\Omega^2}}\right]. \tag{16}$$

Fig. 7 shows a plot of the optimum contact angle based on the above equation. Note that Eqn. (16) is valid for both inner-

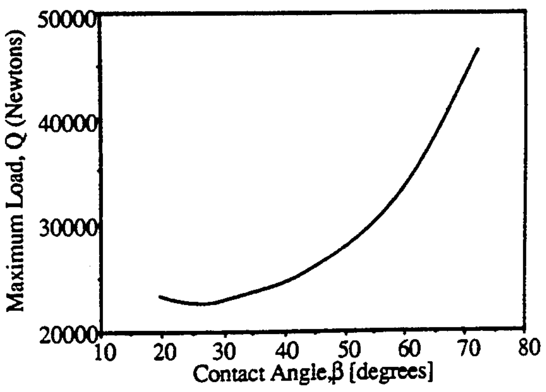


Fig. 5. Maximum contact load as a function of contact angle.

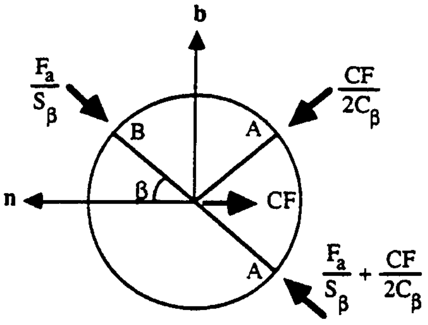


Fig. 6. Free-body diagram of the ball.

and outer-raceways driving. However, according to the no-slip conditions derived above, the orbital speed is about 14% higher for outer-raceway driving as opposed to inner-raceway driving. Thus, the centripetal force is about 29% higher for the outer-raceway driving case. This could be crucial when operating at extremely high speeds. Since the no-slip assumption overestimates the centripetal force compared to the exact solution, the desired contact angle should be slightly larger than that obtained from Eqn. (16).

The dimensionless parameter, $8F_a/mr_m\dot{\Omega}^2$, is representative of the ratio between the thrust load and the centripetal force. For extreme cases, as the parameter approaches zero, the optimal contact angle should approach zero. As the parameter approaches infinity, the optimal contact angle approaches 90°. Finally, when this parameter equals one, the optimal contact angle is 45°.

EXAMPLE

As an example, we use the bearing dimensions used earlier. The closed-form solution, Eqn. (16), arrives at an optimum contact angle $\beta = 22.1°$. Using the exact theory represented by Eqns. (11-13) and much computation, the optimum contact angle $\beta = 25°$ as shown in Fig. 5. The approximate closed-form solution has underestimated the optimum contact angle by about 10% for the conditions examined. However, the difference in the resultant contact force is almost insignificant. Additionally, one notices from Fig. 5 that the selection of a contact angle equal to 50° would result in a maximum load approximately 22% higher. For an actual design, we recommend the use of the closed-form solution to arrive at an approximate contact angle and contact load. If the result were physically realizable, the designer would then employ the exact theory to evaluate additional details of the design. As a matter of interest, the Poplawski and Mauriello bearing's contact angle was equal to 24.5° and this bearing operated at speeds as high as 35,000 rpm.

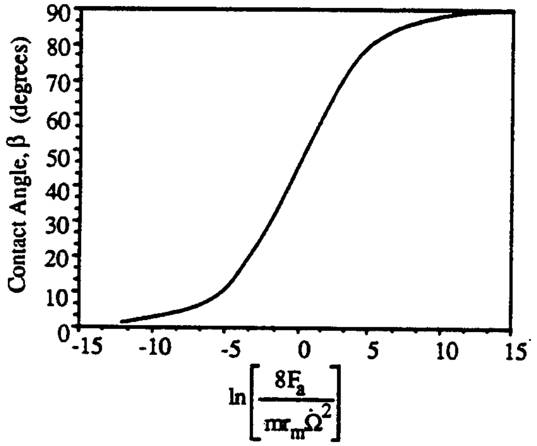


Fig. 7. Optimum contact angle as a function of dimensionless thrust load/axial force ratio.

CONCLUSIONS

This paper has provided a kinematic analysis of ball bearings and has applied the results of this analysis to provide the necessary ancillary equations to allow the solution of the steady-state Newton-Euler equations governing ball motion in thrust ball bearings. Previous researchers have applied numerous extraneous assumptions to solve these equations and thus, their results are not accurate. Additionally, we have presented the no-slip conditions for ball bearings based on the kinematic analysis correcting those that have appeared in the literature for many years. Interestingly eno:.gh, we have shown that the no-slip assumption leads to reasonable results for the low coefficients of friction commonly found in ball bearings.

Through our analysis, we have shown the importance of contact angle on both ball motion and contact force. We have also developed an approximate closed-form solution and have utilized it in determining the optimal contact angle to be used in high-speed ball bearings for minimum contact force. Finally, we have used a simple example to illustrate the usefulness of the theory.

This work, like any other, has some shortcomings. First, we have not considered fluid-film lubrication nor the effects of frictional heating in the contact locations. Additionally, we have assumed Coulomb friction whereas in reality, friction is a complex function of a number of variables. Finally, we have assumed rigid body motion and have not considered the effects of contact deformation. However, we believe that the design method developed can be quite useful in light of the agreement between the developed theory and previously published experimental data. Furthermore, we would hope that the basic theory would be extended in the near future to allow consideration of each of the effects mentioned.

ACKNOWLEDGEMENT

This work has been supported in part by US Army Grant No. DAAL03-90-G-0005.

REFERENCES

Allen, R.K. (1964). Rolling Bearings, 3'rd ed., Sir Isaac Pitman & Sons Ltd., London.

Forsythe, G.E., Malcolm, M.A. and C.B. Moler (1977). Computer Methods for Mathematical Computations, Prentice-Hall, Englewood Cliffs, NJ.

Gupta, P.K. (1975). Transient ball motion and skid in ball bearings. *J. Lub. Tech., Trans. ASME*, 97, 261-269.

Gupta, P.K. (1979). Dynamics of rolling element bearings-part III: ball bearing analysis. *J. Lub. Tech., Trans. ASME*, 101, 312-318.

Hamrock, B.J. and D. Dowson (1981). Ball Bearing Lubrication, The Elastohydrodynamics of Elliptical Contacts, John Wiley & Sons, New York.

Harris, T.A. (1971). Ball motion in thrust-loaded, angular contact bearings with Coulomb friction. *J. Lub. Tech., Trans. ASME*, 93, 32-38.

Harris, T.A. (1984). Rolling Bearing Analysis, 2'nd ed., John Wiley & Sons, New York.

Houghton, P.S. (1976). Ball and Roller Bearings, Applied Science Publishers Ltd.

Jones, A.B. (1959). Ball motion and sliding friction in ball bearings. *J. Basic Engrg, Trans. ASME*, 81, 1-12.

Palmgren, A. (1945). Ball and Roller Bearing Engineering, 2'nd ed., S.H. Burbank & Co., Philadelphia.

Poplawski, J.V. and J.A. Mauriello (1969). Skidding in lightly loaded high-speed ball thrust bearings. ASME #69-LUBS-20.

Q. J. Ge
Assistant Professor,
Department of Mechanical Engineering,
State University of New York
at Stony Brook,
Stony Brook, NY 11794

B. Ravani
Professor,
Department of Mechanical and
Aeronautical Engineering,
University of California at Davis,
Davis, CA 95616

Computer Aided Geometric Design of Motion Interpolants

This paper studies continuous computational geometry of motion and develops a method for Computer Aided Geometric Design (CAGD) of motion interpolants. The approach uses a mapping of spatial kinematics to convert the problem of interpolating displacements to that of interpolating points in the space of the mapping. To facilitate the point interpolation, the previously unorientable mapping space is made orientable. Methods are then developed for designing spline curves in the mapping space with tangent, curvature and torsion continuities. The results have application in computer animation of three-dimensional objects used in computer graphics, computer vision and simulation of mechanical systems.

1 Introduction

This paper deals with computational geometry of motion. Kinematicians and machine theorists have, for many years, studied problems in kinematic geometry of machines and motions (see, for example, Hunt, 1978; Bottema and Roth, 1979; and McCarthy, 1990). Computer scientists and mathematicians have developed geometric algorithms and methods for problems in computer graphics and computer aided design (see, for example, Faux and Pratt, 1979; Farin, 1993). There has been, however, no merging of the two fields. This paper combines and modifies concepts from kinematics and from the field of Computer Aided Geometric Design (CAGD) and provides a solution to one of the most basic problems in computational geometry of motion. The problem, here referred to as *motion interpolation*, is that of finding in-between displacements from a set of given displacements to provide a desired, possibly, smooth, animation of a rigid body for its full range of motion. In computer simulation of large mechanical systems (see, for example, Tsai and Haug, 1991), generation of tightly spaced displacements of a moving body along its trajectory may not be cost effective. Instead, a series of displacements are generated using the dynamic equations of motion and then the in-between displacements can be constructed using appropriate motion interpolants. In robotics, generation of Cartesian trajectories requires interpolation of specified target configurations[1] of the end-effector. In computer vision and computer graphics (see, for example, Turner et al., 1991), reconstructing motion of a moving object or a camera that views the scene also requires interpolating in-between discrete images of the object or the scene.

There are two basic issues in design of motion interpolants. The first one is very basic to kinematics and concerns representation of displacements. The second basic issue is computational geometric in nature and is related to parameterization and piecing of motion interpolants. The traditional approach for computer animation of three-dimensional objects has separated the interpolations of translations and rotations (Reeves, 1981). Although the handling of translation is straightforward, rotation interpolation has proved to be difficult. If orthogonal rotation matrices are used, for example, their interpolation is not in general orthogonal and therefore does not represent a rotation. Early approaches have involved independent interpolation of the Euler angles. This is cumbersome resulting in speeding up or slowing down the motion and depends on temporal relationships. More recently, Shoemake (1985), Duff (1986) and Pletinckx (1989) used quaternions for animating rotations. In this paper we extend these works to achieve second order continuity in motion interpolation. We also use a kinematic mapping of spatial kinematics to develop a general framework for geometric design of complete motion interpolants (including both translations and rotations). The kinematic mapping was introduced by Ravani and Roth (1984) for solving problems in spatial kinematics and mechanisms and is used, in this paper, to transform the problem of designing motion interpolants into that of interpolating points in a special projective three-space called the *image space* of the mapping. In this manner, a piecewise parametric motion is represented by curve segments in the image space. Continuity conditions for piecing motion segments then correspond to geometric continuity conditions for the corresponding curve segments in the image space. The image space, however, is not Euclidean but has an elliptic dual metric. In addition, it is not orientable and as a result point interpolation in this space cannot be defined unambiguously. We modify the image space so that it is orientable and then develop methods for interpolation and approximation in this space with not only tangent (first order) continuity but also higher order continuities of curvature and torsion.

The outline of the paper is as follows. We first provide a brief description for the image space of kinematic mapping. We then introduce an orientable version of the image space by replacing a point with two coincident but oppositely oriented copies. Section 3 presents a linear interpolation in the image

[1] Positions and orientations.

Contributed by the Design Automation Committee for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received Feb. 1991; revised Jan. 1994. Associate Technical Editor: G. A. Gabriele.

space, which corresponds to a screw motion of the animated body. Section 4 describes the generalization of these linear results to cubic interpolation of displacements. Section 5 develops the differential geometry of the image curves in so far as it is needed for developing continuity conditions required in design of piecewise parametric motions. The results of this section are then used in Section 6 to design cubic splines in the image space with tangent or first order continuity and with higher order continuities of curvature and torsion.

2 Displacements as Points in the Image Space

This section first reviews the image space of kinematic mapping (Ravani and Roth, 1984) as a geometric representation of a spatial displacement and then extends the results to the representation of oriented displacements.

2.1 The Classical Image Space. The general displacement of a rigid body in physical space (E^3) is commonly represented by a configuration $M=([R], \mathbf{d})$ of a moving frame with respect to a fixed frame, where $[R]$ is a 3×3 rotation matrix and $\mathbf{d}=(d_x, d_y, d_z)$ is a translation vector. The matrix $[R]$, when written in terms of the Euler parameters of the rotation, $\mathbf{X}=(X_1, X_2, X_3, X_4)$, is given by (Bottema and Roth, 1979):

$$[R]=\frac{1}{\Delta^2}\begin{bmatrix} X_1^2-X_2^2-X_3^2+X_4^2 & 2(X_1X_2-X_3X_4) & 2(X_1X_3+X_2X_4) \\ 2(X_1X_2+X_3X_4) & -X_1^2+X_2^2-X_3^2+X_4^2 & 2(X_2X_3-X_1X_4) \\ 2(X_1X_3-X_2X_4) & 2(X_2X_3+X_1X_4) & -X_1^2-X_2^2+X_3^2+X_4^2 \end{bmatrix}, \quad (1)$$

where $\Delta^2=X_1^2+X_2^2+X_3^2+X_4^2$. The Euler parameters are defined by the rotation angle θ and the direction vector $\mathbf{s}=(s_x, s_y, s_z)$ of the rotation axis as

$$X_1=s_x\sin(\theta/2), \quad X_2=s_y\sin(\theta/2), \\ X_3=s_z\sin(\theta/2), \quad X_4=\cos(\theta/2). \quad (2)$$

Since the motion of a rigid body in E^3 has six degrees of freedom, the geometry for which a displacement as an element is six-dimensional. This has led to the idea of using a set of six independent parameters such as $(X_1/X_4, X_2/X_4, X_3/X_4, d_x, d_y, d_z)$ to represent a displacement as a point in a six-dimensional space. A well-known extension of this idea is the notion of "configuration space" in the context of robot motion planning (Lozano-Perez, 1981; and Canny, 1986).

More elegant representations of a displacement exist where the Euler parameters $\mathbf{X}=(X_1, X_2, X_3, X_4)$ and the translation vector $\mathbf{d}=(d_x, d_y, d_z)$ are considered as defining the direction and location of a line in a space of four dimensions, respectively. This idea was first introduced by Study in 1891 who developed a geometric representation for spatial displacements in terms of what is now referred to as *Study vectors*. Study vectors are a pair of vectors $(\mathbf{X}, -2\mathbf{X}^0)$ that generalize to four dimensions Plücker vectors of line in three-dimensional space, where $\mathbf{X}$ denotes the vector of Euler parameters and $\mathbf{X}^0=(X_1^0, X_2^0, X_3^0, X_4^0)$ is given by

$$\begin{bmatrix} X_1^0 \\ X_2^0 \\ X_3^0 \\ X_4^0 \end{bmatrix}=\frac{1}{2}\begin{bmatrix} 0 & -d_z & d_y & d_x \\ d_z & 0 & -d_x & d_y \\ -d_y & d_x & 0 & d_z \\ -d_x & -d_y & -d_z & 0 \end{bmatrix}\begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ X_4 \end{bmatrix}. \quad (3)$$

The four-dimensional space of lines defined by Study vectors is called the *Soma space*, see Bottema and Roth (1979).

Ravani and Roth (1984) introduced a more compact representation of the Soma space by rewriting the Study vectors as a four-dimensional vector of dual numbers, $\hat{\mathbf{X}}=(\hat{X}_1, \hat{X}_2, \hat{X}_3, \hat{X}_4)$, where $\hat{X}_i=X_i+\epsilon X_i^0$ $(i=1, 2, 3, 4)$. These dual numbers were interpreted as a set of homogeneous coordinates that defines a geometric mapping of displacements into points of a projective space with three dual dimensions. This mathematical space is referred to as the *image space* of spatial kinematics and each point in the space, called the *image point*, represents a spatial displacement. The image space is an unorientable projective space, for the points $\hat{\mathbf{X}}$ and $-\hat{\mathbf{X}}$ are considered to be identical.

The parameters X_i, X_i^0 $(i=1, 2, 3, 4)$ are not independent but must satisfy the fundamental relation:

$$X_1X_1^0+X_2X_2^0+X_3X_3^0+X_4X_4^0=0. \quad (4)$$

An image point with coordinates satisfying (4) is said to have *normalized coordinates*. A general set of four dual numbers to be used as homogeneous coordinates of an image point may not satisfy (4) and therefore must be normalized. A simple normalizing procedure is provided in Appendix A such that not only (4) is satisfied but the real part has unit length as well. For the rest of the paper, it is assumed that the coordinates of an image point are normalized and have unit length unless explicitly stated otherwise.

The image space of Ravani and Roth has been shown to be a useful tool for analysis, synthesis, and classification of spatial motions (Ravani and Roth, 1984; Ge and McCarthy, 1991a). It has also been used for obtaining an explicit representation of joint-space obstacles for robot motion planning (Ge and McCarthy, 1990) and for characterizing functional constraints in a manufactured assembly (Ge and McCarthy, 1991b). The image space in its original form, however, is not quite suitable for computer aided geometric design of motion interpolants, for it is not orientable. This means that there is no consistent way of defining "left" or "right" handedness of a coordinate tetrahedron for the space. It also means that there does not exist the notion of "betweenness" and as a result one cannot define a line-segment uniquely. These are serious problems to proper interpolation of a point set. In the next section, we present an orientable version of this image space to resolve these problems.

2.2 The Orientable Image Space. We start the introduction of the orientable image space by revisiting the rotation matrix $[R]$. Given $[R]$, the rotation angle θ and rotation axis $\mathbf{s}$ may be extracted from its elements, see Bottema and Roth (1979), Paul (1981) or Shoemake (1985). The result is, however, not unique. For a given $[R]$, there corresponds a pair of oppositely oriented rotations, one is about the axis $\mathbf{s}$ with rotation angle θ and is termed a "forward" rotation; the other is about the axis $-\mathbf{s}$ with rotation angle $2\pi-\theta$ and is termed a "backward" rotation.

In view of (2), we conclude that the two oppositely oriented rotations can be represented by the *signed* Euler parameters $\mathbf{X}=(X_1, X_2, X_3, X_4)$ and $-\mathbf{X}=(-X_1, -X_2, -X_3, -X_4)$, respectively. It follows that a general configuration $M=([R], \mathbf{d})$ may be obtained by either of two oppositely oriented displacements: one is a "forward" displacement and consists of a rotation $\mathbf{X}$ and a translation $\mathbf{d}$; the other is a "backward" displacement and consists of a rotation $-\mathbf{X}$ and a translation $\mathbf{d}$. In view of (3), these two oppositely oriented displacements can be represented by two sets of image space coordinates, $\hat{\mathbf{X}}$ and $-\hat{\mathbf{X}}$, respectively. This leads naturally to the notion of *signed* homogeneous coordinates for the image space. Instead of identifying $\hat{\mathbf{X}}$ and $-\hat{\mathbf{X}}$, we treat them as defining two distinct points which occupy the same position in the image space but with "opposite orientations," namely "forward" and "backward." To borrow a term from spherical geometry, we may call these two points *antipodal points*. Thus a general displacement corresponds to not *one* image point but *two* antipodal points. The image space consists of such pairs of coincident but oppositely oriented points is here referred to as the *orientable image space* and is denoted as Σ. In general, the signed homogeneous coordinates $\hat{\mathbf{X}}$ and $\hat{w}\hat{\mathbf{X}}$, where $\hat{w} = w + \epsilon w^0$ is a nonpure dual number, are considered to defined the same point if $w > 0$; they define two antipodal points if $w < 0$.

Specialists in projective geometry will notice that geometry of the orientable image space Σ is equivalent to geometry of a unit hyperspace (denoted by H^3) in a space of four dual dimensions, with oriented points and oriented lines in Σ corresponding to points and oriented great circles on H^3, respectively. The classical image space is topologically different from Σ. Geometry of the classical image space is identical to unoriented spherical geometry of three dual dimensions, for one point in the space corresponds to two diametrically opposite points on an unorientable unit dual hyperspace.

The distance between two points $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$ in Σ is in general a dual number and is defined analogous to the angular distance in elliptic geometry of real three-space. The dual-number distance is the dual angle $\hat{\phi} = \phi + \epsilon h$ obtained from:

$$\cos \hat{\phi} = \frac{\hat{\mathbf{X}} \cdot \hat{\mathbf{Y}}}{(\hat{\mathbf{X}} \cdot \hat{\mathbf{X}})^{1/2} (\hat{\mathbf{Y}} \cdot \hat{\mathbf{Y}})^{1/2}}, \tag{5}$$

where $\hat{\mathbf{X}} \cdot \hat{\mathbf{Y}} = \hat{X}_1 \hat{Y}_1 + \hat{X}_2 \hat{Y}_2 + \hat{X}_3 \hat{Y}_3 + \hat{X}_4 \hat{Y}_4$ is the standard scalar product. The dual angle $\hat{\phi}$ is uniquely defined, provided that the real angle ϕ is restricted to the range $[0, \pi]$ (see the Appendix A). Kinematically, the dual angle specifies the magnitude of a relative screw displacement between the displacements represented by $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$: 2ϕ is the angle of rotation about the screw axis of the relative displacement and $2h$ is the distance of translation along the same axis. When two image points satisfy $\hat{\mathbf{X}} \cdot \hat{\mathbf{Y}} = 0$, the distance between them is $\pi/2$ and they are said to be a pair of *polar* points. The corresponding displacements are related by a half turn (a pure rotation with angle π).

2.3 Curves and Planes in the Image Space. Since the image space Σ has three *dual* dimensions, a general curve in Σ is a one-dual-dimensional curve or a *twofold curve*. A general plane in Σ has two dual dimensions and is here referred to as a *twofold plane*. A twofold curve is the mapping of a two-degree-of-freedom motion. A one-degree-of-freedom motion maps into a special curve in Σ called a *unifold curve*. A twofold plane is the mapping of a four-degree-of-freedom motion.

The simplest twofold curve in Σ is an oriented twofold straight line. Given two oriented points $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$, the twofold line joining from $\hat{\mathbf{X}}$ to $\hat{\mathbf{Y}}$ is unique and can be represented by a directed bivector $\hat{\mathbf{X}} \wedge \hat{\mathbf{Y}}$, where the symbol "$\wedge$" denotes the vector wedge product which generalizes to higher dimensions the vector cross product of three-dimensional vector algebra, see Flanders (1963) and McCarthy (1990). The six components of $\hat{\mathbf{X}} \wedge \hat{\mathbf{Y}}$ are the signed Plücker coordinates of the line. The line joining from $\hat{\mathbf{Y}}$ to $\hat{\mathbf{X}}$ is given by the bivector $\hat{\mathbf{Y}} \wedge \hat{\mathbf{X}}$ and has the opposite sense direction to the line $\hat{\mathbf{X}} \wedge \hat{\mathbf{Y}}$. Kinematically, a twofold line $\hat{\mathbf{X}} \wedge \hat{\mathbf{Y}}$ is the mapping of a two-degree-of-freedom screw motion that contains the displacements represented by $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$. The screw motion consists of two independent simple motions about the fixed screw axis, a rotation and a translation. If the translation is made dependent on the rotation, then the resulting motion becomes a one-degree-of-freedom screw motion which maps into a unifold line in Σ.

A general twofold plane is the mapping of a four-degree-of-freedom line-symmetric motion. To see this, we consider a plane P defined by three oriented points $\hat{\mathbf{X}}$, $\hat{\mathbf{Y}}$ and $\hat{\mathbf{Z}}$. It can be represented by the directed trivector $\hat{\mathbf{X}} \wedge \hat{\mathbf{Y}} \wedge \hat{\mathbf{Z}}$. Since the Σ space is an oriented projective dual three-space, a formal duality exists between oriented points and oriented twofold planes. It follows that the four components, $\hat{\mathbf{F}} = (\hat{F}_1, \hat{F}_2, \hat{F}_3, \hat{F}_4)$, of the trivector $\hat{\mathbf{X}} \wedge \hat{\mathbf{Y}} \wedge \hat{\mathbf{Z}}$ define the pole of the plane P. Any point of the plane, say $\hat{\mathbf{G}}$, satisfies the relation $\hat{\mathbf{F}} \cdot \hat{\mathbf{G}} = 0$ and thus the two displacements represented by $\hat{\mathbf{F}}$ and $\hat{\mathbf{G}}$ are related by a half-turn. Therefore the twofold plane $\hat{\mathbf{X}} \wedge \hat{\mathbf{Y}} \wedge \hat{\mathbf{Z}}$ represents a four-degree-of-freedom line-symmetric motion that contains the displacements $\hat{\mathbf{X}}$, $\hat{\mathbf{Y}}$ and $\hat{\mathbf{Z}}$. A planar twofold curve is the mapping of a two-degree-of-freedom line-symmetric motion and a planar unifold curve is the mapping of a one-degree-of-freedom line-symmetric motion.

The point coordinates of a unifold image curve depend on a real parameter t. They are denoted by $\hat{\mathbf{W}}(t) = \hat{w}(t)\hat{\mathbf{X}}(t)$ where $\hat{\mathbf{W}}(t) = (\hat{W}_1(t), \hat{W}_2(t), \hat{W}_3(t), \hat{W}_4(t))$ represents the nonnormalized coordinates, $\hat{\mathbf{X}}(t) = (\hat{X}_1(t), \hat{X}_2(t), \hat{X}_3(t), \hat{X}_4(t))$ represents the unit-normalized coordinates, and $\hat{w}(t)$ denotes the normalizing factor (see Appendix B). If the coordinates $\hat{W}_i(t)$ $(i = 1, 2, 3, 4)$ are specified by linear, quadratic, or cubic functions of t, the corresponding unifold image curve is a straight line, a conic, or a cubic, respectively. The resulting motion is termed *linear, conic,* or *cubic motions*, respectively. Physically, a linear motion is a screw motion and a conic motion is a line-symmetric motion.

The velocity distribution of a spatial motion is determined by the dual velocity $\hat{\mathbf{V}}$, which can be expressed in terms of the point coordinates of an image curve $\hat{\mathbf{X}}(t)$ and its first derivative $\dot{\hat{\mathbf{X}}}(t)$ using quaternion algebra. Let $(\mathbf{i}, \mathbf{j}, \mathbf{k}, 1)$ denote the quaternion basis. Then the dual velocity is given by the quaternion product:

$$\hat{\mathbf{V}} = 2\dot{\hat{\mathbf{X}}}\hat{\mathbf{X}}^{-1}, \tag{6}$$

where $\hat{\mathbf{X}} = \hat{X}_1\mathbf{i} + \hat{X}_2\mathbf{j} + \hat{X}_3\mathbf{k} + \hat{X}_4$ is a unit quaternion and $\hat{\mathbf{X}}^{-1} = -\hat{X}_1\mathbf{i} - \hat{X}_2\mathbf{j} - \hat{X}_3\mathbf{k} + \hat{X}_4$ is its inverse. This result is obtained from dualization of a similar formula for spherical kinematics presented in Bottema and Roth (1979).

3 Linear Interpolation of Two Displacements

Two displacements D_0 and D_1 of a rigid body correspond to two configurations M_0 and M_1 of a moving frame with respect to a fixed frame, which in turn correspond to two pairs of oppositely oriented points $\pm\hat{\mathbf{X}}_0$ and $\pm\hat{\mathbf{X}}_1$, respectively, in the orientable image space Σ. In this paper we choose orientations (or signs) of the image points $\hat{\mathbf{X}}_0$ and $\hat{\mathbf{X}}_1$ such that $\mathbf{X}_0 \cdot \mathbf{X}_1 \geq 0$. Geometrically, this means that the angular distance ϕ between $\hat{\mathbf{X}}_0$ and $\hat{\mathbf{X}}_1$ is in the range $[0, \pi/2]$. Two such oriented points are referred to as *similarly oriented points*. In this way, the problem of linearly interpolating two displacements D_0 and D_1 becomes that of finding a unifold line-segment that joins two similarly oriented points $\hat{\mathbf{X}}_0$ and $\hat{\mathbf{X}}_1$. Let the unifold line-segment be given by

$$\hat{\mathbf{W}}(t) = \hat{w}(t)\hat{\mathbf{X}}(t) = \hat{\mathbf{a}}_0 + \hat{\mathbf{a}}_1 t \tag{7}$$

where $\hat{\mathbf{W}}(t)$, $\hat{\mathbf{X}}(t)$ denote the general and unit-normalized coordinates, respectively, and $\hat{w}(t)$ denotes the normalizing factor. We seek to determine the coefficient vectors $\hat{\mathbf{a}}_i$ $(i = 0, 1)$ such that

$$\hat{\mathbf{X}}(0) = \hat{\mathbf{X}}_0, \quad \hat{\mathbf{X}}(1) = \hat{\mathbf{X}}_1. \tag{8}$$

The result is the following unifold linear interpolant:

$$\hat{\mathbf{W}}(t) = \hat{w}(t)\hat{\mathbf{X}}(t) = (1 - t)\hat{w}_0\hat{\mathbf{X}}_0 + t\hat{w}_1\hat{\mathbf{X}}_1, \quad t = [0, 1] \tag{9}$$

where $\hat{w}_0 = \hat{w}(0)$ and $\hat{w}_1 = \hat{w}(1)$ can be arbitrary dual numbers. In practice we require $\hat{w}_0$ and $\hat{w}_1$ to have positive real parts so that all displacements of the linear motion are similarly oriented. The normalizing factor $\hat{w}(t)$ is obtained from (9) as

$$\hat{w}(t) = [(1 - t)^2\hat{w}_0^2 + t^2\hat{w}_1^2 + 2t(1 - t)\hat{w}_0\hat{w}_1(\hat{\mathbf{X}}_0 \cdot \hat{\mathbf{X}}_1)]^{1/2}. \tag{10}$$

The unifold line-segment (9) is the image curve of a screw motion that interpolates through the configurations[2] M_0 and M_1.

To see the effect of the dual-number factors $\hat{w}_0$ and $\hat{w}_1$ on

[2] For the sake of convenience, we sometimes use the words "displacements" and "configurations" interchangeably in this paper.

the velocity distribution of the animated body, we use (6) to obtain the dual velocity:

$$\hat{\mathbf{V}} = \frac{\hat{\mathbf{X}}_1\hat{\mathbf{X}}_0^{-1} - \hat{\mathbf{X}}_0\hat{\mathbf{X}}_1^{-1}}{(1-t)^2(\hat{w}_0/\hat{w}_1) + t^2(\hat{w}_1/\hat{w}_0) + 2t(1-t)(\hat{\mathbf{X}}_0\cdot\hat{\mathbf{X}}_1)} \qquad (11)$$

where the term $(\hat{\mathbf{X}}_1\hat{\mathbf{X}}_0^{-1} - \hat{\mathbf{X}}_1\hat{\mathbf{X}}_1^{-1})$ is a vector quaternion. Equation (11) shows that the screw axis is fixed (as anticipated for a screw motion) but in general both the angular velocity and the pitch of the screw motion are functions of the parameter t and the ratio $\hat{w}_0/\hat{w}_1$. The choice of $\hat{w}_0/\hat{w}_1$ influences the angular velocity and the pitch of the screw motion but it is impossible to choose the ratio such that the pitch or angular velocity remains constant throughout the screw motion.

4 Cubic Interpolation of Displacements

In the preceding section, we dealt with the problem of two-configuration interpolation using a linear motion interpolant. In this section, we develop cubic parametric motions that interpolate through not only two given configurations but two instantaneous dual velocities as well. This is referred as as *first order motion interpolation*.

4.1 First Order End Conditions. In the image space Σ, the two configurations M_0, M_1 are represented by two similarly oriented image points $\hat{\mathbf{X}}_0, \hat{\mathbf{X}}_1$, respectively. Let the dual velocity at M_i $(i=0, 1)$ be $\hat{\mathbf{V}}_i = 2\hat{\Omega}_i\hat{\mathbf{v}}_i$, where $2\hat{\Omega}_i = 2(\Omega_i + \epsilon\Omega_i^0)$ $(\Omega_i \geq 0, \Omega_i{}^0 \geq 0)$ is the dual speed and $\hat{\mathbf{v}}_i$ is the unit dual vector representing the instantaneous screw axis. We define the "tangent" of the desired interpolating motion at M_i $(i=0, 1)$ in terms of the derivative of the image curve $\hat{\mathbf{X}}(t)$ of the motion. In view of (6), the derivative of $\hat{\mathbf{X}}(t)$ at M_i $(i=0, 1)$ is given by the quaternion product:

$$\dot{\hat{\mathbf{X}}}_i = (1/2)\hat{\mathbf{V}}_i\hat{\mathbf{X}}_i = \hat{\Omega}_i\hat{\mathbf{T}}_i$$

where $\hat{\mathbf{T}}_i = \hat{\mathbf{v}}_i\hat{\mathbf{X}}_i$. It is not difficult to show that $\hat{\mathbf{T}}_i$ satisfies $\hat{\mathbf{T}}_i\cdot\hat{\mathbf{T}}_i = 1$ and $\hat{\mathbf{T}}_i\cdot\hat{\mathbf{X}}_i = 0$. Therefore $\dot{\hat{\mathbf{X}}}_i$ is a polar point of $\hat{\mathbf{X}}_i$ and can be interpreted as a "tangent vector" at $\hat{\mathbf{X}}_i$ with $\hat{\mathbf{T}}_i$ being a "unit tangent vector" and $\hat{\Omega}_i$ being its magnitude.

The problem of finding a motion that interpolates through configurations M_0, M_1 as well as the dual velocities $\hat{\mathbf{V}}_0$, $\hat{\mathbf{V}}_1$ becomes that of finding a unifold curve $\hat{\mathbf{X}}(t)$ in Σ such that the following interpolation conditions are satisfied:

$$\begin{aligned} \hat{\mathbf{X}}(0) &= \hat{\mathbf{X}}_0, & \dot{\hat{\mathbf{X}}}(0) &= \hat{\Omega}_0\hat{\mathbf{T}}_0, \\ \hat{\mathbf{X}}(1) &= \hat{\mathbf{X}}_1, & \dot{\hat{\mathbf{X}}}(1) &= \hat{\Omega}_1\hat{\mathbf{T}}_1. \end{aligned} \qquad (12)$$

The conditions (12) have similarities to Hermite interpolation conditions in CAGD and are therefore termed *Hermite type motion interpolation conditions*.

4.2 Hermite Type Interpolation. A unifold cubic parametric curve in the image space Σ is of the form

$$\hat{\mathbf{W}}(t) = \hat{w}(t)\hat{\mathbf{X}}(t) = \hat{\mathbf{a}}_0 + \hat{\mathbf{a}}_1 t + \hat{\mathbf{a}}_2 t^2 + \hat{\mathbf{a}}_3 t^3, \qquad (13)$$

where $\hat{\mathbf{W}}(t)$ denotes general coordinates of a point on the cubic, $\hat{\mathbf{X}}(t)$ denotes its normalized coordinates, and $\hat{w}(t)$ denotes the normalizing factor. We seek to determine the vectors of coefficients $\hat{\mathbf{a}}_i$ $(i=0, 1, 2, 3)$ such that the conditions (12) are satisfied. Since the first derivative $\dot{\hat{\mathbf{W}}} = \dot{\hat{w}}\hat{\mathbf{X}} + \hat{w}\dot{\hat{\mathbf{X}}}$, the Hermite conditions (12) are equivalent to

$$\begin{aligned} \hat{\mathbf{W}}(0) &= \hat{w}(0)\hat{\mathbf{X}}_0, & \hat{\mathbf{W}}(1) &= \hat{w}(1)\hat{\mathbf{X}}_1, \\ \dot{\hat{\mathbf{W}}}(0) &= \dot{\hat{w}}(0)\hat{\mathbf{X}}_0 + \hat{w}(0)\hat{\Omega}_0\hat{\mathbf{T}}_0, & \dot{\hat{\mathbf{W}}}(1) &= \dot{\hat{w}}(1)\hat{\mathbf{X}}_1 + \hat{w}(1)\hat{\Omega}_1\hat{\mathbf{T}}_1. \end{aligned} \qquad (14)$$

From (13) and (14), the coefficient vector $\hat{\mathbf{a}}_i$ $(i=0, 1, 2, 3)$ can be solved to yield the following cubic interpolant:

$$\hat{\mathbf{W}}(t) = [1\ t\ t^2\ t^3][M_h]\begin{bmatrix} \hat{w}(0)\hat{\mathbf{X}}_0 \\ \dot{\hat{w}}(0)\hat{\mathbf{X}}_0 + \hat{w}(0)\hat{\Omega}_0\hat{\mathbf{T}}_0 \\ \dot{\hat{w}}(1)\hat{\mathbf{X}}_1 + \hat{w}(1)\hat{\Omega}_1\hat{\mathbf{T}}_1 \\ \hat{w}(1)\hat{\mathbf{X}}_1 \end{bmatrix} \qquad (15)$$

where $[M_h]$ is the standard Hermite basis matrix (see Faux and Pratt, 1979):

$$[M_h] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -3 & -2 & -1 & 3 \\ 2 & 1 & 1 & -2 \end{bmatrix}.$$

The interpolant (15) is here referred to as a *Hermite type cubic motion interpolant*. For a given set of Hermite conditions (12), the corresponding interpolant is not unique, because the dual numbers $\hat{w}(0)$, $\hat{w}(1)$, $\dot{\hat{w}}(0)$, $\dot{\hat{w}}(1)$ can be arbitrary chosen. In practice we require $\hat{w}(0)$ and $\hat{w}(1)$ to have positive real parts.

The Hermite type interpolant (15) can be rewritten as

$$\hat{\mathbf{W}}(t) = \hat{w}(t)\hat{\mathbf{X}}(t) = [1\ t\ t^2 t^3][M_H]\begin{bmatrix} \hat{w}(0)\hat{\mathbf{X}}_0 \\ \hat{w}(0)\hat{\Omega}_0\hat{\mathbf{T}}_0 \\ \hat{w}(1)\hat{\Omega}_1\hat{\mathbf{T}}_1 \\ \hat{w}(1)\hat{\mathbf{X}}_1 \end{bmatrix}. \qquad (16)$$

where the matrix $[M_H]$, called the *generalized Hermite basis matrix*, is given by

$$[M_H] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \hat{\gamma}_0 & 1 & 0 & 0 \\ -3-2\hat{\gamma}_0 & -2 & -1 & 3-\hat{\gamma}_1 \\ 2+\hat{\gamma}_0 & 1 & 1 & -2+\hat{\gamma}_1 \end{bmatrix} \qquad (17)$$

where $\hat{\gamma}_0 = \dot{\hat{w}}(0)/\hat{w}(0)$ and $\hat{\gamma}_1 = \dot{\hat{w}}(1)/\hat{w}(1)$ are arbitrary dual numbers.

4.3 Bernstein-Bézier Type Interpolation. Let $\hat{\mathbf{b}}_0 = \hat{\mathbf{X}}_0$, $\hat{\mathbf{b}}_3 = \hat{\mathbf{X}}_1$. Then the Hermite interpolation (16) can be put into the following form:

$$\hat{\mathbf{W}}(t) = \hat{w}(t)\hat{\mathbf{X}}(t) = [1\ t\ t^2\ t^3][M_B]\begin{bmatrix} \hat{w}(0)\hat{\mathbf{b}}_0 \\ \hat{w}(0)\hat{\Omega}_0\hat{\mathbf{b}}_1' \\ \hat{w}(1)\hat{\Omega}_1\hat{\mathbf{b}}_2' \\ \hat{w}(1)\hat{\mathbf{b}}_3 \end{bmatrix} \qquad (18)$$

where the matrix $[M_B]$ is given by

$$[M_B] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3+\hat{\gamma}_0 & 3 & 0 & 0 \\ 3-2\hat{\gamma}_0 & -6 & 3 & -\hat{\gamma}_1 \\ -1+\hat{\gamma}_0 & 3 & -3 & 1+\hat{\gamma}_1 \end{bmatrix}$$

and $\hat{\mathbf{b}}_1' = \hat{\mathbf{X}}_0 + \hat{\mathbf{T}}_0/3$, $\hat{\mathbf{b}}_2' = \hat{\mathbf{X}}_1 - \hat{\mathbf{T}}_1/3$ are points on the tangent lines $\hat{\mathbf{X}}_0 \wedge \hat{\mathbf{T}}_0$, $\hat{\mathbf{X}}_1 \wedge \hat{\mathbf{T}}_1$, respectively. The matrix $[M_B]$ becomes the standard Bernstein-Bézier basis matrix when $\hat{\gamma}_0 = \hat{\gamma}_0 = 0$:

$$[M_b] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}. \qquad (19)$$

Therefore, the matrix $[M_B]$ is termed *generalized cubic Bernstein-Bézier basis matrix* and the interpolant (18) is termed *generalized Bernstein-Bézier type motion interpolant*.

The generalized Bernstein-Bézier interpolant can be rewritten as:

$$\hat{\mathbf{W}}(t) = \hat{w}(t)\hat{\mathbf{X}}(t) = [1\ t\ t^2\ t^3][M_b]\begin{bmatrix} \hat{w}_0\hat{\mathbf{b}}_0 \\ \hat{w}_1\hat{\mathbf{b}}_1 \\ \hat{w}_1\hat{\mathbf{b}}_2 \\ \hat{w}_3\hat{\mathbf{b}}_3 \end{bmatrix} \qquad (20)$$

where $[M_b]$ is given by (19) and

$$\hat{w}_0=\hat{w}(0), \qquad \hat{w}_1=\hat{w}(0)\hat{\Omega}_0\hat{g}_0,$$
$$\hat{w}_2=\hat{w}(1)\hat{\Omega}_1\hat{g}_1, \qquad \hat{w}_3=\hat{w}(1),$$
$$\hat{\mathbf{b}}_1=[\hat{\mathbf{b}}_1'+(\hat{\gamma}_0/3)\hat{\mathbf{b}}_0]/\hat{g}_0, \quad \hat{g}_0=|\hat{\mathbf{b}}_1'+(\hat{\gamma}_0/3)\hat{\mathbf{b}}_0|,$$
$$\hat{\mathbf{b}}_2=[\hat{\mathbf{b}}_2'-(\hat{\gamma}_1/3)\hat{\mathbf{b}}_3]/\hat{g}_0, \quad \hat{g}_1=|\hat{\mathbf{b}}_2'+(\hat{\gamma}_0/3)\hat{\mathbf{b}}_3|.$$

Note that $\hat{\mathbf{b}}_1$, $\hat{\mathbf{b}}_2$ are arbitrary points on the tangents $\hat{\mathbf{b}}_0\wedge\hat{\mathbf{b}}_1'$, $\hat{\mathbf{b}}_2'\wedge\hat{\mathbf{b}}_3$, respectively. Note also that the Bézier interpolant (20) is invariant to change of both fixed and moving coordinate frames.

5 Differential Properties and Smoothness Conditions

In this section, we first summarize, in the language of orientable image space Σ, the differential properties of image curves, which were developed by McCarthy and Ravani (1986) for the case of the classical image space. We then apply the results to the problem of smooth joining of two motions segments.

5.1 Differential Properties. The differential geometry of curves in Σ is developed analogous to the differential geometry of curves in Euclidean three-space. Let $\hat{\mathbf{X}}(t)$ denote the signed unit-normalized coordinates of an image curve. The fundamental result is the Frenet equations which characterize the curve in terms of the differential motion of a special tetrahedron with oriented points $\hat{\mathbf{T}}$, $\hat{\mathbf{N}}$, $\hat{\mathbf{B}}$, $\hat{\mathbf{X}}$ as its vertices:

$$\begin{aligned} d\hat{\mathbf{T}}/d\hat{s}&=\hat{\kappa}\hat{\mathbf{N}}-\hat{\mathbf{X}},\\ d\hat{\mathbf{N}}/d\hat{s}&=-\hat{\kappa}\hat{\mathbf{T}}+\hat{\tau}\hat{\mathbf{B}},\\ d\hat{\mathbf{B}}/d\hat{s}&=-\hat{\tau}\hat{\mathbf{N}},\\ d\hat{\mathbf{X}}/d\hat{s}&=\hat{\mathbf{T}}. \end{aligned} \tag{21}$$

The dual numbers $\hat{s}$, $\hat{\kappa}$ and $\hat{\tau}$ are the dual arc-length, curvature and torsion of $\hat{\mathbf{X}}(t)$, respectively. The vertex $\hat{\mathbf{T}}$ is defined in the same way as the unit tangent vector of a Euclidean curve:

$$\hat{\mathbf{T}}=\frac{d\hat{\mathbf{X}}}{d\hat{s}}=\frac{d\hat{\mathbf{X}}/dt}{|d\hat{\mathbf{X}}/dt|} \tag{22}$$

The vertices $\hat{\mathbf{N}}$ and $\hat{\mathbf{B}}$ are defined in such a way that the four points $\hat{\mathbf{X}}$, $\hat{\mathbf{T}}$, $\hat{\mathbf{N}}$, $\hat{\mathbf{B}}$ form a self-polar tetrahedron called the *Frenet tetrahedron*. The manner of defining $\hat{\mathbf{N}}$, $\hat{\mathbf{B}}$ is similar to that of the normal and binormal vectors of a curve in Euclidean three-space, see DoCarmo (1976) and McCarthy (1987).

The tangent line-segment of $\hat{\mathbf{X}}(t)$ is given by the following bivector:

$$\hat{\mathbf{X}}\wedge\dot{\hat{\mathbf{X}}}=\hat{v}\hat{\mathbf{X}}\wedge\hat{\mathbf{T}}, \tag{23}$$

where $\hat{v}=|d\hat{\mathbf{X}}/dt|$ and $\hat{\mathbf{T}}$ is the polar point defined by (22). The curvature function and the osculating plane are given by the following 3-vector:

$$\hat{\mathbf{X}}\wedge\dot{\hat{\mathbf{X}}}\wedge\ddot{\hat{\mathbf{X}}}=\hat{v}^3\hat{\kappa}\hat{\mathbf{X}}\wedge\hat{\mathbf{T}}\wedge\hat{\mathbf{N}}. \tag{24}$$

The torsion property is given by

$$\hat{\mathbf{X}}\wedge\dot{\hat{\mathbf{X}}}\wedge\ddot{\hat{\mathbf{X}}}\wedge\dddot{\hat{\mathbf{X}}}=\hat{v}^6\hat{\kappa}^2\hat{\tau}\hat{\mathbf{X}}\wedge\hat{\mathbf{T}}\wedge\hat{\mathbf{N}}\wedge\hat{\mathbf{B}}. \tag{25}$$

The wedge products of the above derivatives can be expressed in terms of the nonnormalized coordinates $\hat{\mathbf{W}}(t)=\hat{w}\hat{\mathbf{X}}(t)$:

$$\begin{aligned} \mathbf{X}\wedge\dot{\hat{\mathbf{X}}}&=(1/\hat{w}^2)\hat{\mathbf{W}}\wedge\dot{\hat{\mathbf{W}}},\\ \hat{\mathbf{X}}\wedge\dot{\hat{\mathbf{X}}}\wedge\ddot{\hat{\mathbf{X}}}&=(1/\hat{w}^3)\hat{\mathbf{W}}\wedge\dot{\hat{\mathbf{W}}}\wedge\ddot{\hat{\mathbf{W}}},\\ \hat{\mathbf{X}}\wedge\dot{\hat{\mathbf{X}}}\wedge\ddot{\hat{\mathbf{X}}}\wedge\dddot{\hat{\mathbf{X}}}&=(1/\hat{w}^4)\hat{\mathbf{W}}\wedge\dot{\hat{\mathbf{W}}}\wedge\ddot{\hat{\mathbf{W}}}\wedge\dddot{\hat{\mathbf{W}}}. \end{aligned}$$

5.2 Smoothness Conditions. Consider two cubic curve segments $\hat{\mathbf{X}}_-$ and $\hat{\mathbf{X}}_+$ that are smoothly joined together. We may think of each segment as existing by itself, with local parameters t_-, t_+, respectively, defined over the interval [0, 1]. We may also think of the two as two segments of one composite curve, with a global parameter u defined over the interval $[u_-, u_+]$. The "left" segment $\hat{\mathbf{X}}_-$ is defined over $[u_-, u_0]$, while the "right" segment $\hat{\mathbf{X}}_+$ is defined over $[u_0, u_+]$. Then the local parameters t_-, t_+ are related to the global parameter u by

$$t_-=\frac{u-u_-}{u_0-u_-}=\frac{u-u_-}{\Delta_-}, \quad t_+=\frac{u-u_0}{u_+-u_0}=\frac{u-u_0}{\Delta_+}.$$

The differential operators for local and global parameters t and u are related by

$$\frac{d}{du}=\frac{1}{\Delta}\frac{d}{dt}, \quad \frac{d^2}{du^2}=\frac{1}{\Delta^2}\frac{d^2}{dt^2}, \quad \frac{d^3}{du^3}=\frac{1}{\Delta^3}\frac{d^3}{dt^3}.$$

In what follows, we present smoothness conditions of a composite cubic Hermite image curve in terms of the continuities of the tangent vector, curvature and torsion. We always assume that the two given segments are continuous at the junction point, i.e., $\hat{\mathbf{X}}_-(1)=\hat{\mathbf{X}}_+(0)$.

The continuity of the tangent vector at $u=u_0$ of the two curve segments $\hat{\mathbf{X}}_-$, $\hat{\mathbf{X}}_+$ requires that the first derivative with respect to a global parameter u is continuous at the junction point:

$$\left.\frac{d\hat{\mathbf{X}}(u)}{du}\right|_{u=u_0}=\frac{1}{\Delta_-}\left.\frac{d\hat{\mathbf{X}}_-(t_-)}{dt_-}\right|_{t_-=1}=\frac{1}{\Delta_+}\left.\frac{\hat{\mathbf{X}}_+(t_+)}{dt_+}\right|_{t_+=0} \tag{26}$$

This is equivalent to

$$\hat{\mathbf{X}}_-(1)\wedge\dot{\hat{\mathbf{X}}}_-(1)=\hat{\alpha}\hat{\mathbf{X}}_+(0)\wedge\dot{\hat{\mathbf{X}}}_+(0) \tag{27}$$

where $\hat{\alpha}=\Delta_-/\Delta_+$. If $\hat{\alpha}$ is any other dual number with positive real part, then (27) represents the continuity of unit tangent vectors.

For curvature continuity, in addition to (27), the curvature 3-vector is required to be continuous:

$$\hat{\mathbf{X}}_-(1)\wedge\dot{\hat{\mathbf{X}}}_-(1)\wedge\ddot{\hat{\mathbf{X}}}_-(1)=\hat{\alpha}^3\hat{\mathbf{X}}_+(0)\wedge\dot{\hat{\mathbf{X}}}_+(0)\wedge\ddot{\hat{\mathbf{X}}}_+(0). \tag{28}$$

For torsion continuity, in addition to (27) and (28), the torsion 4-vector is also required to be continuous:

$$\begin{aligned} &\hat{\mathbf{X}}_-(1)\wedge\dot{\hat{\mathbf{X}}}_-(1)\wedge\ddot{\hat{\mathbf{X}}}_-(1)\wedge\dddot{\hat{\mathbf{X}}}_-(1)\\ &\qquad=\hat{\alpha}^6\hat{\mathbf{X}}_+(0)\wedge\dot{\hat{\mathbf{X}}}_+(0)\wedge\ddot{\hat{\mathbf{X}}}_+(0)\wedge\dddot{\hat{\mathbf{X}}}_+(0). \end{aligned} \tag{29}$$

5.3 Cubic Parametric Curves. Given an image curve associated with a cubic interpolation of four distinct displacements, we can obtain the differential properties in terms of control points and weight factors and then use these properties to characterize the smoothness of the resulting motion. In what follows we only provide these properties at end points of the cubic Hermite type interpolation given by (16).

The tangent line-segments at the end points of the Hermite type image curve (16) are given by

$$\hat{\mathbf{X}}(0)\wedge\dot{\hat{\mathbf{X}}}(0)=\hat{\Omega}_0\hat{\mathbf{X}}_0\wedge\hat{\mathbf{T}}_0, \tag{30}$$

$$\hat{\mathbf{X}}(1)\wedge\dot{\hat{\mathbf{X}}}(1)=\hat{\Omega}_1\hat{\mathbf{X}}_1\wedge\hat{\mathbf{T}}_1. \tag{31}$$

Let $\hat{w}_0=\hat{w}(0)$ and $\hat{w}_1=\hat{w}(1)$. The curvature properties at the end points are given by

$$\hat{\mathbf{X}}(0)\wedge\dot{\hat{\mathbf{X}}}(0)\wedge\ddot{\hat{\mathbf{X}}}(0)=6\hat{\Omega}_0\frac{\hat{w}_1}{\hat{w}_0}\hat{\mathbf{X}}_0\wedge\hat{\mathbf{T}}_0\wedge\left[\left(1-\frac{\hat{\gamma}_1}{3}\right)\hat{\mathbf{X}}_1-\frac{1}{3}\hat{\Omega}_1\hat{\mathbf{T}}_1\right], \tag{32}$$

$$\hat{\mathbf{X}}(1)\wedge\dot{\hat{\mathbf{X}}}(1)\wedge\ddot{\hat{\mathbf{X}}}(1)=6\hat{\Omega}_1\frac{\hat{w}_0}{\hat{w}_1}\left[\left(1+\frac{\hat{\gamma}_0}{3}\right)\hat{\mathbf{X}}_0+\frac{\hat{\Omega}_0}{2}\hat{\mathbf{T}}_0\right]\wedge\hat{\mathbf{X}}_1\wedge\hat{\mathbf{T}}_1. \tag{33}$$

The torsion properties at the end points are given by

$$\hat{\mathbf{X}}(0)\wedge\dot{\hat{\mathbf{X}}}(0)\wedge\ddot{\hat{\mathbf{X}}}(0)\wedge\dddot{\hat{\mathbf{X}}}(0)=12\frac{\hat{w}_1^2}{w_0^2}\hat{\Omega}_0\hat{\Omega}_1\hat{\mathbf{X}}_0\wedge\hat{\mathbf{T}}_0\wedge\hat{\mathbf{X}}_1\wedge\hat{\mathbf{T}}_1,$$

$$\hat{\mathbf{X}}(1)\wedge\dot{\hat{\mathbf{X}}}(1)\wedge\ddot{\hat{\mathbf{X}}}(1)\wedge\dddot{\hat{\mathbf{X}}}(1)=12\frac{\hat{w}_1^2}{w_0^2}\hat{\Omega}_0\hat{\Omega}_1\hat{\mathbf{X}}_0\wedge\hat{\mathbf{T}}_0\wedge\hat{\mathbf{X}}_1\wedge\hat{\mathbf{T}}_1,$$

Let $\hat{\mathbf{X}}_{-1}$, $\hat{\Omega}_{-1}\hat{\mathbf{T}}_{-1}$, $\hat{\mathbf{X}}_0$, and $\hat{\Omega}_0\hat{\mathbf{T}}_0$ denote the four control points of the "left" cubic segment $\hat{\mathbf{X}}_-(t_-)$, and $\hat{\mathbf{X}}_0$, $\hat{\Omega}_0\hat{\mathbf{T}}_0$, $\hat{\mathbf{X}}_1$, and

$\hat{\Omega}_1\hat{\mathbf{T}}_1$ denote those of the "right" cubic segment $\hat{\mathbf{X}}_+(t_+)$. Then the substitution of (30) and (31) into (27) yields $\hat{\alpha}=1$. The curvature continuity condition is given by

$$\hat{w}_{-1}\left[\left(1+\frac{\hat{\gamma}_{-1}}{3}\right)\hat{\mathbf{X}}_{-1}+\frac{1}{3}\hat{\Omega}_{-1}\hat{\mathbf{T}}_{-1}\right]\wedge\hat{\mathbf{X}}_0\wedge\hat{\mathbf{T}}_0$$
$$=\hat{w}_1\hat{\mathbf{X}}_0\wedge\hat{\mathbf{T}}_0\wedge\left[-\frac{1}{3}\hat{\Omega}_1\hat{\mathbf{T}}_1+\left(1-\frac{\hat{\gamma}_1}{3}\right)\hat{\mathbf{X}}_1\right].$$

Note that the weight $\hat{w}_0$ and the dual speed $\hat{\Omega}_0$ have no influence on the curvature continuity at $\hat{\mathbf{X}}_0$. The condition (34) can be further reduced to

$$(\hat{\Omega}_1\hat{\mathbf{T}}_1+\hat{\mathbf{Z}}_0)\wedge\hat{\mathbf{X}}_0\wedge\hat{\mathbf{T}}_0=0, \tag{35}$$

where

$$\hat{\mathbf{Z}}_0=\frac{\hat{w}_{-1}}{\hat{w}_1}\hat{\Omega}_{-1}\hat{\mathbf{T}}_{-1}+\frac{\hat{w}_{-1}}{\hat{w}_1}(3+\hat{\gamma}_{-1})\hat{\mathbf{X}}_{-1}-(3-\hat{\gamma}_{+1})\hat{\mathbf{X}}_1. \tag{36}$$

Equation (35) indicates that the three points $(\hat{\Omega}_1\hat{\mathbf{T}}_1+\hat{\mathbf{Z}}_0)$, $\hat{\mathbf{X}}_0$, and $\hat{\mathbf{T}}_0$ are collinear. In other words, $(\hat{\Omega}_1\hat{\mathbf{T}}_1+\hat{\mathbf{Z}}_0)$ is expressible as a linear combination of $\hat{\mathbf{X}}_0$, and $\hat{\mathbf{T}}_0$:

$$\hat{\Omega}_1\hat{\mathbf{T}}_1+\hat{\mathbf{Z}}_0=\hat{c}_0\hat{\mathbf{X}}_0+\hat{c}_0'\hat{\mathbf{T}}_0, \tag{37}$$

where $\hat{c}_0$, $\hat{c}_0'$ are two arbitrary dual numbers.

The condition for torsion continuity at $\hat{\mathbf{X}}_0$ is given by

$$\hat{w}_{-1}^2\hat{\Omega}_{-1}\hat{\mathbf{X}}_{-1}\wedge\hat{\mathbf{T}}_{-1}\wedge\hat{\mathbf{X}}_0\wedge\hat{\mathbf{T}}_0=\hat{w}_1^2\hat{\Omega}_1\hat{\mathbf{X}}_0\wedge\hat{\mathbf{T}}_0\wedge\hat{\mathbf{X}}_1\wedge\hat{\mathbf{T}}_1. \tag{38}$$

Note that the weight $\hat{w}_0$ and the dual speed $\hat{\Omega}_0$ have no influence on the torsion continuity at $\hat{\mathbf{X}}_0$. Eliminate $\hat{\mathbf{T}}_1$ from (38) using (34) to obtain

$$\hat{\Omega}_{-1}(\hat{w}_{-1}\hat{\mathbf{X}}_{-1}+\hat{w}_1\hat{\mathbf{X}}_1)\wedge\hat{\mathbf{T}}_{-1}\wedge\hat{\mathbf{X}}_0\wedge\hat{\mathbf{T}}_0$$
$$=\hat{w}_1(3+\hat{\gamma}_{-1})\hat{\mathbf{X}}_{-1}\wedge\hat{\mathbf{X}}_1\wedge\hat{\mathbf{X}}_0\wedge\hat{\mathbf{T}}_0. \tag{39}$$

6 A Piecewise Cubic Hermite Interpolation

This section solves the following problem:

Given: Image points $\hat{\mathbf{X}}_0, \hat{\mathbf{X}}_1, \cdots, \hat{\mathbf{X}}_{L-1}$ representing L configurations of a rigid body in physical space.

Find: A piecewise cubic motion of the object that passes through the given configurations such that the corresponding image curve has curvature and torsion continuities.

The given image points are chosen as junction Hermite points of the piecewise cubic interpolation. The goal is to determine a tangent vector $\hat{\Omega}_i\hat{\mathbf{T}}_i$ at each image point $\hat{\mathbf{X}}_i$, where $\hat{\Omega}_i$ denotes half of the dual speed and $\hat{\mathbf{T}}_i$ denotes the polar point, such that the resulting interpolation has the prescribed continuities.

Consider three cubic segments with the junction points denoted by $\hat{\mathbf{X}}_{i-1}$, $\hat{\mathbf{X}}_i$, $\hat{\mathbf{X}}_{i+1}$, and $\hat{\mathbf{X}}_{i+2}$, where $i=1, 2, \cdots, L-3$. Assume that the tangent vectors $\hat{\Omega}_{i-1}\hat{\mathbf{T}}_{i-1}$, $\hat{\Omega}_i\hat{\mathbf{T}}_i$ are known from the continuity conditions of the preceding cubic segments. In what follows we develop a method for determining the tangent vector $\hat{\Omega}_{i+1}\hat{\mathbf{T}}_{i+1}$ such that these cubic pieces satisfy the conditions for curvature continuity at $\hat{\mathbf{X}}_i$ and torsion continuity at $\hat{\mathbf{X}}_{i+1}$. This is the essential part of the proposed algorithm for generating cubic Hermite interpolating spline motions.

Position and tangent continuities are implied by the Hermite interpolation. In view of (36) and (37), the curvature continuity at $\hat{\mathbf{X}}_i$ is given by

$$\hat{\Omega}_{i+1}\hat{\mathbf{T}}_{i+1}=\hat{c}_i\hat{\mathbf{X}}_i+\hat{c}_i'\hat{\mathbf{T}}_i-\hat{\mathbf{Z}}_i \tag{40}$$

where

$$\hat{\mathbf{Z}}_i=\frac{\hat{w}_{i-1}}{\hat{w}_{i+1}}\hat{\Omega}_{i-1}\hat{\mathbf{T}}_{i-1}+\frac{\hat{w}_{i-1}}{\hat{w}_{i+1}}(3+\hat{\gamma}_{i-1})\hat{\mathbf{X}}_{i-1}-(3-\hat{\gamma}_{i+1})\hat{\mathbf{X}}_{i+1}.$$

The choice of $\hat{c}_i$, $\hat{c}_i'$ must be such that $\hat{\mathbf{T}}_{i+1}$ is a polar point of $\hat{\mathbf{X}}_{i+1}$ and that the cubic segments satisfy torsion continuity condition at $\hat{\mathbf{X}}_{i+1}$.

The point $\hat{\mathbf{T}}_{i+1}$ is a polar point of $\hat{\mathbf{X}}_{i+1}$ if $\hat{\mathbf{T}}_{i+1}\bullet\hat{\mathbf{X}}_{+1}=0$ which yields, after the substitution of (40):

$$\hat{c}_i\hat{\mathbf{X}}_i\bullet\hat{\mathbf{X}}_{i+1}+\hat{c}_i'\hat{\mathbf{T}}_i\bullet\hat{\mathbf{X}}_{i+1}=\hat{\mathbf{Z}}_i\bullet\hat{\mathbf{X}}_{i+1}. \tag{41}$$

In view of (39), the condition for torsion continuity at $\hat{\mathbf{X}}_{i+1}$ is given by

$$\hat{\Omega}_i(\hat{w}_i\hat{\mathbf{X}}_i+\hat{w}_{i+2}\hat{\mathbf{X}}_{i+2})\wedge\hat{\mathbf{T}}_i\wedge\hat{\mathbf{X}}_{i+1}\wedge\hat{\mathbf{T}}_{i+1}$$
$$=\hat{w}_{i+2}(3+\hat{\gamma}_i)\hat{\mathbf{X}}_i\wedge\hat{\mathbf{X}}_{i+2}\wedge\hat{\mathbf{X}}_{i+1}\wedge\hat{\mathbf{T}}_{i+1}. \tag{42}$$

After the substitution of (40), Eq. (42) becomes

$$[\hat{\Omega}_i\hat{c}_i-(3+\hat{\gamma}_i)\hat{c}_i']\hat{\mathbf{X}}_i\wedge\hat{\mathbf{X}}_{i+1}\wedge\hat{\mathbf{X}}_{i+2}\wedge\hat{\mathbf{T}}_i=\hat{\mathbf{Q}}_i, \tag{43}$$

where

$$\hat{\mathbf{Q}}_i=\hat{\Omega}_i\left(\frac{\hat{w}_i}{\hat{w}_{i+2}}\hat{\mathbf{X}}_i+\hat{\mathbf{X}}_{i+2}\right)\wedge\hat{\mathbf{X}}_{i+1}\wedge\hat{\mathbf{T}}_i\wedge\hat{\mathbf{Z}}_i$$
$$-(3+\hat{\gamma}_i)\hat{\mathbf{X}}_i\wedge\hat{\mathbf{X}}_{i+1}\wedge\hat{\mathbf{X}}_{i+2}\wedge\hat{\mathbf{Z}}_i. \tag{44}$$

Equation (43) is essentially a scalar equation, and together with (41), we have a system of two dual-number equations with two dual-number unknowns, namely, $\hat{c}_i$ and $\hat{c}_i'$. Once $\hat{c}_i$ and $\hat{c}_i'$ are solved, Eq. (40) can be used to determine $\hat{\Omega}_{i+1}$ and $\hat{\mathbf{T}}_{i+1}$.

With the above method, one can generate the tangent vectors $\hat{\Omega}_i\hat{\mathbf{T}}_i$ ($i=2, 3, \cdots, L-2$) such that the resulting piecewise cubic Hermite image curve has curvature continuity at $\hat{\mathbf{X}}_1$, torsion continuity at $\hat{\mathbf{X}}_{L-2}$, and curvature and torsion continuities at the inbetween points $\hat{\mathbf{X}}_j$ ($j=2, 3, \cdots, L-3$), provided that the first two tangent vectors $\hat{\Omega}_0\hat{\mathbf{T}}_0$, $\hat{\Omega}_1\hat{\mathbf{T}}_1$ are known. We now left with two tasks of specifying the end conditions: one is to select $\hat{\Omega}_0\hat{\mathbf{T}}_0$, $\hat{\Omega}_1\hat{\mathbf{T}}_1$ such that the resulting image curve is torsion-continuous at $\hat{\mathbf{X}}_1$; and the other is to select the last tangent vector $\hat{\Omega}_{L-1}\hat{\mathbf{T}}_{L-1}$ such that the image curve is curvature-continuous at $\hat{\mathbf{X}}_{L-2}$.

To obtain the first two tangent vectors $\hat{\Omega}_0\hat{\mathbf{T}}_0$, $\hat{\Omega}_1\hat{\mathbf{T}}_1$, we specify two unit dual vectors $\hat{\mathbf{v}}_0$, $\hat{\mathbf{v}}_1$ that represent the instantaneous screw axes at configurations $\hat{\mathbf{X}}_0$, $\hat{\mathbf{X}}_1$, respectively, and then compute $\hat{\mathbf{T}}_0$, $\hat{\mathbf{T}}_1$ from the following quaternion product:

$$\hat{\mathbf{T}}_0=(1/2)\hat{\mathbf{v}}_0\hat{\mathbf{X}}_0, \quad \hat{\mathbf{T}}_1=(1/2)\hat{\mathbf{v}}_1\hat{\mathbf{X}}_1.$$

To achieve torsion continuity at $\hat{\mathbf{X}}_1$, we obtain the dual speed $\hat{\Omega}_0$

$$\hat{\Omega}_0=\frac{\hat{w}_2(3+\hat{\gamma}_0)\,|\hat{\mathbf{X}}_0\wedge\hat{\mathbf{X}}_2\wedge\hat{\mathbf{X}}_1\wedge\hat{\mathbf{T}}_1|}{|(\hat{w}_0\hat{\mathbf{X}}_0+\hat{w}_2\hat{\mathbf{X}}_2)\wedge\hat{\mathbf{T}}_0\wedge\hat{\mathbf{X}}_1\wedge\hat{\mathbf{T}}_1|},$$

in view of (39). The dual speed $\hat{\Omega}_1$ can be arbitrary chosen. To obtain the last tangent vectors, we specify an additional image point $\hat{\mathbf{X}}_L$ and then compute $\hat{\Omega}_{L-1}\hat{\mathbf{T}}_{L-1}$ using (41) and (43) so that curvature continuity at $\hat{\mathbf{X}}_{L-2}$ can be achieved. It should be noted that the path of the interpolating motion is very sensitive to the choice of these end conditions. Further research is needed to obtain a set of "optimal" end conditions, such as those that would produce not only smooth but also naturally looking motions.

The weights $\hat{w}_i$ ($i=-1, 0, \cdots, L-1$) have positive real parts and can be set to $\hat{w}_i=1$ initially. The weights $\hat{\gamma}_i$ ($i=-1, 0, \cdots, L-1$) are arbitrary dual numbers and can be set to $\hat{\gamma}_i=0$ initially. These parameters can be adjusted to fine-tune the interpolating motion. Figure 1 shows a set of seven input configurations and Fig. 2 shows a piecewise cubic Hermite motion with curvature and torsion continuities.

Conclusions

In this paper we have presented a new approach for smooth motion interpolation based on an orientable kinematic mapping. The mapping reduces the problem of motion interpolation to that of designing an interpolating curve in an orientable projective dual three-space. Methods for cubic interpolation and piecewise cubic interpolation of displacements have been developed taking advantage of existing techniques in CAGD. The results, in addition to their theoretical interest in com-

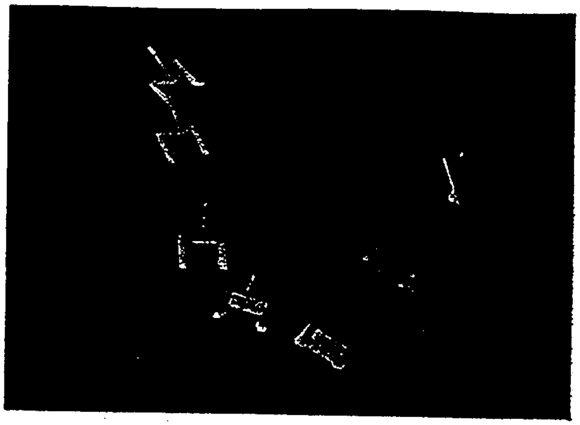


Fig. 1 A set of seven configurations of an object

Fig. 2 A piecewise cubic Hermite interpolating motion with curvature and torsion continuities

putational geometry of motion, have a number of applications in engineering and computer science.

Acknowledgment

The first author gratefully acknowledges the support by National Science Foundation (NSF) Research Initiation Award MSS-9396265 to the State University of New York at Stony Brook. The second author gratefully acknowledges the support by NSF grant DMC-8796348 and the US Army Research Office grant DAAL03-90-0005 to the University of California at Davis.

References

Bottema, O., and Roth, B., 1979, *Theoretical Kinematics*, North Holland Pulb., Amsterdam, 558 pp.

Canny, J., 1986, "Collision Detection for Moving Polyhedra," *IEEE Trans. on Patt. Anal. and Mach. Intel.*, PAMI-8(2), pp. 200–209.

DoCarmo, M. P., 1976, *Differential Geometry of Curves and Surfaces*, Prentice-Hall, New Jersey, 503 pp.

Duff, T., 1986, "Quaternion Splines for Animating Orientation," Technical Report, AT&T Bell Laboratories.

Farin, G., 1993, *Curves and Surfaes for CAGD: A Practical Guide*, 3rd ed., Academic Press, San Diego, 473 pp.

Faux, I. D., and Pratt, M. J., 1979, *Computational Geometry for Design and Manufacturing*, Ellis Horwood Limited, Chichester, England, 331 pp.

Flanders, H., 1963, *Differential Forms with Application to the Physical Sciences*, Academic Press, New York, 205 pp.

Ge, Q. J., and McCarthy, J. M., 1990, "An Algebraic Formulation of Configuration-Space Obstacles for Spatial Robots," *Proc. IEEE Int'l. Conf. on Robotics and Automation*, Cincinnati, OH, Vol. 3, pp. 1542–1547.

Ge, Q. J., and McCarthy, J. M., 1991a, "The Algebraic Classification of the Image Curves of Spherical Four-Bar Motion," ASME JOURNAL OF MECHANICAL DESIGN, Vol. 113, No. 3, pp. 227–231.

Ge, Q. J., and McCarthy, J. M., 1991b, "Functional Constraints as Algebraic Manifolds in a Clifford Algebra," *IEEE Trans. on Robotics and Automation*, Vol. 7, No. 5, pp. 670–677.

Hunt, K. H., 1978, *Kinematic Geometry of Mechanisms*, Clarendon Press, Oxford, 465 pp.

Lozano-Perez, T., 1981, "Automatic Planning of Manipulator Transfer Movements," *IEEE Trans. on Sys., Man, and Cyber.*, SMC-11(10), pp. 681–698.

McCarthy, J. M., and Ravani, B., 1986, "Differential Kinematics of Spherical and Spatial Motions using Kinematic Mapping," ASME *Journal of Applied Mechanics*, Vol. 53, pp. 15–22.

McCarthy, J. M., 1987, "The Differential Geometry of Curves in an Image Spacce of Spherical Kinematics," *J. of Mech. Mach. Theory*, Vol. 22, No. 3, pp. 205–211.

McCarthy, J. M., 1990, *Introduction to Theoretical Kinematics*, MIT Press, Cambridge, MA, 130 pp.

Paul, R. P., 1981, *Robot Manipulator—Mathematics, Programming, and Control*, Boston, MIT Press.

Pletinckx, D., 1989, "Quaternion Calculus as a Basic Tool in Computer Graphics," *The Visual Computer*, Vol. 5, pp. 2–13.

Ravani, B., and Roth, B., 1984, "Mappings of Spatial Kinematics," ASME JOURNAL OF MECHANISMS, TRANSMISSIONS, AND AUTOMATION IN DESIGN, Vol. 106, No. 3, pp. 341–347.

Reeves, W., 1981, "Inbetweening for Computer Animation Utilizing Moving Point Constraints" *ACM Siggraph*, Vol. 15, No. 3, pp. 263–269.

Shoemake, K., 1985, "Animating Rotation with Quaternion Curves," *ACM Siggraph*, Vol. 19, No. 3, pp. 245–254.

Study, E., 1891, "Von den Bewegungen und Umlegungen," *Math. Ann.*, Vol. 39, pp. 441–566.

Tsai, F. F., and Haug, E. J., 1991, "Real Time Multi-Body Systems Dynamics Simulation, Part I: Modified Recursive Form and Topological Analysis," *Mechanics of Structures and Machines*, Vol. 19, No. 1, pp. 99–127.

Turner, R., Balaguer, F., Gobbett, E., and Thalmann, D., 1991, "Physical Based Interactive Camera Motion Control using 3D Input Devices," *Proc. 1991 CGI Conf.*, N. M. Patrikalakis, ed., Springer-Verlag Publ., pp. 135–145.

APPENDIX A

Dual Angle

Let $\hat{\mathbf{X}} = \mathbf{X} + \epsilon\hat{\mathbf{X}}^0$ and $\hat{\mathbf{Y}} = \mathbf{Y} + \epsilon\hat{\mathbf{Y}}^0$ denote the unit-normalized coordinates of two image points. Then the dual distance $\hat{\phi} = \phi + \epsilon h$ (where $\phi = [0, \pi]$) between them may be symbolically given by $\hat{\phi} = \cos^{-1}(\hat{\mathbf{X}} \cdot \hat{\mathbf{Y}})$. If $\mathbf{X} \neq \mathbf{Y}$, then we have

$$\cos^{-1}(\hat{\mathbf{X}} \cdot \hat{\mathbf{Y}}) = \cos^{-1}(\mathbf{X} \cdot \mathbf{Y}) - \epsilon(\mathbf{X} \cdot \mathbf{Y}^0 + \mathbf{X}^0 \cdot \mathbf{Y})/(1 - \mathbf{X} \cdot \mathbf{Y})^{1/2}.$$

In the special case when $\mathbf{X} = \mathbf{Y}$, the displacements $\hat{\mathbf{X}}$ and $\hat{\mathbf{Y}}$ are related by a pure translation of distance $2|\mathbf{Y}^0\mathbf{Y}^{-1} - \mathbf{X}^0\mathbf{X}^{-1}|$, where $|\mathbf{Y}^0\mathbf{Y}^{-1} - \mathbf{X}^0\mathbf{X}^{-1}|$ denotes the magnitude of the vector quaternion $(\mathbf{Y}^0\mathbf{Y}^{-1} - \mathbf{X}^0\mathbf{X}^{-1})$. Therefore we may define $\cos^{-1}(\hat{\mathbf{X}} \cdot \hat{\mathbf{Y}})$ as

$$\cos^{-1}(\hat{\mathbf{X}} \cdot \hat{\mathbf{Y}}) = \epsilon|\mathbf{Y}^0\mathbf{Y}^{-1} - \mathbf{X}^0\mathbf{X}^{-1}|.$$

APPENDIX B

Normalization

General image space coordinates, $\hat{\mathbf{W}} = \mathbf{W} + \epsilon\mathbf{W}^0$, may be normalized such that the corresponding normalized coordinates, $\hat{\mathbf{X}} = \mathbf{X} + \epsilon\mathbf{X}^0$, satisfy $\mathbf{X} \cdot \mathbf{X} = 1$ and $\mathbf{X} \cdot \mathbf{X}^0 = 0$. This involves computing the norm, $\hat{w} = w + \epsilon w^0$, of $\hat{\mathbf{W}}$:

$$w = (\mathbf{W} \cdot \mathbf{W})^{1/2}, \quad w^0 = (\mathbf{W} \cdot \mathbf{W}^0)/w$$

and follows by dividing $\hat{\mathbf{W}}$ with $\hat{w}$:

$$\hat{\mathbf{X}} = \hat{\mathbf{W}}/\hat{w} = (\hat{W}_1/\hat{w},\ \hat{W}_2/\hat{w},\ \hat{W}_3/\hat{w},\ \hat{W}_4/\hat{w}).$$

Division of a dual number $\hat{W}_i = W_i + \epsilon W_i^0$ by $\hat{w}$ is possible and unambiguous if $w \neq 0$ (Bottema and Roth, 1979):

$$\frac{W_i + \epsilon W_i^0}{w + \epsilon w^0} = \frac{W_i}{w} + \epsilon\frac{wW_i^0 - w^0 W_i}{w^2}.$$

Geometric Construction of Bézier Motions

Q. J. Ge
Assistant Professor,
Department of Mechanical Engineering,
State University of New York
at Stony Brook,
Stony Brook, NY 11794

B. Ravani
Professor,
Department of Mechanical and
Aeronautical Engineering,
University of California at Davis,
Davis, CA 95616

This paper deals with discrete computational geometry of motion. It combines concepts from the fields of kinematics and computer aided geometric design and develops a computational geometric framework for geometric construction of motions useful in mechanical systems animation, robot trajectory planning and key framing in computer graphics. In particular, screw motion interpolants are used in conjunction with deCasteljau-type methods to construct Bézier motions. The properties of the resulting Bézier motions are studied and it is shown that the Bézier motions obtained by application of the deCasteljau construction are not, in general, of polynomial type and do not possess the useful subdivision property of Bernstein-Bézier curves. An alternative form of deCasteljau algorithm is presented that results in Bézier motions with subdivision property and Bernstein basis function. The results are illustrated by examples.

Introduction

One of the basic problems in computer aided animation of three-dimensional objects is that of motion approximation. This is the problem of finding a motion that approximates a sequence of arbitrary displacements or configurations[1] of an object. If the motion is made to interpolate through these configurations (called *key configurations*), then the problem of motion approximation becomes that of motion interpolation. Such motion approximation methods are of fundamental importance in computer animation of three-dimensional objects and have applications in computer graphics, computer vision and simulation of mechanical systems.

Our work, in this paper, is motivated by the work by Taylor (1979) and Shoemake (1985) who used quaternions to develop methods for animating rotations. Taylor (1979) used quaternions for linear interpolations of rotations. Shoemake (1985) interpreted quaternions as defining a hypersphere and used the deCasteljau algorithm to obtain the so-called *spherical Bézier curves* for animating rotations. The work of Shoemake was further refined and extended by Duff (1986) at AT&T, and Pletinckx (1989). These researchers, however, did not study some of the fundamental properties of the resulting motions. Shoemake (1985) raised several unanswered questions on the characteristics of the resulting spherical Bézier curves. Here we develop a theoretical framework that enables us to study the basic characteristics of the resulting motion and provide answers to some of these questions. We show, for example, that a motion obtained by application of deCasteljau construction is not in general of Bernstein form and is nonalgebraic. Furthermore, such a motion does not have the subdivision property of Bernstein-Bézier curves. We then present a modification of this algorithm that results in Bézier motions with subdivision property and basis functions which are of Bernstein form. This paper compliments the analytical results presented in our companion paper (Ge and Ravani, 1994) in providing discrete (rather than continuous) computational algorithms for motion interpolation and approximation.

The organization of the paper is as follows. First we discuss geometric representation of spatial displacements in terms of an image space of a kinematic mapping (see Ravani and Roth, 1984). Section 2 deals with the simplest form of motion interpolation namely that of constant speed screw motions. Section 3 presents a deCasteljau construction algorithm based on repeated screw-motion interpolations for the design of non-algebraic Bézier motions. In this section, we also study the kinematic properties of the resulting Bézier motions. In Section 4, we will present a slightly modified form of the deCasteljau algorithm using special projection that results in Bézier motions which are of Bernstein form.

1 Geometric Representation of Displacements

It is well known that a general displacement in a Euclidean three-space (denoted by E^3) has a fixed line, call the *screw axis*. The position and direction of the screw axis in E^3 remains the same before and after the displacement. The sense of direction of the axis is, however, undetermined. The spatial displacement is commonly characterized as an *unoriented* screw displacement, which is a rotation about and a translation along an *undirected* screw axis.

A screw axis with a uniquely defined sense of direction is called a *directed* screw axis or a *spear* (see Bottema and Roth, 1979). A screw displacement about a directed screw axis is an *oriented* screw displacement. Two screw displacements are considered to be "oppositely oriented" if their screw axes occupy the same position in space but with opposite sense of direction.

[1] Positions and orientations.

Contributed by the Design Automation Committee for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received Feb. 1992; revised Jan. 1994. Associate Technical Editor: G. A. Gabriele.

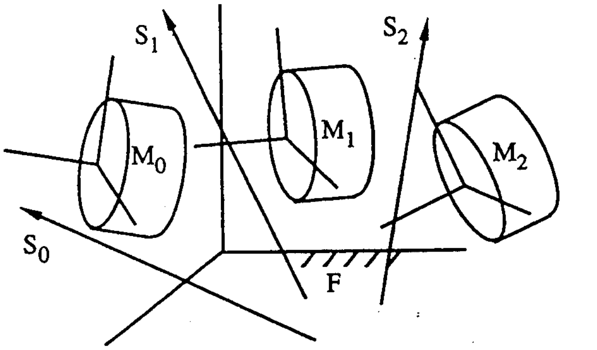


Fig. 1 Three configurations (M_0, M_1 and M_2) belonging to a motion of body M in physical space F. Lines S_0, S_1 and S_2 denote, respectively, the screw axes for displacements F—M_0, F—M_1, and F—M_2.

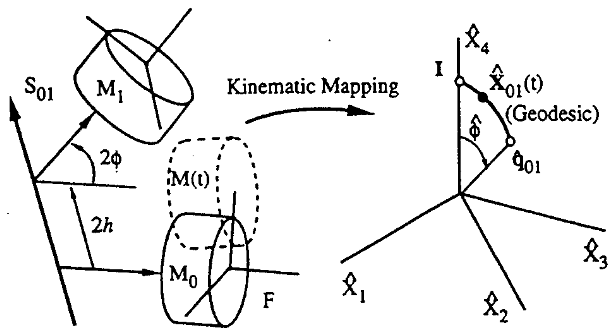


Fig. 2 A screw motion $M(t)$ from M_0 to M_1 corresponds to a line-segment (or geodesic) $\hat{X}_{01}(t)$ from I to $\hat{q}_{01}$ in Σ.

In this way, a general displacement can be reduced to one of the two oppositely oriented screw displacements. These two displacements are geometrically equivalent but topologically different; one may be called a "forward" screw displacement and the other may be called a "backward" screw displacement.

The orientation of a screw displacement is not an issue when only discrete displacements are considered. When a set of screw displacements are used to construct a continuous motion, however, one has to pay special attention to their orientation. Consider a motion of a rigid body in the Euclidean three-space E^3. We attach coordinate frames M and F to the body and the space, respectively. Then each of the configurations that the body attains during the motion is represented by a screw displacement from F to M, see Fig. 1. To reconstruct a smooth and natural motion from these screw displacements, one has to choose the sense of direction of the corresponding screw axes consistently throughout the motion.

Algebraically, an undirected screw axis S is commonly represented by a pair of normalized *Plücker vectors* $(\mathbf{s}, \mathbf{s}_0)$ where $\mathbf{s} = (s_x, s_y, s_z)$ is a unit vector along S and $\mathbf{s}_0 = (s_x^0, s_y^0, s_z^0)$ is the moment of $\mathbf{s}$ about the origin of a fixed reference frame in E^3. They must satisfy the condition $\mathbf{s} \cdot \mathbf{s}^0 = 0$. If the dual-number unit ϵ (defined by $\epsilon^2 = 0$) is used, Plücker vectors can be written as a *dual vector* or a vector of dual numbers:

$$\hat{\mathbf{s}} = \mathbf{s} + \epsilon \mathbf{s}^0 = (\hat{s}_x, \hat{s}_y, \hat{s}_z) \tag{1}$$

where $\hat{s}_i = s_i + \epsilon s_i^0$ $(i = x, y, z)$ are dual numbers. The dual vector $\hat{\mathbf{s}}$ has the property $\hat{\mathbf{s}} \cdot \hat{\mathbf{s}} = 1$ and is termed a *unit line vector*.

A directed screw axis S can be represented algebraically by a *directed* unit line vector. By this we mean that $\hat{\mathbf{s}}$ and $-\hat{\mathbf{s}}$ represent a pair of oppositely directed screw axes instead of the same axis. The use of directed screw axis instead of undirected screw axis removes the ambiguity in defining the magnitude of a screw displacement. The magnitude is defined in terms of the *dual angle* $\hat{\theta} = \theta + \epsilon l$ between two directed lines m_1, m_2 that are perpendicular to the directed screw axis S, where θ is the angle from m_1 to m_2 about S, according to the right-hand screw rule, and l is the *signed* distance from m_1 to m_2 along S. For a pair of oppositely oriented but geometrically equivalent screw displacements, their dual angles $\hat{\theta}$ and $\hat{\theta}'$ are related by the ange 2π, i.e., $\hat{\theta} + \hat{\theta}' = 2\pi$. Thus a spatial displacement may be represented either by a forward screw displacement about the screw axis $\hat{\mathbf{s}}$ with a dual angle $\hat{\theta}$ or by a backward screw displacement about the screw axis $-\hat{\mathbf{s}}$ with a dual angle $2\pi - \hat{\theta}$.

Although a directed screw axis $\hat{\mathbf{s}}$ and a dual angle $\hat{\theta}$ completely prescribe an oriented screw displacement, they are seldomly used directly as a representation of the displacement. Instead, they are often used to define the *dual Euler parameters*, denoted by $\hat{\mathbf{X}} = (\hat{X}_1, \hat{X}_2, \hat{X}_3, \hat{X}_4)$, as (see Bottema and Roth, 1979):

$$\begin{aligned} \hat{X}_1 &= \hat{s}_1 \sin(\hat{\theta}/2), \quad \hat{X}_2 = \hat{s}_2 \sin(\hat{\theta}/2), \\ \hat{X}_3 &= \hat{s}_3 \sin(\hat{\theta}/2), \quad \hat{X}_4 = \cos(\hat{\theta}/2), \end{aligned} \tag{2}$$

where

$$\begin{aligned} \sin(\hat{\theta}/2) &= \sin(\theta/2) + \epsilon(l/2)\cos(\theta/2), \\ \cos(\hat{\theta}/2) &= \cos(\theta/2) - \epsilon(l/2)\sin(\theta/2). \end{aligned}$$

These parameters are considered to be *signed*, for a backward screw displacement, prescribed by $-\hat{\mathbf{s}}$ and $2\pi - \hat{\theta}$, corresponds to the dual Euler parameters $-\hat{\mathbf{X}} = (-\hat{X}_1, -\hat{X}_2, -\hat{X}_3, -\hat{X}_4)$.

In this paper, we consider the signed dual Euler parameters as a set of four *signed* homogeneous coordinates that define an oriented point in a projective space with three dual dimensions. By this we mean that the dual Euler parameters $\hat{\mathbf{X}} = (\hat{X}_1, \hat{X}_2, \hat{X}_3, \hat{X}_4)$ and $\hat{w}\hat{\mathbf{X}} = (\hat{w}\hat{X}_1, \hat{w}\hat{X}_2, \hat{w}\hat{X}_3, \hat{w}\hat{X}_4)$, where $\hat{w} = w + \epsilon w^0$ is a nonpure dual number, represent one and the same point for all $w > 0$, and that, they represent a pair of oppositely oriented points for all $w < 0$. In particular, the two points, $\hat{\mathbf{X}}$ and $-\hat{\mathbf{X}}$, known as *antipodal* points, are considered to be two distinct points which occupy the same position in the space but with opposite orientations. In this way, a general displacement, which is geometrically equivalent to a pair of oppositely oriented screw displacements, corresponds to not *one* point but *two* antipodal points in this projective dual three-space. The resulting image space of spatial displacements is the orientable version of the kinematic mapping discussed by Ravani and Roth (1984) and is described in more detail in Ge and Ravani (1994).

2 Geometric Construction of Screw Motions

This section presents a method for geometric construction of an interpolating screw motion that enables the designer to control the speed of the interpolation. This forms the basis for developing a deCasteljau construction algorithm for motion interpolation and approximation.

2.1 Starting From a Reference Configuration. A finite displacement of a rigid body M from an initial configuration M_0 to the next configuration M_1 is shown in Fig. 2. For the time being, we choose the fixed reference frame F to be coincident with M_0. Then the configuration M_0 is represented by the identity displacement F—M_0 which corresponds to a special point $\mathbf{I} = (0, 0, 0, 1)$, called the *identity point*, in the image space Σ. The next configuration M_1 with respect to F is represented by a forward screw displacement F—M_1. Its screw axis, S_{01}, is represented by a directed unit line vector $\hat{\mathbf{s}}_{01} = (\hat{s}_x, \hat{s}_y, \hat{s}_z)$ and its magnitude is given by the dual angle $2\hat{\phi} = 2\phi + \epsilon(2h)$. These screw parameters define the image point, $\hat{\mathbf{q}}_{01}$, of the forward screw displacement F—M_1:

$$\hat{\mathbf{q}}_{01} = (\hat{s}_x \sin\hat{\phi}, \hat{s}_y \sin\hat{\phi}, \hat{s}_z \sin\hat{\phi}, \cos\hat{\phi}).$$

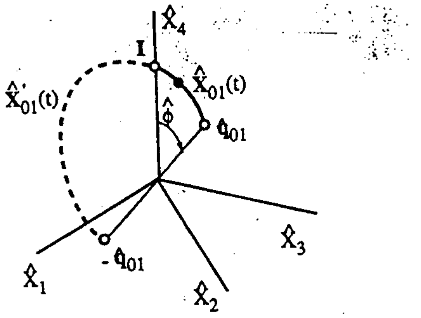


Fig. 3 Two oppositely oriented screw motion interpolants, $\hat{X}_{01}(t)$ and $\hat{X}'_{01}(t)$

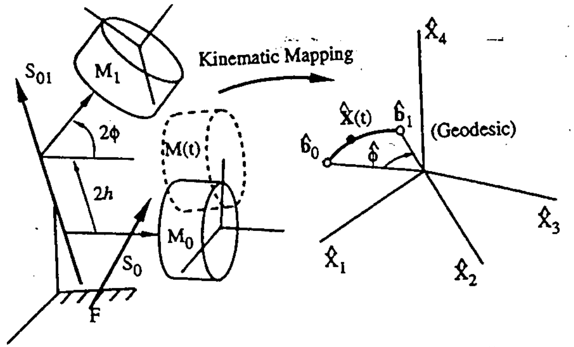


Fig. 4 A screw motion $M(t)$ from M_0 to M_1 corresponds to a line segment $\hat{X}(t)$ from $\hat{b}_0$ to $\hat{b}_1$ in Σ

Now the problem of determining the interpolating screw motion $M(t)$ becomes that of determining a unifold line-segment (or geodesic) $\hat{\mathbf{X}}_{01}(t)$ in Σ such that $\hat{\mathbf{X}}_{01}(0)=\mathbf{I}$ and $\hat{\mathbf{X}}_{01}(1)=\hat{\mathbf{q}}_{01}$, see Fig. 2. The Plücker coordinates of this line is given by the vector wedge product $\mathbf{I}\wedge\hat{\mathbf{q}}_{01}=(\hat{s}_x\sin\hat{\phi},\ \hat{s}\sin\hat{\phi},\ \hat{s}_z\sin\hat{\phi},\ 0,\ 0,\ 0)$. This indicates that the dual vector $\hat{\mathbf{s}}_{01}=(\hat{s}_x,\ \hat{s}_y,\ \hat{s}_z)$ represents the direction of the line-segment $\hat{\mathbf{X}}_{01}(t)$. In view of the additional fact that the length of the line segment is the dual angle $\hat{\phi}$, we obtain the following parametric equation for the line-segment (or geodesic) from $\mathbf{I}$ to $\hat{\mathbf{q}}_{01}$:

$$\hat{X}_{01}(t)=(\hat{s}_x\sin(u(1+\epsilon p)\hat{\phi}),\ \hat{s}_y\sin(u(1+\epsilon p)\hat{\phi}),\ \hat{s}_z\sin(u(1+\epsilon p)\hat{\phi}),\ \cos(u(1+\epsilon p)\hat{\phi})),\quad (3)$$

where $u=u(t)$ is a real-valued timing function defined such that $u(0)=0$ and $u(1)=1$; and $p=p(t)$ is another real valued function defined such that $p(0)=p(1)=0$. The function $u(t)$ allows control of angular velocity of the resulting screw motion and $p(t)$ allows variation of the pitch, $p(t)+h/\phi$, of the screw motion. In this paper we choose $u(t)=t$ and $p(t)=0$ so that the screw motion defined by (3) has not only a constant angular velocity but also a constant pitch. In this case, (3) becomes

$$\hat{\mathbf{X}}_{01}(t)=(\hat{s}_x\sin(t\hat{\phi}),\ \hat{s}_y\sin(t\hat{\phi}),\ \hat{s}_z\sin(t\hat{\phi}),\ \cos(t\hat{\phi}))\quad (4)$$

and is called a *uniform-speed screw interpolant.*[2]

Note that in representing the configuration M_1 with respect to F, the choice of a backward screw displacement $-\hat{\mathbf{q}}_{01}$ instead of the forward displacement $\hat{\mathbf{q}}_{01}$ results in a screw interpolant $\hat{\mathbf{X}}'_{01}(t)$ that has opposite orientation to $\hat{\mathbf{X}}_{01}(t)$ (Fig. 3).

Quaternion algebra is an elegant tool for handling transformations in elliptic three-space (Sommerville, 1914) and, consequently, in the image space (Ravani and Roth, 1984). If the image point $\hat{\mathbf{q}}_{01}$ is represented by a quaternion, then $\hat{\mathbf{X}}_{01}(t)$ given by (4) is expressible as a power of the quaternion $\hat{\mathbf{q}}_{01}$. Let $(\mathbf{i},\ \mathbf{j},\ \mathbf{k},\ 1)$ denote the quaternion basis, then $\hat{\mathbf{q}}_{01}=\hat{\mathbf{s}}_{01}\sin\hat{\phi}+\cos\hat{\phi}$ where $\hat{\mathbf{s}}_{01}=\hat{s}_x\mathbf{i}+\hat{s}_y\mathbf{j}+\hat{s}_z\mathbf{k}$ is a vector quaternion. Since $\hat{\mathbf{s}}_{01}^2=-(\hat{s}_x^2+\hat{s}_y^2+\hat{s}_z^2)=-1$, de Moivre's theorem shows that

$$\hat{\mathbf{X}}_{01}(t)=\hat{\mathbf{s}}_{01}\sin(t\hat{\phi})+\cos(t\hat{\phi})=\hat{\mathbf{q}}_{01}^t.\quad (5)$$

In view of the dual-velocity formula $\hat{\mathbf{V}}=2(d\hat{\mathbf{X}}/dt)\hat{\mathbf{X}}^{-1}$ (Ge and Ravani, 1994), it can be shown that the dual velocity of the screw interpolant (5) is given by $\hat{\mathbf{V}}=\hat{\phi}\hat{\mathbf{s}}_{01}$. This confirms that $\hat{\mathbf{X}}_{01}(t)$ corresponds to a constant-speed screw motion.

2.2 Starting From an Arbitrary Configuration.

We now study the screw motion from configuration M_0 to M_1 for the case when the fixed frame F is not coincident with M_0 (see Fig. 4). Let $\hat{\mathbf{b}}_0$ and $\hat{\mathbf{b}}_1$ denote the image points of the configurations M_0 and M_1 relative to F, respectively. Let $\hat{\mathbf{q}}_{01}$ denote the image point of the displacement $M_0\rightarrow M_1$ measured with respect to M_0. The fact that the displacement $F\rightarrow M_1$ is the composition of the displacements $F\rightarrow M_0$ and $M_0\rightarrow M_1$ can be expressed by the quaternion product $\hat{\mathbf{b}}_1=\hat{\mathbf{b}}_0\hat{\mathbf{q}}_{01}$. From this we obtain $\hat{\mathbf{q}}_{01}=\hat{\mathbf{b}}_0^{-1}\hat{\mathbf{b}}_1$ where $\hat{\mathbf{b}}_0^{-1}$ is the inverse of the unit quaternion $\hat{\mathbf{b}}_0$. Thus the constant-speed screw interpolant from M_0 to M_1 described with respect to F is given by

$$\hat{\mathbf{X}}(t)=\hat{\mathbf{b}}_0\hat{\mathbf{q}}_{01}^t=\hat{\mathbf{b}}_0(\hat{\mathbf{b}}_0^{-1}\hat{\mathbf{b}}_1)^t,\quad t=[0,\ 1].\quad (6)$$

Kinematically, Eq. (6) means that the intermediate displacement represented by $\hat{\mathbf{X}}(t)$ is the composition of the two displacements $F\rightarrow M_0$ and $M_0\rightarrow M(t)$, respectively. The screw interpolant (6) is invariant with respect to change of both the moving and fixed-coordinate frames.

When the two configurations $\hat{\mathbf{b}}_0$ and $\hat{\mathbf{b}}_1$ are not related by a pure translation, the formula for screw interpolation given by the right hand side of (6), denoted by $\hat{\mathbf{L}}(\hat{\mathbf{b}}_0,\ \hat{\mathbf{b}}_1;\ t)$, can be put into the following vector form

$$\hat{\mathbf{L}}(\hat{\mathbf{b}}_0,\ \hat{\mathbf{b}}_1;\ t)=\frac{\sin((1-t)\hat{\phi})}{\sin\hat{\phi}}\hat{\mathbf{b}}_0+\frac{\sin(t\hat{\phi})}{\sin\hat{\phi}}\hat{\mathbf{b}}_1\quad (7)$$

where $\hat{\phi}=\phi+\epsilon h$ $(\phi\neq 0)$ is the dual distance between the points $\hat{\mathbf{b}}_0$ and $\hat{\mathbf{b}}_1$, see Fig. 4. Note that the real part of (7) is the *spherical linear interpolation* used in Shoemake (1985) and can be obtained directly using analytical elliptic geometry (see Sommerville, 1914).

When the two configurations are related by a pure translation, i.e., when $\phi=0$, Eq. (7) becomes a linear interpolation:

$$\hat{\mathbf{L}}(\hat{\mathbf{b}}_0,\ \hat{\mathbf{b}}_1;\ t)=(1-t)\hat{\mathbf{b}}_0+t\hat{\mathbf{b}}_1\quad (8)$$

since

$$\lim_{\phi\to 0}\frac{\sin((1-t)\hat{\phi})}{\sin\hat{\phi}}=1-t,\quad \lim_{\phi\to 0}\frac{\sin(t\hat{\phi})}{\sin\hat{\phi}}=t.\quad (9)$$

This treatment is necessary since division by a pure dual number ϵh is not defined (Bottema and Roth, 1979). Let $\mathbf{d}_0$, $\mathbf{d}_1$, and $\mathbf{d}(t)$ denote the vectors in the fixed frame F which represent the origins of the body-fixed coordinate frames at M_0, M_1, and $M(t)$, respectively. It is easy to show that Eq. (8) is equivalent to linear interpolation of the origins:

$$\mathbf{d}(t)=(1-t)\mathbf{d}_0+t\mathbf{d}_1.$$

3 Nonalgebraic Bézier Motions

The screw interpolant in the preceding section can be applied repeatedly for constructing curves in the image space in a manner similar to the deCasteljau construction known in computer aided geometric design (see Farin, 1993). This idea starts with Shoemake (1985) who constructed the so-called *spherical Bézier curves* in the space of unit quaternions for animating rotations. The following deCasteljau algorithm may be considered as a dual form of Shoemake's spherical deCasteljau algorithm.

3.1 A DeCasteljau Algorithm for Bézier Motions.

Let $\hat{\mathbf{b}}_i$ $i=0,\ 1,\ \cdots,\ n$ denote $n+1$ image points of spatial displace-

[2] Here the term "screw" is used instead of "linear" to differentiate the linear motion of the form (26).

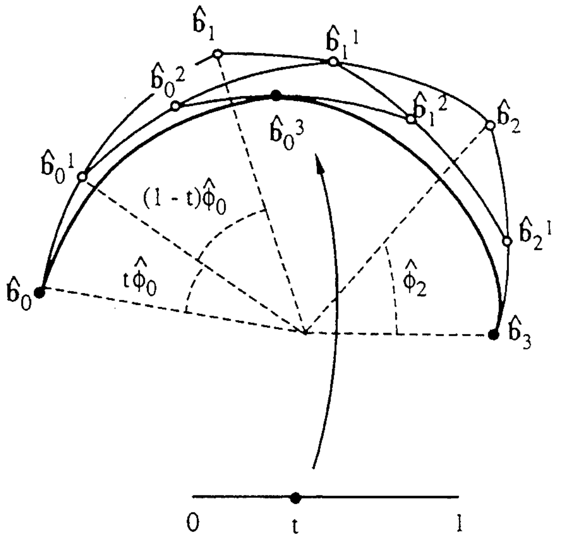


Fig. 5 A deCasteljau algorithm in Σ

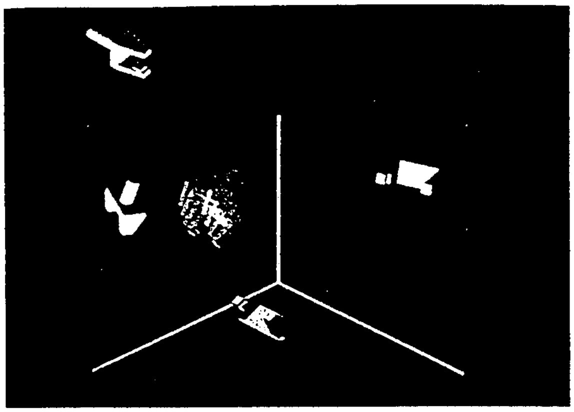


Fig. 6 Four Bézier control configurations together with a few configurations belonging to the Bézier motion

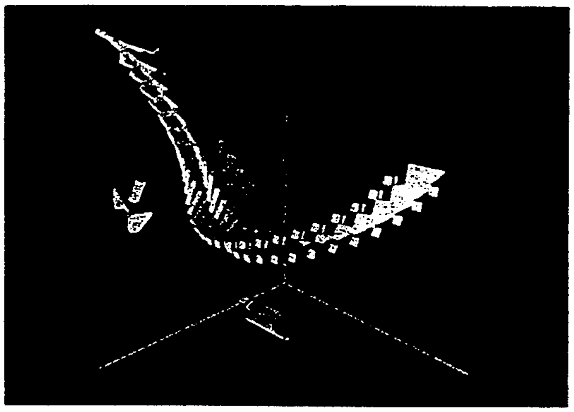


Fig. 7 The entire Bézier motion of rank 3

ments. They form a control polygon in the image space Σ, see Fig. 5 for the case when $n=3$. For $r=1, 2, \cdots, n$, $i=0, 1, \cdots, n-r$ and $0 \leq t \leq 1$, an intermediate point on each screw interpolant is given by the recursive formula:

$$\hat{\mathbf{b}}_i^r(t) = \hat{\mathbf{b}}_i^{r-1}(t)[(\hat{\mathbf{b}}_i^{r-1}(t))^{-1}\hat{\mathbf{b}}_{i+1}^{r-1}(t)]^t, \tag{10}$$

where $\hat{\mathbf{b}}_i^0 = \hat{\mathbf{b}}_i$, $\hat{\phi}_i^0 = \hat{\phi}_i$. The points $\hat{\mathbf{b}}_i^{r-1}(\mathrm{t})$, $\hat{\mathbf{b}}_{i+1}^{r-1}(t)$, $\hat{\mathbf{b}}_i^r(t)$ denote, respectively, the starting point, end point and intermediate point. Let $\hat{\phi}_i^{r-1} = \phi_i^{r-1} + \epsilon(\phi_i^0)^{r-1}$ denote the dual distance between $\hat{\mathbf{b}}_i^{r-1}(t)$ and $\hat{\mathbf{b}}_{i+1}^{r-1}(t)$. Equation (10) is equivalent to the following unifold dual-spherical linear interpolation when $\phi_i^{r-1} \neq 0$:

$$\hat{\mathbf{b}}_i^r(\mathrm{t}) = \frac{\sin((1-t)\hat{\phi}_i^{r-1})}{\sin\hat{\phi}_i^{r-1}}\hat{\mathbf{b}}_i^{r-1}(t) + \frac{\sin(t\hat{\phi}_i^{r-1})}{\sin\hat{\phi}_i^{r-1}}\hat{\mathbf{b}}_{i+1}^{r-1}(t). \tag{11}$$

When $\phi_i^{r-1} = 0$, Eq. (10) reduces to the linear interpolation

$$\hat{\mathbf{b}}_i^r(t) = (1-t)\hat{\mathbf{b}}_i^{r-1}(t) + t\hat{\mathbf{b}}_{i+1}^{r-1}(t). \tag{12}$$

By varying the parameter t in the range [0, 1], the point $\hat{\mathbf{b}}_0^n(t)$ traces out a curve in Σ with control points $\hat{\mathbf{b}}_i$ ($i=0, 1, \cdots, n$). In Section 3.3, we will see that although $\hat{\mathbf{b}}_0^n(t)$ is in genreal not a polynomial curve, it satisfies end-point interpolation conditions in a manner similar to the Bézier curves in E^3. We, therefore, call this curve a *Bézier image curve* of rank n. The corresponding motion is termed a *Bézier motion* of rank n. Figure 6 shows four control configurations of a robot gripper together with several configurations belonging to the Bézier motion of rank 3. Figure 7 shows the entire Bézier motion resulting from the application of the above deCasteljau's construction.

We note that if all control points are collinear in Σ, the Bézier image curve degenerates into a unifold geodesic line-segment which corresponds to a screw motion. If all control points are coplanar, the Bézier image curve becomes a unifold planar curve which corresponds to a line-symmetric motion.

3.2 The Derivatives of Bézier Image Curves. If we expand the derivative of the screw interpolant (11) using the chain rule, we obtain

$$\begin{aligned}\dot{\hat{\mathbf{b}}}_i^r(t) = {} & \hat{\phi}_i^{r-1}\hat{\mathbf{T}}_i^r(t) + \frac{\sin((1-t)\hat{\phi}_i^{r-1})}{\sin\hat{\phi}_i^{r-1}}\dot{\hat{\mathbf{b}}}_i^{r-1}(t) \\ & + \frac{\sin(t\hat{\phi}_i^{r-1})}{\sin\hat{\phi}_i^{r-1}}\dot{\hat{\mathbf{b}}}_{i+1}^{r-1}(t) + \dot{\hat{\phi}}_i^{r-1}\Bigg(t\hat{\mathbf{T}}_i^r(t) + \frac{\cos((1-t)\hat{\phi}_i^{r-1}}{\sin\hat{\phi}_i^{r-1}}\hat{\mathbf{b}}_i^{r-1}(t) \\ & - \frac{\cos\hat{\phi}_i^{r-1}}{\sin\hat{\phi}_i^{r-1}}\hat{\mathbf{b}}^r(t)\Bigg)\end{aligned} \tag{13}$$

where

$$\hat{\mathbf{T}}_i^r(\mathrm{t}) = \frac{\cos(t\hat{\phi}_i^{r-1})\hat{\mathbf{b}}_{i+1}^{r-1}(t) - \cos((1-t)\hat{\phi}_i^{r-1})\hat{\mathbf{b}}_i^{r-1}(t)}{\sin\hat{\phi}_i^{r-1}} \tag{14}$$

is a polar point on the line-segment $\hat{\mathbf{b}}_i^r(t)$, for $\hat{\mathbf{T}}_i^r(t)\cdot\hat{\mathbf{b}}_i^r(t) = 0$ and $\hat{\mathbf{T}}_i^r(t)\cdot\hat{\mathbf{T}}_i^r(t) = 1$. The image space can be interpreted, at least locally, as a hypersphere in a spacce of four dual dimensions (see McCarthy and Ravani, 1986). Then if $\hat{\mathbf{b}}_i^r(t)$ is interpreted as a great circular arc on the hypersphere, $\hat{\mathbf{T}}_i^r(t)$ is the unit tangent vector to the arc at t.

The derivative of a Bézier image curve may be obtained using the recursive formula (13). The explicit formula for the derivative is in general rather complicated except at the end points $t=0$ and $t=1$. When $t=0$, we have $\hat{\phi}_i^{r-1}(0) = \hat{\phi}_i$, $\dot{\hat{\phi}}_i^{r-1}(0) = 0$, $\hat{\mathbf{T}}_i^r(0) = \hat{\mathbf{T}}_i^1(0)$ and therefore Eq. (13) is reduced to

$$\dot{\hat{\mathbf{b}}}_i^r(0) = \hat{\phi}_i\hat{\mathbf{T}}_i^1(0) + \dot{\hat{\mathbf{b}}}_i^{r-1}(0). \tag{15}$$

When $t = 1$, we have $\hat{\phi}_i^{r-1}(1) = \hat{\phi}_{i+r-1}$, $\dot{\hat{\phi}}_i^{r-1}(1) = 0$, $\hat{\mathbf{T}}_i^r(1) = \hat{\mathbf{T}}_{i+r-1}^1(1)$ and Eq. (13) is reduced to

$$\dot{\hat{\mathbf{b}}}_i^r(1) = \hat{\phi}_{i+r-1}\hat{\mathbf{T}}_{i+r-1}^1(1) + \dot{\hat{\mathbf{b}}}_{i+1}^{r-1}(1). \tag{16}$$

From (15) and (16) we obtain the derivatives of a Bézier image curve at $t=0$ and $t=1$:

$$\dot{\hat{\mathbf{b}}}_0^n(1) = n\hat{\phi}_0\hat{\mathbf{T}}_0^1(0), \quad \dot{\hat{\mathbf{b}}}_0^n(1) = n\hat{\phi}_{n-1}\hat{\mathbf{T}}_{n-1}^1(1) \tag{17}$$

where $\hat{\mathbf{T}}_0^1(0)$, $\hat{\mathbf{T}}_{n-1}^1(1)$ are unit tangent vectors given by

$$\hat{\mathbf{T}}_0^1(0) = \frac{\hat{\mathbf{b}}_1 - \hat{\mathbf{b}}_0\cos\hat{\phi}_0}{\sin\hat{\phi}_0}, \quad \hat{\mathbf{T}}_{n-1}^1(1) = \frac{\hat{\mathbf{b}}_n\cos\hat{\phi}_{n-1} - \hat{\mathbf{b}}_{n-1}}{\sin\hat{\phi}_{n-1}}. \tag{18}$$

For a Bézier image curve of rank 3, its tangent bivector can also be given in terms of its control points as

$$\hat{\mathbf{b}}_0^3(0)\wedge\dot{\hat{\mathbf{b}}}_0^3(0) = 3\frac{\hat{\phi}_0}{\sin\hat{\phi}_0}\hat{\mathbf{b}}_0\wedge\hat{\mathbf{b}}_1, \tag{19}$$

$$\hat{\mathbf{b}}_0^3(1)\wedge\hat{\mathbf{T}}_0^3(1) = 3\frac{\hat{\phi}_2}{\sin\hat{\phi}_2}\hat{\mathbf{b}}_2\wedge\hat{\mathbf{b}}_3, \tag{20}$$

where $\hat{\phi}_0$, $\hat{\phi}_2$ are the dual angular distances between $\hat{\mathbf{b}}_0$, $\hat{\mathbf{b}}_1$ and $\hat{\mathbf{b}}_2$, $\hat{\mathbf{b}}_3$, respectively.

The second derivative of the screw interpolant (11) is obtained by differentiating (13). The resulting recursive formula can be used to obtain the second derivative of a Bézier image curve. The derivation is rather lengthy and is not included in this paper. At $t=0$, the recursive formula for computing the second derivative is given by

$$\ddot{\hat{\mathbf{b}}}_i^r(0) = -\hat{\phi}_i\hat{\phi}_i\hat{\mathbf{b}}_i + \dot{\hat{\phi}}_i^{r-1}(0)\hat{\mathbf{T}}_i^1(0) + 2\hat{\phi}_i\hat{\mathbf{D}}_i^r(0) + \ddot{\hat{\mathbf{b}}}_i^{r-1}(0), \tag{21}$$

where

$$\hat{\mathbf{D}}_i^r(0) = \frac{\dot{\hat{\mathbf{b}}}_{i+1}^{r-1}(0) - \dot{\hat{\mathbf{b}}}_i^{r-1}(0)\cos\hat{\phi}_i}{\sin\hat{\phi}_i}.$$

Equation (21) can be used to determine the curvature 3-vector of a Bézier image curve of rank 3 at $t=0$:

$$\hat{\mathbf{b}}_0^3(0)\wedge\dot{\hat{\mathbf{b}}}_0^3(0)\wedge\ddot{\hat{\mathbf{b}}}_0^3(0) = 18\frac{\hat{\phi}_0^2\hat{\phi}_1}{\sin^2\hat{\phi}_0\sin\hat{\phi}_1}\hat{\mathbf{b}}_0\wedge\hat{\mathbf{b}}_1\wedge\hat{\mathbf{b}}_2. \tag{22}$$

Similarly, the curvature 3-vector at $t=1$ is obtained as

$$\hat{\mathbf{b}}_0^3(1)\wedge\dot{\hat{\mathbf{b}}}_0^3(1)\wedge\ddot{\hat{\mathbf{b}}}_0^3(1) = 18\frac{\hat{\phi}_2^2\hat{\phi}_1}{\sin^2\hat{\phi}_2\sin\hat{\phi}_1}\hat{\mathbf{b}}_1\wedge\hat{\mathbf{b}}_2\wedge\hat{\mathbf{b}}_3. \tag{23}$$

The dual curvature $\hat{\kappa}(t)$ at $t=0$ is given by

$$\hat{\kappa}(0) = \frac{|\hat{\mathbf{b}}_0^3(0)\wedge\dot{\hat{\mathbf{b}}}_0^3(0)\wedge\ddot{\hat{\mathbf{b}}}_0^3(0)|}{|\hat{\mathbf{b}}_0^3(0)\wedge\dot{\hat{\mathbf{b}}}_0^3(0)|} = \frac{2\hat{\phi}_1\sin\hat{\psi}_0}{3\hat{\phi}_0\sin\hat{\phi}_0\sin\hat{\phi}_1} \tag{24}$$

where $\hat{\psi}_0$ is the dual angular distance from the point $\hat{\mathbf{b}}_2$ to the line $\hat{\mathbf{b}}_0\hat{\mathbf{b}}_1$.

It is interesting to note, in the limiting case when ϕ_i—0 ($i=0$, 1), the tangent and curvature properties of the Bézier image curves of rank 3 at $t=0$ and $t=1$ approach to those of Bézier cubics in Euclidean three-space at $t=0$ and $t=1$, respectively. For example, at $t=0$, the tangent bivector (19) becomes $3\hat{\mathbf{b}}_0\wedge\hat{\mathbf{b}}_1$, the curvature 3-vector (22) becomes $18\hat{\mathbf{b}}_0\wedge\hat{\mathbf{b}}_1\wedge\hat{\mathbf{b}}_2$, and the dual curvature becomes $2\hat{\psi}_0/3\hat{\phi}_0^2$.

3.3 More Properties of Bézier Motions.

Equations (17) and (18) show that, similar to Bézier curves in Euclidean three-space, the image curve $\hat{\mathbf{b}}_0^n(t)$ is tangent to the first leg $\hat{\mathbf{b}}_0\hat{\mathbf{b}}_1$ and the last leg $\hat{\mathbf{b}}_{n-1}\hat{\mathbf{b}}_n$ of its control polygon. In addition, the magnitude of the tangent is n times that of the leg itself. Furthermore, it can be shown that the segment $\hat{\mathbf{b}}_0^{n-1}\hat{\mathbf{b}}_1^{n-1}$ is tangent to the curve $\hat{\mathbf{b}}_0^n(t)$. Kinematically, this means that the corresponding Bézier motion of rank n (denoted by $\hat{\mathbf{b}}_0^n(t)$) has the following properties:

(*1*) End configuration interpolation.
(*2*) At end configurations $\hat{\mathbf{b}}_0$ and $\hat{\mathbf{b}}_n$, the instantaneous screw axes of $\hat{\mathbf{b}}_0^n(t)$ are given by the screw axes of screw motions $\hat{\mathbf{b}}_0^1(t)$ and $\hat{\mathbf{b}}_{n-1}^1(t)$, respectively.
(*3*) The initial speed of $\hat{\mathbf{b}}_0^n(t)$ at configuration $\hat{\mathbf{b}}_0$ is n times that of the screw motion $\hat{\mathbf{b}}_0^1(t)$ and the final speed of $\hat{\mathbf{b}}_0^n(t)$ at $\hat{\mathbf{b}}_n$ is n times that of the screw motion $\hat{\mathbf{b}}_{n-1}^1(t)$.
(*4*) The instantaneous screw axis at the configuration $\hat{\mathbf{b}}_0^n(t)$ is the same as that of the screw motion from configuration $\hat{\mathbf{b}}_0^{n-1}$ to $\hat{\mathbf{b}}_1^{n-1}$.

Bézier motions, obtained in this fashion, are also coordinate-frame invariant since screw motion interpolants are coordinate-frame invariant.

Figure 8 shows an entire Bézier motion of rank 3 and three constant-speed screw motions defined by the four given control configurations. Clearly, the screw motion from $\hat{\mathbf{b}}_0$ to $\hat{\mathbf{b}}_1$ is "tangent" to the Bézier motion at $\hat{\mathbf{b}}_0$ and the screw motion from $\hat{\mathbf{b}}_2$ to $\hat{\mathbf{b}}_3$ is "tangent" to the Bézier motion at $\hat{\mathbf{b}}_3$.

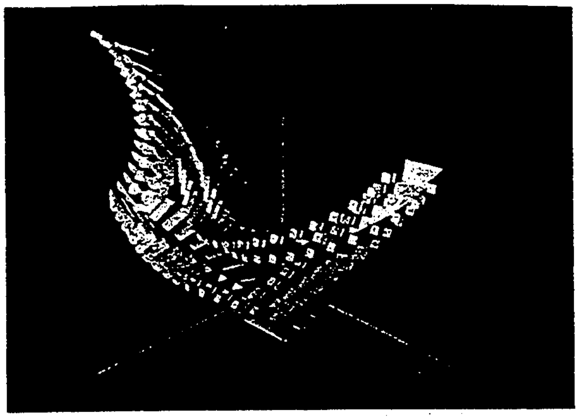


Fig. 8 Bézier motion of rank 3 and the three constant speed screw motions

In addition to the aforementioned end-point interpolation properties, in the special case when all configurations share the same orientation, the Bézier image curve $\hat{\mathbf{b}}_0^n(t)$ reduces to a Bernstein-Bézier polynomial curve, for it is the result of repeated linear interpolations of form (12). The corresponding motion is translational motion and end point of the rigid body undergoing the motion traces out a Bernstein-Bézier polynomial curve in the Euclidean three-space E^3.

In genreal, however, the Bézier image curve $\hat{\mathbf{b}}_0^n(t)$ is fundamentally different from a Bernstein-Bézier curve in E^3. First, while a Bernstein-Bézier curve in E^3 is a polynomial curve, the image curve $\hat{\mathbf{b}}_0^n(t)$ is in general not algebraic but transcendental, for it has in general infinitely many intersections with a properly oriented twofold plane in the image space. Second, while a Bernstein-Bézier curves in E^3 has subdivision property, the image curve $\hat{\mathbf{b}}_0^n(t)$ in general does not possess this useful property of subdivision (see Appendix A for the proof). Consequently, the image curve traced out by the point $\hat{\mathbf{b}}_0^n(t)$ when t varies from 0 to 1 is in general different from the image curve resulting from subdividing the control polygon formed by $\hat{\mathbf{b}}_i$ ($i=0, 1, \cdots, n$).

It is noted here that the real part of the image curve $\mathbf{b}_0^n(t)$ is the same as the "spherical Bézier curve" presented by Shoemake (1985). Pottmann (1992) has also pointed out the lack of subdivision property of the Bézier curves described in Shoemake (1985).

4 Bernstein-Bézier Polynomial Motions

This section develops a deCasteljau type algorithm for Bézier motions which have Bernstein polynomial basis functions. Such Bernstein-Bézier type motions have been developed analytically in our companion paper (Ge and Ravani, 1994) and have the useful subdivision property.

From Ge and Ravani (1994), we know that a unifold image curve in the image space Σ may be expressed in terms of non-normalized coordinates (denoted by $\hat{\mathbf{W}}(t)$) or, equivalently, in terms of unit-normalized coordinates (denoted by $\hat{\mathbf{X}}(t)$). They are related by $\hat{\mathbf{W}}(t) = \hat{w}(t)\hat{\mathbf{X}}(t)$ where $\hat{w}(t)$ is a dual-number normalizing function. The normalizing function $\hat{w}(t)$ may be interpreted geometrically as a projection operator. A general curve $\hat{\mathbf{W}}(t)$ in the image space Σ does not necessarily represent a rigid body motion. It represents a motion (denoted by $\hat{\mathbf{X}}(t)$) when it has unit-normalized coordinates. We shall refer to $\hat{\mathbf{W}}(t)$ as the preimage of $\hat{\mathbf{X}}(t)$. This geometric interpretation is useful in explaining the subdivision property associated with Bézier polynomial image curves to be developed in this section.

The Bézier cubic curves in the image space Σ are given by (Ge and Ravani, 1994)

$$\hat{\mathbf{W}}(t)=\hat{w}(t)\hat{\mathbf{X}}(t)=[1\ t\ t^2\ t^3][M_b]\begin{bmatrix}\hat{w}_0\hat{\mathbf{b}}_0\\ \hat{w}_1\hat{\mathbf{b}}_1\\ \hat{w}_2\hat{\mathbf{b}}_2\\ \hat{w}_3\hat{\mathbf{b}}_3\end{bmatrix} \qquad (25)$$

where $\hat{\mathbf{b}}_i$ $(i=1, 2, 3, 4)$ denote the Bézier control points in Σ that represent the key configurations, and $\hat{w}_i$ $(i=1, 2, 3, 4)$ denote the dual-number weight factors. Eq. 925) can also be expressed in terms of Bernstein polynomials $B_i^3(t)$:

$$\hat{\mathbf{W}}(t)=\hat{w}(t)\hat{\mathbf{X}}(t)=\sum_{i=0}^{3}\hat{w}_i\hat{\mathbf{b}}_iB_i^3(t)$$

and is said to define *Bernstein-Bézier cubic motions*.

Similar to a Bernstein-Bézier curve in a Euclidean three-space E^3, a Bernstein-Bézier unifold image curve $\hat{\mathbf{X}}(t)$ given by Eq. (25) can be alternatively generated by repeated linear interpolation of its control image points. Given the control points $\hat{\mathbf{b}}_o$ and the weights $\hat{w}_i$ $(i=0, 1, \cdots, n)$, the deCasteljau algorithm for constructing a Bernstein-Bézier polynomial motion (of degree n) proceeds as follows:

(*1*) For $i=0, 1, \cdots, n$, set $\hat{\mathbf{W}}_i^0=\hat{w}_i\hat{\mathbf{b}}_i$;

(*2*) For $r=1, 2, \cdots, n$ and $i=0, 1, \cdots, n-r$, set

$$\hat{\mathbf{W}}_i^r=(1-t)\hat{\mathbf{W}}_i^{r-1}+t\hat{\mathbf{W}}_{i+1}^{r-1}; \qquad (26)$$

(*3*) Then $\hat{\mathbf{X}}(t)=\hat{\mathbf{W}}_0^n/|\hat{\mathbf{W}}_0^n|$ where $|\hat{\mathbf{W}}_0^n|$ denotes the magnitude of $\hat{\mathbf{W}}_0^n$.

In the above, each intermediate point $\hat{\mathbf{b}}_i^r$ is given by $\hat{\mathbf{W}}_i^r/|\hat{\mathbf{W}}_i^r|$.

This deCasteljau algorithm may be used to *subdivide* a Bernstein-Bézier image curve. The algorithm subdivides the preimage $\hat{\mathbf{W}}(t)$ of the image curve$\hat{\mathbf{X}}(t)$. The intermediate preimage points, $\hat{w}_i\hat{\mathbf{b}}_i^r$, may be projected onto the curved image space to provide us with the control polygons for the "left" and "right" curve segment.

One important drawback for the above Bernstein-Bézier motion is that it is difficult to adjust the speed of the motion intuitively, for it is very difficult (if not impossible) to select the constant weights $\hat{w}_i$ such that the linear motion of the form (26) has constant angular velocity or pitch (Ge and Ravani, 1994).

Conclusions

This paper has provided a theoretical foundation for construction of deCasteljau algorithms for motion approximation. It has shown that Bézier motions obtained by the application of deCasteljau's construction are not of Bernstein form and lack the subdivision property of Bernstein-Bézier curves used in CAGD. Certain differential and kinematic properties of the Bézier motions have also be studied. An alternative form of deCasteljau algorithm was presented that results in Bernstein-Bézier motions with subdivision property. The results have applications in motion animation, kinematics and CAD/CAM.

Acknowledgment

The first author gratefully acknowledges the support by National Science Foundation (NSF) Research Intiation Award MSS-9396265 to the State University of New York at Stony Brook. The second author gratefully acknowledges the support by NSF grant DMC-8796348 and the US Army Research Office grant DAAL03-90-0005 to the University of California at Davis.

References

Barsky, B. A., and DeRose, T. D., 1989, "Geometric Continuity of Parametric Curves: Three Equivalent Characterizations," *IEEE Computer Graphics and Applications*, Vol. 9, No. 6, pp. 60–68.

Boehm, W., 1987, "Smooth Curves and Surfaces," *Geometric Modeling: Algorithms and New Trends*, G. Farin, ed., SIAM, Philadelphia, PA, pp. 175–184.

Bottema, O., and Roth, B., 1979, *Theoretical Kinematics*, North Holland Publ., Amsterdam, pp. 51–60, 150–152, 521–523.

Duff, T., 1986, "Quaternion Splines for Animating Orientation," Technical Report, AT&T Bell Laboratories.

Farin, G., 1993, *Curves and Surfaces for CAGD: A Practical Guide*, 3rd ed., Academic Press, San Diego, 473 pp.

Flanders, H., 1963, *Differential Forms with Application to the Physical Sciences*, Academic Press, New York, 205 pp.

Ge, Q. J., and Ravani, B., 1994, "Computer Aided Geometric Design of Motion Interpolants," ASME Journal of Mechanical Design.

McCarthy, J. M., and Ravani, B., 1986, "Differential Kinematics of Spherical and Spatial Motions using Kinematic Mapping," ASME *Journal of Applied Mechanics*, Vol. 53, pp. 15–22.

McCarthy, J. M., 1990, *Introduction to Theoretical Kinematics*, MIT Press, Cambridge, MA, pp. 19–22, 60–63.

Pletinckx, D., 1989, *Quaternion Calculus as a Basic Tool in Computer Graphics," The Visual Computer*, Vol. 5, pp. 2–13.

Pottmann, H., 1992, Personal Communications.

Ravani, B., and Roth, B., 1984, "Mappings of Spatial Kinematics, ASME Journal of Mechanisms, Transmissions, and Automation in Design, Vol. 106, No. 3, pp. 341–347.

Shoemake, K., 1985, "Animating Rotation with Quaternion Curves," *ACM Siggraph*, Vol. 19, No. 3, pp. 245–254.

Sommerville, D. M. Y., 1914, *Elements of Non-Euclidean Geometry*, Dover Publ (reprint 1958).

Taylor, R. H., 1979, "Planning and Execution of Straight-Line Manipulator Trajectories," *IBM J. of Research and Development*, Vol. 23, No. 4, pp. 253–264.

APPENDIX A

This appendix proves that the deCasteljau algorithm for constructing the nonalgebraic Bézier image curves does not possess the subdivision property. For convenience, we consider only the construction of a nonalgebraic Bézier image curve of rank 2 using three control points.

Let $\hat{\mathbf{b}}_0$, $\hat{\mathbf{b}}_1$, $\hat{\mathbf{b}}_2$ be the Bézier control points in the image space Σ and let s, t be two parametes such that $0<s<t<1$. Then the deCasteljau algorithm yields two parametric points on the Bézier image curve (Fig. 9):

$$\hat{\mathbf{b}}_0^2(s)=\hat{\mathbf{b}}_0^1(s)[(\hat{\mathbf{b}}_0^1(s))^{-1}\hat{\mathbf{b}}_1^1(s)]^s, \qquad (27)$$

$$\hat{\mathbf{b}}_0^2(t)=\hat{\mathbf{b}}_0^1(t)[(\hat{\mathbf{b}}_0^1(t))^{-1}\hat{\mathbf{b}}_1^1(t)]^t. \qquad (28)$$

Let $\hat{\mathbf{c}}$ denote the intersection point of the two lines joining $\hat{\mathbf{b}}_0^1(s)$ to $\hat{\mathbf{b}}_1^1(s)$ and $\hat{\mathbf{b}}_0^1(t)$ to $\hat{\mathbf{b}}_1^1(t)$. If the subdivision property exists for the deCasteljau algorithm, then the point $\hat{\mathbf{b}}_0^2(s)$ can be alternatively generated by subdividing the triangle formed by the three points $\hat{\mathbf{b}}_0$, $\hat{\mathbf{b}}_0^1(t)$, and $\hat{\mathbf{b}}_0^2(t)$ in the ratio (s/t):1. This leads to the following

$$\hat{\mathbf{c}}=\hat{\mathbf{b}}_0^1(t)[(\hat{\mathbf{b}}_0^1(t))^{-1}\hat{\mathbf{b}}_1^2(t)]^{s/t}, \qquad (29)$$

$$\hat{\mathbf{b}}_0^2(s)=\hat{\mathbf{b}}_0^1(s)[(\hat{\mathbf{b}}_0^1(s))^{-1}\hat{\mathbf{c}}]^{s/t}. \qquad (30)$$

Substitute (28) into (29) to obtain

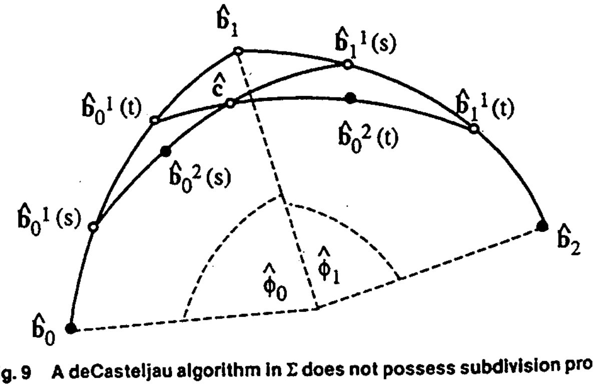


Fig. 9 A deCasteljau algorithm in Σ does not possess subdivision property

$$\hat{\mathbf{c}} = \hat{\mathbf{b}}_0^1(t)[(\hat{\mathbf{b}}_0^1(t))^{-1}\hat{\mathbf{b}}_1^1(t)]^s. \qquad (31)$$

From (27) and (30), we obtain

$$\hat{\mathbf{c}} = \hat{\mathbf{b}}_0^1(s)[(\hat{\mathbf{b}}_0^1(s))^{-1}\hat{\mathbf{b}}_1^1(s)]^t. \qquad (32)$$

In what follows we show that in general (31) and (32) cannot hold simultaneously and therefore conclude that the deCasteljau algorithm in Σ does not in general possess subdivision property. Let $\hat{\psi}_s$, $\hat{\psi}_t$ denote the dual angular lengths of the segments $\hat{\mathbf{b}}_0^1(s)\hat{\mathbf{b}}_1^1(s)$ and $\hat{\mathbf{b}}_0^1(t)\hat{\mathbf{b}}_1^1(t)$, respectively. Then Eqs. (31), (32) mean that $\hat{\mathbf{c}}$ divides the segment $\hat{\mathbf{b}}_0^1(t)\hat{\mathbf{b}}_1^1(t)$ in the ratio $\sin(s\hat{\psi}_t)/\sin((1-s)\hat{\psi}_t)$ and divides the segment $\hat{\mathbf{b}}_0^1(s)\hat{\mathbf{b}}_1^1(s)$ in the ratio $\sin(t\hat{\psi}_s)/\sin((1-t)\hat{\psi}_s)$, respectively. If this is true then the triangle $\hat{\mathbf{b}}_0^1(t)\hat{\mathbf{b}}_1\hat{\mathbf{b}}_1^1(t)$ and its transversal $\hat{\mathbf{b}}_0^1(s)\hat{\mathbf{b}}_1^1(s)$ must satisfy the Menalaus' theorem of collinearity:

$$\frac{\sin((1-s)\hat{\phi}_0)}{\sin((t-s)\hat{\phi}_0)} \cdot \frac{\sin(s\hat{\psi}_t)}{\sin((1-s)\hat{\psi}_t)} \cdot \frac{\sin((t-s)\hat{\phi}_1)}{\sin(s\hat{\phi}_1)} = 1, \qquad (33)$$

where $\hat{\phi}_0$, $\hat{\phi}_1$ are the angular lengths of the segments $\hat{\mathbf{b}}_0\hat{\mathbf{b}}_1$, $\hat{\mathbf{b}}_1\hat{\mathbf{b}}_2$, respectively (Details on Menelaus' theorem in non-Euclidean geometry can be found in Sommerville, 1914). In the special case when $\phi_0 = \phi_1 = 0$, i.e., when three given configurations share the same orientation, Eq. (33) holds and it reduces to:

$$\frac{1-s}{t-s} \cdot \frac{s}{1-s} \cdot \frac{t-s}{s} = 1,$$

by virtue of Eq. (9). In general, however, Eq. (33) does not hold. Thus we conclude that (31) and (32) do not hold simultaneously for arbitrary spaced control points $\hat{\mathbf{b}}_0$, $\hat{\mathbf{b}}_1$, and $\hat{\mathbf{b}}_2$. This completes the proof.

Bézier Curves on Riemannian Manifolds and Lie Groups with Kinematics Applications

F. C. Park
Assistant Professor.

B. Ravani
Professor.

Department of Mechanical and
Aeronautical Engineering,
University of California,
Davis, CA

In this article we generalize the concept of Bézier curves to curved spaces, and illustrate this generalization with an application in kinematics. We show how De Casteljau's algorithm for constructing Bézier curves can be extended in a natural way to Riemannian manifolds. We then consider a special class of Riemannian manifold, the Lie groups. Because of their group structure Lie groups admit an elegant, efficient recursive algorithm for constructing Bézier curves. Spatial displacements of a rigid body also form a Lie group, and can therefore be interpolated (in the Bézier sense) using this recursive algorithm. We apply this alogorithm to the kinematic problem of trajectory generation or motion interpolation for a moving rigid body. The orientation trajectory of motions generated in this way have the important property of being invariant with respect to choices of inertial and body-fixed reference frames.

1 Introduction

One of the cornerstones of geometric design has been the work of P. Bézier on the free-form curve design method that bears his name. Bézier curves are not only flexible and easy to generate, but offer a simple geometric representation of a curve in terms of its control polygon, as well as a firm mathematical foundation based on Bernstein polynomials. In applying the Bézier method to the kinematic problem of trajectory generation or motion interpolation for a moving rigid body, however, the classical geometric design techniques need to be extended to curved spaces. In principle one can obtain a collection of local coordinate charts for a given curved space, and apply existing Euclidean interpolation techniques to these coordinates. The resulting curves, however, will depend on the choice of local coordinates, which clearly leaves something to be desired from both a mathematical as well as an engineering perspective. Another requirement motivated by the moving rigid body problem is that, to the extent possible, the resulting motions should not depend on the choice of inertial or body-fixed reference frames; in the language of Lie groups this can be phrased as the question of whether a group admits a bi-invariant Riemannian metric. Using standard results from Lie theory it can be shown that bi-invariant orientation trajectories can be constructed, but that in general there is no bi-invariant metric for the spatial displacements (see, e.g., Park et al., 1993).

The goal of this article is to generalize the concept of Bézier curves to curved spaces. The existing theory of Bézier curves has been formulated only for curves in Euclidean space, and only recently have attempts been made at extending Bézier's original construction to particular curved spaces. Shoemake (1985) presents a class of methods for generating curves on rotations that are based on unit quaternion representation. Although unit quaternions have certain well-known advantages over other representations of rotations (e.g., Euler angles), Shoemake's approach is essentially coordinate dependent: the resulting motions are not invariant with respect to choice of inertial and body-fixed frames, and his methods do not adequately address the underlying geometry of the space of rotations (e.g., the 2-1 nature of the unit quaternion representation). In more recent work, Ge and Ravani (1991, 1994) have recognized these geometric issues, and formulated rigid body motion interpolation in terms of the image space manifold of kinematic mapping (see e.g., Bottema and Roth, 1979 or Ravani and Roth, 1984). In particular, the 2-1 ambiguity is resolved by defining an orientation on the space of unit quaternions, and trajectories are constructed in terms of the first fundamental form of the image space with bi-invariant rotation parts. Juettler (1994) has provided a theoretical evaluation of several approaches for motion interpolation and has discussed coordinate frame dependency of some of these approaches.

In this article, we formulate a general framework for constructing Bézier curves on Riemannian manifolds, and then focus specifically on a special class of Riemannian manifold, the compact *Lie groups*. This class covers a wide range of curved spaces that arise frequently in kinematics: for example, the group of proper rotations SO(n) lies at the heart of the rigid-body motion interpolation problem, and the special unitary group SU(2) can also be identified with unit quaternions.

This article is organized as follows. In Section 2 we show how Bézier curves can be generalized to arbitrary Riemannian manifolds, and discuss some of the computational issues. In

Contributed by the Mechanisms Committee for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received Jan. 1994; revised Sept. 1994. Associate Technical Editor: J. Angeles.

Section 3 we present a method of constructing Bézier curves on compact Lie groups, based on a generalization of De Casteljau's algorithm. In Section 4 we apply these curve generation techniques to the design of smooth trajectories on the rigid-body motions.

2 Bézier Curves on Riemannian Manifolds

2.1 A Geometric Interpretation of Classical Bézier Curves. Before laying down the mathematical framework for Riemannian manifolds, it is instructive to review the geometric interpretation of Bézier curves from the perspective of both Bézier's original construction and of De Casteljau's algorithm (1963), which, incidentally, precedes Bézier's work. Bézier (1963) geometrically defines a curve from four ordered vertexes—the polygon formed by these vertexes is called the *control polygon*—subject to a set of geometric constraints. Specifically, let the ordered vertices be $\{(0, 0, 0), (1, 0, 0), (1, 0, 1), (1, 1, 1)\}$, and let $\mathcal{C}: [0, 1] \rightarrow \mathcal{R}^3$ be a curve such that

- $\mathcal{C}(0) = (0, 0, 0)$ and $\mathcal{C}(1) = (1, 1, 1)$.
- The tangent at $\mathcal{C}(0)$ is parallel to the x-axis, and the tangent at $\mathcal{C}(1)$ is parallel to the z-axis.
- The osculating plane at $\mathcal{C}(0)$ is parallel to the x-y plane, and the osculating plane at $\mathcal{C}(1)$ is parallel to the y-z plane.

The curve satisfying these constraints is given by a cubic polynomial. Reversing the order of the vertexes results in the same curve. It is important to note that the resulting curve is determined entirely from geometric constraints, and that the method of construction can be extended naturally to an arbitrary number of vertexes arranged randomly in Euclidean space. The Bézier curve can also be viewed as an approximation to the control polygon in terms of Bernstein polynomials (as discovered later by R. Forrest, 1972).

De Casteljau presents another method of constructing a smooth curve given a control polygon, which turns out to be identical to the Bézier curve, but directly exploits the relationship with Berstein polynomials. This algorithm is perhaps best illustrated by Fig. 1. Specifically, let the ordered set of vertices in $\mathcal{R}^3$ be given by $\{p_0, p_1, \ldots, p_n\}$, and define the polynomials

$$p_i^k(t) = (1-t)p_{i-1}^{k-1} + tp_i^{k-1}$$

where $p_i^0 = p_i$. The curve given by $p_n^n(t)$, $0 \le t \le 1$ then corresponds to the Bézier curve. Geometrically De Casteljau's algorithm constructs the curve by successive linear interpolation between the vertexes of the control polygon.

That Bézier's original construction and De Casteljau's algorithm are equivalent is remarkable, and can fundamentally be traced to the fact that the curve lies in Euclidean space. The polynomial representation of these curves, which makes them computationally attractive, is also due to the underlying space being Euclidean. In order to generalize these two methods of curve construction to curved spaces one must first generalize the underlying geometric concepts. In the De Casteljau method the concept of linear interpolation between two points in a curved space needs to be defined; this can be readily done on a Riemannian manifold, where the *minimal geodesic* plays the role of the straight line for curved spaces, and lengths can be measured in terms of the Riemannian metric. Bézier's construction, however, does not seem to generalize in a natural way to the Riemannian setting. Although tangency between curves is well-defined, the notion of an osculating plane relies inherently on the manifold being embedded in some larger ambient Euclidean space, and in general there is no natural way to do this. It is also more desirable to define a Bézier curve in terms of the intrinsic geometry of the manifold, rather than the underlying space in which it lies. For Riemannian manifolds, therefore, the natural way to define Bézier curves is by generalizing De Casteljau's algorithm. Naturally for certain manifolds the minimal geodesic between two points may not always be unique, so that a number of subtleties (addressed below) will arise.

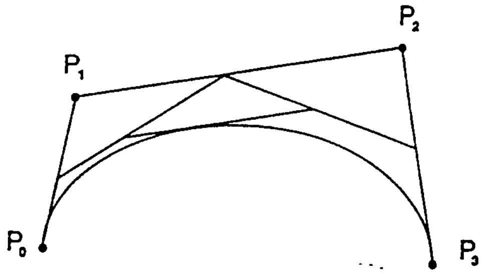


Fig. 1 The De Casteljau algorithm for $N = 4$

Before proceeding some clarification on what we mean by "geometric" may help justify our emphasis on the so-called "geometric" construction of Bézier curves. In principle any differentiable manifold of dimension n can be locally represented by a set of coordinates $(x_1, \ldots, x_n)$. One might therefore be tempted to simply construct Bézier curves in terms of these local coordinates, and regard the corresponding curve on the manifold as the generalized Bézier curve. The flaw with this construction, of course, is that the resulting curves depend on the choice of local coordinates. Any geometric scheme for generalizing Bézier curves must by definition be coordinate-invariant. For Lie groups we shall impose an additional requirement of bi-invariance.

2.2 Bézier Curves on Riemannian Manifolds. Clearly the key to generalizing De Casteljau's algorithm to curved spaces is the notion of linear interpolation, or even more fundamentally, lines. On a Riemannian manifold the geodesics (with respect to the given Riemannian metric) play the role of lines. We now formally review these concepts.

Let M be a Riemannian manifold of dimension n, with local coordinates $(x_1, x_2, \ldots, x_n)$, and Riemannian metric $ds^2 = g_{ij}(x)dx_idx_j$ (we adopt the physicists' convention of summation over repeated indices). If a curve $\mathcal{C}$ on M is given in local coordinates by $x(t)$, $0 \le t \le 1$, then the length of the curve is given by the integral

$$L = \int_0^1 \left(g_{ij}(x) \frac{dx_i}{dt} \frac{dx_j}{dt} \right)^{1/2} dt$$

Just as a line in Euclidean space can be considered as the shortest path between two points, on a Riemannian manifold the minimum length curve joining two points can be regarded as the analog of the straight line. A parametrized curve $\mathcal{C}$ is a *geodesic* if it is a critical point of the *energy* functional

$$E = \int_0^1 g_{ij}(x) \frac{dx_i}{dt} \frac{dx_j}{dt} dt$$

and is a *minimal geodesic* if it minimizes E. Observe that the integrands of E and L differ by a square: the length L is invariant with respect to reparametrizations of $\mathcal{C}$, whereas E clearly depends on the parametrization. Interestingly, the curves minimizing E also minimize L, and moreover are parametrized with respect to arc-length. In local coordinates the geodesics satisfy the system of differential equations

$$\frac{d^2x_i}{dt^2} + \sum_{j,k} \Gamma^i_{jk}(x(t)) \frac{dx_k}{dt} \frac{dx_j}{dt} = 0 \qquad (1)$$

for $1 \le i \le n$, where

$$\Gamma^i_{jk} = \frac{1}{2} \sum_{l=1}^{n} g^{il} \left(\frac{\partial g_{kl}}{\partial x_j} + \frac{\partial g_{lj}}{\partial x_k} - \frac{\partial g_{jk}}{\partial x_l} \right)$$

and $(g^{il}) = (g_{il})^{-1}$. The Γ^i_{jk} are known as the *Christoffel symbols of the second kind.*

Local existence and uniqueness of solutions for the Euler-Lagrange equations of geodesics can be shown for a given set of initial conditions (see, e.g., Gallot et al., 1990). This result implies that any two points which are "close enough" to each other are joined by a unique minimal length geodesic. This property is generally not global; for example, the geodesics on the two-sphere S^2 are given by the great circles, and any two antipodal points will have any number of minimal geodesics. If any geodesic on M can be extended to a geodesic defined on all of $\mathcal{R}$, then the manifold is said to be *geodesically complete.* A theorem of Hopf-Rinow asserts that if a manifold is geodesically complete, then any two points of M can be joined by a minimal geodesic; note that the theorem does not claim that this geodesic is unique. For the purposes of this paper we shall only consider geodesically complete manifolds. The interested reader is referred to Gallot et al. (1990) for the technical requirements of such manifolds.

Having established the minimal geodesics as the Riemannian analog of straight lines, Bézier curves can now be easily generalized via De Casteljau's algorithm. Let the n ordered points of M forming the control polygon be labelled $\{p_0, p_1, \ldots, p_n\}$. Represent the minimal geodesic between any two points $p, q \in M$ by the curve $\exp(p, q, t)$, $0 \leq t \leq 1$, where $\exp(p, q, 0) = p$ and $\exp(p, q, 1) = q$. Define the sequence of curves on M

$$p_i^k(t) = \exp(p_{i-1}^{k-1}\ (t),\ p_i^{k-1}\ (t),\ t)$$

Here $p_i^0(t) = p_i$. The Bézier curve is then given by

$$p_n^n(t) = \exp(p_{n-1}^{n-1}(t),\ p_n^{n-1}(t),\ t)$$

which is analogous to the De Casteljau algorithm for Euclidean space.

It is clear that constructing Bézier curves on Riemannian manifolds by this algorithm is computationally more involved than for the Euclidean case: computing the geodesic between any two points involves the solution of the nonlinear differential equation ([1]), a two-point boundary value problem (and therefore more difficult than integrating a differential equation with only initial conditions). Even if we assume that the geodesics forming the control polygon have been precomputed and stored in a table, for each instant t the geodesic equations still need to be solved $(n - 1)(n - 2)/2$ times. Clearly this presents difficulties for interactive design applications.

One method of obtaining approximate Bézier curves in real-time is to compute only a discrete set of points on the curve, and to interpolate between these points using local coordinates. Specifically, the Bézier curve $p_n^n(t)$ can be sampled N times at uniformly-spaced intervals of t using the algorithm given above. Labelling these points $P(i)$, $i = 1, \ldots, N$, one can then obtain smooth interpolants between adjacent points $P(i - 1)$ and $P(i)$ in terms of local coordinates. While any convenient interpolating spline can be chosen, care must be taken to ensure continuity of the proper order at the knot points. The obvious drawback of this approach, of course, is that the resulting curve depends on the choice of local coordinates. Nevertheless, it is a practical means of designing curves on Riemannian manifolds in real-time that are nearly coordinate-invariant.

3 Bézier Curves on Lie Groups

3.1 Lie Groups. We now specialize to a special class of Riemannian manifold, the matrix Lie groups. A Lie group[1] **G** is a differentiable manifold and an algebraic group whose operation $(x, y) - xy^{-1}$ is smooth. Some well-known examples of Lie groups include Gl(n), the general linear group of $n \times n$ real nonsingular matrices, and Sl(n), the special linear group of $n \times n$ real nonsingular matrices of unit determinant.

Let p be a point on a matrix Lie group **G**, and $X(t)$ a smooth curve on **G** defined over some open interval of 0 such that $X(0) = p$. The derivative $\dot{X}(0)$ is said to be a *tangent* vector to **G** at p; the set of all tangent vectors at p, denoted $T_p\mathbf{G}$, forms a vector space, called the *tangent* space to **G** at p. The tangent space at the identity $p = I$ is given a special name, called the *Lie algebra* of **G**, and denoted by a lower-case g. On a matrix Lie group the Lie algebra is also given by matrices. For example, the Lie algebra of SO(3), denoted so(3), is the set of 3×3 real skew-symmetric matrices (see, e.g., Belinfante and Kolman, 1972).

More generally a Lie algebra is a vector space, **V**, together with a bilinear map $[\cdot,\cdot]$: $\mathbf{V} \times \mathbf{V} - \mathbf{V}$ (called the *Lie bracket*) that satisfies, for any $X, Y, Z \in \mathbf{V}$, (*i*) $[X, X] = 0$, and (*ii*) $[X, [Y, Z]] + [Z, [X, Y]] + [Y, [Z, X]] = 0$. For matrix Lie algebras the Lie bracket is given by the matrix commutator: if $X, Y \in g$ are square matrices, then $[X, Y] = XY - YX$.

Defined on each Lie algebra is the exponential mapping into the corresponding Lie group. On matrix groups the exponential mapping corresponds to the usual matrix exponential, i.e., if $A \in g$, then $\exp A = I + A + A^2/2! + \ldots$ is an element of **G**. Observe that e^{At}, $t \in \mathcal{R}$, itself forms a group, in this case a subgroup of the Lie group. Such groups are called *one-parameter subgroups* of a Lie group, and play a special role in the description of minimal-length paths on Lie groups as we show below. Before lengths of paths can be defined we first need to examine Riemannian metrics on Lie groups.

If $X(t)$ is a differentiable curve on **G** as before, then $\dot{X} \in T_p\mathbf{G}$. One can exploit the group structure of **G** to represent X as an element of the Lie algebra as follows. Let $g, h \in \mathbf{G}$, and define the *left* and *right translation* maps $L_{h^{-1}}$: $\mathbf{G} - \mathbf{G}$, $R_{h^{-1}}$: $\mathbf{G} - \mathbf{G}$ by $L_{h^{-1}}(g) = h^{-1}g$ and $R_{h^{-1}}(g) = gh^{-1}$, respectively. It follows that their derivatives $dL_{h^{-1}}$ and $dR_{h^{-1}}$ are mappings from $T_{h^{-1}}\mathbf{G}$ to $T_e\mathbf{G} = g$. By applying these two maps to $\dot{X}$ it can be seen that $X^{-1}\dot{X}$ and $\dot{X}X^{-1}$ are elements of g. Any tangent vector can therefore be identified with an element of g by either left or right translation. Since g is a vector space, any inner product on g will define a Riemannian metric for **G**.

Let Q be a quadratic form on g, and $X(t)$ a curve on **G**. Then $\dot{X} \in T_X\mathbf{G}$, and $X^{-1}\dot{X} \triangleq V_L$ and $\dot{X}X^{-1} \triangleq V_R$ are elements of g. Observe that V_R and V_L are related by $V_R = XV_LX^{-1} \triangleq \mathrm{Ad}_X(\mathrm{V}_L)$. Q defines an inner product on g, which in turn defines two classes of Riemannian metric on **G**:

$$\langle \dot{X}, \dot{X} \rangle_L \triangleq \frac{1}{2} V_L^T Q V_L$$

$$\langle \dot{X}, \dot{X} \rangle_R \triangleq \frac{1}{2} V_R^T Q V_R$$

$\langle \cdot,\cdot \rangle_L$ and $\langle \cdot,\cdot \rangle_R$ are the *left-* and *right-invariant* Riemannian metrics on **G** defined by Q. If $\langle\cdot,\cdot\rangle_L = \langle\cdot,\cdot\rangle_R$ the metric is said to be *bi-invariant*. Any Lie group admits a left- or right-invariant metric from the construction above, but not all Lie groups admit a bi-invariant metric. One well-known condition in which a bi-invariant metric is always guaranteed to exist is if the Lie group is compact. In this case the geodesics of **G** (with respect to the bi-invariant metric) are the one-parameter subgroups of **G** and its translates, i.e., if $A \in g$ and $t \in \mathcal{R}$, then e^{At} is a geodesic as is He^{At} and $e^{At}H$ for any $H \in \mathbf{G}$.

On compact **G** it is known that any $g \in \mathbf{G}$ lies on a one-parameter subgroup. Hence, given any two points in **G** there always exists a geodesic (with respect to the bi-invariant metric) connecting them. To find the minimal geodesic we must consider the inverse of the exponential map exp: $g - \mathbf{G}$. If **G** is compact then it is well-known that exp is onto, but typically

[1] For a comprehensive account of applications of Lie groups in kinematics the reader is referred to Karger and Novak (1985).

its inverse map will be multiple-valued. We therefore define log: $\mathbf{G}-g$ by

$$\log(A)=a$$

such that a^TQa is minimal among all possible $a \in g$ satisfying $\exp(a) = A$. The minimal geodesic between A and B is then given by $X(t) = A\exp(\Omega t)$, $0 \le t \le 1$, where $\Omega = \log(A^{-1}B)$. Moreover, if A and B are left-translated by some constant C to CA and CB, then the minimal geodesic is given by $CX(t)$. Similarly, the minimal geodesic between AB and AC is $X(t)C$. These properties follow from the bi-invariance of the Riemannian metric, and can be verified by an elementary calculation.

3.2 The De Casteljau Algorithm on Compact Lie Groups. Having established that the geodesics on compact Lie groups (with respect to the bi-invariant Riemannian metric) are the one-parameter subgroups and its translates, we now describe the algorithm for generating Bézier curves on such spaces. Let $\mathbf{G}$ be the Lie group with Lie algebra g, exp: $g-\mathbf{G}$ the exponential map, and log: $\mathbf{G}-g$ the inverse map that provides the minimal norm value as described above. Let $\{p_0, p_1, \ldots, p_n\}$ be the ordered set of points in $\mathbf{G}$ that form the control polygon. The Bézier curve is now defined recursively as before:

$$p_i^k(t)=p_{i-1}^{k-1}(t)\exp(t\log[(p_{i-1}^{k-1})(t)]^{-1}p_i^{k-1}(t)),\ p_i^0(t)=p_i$$

and the Bézier curve is given by

$$p_n^n(t)=p_{n-1}^{n-1}(t)\exp(t\log[(p_{n-1}^{n-1})(t)]^{-1}p_n^{n-1}(t))$$

We now restrict our attention to the rotation group SO(3) in the next section.

4 Kinematics Application: Bézier Curves on SO(3)

The rotation group SO(3), consisting of the 3×3 real orthogonal matrices with unit determinant, forms a Lie group, with its Lie algebra so(3) given by the vector space of 3×3 real skew-symmetric matrices of the form

$$[\omega] \triangleq \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}$$

The following explicit formulas for the exponential and logarithm mappings on SO(3) and so(3) are well-known:

Lemma 1 *Given $\beta \in so(3)$,*

$$\exp[\omega]=I+\frac{\sin\|\omega\|}{\|\omega\|}\cdot[\omega]+\frac{1-\cos\|\omega\|}{\|\omega\|^2}\cdot[\omega]^2$$

where $[\omega]$ is the skew-symmetric matrix representation, and $\|\omega\|$ the standard Euclidean norm.

Lemma 2 *Given $\Theta \in SO(3)$ such that* $\mathrm{Tr}(\Theta)\neq -1$. *Then*

$$\log\Theta=\frac{\phi}{2\sin\phi}(\Theta-\Theta^T)$$

where ϕ satisfies $1 + 2\cos\phi = \mathrm{Tr}(\theta)$, $|\phi| < \pi$. Furthermore, $\|\log\Theta\|^2 = \phi^2$.

Remark 1 When $\mathrm{Tr}(\Theta) = -1$, $\log\Theta$ can have two possible values on the closed ball of radius π; if $\hat{\omega}$ is a unit length eigenvector of Θ associated with the eigenvalue 1, then a simple calculation shows that $\log\Theta = \pm\pi[\hat{\omega}]$.

Remark 2 Lemmas 1 and 2 suggest the standard visualization of SO(3) as a solid ball of radius π, centered at the origin with the antipodal points identified; a point ω in the ball represents a rotation by an angle $\|\omega\|$ about the line passing from the origin through ω. The rotations whose traces equal -1 have a rotation angle of π, and correspond to points on the boundary of the solid ball.

Remark 3 It is clear from above that the logarithm on SO(3) is multiple-valued: given $\Theta \in SO(3)$ such that $[\omega] = \log\Theta$, $\|\omega\| \le \pi$, then the set of all possible values of $\log\Theta$ is

$$[\omega]+2\pi n[\hat{\omega}],\ n\in\mathbf{Z}$$

where $\hat{\omega} = \omega/\|\omega\|$. This is akin to the situation in the complex plane, where if $e^{i\phi}$ is a point on the unit circle for some $0 \le \phi \le 2\pi$, then $e^{i(\phi+2\pi n)}$ corresponds to the same point for any integer n.

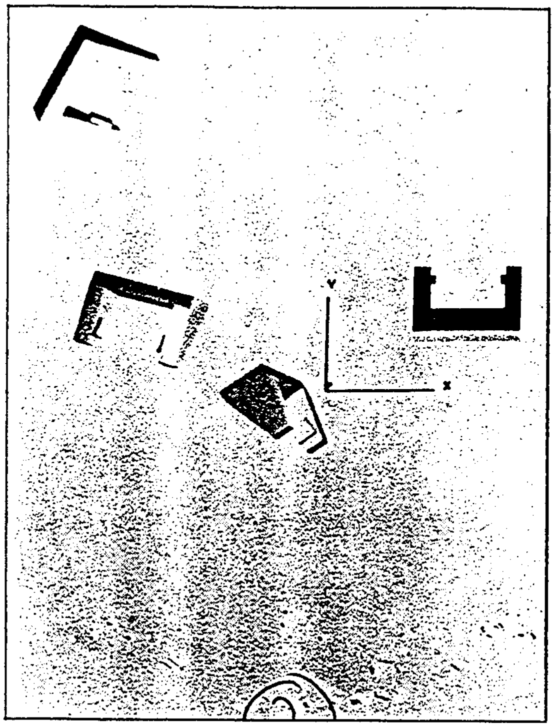


Fig. 2 Four control configurations of an end-effector

SO(3) is a compact Lie group and as such admits a bi-invariant Riemannian metric $\langle\cdot,\cdot\rangle$. To see how the metric is applied, let $\Theta(t)$ be a curve on SO(3), and $\Theta^{-1}\dot{\Theta} = [\omega_b]$, $\dot{\Theta}\Theta^{-1} = [\omega_s]$. Then $\langle\dot{\Theta}, \dot{\Theta}\rangle = c\omega_b^T\omega_b = c\omega_s^T\omega_s$, where c is a positive constant scale factor; the Riemannian metric is therefore determined by the quadratic form $Q = cI$ on so(3). For convenience we shall henceforth set $c = 1$.

In terms of the bi-invariant metric the geodesics on SO(3) are given by the translates of the one-parameter subgroups e^{At}, $A \in$ so(3) and $t \in \Re$. The *minimal* geodesic between two elements $\Theta_1, \Theta_2 \in$ SO(3) is given by the curve

$$\Theta(t)=\Theta_1 e^{\Omega_{12}t},\quad 0\le t\le 1$$

where $\Omega_{12} = \log(\Theta_1^{-1}\Theta_2)$, or, equivalently,

$$\Theta(t)=e^{\Omega_{21}t}\Theta_1,\quad 0\le t\le 1$$

where $\Omega_{21} = \log(\Theta_2\Theta_1^{-1})$.

With this simple characterization of the minimal geodesics, Bézier curves can now be constructed in a straightforward manner on SO(3). For example, with 3 control points Θ_0, Θ_1, and Θ_2, the Bézier curve is given by

$$\Theta(t)=\Theta_0 e^{\Omega_{01}t}e^{(\log(e^{-\Omega_{01}t}\Theta_0^{-1}\Theta_1 e^{\Omega_{12}t}))t}$$

for $0 \le t \le 1$, where $\Omega_{01} = \log(\Theta_0^{-1}\Theta_1)$ and $\Omega_{12} = \log(\Theta_1^{-1}\Theta_2)$.

The general case with n control points is as follows. Let Θ_0^0, $\Theta_1^0, \ldots, \Theta_n^0$ be the ordered set of control points, and define

$$\Theta_i^k(t)=\Theta_i^{k-1}(t)e^{(\log((\Theta_i^{k-1})^{-1}\Theta_{i+1}^{k-1}))t}$$

where k ranges from 1 to n, and i from 0 to $n - k$. The Bézier curve is then given by $\Theta_0^n(t)$.

Remark 4 The above construction can also be applied to design Bézier curves in SE(3), where the one-parameter subgroups e^{At} are now the screw motions. It is well-known,

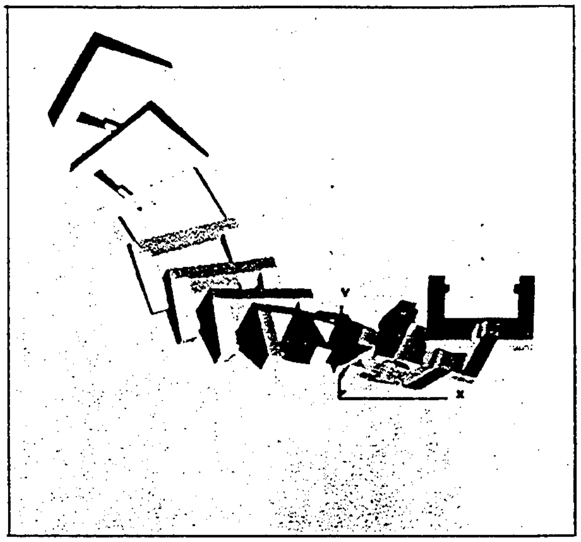


Fig. 3 A continuous motion interpolating the control configurations

however, that SE(3) does not admit a bi-invariant Riemannian metric (see, e.g., Duffy, 1990), and that the one-parameter subgroups on SE(3) are no longer geodesics with respect to any left- or right-invariant Riemannian metric. Rather, in this case the geodesics on SE(3) are simply the projections of the geodesics on $\mathcal{R}^3 \times SO(3)$ (Park, Murray, and McCarthy, 1993). Hence, given a particular left- or right-invariant Riemannian metric on SE(3), the corresponding Bézier curve can be constructed by combining the appropriate Bézier curves in $\mathcal{R}^3$ and SO(3). From a physical viewpoint this is more appealing, since there is nothing natural about the screw motions from the point of view of dynamics. In fact, in the absence of external forces one would expect the motion of the center of mass of a rigid body to be linear, while the orientation is governed by Euler's equations.

4.1 Example. The Cartesian space trajectory of a robot is to be designed such that it interpolates between the two end configurations of the end-effector shown in Fig. 2. Four control configurations are used (as shown in Fig. 2) and the trajectory of the end-effector is generated using the Bézier curves in $\mathcal{R}^3$ and SO(3) as discussed above. The resulting motion or trajectory of the end-effector is shown in Fig. 3.

5 Conclusion

By generalizing the notion of straight lines to curved spaces, Bézier curves can be defined on Riemannian manifolds by a suitable generalization of De Casteljau's algorithm. In the case when the Riemannian manifold is a compact Lie group appealing formulas exist for the minimal geodesics, which are given by matrix exponentials. The algorithm has been illustrated for the particular case of SO(3), with explicit formulas given for the matrix exponential and logarithm. The resulting orientation trajectories are invariant with respect to the choice of inertial or body-fixed reference frames for the rigid body. These results have direct applications to kinematics and animation of rigid body motions, as well as to any problem in which the physical aspects are described by a Lie group.

Acknowledgment

This research was supported in part by NSF grant DMC-8796348 and the US Army Research Office grant DAAL03-90-0005 to the University of California at Davis.

References

1 Belinfante, J. G., and Kolman, B., 1972, *A Survey of Lie Groups and Lie Algebras with Applications and Computational Methods*, Siam, Philadelphia.

2 Bézier, P., 1986, *The Mathematical Basis of the UNISURF CAD System*, Butterworths, London.

3 Bottema, O., and Roth, B., 1987, *Theoretical Kinematics*, Dover, New York.

4 Chevalley, C., 1946, *Theory of Lie Groups*, Princeton University Press.

5 De Casteljau, F., 1963, "Outillage méthodes calcul.," *André Citroēn Automobiles SA*, Paris.

6 Duffy, J., 1990, "The Fallacy of Modern Hybrid Control Theory That is Based on 'Orthogonal Complements' of Twist and Wrench Spaces," *Journal of Robotic Systems*, Vol. 7, No. 2.

7 Forrest, R., 1972, "Interactive Interpolation and Approximation by Bézier Polynomials," *Computer J.*, Vol. 15, pp. 71–79.

8 Gallot, S., Hulin, D., and Lafontaine, J., 1990, *Riemannian Geometry*, Springer-Verlag, Berlin.

9 Ge, Q. J., and Ravani, B., 1991, "Computer Aided Geometric Design of Motion Interpolants," *Proceedings of ASME Design Automation Conference*, Vol. 2, pp. 33–41.

10 Ge, Q. J., and Ravani, B., 1994, "Geometric Construction of Bézier Motions," ASME Journal of Mechanical Design, September.

11 Juettler, B., 1994, "Visualization of Moving Objects Using Dual Quaternion," *Curves, Computers and Graphics*, Vol. 18, No. 3.

12 Karger, A., and Novak, J., 1985, *Space Kinematics and Lie Groups*, Gordon and Broach.

13 Park, F. C., Murray, A. P., and McCarthy, J. M., 1993, "Designing Mechanisms for Workspace Fit," *Computational Kinematics*, J. Angeles, G. Hommel, and P. Kovacs, eds., Kluwer, Amsterdam.

14 Ravani, B., and Roth, B., 1984, "Mappings of Spatial Kinematics," ASME Journal of Mechanisms, Transmissions, and Automation in Design, Vol. 106, pp. 341–347, September.

Smooth Interpolation of Rotations

Frank C. Park
Assistant Professor
Mechanical and Aerospace Engineering Dept.
University of California, Irvine
Irvine, CA 92717

Bahram Ravani
Professor
Mechanical and Aeronautical Engineering Dept.
University of California, Davis
Davis, CA 95616

Abstract

This paper presents a variational method for smooth interpolation of the rotation group SO(3). By viewing SO(3) as a Lie group equipped with a natural Riemannian metric, we apply the coordinate-invariant methods of Riemannian geometry to construct spline curves that are invariant with respect to both left and right translations and generate curves that approximately minimize a measure of smoothness. Such spline curves, in the context of moving rigid bodies, generate orientation trajectories that are independent of the choice of inertial or body-fixed reference frames and are smooth. Based on this construction, a computationally efficient algorithm for interpolating multiple points in SO(3) is presented.

1 Introduction

One of the frequently encountered problems in applications ranging from computer graphics and animation to robot trajectory planning is the interpolation of smooth curves in SO(3), the space of rotations. In this paper we address the following problem: given an ordered set of n rotation matrices $\{R_1, R_2, \ldots, R_n\}$ (the *knot points*), and a set of n scalars $t_1 <$

$t_2 < \cdots < t_n$ (the *knot times*), find a $\mathcal{C}^2$ curve $R : [t_1, t_n] \rightarrow \mathrm{SO}(3)$ such that $R(t_i) = R_i$, $i = 1, 2, \ldots, n$. Our goal is to find a computationally efficient method of spline interpolation in SO(3) that produces reasonably smooth curves.

Well-established techniques exist for spline interpolation in vector spaces, but for the most part these techniques have yet to be generalized in a uniform, coordinate-free way to curved spaces like SO(3). Cubic splines, for example, are useful because of their simplicity and computational efficiency, and are further justifed by the physical argument that they minimize the smoothness criterion

$$\int \|\ddot{x}\|^2 dt \tag{1}$$

To generalize this approach to curved spaces, one might begin by formulating the corresponding second-order smoothness measure; more precisely, the curved space operator that plays the role of the second derivative needs to be defined. In principle one could choose a set of local coordinates for the space and, proceeding exactly as in Euclidean space, construct cubic splines in terms of these coordinates. However, curves generated in this fashion are known to depend on the choice of coordinates, so that this method is geometrically ill-defined. An even more subtle issue that arises in the SO(3) case is the question of *translation invariance*: given two ordered sets $\{R_1, \ldots, R_n\}$ and $\{\bar{R}_1, \ldots, \bar{R}_n\}$ in SO(3), where R_i and $\bar{R}_i$ are related by $\bar{R}_i = QR_iS$ for Q, S constant SO(3) matrices, it is reasonable to demand that the interpolating curves through these two sets, denoted $R(t)$ and $\bar{R}(t)$, respectively, be related by $\bar{R}(t) = QR(t)S$. Physically this reflects the fact that the choice of inertial and body-fixed reference frames for a rigid body should not influence the orientation trajectory of the interpolated motion.

In this paper we present a variational method for smooth interpolation in SO(3) that is both left and right translation invariant (or *bi-invariant*), and generates curves that approximately minimize a certain geometrically-defined measure of smoothness. In our approach SO(3) is regarded as a Lie group with a natural Riemannian metric, and the role of the second-order derivative operator is played by the symmetric covariant derivative compatible with this metric. On Riemannian manifolds this covariant derivative is the natural coordinate-invariant generalization of second-order derivatives in Euclidean space, and in SO(3) the corresponding second-order smoothness functional turns out to be simply the integral of the squared-norm of the angular acceleration. We review the specific form of the functional and its associated Euler-Lagrange equations for SO(3). In particular, our analysis relies on using the *canonical coordinates of the first kind* as local coordinates on SO(3). Because solving the Euler-Lagrange equations is impractical for interactive CAD applications, a method based on the canonical coordinates is presented that preserves bi-invariance, yet produces simple solutions that approximately minimize the second-order smoothness functional. These curves, as will be seen, properly reduce to the minimal geodesics under appropriate boundary conditions.

The paper is organized as follows. In Section 2 we review the geometry of SO(3) as a Lie group, and derive the corresponding canonical coordinates of the first kind. In Section 3 we express the second-order smoothness functional in terms of the canonical coordinates, and show that under some mild assumptions, the optimal curves are given by the exponential of a cubic matrix polynomial. To illustrate the efficiency of this approach the case of two point interpolation is considered in some detail. In Section 4 the two point interpolation results are extended to an algorithm for multiple point interpolation. We conclude with some remarks on how the interpolation of rigid body motions might be addressed within the given geometric framework.

Before proceeding we mention some of the relevant previous work in SO(3) interpolation. One of the more widely cited approaches is the work of Shoemake (1985), who presents a class of interpolation schemes based on the unit quaternion representation for rotations. While the unit quaternions are known to provide a globally nonsingular four-parameter representation for rotations, Shoemake's algorithm essentially applies existing Euclidean interpolation techniques to this particular set of coordinates, so that the resulting curves will not in general be bi-invariant and the interpolated motions are not necessarily Euclidean. Also, some of the expected characteristics are not preserved in the resulting Bézier curves. A more careful geometric analysis of Quaternion curves is given by Ge and Ravani (1994a, 1994b), in which the underlying curved geometry of the space of quaternions was considered in performing the interpolation and actual Euclidean motions were generated with a proper analysis and evaluation of the characteristics of the resulting Bézier representations. Jutler (1994) has presented a similar investigation working with dual quaternion curves and discussing some of the issues associated with the dependence of the existing methods on coordinate system used. Hart, Francis and Kauffman (1994) have presented an interesting method for visualization of quaternion curves representing three dimensional rotations.

The unit quaternions, in fact, can be identified with the Lie group SU(2) of the 2 × 2 special unitary matrices, so that the methods described here can be extended in an straight forward manner to SU(2). The present circle of geometric ideas has also been applied to formulate Bézier curves on SO(3) and general compact Lie groups (Park and Ravani 1995). Other relevant work in motion interpolation include the work of Barr (1993), Wagner and Jutler (1994), Pottmann and Wagner (1993).

None of these authors, however, have considered the problem of motion design in the general frame work of minimizing a certain measure of smoothness on SO(3). Furthermore, they have not utilized Riemanian geometry to deal with the inherent curved nature of the underlying space of three dimensional rotations producing interpolated motions that are completely coordinate independent.

This paper builds upon the formulation presented in Park and Ravani (1995); but develops a variational method for design of cubic splines for interpolating multiple points in SO(3) generating smooth multi-segment rotations which are independent of the choice of the coordinate system.

2 The Geometry of SO(3)

We begin with a review of the necessary background on SO(3) as a matrix Lie group; the development closely parallels that of (Park and Ravani 1994), and additional background can be found in, *e.g.*, (Belinfante and Kolman 1972).

SO(3) as a Lie Group

Recall that SO(3) is the set of all 3 × 3 real orthogonal matrices with unit determinant. SO(3) has the structure of a group and a differentiable manifold, and is an example of a *Lie group*. The rigid-body motions SE(3) can also be regarded as a Lie group under matrix multiplication, with elements of the form

$$\begin{bmatrix} R & b \\ 0 & 1 \end{bmatrix} \tag{2}$$

where $R \in \mathrm{SO}(3)$ and $b \in \Re^3$. Some other well-known examples of matrix Lie groups include Gl(n), the general linear group of $n \times n$ real nonsingular matrices, and the special linear group Sl(n), which is a subgroup of Gl(n) whose elements have unit determinant.

More generally let $\mathbf{G}$ denote a matrix Lie group, and let $X(t)$ be a differentiable curve in $\mathbf{G}$ defined over some open interval containing 0 such that $X(0) = p$. The derivative $\dot{X}(0)$ is said to be a *tangent vector* to $\mathbf{G}$ at p; the set of all tangent vectors at p, denoted $\mathrm{T}_p\mathbf{G}$, forms a vector space, called the *tangent space to* $\mathbf{G}$ *at* p. The tangent space at the identity $p = I$ is given a special name, called the *Lie algebra* of $\mathbf{G}$, and denoted by the lower-case $\mathbf{g}$. On SO(3) it is easily seen that the Lie algebra so(3) consists of the 3×3 skew-symmetric matrices: if $R(t)$ is a curve in SO(3) such that $R(0) = I$, then differentiating both sides of $R^T(t)R(t) = I$, it follows that $\dot{R}^T(0) + \dot{R}(0) = 0$, so that elements of so(3) are matrices of the form

$$[r] \triangleq \begin{bmatrix} 0 & -r_3 & r_2 \\ r_3 & 0 & -r_1 \\ -r_2 & r_1 & 0 \end{bmatrix} \tag{3}$$

where $r \in \Re^3$. Note that an element $[r] \in \mathrm{so}(3)$ can also be represented as a vector $r \in \Re^3$; if it is clear from the context which representation is meant then an element of so(3) will simply be denoted as r.

More generally a Lie algebra is a vector space, $\mathbf{V}$, together with a bilinear map $[\cdot,\cdot] : \mathbf{V} \times \mathbf{V} \to \mathbf{V}$ (called the *Lie bracket*) that satisfies, for every $\eta, \mu, \xi \in \mathbf{V}$, (i) $[\eta, \eta] = 0$, and (ii) $[\eta, [\mu, \xi]] + [\xi, [\eta, \mu]] + [\mu, [\xi, \eta]] = 0$. From (i) and the bilinearity property it follows that $[\eta, \mu] = -[\mu, \eta]$. For matrix Lie groups and Lie algebras the corresponding Lie bracket reduces to the standard matrix commutator: if η and μ are square matrices, then $[\eta, \mu] = \eta\mu - \mu\eta$. In particular, on so(3) it is easily verified that the Lie bracket of two elements corresponds to their vector product: $[r_1, r_2] = [r_1][r_2] - [r_2][r_1] = [r_1 \times r_2]$.

The Exponential Mapping

An important connection between a Lie group and its Lie algebra is the *exponential mapping*; defined on each Lie algebra is the exponential mapping into the corresponding Lie group. On matrix groups the exponential mapping is given by the usual matrix exponential, *i.e.*, if A is an element of the Lie algebra, then $\exp A = I + A + \frac{A^2}{2!} + \ldots$ is an element of the Lie group. On so(3) the exponential mapping is *onto*, *i.e.*, for every $R \in \mathrm{SO}(3)$ there exists an $[r] \in \mathrm{so}(3)$ such that $\exp[r] = R$. On SO(3) and its Lie algebra well-known explicit formulas exist for the exponential and its inverse: if $[r] \in \mathrm{so}(3)$, then

$$\exp[r] = I + \frac{\sin\|r\|}{\|r\|} \cdot [r] + \frac{1 - \cos\|r\|}{\|r\|^2} \cdot [r]^2 \tag{4}$$

where $\|r\|$ is the standard Euclidean norm. Alternatively, if $R \in \mathrm{SO}(3)$ such that $\mathrm{Tr}(R) \neq -1$, then

$$\log R = \frac{\phi}{2\sin\phi}(R - R^T) \tag{5}$$

where ϕ satisfies $1 + 2\cos\phi = \mathrm{Tr}(R)$ and $\|\log R\|^2 = \phi^2$. In the case when $\mathrm{Tr}(R) = -1$ two possible solutions for $\log R$ are as follows: if $\hat{r}$ is a unit length eigenvector of R associated with the eigenvalue 1, then $\log R = \pm\pi[\hat{r}]$.

From the above formulas SO(3) can be visualized as a solid ball of radius π, centered at the origin with the antipodal points identified; a point r in the ball represents a rotation by an angle $\|r\|$ about the line passing from the origin through r. Conversely, any $R \in \mathrm{SO}(3)$ can be represented by the set of points corresponding to $\log R$. Note that this representation is unique when restricted to the interior of the solid ball. In general, if $[r]$ is one solution to $\log R$, then $R = e^{[r](1+\frac{2\pi k}{\|r\|})}$ for any integer k. The exponential mapping provides a set of local coordinates for a Lie group over a neighborhood of the identity; Chevalley (1946) calls these coordinates the *canonical coordinates* (of the first kind). On SO(3) we see that the canonical coordinates are obtained from the logarithm formula.

The Lie algebra so(3) can be viewed as providing local coordinates for SO(3) via the exponential map. Another useful interpretation involves angular velocities. If $R(t)$ is a curve in SO(3) describing the orientation of a rigid body relative to an inertial reference frame, then it is not difficult to see that both $\dot{R}R^{-1}$ and $R^{-1}\dot{R}$ are skew-symmetric, and therefore elements of so(3). $R^{-1}\dot{R}$ is in fact the angular velocity of the rigid body in body-fixed frame coordinates, whereas $\dot{R}R^{-1}$ is the angular velocity in inertial frame coordinates.

SO(3) as a Riemannian Manifold

Let $\mathcal{M}$ be a Riemannian manifold[1] of dimension n, with local coordinates $(x_1, x_2, \ldots, x_n)$, and Riemannian metric $ds^2 = \sum_{i,j} g_{ij}(x)dx_idx_j$. If a curve on M is given in local coordinates

[1] See (Gallot *et al* 1990) for a comprehensive introduction to Riemannian manifolds.

by $x(t)$, $0 \leq t \leq 1$, then the length of the curve is given by the integral

$$L = \int_0^1 \sum_{i,j} \left(g_{ij}(x) \frac{dx_i}{dt} \frac{dx_j}{dt} \right)^{\frac{1}{2}} dt \tag{6}$$

Just as a line in Euclidean space can be considered as the shortest path between two points, on a Riemannian manifold the minimum length curve joining two points can be regarded as the analog of the straight line. However, instead of L one usually considers the *energy*

$$E = \int_0^1 \sum_{i,j} g_{ij}(x) \frac{dx_i}{dt} \frac{dx_j}{dt} \, dt \tag{7}$$

The curves that minimize E are called *minimal geodesics.* It can be shown that the curves minimizing E also minimize L, and are automatically parametrized according to arc-length. In local coordinates the minimal geodesics must satisfy the system of differential equations

$$\frac{d^2 x_k}{dt^2} + \sum_{i,j} \Gamma^k_{ij} \frac{dx_i}{dt} \frac{dx_j}{dt} = 0 \tag{8}$$

for $k = 1, 2, \ldots, n$, where

$$\Gamma^k_{ij} = \frac{1}{2} \sum_{l=1}^{n} g^{kl} \left(\frac{\partial g_{il}}{\partial x_j} + \frac{\partial g_{jl}}{\partial x_i} - \frac{\partial g_{ij}}{\partial x_l} \right) \tag{9}$$

and $(g^{kl}) = (g_{kl})^{-1}$.

Clearly the minimal geodesics depend strongly on the choice of Riemannian metric. In general one cannot hope to find a "natural" Riemannian metric for a given manifold, in the sense that the metric is determined by the geometry of the space. However, on compact Lie groups such as SO(3) there does exist a natural metric determined by the requirement of bi-invariance. Recall from earlier that both $R^{-1}\dot{R} \triangleq [\omega_b]$ and $\dot{R}R^{-1} \triangleq \omega_s]$ are elements of so(3) that correspond to the angular velocity in body-fixed and inertial frame coordinates, respectively. Since any tangent vector $\dot{R}$ can be identified with an element of so(3) by either left or right translation, any inner product on so(3) defines two distinct Riemannian metrics on SO(3). Let this inner product be given by the symmetric positive-definite quadratic form Q. The *left-invariant Riemannian metric* induced from Q is then $\langle \dot{R}, \dot{R} \rangle_l = \frac{1}{2} \omega_b^T Q \omega_b$; similarly, the right-invariant Riemannian metric is given by $\langle \dot{R}, \dot{R} \rangle_r = \frac{1}{2} \omega_s^T Q \omega_s$. If $\frac{1}{2} \omega_s^T Q \omega_s = \frac{1}{2} \omega_b^T Q \omega_b$, then Q is said to define a *bi-invariant Riemannian metric.* Clearly equality holds if and only if $Q = cI$, for $c > 0$ any scalar constant.

Not all Lie groups have bi-invariant Riemannian metrics (*e.g.*, SE(3)), but for compact Lie groups like SO(3) one is always guaranteed to exist. In this case the geodesics (with respect to the bi-invariant metric) are the *one-parameter subgroups* and its translates: on SO(3), for example, the minimal geodesic between R_1 and R_2 is given by

$$R(t) = R_1 e^{[r_{12}]t} = e^{[r_{21}]t} R_1, \qquad 0 \leq t \leq 1 \tag{10}$$

where $[r_{12}]$ is the minimum norm value of $\log(R_1^{-1} R_2)$, and $[r_{21}] = R_1 [r_{12}] R_1^T$.

3 Two Point Interpolation

Bi-invariant Solutions

On Riemannian manifolds second derivatives are generalized by the symmetric covariant derivative compatible with the Riemannian metric,[2] denoted by the symbol ∇. The equivalent energy functional to Equation (1) is then given by

$$J(x) = \int \langle \nabla_{\frac{\partial}{\partial t}} \dot{x}, \nabla_{\frac{\partial}{\partial t}} \dot{x} \rangle \, dt \tag{11}$$

where $x(t)$ denotes the curve and $\langle \cdot, \cdot \rangle$ the Riemannian metric. In local coordinates $\nabla_{\frac{\partial}{\partial t}} \dot{x}$ is just the left-hand side of Equation (8), where the Riemannian metric is given by $g_{ij}(x)$. Since $J(x)$ is a second-order functional, four boundary conditions are required to specify a unique solution. In general the geodesics will not be admissible curves, but when they are the integrand vanishes, so that the geodesics minimize $J(x)$,

The Euler-Lagrange equations for $J(x)$ (sometimes referred to as the equations for *geodetic deviation*) are

$$\nabla^3_{\frac{\partial}{\partial t}} \dot{x} + R(\nabla_{\frac{\partial}{\partial t}} \dot{x}, \dot{x})(\dot{x}) = 0 \tag{12}$$

(Noakes *et al* 1989, Milnor 1969) where R is the Riemannian curvature tensor of ∇. These equations are quite complex when expressed in local coordinates. For matrix Lie groups with a left- or right-invariant Riemannian metric it is often more convenient to derive the equations directly from the first-order necessary conditions. Specifically, let **G** and **g** be the matrix Lie group and its corresponding Lie algebra, respectively, and $\langle \cdot, \cdot \rangle$ an inner product

[2] Again, see (Gallot *et al* 1990) for a complete discussion of covariant derivatives.

on g defining a left-invariant metric. The objective then is to find a curve $U(t)$ in g that minimizes

$$J(U) = \int_0^1 \langle \dot{U}, \dot{U} \rangle \, dt \tag{13}$$

subject to

$$\dot{X}(t) = X(t)U(t) \tag{14}$$

with $X(0)$, $X(1)$, $U(0)$, and $U(1)$ given. (If the right-invariant metric were used instead the constraint would then be $\dot{X}(t) = U(t)X(t)$.)

On SO(3) the smoothness functional with respect to the bi-invariant metric turns out to be the integral of the squared Euclidean norm of the angular acceleration. In order to express the functional in terms of the canonical coordinates the following result is needed. Let $R(t)$ be a curve in SO(3) parametrized in canonical coordinates by $R(t) = R\exp[\xi(t)]$, where $\xi(t)$ is a curve in $\Re^3$ and $R \in \mathrm{SO}(3)$ is some given constant. It can be shown that the angular velocity in body coordinates, denoted $\omega(t)$, is

$$R^{-1}(t)\dot{R}(t) = [\omega(t)] = \int_0^1 e^{-[\xi(t)]s}[\dot{\xi}(t)]e^{[\xi(t)]s} \, ds \tag{15}$$

which can be further simplified to the vector equation $\omega(t) = A(\xi)\dot{\xi}(t)$, where

$$A(\xi) = I - \frac{1 - \cos\|\xi\|}{\|\xi\|^2}[\xi] + \frac{\|\xi\| - \sin\|\xi\|}{\|\xi\|^3}[\xi]^2 \tag{16}$$

The smoothness functional in canonical coordinates is then

$$J(\xi) = \int_0^1 \|\frac{d}{dt}(A(\xi)\dot{\xi})\|^2 dt \tag{17}$$

We now show that the solutions that minimize $J(\xi)$ are "invariant" (in a sense to be made precise below) with respect to right- and left-translations of the boundary values. This result is not surprising considering that $J(\xi)$ is defined in terms of the bi-invariant Riemannian metric on SO(3). Nevertheless, the actual calculations turn out to be useful for deriving the form of the cubic spline solution. In what follows it may be helpful to bear in mind the actual engineering problem being addressed, which is to interpolate a smooth orientation trajectory for a rigid body between two given orientations, subject to angular velocity constraints at both endpoints. Suppose that inertial and body-fixed reference frames have been chosen. The interpolation problem can then be stated mathematically as follows: find a curve $R(t)$ in SO(3) that minimizes $J(\xi)$, while satisfying the boundary conditions $R(0) = R_0$, $R(1) = R_1$, $R^{-1}(0)\dot{R}(0) = [\omega_0]$, and $R^{-1}(1)\dot{R}(1) = [\omega_1]$, where $R_0, R_1 \in \mathrm{SO}(3)$, $\omega_0, \omega_1 \in \Re^3$ are given. For convenience we parametrize the admissible curves according to $R(t) = R_0 e^{[\xi(t)]}$, where $\xi(t)$ is a curve in $\Re^3$. $\xi(t)$ must then satisfy

$$(i) \qquad \xi(0) = 0 \tag{18}$$

$$(ii) \quad [\xi(1)] = \log(R_0^{-1}R_1) \tag{19}$$

$$(iii) \qquad \dot{\xi}(0) = \omega_0 \tag{20}$$

$$(iv) \quad A(\xi(1))\dot{\xi}(1) = \omega_1 \tag{21}$$

where $A(\xi)$ is as given in Equation (16). If now a different body-fixed frame is chosen, the boundary values R_0 and R_1 are then right-translated by some $\Theta \in \mathrm{SO}(3)$ to $\tilde{R}_0 = R_0\Theta$ and $\tilde{R}_1 = R_1\Theta$, respectively, while the velocity vectors ω_0 and ω_1 are transformed to $\Theta^{-1}\omega_0$ and $\Theta^{-1}\omega_1$, respectively. The new curve $\tilde{R}(t) = \tilde{R}_0 e^{[\tilde{\xi}(t)]}$ must then minimize $J(\tilde{\xi}) = \int_0^1 \|\frac{d}{dt}(A(\tilde{\xi})\dot{\tilde{\xi}})\|^2 dt$ while satisfying

$$(i) \qquad \tilde{\xi}(0) = 0 \tag{22}$$

$$(ii) \quad [\tilde{\xi}(1)] = \log(\tilde{R}_0^{-1}\tilde{R}_1) = \log(\Theta^{-1}R_0^{-1}R_1\Theta) = \Theta^{-1}[\xi(1)]\Theta \tag{23}$$

$$(iii) \qquad \dot{\tilde{\xi}}(0) = \Theta^{-1}\omega_0 \tag{24}$$

$$(iv) \qquad A(\tilde{\xi}(1))\dot{\tilde{\xi}}(1) = \Theta^{-1}\omega_1 \tag{25}$$

If we consider $\tilde{\xi}(t)$ to be of the form $\Theta^{-1}\xi(t)$, then the above boundary conditions are identical to those of Equations (18)-(21). Moreover, since $A(\tilde{\xi}) = \Theta^{-1}A(\xi)\Theta$, it follows that $A(\tilde{\xi})\dot{\tilde{\xi}} = \Theta^{-1}A(\xi)\dot{\xi}$, or $J(\tilde{\xi}) = J(\xi)$. Therefore, if $\xi(t)$ is a solution to the original variational problem, then $\tilde{\xi}(t) = \Theta^{-1}\xi(t)$ is a solution to the latter. Using standard matrix exponential identities one can now show that $\tilde{R}(t) = \tilde{R}_0 e^{[\Theta^{-1}\xi(t)]} = R_0\Theta e^{\Theta^{-1}[\xi(t)]\Theta} = R_0 e^{[\xi(t)]}\Theta = R(t)\Theta$ as claimed.

In a similar fashion one can show that if R_0 and R_1 are left-translated by some constant $\Theta \in \mathrm{SO}(3)$ to ΘR_0 and ΘR_1, respectively (corresponding to a change in the inertial frame), then the new solution is $\Theta R(t)$, the left-translate of the original solution. Observe that the geodesics on SO(3) are special cases when $\xi(t)$ is linear in t.

3.1 A Cubic-Spline Solution

For interactive applications, solving the above two-point boundary value problem is not practical. However, if the two endpoints are assumed reasonably close to one another then the solution curve simplifies to a cubic polynomial in canonical coordinates. Specifically, let R_0 and R_1 denote the endpoints, and assume $\| \log(R_0^{-1}R_1) \|$ is small. Then the interpolating curve $R(t) = R_0 e^{[\xi(t)]}$ will be such that $\xi(t)$ can also be assumed of small magnitude, and $A(\xi) \approx I$. Therefore the smoothness functional of Equation (17) can be approximated by

$$\int_0^1 \|\ddot{\xi}(t)\|^2 dt \tag{26}$$

whose solutions are clearly cubics. SO(3) curves whose image in the canonical coordinates are cubics will be referred to as cubic splines in SO(3).

The cubic spline solution to the two-point interpolation problem on SO(3) is as follows. Given the boundary conditions $R(0) = R_0$, $R(1) = R_1$, $R^{-1}(0)\dot{R}(0) = [\omega_0]$, $R^{-1}(1)\dot{R}(1) = [\omega_1]$, the solution curve is $R(t) = R_0 e^{[at^3+bt^2+ct]}$, where $a, b, c \in \Re^3$ are constants satisfying

- $a + b + c = \epsilon$, where $[\epsilon] = \log(R_0^{-1}R_1)$
- $c = \omega_0$
- $A(\epsilon)(3a + 2b + c) = \omega_1$, where $A(\epsilon) = I - \frac{\cos\|\epsilon\|}{\|\epsilon\|^2}[\epsilon] + \frac{\|\epsilon\| - \sin\|\epsilon\|}{\|\epsilon\|^3}[\epsilon]^2$.

Using calculations analogous to those of the previous section, it can be shown that the cubic splines are left-invariant. That is, if $R(t)$ is a cubic spline satisfying the above boundary conditions, then the cubic spline $\tilde{R}(t)$ that satisfies the new boundary conditions $\tilde{R}(0) = \Theta R_0$, $\tilde{R}(1) = \Theta R_1$, $\tilde{R}^{-1}(0)\dot{\tilde{R}}(0) = [\omega_0]$, and $\tilde{R}^{-1}(1)\dot{\tilde{R}}(1) = [\omega_1]$, is given by $\Theta R(t)$. Right-invariance can also be shown similarly. The cubic splines on SO(3), therefore, are bi-invariant.

The following matrix identities are instrumental in determining the coefficients of right- or left-translated cubic splines. First, for any matrix A, $Pe^AP^{-1} = e^{PAP^{-1}}$. Secondly, if $R \in \mathrm{SO}(3)$, then $R[\omega]R^T = [R\omega]$ for any $\omega \in \Re^3$. With these identities it can be shown that if $R(t) = R_0 e^{[at^3+bt^2+ct]}$, then $R(t)\Theta = R_0\Theta e^{[\Theta^{-1}(at^3+bt^2+ct)]}$, *i.e.*, the coefficients of the right-translated cubic spline are given by $\tilde{a} = \Theta^{-1}a$, $\tilde{b} = \Theta^{-1}b$, and $\tilde{c} = \Theta^{-1}c$. Finally, observe that when $\omega_0 = \omega_1$, the cubic splines reduce to geodesics. This feature, along with the bi-invariance property and computational efficiency, is what makes cubic splines an attractive choice for interpolating trajectories on SO(3).

4 Interpolation of Multiple Points

We now present a complete algorithm for interpolating through multiple points in SO(3) using cubic splines. Analogous to the Euclidean case, the interpolated curve in SO(3) maintains continuity of both angular velocities and accelerations at the knot points. The algorithm we present requires the following as inputs: an ordered set of $n+1$ rotation matrices $\{R_0, R_1, \ldots, R_n\}$ (the knot points), a set of $n+1$ scalars $t_0 < t_1 < \cdots < t_n$ (the knot times), an initial angular velocity $\omega_0 \in \Re^3$, and an intial angular acceleration $\alpha_0 \in \Re^3$. Both ω_0 and α_0 are expressed in body-fixed reference frame coordinates. This set of inputs, while a slight departure from the usual set for Euclidean cubic splines, is chosen for convenience; for different inputs (*e.g.*, specifying final velocities rather than initial accelerations) the corresponding algorithm can be derived using results from the following analysis.

The interpolated curve is of the form

$$R(t) = \begin{cases} R_1(t) = R_0 e^{[\xi_1(t)]}, & t_0 \le t \le t_1 \\ \vdots & \vdots \\ R_n(t) = R_{n-1} e^{[\xi_n(t)]}, & t_{n-1} \le t \le t_n \end{cases} \tag{27}$$

where

$$\xi_i(t) = a_i \left(\frac{t - t_{i-1}}{t_i - t_{i-1}}\right)^3 + b_i \left(\frac{t - t_{i-1}}{t_i - t_{i-1}}\right)^2 + c_i \left(\frac{t - t_{i-1}}{t_i - t_{i-1}}\right) \tag{28}$$

Here a_i, b_i, c_i, $i = 1, \ldots, n$, are constant vectors in $\Re^3$ that are determined using the following formulas for the angular velocity and acceleration. The angular velocity in body-fixed coordinates is, from Equation (16),

$$\omega(t) = \begin{cases} \omega_1(t) = A_1(t)\dot{\xi}_1(t), & t_0 \le t \le t_1 \\ \vdots & \vdots \\ \omega_n(t) = A_n(t)\dot{\xi}_n(t), & t_{n-1} \le t \le t_n \end{cases} \tag{29}$$

where

$$A_i(t) = I - \frac{\cos\|\xi_i(t)\|}{\|\xi_i(t)\|^2}[\xi_i(t)] + \frac{\|\xi_i(t)\| - \|\sin\xi_i(t)\|}{\|\xi_i(t)\|^3}[\xi_i(t)]^2 \tag{30}$$

$$\dot{\xi}_i(t) = 3a_i\left(\frac{t-t_{i-1}}{t_i-t_{i-1}}\right)^2 + 2b_i\left(\frac{t-t_{i-1}}{t_i-t_{i-1}}\right) + c_i \tag{31}$$

The angular acceleration is

$$\alpha(t) = \begin{cases} \alpha_1(t) = \dot{\omega}_1(t), & t_0 \le t \le t_1 \\ \vdots & \vdots \\ \alpha_n(t) = \dot{\omega}_n(t), & t_{n-1} \le t \le t_n \end{cases} \tag{32}$$

where

$$\begin{aligned} \alpha_i(t) &= \ddot{\xi}_i - \tfrac{\langle \xi_i, \dot{\xi}_i \rangle}{\|\xi\|^4}(\cos\|\xi\| - \|\xi\|\sin\|\xi\| - 1)(\xi \times \dot{\xi}) - \tfrac{1-\cos\|\xi\|}{\|\xi\|^2}(\xi \times \ddot{\xi}) \\ &+ \tfrac{\langle \xi, \dot{\xi} \rangle}{\|\xi\|^5}(3\sin\|\xi\| - \|\xi\|\cos\|\xi\| - 2\|\xi\|)(\xi \times (\xi \times \dot{\xi})) \\ &+ \tfrac{\|\xi\| - \sin\|\xi\|}{\|\xi\|^3}(\dot{\xi} \times (\xi \times \dot{\xi}) + \xi \times (\xi \times \ddot{\xi})) \end{aligned} \tag{33}$$

Here $\langle \cdot, \cdot \rangle$ denotes the Euclidean norm in $\Re^3$, ξ_i and $\dot{\xi}_i$ are as above, and

$$\ddot{\xi}_i(t) = 6a_i\left(\frac{t-t_{i-1}}{t_i-t_{i-1}}\right) + 2b_i \tag{34}$$

The initially known quantities are, in addition to the knot points and knot times, $\omega_1(t_0) = \omega_0$ and $\alpha_1(t_0) = \alpha_0$. From these initial conditions a_1, b_1, and c_1 can be uniquely determined, from which $\omega_1(t_1) = \omega_1$ and $\alpha_1(t_1) = \alpha_1$ can in turn be determined. By the continuity requirements $\omega_1(t_1) = \omega_2(t_1)$ and $\alpha_1(t_1) = \alpha_2(t_1)$, the vector coefficients a_2, b_2, and c_2 can now be determined. This procedure is repeated until all the coefficients a_i, b_i, c_i, $i = 1, \ldots, n$ have been found. We now present the complete algorithm.

- **Given:**

  $\{R_0, R_1, \ldots, R_n\}$ = knot points

  $\{t_0, t_1, \ldots, t_n\}$ = knot times

  ω_0 = angular velocity at t_0 in body-fixed coordinates

  α_0 = angular acceleration at t_0 in body-fixed coordinates

- **Preprocessing:** for $i = 1$ to n find

$$\begin{aligned} [\epsilon_i] &= \log(R_{i-1}^T R_i) \\ \Lambda_i &= I - \tfrac{\cos\|\epsilon_i\|}{\|\epsilon_i\|^2}[\epsilon_i] + \tfrac{\|\epsilon_i\| - \|\sin\epsilon_i\|}{\|\epsilon_i\|^3}[\epsilon_i]^2 \end{aligned}$$

- **Initialization:**

$$\begin{aligned} c_1 &= \omega_0 \\ b_1 &= \alpha_0/2 \\ a_1 &= \epsilon_1 - b_1 - c_1 \end{aligned}$$

- **Recursion:** for $i = 2$ to n do

$$\begin{aligned}
s &= \|\epsilon_i\| \quad \text{(temp. var.)} \\
t &= 3a_{i-1} + 2b_{i-1} + c_{i-1} \quad \text{(temp. var.)} \\
u &= 6a_{i-1} + 2b_{i-1} \quad \text{(temp. var.)} \\
c_i &= \Lambda_{i-1} c_{i-1} \\
b_i &= \tfrac{1}{2}\Big(u - \tfrac{\langle s,t\rangle}{\|s\|^4}(\cos\|s\| - \|s\|\sin\|s\| - 1)(s \times t) - \tfrac{1-\cos\|s\|}{\|s\|^2}(s \times u) \\
&\quad + \tfrac{\langle s,t\rangle}{\|s\|^5}(3\sin\|s\| - \|s\|\cos\|s\| - 2\|s\|)(s \times (s \times t)) \\
&\quad + \tfrac{\|s\| - \sin\|s\|}{\|s\|^3}(t \times (s \times t) + s \times (s \times u))\Big) \\
a_i &= \epsilon_i - b_i - c_i
\end{aligned}$$

- **Result:** for $t_{i-1} \le t \le t_i$,

$$R(t) = R_{i-1} \exp[a_i \left(\frac{t - t_{i-1}}{t_i - t_{i-1}}\right)^3 + b_i \left(\frac{t - t_{i-1}}{t_i - t_{i-1}}\right)^2 + c_i \left(\frac{t - t_{i-1}}{t_i - t_{i-1}}\right)]$$

The ϵ_i are found from the log formula of Equation (5). In cases where ϵ_i has two possible values (corresponding to the two antipodal points on the sphere of radius π), either value will still generate the same orientation trajectory in SO(3). The interpolated curve $R(t)$ is then evaluated using the exponential formula of Equation (4).

Example

In this section we provide a simple example to illustrate the utility of the interpolation technique developed in this paper. Figure 1 shows several positions of an end effector of a robot manipulator. These positions are used as control positions and a cubic interpolation is performed on SO(3) to generate the motion of the end effector depicted in Figure 2. We have separately interpolated the orientations from the position of a point on the end effector. The techniques presented in this paper are used for the interpolation of the orientation parts of the trajectory.

5 Conclusions

By viewing SO(3) as a Lie group equipped with a natural Riemannian metric, we have presented an algorithm for interpolating through multiple points in SO(3) that can be interpreted as a type of generalized cubic spline in rotation space. The main advantage of this approach is that bi-invariant curves can now be generated in a computationally efficient way. In the context of moving rigid bodies, bi-invariance ensures that the orientation trajectories are independent of choice of inertial or body-fixed reference frames. Rotational cubic splines are also an effective compromise between the computational needs for interactive CAD versus greater curve smoothness.

In extending the rotational cubic spline techniques to the interpolation of general motions for rigid bodies, several additional issues need to be addressed. First, it is a well-known classical result that SE(3), the Lie group of rigid-body displacements, does not admit a bi-invariant Riemannian metric. One physical consequence of this fact is that there is no interpolation scheme that is bi-invariant; if one were to imagine infinitely large rigid bodies, then any method of motion interpolation will ultimately depend on the choice of inertial or body-fixed reference frame. It is possible, however, to relax the requirement of bi-invariance, in which case left- or right-invariant motions can be generated using the above construction. The most straightforward approach is to interpolate the orientation and position (of some special point on the rigid body) trajectories separately. Alternatively, the exponential and logarithm mappings on SE(3) and its Lie algebra (see, *e.g.*, Park *et al* 1993) can be applied to construct cubic splines in the same way as for SO(3). The difference between the two approaches is best illustrated by the problem of interpolating between two configurations of a rigid body. Assuming the body-fixed frame has been attached to the center of mass, and the initial generalized velocities and accelerations are given, the former approach results in a linear motion of the center of mass. The latter method, however, produces a screw motion as the final trajectory. Interpolating the positions and orientations separately would therefore seem more natural from the point of view of dynamics.

6 Acknowledgement

This work was supported in part by US Army Research Office grant number DAAL03-90-G-0005 and in part by California Department of Transportation through the AHMCT program.

References

[1] Barr, A. H., Currin, B., Gabriel, S. and Hughes, J. F. 1992. Smooth Interpolation of Orientations with Angular Velocity Constraints using Quaternions. *Computer Graphics* 26(2), 313-320.

[2] Belinfante, J. G., and Kolman, B. 1972. *A Survey of Lie Groups and Lie Algebras with Applications and Computational Methods.* Philadelphia: Siam.

[3] Chevalley, C. 1946. *Theory of Lie Groups.* Princeton: Princeton University Press.

[4] Gallot, S., Hulin, D., and Lafontaine, J. 1990. *Riemannian Geometry.* Berlin: Springer-Verlag.

[5] Ge, Q. J., and Ravani, B. 1994. Computer-aided design of motion interpolants. *ASME J. Mechanical Design*116: 756-762.

[6] Ge, Q. J., and Ravani, B. 1994. Geometric construction of Bèzier motions. *ASME Journal of Mechanical Design* 116: 749-755.

[7] Ge, Q. J., and Ravani, B. 1993. Computational Geometry and Motion Approximation. in *Computational Kinematics.* J. Angeles, G. Hommel, and P. Kovacs, Eds. Amsterdam: Kluwer.

[8] Hart, J., C., Francis, G. K., and Kauffman, L. H., 1994. Visualizing Quaternion Rotation. *ACM Trans. on Graphics* 13 (3): 256-276.

[9] Juttler, B., 1994. Visulaization of moving objects using dual quaternion curves. *Computers and Graphics* 18(3).

[10] Milnor, J. W. 1969. *Morse Theory.* Princeton: Princeton University Press.

[11] Noakes, L., Heinzinger, G., and Paden, B. 1989. Cubic splines on curved spaces. *IMA Journal of Mathematical Control and Applications* 6:465-473.

[12] Park, F. C., and Ravani, B. 1994. Bèzier curves on Riemannian manifolds and Lie groups with kinematics applications. *ASME J. Mechanical Design* (to appear).

[13] Park, F. C., Murray, A. P., and McCarthy, J. M. 1993. Designing mechanisms for workspace fit. in *Computational Kinematics.* J. Angeles, G. Hommel, and P. Kovacs, Eds. Amsterdam: Kluwer.

[14] Shoemake, K. 1985. Animating rotation with quaternion curves. *ACM Siggraph* 19(3):245-254.

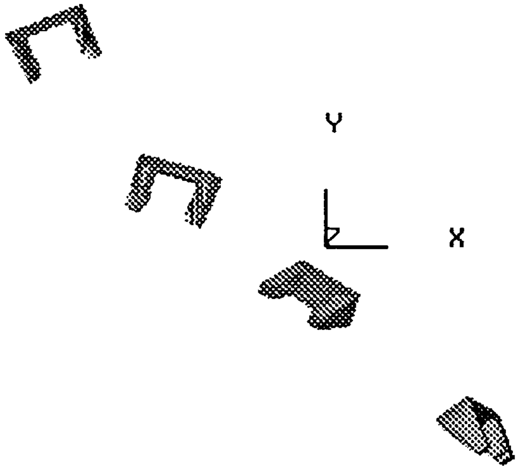


Figure 1. End Configurations

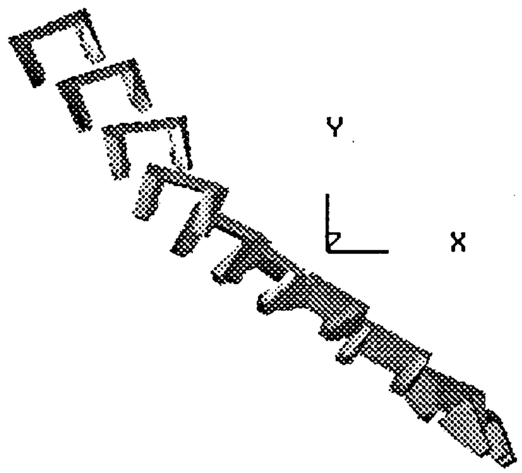


Figure 2. Interpolated Motion

Proceedings of the 1991 IEEE
International Conference on Robotics and Automation
Sacramento, California - April 1991

Kinematic Localization for World Model Calibration in Off-Line Robot Programming Using Clifford Algebra

B. Ravani
Associate Professor

Q. J. Ge
Research Associate

Computer Integrated Design and Manufacturing Laboratory
Department of Mechanical, Aeronautical and Materials Engineering
University of California-Davis
Davis, CA 95616

Abstract

This paper deals with calibrating spatial relationships among CAD models of objects used in off line robot programming. An approach based on a Clifford algebra is used and simple algorithms are developed for this kinematic localization problem. The position and orientation of each object in a workcell are represented by multivectors of different ranks and the localization problem is converted into that of solving a set of multivector equations. Given locations of points in two frames, an averaging process is developed which yields the best orientation as the root of four linear equations and the best position as a linear combination of the best rotation and the centroids of two point systems. If normal vectors are also sensed, the Clifford algebra representation facilitates definition of a measure of orientation error compatible with the position error.

Introduction

CAD-based graphical off-line robot programming and simulation requires a computer representation of the robot workcell environment. This representation, called the *world model*, provides the necessary information about the robot workcell for a task planner to generate appropriate robot motion control commands. Errors in task execution are due to inaccuracies of the world model and uncertainties introduced by sensors and actuators during execution. The latter set of errors can be eliminated or reduced by sensory control. The first set of errors may be corrected by calibrating the world model before task execution. This paper deals with the first level of world model calibration which involves calculation of location of coordinate frames attached to a CAD model and is here referred to as the *kinematic localization* problem.

Calibration of the world model first requires sensory interactions with the robot workcell. In the past there have been several studies related to sensory monitoring of the robot workcell environment, see Grossman and Taylor (1978), and Ishii et al (1988). Any of the systems proposed in these papers can be used with a kinematic localization algorithm for calibration purposes. Kinematic localization problems have mostly been considered in the field of computer vision for calculation of position and orientation of objects (see, for example, Grimson and Lozano-Perez (1984), Faugeras and Herbert (1986) and Gunnarsson and Prinz (1987)). These works however have not been used for world model calibration and more importantly have not completely exploited the geometric nature and the kinematic structure of the problem. The resulting algorithms therefore have suffered from unnecessary nonlinearities or computational requirements.

In this paper, a representation based on Clifford algebra is used for the kinematic relationship between two frames. Clifford algebra was proposed by Clifford (1876). For recent account of Clifford algebra, see Hestenes and Sobczyk (1984), McCarthy (1990). The Clifford algebra formulation converts the localization problem into that of solving a set of multivector equations. When unit vector normals are also measured, in addition to point measurements, this representation facilitates definition of a measure of orientation error compatible with the position error. More importantly, it also facilitates exploitation of the geometric nature and kinematic structure between two point system measured with respect to the world frame and to the natural frame. This results in an averaging process which reduces the localization problem to that of solving a set of four linear equations. The orientation of an object obtained as the root of the equations is shown to be a least squares solution and the best position of the ob-

CH2969-4/91/0000/0584$01.00 © 1991 IEEE

ject is a linear combination of the optimal orientation and the centroids of the point systems in two frames.

1 The Clifford Algebra

The Clifford algebra $C(\mathbf{R}^3)$ of a 3-dimensional vector space $(\mathbf{R}^3)$ is generated from elements of the space by a product operation called the *geometric product.* This product is linear and associative, and in addition, the products of basis vectors of $\mathbf{R}^3$ satisfy the following properties (see McCarthy 1990 and Ge 1990):

$$\mathbf{e}_m\mathbf{e}_n = \begin{cases} -\mathbf{e}_n\mathbf{e}_m & \text{for } m \neq n, \\ 1 & \text{for } m = n, \end{cases} \tag{1}$$

where $\mathbf{e}_m$, $m = 1, 2, 3$, are the basis vectors of $\mathbf{R}^3$ such that $\mathbf{e}_m$ has a 1 in the m^{th} position and zeroes elsewhere.

The geometric product $\mathbf{e}_m\mathbf{e}_n$ is not a vector nor a scalar but a new entity called *rank 2 multivector* or *bivector.* In Clifford algebra, a scalar is considered as a *rank 0 multivector* and a vector is a *rank 1 multivector.* The geometric product of all three basis vectors

$$i = \mathbf{e}_1\mathbf{e}_2\mathbf{e}_3, \tag{2}$$

is a *rank 3 multivector* or *trivector.* It has the property that $i^2 = -1$ and commutes with every vector in the space. Furthermore, it facilitates representation of basis bivectors in terms of basis vectors:

$$\mathbf{e}_2\mathbf{e}_3 = i\mathbf{e}_1,\ \mathbf{e}_3\mathbf{e}_1 = i\mathbf{e}_2,\ \mathbf{e}_1\mathbf{e}_2 = i\mathbf{e}_3. \tag{3}$$

The geometric product of two general vectors, $\mathbf{x} = x_1\mathbf{e}_1 + x_2\mathbf{e}_2 + x_3\mathbf{e}_3$ and $\mathbf{y} = y_1\mathbf{e}_1 + y_2\mathbf{e}_2 + y_3\mathbf{e}_3$, is a combination of a scalar and a bivector:

$$\mathbf{xy} = \mathbf{x} \cdot \mathbf{y} + i(\mathbf{x} \times \mathbf{y}), \tag{4}$$

where $\mathbf{x} \cdot \mathbf{y}$ is the vector scalar product and $\mathbf{x} \times \mathbf{y}$ is the vector cross product.

A general element of the Clifford algebra of $\mathbf{R}^3$ is a combination of multivectors with ranks from 0 to 3. It includes a scalar term, a vector term with three basis vectors $\mathbf{e}_m$, $m = 1, 2, 3$, a bivector term with three basis bivectors $i\mathbf{e}_m$, and a trivector term $i = \mathbf{e}_1\mathbf{e}_2\mathbf{e}_3$. The set of elements of as combinations of scalars and bivectors constitutes the even subalgebra of $C(\mathbf{R}^3)$. The set of elements of as combinations of vectors and trivectors constitutes the odd subalgebra. The conjugates for these elements are obtained by replacing i with $-i$.

The product of a vector $\mathbf{x} = x_1\mathbf{e}_1 + x_2\mathbf{e}_2 + x_3\mathbf{e}_3$ with an even element $\mathbf{q} = iq_1\mathbf{e}_1 + iq_2\mathbf{e}_2 + iq_3\mathbf{e}_3 + q_4$ can be expressed in matrix form as

$$\mathbf{xq} = [x^-]\mathbf{q}^*, \quad \mathbf{qx} = [x^+]\mathbf{q}^*,$$

where $\mathbf{q}^* = (-q_1, -q_2, -q_3, q_4)$ is the conjugate of $\mathbf{q}$ and $[x^-]$, $[x^+]$ are 4×4 skew symmetric matrices given by

$$\begin{aligned} [x^-] &= \begin{bmatrix} 0 & x_3 & -x_2 & x_1 \\ -x_3 & 0 & x_1 & x_2 \\ x_2 & -x_1 & 0 & x_3 \\ -x_1 & -x_2 & -x_3 & 0 \end{bmatrix}, \\ [x^+] &= \begin{bmatrix} 0 & -x_3 & x_2 & x_1 \\ x_3 & 0 & -x_1 & x_2 \\ -x_2 & x_1 & 0 & x_3 \\ -x_1 & -x_2 & -x_3 & 0 \end{bmatrix}. \end{aligned} \tag{5}$$

The product of these two matrices commutes, i.e. $[x^-][x^+] = [x^+][x^-]$. In addition, we have $[x^+][x^+] = [x^-][x^-] = -|\mathbf{x}|^2[I]$ where $|\mathbf{x}|^2$ is the length square of the vector $\mathbf{x}$ and $[I]$ is the 4×4 identity matrix.

It can also be shown that the product of three elements, $\mathbf{x}$ (a vector), $\mathbf{q}$ (an element of $C^+(\mathbf{R}^3)$) and $\mathbf{y}$ (another vector) can be expressed in matrix form as

$$\mathbf{xqy} = -[x^+][y^-]\mathbf{q}, \tag{6}$$

where $[y^-]$ is obtained from $[x^-]$ by replacing $\mathbf{x}$ with $\mathbf{y}$.

2 Kinematic Relationships in the World Model

A world model usually includes geometrical, relational as well as physical descriptions of the workcell. The world model RWORLD proposed by Ravani (1988) uses a multi-primitive representation of the workcell environment at an abstract level. Frame primitive is used to mark the position and orientation of a fixed reference frame in space (called the *world frame*). It can be used to mark the location of an object or device, in which case they are referred to as *natural frames.* It can also be used to mark feature locations on objects or devices, in which case they are referred to as *auxiliary frames.*

This section identifies the spatial relationship between a natural frame and a world frame (or between a auxiliary frame and a natural frame) with an element of the Clifford algebra of $\mathbf{R}^3$. The Clifford algebra representation is closely related to the quaternion representation of transformations, see Bottema and Roth (1979), McCarthy (1990) and Ge (1990).

The kinematic relationship between two frames consists of an orientation relationship and a positional relationship. The orientation relationship is represented by the following even element (Ge 1990):

$$\mathbf{q} = i\mathbf{s}\sin(\theta/2) + \cos(\theta/2), \tag{7}$$

where $\mathbf{s} = s_x\mathbf{e}_1 + s_y\mathbf{e}_2 + s_z\mathbf{e}_3$ is the unit vector along the axis of rotation and θ the angle of rotation. The four components of $\mathbf{q} = (q_1, q_2, q_3, q_4)$ given by (7) satisfy the relation

$$q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1, \tag{8}$$

and are called the *Euler parameters* of rotation.

If the location of the origin of one frame relative to another is known, say it is specified by the vector $\mathbf{d} = d_1\mathbf{e}_1 + d_2\mathbf{e}_2 + d_3\mathbf{e}_3$, then the positional relationship is represented by the following geometric product

$$\mathbf{q}^0 = (1/2)\mathbf{q}\mathbf{d}, \tag{9}$$

where $\mathbf{q}$ is the orientation defined by (7). In general, the positional relationship is given by

$$\mathbf{q}^0 = (1/2)(\mathbf{q}\mathbf{x}' - \mathbf{x}\mathbf{q}), \tag{10}$$

where $\mathbf{x} = x_1\mathbf{e}_1 + x_2\mathbf{e}_2 + x_3\mathbf{e}_3$ and $\mathbf{x}' = x_1'\mathbf{e}_1 + x_2'\mathbf{e}_2 + x_3'\mathbf{e}_3$ are the coordinate vectors of a point measured relative to the natural frame and to the world frame. $\mathbf{q}^0$ is an element of the odd subalgebra, i.e. a combination of a vector and a trivector.

Thus the pair of elements, $(\mathbf{q}, \mathbf{q}^0)$, uniquely determine the kinematic relationship between two frames. Note that the pair $(-\mathbf{q}, -\mathbf{q}^0)$ represents the same relationship as $(\mathbf{q}, \mathbf{q}^0)$. Furthermore, the definition of $\mathbf{q}^0$ implies that the components of $\mathbf{q}$ and $\mathbf{q}^0$ satisfy

$$-q_1q_1^0 - q_2q_2^0 - q_3q_3^0 + q_4q_4^0 = 0. \tag{11}$$

For orientation relationship, the element $\mathbf{q}^0 = 0$ since $\mathbf{d} = 0$. Therefore, the pair of unit vectors $(\mathbf{u} : \mathbf{u}')$ represents the direction of a line in both frames satisfy

$$\mathbf{q}\mathbf{u}' - \mathbf{u}\mathbf{q} = 0. \tag{12}$$

3 Minimal Solution

To calibrate the kinematic relationship between two frames for CAD models is to determine the real transformation from one frame to another. In the Clifford algebra representation, it is to determine the pair of elements $(\mathbf{q}, \mathbf{q}^0)$ corresponding to the transformation. This section focuses on calibrating the kinematic relationship between an object's natural frame and the world frame. We first discuss the minimum information required for the calibration with the assumption that the sensory information about geometric features in both frames are available. We then provide an algorithm for calibration with point measurements only in the world frame.

3.1 Minimum Sensory Information

To determine the orientation relationship $\mathbf{q}$, directions of two distinct lines are required. Let the two directions be given by two pairs of unit vectors $(\mathbf{u} : \mathbf{u}')$ and $(\mathbf{v} : \mathbf{v}')$ where $\mathbf{u}$, $\mathbf{v}$ are measured relative to the object's natural frame and $\mathbf{u}'$, $\mathbf{v}'$ are measured relative to the world frame. Each pair satisfies (12), i.e.

$$\mathbf{q}\mathbf{u}' - \mathbf{u}\mathbf{q} = 0, \tag{13}$$

$$\mathbf{q}\mathbf{v}' - \mathbf{v}\mathbf{q} = 0. \tag{14}$$

The substitution of $\mathbf{q} = i\mathbf{s}\sin(\theta/2) + \cos(\theta/2)$ into (13) yields, after some algebra

$$\begin{aligned}\sin(\theta/2)\mathbf{s} \times (\mathbf{u}' + \mathbf{u}) - \cos(\theta/2)(\mathbf{u}' - \mathbf{u})\\ -i\sin(\theta/2)\mathbf{s} \cdot (\mathbf{u}' - \mathbf{u}) = 0.\end{aligned} \tag{15}$$

Eq.(15) separates into a vector equation

$$\sin(\theta/2)\mathbf{s} \times (\mathbf{u}' + \mathbf{u}) - \cos(\theta/2)(\mathbf{u}' - \mathbf{u}) = 0. \tag{16}$$

and a scalar equation $\sin(\theta/2)\mathbf{s} \cdot (\mathbf{u}' - \mathbf{u}) = 0$. The vector equation is the well-known Rodrigues' equation and the scalar equation indicates that the rotation axis $\mathbf{s}$ is perpendicular to the vector $(\mathbf{u}' - \mathbf{u})$.

Taking the cross product of both sides of (16) with $(\mathbf{v}' - \mathbf{v})$ yields

$$\mathbf{p} = \tan(\frac{\theta}{2})\mathbf{s} = \frac{(\mathbf{v}' - \mathbf{v}) \times (\mathbf{u}' - \mathbf{u})}{(\mathbf{v}' - \mathbf{v}) \cdot (\mathbf{u}' + \mathbf{u})}. \tag{17}$$

Note that in obtaining (17) we have used the fact that $\mathbf{s} \cdot (\mathbf{v}' - \mathbf{v}) = 0$ which can be derived from (14), see also Bottema and Roth (1979). Thus the orientation relationship is specified by

$$\mathbf{q} = (1 + i\mathbf{p})/(\sqrt{1 + |\mathbf{p}|^2}),$$

where $|\mathbf{p}|$ is the length of the vector $\mathbf{p}$.

To obtain the positional relationship $\mathbf{q}^0$, the location for one point of the object is required. Let $(\mathbf{x} : \mathbf{x}')$ be a pair of vectors for the locations of a given point relative to the two frames. Then $\mathbf{q}^0$ is given by (10).

For point measurements, a minimum number of three non-collinear points are required to determine both $\mathbf{q}$ and $\mathbf{q}^0$. This is evident by the fact that two direction vectors can be constructed to determine $\mathbf{q}$ and one of the three points can be used to further determine $\mathbf{q}^0$.

3.2 Calibration with Information only in the World Frame

Given a set of point locations in the world frame, any transformation $(\mathbf{q}, \mathbf{q}^0)$ can be used to map these locations to the natural frame. Therefore, the uniqueness of the relationship $(\mathbf{q}, \mathbf{q}^0)$ is not a meaningful topic of discussion. The focus here is on how to select a natural frame for a given set of point locations in the world frame. We have obtained formulas for uniquely determining $(\mathbf{q}, \mathbf{q}^0)$ in terms of locations of the minimum number of points measured in both the natural frame and the world frame. These formulas can be regarded as defining $(\mathbf{q}, \mathbf{q}^0)$ as functions of the unknown point locations in the natural frame. The number of the unknowns can be considered as the "degree of freedom" of the calibration system which allows us to select a specific natural frame. In what follows we propose a special choice of natural frame for a given set of point measurements.

Three non-collinear points, $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$ measured relative to the world frame, determines a plane. The cross product of two vectors on this plane, $\mathbf{u} = \mathbf{z} - \mathbf{x}$ and $\mathbf{v} = \mathbf{y} - \mathbf{x}$, yields the unit normal for the plane:

$$\mathbf{w} = \frac{\mathbf{u} \times \mathbf{v}}{|\mathbf{u} \times \mathbf{v}|}.$$

We select the orientation of the natural frame such that its z-axis ($\mathbf{e}_3$) is parallel to the unit normal $\mathbf{w}$ and its x-axis ($\mathbf{e}_1$) is parallel to the vector $\mathbf{u}$. Thus the two pairs of direction vectors ($\mathbf{w}$, $\mathbf{e}_3$) and ($\mathbf{u}/|\mathbf{u}|$, $\mathbf{e}_1$) can be used to determine the orientation $\mathbf{q}$. Furthermore, if the point $\mathbf{x}$ is selected as the origin of the natural frame, then the position is given by $\mathbf{q}^0 = (1/2)\mathbf{q}\mathbf{x}$.

3.3 Calibration of Kinematic Relationship Between Any Two Frames

To calibrate the kinematic relationship between any two frames, we calibrate first the spatial relationship of each frame relative to the world frame and then derive the relationship between the two given frames from it. Let $(\mathbf{q}_1, \mathbf{q}_1^0)$ represent the location of one frame N_1 relative to the world frame and $(\mathbf{q}_2, \mathbf{q}_2^0)$ represent the location of another frame N_2, Figure 1. Let $\mathbf{x}$, $\mathbf{x}_1$ and $\mathbf{x}_2$ be the coordinate vectors for a point measured relative to the world frame, the frame N_1 and the frame N_2, respectively. They are related by the transformations (10) as

$$\mathbf{q}_1\mathbf{x} - \mathbf{x}_1\mathbf{q}_1 = 2\mathbf{q}_1^0, \tag{18}$$

$$\mathbf{q}_2\mathbf{x} - \mathbf{x}_2\mathbf{q}_2 = 2\mathbf{q}_2^0. \tag{19}$$

Eliminate $\mathbf{x}$ from these two equations to obtain

$$(\mathbf{q}_2\mathbf{q}_1^*\mathbf{x}_1 - \mathbf{x}_2\mathbf{q}_2\mathbf{q}_1^*) = 2(\mathbf{q}_2^0\mathbf{q}_1^* - \mathbf{q}_2(\mathbf{q}_1^0)^*),$$

The pair of elements $(\mathbf{q}, \mathbf{q}^0)$ representing the transformation from N_1 to N_2 is therefore given by

$$\mathbf{q} = \mathbf{q}_2\mathbf{q}_1^*, \quad \mathbf{q}^0 = \mathbf{q}_2^0\mathbf{q}_1^* - \mathbf{q}_2(\mathbf{q}_1^0)^*.$$

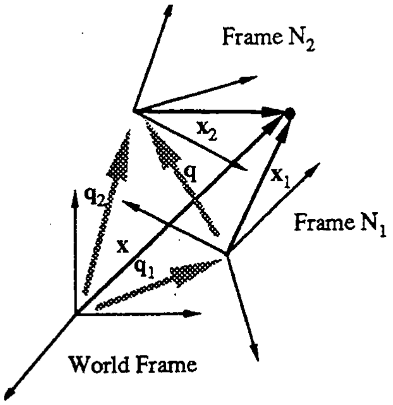


Figure 1: The composition of kinematic relationships.

4 Least Squares Solution

In the presence of measurement errors, greater accuracy in determining $(\mathbf{q}, \mathbf{q}^0)$ may be attained by measuring more than three points. This section presents an averaging process which filters the measurement errors and yields a solution that is symmetric to the input data. The process exploits the geometric nature and kinematic structure between two measured point systems to convert the kinematic localization problem to that of solving a set of linear equations. The solution to the problem is also optimal in the sense that it minimizes the squares of the position errors. Furthermore, if unit vector normals are also measured then the Clifford algebra representation facilitates definition of a metric measure of orientation error compatible with the position error.

4.1 Point Measurements

Let n vectors, $\mathbf{x}_i$ where $i = 1, 2, \cdots, n$, represent n positions of points relative to object's natural frame, and the measured values of the n position vectors be denoted as $\tilde{\mathbf{x}}_i$ with $i = 1, 2, \cdots, n$. The kinematic localization problem is to obtain the orientation $\mathbf{q}$ and the position $\mathbf{q}^0$ of the natural frame such that they best satisfy the following transformation equations:

$$\mathbf{q}\tilde{\mathbf{x}}_i - \mathbf{x}_i\mathbf{q} - 2\mathbf{q}^0 = 0, \quad i = 1, 2, \cdots, n. \tag{20}$$

Sum all n equations in (20) and we have

$$\mathbf{q}^0 = (1/2)(\mathbf{q}\tilde{\mathbf{x}}_c - \mathbf{x}_c\mathbf{q}), \tag{21}$$

where the vectors $\tilde{\mathbf{x}}_c$ and $\mathbf{x}_c$ defined by

$$\tilde{\mathbf{x}}_c = \frac{1}{n}\sum_{i=1}^{n}\tilde{\mathbf{x}}_i, \quad \mathbf{x}_c = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i, \tag{22}$$

represent the centroids of two systems of position vectors, $\tilde{\mathbf{x}}_i$ and $\mathbf{x}_i$ where $i = 1, 2, \cdots, n$.

After the substitution of (21) into (20), the kinematic localization problem becomes that of finding the rotation $\mathbf{q}$ that best satisfies the n equations

$$\mathbf{q}(\tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_c) - (\mathbf{x}_i - \mathbf{x}_c)\mathbf{q} = 0, \quad i = 1, 2, \cdots, n. \qquad (23)$$

Postmultiply both sides of (23) with $\tilde{\mathbf{x}}_i$ to obtain

$$\mathbf{q}|\tilde{\mathbf{x}}_i|^2 - \mathbf{q}\tilde{\mathbf{x}}_c\tilde{\mathbf{x}}_i - \mathbf{x}_i\mathbf{q}\tilde{\mathbf{x}}_i + \mathbf{x}_c\mathbf{q}\tilde{\mathbf{x}}_i = 0. \qquad (24)$$

Premultiply both sides of (23) with $\mathbf{x}_i$ and we have

$$\mathbf{x}_i\mathbf{q}\tilde{\mathbf{x}}_i - \mathbf{x}_i\mathbf{q}\tilde{\mathbf{x}}_c - |\mathbf{x}_i|^2\mathbf{q} + \mathbf{x}_i\mathbf{x}_c\mathbf{q} = 0. \qquad (25)$$

The subtraction of (24) from (25) yields

$$\begin{aligned} 2\mathbf{x}_i\mathbf{q}\tilde{\mathbf{x}}_i - (\mathbf{x}_i\mathbf{q}\tilde{\mathbf{x}}_c + \mathbf{x}_c\mathbf{q}\tilde{\mathbf{x}}_i) + \mathbf{q}\tilde{\mathbf{x}}_c\tilde{\mathbf{x}}_i + \mathbf{x}_i\mathbf{x}_c\mathbf{q} \\ -(|\tilde{\mathbf{x}}_i|^2 + |\mathbf{x}_i|^2)\mathbf{q} = 0. \end{aligned} \qquad (26)$$

Sum all equations of the form (25) for $i = 1, 2, \cdots, n$ and in view of (22), we obtain

$$2\sum_{i=1}^{n}(\mathbf{x}_i\mathbf{q}\tilde{\mathbf{x}}_i - \mathbf{x}_c\mathbf{q}\tilde{\mathbf{x}}_c) = \sum_{i=1}^{n}(|\tilde{\mathbf{x}}_i|^2 - |\tilde{\mathbf{x}}_c|^2 + |\mathbf{x}_i|^2 - |\mathbf{x}_c|^2)\mathbf{q}. \qquad (27)$$

The best fit orientation $\mathbf{q}$ is the solution of the multivector equation (27). Using (6), this equation can be put in the matrix form

$$[A]\mathbf{q} = \lambda\mathbf{q} \qquad (28)$$

where $[A]$ is the 4×4 symmetric matrix given by

$$[A] = 2\sum_{i=1}^{n}([x_i^+][\tilde{x}_i^-] - [x_c^+][\tilde{x}_c^-]), \qquad (29)$$

λ is a scalar given by

$$\lambda = -\sum_{i=1}^{n}(|\tilde{\mathbf{x}}_i|^2 - |\tilde{\mathbf{x}}_c|^2 + |\mathbf{x}_i|^2 - |\mathbf{x}_c|^2). \qquad (30)$$

The matrices in (29), $[\tilde{x}_i^-]$, $[x_i^+]$ and $[\tilde{x}_c^-]$, $[x_c^+]$ are skew-symmetric matrices obtained from (5) by replacing $\mathbf{x}$ with $\tilde{\mathbf{x}}_i$, $\mathbf{x}_i$, $\tilde{\mathbf{x}}_c$ and $\mathbf{x}_c$, respectively. The matrix $[A]$ is symmetric since the matrix products in (29) commute. Since $\mathbf{q}$ can not be a zero vector, the matrix $[A] - \lambda[I]$, where $[I]$ is a 4×4 identity matrix, must be singular. This indicates that λ is also an eigenvalue of the symmetric matrix $[A]$ and $\mathbf{q}$ is the corresponding eigenvector.

In the remainder of this section we prove that the kinematic relationship $(\mathbf{q}, \mathbf{q}^0)$ obtained in the above fashion is also the least squares solution to the set of n equations (20), which can be put in matrix form as:

$$[P_i]\mathbf{q}^* - 2\mathbf{q}^0 = 0, \quad i = 1, 2, \cdots, n. \qquad (31)$$

where $[P_i]$ is a 4×4 skew symmetric matrix given by

$$[P_i] = [\tilde{x}_i^+] - [x_i^-]. \qquad (32)$$

The least squares solution to (31) is to minimize the sum of the error squares

$$E_x = \sum_{i=1}^{n}([P_i]\mathbf{q}^* - 2\mathbf{q}^0)^T([P_i]\mathbf{q}^* - 2\mathbf{q}^0), \qquad (33)$$

by variation of $\mathbf{q}^*$ and $\mathbf{q}^0$ subject to (8) and (11), i.e.

$$(\mathbf{q}^*)^T\mathbf{q}^* = 1, \quad (\mathbf{q}^*)^T\mathbf{q}^0 = 0. \qquad (34)$$

These two conditions, however, are not real constraints on the problem. The condition $(\mathbf{q}^*)^T\mathbf{q}^0 = 0$ is implied by the skew symmetry of $[P_i]$, and the optimal solution $(\mathbf{q}^*, \mathbf{q}^0)$ is proportional to the length $\sqrt{(\mathbf{q}^*)^T\mathbf{q}^*}$ since $\mathbf{q}^0$ is a linear function of the components of $\mathbf{q}$ as indicated by (9). Therefore, the localization problem is reduced to a unconstrained least squares problem.

Differentiate E_x first with respect to $\mathbf{q}^0$ and we have

$$-4\sum_{i=1}^{n}([P_i]\mathbf{q}^* - 2\mathbf{q}^0) = 0.$$

This leads to

$$\mathbf{q}^0 = (1/2)[P_c]\mathbf{q}^*, \qquad (35)$$

where the matrix $[P_c] = \frac{1}{n}\sum_{i=1}^{n}[P_i]$. It is obvious that (35) is the matrix form of (21).

Now differentiate E_x with respect to $\mathbf{q}^*$ to obtain

$$2\sum_{i=1}^{n}[P_i]^T([P_i]\mathbf{q}^* - 2\mathbf{q}^0) = 0. \qquad (36)$$

Substitute (35) into (36) and we have, after rewriting the result in multivector form:

$$2\sum_{i=1}^{n}(\tilde{\mathbf{x}}_i\mathbf{q}^*\mathbf{x}_i - \tilde{\mathbf{x}}_c\mathbf{q}^*\mathbf{x}_c) = \sum_{i=1}^{n}(|\tilde{\mathbf{x}}_i|^2 - |\tilde{\mathbf{x}}_c|^2 + |\mathbf{x}_i|^2 - |\mathbf{x}_c|^2)\mathbf{q}^*. \qquad (37)$$

Taking conjugation on both sides of (37) yields the desired (27).

In view of (35) and (36), it can be shown that the minimum error $E_x = 0$. This completes the proof that the result of the averaging process outlined by (21), (23), (26) and (27) is the least squares solution.

4.2 Including Normal Vector Measurements

Let the measured values of m unit vectors $\mathbf{u}_j$ be denoted as $\tilde{\mathbf{u}}_j$ where $j = 1, 2, \cdots, m$. We seek to find $\mathbf{q}$ and

$\mathbf{q}^0$ such that they best satisfy both the point and the direction equations:

$$\text{for points: } [P_i]\mathbf{q}^* - 2\mathbf{q}^0 = 0, \quad i = 1, 2, \cdots, n \qquad (38)$$

$$\text{for directions: } [D_j]\mathbf{q}^* = 0, \quad j = 1, 2, \cdots, m. \qquad (39)$$

In (38), the matrix $[P_i]$ is given by (32) and $[D_j] = [\tilde{u}_j^+] - [u_j^-]$ where $[\tilde{u}_j^+]$, $[u_j^-]$ are obtained from (5) by replacing x with $\tilde{u}_j$, u_j, respectively.

The least squares solution to (38) and (39) is the pair $\mathbf{q}^*$ and $\mathbf{q}^0$ that minimizes the sum:

$$\begin{aligned} E = \textstyle\sum_{i=1}^{n} \alpha_i([P_i]\mathbf{q}^* - 2\mathbf{q}^0)^T([P_i]\mathbf{q}^* - 2\mathbf{q}^0) \\ + \textstyle\sum_{j=1}^{m} \beta_j([D_j]\mathbf{q}^*)^T([D_j]\mathbf{q}^*), \end{aligned} \qquad (40)$$

where α_i and β_j are the weighting factors.

The same procedure as the localization based on point measurements is applied to reduce the problem to that of solving the following linear equations

$$[B]\mathbf{q} = \lambda\mathbf{q} \qquad (41)$$

where $[B]$ is a 4×4 symmetric matrix given by

$$[B] = 2\sum_{i=1}^{n} \alpha_i([x_i^+][\tilde{x}_i^-] + [x_c^+][\tilde{x}_c^-]) + 2\sum_{j=1}^{m} \beta_j[u_j^+][\tilde{u}_j^-], \qquad (42)$$

and λ is a scalar given by

$$\lambda = -\sum_{i=1}^{n} \alpha_i(|\tilde{\mathbf{x}}_i|^2 - |\tilde{\mathbf{x}}_c|^2 + |\mathbf{x}_i|^2 - |\mathbf{x}_c|^2) - \sum_{j=1}^{m} \beta_j(|\tilde{\mathbf{u}}_j|^2 + |\mathbf{u}_j|^2). \qquad (43)$$

The vectors $\tilde{\mathbf{x}}_c$ and $\mathbf{x}_c$ defined by

$$\tilde{\mathbf{x}}_c = \frac{\sum_{i=1}^{n} \alpha_i \tilde{\mathbf{x}}_i}{\sum_{i=1}^{n} \alpha}, \quad \mathbf{x}_c = \frac{\sum_{i=1}^{n} \alpha_i \mathbf{x}_i}{\sum_{i=1}^{n} \alpha}, \qquad (44)$$

are the weighted centroids of points measured relative to the world frame and to the natural frame, respectively.

Conclusion

This paper develops simple algorithms for calibrating the kinematic relationships among the CAD models of objects based on a Clifford algebra. This formulation converts the kinematic localization problem into that of solving a set of multivector equations. It facilitates definition of a metric measure of orientation error compatible with the position error. The geometric and kinematic structures of two measured point systems are exploited in an averaging process which reduces the problem to that of solving four linear equations. The solution to the problem is symmetric to input data and minimizes the squares of position and orientation errors. The averaging process presented in this paper is directly applicable to localization problems in computer vision.

Acknowledgement

The support by US Army Research Office grant DAAL03-90-G-0005 is gratefully acknowledged.

References

[1] Bottema, O. and Roth B. 1979, *Theoretical Kinematics*. North Holland Publ., Amsterdam.

[2] Clifford, W. K. 1876, On the Classification of Geometric Algebras. *Mathematical Papers* (edited by Tucker, R.). London: Macmillan, 1882, 658pp.

[3] Faugeras, O. D. and Herbert, M. 1986, The representation, recognition, and locating of 3-D objects. *Int'l J. of Robotics and Res.*, 5(3):27-52.

[4] Grimson, W. E. and Lozano-Perez, T. 1984, Model-based recognition and localization from sparse range or tactile data. *Int'l J. of Robotics Res.*, 3(3):3-35.

[5] Ge, Q. J. 1990, *Kinematic Constraints as Algebraic Manifolds in the Clifford Algebra of Projective Three Space*, Ph.D. Dissertation, Dept. of Mech. Eng., University of California, Irvine.

[6] Grossman, D. D. and Taylor, R. H. 1978, Interactive generation of object models with a manipulator. *IEEE Trans. Syst. Man and Cybern.* SMC-8(9):667-679.

[7] Gunnarsson, K. T. and Prinz, F. B. 1987, CAD model-based localization of parts in manufacturing. *Computer*, August. pp. 66-74.

[8] Hestenes, D. and Sobczyk, G. 1984, *Clifford Algebra to Geometric Calculus*. Fundamental Theories of Physics, D. Reidel Pub. Co., Dordrecht, pp 314.

[9] Ishii, M., Sakane, S., Mikami, Y. and Kakikura, M. 1987, A 3-D sensor system for teaching robot paths and environment. *Int'l J. of Robotics Res.*, 6(2):45-59.

[10] McCarthy, J. M. 1990, *Introduction to Theoretical Kinematics*. MIT Press, Cambridge, Massachusetts, 130pp.

[11] Ravani, B. 1988, World Modeling for CAD Based Robot Programming and Simulation, *CAD Based Programming for Sensory Robots*, pp. 67-89, NATO ASI Series Vol. F50, edited by B. Ravani.

Computation of Spatial Displacements from Redundant Geometric Features

Q. J. Ge
Assistant Professor,
Department of Mechanical Engineering,
State University of New York
at Stony Brook,
Stony Brook, NY 11794

B. Ravani
Professor,
Computer Integrated Design and
Manufacturing Laboratory,
Department of Mechanical and
Aeronautical Engineering,
University of California-Davis,
Davis, CA 95616

This paper follows a previous one on the computation of spatial displacements (Ravani and Ge, 1993). The first paper dealt with the problem of computing spatial displacements from a minimum number of simple features of points, lines, planes, and their combinations. The present paper deals with the same problem using a redundant set of the simple geometric features. The problem for redundant information is formulated as a least squares problem which includes all simple features. A Clifford algebra is used to unify the handling of various feature information. An algorithm for determining the best orientation is developed which involves finding the eigenvector associated with the least eigenvalue of a 4 × 4 symmetric matrix. The best translation is found to be a rational cubic function of the best orientation. Special cases are discussed which yield the best orientation in closed form. In addition, simple algorithms are provided for automatic generation of body-fixed coordinate frames from various feature information. The results have applications in robot and world model calibration for off-line programming and computer vision.

Introduction

A solution of the problem of computing a spatial displacement from position data of a minimum number of simple features of points, lines, planes, and their combinations is provided by Ravani and Ge (1993). In practical applications, however, redundant features are measured to filter errors in sensor measurements and to improve reliability of the measuring system. This paper formulates this case as a least squares problem and determines the solution in closed form taking advantage of the geometric structure of the problem and considering different features of points, lines, planes and their combinations.

The problem of computing a spatial displacement has been studied by many researchers using only point features. Most researchers used position data of three noncollinear points, see for example, Beggs (1966), Bottema and Roth (1979), Laub and Shiflett (1982), and Angeles (1986). A comparison of these methods can be found in Fenton and Shi (1990). When more than three points are considered, least squares approximation methods are commonly adopted. Spoor and Veldpaus (1980) developed a least squares method for computing spatial displacements of which rotations were represented by orthonormal matrices. The orthonormality of rotation matrices (or rigidity) were enforced through the use of Lagrange multipliers. Although not explicitly stated, the best translation vector was obtained as the difference between the centroid of the coordinates in one system and the displaced centroid of the coordinates in the other system. Computation of the best rotation matrix was more involved including the solution of an eigenvector problem of a 3 × 3 symmetric matrix and matrix multiplication. But the solution was of closed-form and no iteration was required. Horn (1987) presented a more general least squares method which included the scaling factor (for applications in photogrammetry) and obtained the same results for the best translation as Spoor and Veldpaus (1980) but with a clearer geometric interpretation. He also showed that if unit quaternions were used to represent rotations, the best rotation was the quaternion obtained as the eigenvector associated with the most positive eigenvalue of a 4 × 4 symmetric matrix. All the above works, however, have only dealt with computation of displacements from point features and have not included other simple geometric features of lines, planes, and their combinations with points.

Ravani and Ge (1993) developed a general framework for computing spatial displacements from position data of simple geometric features which include not only points but also lines, planes and their combinations with points. They discussed issues related to uniqueness of the computation and minimum number of required features and observed that for the duality between points and planes to be valid, orientations of these features need be considered. They concluded that if orientation information was not specified than specification of four (rather than three) points features was necessary for computation of a unique displacement. They represented simple features of points, lines, and planes by multivectors and provided equations for displacements of these features using a Clifford algebra of multivectors.

This paper builds on the work of Ravani and Ge (1993) to

Contributed by the Design Automation Committee for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received Oct. 1993; revised May 1994. Associate Technical Editor: D. A. Hoeltzel.

study the problem of computing spatial displacements from redundant position data of point, line, and plane features and their combinations. Here orientation is not an important issue since redundant rather than minimum position data is considered. A least squares approach is adopted and all simple features are represented in homogeneous coordinates and are manipulated as multivectors. The use of homogeneous coordinates allows representations of the features that are independent of the measuring techniques employed. An algorithm for determining the best orientation is then developed which involves finding the eigenvector associated with the least eigenvalue of a 4 × 4 symmetric matrix. The best translation is found to be a rational cubic function of the best orientation. Special cases are also discussed which yield the best orientation in closed form. In addition, simple algorithms are developed for automatic generation of body-fixed coordinate frames from various feature information.

In practical applications, the position data of all simple features of points, lines, and planes can be determined, by using feature extraction algorithms, from visual data generated by computer vision hardware. Point and line features can also be measured directly by using, for example, a touch-trigger probe and a theodolite, respectively. This method for computing spatial displacements from point features has found a number of applications in various fields. In robotics and automation, it can be used to estimate the pose (position and orientation) of a workpiece to be grasped by a robot (Chen et al., 1980) and to determine a part location in CAD model-based flexible manufacturing (Gunnarsson and Prinz, 1987); in artificial intelligence, it can be used for object identification and localization (Grimson and Lozano-Perez, 1984) and for rigid-body motion estimation (Luo and Yang, 1990); in biomechanics, it can be used for kinematic analysis of bone movements (Spoor and Veldpaus, 1980); and in photogrammetry it can be used to recover the transformation between two coordinate systems (Horn, 1987). The results in this paper, in addition to their theoretical interest in computational geometry of motions, facilitate the use of measuring systems for the aforementioned applications that can handle not only point features but also lines and plane features.

The outline of the paper is as follows. We first formulate the problems of computing spatial displacements from points, lines, and planes, individually. We then put them together to obtain a formulation that applies to all cases. The result is a constrained least squares problem and a Lagrange multiplier technique is employed for the solution. Special cases are then discussed from which closed-form solutions are obtained. The last section presents simple algorithms for automatic generation of body-fixed coordinate frames.

1 Preliminaries

This section gives a brief account on representations of simple features of points, lines, planes and uses a Clifford algebra to represent displacements of these simple features.

1.1 Representations of Simple Features. The position data of a point feature can be obtained directly by using a coordinate measuring machine; it can also be computed from the position data of line and plane features, as the intersection of a line with a plane or the intersection of three planes. The location of a line feature can be determined from the position data of two distinct points, or a point and a direction, or two planes. The location of a plane feature is determined by measuring the normal direction and the distance from the origin. It can also be determined indirectly by measuring a point and a line or three points on the plane.

In projective geometry, points and planes are represented by homogeneous coordinates and lines are represented by Plücker coordinates so that the principle of duality can be applied when manipulating these features. Ravani and Ge (1993) used a Clifford algebra of projective three-space and associated point, line, and plane features with multivectors of rank 1, 2, and 3. In this way, the geometric operations such as meet and joint of these features can be carried out algebraically by using the wedge product and dual operation of multivectors. This allows for conversion of various representations of point, line, and plane features (due to different measuring techniques) to their respective standard homogeneous form. This facilitates unified handling of these features that are independent of the measuring techniques.

In what follows, points are represented by homogeneous coordinates of the form $(\mathbf{x}, 1)$ where $\mathbf{x} = (x_1, x_2, x_3)$ are the Cartesian coordinates; lines are represented by the Plücker coordinates of the form $(\mathbf{u}, \mathbf{u}^0)$ where $\mathbf{u} = (u_1, u_2, u_3)$ is a unit vector along the line and $\mathbf{u}^0 = (u_1^0, u_2^0, u_3^0)$ is the moment of $\mathbf{u}$ about the origin; and planes are represented by homogeneous coordinates of the form $\mathbf{A} = (\mathbf{a}, a_4)$ where $\mathbf{a} = (a_1, a_2, a_3)$ is the unit vector normal to the plane and a_4 is the distance of the plane from the origin.

1.2 Spatial Displacements of Simple Features. Let P and $\bar{P}$ denote, respectively, two coordinate systems representing two distinct positions of a rigid body in space. The displacement from P to $\bar{P}$ is a combination of a rotation and a translation. The rotation about an axis S with an angle θ is given by a set of four numbers, $\mathbf{q} = (q_1, q_2, q_3, q_4)$ (called the *Euler parameters*) where:

$$q_1 = s_1 \sin\left(\frac{\theta}{2}\right), \quad q_2 = s_2 \sin\left(\frac{\theta}{2}\right),$$
$$q_3 = s_3 \sin\left(\frac{\theta}{2}\right), \quad q_4 = \cos\left(\frac{\theta}{2}\right),$$

and $\mathbf{s} = s_1\mathbf{e}_1 + s_2\mathbf{e}_2 + s_3\mathbf{e}_3$ is a unit vector along S. The translation is given by another set of four numbers, $\mathbf{q} = (q_1^0, q_2^0, q_3^0, q_4^0)$ where

$$\begin{bmatrix} q_1^0 \\ q_2^0 \\ q_3^0 \\ q_4^0 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 0 & -d_3 & d_2 & d_1 \\ d_3 & 0 & -d_1 & d_2 \\ -d_2 & d_1 & 0 & d_3 \\ -d_1 & -d_2 & -d_3 & 0 \end{bmatrix}\begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \tag{1}$$

and $\mathbf{d} = d_1\mathbf{e}_1 + d_2\mathbf{e}_2 + d_3\mathbf{e}_3$ is the translation vector from P to $\bar{P}$. The pair of vectors $\mathbf{q}$ and $\mathbf{q}^0$ satisfy the relation

$$\mathbf{q}^T\mathbf{q}^0 = q_1q_1^0 + q_2q_2^0 + q_3q_3^0 + q_4q_4^0 = 0 \tag{2}$$

and are called the *Study vectors* (Bottema and Roth, 1979). Note that the Euler parameters are normalized such that

$$\mathbf{q}^T\mathbf{q} = q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1. \tag{3}$$

A special algebra, called the *Clifford algebra*, is used by Ravani and Ge (1993) to manipulate the study vectors $(\mathbf{q}, \mathbf{q}^0)$ and the coordinates of simple geometric features of points, lines, and planes. Clifford algebra is an associative algebra introduced by Clifford (1876) for manipulating not only vectors (representing points) but also multivectors (representing lines and planes). In this algebra, $\mathbf{q}$ is represented by an even element (a combination of a bivector and a scalar):

$$\mathbf{q} = (s_1\mathbf{e}_3\mathbf{e}_2 + s_2\mathbf{e}_1\mathbf{e}_3 + s_3\mathbf{e}_2\mathbf{e}_1)\sin\left(\frac{\theta}{2}\right) + \cos\left(\frac{\theta}{2}\right) \tag{4}$$

where the geometric products of the unit vectors are defined as:

$$\mathbf{e}_m\mathbf{e}_n = -\mathbf{e}_n\mathbf{e}_m, \quad \text{for } m \neq n, \quad \mathbf{e}_1^2 = \mathbf{e}_2^2 = \mathbf{e}_3^2 = 1.$$

In terms of the geometric product, the definition (1) becomes

$$\mathbf{q}^0 = (1/2)\mathbf{dq}. \tag{5}$$

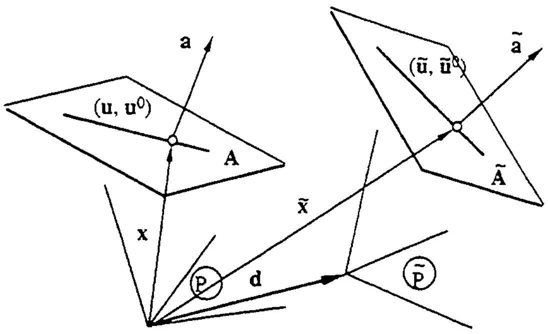

a
$(\mathbf{u}, \mathbf{u}^0)$
A
x
P
d
$\tilde{\mathbf{x}}$
$(\tilde{\mathbf{u}}, \tilde{\mathbf{u}}^0)$
$\tilde{\mathbf{a}}$
$\tilde{A}$
$\tilde{P}$

Fig. 1 A point, a line, and a plane at two different positions P and $\tilde{P}$

The "square" of a vector $\mathbf{x} = x_1\mathbf{e}_1 + x_2\mathbf{e}_2 + x_3\mathbf{e}_3$, $\mathbf{x}^2$, is simply $\mathbf{xx} = |\mathbf{x}|^2$ where $|\mathbf{x}| = (x_1^2 + x_2^2 + x_3^2)^{1/2}$ is the Euclidean norm of $\mathbf{x}$. More details about Clifford algebra can be found in Hestenes (1986) and McCarthy (1990).

At the position P, the homogeneous coordinates of a point, a line, and the polar point of a plane are denoted by $(\mathbf{x}, 1)$, $(\mathbf{u}, \mathbf{u}^0)$, and $(\mathbf{a}, a_4)$, respectively; and at the position $\tilde{P}$, the homogeneous coordinates of the corresponding features are denoted by $(\tilde{\mathbf{x}}, 1)$, $(\tilde{\mathbf{u}}, \tilde{\mathbf{u}}^0)$, and $(\tilde{\mathbf{a}}, \tilde{a}_4)$, respectively, see Fig. 1. The displacements $(\mathbf{q}, \mathbf{q}^0)$ of these features from P and $\tilde{P}$ are given by the following multivector equations (Ravani and Ge, 1993).

For points features, we have

$$\tilde{\mathbf{x}}\mathbf{q} - \mathbf{q}\mathbf{x} - 2\mathbf{q}^0 = 0; \tag{6}$$

For line features, we have

$$\tilde{\mathbf{u}}\mathbf{q} - \mathbf{q}\mathbf{u} = 0; \tag{7}$$

$$\tilde{\mathbf{u}}^0\mathbf{q} - \mathbf{q}\mathbf{u}^0 + \tilde{\mathbf{u}}\mathbf{q}^0 - \mathbf{q}^0\mathbf{u} = 0; \tag{8}$$

For plane features, we have

$$\tilde{\mathbf{a}}\mathbf{q} - \mathbf{q}\mathbf{a} = 0; \tag{9}$$

$$\tilde{a}_4 - a_4 - 2(\mathbf{qa}) \cdot \mathbf{q}^0 = 0. \tag{10}$$

The product of a vector $\mathbf{x} = x_1\mathbf{e}_1 + x_2\mathbf{e}_2 + x_3\mathbf{e}_3$ with the even element $\mathbf{q}$ defined by (4) can be conveniently expressed in the matrix form

$$\mathbf{xq} = [x^+]\mathbf{q}, \quad \mathbf{qx} = [x^-]\mathbf{q},$$

where $[x^+]$ and $[x^-]$ are 4×4 skew-symmetric matrices given by

$$[x^+] = \begin{bmatrix} 0 & -x_3 & x_2 & x_1 \\ x_3 & 0 & -x_1 & x_2 \\ -x_2 & x_1 & 0 & x_3 \\ -x_1 & -x_2 & -x_3 & 0 \end{bmatrix},$$

$$[x^-] = \begin{bmatrix} 0 & x_3 & -x_2 & x_1 \\ -x_3 & 0 & -x_1 & x_2 \\ x_2 & -x_1 & 0 & x_3 \\ -x_1 & -x_2 & -x_3 & 0 \end{bmatrix}. \tag{11}$$

The product of these two matrices commutes, i.e., $[x^-][x^+] = [x^+][x^-]$. The product of a vector $\tilde{\mathbf{x}}$, an even element $\mathbf{q}$, and another vector $\mathbf{x}$ is given by

$$\tilde{\mathbf{x}}\mathbf{q}\mathbf{x} = -[\tilde{x}^+][x^-]\mathbf{q}, \tag{12}$$

where $[\tilde{x}^+]$ is obtained from $[x^+]$ by replacing $\mathbf{x}$ with $\tilde{\mathbf{x}}$.

2 Sums of Squares of Errors

This section uses Clifford-algebra equations for displacements of points, lines, and plane features to obtain the sums of squares of errors for these features. The results are then combined together to form the sum of squares of errors for all three types of features.

2.1 Sum of Squares of Errors for Point Features. Let the coordinate of n points at the two positions P and $\tilde{P}$ be given by the Cartesian vectors $\mathbf{x}_i$ and $\tilde{\mathbf{x}}_i$, where $i = 1, 2, \ldots, n$. The displacement of each of these point features is given by the multivector equation of the form (6). Due to measurement error, it is impossible to find $\mathbf{q}$ and $\mathbf{q}^0$ such that (6) is satisfied for each point. Instead there will be in general a residual error:

$$\mathbf{E}_i = \tilde{\mathbf{x}}_i\mathbf{q} - \mathbf{q}\mathbf{x}_i - 2\mathbf{q}^0. \tag{13}$$

An overall measure of the error as a function of $\mathbf{q}$ and $\mathbf{q}^0$, denoted by $E_x(\mathbf{q}, \mathbf{q}^0)$, is given by the sum of squares of these errors

$$E_x(\mathbf{q}, \mathbf{q}^0) = \sum_{i=1}^{n} \mathbf{E}_i \cdot \mathbf{E}_i.$$

Expand the above to obtain

$$E_x(\mathbf{q}, \mathbf{q}^0) = -2\sum_{i=1}^{t} (\tilde{\mathbf{x}}_i\mathbf{q}) \cdot (\mathbf{q}\mathbf{x}_i) - 4n(\tilde{\mathbf{x}}_c\mathbf{q} - \mathbf{q}\mathbf{x}_c) \cdot \mathbf{q}^0 + 4n\mathbf{q}^0 \cdot \mathbf{q}^0 + \left(\sum_{i=1}^{n} |\tilde{\mathbf{x}}_i|^2 + \sum_{i=1}^{n} |\mathbf{x}_i|^2\right)\mathbf{q} \cdot \mathbf{q}$$

where

$$\mathbf{x}_c = \frac{1}{n}\sum_{i=1}^{n} \tilde{\mathbf{x}}_i, \quad \mathbf{x}_c = \frac{1}{n}\sum_{i=1}^{n} \mathbf{x}_i \tag{14}$$

are the centroids of two systems of position vectors and $|\tilde{\mathbf{x}}_i|$, $|\mathbf{x}_i|$ are the Euclidean norms of $\tilde{\mathbf{x}}_i$, $\mathbf{x}_i$, respectively. In matrix form, the error function is given by

$$E_x(\mathbf{q}, \mathbf{q}^0) = \mathbf{q}^T[A_x]\mathbf{q} + \mathbf{q}^T[B_x]\mathbf{q}^0 + 4n(\mathbf{q}^0)^T\mathbf{q}^0 \tag{15}$$

where $[A_x]$ is a 4×4 symmetric matrix given by

$$[A_x] = 2\sum_{i=1}^{n} [\tilde{x}_i^+][x_i^-] + \left(\sum_{i=1}^{n} |\tilde{\mathbf{x}}_i|^2 + \sum_{i=1}^{n} |\mathbf{x}_i|^2\right)[I] \tag{16}$$

and $[B_x]$ is a 4×4 skew-symmetric matrix given by

$$[B_x] = -4n([\tilde{x}_c^+] - [x_c^-]). \tag{17}$$

In the above, $[I]$ denotes the 4×4 identity matrix; the skew-symmetric matrices $[\tilde{x}_i^+]$, $[\tilde{x}_c^+]$, $[x_i^-]$, and $[x_c^-]$ are defined by (11).

2.2 Sum of Squares of Errors for Line Features. Let the coordinates of m lines at the two positions P and $\tilde{P}$ be given by the Plücker vectors $(\mathbf{u}_i, \mathbf{u}_i^0)$ and $(\tilde{\mathbf{u}}_i, \tilde{\mathbf{u}}_i^0)$, respectively, where $i = 1, 2, \ldots, m$. The displacement of a line $(\mathbf{u}, \mathbf{u}^0)$ from P to $\tilde{P}$ is given by (7) and (8). Since the position data are imperfect, there will be errors in both orientation and position:

$$\mathbf{E}_{oi} = \tilde{\mathbf{u}}_i\mathbf{q} - \mathbf{q}\mathbf{u}_i$$

and

$$\mathbf{E}_{pi} = \tilde{\mathbf{u}}_i^0\mathbf{q} - \mathbf{q}\mathbf{u}_i^0 + \tilde{\mathbf{u}}_i\mathbf{q}^0 - \mathbf{q}^0\mathbf{u}_i. \tag{18}$$

An overall measure of the orientation error is given by

$$E_o(\mathbf{q}) = \sum_{i=1}^{m} \mathbf{E}_{oi} \cdot \mathbf{E}_{oi}$$

which can be expanded to obtain

$$E_o(\mathbf{q}) = -2\sum_{i=1}^{m} (\tilde{\mathbf{u}}_i\mathbf{q}) \cdot (\mathbf{q}\mathbf{u}_i) + \left(\sum_{i=1}^{m} |\tilde{\mathbf{u}}_i|^2 + \sum_{i=1}^{m} |\mathbf{u}_i|^2\right)\mathbf{q} \cdot \mathbf{q}.$$

In matrix form the equation above is given by

$$E_o(\mathbf{q}) = \mathbf{q}^T[A_o]\mathbf{q} \tag{19}$$

where

$$[A_o] = 2\sum_{i=1}^{m} [\tilde{u}_i^+][u_i^-] + \left(\sum_{i=1}^{m} |\tilde{\mathbf{u}}_i|^2 + \sum_{i=1}^{m} |\mathbf{u}_i|^2\right)[I] \quad (20)$$

is a 4 × 4 symmetric. matrix.

An overall measure of the position error is given by

$$E_p(\mathbf{q}, \mathbf{q}^0) = \sum_{i=1}^{m} \mathbf{E}_{pi} \cdot \mathbf{E}_{pi}$$

which can be expanded and put into the following matrix form

$$E_p(\mathbf{q}, \mathbf{q}^0) = \mathbf{q}^T[A_p]\mathbf{q} + \mathbf{q}^T[B_u]\mathbf{q}^0 + (\mathbf{q}^0)^T[C_u]\mathbf{q}^0.$$

The coefficient matrices $[A_p]$, $[C_u]$ are symmetric matrices given by

$$[A_p] = 2\sum_{i=1}^{m} \left[(\tilde{u}_i^0)^+\right]\left[(u_i^0)^-\right] + \left(|\tilde{\mathbf{u}}_i^0|^2 + |\tilde{\mathbf{u}}_i|^2 + |\mathbf{u}_i^0|^2 + |\mathbf{u}_i|^2\right)[I],$$

$$[C_u] = 2\sum_{i=1}^{m} [\tilde{u}_i^+][u_i^-].$$

and the matrix $[B_u]$ has both symmetric and skew-symmetric parts, $[B_+]$ and $[B_-]$, which are given by

$$[B_+] = 2\sum_{i=1}^{m} \left(\left[(\tilde{u}_i^0)^+\right][u_i^-] + [\tilde{u}_i^+]\left[(u_i^0)^-\right]\right),$$

$$[B_-] = -2\sum_{i=1}^{m} \left(\left[(\tilde{u}_i^0)^+\right][u_i^+] + \left[(u_i^0)^-\right][u_i^-]\right).$$

An overall measure of both the position and orientation error is given by the following combination:

$$\begin{aligned} E_u(\mathbf{q}, \mathbf{q}^0) &= \beta E_o(\mathbf{q}) + E_p(\mathbf{q}, \mathbf{q}^0) \\ &= \mathbf{q}^T[A_u]\mathbf{q} + \mathbf{q}^T[B_u]\mathbf{q}^0 + (\mathbf{q}^0)^T[C_u]\mathbf{q}^0 \quad (21) \end{aligned}$$

where β is a positive weight and $[A_u] = \beta[A_o] + [A_p]$.

2.3 Sum of Squares of Errors for Plane Features. Let $\mathbf{A}_i = (\mathbf{a}_i, a_4)$ and $\mathbf{A}_i = (\tilde{\mathbf{a}}, \tilde{a}_4)$ $(i = 1, 2, \ldots, k)$ be the coordinate of k planes at P and $\tilde{P}$, respectively. Similar to the measure of orientation errors of line features, an overall measure of the orientation errors of the plane features is given by $\mathbf{q}^T[A_a]\mathbf{q}$ where $[A_a]$ is obtained from (20) by replacing $\mathbf{u}_i$, $\tilde{\mathbf{u}}_i$ with $\mathbf{a}_a$, $\tilde{\mathbf{a}}_i$, respectively.

The errors in the locations of the plane features are given by (10) as

$$e_i = \tilde{a}_{4i} - a_{4i} - 2(\mathbf{q}\mathbf{a}_i) \cdot \mathbf{q}^0.$$

The square of the errors of this form results in a error function that is quartic in $(\mathbf{q}, \mathbf{q}^0)$, which is difficult to handle for the least squares solution method proposed by this paper. For this reason, we use instead the points and lines of intersection among the given planes to determine the location errors of the planes. In this way, the sum of squares of errors for plane features is reduced to a combination of the sums of squares of errors for point and line features. For the rest of this paper, we focus our attention to the least squares solution for computing a spatial displacement from redundant point and line features.

2.4 Sum of Squares of Errors for Points and Lines. Combining the results in Sections 2.1 and 2.2, we obtain the sum of squares of errors for combinations of points and lines:

$$E(\mathbf{q}, \mathbf{q}^0) = \alpha_x E_x(\mathbf{q}, \mathbf{q}^0) + \alpha_u E_u(\mathbf{q}, \mathbf{q}^0)$$

where α_x, α_u are positive weights. The substitutions of (15) and (21) into the above yields

$$E(\mathbf{q}, \mathbf{q}^0) = \mathbf{q}^T[A]\mathbf{q} + \mathbf{q}^T[B]\mathbf{q}^0 + (\mathbf{q}^0)^T[C]\mathbf{q}^0 \quad (22)$$

where $[A]$ and $[C]$ are 4 × 4 symmetric matrices given by

$$[A] = \alpha_x[A_x] + \alpha_u[A_u], \quad [C] = 4\alpha_x n[I] + \alpha_u[C_u],$$

respectively, and $[B]$ is a general 4 × 4 matrix given by

$$[B] = \alpha_x[B_x] + \alpha_u[B_u].$$

It is clear that the grand sum of squares of errors for combinations of points, lines, and planes can be put into the same form as (22).

The problem of computing spatial displacements from redundant position data of points, lines, planes, and their combinations is reduced to that of minimizing $E(\mathbf{q}, \mathbf{q}^0)$ by variation of $\mathbf{q}$ and $\mathbf{q}^0$ subject to the constraints (3) and (2), i.e.,

$$\mathbf{q}^T\mathbf{q} = 1, \quad \mathbf{q}^T\mathbf{q}^0 = 0. \quad (23)$$

3 An Algorithm for the Least Squares Solution

This section deals with the constrained minimization problem in the preceding section using Lagrange multipliers and develops a simple algorithm for solving the least squares problem.

3.1 Conditions for Minimum. The Lagrangian function to be minimized is as follows

$$L(\mathbf{q}, \mathbf{q}^0, \lambda, \lambda^0) = E(\mathbf{q}, \mathbf{q}^0) - \lambda(\mathbf{q}^T\mathbf{q} - 1) - \lambda^0(\mathbf{q}^T\mathbf{q}^0). \quad (24)$$

The conditions for minimum, i.e., $\partial L/\partial \mathbf{q} = 0$ and $\partial L/\partial \mathbf{q}^0 = 0$, lead to

$$2[A]\mathbf{q} + [B]\mathbf{q}^0 - 2\lambda\mathbf{q} - \lambda^0\mathbf{q}^0 = 0 \quad (25)$$

and

$$[B]^T\mathbf{q} + 2[C]\mathbf{q}^0 - \lambda^0\mathbf{q} = 0. \quad (26)$$

These two vector equations are linear in $\mathbf{q}$ and $\mathbf{q}^0$. Together with the two quadratic constraint equations $\mathbf{q}^T\mathbf{q} = 1$ and $\mathbf{q}^T\mathbf{q} = 0$, they constitute a system of ten equations needed to solve the ten unknowns, $\mathbf{q}$, $\mathbf{q}^0$, λ, and λ^0.

To solve these ten equations, we first represent the best translation $\mathbf{q}^0$ in terms of the best orientation $\mathbf{q}$. In view of (26), we obtain

$$\mathbf{q}^0 = -(1/2)[C]^{-1}[B]^T\mathbf{q} + (1/2)[C]^{-1}\mathbf{q}\lambda^0. \quad (27)$$

Substitute (27) into (25) to obtain the following equation:

$$4[A']\mathbf{q} - 4\lambda\mathbf{q} + \lambda^0[B']\mathbf{q} - (\lambda^0)^2[C]^{-1}\mathbf{q} = 0, \quad (28)$$

where $[A']$ and $[B']$ are symmetric matrices given by

$$[A'] = [A] - (1/4)[B][C]^{-1}[B]^T, \quad (29)$$

$$[B'] = [B][C]^{-1} + [C]^{-1}[B]^T. \quad (30)$$

The Lagrange multipliers, λ and λ^0, in (28) are obtained as homogeneous functions of $\mathbf{q}$:

$$\lambda^0 = \frac{\mathbf{q}^T[C]^{-1}[B]^T\mathbf{q}}{\mathbf{q}^T[C]^{-1}\mathbf{q}}; \quad (31)$$

$$\lambda = \frac{\mathbf{q}^T[A]\mathbf{q}}{\mathbf{q}^T\mathbf{q}} - \frac{\mathbf{q}^T[B][C]^{-1}[B]^T\mathbf{q}}{4\mathbf{q}^T\mathbf{q}} + \lambda^0\frac{\mathbf{q}^T[B][C]^{-1}\mathbf{q}}{4\mathbf{q}^T\mathbf{q}}. \quad (32)$$

The substitution of (31) and (32) into (28) results in four nonlinear equations in the components of $\mathbf{q}$ and therefore, in general, the best orientation $\mathbf{q}$ does not seem to exist in closed form. The best translation $\mathbf{q}^0$, however, is a rational cubic function of the best orientation $\mathbf{q}$, in view of (27) and (31).

3.2 A Simple Algorithm. We now develop a simple algorithm for computing the best orientation **q**. We first rewrite (28) in the following form:

$$\left[D(\lambda^0)\right]\mathbf{q} = \lambda\mathbf{q} \tag{33}$$

where $[D(\lambda^0)]$ is a symmetric matrix given by

$$\left[D(\lambda^0)\right] = [A'] + (1/4)\lambda^0[B'] - (1/4)(\lambda^0)^2[C]^{-1}. \tag{34}$$

This indicates that the best orientation **q** corresponds to one of the unit eigenvectors of the 4 × 4 symmetric matrix $[D(\lambda^0)]$, provided that λ^0 is known. Since there are four real eigenvectors for the real symmetric matrix $[D(\lambda^0)]$, there are four local optimal orientations for the least squares problem. The global optimal orientation **q** is the eigenvector associated with the least eigenvalue of $[D(\lambda^0)]$, see Appendix A.

The eigenvalues of $[D(\lambda^0)]$ are the solutions of the quartic characteristic equation

$$\det\left[D(\lambda^0) - \lambda I\right] = 0.$$

The eigenvalues can be obtained in closed form. Once the least eigenvalue λ is selected, the corresponding unit eigenvector **q** is obtained by solving the homogeneous equation

$$\left[D(\lambda^0) = \lambda I\right]\mathbf{q} = 0. \tag{35}$$

Appendix B provides a simple solution method for (35) using vector wedge product.

The solution method for obtaining the eigenvalues and eigenvectors for the 4 × 4 real symmetric matrix (34) is the core of the following simple algorithm for determining the best orientation **q**. The algorithm proceeds as follows:

(*1*) Compute two constant symmetric matrices $[A']$ and $[B']$ using (29) and (30), respectively.

(*2*) Select an initial orientation as represented by a unit quaternion **q**.

(*3*) Compute the value of $\lambda^0(\mathbf{q})$ using (31) and then determine the matrix function $[D(\lambda^0)]$ using (34).

(*4*) Find the least eigenvalue λ and the associated unit eigenvector $\mathbf{q}_d$ of the matrix $[D(\lambda^0)]$.

(*5*) Compute $\phi = \cos^{-1}(\mathbf{q}_d \cdot \mathbf{q})$ and let the new **q** be

$$(\mathbf{q} + \mathbf{q}_d)/(2\cos(\phi/2)).$$

(*6*) If the norm $|\phi|$ is sufficiently small, stop; otherwise go to step 3 and repeat the procedure.

The issue of choosing an initial orientation for the above algorithm will be discussed in the next section.

4 Special Cases

This section discusses special cases of the least squares problem when only point features or line features are considered. The kinematic and geometric insight gained by studying these special cases is then incorporated into the choice of an initial orientation for the least square algorithm developed in the preceding section.

4.1 Point Features. When only n point features are considered, the matrices $[A]$, $[B]$, and $[C]$ become $[A_x]$, $[B_x]$, and $4n[I]$, respectively. The matrix $[B_x]$ given by (17) is skew-symmetric and therefore the Lagrange multiplier λ^0 given by (31) is no longer a function of **q** but equals to zero, i.e., $\lambda^0 = 0$. Thus the least square algorithm yields the best orientations **q** in one step and no iteration is required. The solution is of closed form and is summarized as follows.

The best orientation **q** is the unit eigenvector associated with the least eigenvalue of the matrix $[A'_x] = [A_x] + \frac{1}{16n}[B_x][B_x]$. The eigenvalue problem is given by

$$[A_x]\mathbf{q} + \frac{1}{16n}[B_x][B_x]\mathbf{q} = \lambda\mathbf{q}. \tag{36}$$

This is essentially the same result as that provided by Horn (1987).

The best translation given by (27) is reduced to

$$\mathbf{q}^0 = (1/8n)[B_x]\mathbf{q} = \frac{1}{2}([\tilde{x}_c^+] - [x_c^-])\mathbf{q}. \tag{37}$$

This can be recast in the multivector form as

$$\mathbf{q}^0 = (1/2)(\tilde{\mathbf{x}}_c\mathbf{q} - \mathbf{q}_{xc}) \tag{38}$$

where $\tilde{\mathbf{x}}_c$ and $\mathbf{x}_c$ are the centroids of two systems of position vectors. This indicates the best translation $\mathbf{q}^0$ is given by the translational offset of the two centroids, a well-known result provided by Spoor and Veldpaus (1980) and Horn (1987).

Note that (36) can be written in the following multivector form after the substitution of (16) and (17):

$$-2\sum_{i=1}^{m}(\tilde{\mathbf{x}}_i\mathbf{q}\mathbf{x}_i - \tilde{\mathbf{x}}_c\mathbf{q}\mathbf{x}_c) + \sum_{i=1}^{n}\left(|\tilde{\mathbf{x}}_i|^2 + |\mathbf{x}_i|^2 - |\tilde{\mathbf{x}}_c|^2 - |\mathbf{x}_c|^2\right)\mathbf{q} = \lambda\mathbf{q}. \tag{39}$$

This will be used later for comparison with the result obtained from an averaging process.

4.2 An Averaging Process. It turns out the closed-form solution above is related to the solution obtained from an averaging process. The residual error for each point is given by (13). Sum all these errors to obtain

$$\sum_{i=1}^{n}\mathbf{E}_i = \tilde{\mathbf{x}}_c\mathbf{q} - \mathbf{q}\mathbf{x}_c - 2\mathbf{q}^0 \tag{40}$$

where $\tilde{\mathbf{x}}_c$ and $\mathbf{x}_c$ are the centroids given by (14). This means that the best translation $\mathbf{q}^0$ makes the sum of errors, $\Sigma_{i=1}^{n}\mathbf{E}_i$, vanishes.

The substitution of (38) into (13) yields

$$\mathbf{E}_i = \tilde{\mathbf{y}}_i\mathbf{q} - \mathbf{q}\mathbf{y}_i \tag{41}$$

where vectors $\tilde{\mathbf{y}}_i$ and $\mathbf{y}_i$ are measured with respect to their respective centroids:

$$\tilde{\mathbf{y}}_i = \tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_c, \quad \mathbf{y}_i = \mathbf{x}_i - \mathbf{x}_c. \tag{42}$$

Equations (41) and (42) lead to

$$\sum_{i=1}^{n}(\tilde{\mathbf{y}}_i\mathbf{E}_i - \mathbf{E}_i\mathbf{y}_i) = \sum_{i=1}^{n}\left(|\tilde{\mathbf{y}}_i|^2 + |\mathbf{y}_i|^2\right)\mathbf{q} - 2\sum_{i=1}^{n}\tilde{\mathbf{y}}_i\mathbf{q}\mathbf{y}_i. \tag{43}$$

Substituting (42) back into (43) and in view of (14), we obtain

$$\sum_{i=1}^{n}(\tilde{\mathbf{x}}_i\mathbf{E}_i - \mathbf{E}_i\mathbf{x}_i) = \sum_{i=1}^{n}\left(|\tilde{\mathbf{x}}_i|^2 + |\mathbf{x}_i|^2 - |\tilde{\mathbf{x}}_c|^2\right)\mathbf{q} - 2\sum_{i=1}^{n}(\tilde{\mathbf{x}}_i\mathbf{q}\mathbf{x}_i = \tilde{\mathbf{x}}_c\mathbf{q}\mathbf{x}_c). \tag{44}$$

The comparison of (44) with (39) results in the following interesting result

$$\sum_{i=1}^{n}(\tilde{\mathbf{x}}_i\mathbf{E}_i - \mathbf{E}_i\mathbf{x}_i) = \lambda\mathbf{q}. \tag{45}$$

In the ideal case when all data are perfect, i.e., $\mathbf{E}_i = 0$ for all $i = 1, 2, \ldots, n$, Eq. (45) yields $\lambda = 0$, This agrees with the result (48) in Appendix A.

4.2 Line Features. When only line features are considered, the Lagrange multiplier λ^0 is a function of **q** since $[B_u]$ is not skew-symmetric and $[C_u]$ is not a multiple of $[I]$. Therefore, the closed-form solution to the problem of simultaneously minimizing the orientation and position errors does not seem to exist.

However, in the special case when all lines are represented

Table 1 The position data of six points and six lines at two distinct configurations

At the initial configuration		
Points ($\mathbf{x}_i$)	Lines ($\mathbf{u}_i$)	Lines ($\mathbf{u}_i^0$)
(8.0, 6.0, 2.0)	(0.1972, 0.1267, 0.9721)	(4.9874, -3.1136, -0.6058)
(4.0, 0.0, 4.0)	(0.2248, 0.5846, 0.7795)	(-8.3797, 0.4647, 2.0687)
(2.0, 11.0, -6.0)	(0.5765, 0.1627, 0.8006)	(-2.4020, -4.8040, 2.7061)
(5.0, 6.0, 7.0)	(0.0609, 0.0000, 0.9981)	(9.9813, -0.1829, -0.6099)
(4.0, 7.0, 9.0)	(-0.1937, -0.1549, 0.9687)	(4.998649, 9.493558, 2.518699)
(4.4, 3.0, 0.0)	(0.8899, -0.1001, -0.4449)	(2.0023, 21.8033, -0.9010)
At the displaced configuration		
(-7.3801, 14.8362, -5.6198)	(0.7386, 0.1980, -0.6443)	(-6.0525, -7.3061, -9.1842)
(-8.2420, 8.3324, -9.2195)	(0.8591, 0.4543, -0.2353)	(5.4624, -13.7875, -6.6742)
(-8.5317, 12.3697, 5.2240)	(0.5169, 0.5520, -0.6542)	(-0.6225, -13.6783, -12.0351)
(-2.7462, 12.0418, -7.7921)	(0.7240, 0.0162, -0.6895)	(-6.9688, -1.8993, -7.3627)
(-0.3589, 11.5897, -8.1025)	(0.6876, -0.2805, -0.6697)	(1.2463, -1.2115, 1.7870)
(-9.5559, 10.3030, -4.7979)	(-0.6616, 0.7414, -0.1119)	(14.8335, 14.6844, 9.5872)

by points and unit vectors rather than the Plücker coordinates, the least squares solution can be found in the closed form. In this case, the equations for position errors (18) are replaced by point equations of the form (13). The resulting least squares problem is essentially the same as that of point features and a closed-form solution can be obtained.

4.4 Choice of Initial Orientation. Now turn back to the general least squares problem of computing spatial displacements from combinations of point, line, and plane features. A good choice of initial orientation is the unit eigenvector associated with the least eigenvalue of the matrix $[A']$ given by (29). This orientation is "quasi-optimal" in the sense that it minimizes the combination of errors in point measurements, and orientation errors in line and plane measurements.

4.5 An Example. We now present an example problem of computing a spatial displacement from the position data of six points and six lines at two distinct configurations. To generate the position data for these features, we first select a spatial displacement $(\mathbf{q}, \mathbf{q}^0)$ and the position data of the point and line features at the initial configuration. We then generate the position data of the given features at the displaced configuration using forward kinematics equations. The rotational component of the spatial displacement is given by a unit quaternion $\mathbf{q} = (0.466609, 0.784751, 0.190885, 0.360561)$; the translational component of the displacement is given by the vector $\mathbf{d} = (-10.0, 5.0, -5.0)$ which can be used to compute $\mathbf{q}^0$ using (5). The position data of the point and line features before and after the spatial displacement are given by Table 1).

We have implemented the algorithm presented in Section 3.2 and have found that, for the position data given in Table 1, the algorithm yields the given displacement $\mathbf{q} = (0.4666, 0.7847, 0.1908, 0.3605)$ and $\mathbf{d} = (-10.0, 5.0, -5.0)$ in one step. This is expected since all position data are perfect. To see how well the algorithm handles imperfect position data, we then add 10 percent random error to the position data of the geometric features at the displaced configuration and find that the algorithm converges to a least squares solution in no more than four iterations. For the imperfect position data shown in Table 2, the algorithm converges in two iterations and the result is presented in Table 3. In obtaining the result, we used weights $\beta = 1.0$, $\alpha_x = 1.0$, and $\alpha_u = 100.0$ Adjustment of these weights allows for change of accuracy requirement of a given set of the geometric features.

5 Automatic Generation of Natural Frames

This section presents simple geometric algorithms for automatic generation of body-fixed coordinate frames (called natural frames) from geometric features of points, lines, planes, and their combinations. This is important in applications such as world model calibration for off-line robot programming, since the computer model of each object has to be automatically updated from the sensory interaction with the actual object, see Ravani (1988). Automatic construction of natural frames from point features have been discussed previously by Angeles (1986) where he used mass distribution to locate a natural frame.

Table 2 The imperfect position data of six points and six lines at the displaced configuration

At the displaced configuration (with 10% random error)		
(-7.0906, 14.6219, -5.9716)	(0.7442, 0.2015, -0.6367)	(-5.9570, -7.2802, -9.2668)
(-7.6305, 7.6276, -10.0261)	(0.8607, 0.4553, -0.2274)	(5.5116, -13.7610, -6.6875)
(-7.9875, 13.1092, 5.3780)	(0.4758, 0.5808, -0.6604)	(-0.4700, -13.4988, -12.2117)
(-2.5652, 13.1537, -8.2065)	(0.7236, 0.0194, -0.6898)	(-6.9683, -1.8993, -7.3631)
(-0.3259, 12.4703, -7.6607)	(0.6974, -0.2970, -0.6521)	(1.2022, -1.1932, 1.8290)
(-9.1206, 9.6892, -5.1000)	(-0.6743, 0.7303.1 -0.1088)	(14.6192, 14.9200, 9.5499)

Table 3 Computing a spatial displacement from imperfect position data

Iterations	The least eigenvalue	The best orientation
1	23.078678	(0.462546, 0.787129, 0.192708, 0.359642)
2	23.085560	(0.462549, 0.787127, 0.192709, 0.359642)
	The best translation	$\mathbf{d} = 2\mathbf{q}^0\mathbf{q}^{-1} = (-10.061083, 4.947477, -4.974046)$

In what follows, the orientation of a natural frame is represented by a set of three orthonormal unit vector, denoted by $\mathbf{e}'_1$, $\mathbf{e}'_2$ and $\mathbf{e}'_3$, and the origin of the natural frame is represented by a vector denoted by $\mathbf{p}$. This section presents simple formulas for determining the origin $\mathbf{p}$ and the coordinate axes $\mathbf{e}'_1$, $\mathbf{e}'_3$. The $\mathbf{e}'_2$ axis is then given by $\mathbf{e}'_2 = \mathbf{e}'_3 \times \mathbf{e}'_1$.

5.1 Oriented Features. An oriented line is given by the *signed* normalized Plücker coordinates $\hat{\mathbf{u}} = (\mathbf{u}, \mathbf{u}^0)$. By the signed Plücker coordinates, we mean $\hat{\mathbf{u}}$ and $-\hat{\mathbf{u}}$ represent two distinct lines having the same position in space but with opposite sense of orientation. An oriented plane is given by the plane vector $\mathbf{A} = (\mathbf{a}, a_4)$ where $\mathbf{a}$ is the unit normal to the plane and a_4 is the distance from the origin to the plane. The two plane vectors $\mathbf{A}$ and $-\mathbf{A}$ represent two planes with the same position but opposite sense of orientation.

In what follows, the notion of oriented features is also applied to point features in order to determine uniquely an oriented plane from three noncollinear points. An oriented point is represented by the homogeneous coordinates of the form $\mathbf{X} = (\mathbf{x}, \sigma)$ where $\mathbf{x}$ is the Cartesian vector of the point and $\sigma = 1$ $(\sigma = -1)$ denotes the positive (negative) orientation of the point (Stolfi, 1989; Ge and Ravani, 1993). Simple formulas for computing the coordinate axes $\mathbf{e}'_3$, $\mathbf{e}'_1$ and the origin $\mathbf{p}$ are presented in Table 4.

5.2 Nonoriented Features. For all the cases presented in Table 4, if simple geometric features are specified without information about the sense of orientation, an additional simple feature may be needed to construct a natural frame. Table 5 contains all possible combinations of nonoriented features of points, lines, and planes required to specify a natural frame completely.

Let us first consider the construction of a natural frame using nonoriented point features. In this case, the direction of the unit vector $\mathbf{e}'_3$ is given by $\mathbf{g}_{no} = (\mathbf{x} \times \mathbf{y} + \mathbf{y} \times \mathbf{z} + \mathbf{z} \times \mathbf{x})$ where $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$ denote the vectors of the Cartesian coordinates of three given points (Table 4). The sense of direction for $\mathbf{e}'_3$, however, can be specified with additional information about the order of the three points (resulting in oriented points) or with an additional point (denoted as $\mathbf{t}$) which is not coplanar with the three given points. Using the point $\mathbf{t}$, the sense of direction for $\mathbf{e}'_3$ is given by sign of $(\mathbf{t} - \mathbf{p}) \cdot \mathbf{g}_{no}$ where $\mathbf{p} = (\mathbf{x} + \mathbf{y} + \mathbf{z})/3$.

All other cases listed in Table 5 can be reduced to four noncoplanar points by obtaining one or more of the following: (*1*) the intersection point between a line and a plane; (*2*)

Table 4 Formulas for natural-frame generation using oriented features

Features		$e'_3 = g/\|g\|$, $e'_1 = h/\|h\|$, and p
Three points $X = (x, \sigma_x)$ $Y = (y, \sigma_y)$ $Z = (z, \sigma_z)$	g	Normal to the plane spanned by the three points $g = (\sigma_x \sigma_y \sigma_z) g_{no}$ where $g_{no} = x \times y + y \times z + z \times x$
	h	$h = z - p$
	p	$p = (x + y + z)/3$
Two lines $\hat{u} = (u, u^0)$ $\hat{v} = (v, v^0)$	g	$g = u \times v$ is along the common perpendicular $\hat{w} = (w, w^0)$ where $w = (u \times v)/\|u \times v\|$ and $w^0 = (u \times v^0 + u^0 \times v)/\|u \times v\|$
	h	$h = u$
	p	$p = (c_u + c_v)/2$ is the midpoint of the two feet of perpendicular $c_u = u \times u^0 + [u \cdot (w \times w^0)]u$ $c_v = v \times v^0 + [v \cdot (w \times w^0)]v$
Three planes $A = (a, a_4)$ $B = (b, b_4)$ $C = (c, c_4)$	g	$g = b$
	h	$h = a \times b$
	p	Point of intersection $p = p_I/[c \cdot (a \times b)]$ where $p_I = c_4(a \times b) + b_4(b \times c) + c_4(c \times a)$
A point and a line $X = (x, \sigma_x)$ $\hat{u} = (u, u^0)$	g	Normal to the plane spanned by the point and the line $g = u^0 - x \times u$
	h	$h = u$
	p	The feet of perpendicular from X to $\hat{u}$ $p = u \times u^0 + (u \cdot x)u$
Two points and a plane $X = (x, \sigma_x)$ $Y = (y, \sigma_y)$ $A = (a, a_4)$	g	$g = a \times (y - x)$
	h	$h = a$
	p	$p = (x \div y)/2$
A point and two planes $X = (x, \sigma_x)$ $A = (a, a_4)$ $B = (b, b_4)$	g	$g = a \times b$
	h	$h = a$
	p	$p = x$
A line and a plane $\hat{u} = (u, u^0)$ $A = (a, a_4)$	g	$g = a \times u$
	h	$h = a$
	p	Point of intersection $p = (a_4 u + a \times u)/(a \cdot u)$

Table 5 Minimum number of nonoriented features for generating a natural frame

Combinations	*a*	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*	*j*	*k*
Points	4	0	3	1	2	0	2	0	1	0	1
Lines	0	0	0	0	0	3	1	1	2	2	1
Planes	0	4	1	3	2	0	0	2	0	1	1

the intersection point of three planes; (*3*) the feet of perpendicular from a point to a line; (*4*) and the feet of common perpendicular to two lines. The formulas for these geometric operations are found in Table 4.

Conclusions

In this paper we have presented a least squares solution to the problem of computing spatial displacements from a redundant set of simple features of points, lines, planes, and their combinations. With the aid of a Clifford algebra, the best orientation is obtained by an iterative procedure which involves finding the eigenvector associated with the least eigenvalue of a 4 × 4 real symmetric matrix. The best translation is a simple rational cubic function of the best orientation. It is also discussed that the closed-form solution exists for the best orientation when all features are represented by points and unit vectors. In addition, simple algorithms are provided for automatic generation of body-fixed coordinate frames from various feature information. It is hoped that the results in this paper will facilitate the use of measuring systems for determining spatial displacements that can handle not only point features but also line and plane features.

Acknowledgments

This work is supported by NSF grant number DMC-8796348, ARO grant number DAAL03-90-G-0005 and California Department of Transportation through the AHMCT program. Also acknowledged is the assistance of Mr. Shashikanth Reddy in implementing the algorithm presented in this paper.

References

1 Angeles, J., 1986, "Automatic Computation of the Screw Parameters of Rigid-Body Motions, Part I: Finitely-Separated Positions," ASME *Journal of Dynamic Systems, Measurement, and Control*, Vol. 108, No. 1, pp. 32–38.

2 Beggs, J. S., 1966, *Advanced Mechanism*, Macmillan, New York.

3 Bottema, O., and Roth, B., 1979, *Theoretical Kinematics*, North Holland Publ., Amsterdam.

4 Chen, N.-Y., Berk, J. R., and Kelley, R. B., 1980, "Estimating Workspace Pose Using the Feature Points Methods," *IEEE Trans. on Robotics and Automation*, Vol. 25, No. 6, pp. 1027–1041.

5 Clifford, W. K., 1876, "On the Classification of Geometric Algebras," *Mathematical Papers*, Tucker, R., ed., London: Macmillan, 1882, 658 pp.

6 Fenton, R. G., and Shi, X., 1990, "Comparison of Methods for Determining Screw Parameters of Finite Rigid Body Motions from Initial and Final Position Data," ASME JOURNAL OF MECHANICAL DESIGN, Vol. 112, No. 4, pp. 472–479.

7 Flanders, H., 1963, *Differential Forms with Application to the Physical Sciences*, Academic Press, New York, 205 pp.

8 Grimson, W. E., and Lozano-Perez, T., 1984, "Model-Based Recognition and Localization from Sparse Range or Tactile Data," *Int'l J. of Robotics Res.*, Vol. 3, No. 3, pp. 3–35.

9 Gunnarsson, K. T., and Prinz, F. B., 1987, "CAD Model-Based Localization of Parts in Manufacturing," *Computer*, August, pp. 66–74.

10 Hestenes, D., 1986, *New Foundations for Classical Mechanics*, D. Reidel Pub. Co., Dordrecht, Holland, 644 pp.

11 Horn, B. K. P., 1987, "Closed-Form Solution of Absolute Orientation Using Unit Quaternions," *Journal of the Optical Society of America*, Vol. 4, No. 4, pp. 629–642.

12 Laub, A. J., and Shiflett, G. R., 1982, "A Linear Algebra Approach to the Analysis of Rigid Body Velocity from Initial Position and Final Position Data," ASME *Journal of Applied Mechanics*, Vol. 49, pp. 213–216.

13 Luo, R. C., and Yang, W. S., 1990, "Motion Estimation of 3-D Objects Using Multisensor Data Fusion," *J. of Robotic Systems*, Vol. 7, No. 3, pp. 419–443.

14 McCarthy, J. M., 1990, *Introduction to Theoretical Kinematics*, MIT Press, Cambridge, Massachusetts, 130 pp.

15 Ravani, B., 1988, "World Modeling for CAD Based Robot Programming and Simulation," *CAD Based Programming for Sensory Robots*, pp. 67–89, NATO ASI Series, Vol. F50, B. Ravani, ed.,

16 Ravani, B., and Ge, Q. J., 1993, "Computation of Spatial Displacements from Geometric Features," ASME JOURNAL OF MECHANICAL DESIGN, Vol. 115, No. 1, pp. 95–102.

17 Spoor, C. W., and Veldpaus, F. E., 1980, "Rigid Body Motion Calculated from Spatial Co-ordinates of Markers," *J. Biomechanics*, Vol. 13, pp. 391–393.

18 Stolfi, J., 1989, *Primitives for Computational Geometry*, Research Report 36, Systems Research Center, Digital Equipment Corp.

APPENDICES

A: Absolute Minimum of the Error Function

Let $\mathbf{q}_s$ be the best orientation and $\mathbf{q}_s^0$ be the best translation in Clifford-algebra form. The minimum total error is obtained by substituting (27) into (22):

$$E(\mathbf{q}_s, \mathbf{q}_s^0) = \mathbf{q}_s^T[A]\mathbf{q}_s - \frac{1}{4}\mathbf{q}_s^T[B][C]^{-1}[B]^T\mathbf{q}_s + \frac{1}{4}\mathbf{q}_s^T[C]^{-1}\mathbf{q}_s(\lambda^0)^2. \quad (46)$$

This can be rewritten as

$$E(\mathbf{q}_s, \mathbf{q}_s^0) = \mathbf{q}_s^T [D(\lambda^0)] \mathbf{q}_s + R(\mathbf{q}_s, \lambda^0). \qquad (47)$$

where $[D(\lambda^0)]$ is given by (34) and

$$R(\mathbf{q}_s, \lambda^0) = -\frac{1}{4}(\mathbf{q}_s)^T [B'] \mathbf{q}_s \lambda^0 + (1/2)\mathbf{q}_s^T [C]^{-1} \mathbf{q}_s (\lambda^0)^2.$$

The term $R(\mathbf{q}_s, \lambda^0)$ vanishes after the substitution of (30), (31) and in view of the fact

$$(\mathbf{q}_s)^T [B][C]^{-1} \mathbf{q}_s = (\mathbf{q}_s)^T [C]^{-1} [B]^T \mathbf{q}_s.$$

Let λ_s be one of the eigenvalues of the matrix $[D(\lambda^0)]$. Then we have $\mathbf{q}_s^T [D(\lambda^0) \mathbf{q} = \lambda_s$ and consequently

$$E(\mathbf{q}_s, \mathbf{q}_s^0) = \lambda_s, \qquad (48)$$

This concludes that all eigenvalues of the matrix $[D(\lambda^0)]$ are non-negative and that the total error $E(\mathbf{q}_s, \mathbf{q}_s^0)$ is at absolute minimum when λ_s is the least eigenvalue of $[D(\lambda^0)]$. In the ideal case when all data are perfect, the total error $E(\mathbf{q}_s, \mathbf{q}_s^0) = 0$ and consequently, $\lambda_s = 0$.

B: Finding the Eigenvector

The four eigenvalues can be solved in closed form from the quartic characteristic equation. Once the least eigenvalue λ_s has been determined, the corresponding eigenvector is obtained by solving the homogeneous equation

$$[D'] \mathbf{q}_s = 0 \qquad (49)$$

where $[D'] = [D(\lambda^0)] - \lambda_s [I]$ is a singular matrix. Let **a**, **b**, **c**, and **d** denote the four row vectors of the matrix $[D']$. Then Eq. (49) implies

$$\mathbf{a} \cdot \mathbf{q}_s = 0, \quad \mathbf{b} \cdot \mathbf{q}_s = 0, \quad \mathbf{c} \cdot \mathbf{q}_s = 0, \quad \mathbf{d} \cdot \mathbf{q}_s = 0. \qquad (50)$$

When the least eigenvalue λ_s is distinct, only three of the four equations above are independent and the unit eigenvector $\mathbf{q}_s$ is unique. Although any three of these equations can be used to solve for $\mathbf{q}_s$, it is numerically more stable for a solution to use all of the four equations. We first obtain the following trivector

$$\mathbf{Q} = \mathbf{a} \wedge \mathbf{b} \wedge \mathbf{c} + \mathbf{a} \wedge \mathbf{c} \wedge \mathbf{d} + \mathbf{b} \wedge \mathbf{c} \wedge \mathbf{d} + \mathbf{a} \wedge \mathbf{b} \wedge \mathbf{d}$$

where "$\wedge$" denotes the vector wedge product. Wedge product generalizes to higher dimensions the vector cross product, see Flanders (1963). The coordinates of this trivector, $\mathbf{Q} = (Q_1, Q_2, Q_3, Q_4)$, can then be normalized to yield the unit eigenvector $\mathbf{q}_s$.

When λ_s is a double eigenvalue, only two of the four equations (50) are independent and the unit eigenvector $\mathbf{q}_s$ is not unique. In fact $\mathbf{q}_s$ traces out a unit circle. The plane containing the unit circle is given by

$$\mathbf{P} = \mathbf{a} \wedge \mathbf{b} + \mathbf{a} \wedge \mathbf{c} + \mathbf{a} \wedge \mathbf{d} + \mathbf{b} \wedge \mathbf{c} + \mathbf{b} \wedge \mathbf{d} + \mathbf{c} \wedge \mathbf{d}.$$

Any vector in this plane can be normalized to yield a desired eigenvector $\mathbf{q}_s$.

When λ_s is a triple eigenvalue, any unit vector perpendicular to $\mathbf{a} + \mathbf{b} + \mathbf{c} + \mathbf{d}$ is a desired eigenvector.

On First-Order Decoupling of Equations of Motion for Constrained Dynamical Systems

T. A. Loduha
Research Assistant.

B. Ravani
Professor.
Mem. ASME.

Department of Mechanical, Aeronautical,
and Materials Engineering,
University of California,
Davis, CA 95616-5294

In this paper we present a method for obtaining first-order decoupled equations of motion for multirigid body systems. The inherent flexibility in choosing generalized velocity components as a function of generalized coordinates is used to influence the structure of the resulting dynamical equations. Initially, we describe how a congruency transformation can be formed that represents the transformation between generalized velocity components and generalized coordinate derivatives. It is shown that the proper choice for the congruency transformation will insure generation of first-order decoupled equations of motion for holonomic systems. In the case of nonholonomic systems, or holonomic systems with unreduced configuration coordinates, we incorporate an orthogonal complement in conjunction with the congruency transformation. A pair of examples illustrate the results. Finally, we discuss numerical implementation of congruency transformations to achieve first-order decoupled equations for simulation purposes.

Introduction

Constrained multirigid body systems refers to systems of interconnected bodies and particles which are subjected to various motion constraints. Such systems are abundantly relevant in engineering for modeling a wide variety of mechanical systems. Much attention has been focused on formulation procedures to yield the differential equations describing the motion of multibody systems (Crandall et al., 1968; Gibbs, 1879; Gibbs, 1961; Hartog, 1948; Huston, 1990; Kane and Levinson, 1985; Roberson and Schwertassek, 1988; Scott, 1988; Storch and Gates, 1989). In most cases the resulting equations are numerically integrated to obtain trajectories characterizing the system's motion. In addition, the equations of motion are often analyzed directly to determine the nature of the nonlinear behavior. This paper demonstrates a method, using Kane's[1] equations (Kane and Levinson, 1985), for generating equations of motion which are decoupled in the highest derivative terms. We will refer to such equations as being *first-order decoupled.* Nonlinear differential equations of this form are more easily integrated numerically, as well as better fit for analysis. The procedure of generalized velocity component selection based on a congruency transformation (Wade, 1951) is developed and used to achieve first-order decoupled form of the equations of motion.

Considerable attention has been placed on contending with holonomic and nonholonomic, linear and nonlinear motion constraints on multirigid body systems (Kamman and Huston, 1984; Kane, 1972; Nikravesh and Haug, 1983; Wang and Huston, 1988; Wehage and Haug, 1982; Wampler et al., 1985; Xu et al., 1990). A common way to deal with constraints is to impose them at an early stage of the analysis by reducing the set of dependent generalized coordinates to an independent one. If the constraints are nonholonomic the generalized coordinate derivatives are reduced accordingly. However, many consider it to be more effective to first perform the dynamical analysis for the unconstrained system, and then reduce the resulting equations to a consistent set with the constraint equations. For example, Kamman and Huston (1984), using Kane's formulation, show that the projection of existing equations of motion onto an *orthogonal complement* yields the desired reduced equations. An orthogonal complement, say C, of matrix **B** would satisfy the equation $\mathbf{BC} = \mathbf{0}$. Ben-Israel and Greville (1974) and Lawson and Hanson (1974) discuss the mathematical significance of the orthogonal complement. The use of the orthogonal complement to impose motion constraints is illustrated by Hemami and Weimer (1981), Huston (1990), Kamman and Huston (1984), Wang and Huston (1988), and Xu et al. (1990). Hemami and Weimer (1981) use the orthogonal complement similarly for contracting equations generated by the Lagrange formulation. Wampler et al. (1985) discuss a method for

[1] It has been pointed out (see Desloge, 1987 and Huston, 1987) that these equations are actually a particular form of the Gibbs-Appell equations (see Gibbs, 1879 and 1961).

Contributed by the Applied Mechanics Division of The American Society of Mechanical Engineers for publication in the ASME Journal of Applied Mechanics.

Discussion on this paper should be addressed to the Technical Editor, Professor Lewis T. Wheeler, Department of Mechanical Engineering, University of Houston, Houston, TX 77204-4792, and will be accepted until four months after final publication of the paper itself in the ASME Journal of Applied Mechanics.

Manuscript received by the ASME Applied Mechanics Division, Feb. 22, 1993; final revision, May 27, 1993. Associate Technical Editor: R. L. Huston.

reduction of existing equations of motion subject to additional constraints by recombining terms from the original equations. A numerical procedure for imposing constraints during integration based on generalized coordinate partitioning is presented by Nikravesh and Haug (1983) and Wehage and Haug (1982). One common element of each of these methods is the application of the constraints after conducting the dynamical analysis. Conversely, we illustrate a method that imposes constraints at an intermediate step of the kinematical analysis to enable decoupling for nonholonomic systems, or systems where the holonomic constraints are yet to be applied.

The following discussion addresses the use of generalized velocity component selection to achieve first-order decoupling of multirigid body systems. The idea of the congruency transformation is explained and utilized here. A simple example is conducted to clarify the procedure for finding the congruency transformation for holonomic systems. Next, a procedure is discussed for decoupling of systems with nonholonomic constraints, or holonomic systems with unreduced configuration coordinate descriptions. The result, using orthogonal complements, is a modified nonsquare transformation between generalized coordinate time derivatives and a reduced set of generalized velocity components. Decoupling using the orthogonal complement is demonstrated by relaxing a constraint from the first example. Lastly, we discuss the numerical application of congruency transformations.

Dynamics of Holonomic Multirigid Body-Systems

First-Order Decoupled Equations of Motion. In the process of formulating equations of motion, for example, using Kane's method, the analyst must choose a linear combination of first time derivatives of generalized coordinates to define generalized velocity components. These, in a general form, were initially introduced by Gibbs (1879, 1961), but were exploited in more detail by Kane (see, for example Kane (1972) and Kane and Levinson (1985)). Kane and his coworkers have referred to these quantities as "generalized speeds." Some of the problems for such a term have been pointed out by Papastavridis (1992), who considers either "nonholonomic components of the velocity vector" or "quasi-velocities" to be more appropriate terms. Singh and Likins (1985) mention the term "derivatives of quasicoordinates" as an alternative. Here, for lack of a better name, we shall simply call them *generalized velocity components*. The resulting equations of motion, using the generalized velocity components, in matrix notation, are of the form

$$\mathbf{M}\dot{\mathbf{u}} = \mathbf{g}(\mathbf{q}, \mathbf{u}) \tag{1}$$

where **u** and **q** are vectors of generalized velocity components and generalized coordinates, respectively. **M** is a matrix whose elements are functions of generalized coordinates and the inertia properties of the system, and **g** is a nonlinear vector function of generalized velocity components and generalized coordinates. The technique presented here will generate equations of this form where **M** can be made diagonal by judicious selection of generalized velocity components. Nonlinear differential equations of this form are dramatically easier to numerically integrate, as the need for computing $\mathbf{M}^{-1}$ at each iteration is eliminated. In addition, performing various analyses of nonlinear behavior—tests for stability, nature of critical points, chaos, etc.—is facilitated if the system equations are written in state plane form. A diagonal **M** matrix in Eq. (1) satisfies this requirement.

Kane's Equations. Consider a system of p rigid bodies whose configuration can be described completely by the set of n generalized coordinates $(q_1, g_2, \ldots, q_r, \ldots, q_n)$, or $\mathbf{q} = [q_1, q_2, \ldots, q_r, \ldots, q_n]^T$. Kane's equations (Kane and Levinson, 1985) are formed by letting the sum of generalized active forces and generalized inertia forces equal zero. These equations are actually a particular form of the Gibbs-Appell equations (see Desloge, 1987; Huston, 1987). Kane (1972) originally referred to them as Lagrange's form of D'Alembert's principle. In these equations the generalized active force vector can be expressed as

$$\mathbf{F} = \sum_{i=1}^{p} \left[\mathbf{V}_i^T \mathbf{f}_i + \Gamma_i^T \tau_i\right] \tag{2}$$

where **f** is the resultant active force acting at the mass center of the ith body, and τ_i is the resultant moment. **V** and Γ_i are the partial velocity and partial angular velocity matrices, which are written as

$$\underset{(3\times n)}{\mathbf{V}_i} = \frac{\partial \mathbf{v}_i}{\partial \mathbf{u}^T} \tag{3}$$

and

$$\underset{(3\times n)}{\Gamma_i} = \frac{\partial \omega_i}{\partial \mathbf{u}^T}. \tag{4}$$

These partial velocities are taken with respect to the generalized velocity component vector **u**. The generalized velocity components are a linear combination of the first time derivatives of generalized coordinates. For ease of substitution this relation is expressed as

$$\dot{\mathbf{q}} = \mathbf{T}\mathbf{u}. \tag{5}$$

We shall refer to the matrix **T** as the *rate transformation* matrix. Since we are considering holonomic systems here, **T** is a square matrix of order n. The generalized inertia force vector is written as

$$\mathbf{F}^* = -\sum_{i=1}^{p} \left[m_i \mathbf{V}_i^T \mathbf{a}_i + \Gamma_i^T \dot{\mathbf{H}}_i\right] \tag{6}$$

where $\mathbf{a}_i$ is the mass center acceleration of the ith body and $\dot{\mathbf{H}}_i$ is the time rate of change of angular momentum of body i with respect to the Newtonian reference frame. Finally, n dynamical equations of motion are obtained by letting the vector sums from Eqs. (2) and (6) equal the zero vector:

$$\mathbf{F} + \mathbf{F}^* = \mathbf{0} \tag{7}$$

Equation (7) is the matrix form of the so-called Kane's equations. It should be pointed out that in this paper, as with other references on constrained multibody systems (see, for example, Huston, 1990), we shall use the term *configuration coordinates* in addition to *generalized coordinates*. In constrained multibody systems configuration coordinates refer to the variables that uniquely describe a system's configuration, but may be dependent upon one another. Generalized coordinates represents reduced, or independent configuration coordinates.

The Influence of Generalized Velocity Component Selection. Since our goal in this paper is to prescribe a choice of generalized velocity components that would yield decoupled equations of motion, it is first necessary to reveal the influence of such a selection on the resulting dynamical equations. In other words, we would like to expose the location of the matrix **T**, from Eq. (5), within Kane's equations. The acceleration of the mass center of the ith body can be written as

$$\mathbf{a}_i = \mathbf{V}_i\dot{\mathbf{u}} + \dot{\mathbf{V}}_i\mathbf{u}. \tag{8}$$

We also know the angular acceleration of body i can be expressed as

$$\dot{\omega}_i = \Gamma_i\dot{\mathbf{u}} + \dot{\Gamma} i\mathbf{u}. \tag{9}$$

From this we can write the derivative of angular momentum as

$$\dot{\mathbf{H}}_i = \mathbf{I}_{\Gamma_i}\dot{\mathbf{u}} + \mathbf{I}_i\dot{\Gamma}_i\mathbf{u} + \mathbf{W}_i\mathbf{I}_i\omega_i \tag{10}$$

where $\mathbf{I}_i$ is the central inertia matrix and $\mathbf{W}_i$ is the angular velocity matrix associated with the ith body, and written in terms of body i's natural frame. Thus, the generalized inertia force becomes

$$\mathbf{F}^* = -\sum_{i=1}^{p}\Big[m_i\mathbf{V}_i^T\mathbf{V}_i\dot{\mathbf{u}} + m_i\mathbf{V}_i^T\dot{\mathbf{V}}_i\mathbf{u} + \Gamma_i^T\mathbf{I}_i\Gamma_i\dot{\mathbf{u}} + \Gamma_i^T\mathbf{I}_i\dot{\Gamma}_i\mathbf{u} + \Gamma_i^T\mathbf{W}_i\mathbf{I}_i\omega_i\Big]. \tag{11}$$

From the relation depicted in Eq. (5), it is easily shown that the partial velocity matrix of Eq. (3) can be expressed as

$$\frac{\partial \mathbf{v}_i}{\tau \mathbf{u}^T} = \frac{\partial \mathbf{v}_i}{\partial \dot{\mathbf{q}}^T}\mathbf{T} = \frac{\partial \mathbf{r}_i}{\partial \mathbf{q}^T}\mathbf{T} \tag{12}$$

where is $\mathbf{r}_i$ the Euclidean position of body i's mass center with respect to the inertial reference frame. We can write this as

$$\mathbf{V}_i = \mathbf{J}_i\mathbf{T} \tag{13}$$

where $\mathbf{J}_i$ is the partial derivative of $\mathbf{r}_i$ body's mass center position with respect to the vector of the generalized coordinates. Similarly, Eq. (4) is

$$\frac{\partial \omega_i}{\partial \mathbf{u}^T} = \frac{\partial \omega_i}{\partial \dot{\mathbf{a}}^T}\mathbf{T} \tag{14}$$

or

$$\Gamma_i = \Omega_i\mathbf{T} \tag{15}$$

where Ω_i is the partial derivative of body i's angular velocity with respect to the time derivative of the generalized coordinate vector. Substitution of Eqs. (13) and (15) into Eq. (6) gives the following expression for the equations of motion:

$$-\sum_{i=1}^{p}\Big[m_i\mathbf{T}^T\mathbf{J}_i^T\mathbf{J}_i\mathbf{T}\dot{\mathbf{u}} + m_i\mathbf{T}^T\mathbf{J}_i^T\frac{d}{dt}(\mathbf{J}_i\mathbf{T})\mathbf{u} + \mathbf{T}^T\Omega_i^T\mathbf{I}_i\Omega_i\mathbf{T}\dot{\mathbf{u}} + \mathbf{T}^T\Omega_i^T\mathbf{I}_i\frac{d}{dt}(\Omega_i\mathbf{T})\mathbf{u} + \mathbf{T}^T\Omega_i^T\mathbf{W}_i\mathbf{I}_i\omega_i - \mathbf{T}^T\mathbf{J}_i^T\mathbf{f}_i - \mathbf{T}^T\Omega_i^T\mathbf{f}_i - \mathbf{T}^T\Omega_i^T\tau_i\Big] = \mathbf{0}. \tag{16}$$

By letting

$$\mathbf{A} = -\sum_{i=1}^{p}\left[m_i\mathbf{J}_i^T\mathbf{J}_i + \Omega_i^T\mathbf{I}_i\Omega_i\right], \tag{17}$$

a complete set of equations of motion can be expressed as

$$\mathbf{T}^T\mathbf{A}\mathbf{T}\dot{\mathbf{u}} - \sum_{i=1}^{p}\Big[m_i\mathbf{T}^T\mathbf{J}_i^T\frac{d}{dt}(\mathbf{J}_i\mathbf{T})\mathbf{u} + \mathbf{T}^T\Omega_i^T\mathbf{I}_i\frac{d}{dt}(\Omega_i\mathbf{T})\mathbf{u} + \mathbf{T}^T\Omega_i^T\mathbf{W}_i\mathbf{I}_i\omega_i - \mathbf{T}^T\mathbf{J}_i^T\mathbf{f}_i - \mathbf{T}^T\Omega_i^T\tau_i\Big] = \mathbf{0} \tag{18a}$$

and

$$\dot{\mathbf{q}} = \mathbf{T}\mathbf{u}. \tag{18b}$$

It is now clear that T manifests itself in the transformation $\mathbf{T}^T\mathbf{A}\mathbf{T}$ in the first term of the left-hand side of Eq. (18*a*), as far as first-order generalized velocity components are concerned. We now wish to explain how the analyst can choose the rate transformation matrix **T** in Eq. (18*b*) to assure that the coefficient matrix for the first-order generalized speed vector in Eq. (18*a*) is diagonal.

Selecting T for Decoupling. Clearly, from the above results, first-order decoupled equations of motion will be generated if $\mathbf{T}^T\mathbf{A}\mathbf{T}$ is a diagonal matrix. Therefore, we first consider how this transformation can be influenced by **T** to satisfy this condition. Notice that if the eigenvectors of **A** are used as the columns of **T**, a diagonal matrix results under the similarity transformation $\mathbf{T}^{-1}\mathbf{A}\mathbf{T}$. Moreover, **A** is symmetric, hence, its eigenvectors can be chosen to be orthogonal. With orthogonal eigenvectors we know that $\mathbf{T}^{-1} = \mathbf{T}^T$, and the similarity transformation effectively appears as the leading matrix for the first-order terms of Eq. (18*a*). Thus, a sufficient condition for obtaining first-order decoupled equations of motion is that the rate transformation matrix **T** be comprised of the eigenvectors of **A**. However, it should be emphasized that it is *not* a necessary condition for decoupling. In fact it can be much less laborious to choose the elements of **T** to satisfy a congruency transformation than to symbolically determine the eigenvectors of **A**. We will now show that decoupling can be achieved by satisfying an alternative sufficiency condition. That is, if **T** is chosen to fulfill a specific congruency transformation, Eq. (18*a*) will be decoupled in first-order terms. Consider the following definition (see, for example, Wade, 1951). If for two given matrices **A** and **N** there exists a nonsingular matrix **T** that satisfies the relation

$$\mathbf{T}^T\mathbf{A}\mathbf{T} = N \tag{19}$$

then **A** and **N** are said to be congruent. Furthermore, we shall utilize a theorem (see Wade, 1951) stating that a symmetric matrix can be reduced by a congruency transformation to a diagonal matrix of the same rank. The principal motivation behind employing congruency transformations to achieve decoupling is that the rate transformation matrix can be formed directly with various combinations of the elements of the **A** matrix. At this point we propose an algorithm for obtaining the matrix **T** that yields a diagonal matrix under the congruency transformation of Eq. (19). This, in turn, would be used in Eq. (18*b*) as a rate transformation matrix. Consider the symmetric **A** matrix written as

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{12} & a_{22} & \cdots & \cdots \\ \cdot & \cdot & \cdot & \cdot \\ a_{1n} & \cdots & \cdots & a_{nn} \end{bmatrix}. \tag{20}$$

The transformation matrix **T** will be composed of m factors, such that it can be written

$$\mathbf{T} = \mathbf{T}_1\mathbf{T}_2\mathbf{T}_3\ldots\mathbf{T}_m \tag{21}$$

where m is the number of degrees-of-freedom less one ($m = n - 1$). $\mathbf{T}_1$ would be constructed as follows:

$$\mathbf{T}_1 = \begin{bmatrix} 1 & -a_{12}/a_{11} & \cdots & -a_{1k}/a_{11} & \cdots & -a_{1n}/a_{11} \\ 0 & 1 & \cdots & \cdots & \cdots & 0 \\ \cdot & \cdot & 1 & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdots & \cdots & 0 & 1 \end{bmatrix}. \tag{22}$$

When the congruency transformation

$$\mathbf{T}_1^T\mathbf{A}\mathbf{T}_1 = A' \tag{23}$$

is performed, the resulting A' is a matrix with zero elements in row one and column one, except for the element a_{11} (see Eq. (24)). In other words, it will be block diagonal with the

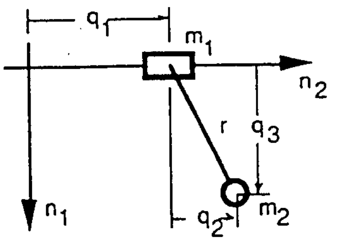


Fig. 1 A sliding pendulum

lower right m by m submatrix remaining undiagonalized, and is given here as

$$A' = \mathbf{T}_1^T\mathbf{A}\mathbf{T}_1 = \begin{bmatrix} a_{11} & 0 & 0 & \dots & 0 \\ 0 & b_{11} & b_{12} & .. & b_{1m} \\ 0 & b_{21} & b_{22} & .. & b_{2m} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & b_{m1} & b_{m2} & .. & b_{mm} \end{bmatrix}. \quad (24)$$

The same procedure is performed on this result, except $\mathbf{T}_2$ is constructed as

$$\mathbf{T}_2 = \begin{bmatrix} 1 & 0 & .. & .. & 0 \\ 0 & 1 & -b_{12}/b_{11} & \dots & -b_{1m}/b_{11} \\ 0 & 0 & 1 & \dots & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix}. \quad (25)$$

The congruency transformation is performed again:

$$\mathbf{T}_2^T\mathbf{T}_1^T\mathbf{A}\mathbf{T}_1\mathbf{T}_2 = A'' \quad (26)$$

This is repeated until a total of m transformations have been completed, and the A matrix is converted to diagonal form. One additional requirement is that T be nonsingular. However, this is assumed since the final diagonal matrix is of the same rank as the original A matrix. Therefore, T must be of full rank. Before discussing the use of the orthogonal complement, an example is presented to illustrate decoupling with the congruency transformation.

Example of Decoupling With the Congruency Transformation. Consider the following system of two particles connected by a rigid, massless rod. Sliding mass m_1 is constrained to move along the horizontal axis $\mathbf{n}_2$, and the mass m_2 must stay on the constant radius arc with respect to m_1 in the $\mathbf{n}_1 - \mathbf{n}_2$ plane, as shown in Fig. 1.

Using the generalized coordinates q_1 and q_2, the matrix A is assembled using $\mathbf{J}_1$ and $\mathbf{J}_2$ for each of the masses. This gives

$$\mathbf{J}_1 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \quad (27)$$

and

$$\mathbf{J}_2 = \begin{bmatrix} 0 & -rsq_2 \\ 1 & rcq_2 \end{bmatrix} \quad (28)$$

which can be combined according to Eq. (17) to obtain

$$\mathbf{A} = \begin{bmatrix} -m_1 - m_2 & -m_2rcq_2 \\ -m_2rcq_2 & -m_2r^2 \end{bmatrix}. \quad (29)$$

Equation (19) is used to form the congruency transformation for A as

$$\mathbf{T} = \begin{bmatrix} 1 & -m_2rcq_2/(m_1 + m_2) \\ 0 & 1 \end{bmatrix}. \quad (30)$$

For this system the resultant applied force vectors for each mass are

$$\mathbf{f}_1 = \begin{bmatrix} m_1g \\ 0 \end{bmatrix} \quad (31)$$

and

$$\mathbf{f}_2 = \begin{bmatrix} m_2g \\ 0 \end{bmatrix}. \quad (32)$$

The above matrices can then be assembled using Eq. (18a) to yield the following dynamical equations:

$$\begin{bmatrix} -m_1 - m_2 & 0 \\ 0 & m_2r^2 - m_2^2r^2c^2q_2/(m_1 - m_2) \end{bmatrix}\dot{\mathbf{u}}$$
$$= \begin{bmatrix} 0 \\ -m_2^2r^2cq_2sq_2u_2^2/(m_1 + m_2) \end{bmatrix} + \begin{bmatrix} 0 \\ -m_2grsq_2 \end{bmatrix}. \quad (33)$$

The complete set of equations of motion are Eq. (33) and the following equation using the rate transformation matrix given in Eq. (30):

$$\dot{\mathbf{q}} = \begin{bmatrix} 1 & -m_2rcq_2/(m_1 + m_2) \\ 0 & 1 \end{bmatrix}\mathbf{u}. \quad (34)$$

Equation (34) shows that the generalized velocity components obtained using the congruency transformation include mass terms. This is slightly different from the definition of the relationship between $\mathbf{u}$ and $\dot{\mathbf{q}}$, as defined by Kane (1985) to be a function of $\mathbf{q}$ and time, and not of the mass properties of the system. It also indicates another problem with the term "generalized speeds," or even "generalized velocity components," for the elements of the vector $\mathbf{u}$, since they depend on mass properties in the more general context used in this paper. We shall, however, continue using "generalized velocity components" since the term "generalized" may be interpreted that $\mathbf{u}$ is not just like a common velocity vector.

Dynamics of Nonholonomic Multibody Systems

The equations of motion given in Eqs. (18a) and (18b) become invalid if the multibody system includes constraints which are nonholonomic. If this is the case, the constraints can be applied to the system with an orthogonal complement, and the transformation between generalized coordinate derivatives and generalized velocity components is no longer one-to-one. Additionally, situations might arise where a model is given with unreduced configuration coordinates along with holonomic constraint equations that are yet to be applied. Again, the orthogonal complement is used to impose such constraints. These ideas are discussed next, and illustrated with an example.

The Orthogonal Complement. Huston (1990) shows that if the motion constraint equations, either holonomic or nonholonomic, are written as

$$\mathbf{B}\dot{\mathbf{q}} = \mathbf{0} \quad (35)$$

and the unconstrained equations of motion are as depicted in Eq. (1), a valid set of reduced equations describing the dynamics of the constrained system is

$$\mathbf{C}^T\mathbf{M}\ddot{\mathbf{q}} = \mathbf{C}^T\mathbf{g}(\dot{\mathbf{q}}, \mathbf{q}) \quad (36)$$

where $\mathbf{C}$ is the orthogonal complement of $\mathbf{B}$. Recall $\mathbf{C}$ is the orthogonal complement of $\mathbf{B}$ if $\mathbf{BC} = \mathbf{0}$ is satisfied. $\mathbf{C}$ can be thought of as a non-square transformation between the unreduced generalized coordinate derivative vector $\dot{\mathbf{q}}$, and a reduced generalized coordinate derivative vector, say $\dot{\mathbf{z}}$:

$$\dot{\mathbf{q}} = \mathbf{C}\dot{\mathbf{z}}. \quad (37)$$

Let us rewrite Eq. (36) in similar form to Eq. (16) by first writing

$$-\sum_{i=1}^{p}\Big[m_i\mathbf{C}^T\mathbf{J}_i^T\mathbf{J}_i\ddot{\mathbf{q}} + m_i\mathbf{C}^T\mathbf{J}_i^T\frac{d}{dt}(\mathbf{J}_i)\dot{\mathbf{q}}$$

$$+\mathbf{C}^T\Omega_i^T\mathbf{I}_i\Omega_i\ddot{\mathbf{q}} + C^T\Omega_i^T\mathbf{I}_i\frac{d}{dt}(\Omega_i)\dot{\mathbf{q}}$$

$$+\mathbf{C}^T\Omega_i^T\mathbf{W}_i\mathbf{I}_i\omega_i - \mathbf{C}^T\mathbf{J}_i^T\mathbf{f}_i - \mathbf{C}^T\Omega_i^T\tau_i\Big] = \mathbf{0}. \quad (38)$$

We can substitute the expression for $\ddot{\mathbf{q}}$ given in Eq. (37) into Eq. (38) to establish equations of motion in terms of the new reduced generalized coordinate vector $\mathbf{z}$ and its derivatives:

$$-\sum_{i=1}^{p}\Big[m_i\mathbf{C}^T\mathbf{J}_i^T\mathbf{J}_i\mathbf{C}\ddot{\mathbf{z}} + m_i\mathbf{C}^T\mathbf{J}_i^T\frac{d}{dt}(\mathbf{J}_i\mathbf{C})\dot{\mathbf{z}} + \mathbf{C}^T\Omega_i^T\mathbf{I}_i\Omega_i\mathbf{C}\ddot{\mathbf{z}}$$

$$+\mathbf{C}^T\Omega_i^T\mathbf{I}_i\frac{d}{dt}(\Omega_iC)\dot{\mathbf{z}} + \mathbf{C}^T\Omega_i^T\mathbf{W}_i\mathbf{I}_i\omega_i$$

$$-\mathbf{C}^T\mathbf{J}_i^T\mathbf{f}_i - \mathbf{C}^T\Omega_i^T\tau_i\Big] = \mathbf{0}. \quad (39)$$

One can see that the role of the matrix **C** in Eq. (39) is equivalent to that of **T** in Eq. (16). Hence, we can view

$$\mathbf{J}_i\mathbf{C} = \frac{\partial \mathbf{v}_i}{\partial \dot{\mathbf{z}}^T} \quad (40)$$

and

$$\Omega_i\mathbf{C} = \frac{\partial \omega_i}{\partial \dot{\mathbf{z}}^T} \quad (41)$$

as special partial velocity matrices. Generalized velocity components can be introduced easily using the transformation

$$\dot{\mathbf{z}} = \mathbf{Tu}. \quad (42)$$

With this, the transformation between unreduced generalized coordinate derivatives and reduced generalized velocity components is written as

$$\dot{\mathbf{q}} = \mathbf{CTu}. \quad (43)$$

A new set of equations of motion in terms of reduced generalized velocity components can now be given as

$$-\sum_{i=1}^{p}\Big[m_i\mathbf{T}^T\mathbf{C}^T\mathbf{J}_i^T\mathbf{J}_i\mathbf{CT}\dot{\mathbf{u}} + m_i\mathbf{T}^T\mathbf{C}^T\mathbf{J}_i^T\frac{d}{dt}(\mathbf{J}_i\mathbf{CT})\mathbf{u}$$

$$+\mathbf{T}^T\mathbf{C}^T\Omega_i^T\mathbf{I}_i\Omega_i\mathbf{CT}\dot{\mathbf{u}} + \mathbf{T}^T\mathbf{C}^T\Omega_i^T\mathbf{I}_i\frac{d}{dt}(\Omega_i\mathbf{CT})\mathbf{u}$$

$$+\mathbf{T}^T\mathbf{C}^T\Omega_i^T\mathbf{W}_i\mathbf{I}_i\omega_i - \mathbf{T}^T\mathbf{C}^T\mathbf{J}_i^T\mathbf{f}_i + \mathbf{T}^T\mathbf{C}^T\Omega_i^T\tau_i\Big] = 0 \quad (44)$$

which are accurate in describing the behavior of the *constrained* system. Note that the matrix product **CT** shown in Eq. (43) represents a modified, nonsquare, rate transformation matrix.

There are several procedures for determining the orthogonal complement matrix as demonstrated by Huston (1990) and Hemami and Weimer (1981). Here, we will consider the zero-eigenvalue approach discussed by Huston. If there are m motion constraints imposed on a system that has n degrees-of-freedom, the m by n matrix **B** from Eq. (35) will be of rank m. **B** premultiplied by its transpose will be a symmetric n by n matrix also of rank m. Hence, $\mathbf{B}^T\mathbf{B}$ will have $(n - m)$ independent eigenvectors associated with the zero eigenvalues. If the columns of **C** are comprised of these eigenvectors we have

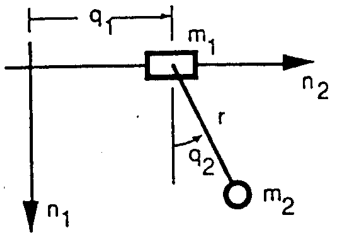


Fig. 2 A sliding pendulum with unreduced configuration coordinates

$$\mathbf{B}^T\mathbf{BC} = \mathbf{0}. \quad (45)$$

Premultiplying by $\mathbf{C}^T$ gives

$$\mathbf{C}^T\mathbf{B}^T\mathbf{BC} = \mathbf{0} \quad (46)$$

or

$$\mathbf{BC} = \mathbf{0}. \quad (47)$$

Therefore, **C** is the orthogonal complement of **B**. With **C** specified, we can now focus on the coefficient matrix for generalized velocity component derivatives from Eq. (44) by grouping all other terms into the function **h**. This gives

$$\mathbf{T}^T\mathbf{C}^T\mathbf{ACT}\dot{\mathbf{u}} = \mathbf{h}(\mathbf{q}, \mathbf{u}). \quad (48)$$

The matrix **A** is the same as defined in Eq. (17). To obtain first-order decoupled equations for the constrained system we simply choose **T** to be the proper congruency transformation, this time for $\mathbf{C}^T\mathbf{AC}$, using the procedure outlined earlier.

Example: Unreduced Configuration Coordinates. We will now repeat the example carried out above, but now using the dependent configuration coordinates shown in Fig. 2.

The constraint imposed by the rigid rod is temporarily removed. The position Jacobians for the two masses now appear as

$$\mathbf{J}_1 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad (49)$$

and

$$\mathbf{J}_2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}. \quad (50)$$

The resulting **A** matrix is

$$\mathbf{A} = \begin{bmatrix} -m_1 - m_2 & 0 & -m_2 \\ 0 & -m_2 & 0 \\ -m_2 & 0 & -m_2 \end{bmatrix}. \quad (51)$$

Now we must find the orthogonal complement to reimpose the constraint. Therefore, the constraint representation should be put in the form of Eq. (35). For the simple pendulum this constraint is written as

$$\mathbf{B}\dot{\mathbf{q}} = [0 \quad q_2 \quad q_3]\dot{\mathbf{q}} = 0. \quad (52)$$

To find an orthogonal complement the zero eigenvectors of $\mathbf{B}^T\mathbf{B}$ must be found. However, for this problem the vectors comprising **C** can be obtained even more simply, using the Gram-Schmidt process, for example. The orthogonal complement is found to be

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \\ 0 & -q_3 \\ 0 & q_2 \end{bmatrix}. \quad (53)$$

01168/S06

ACG

We can form $\mathbf{C}^T\mathbf{A}\mathbf{C}$ to allow determination of $\mathbf{T}$ as

$$\mathbf{T} = \begin{bmatrix} 1 & -m_2q_2/(m_1+m_2) \\ 0 & 1 \end{bmatrix}. \qquad (54)$$

The necessary components are now available to produce the constrained dynamical equations of motion using Eq. (44). The applied force vectors remain the same as in the previous example. The dynamical equations in terms of the reduced generalized velocity component vector are

$$\begin{bmatrix} -m_1 - m_2 & 0 \\ 0 & -m_2^2q_2^2/(m_1+m_2) + m_2r^2 \end{bmatrix}\dot{\mathbf{u}}$$

$$= \begin{bmatrix} 0 \\ -m_2^2q_3q_2u_2^2/(m_1+m_2) \end{bmatrix} + \begin{bmatrix} 0 \\ m_2gq_3 \end{bmatrix}. \qquad (55)$$

The additional equations of motion stemming from the transformation between configuration coordinates and reduced generalized velocity components are

$$\dot{\mathbf{q}} = \begin{bmatrix} 1 & -m_2q_2/(m_1+m_2) \\ 0 & 1 \end{bmatrix}\mathbf{u}. \qquad (56)$$

Numerical Implementation of Congruency Transformations

Very often one must analyze large-scale systems where it would be very tedious to symbolically obtain the proper congruency transformation. For this reason, we now discuss numerical implementation of the congruency transformation so that it may be used on more complicated systems. Recall the expression for the equations of motion shown in Eq. (16), except here we do not require the matrix $\mathbf{T}$ to be a diagonalizing congruency transformation. In this case $\mathbf{T}$ may be any matrix that yields a valid vector of generalized velocity components. The equations of motion for a holonomic system are

$$-\sum_{i=1}^{p}\Bigg[m_i\mathbf{T}^T\mathbf{J}_i^T\mathbf{J}_i\mathbf{T}\dot{\mathbf{u}} + m_i\mathbf{T}^T\mathbf{J}_i^T\frac{d}{dt}(\mathbf{J}_i\mathbf{T})\mathbf{u}$$

$$+\mathbf{T}^T\Omega_i^T\mathbf{I}_i\Omega_i\mathbf{T}\dot{\mathbf{u}} + \mathbf{T}^T\Omega_i^T\mathbf{I}_i\frac{d}{dt}(\Omega_i\mathbf{T})\mathbf{u}$$

$$+\mathbf{T}^T\Omega_i^T\mathbf{W}_i\mathbf{I}_i\omega_i - \mathbf{T}^T\mathbf{J}_i^T\mathbf{f}_i - \mathbf{T}^T\Omega_i^T\tau_i\Bigg] = \mathbf{0}. \qquad (57)$$

Notice that the derivative of the rate transformation matrix $\mathbf{T}$ appears in two of the terms in Eq. (57). Even though it would be possible to select $\mathbf{T}$ to be a diagonalizing congruency transformation at each iteration of the numerical integration, we would be left with the cumbersome task of specifying its time derivative as well. Hence, the following development allows selection of a matrix Ψ which is independent of the original dynamical equations (Eq. (57)), and does not appear in derivative form. However, one must still choose a valid linear combination of generalized coordinate derivatives (prescribed by $\mathbf{T}$) to define generalized velocity components, as usually done with conventional application of Kane's equations. For example, it is common to choose a trivial set of generalized velocity components by selecting $\mathbf{T}$ as the identity matrix. Denote the matrix products premultiplying $\mathbf{u}$ from Eq. (57) as

$$\mathbf{A} = \mathbf{T}^T\left(-\sum_{i=1}^{p}\left[m_i\mathbf{J}_i^T\mathbf{J}_i + \Omega_i^T\mathbf{I}_i\Omega_i\right]\right)^T \qquad (58)$$

and the remaining vector sum as $\mathbf{h}(\mathbf{q},\mathbf{u})$ so the equations of motion can be written as

$$\mathbf{A}\dot{\mathbf{u}} = \mathbf{h}(\mathbf{q},\mathbf{u}). \qquad (59)$$

Let us introduce the vector $\mathbf{x}$ in the following expression:

$$\dot{\mathbf{u}} = \Psi\mathbf{x}. \qquad (60)$$

Substituting this into Eq. (59) yields

$$\mathbf{A}\Psi\mathbf{x} = \mathbf{h}(\mathbf{q},\mathbf{u}). \qquad (61)$$

We can now premultiply both sides of Eq. (61) by Ψ^T giving us

$$\Psi^T\mathbf{A}\Psi\mathbf{x} = \Psi^T\mathbf{h}(\mathbf{q},\mathbf{u}). \qquad (62)$$

However, notice the left-hand side of Eq. (62) is a congruency transformation on the symmetric matrix $\mathbf{A}$. If Ψ is chosen as prescribed earlier, $\Psi^T\mathbf{A}\Psi$ will be diagonal. Let $\mathbf{L} = \Psi^T\mathbf{A}\Psi$ so Eq. (62) becomes

$$\mathbf{L}\mathbf{x} = \Psi^T\mathbf{h}(\mathbf{q},\mathbf{u}). \qquad (63)$$

Vector $\mathbf{x}$ can be written as

$$\mathbf{x} = \mathbf{L}^{-1}\Psi^T\mathbf{h}(\mathbf{q},\mathbf{u}) \qquad (64)$$

when the inverse of $\mathbf{L}$ is simply comprised of the reciprocals of its diagonal elements. Finally, the resulting first-order decoupled form in terms of $\dot{\mathbf{u}}$ is

$$\dot{\mathbf{u}} = \Psi\mathbf{L}^{-1}\Psi^T\mathbf{h}(\mathbf{q},\mathbf{u}). \qquad (65)$$

We see that Eq. (65) is well structured for direct numerical integration without inversion of the $\mathbf{A}$ matrix.

Conclusion

We have proposed and demonstrated a method for creating rigid-body equations of motion that are decoupled in first-order terms. This is achieved by properly choosing a congruency transformation that specifies generalized velocity components. For nonholonomic systems, or holonomic systems with unreduced configuration coordinates, the congruency transformation is used in conjunction with an orthogonal complement to the constraint array. In both cases, the resulting equations are in a form that make it convenient for nonlinear behavior analyses. Moreover, it becomes an easy matter to implement general integration routines for first-order differential equations to obtain generalized coordinate trajectories for simulation purposes.

Acknowledgment

The financial support of this work by US Army Research Office grant no. DAAL-03-90-G-0005 is gratefully acknowledged.

References

Ben-Israel, A., and Greville, T. N. E., 1974, *Generalized Inverses: Theory and Applications*, John Wiley and Sons, New York.

Crandall, S. H., Karnopp, D. C., Kurtz, E. F., Jr., and Pridmore-Brown, D. C., 1968, *Dynamics of Mechanical and Electromechanical Systems*, Robert E. Krieger, Malabar, FL.

Desloge, E. A., 1987, "Relationship Between Kane's Equations and the Gibbs-Appell Equations," Engineering Notes, *Journal of Guidance, Control and Dynamics*, Vol. 10, Jan.–Feb., pp. 120–122.

Gibbs, J. W., 1879, "On the Fundamental Formulae of Dynamics," *American Journal of Mathematics*, Vol. II, pp. 49–64.

Gibbs, J. W., 1961, *The Scientific Papers of J. Willard Gibbs*, Vol. II, Dover, New York.

Hartog, J. P. D., 1948, *Mechanics*, Dover Publications, New York.

Hemami, H., and Weimer, F. C., 1981, "Modeling of Nonholonomic Dynamic Systems With Applications," ASME JOURNAL OF APPLIED MECHANICS, Vol. 48, pp. 177–182.

Huston, R. L., 1990, *Multibody Dynamics*, Butterworth-Heinemann, Boston.

Huston, R. L., 1987, "On the Equivalence of Kane's Equations and

Gibbs' Equations for Multibody Dynamics Formulations," *Mechanics Research Communications*, Vol. 14, No. 2, pp. 123–131.

Kamman, J. W., and Huston, R. L., 1984, "Dynamics of Constrained Multibody Systems," ASME JOURNAL OF APPLIED MECHANICS, Vol. 51, pp. 899–903.

Kane, T. R., 1972, *Dynamics*, Stanford, CA.

Kane, T. R., and Levinson, D. A., 1985, *Dynamics: Theory and Applications*, McGraw-Hill, New York.

Lawson, C. L., and Hanson, R. J., 1974, *Solving Least Squares Problems*, Prentice-Hall, Englewood Cliffs, N.J.

Nikravesh, P. E., and Haug, E. J., 1983, "Generalized Coordinate Partitioning for Analysis of Mechanical Systems with Nonholonomic Constraints," ASME *Journal of Mechanisms, Transmissions, and Automation in Design*, Vol. 105, pp. 379–384.

Papastavridis, J. G., 1992, "Discussion on the Paper Entitled: Dynamics of Nonholonomic Mechanical Systems Using a Natural Orthogonal Complement," ASME JOURNAL OF APPLIED MECHANICS, Vol. 59, pp. 242–244.

Papastavridis, J. G., 1988, "On the Nonlinear Appell's Equations and the Determination of Generalized Reaction Forces," *Int. J. Engng. Sci.*, Vol. 26, No. 6, pp. 609–625.

Roberson, R. E., and Schwertassek, R., 1988, *Dynamics of Multibody Systems*, Springer-Verlag, Berlin.

Scott, D., 1988, "Can a Projection Method of Obtaining Equations of Motion Compete with Lagrange's Equations," *American Journal of Physics*, Vol. 56, May, pp. 451–456.

Singh, R. P., and Likins, P. W., 1985, "Generalized Coordinate Partitioning for Constrained Dynamical Systems," ASME JOURNAL OF APPLIED MECHANICS, Vol. 52, pp. 943–948.

Storch, J., and Gates, S., 1989, "Motivating Kane's Method for Obtaining Equations of Motion for Dynamic Systems," Engineering Notes, *Journal of Guidance, Control and Dynamics*, Vol. 12, July–Aug., pp. 593–595.

Wade, T. L., 1951, *The Algebra of Vectors and Matrices*, Addison-Wesley, Cambridge, MA.

Wang, J. T., and Huston, R. L., 1988, "Computational Methods in Constrained Multibody Dynamics: Matrix Formalisms," *Computers and Structures*, Vol. 29, No. 2, pp. 331–338.

Wehage, R. A., and Haug, E. J., 1982, "Generalized Coordinate Partitioning for Dimension Reduction in Analysis of Constrained Dynamic Systems," ASME *Journal of Mechanical Design*, Vol. 104, pp. 247–255.

Wampler, C., Buffinton, K., and Shu-hui, J., 1985, "Formulation of Equations of Motion for Systems Subject to Constraints," ASME JOURNAL OF APPLIED MECHANICS, Vol. 52, pp. 465–470.

Xu, M., Liu, C., and Huston, R. L., 1990, "Analysis of Nonlinearly Constrained Non-Holonomic Multibody Systems," *International Journal of Non-Linear Mechanics*, Vol. 25, No. 5, pp. 511–519.

0116658

DE-Vol. 69-1, Advances in Design Automation – 1994
Volume 1
ASME 1994

MOTION INTERPOLATION USING DYNAMICS

T. A. Loduha and B. Ravani
Department of Mechanical and Aeronautical Engineering
University of California, Davis
Davis, California

ABSTRACT

In this paper we present a method for dynamically interpolating the motion of a rigid body in space between known end-positions. We begin by creating an image space representation of the equations of motion for a rigid body. These equations are written as a system of first-order state equations whose trajectory is optimized based on the minimization of a certain performance index. This leads to a set of boundary-value equations between the end positions, which when solved give the interpolated motion. Finally, a program developed to perform such interpolations numerically is utilized by generating a trajectory between given terminal states.

INTRODUCTION

Within the field of computer graphics, especially as it applies to mechanical system simulation and computer animation the problem of finding a mathematical representation of a rigid body motion that approximates a sequence of arbitrary displacements is commonly encountered. In other words the end-positions of a rigid body spatial trajectory are known, but the trajectory itself is unknown. Throughout this paper, the term 'position' refers to the position of a point on a rigid body, as well as the orientation of the body. The problem of determining the intermediate positions has been referred to as key framing, or geometric design of motion interpolants (see Ravani and Ge (1994)). It has applications in computer graphics, computer vision, robotics, bio-mechanics, other fields. Figure 1 illustrates the interpolated trajectory of a rigid body. Many researchers have explored methods to *interpolate* the unknown motion of the body. Traditionally, the problem has been separated into interpolations of translations and rotations. Dealing with the translation is straight forward, however the rotation interpolation poses considerable difficulties. Initially, attempts were made at individual interpolation of the Euler angles, but the resulting motions seemed to speed up or slow down. More recently, Shoemake (1985) and Pletickx (1989) investigated subdivision methods using quaternions to interpolate rotations. Ravani and Ge (1994) transform the motion interpolation problem into a point interpolation in the image space, and extend the results of Shoemake by properly considering the geometry of the underlying space of three dimensional displacements. Needless to say, the common trait of these developments is their kinematic foundation. That is, these methods have provided ways to create the unknown trajectory without regard to the dynamics of the motion of the rigid body being animated. The motivation behind the developments in this paper is to *dynamically* interpolate the motion of the body between the given end-positions for animation and other purposes. Thus the result is a dynamically consistent specification of the pathway followed by the body, as well as its velocity distribution.

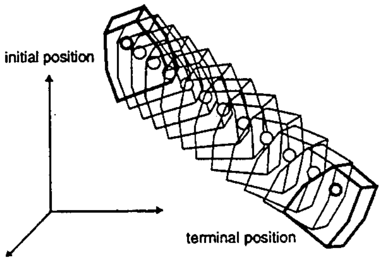


Figure 1: Dynamically Interpolated Motion

To solve the problem of dynamically-based interpolation, an image space representation of the equations of motion for a single rigid body is first created. These equations become the system equations for a trajectory optimization based on the minimization of a certain performance index. The solution of the optimization problem describes the pathway of an image point, and the velocity of that point along the path.

This can then be resolved into finite and infinitesimal rigid body displacement information specifying the reconstructed motion.

MOTION REPRESENTATION

In this section we convert all kinematical entities such as Euclidean point position and velocity, rigid body orientation parameters, and angular velocity of the body to an image space representation. The new kinematical description is used to generate a unique set of equations of motion in terms of an image space state vector. We see that the image space allows us to treat the position and orientation of a rigid body in space as a single point in a dual four-dimensional space. More importantly, the time derivative of this image space point position gives us an unambiguous representation of the rate of change in orientation of the body. Hence, we can introduce a sixteenth order state vector that contains the orientation and time rate of change of orientation data. In an essence, the image space helps us circumvent the incomplete analogy between velocity and angular velocity. By "incomplete analogy" we mean that in the Euclidean space of three dimensions velocity can be found from the derivative of a position vector, but angular velocity, in general, can not be determined from the derivative of an orientation vector. With the mapping to the image space, position and orientation are captured as one entity - the image space position. The derivative of this point gives us the velocity and angular velocity of the body. The image space equations of motion then become the first order state space system whose trajectory is optimized based on effort minimization.

Newton–Euler Equations in the Image Space

Our goal in this section is to map the equations describing the dynamical motion of a rigid body to the image space. The kinematic advantages of the image space are illustrated by Ravani and Ge (1994), and here we will demonstrate various dynamic benefits of this representation. Using the image space representation, the components of a point position in the image space become the configuration coordinates, as opposed to the Euclidean position and angular orientation coordinates ordinarily used. The final results of this section are the equations of motion for a single rigid body in state space form. They will be first-order nonlinear equations in terms of a state vector z (to be defined later). The components of this vector completely encompass the rigid body displacement and time rate of change of displacement information. Here, we begin with the well-known Newton-Euler equations of motion for a rigid body, and transform them to a system of sixteen first order equations.

The Newton-Euler equations for a rigid body with a body fixed coordinate system aligned in the principal directions and located at the center of mass are written as

$$
\begin{aligned}
f_1 &= m\ddot{d}_1 \\
f_2 &= m\ddot{d}_2 \\
f_3 &= m\ddot{d}_3
\end{aligned}
\tag{1}
$$

and

$$
\begin{aligned}
\tau_1 &= I_1\dot{\omega}_1 - (I_2 - I_3)\omega_2\omega_3 \\
\tau_2 &= I_2\dot{\omega}_2 - (I_3 - I_1)\omega_3\omega_1 \\
\tau_3 &= I_3\dot{\omega}_3 - (I_1 - I_2)\omega_1\omega_2
\end{aligned}
\tag{2}
$$

where f_i and τ_i $(i = 1, 2, 3)$ are the applied moments and forces in the $\mathbf{e}_1$, $\mathbf{e}_2$ and $\mathbf{e}_3$ directions of a body-fixed coordinate frame. I_1 , I_2 and I_3 are the principal centroidal moments of inertia and ω_i $(i = 1, 2, 3)$ are the angular velocity components expressed in terms of the body-fixed frame. We can now convert all kinematical entities to the image space. If the image space point $\hat{\mathbf{x}}$ is written in quaternion form as

$$\hat{\mathbf{x}} = \hat{x}_1\mathbf{i} + \hat{x}_2\mathbf{j} + \hat{x}_3\mathbf{k} + \hat{x}_4 \tag{3}$$

the dual velocity distribution of the body can be expressed as the following quaternion product:

$$\hat{\mathbf{V}} = 2\dot{\hat{\mathbf{x}}}\hat{\mathbf{x}}^* \tag{4}$$

where $\hat{\mathbf{x}}^*$ is the conjugate of the dual quaternion. This dual velocity entity can be thought of as a dual number combination of angular and linear velocity and written as

$$\hat{\mathbf{V}} = \begin{bmatrix} \omega \\ 0 \end{bmatrix} + \epsilon \begin{bmatrix} \dot{\mathbf{d}} \\ 0 \end{bmatrix} \tag{5}$$

where ω is the angular velocity of the body and $\dot{\mathbf{d}}$ is the velocity vector of the origin of the body-fixed frame. The dual quaternion product can be expanded in terms of the dual Euler parameters and their derivatives, and written in matrix form as

$$\hat{\mathbf{V}} = 2\dot{\hat{\mathbf{D}}}\hat{\mathbf{x}} \tag{6}$$

where the dual matrix $\hat{\mathbf{D}}$ is written as

$$\hat{\mathbf{D}} = \mathbf{D} + \epsilon\mathbf{D}^0. \tag{7}$$

The real part of $\hat{\mathbf{D}}$ is

$$\mathbf{D} = \begin{bmatrix} -x_4 & x_3 & -x_2 & x_1 \\ -x_3 & -x_4 & x_1 & x_2 \\ x_2 & -x_1 & -x_4 & x_3 \\ x_1 & x_2 & x_3 & x_4 \end{bmatrix} \tag{8}$$

and the dual part is

$$\mathbf{D}^0 = \begin{bmatrix} -x_4^0 & x_3^0 & -x_2^0 & x_1^0 \\ -x_3^0 & -x_{4^0} & x_1^0 & x_2^0 \\ x_2^0 & -x_1^0 & -x_4^0 & x_3^0 \\ x_1^0 & x_2^0 & x_3^0 & x_4^0 \end{bmatrix}. \tag{9}$$

The real and dual parts of the first three components of $\hat{\mathbf{V}}$ comprise, respectively, the angular velocity and the velocity of the origin of the body's fixed frame, as stated in Equation 5. The matrix product of Equation 6 can be separated into real and dual parts as

$$\hat{\mathbf{V}} = 2\dot{\mathbf{D}}\mathbf{x} + 2\epsilon(\dot{\mathbf{D}}\mathbf{x}^0 + \dot{\mathbf{D}}^0\mathbf{x}). \tag{10}$$

Therefore, using this relation and Equation 5, the angular and linear velocity terms become

$$\begin{bmatrix} \omega \\ 0 \end{bmatrix} = 2\dot{\mathbf{D}}\mathbf{x} \tag{11}$$

and

$$\begin{bmatrix} \dot{\mathbf{d}} \\ 0 \end{bmatrix} = 2(\dot{\mathbf{D}}\mathbf{x}^0 + \dot{\mathbf{D}}^0\mathbf{x}). \tag{12}$$

From these we can obtain the necessary derivatives and products $\dot{\omega}_1$, $\dot{\omega}_2$ and $\dot{\omega}_3$, $\ddot{d}_1$, $\ddot{d}_2$ and $\ddot{d}_3$, and $\omega_2\omega_3$, $\omega_3\omega_1$ and $\omega_1\omega_2$ in terms of image space components to be substituted into Equations 1 and 2. However, before doing this, it is helpful to define a sixteen element state vector $\mathbf{z}$ as

$$\mathbf{z} = \begin{bmatrix} \dot{\mathbf{x}}_i \\ \dot{\mathbf{x}}_i^0 \\ \mathbf{x}_i \\ \mathbf{x}_i^0 \end{bmatrix} = 2\epsilon(\dot{\mathbf{D}}\mathbf{x}^0 + \dot{\mathbf{D}}^0\mathbf{x}). \tag{13}$$

We also define

$$\mathbf{z}_a = \begin{bmatrix} z_1 & z_2 & \ldots\ldots & z_8 \end{bmatrix}^T \tag{14}$$

and

$$\mathbf{z}_b = \begin{bmatrix} z_9 & z_{10} & \ldots\ldots & z_{16} \end{bmatrix}^T. \tag{15}$$

$\mathbf{z}_b$ can be thought of as the position of a point in an eight-dimensional vector space. This position corresponds to a homogeneous representation of the position and orientation of the rigid body in three dimensions. Analogously, $\mathbf{z}_a$ is the velocity of that point (note $\mathbf{z}_a = \dot{\mathbf{z}}_b$), and defines the time rate of change of position and orientation of the body.

Equation 11 is expanded and elements of state vector $\mathbf{z}$ are substituted to get the components of the angular velocity vector as

$$\begin{aligned}
\omega_1 &= 2(-z_4z_9 + z_3z_{10} - z_2z_{11} + z_1z_{12}) \\
\omega_2 &= 2(-z_3z_9 - z_4z_{10} + z_1z_{11} + z_2z_{12}) \\
\omega_3 &= 2(z_2z_9 - z_1z_{10} - z_4z_{11} + z_3z_{12}).
\end{aligned} \tag{16}$$

These expressions are time differentiated to yield, after some simplification,

$$\begin{aligned}
\dot{\omega}_1 &= 2(-\dot{z}_4z_9 + \dot{z}_3z_{10} - \dot{z}_2z_{11} + \dot{z}_1z_{12}) \\
\dot{\omega}_2 &= 2(-\dot{z}_3z_9 - \dot{z}_4z_{10} + \dot{z}_1z_{11} + \dot{z}_2z_{12}) \\
\dot{\omega}_3 &= 2(\dot{z}_2z_9 - \dot{z}_1z_{10} - \dot{z}_4z_{11} + \dot{z}_3z_{12}).
\end{aligned} \tag{17}$$

In a similar manner we obtain the velocity and acceleration components of the origin of the body-fixed frame:

$$\begin{aligned}
\dot{d}_1 &= 2(-z_4z_{13} + z_3z_{14} - z_2z_{15} + z_1z_{16} - \\
&\quad z_8z_9 + z_7z_{10} - z_6z_{11} + z_5z_{12}) \\
\dot{d}_2 &= 2(-z_3z_{13} - z_4z_{14} + z_1z_{15} + z_2z_{16} - \\
&\quad z_7z_9 - z_8z_{10} + z_5z_{11} + z_6z_{12}) \\
\dot{d}_3 &= 2(z_2z_{13} - z_1z_{14} - z_4z_{15} + z_3z_{16} + \\
&\quad z_6z_9 - z_5z_{10} - z_8z_{11} + z_7z_{12})
\end{aligned} \tag{18}$$

and, upon time differentiation,

$$\begin{aligned}
\ddot{d}_1 &= 2(-\dot{z}_4z_{13} + \dot{z}_3z_{14} - \dot{z}_2z_{15} + \dot{z}_1z_{16} - \\
&\quad \dot{z}_8z_9 + \dot{z}_7z_{10} - \dot{z}_6z_{11} + \dot{z}_5z_{12}) \\
\ddot{d}_2 &= 2(-\dot{z}_3z_{13} - \dot{z}_4z_{14} + \dot{z}_1z_{15} + \dot{z}_2z_{16} - \\
&\quad \dot{z}_7z_9 - \dot{z}_8z_{10} + \dot{z}_5z_{11} + \dot{z}_6z_{12}) \\
\ddot{d}_3 &= 2(\dot{z}_2z_{13} - \dot{z}_1z_{14} - \dot{z}_4z_{15} + \dot{z}_3z_{16} + \\
&\quad \dot{z}_6z_9 - \dot{z}_5z_{10} - \dot{z}_8z_{11} + \dot{z}_7z_{12}).
\end{aligned} \tag{19}$$

Finally, we denote the angular velocity product quantities as:

$$\begin{aligned}
h_{12} &= \omega_1\omega_2 \\
h_{23} &= \omega_2\omega_3 \\
h_{13} &= \omega_1\omega_3
\end{aligned} \tag{20}$$

The above quantities are substituted into Equations 1 and 2, and the six Newton-Euler equations become

$$\begin{aligned}
u_1 &= 2m(-\dot{z}_4z_{13} + \dot{z}_3z_{14} - \dot{z}_2z_{15} + \dot{z}_1z_{16} - \\
&\quad \dot{z}_8z_9 + \dot{z}_7z_{10} - \dot{z}_6z_{11} + \dot{z}_5z_{12}) \\
u_2 &= 2m(-\dot{z}_3z_{13} - \dot{z}_4z_{14} + \dot{z}_1z_{15} + \dot{z}_2z_{16} - \\
&\quad \dot{z}_7z_9 - \dot{z}_8z_{10} + \dot{z}_5z_{11} + \dot{z}_6z_{12}) \\
u_3 &= 2m(\dot{z}_2z_{13} - \dot{z}_1z_{14} - \dot{z}_4z_{15} + \dot{z}_3z_{16} + \\
&\quad \dot{z}_6z_9 - \dot{z}_5z_{10} - \dot{z}_8z_{11} + \dot{z}_7z_{12})
\end{aligned} \tag{21}$$

and

$$\begin{aligned}
u_4 &= 2I_1(-\dot{z}_4z_9 + \dot{z}_3z_{10} - \dot{z}_2z_{11} + \dot{z}_1z_{12}) - \\
&\quad (I_2 - I_3)h_{23} \\
u_5 &= 2I_2(-\dot{z}_3z_9 - \dot{z}_4z_{10} + \dot{z}_1z_{11} + \dot{z}_2z_{12}) - \\
&\quad (I_3 - I_1)h_{31} \\
u_6 &= 2I_3(\dot{z}_2z_9 - \dot{z}_1z_{10} - \dot{z}_4z_{11} + \dot{z}_3z_{12}) - \\
&\quad (I_1 - I_2)h_{12}
\end{aligned} \tag{22}$$

where $\mathbf{u}$ is simply a vector of applied forces and torques, and is written here as

$$\mathbf{u} = \begin{bmatrix} f_1 & f_2 & f_3 & \tau_1 & \tau_2 & \tau_3 \end{bmatrix}^T. \tag{23}$$

Since the equations of motion are now written in terms of eight configuration coordinates, we need an additional two equations to have a complete set. These can be generated from the image space constraints given in (Ravani and Ge, 1994). We rewrite them here in terms of the state vector $\mathbf{z}$ as

$$z_9^2 + z_{10}^2 + z_{11}^2 + z_{12}^2 = 1 \tag{24}$$

and

$$z_9z_{13} + z_{10}z_{14} + z_{11}z_{15} + z_{12}z_{16} = 0. \tag{25}$$

If we time differentiate each of these equations twice we obtain the two additional state equations as

$$\begin{aligned}
&\dot{z}_1z_{13} + \dot{z}_5z_9 + \dot{z}_2z_{14} + \dot{z}_6z_{10} + \\
&\dot{z}_3z_{15} + \dot{z}_7z_{11} + \dot{z}_4z_{16} + \dot{z}_8z_{12} + \\
&2(z_1z_5 + z_2z_6 + z_3z_7 + z_4z_8) = 0
\end{aligned} \tag{26}$$

and

$$\dot{z}_1 z_{13} + \dot{z}_5 z_9 + \dot{z}_2 z_{14} + \dot{z}_6 z_{10} + z_1^2 + z_2^2 + z_3^2 + z_4^2 = 0. \quad (27)$$

Equations 21, 22, 26 and 27 can be ordered, and put into matrix format, yielding

$$\mathbf{H}\dot{\mathbf{z}}_a = \mathbf{g} \quad (28)$$

where

$$\mathbf{g} = \begin{bmatrix} u_4/2I_1 + (I_2 - I_3)h_{23}/2I_1 \\ u_5/2I_2 + (I_3 - I_1)h_{31}/2I_2 \\ u_6/2I_3 + (I_1 - I_2)h_{12}/2I_3 \\ -z_1^2 - z_2^2 - z_3^2 - z_4^2 \\ u_1/2m \\ u_2/2m \\ u_3/2m \\ -2(z_1 z_5 + z_2 z_6 + z_3 z_7 + z_4 z_8) \end{bmatrix}. \quad (29)$$

The coefficient matrix H is partitioned as

$$\mathbf{H} = \begin{bmatrix} \mathbf{C} & \mathbf{0} \\ \mathbf{A} & \mathbf{B} \end{bmatrix} \quad (30)$$

where

$$\mathbf{A} = \begin{bmatrix} -z_{16} & z_{15} & -z_{14} & z_{13} \\ -z_{15} & -z_{16} & z_{13} & z_{14} \\ z_{14} & -z_{13} & -z_{16} & z_{15} \\ z_{13} & z_{14} & z_{15} & z_{16} \end{bmatrix}, \quad (31)$$

$$\mathbf{B} = \begin{bmatrix} z_{12} & -z_{11} & z_{10} & -z_9 \\ z_{11} & z_{12} & -z_9 & -z_{10} \\ -z_{10} & z_9 & z_{12} & -z_{11} \\ z_9 & z_{10} & z_{11} & z_{12} \end{bmatrix} \quad (32)$$

and

$$\mathbf{C} = \mathbf{B}. \quad (33)$$

The advantage of this particular ordering becomes clear when we consider one of the Schur formulas for the inverse of a partitioned matrix (see, for example, Brewer (Brewer, 1990)). This formula states that the matrix H partitioned as shown in Equation 30 has the inverse

$$\mathbf{H}^{-1} = \begin{bmatrix} \mathbf{C}^{-1} & \mathbf{0} \\ -\mathbf{B}^{-1}\mathbf{A}\mathbf{C}^{-1} & \mathbf{B}^{-1} \end{bmatrix}. \quad (34)$$

Symbolic inversion of H becomes even easier when we utilize the fact that B is an orthogonal matrix. It is a simple matter to verify the orthogonality of B by realizing it satisfies the following two conditions: the inner products between each of the column vectors, or between each row vectors comprising B are zero, and each row or column vector is of unit length. The first condition can be visually verified and the second follows from Equation 24. Therefore $\mathbf{B}^{-1} = \mathbf{B}^T$, and $\mathbf{H}^{-1}$ can be written as

$$\mathbf{H}^{-1} = \begin{bmatrix} \mathbf{B}^T & \mathbf{0} \\ -\mathbf{B}^T\mathbf{A}\mathbf{B}^T & \mathbf{B}^T \end{bmatrix}. \quad (35)$$

Hence, we can rewrite Equation 28 as

$$\dot{\mathbf{z}}_a = \mathbf{H}^{-1}\mathbf{g} \quad (36)$$

keeping in mind that numerical inversion of H is not necessary. The complete equations of motion are

$$\dot{\mathbf{z}} = \begin{bmatrix} \mathbf{H}^{-1}\mathbf{g} \\ \mathbf{z}_a \end{bmatrix}. \quad (37)$$

Notice these equations are first-order decoupled - a characteristic that is necessary to initiate the development of the optimal control problem in the next section.

OPTIMAL TRAJECTORY GENERATION

If the image space rigid body equations of motion are viewed as a nonlinear state space system we can cast the dynamic interpolation problem into the determination of an optimal trajectory based on optimal control theory. In this paper the optimal rigid body trajectory is the one that minimizes the total force and torque applied to the body, as well as their derivatives, over a certain time period. Other performance indices could be chosen, such as power or total energy minimization. However, we restrict our analysis to applied force and torque minimization in the following development. The equations of motion depicted in Equations 1 and 2 can be written as

$$\dot{\mathbf{z}} = \mathbf{a}(\mathbf{z}, \mathbf{u}) \quad (38)$$

where, according to Equation 37

$$\mathbf{a}(\mathbf{z}, \mathbf{u}) = \begin{bmatrix} \mathbf{H}^{-1}\mathbf{g} \\ \mathbf{z}_a \end{bmatrix}. \quad (39)$$

The performance measure we wish to minimize is

$$J(\mathbf{u}) = \int_{t_i}^{t_f} g(\mathbf{u})dt \quad (40)$$

where

$$g(\mathbf{u}) = \frac{1}{2}\mathbf{u}^T\mathbf{u}. \quad (41)$$

It is convenient to use the function $\mathcal{H}$, referred to as the Hamiltonian, which is defined as (see, for example, Bryson and Ho (1975) and Kirk (1970))

$$\mathcal{H} = g(\mathbf{u}) + \lambda^T\mathbf{a} \quad (42)$$

where λ is a vector of Lagrange multipliers. From this we can state the necessary conditions to determine an optimal trajectory as

$$\dot{\mathbf{z}} = \frac{\partial\mathcal{H}}{\partial\lambda}, \quad (43)$$

(which is equivalent to Equation 38)

$$\dot{\lambda} = -\frac{\partial\mathcal{H}}{\partial\mathbf{z}} \quad (44)$$

and

$$0 = \frac{\partial\mathcal{H}}{\partial\mathbf{u}}. \quad (45)$$

Equations 43, 44 and 45 comprise a set of thirty two boundary value equations, which when solved, give us the optimal state trajectories of the body. These states translate to mass center position and velocity, and orientation and rate of change of orientation trajectories for the body between the given endpoints. Naturally, these trajectories satisfy the boundary conditions imposed by the analyst. However, it must be emphasized that Equations 43, 44 and 45 do not require the force and torque components to satisfy any specific boundary conditions. Hence, Equation 38 should be modified to make the necessary force and moment accommodations. We can append the relation

$$\dot{\mathbf{u}} = \mathbf{w} \tag{46}$$

to Equation 38 to obtain

$$\dot{\mathbf{z}} = \mathbf{a}(\mathbf{z}, \mathbf{w}) \tag{47}$$

where the state vector $\mathbf{z}$ is now a 22-element vector containing the original sixteen states along with the components of force and torque. With this adjustment the forces and moments are now treated as states, and $\mathbf{w}$, the time derivative of the applied force vector, becomes the control vector. The vector $\mathbf{a}$ is rewritten as

$$\mathbf{a}(\mathbf{z}, \mathbf{w}) = \begin{bmatrix} \mathbf{H}^{-1}\mathbf{g} \\ \mathbf{z}_a \\ \mathbf{w} \end{bmatrix}. \tag{48}$$

We also alter g to include the new control vector:

$$g(\mathbf{u}, \mathbf{w}) = \frac{1}{2}(\mathbf{u}^T\mathbf{u} + \mathbf{w}^T\mathbf{w}) \tag{49}$$

Equation 44 remains the same as given, keeping in mind that z and $\mathcal{H}$ have changed. Equation 45 becomes

$$0 = \frac{\partial \mathcal{H}}{\partial \mathbf{w}}. \tag{50}$$

The modified set of boundary value equations given by Equations 47, 44 and 50 allow the additional specification of the force state at each endpoint, and the resulting force trajectory must satisfy these conditions. The modification of g to include the time derivative of the u will help minimize sharp changes in the applied forces to the body. We continue by further analyzing the form of the co-state equations.

For notational ease define the Lagrange multiplier vectors

$$\lambda_a = \begin{bmatrix} \lambda_1 & \lambda_2 & \ldots\ldots & \lambda_8 \end{bmatrix}^T, \tag{51}$$

$$\lambda_b = \begin{bmatrix} \lambda_9 & \lambda_{10} & \ldots\ldots & \lambda_{16} \end{bmatrix}^T \tag{52}$$

and

$$\lambda_c = \begin{bmatrix} \lambda_{17} & \lambda_{18} & \ldots\ldots & \lambda_{22} \end{bmatrix}^T \tag{53}$$

thus,

$$\lambda = \begin{bmatrix} \lambda_a & | & \lambda_b & | & \lambda_c \end{bmatrix}^T. \tag{54}$$

Next, we partition z similarly as

$$\mathbf{z} = \begin{bmatrix} \mathbf{z}_a & | & \mathbf{z}_b & | & \mathbf{z}_c \end{bmatrix}^T \tag{55}$$

thus,

$$\dot{\mathbf{z}}_a = \mathbf{H}^{-1}\mathbf{g}, \tag{56}$$

$$\dot{\mathbf{z}}_b = \mathbf{z}_a \tag{57}$$

and

$$\dot{\mathbf{z}}_c = \mathbf{w}. \tag{58}$$

The partial derivative given in Equation 43 can now be separated, taking advantage of the differing composition of the sub-components of $\dot{\mathbf{z}}$. The first portion is comprised of the first eight elements of λ, and is written as

$$\dot{\lambda}_a^T = -\lambda^T \frac{\partial \dot{\mathbf{z}}}{\partial \mathbf{z}_a^T} \tag{59}$$

or

$$\dot{\lambda}_a^T = -(\lambda_a^T \frac{\partial \dot{\mathbf{z}}_a}{\partial \mathbf{z}_a^T} + \lambda_b^T \frac{\partial \dot{\mathbf{z}}_b}{\partial \mathbf{z}_a^T} + \lambda_c^T \frac{\partial \dot{\mathbf{z}}_c}{\partial \mathbf{z}_a^T}). \tag{60}$$

Upon substitution of expressions from Equations 56, 57 and 58 the last equation becomes

$$\dot{\lambda}_a^T = -(\lambda_a^T \frac{\partial(\mathbf{H}^{-1}\mathbf{g})}{\partial \mathbf{z}_a^T} + \lambda_b^T \frac{\partial \mathbf{z}_a}{\partial \mathbf{z}_a^T} + \lambda_c^T \frac{\partial \mathbf{w}}{\partial \mathbf{z}_a^T}). \tag{61}$$

$\mathbf{H}$ and $\mathbf{w}$ are not functions of the first eight elements of $\mathbf{z}$, which comprise $\mathbf{z}_a$, so we simplify Equation 61 into

$$\dot{\lambda}_a^T = -(\lambda_a^T \mathbf{H}^{-1} \frac{\partial \mathbf{g}}{\partial \mathbf{z}_a^T} + \lambda_b^T). \tag{62}$$

The second partial derivative portion is

$$\dot{\lambda}_b^T = -\lambda^T \frac{\partial \dot{\mathbf{z}}}{\partial \mathbf{z}_b^T} \tag{63}$$

or

$$\dot{\lambda}_b^T = -(\lambda_b^T \frac{\partial(\mathbf{H}^{-1}\mathbf{g})}{\partial \mathbf{z}_b^T} + \lambda_b^T \frac{\partial \mathbf{z}_a}{\partial \mathbf{z}_b^T} + \lambda_c^T \frac{\partial \mathbf{w}}{\partial \mathbf{z}_b^T}). \tag{64}$$

Here, we can use some matrix manipulation and the kronecker product "⊗" to rewrite Equation 64, devoid of the partial derivative of $\mathbf{H}^{-1}$, as

$$\dot{\lambda}_b^T = -\lambda_b^T(-\mathbf{H}^{-1} \frac{\partial \mathbf{H}}{\partial \mathbf{z}_b^T}(\mathbf{I}_8 \otimes \mathbf{H}^{-1})(\mathbf{I}_8 \otimes \mathbf{g}) + \mathbf{H}^{-1} \frac{\partial \mathbf{g}}{\partial \mathbf{z}_b^T}). \tag{65}$$

Finally, since performance measure g is a function of u we must write

$$\dot{\lambda}_c^T = -\frac{\partial g}{\partial \mathbf{z}_c^T} - \lambda^T \frac{\partial \dot{\mathbf{z}}}{\partial \mathbf{z}_c^T} \tag{66}$$

or

$$\dot{\lambda}_c^T = -\frac{\partial \mathbf{u}^T\mathbf{u}}{\partial \mathbf{z}_c^T} - (\lambda_a^T \frac{\partial(\mathbf{H}^{-1}\mathbf{g})}{\partial \mathbf{z}_c^T} + \lambda_b^T \frac{\partial \mathbf{z}_a}{\partial \mathbf{z}_c^T} + \lambda_c^T \frac{\partial \mathbf{w}}{\partial \mathbf{z}_c^T}). \tag{67}$$

Recall that $\mathbf{z}_c^T$ is actually the applied force vector u so Equation 67 is written as

$$\dot{\lambda}_c^T = -\lambda_a^T \mathbf{H}^{-1} \frac{\partial \mathbf{g}}{\partial \mathbf{u}^T} - \frac{1}{2} \frac{\partial \mathbf{u}^T\mathbf{u}}{\partial \mathbf{u}^T} \tag{68}$$

which becomes

$$\dot{\lambda}_c^T = -\lambda_a^T \mathbf{H}^{-1} \frac{\partial \mathbf{g}}{\partial \mathbf{u}^T} - \mathbf{u}^T. \tag{69}$$

Equations 62, 65 and 69, often referred to as the *co-state* or *adjoint* equations (1975; 1970), complete the set of forty

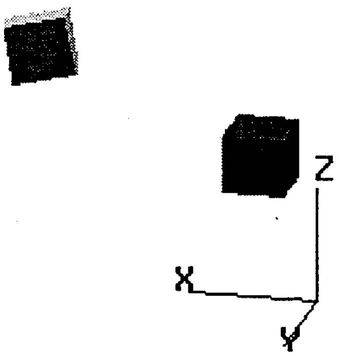


Figure 2: Initial and Final Positions

four boundary value equations. However, notice that the control vector w appears explicitly in vector a in the state equations. Since this vector is unknown prior to the integration of the boundary equations, it must be replaced with components of z and λ. The condition stated in Equation 45 provides the relationship to enable this substitution. Equation 45 can be expanded to give

$$0 = \frac{\partial g(\mathbf{u},\mathbf{w})}{\partial \mathbf{w}^T} + \lambda^T \frac{\partial \dot{\mathbf{z}}}{\partial \mathbf{w}^T}. \qquad (70)$$

This becomes

$$0 = \frac{1}{2}\frac{\partial \mathbf{w}^T\mathbf{w}}{\partial \mathbf{w}^T} + \lambda_c^T \frac{\partial \mathbf{w}}{\partial \mathbf{w}^T} \qquad (71)$$

or

$$\mathbf{w}^T = -\lambda_c^T. \qquad (72)$$

Equation 72 is the explicit expression for the control vector enabling elimination of w from the state equations and costate equations. We illustrate these ideas with the following example.

EXAMPLE

The ideas developed in this paper are demonstrated here through numerical integration of the boundary value equations just derived. We will perform dynamic interpolation of a rigid body motion between an initial and final state. The information comprising these states are the image space velocity $\mathbf{z}_a$, the image space position $\mathbf{z}_b$ and the vector of applied forces and torques $\mathbf{u}$. For the initial state we have

$$\mathbf{z}_a = \begin{bmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \end{bmatrix} \mathbf{z}_b = \begin{bmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 1.0 \\ 0.25 \\ 0.25 \\ 0.25 \\ 0.0 \end{bmatrix} \mathbf{u} = \begin{bmatrix} 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \\ 0.0 \end{bmatrix}, \qquad (73)$$

and at the final state

$$\mathbf{z}_a = \begin{bmatrix} 0.37 \\ 0.37 \\ 0.38 \\ -2.0 \\ -1.9 \\ -1.9 \\ -1.9 \\ -8.3 \end{bmatrix} \mathbf{z}_b = \begin{bmatrix} 0.55 \\ 0.55 \\ 0.55 \\ 0.30 \\ 0.48 \\ 0.47 \\ 0.47 \\ -2.5 \end{bmatrix} \mathbf{u} = \begin{bmatrix} 3.1 \\ 3.1 \\ 3.0 \\ 3.2 \\ 3.1 \\ 3.2 \end{bmatrix}. \qquad (74)$$

The initial and final positions of the rigid body appear as shown in Figure 2.

The elapsed time during the motion interpolation is a user-defined parameter. For this example we shall choose $\Delta t = 2.0$ seconds. The resulting graphical representation of the dynamically interpolated motion is illustrated in Figure 3.

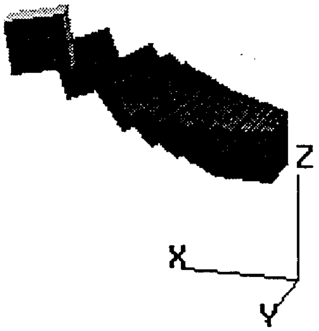


Figure 3: Dynamically Interpolated Motion

CONCLUSION

We have proposed and demonstrated a method for dynamically interpolating the spatial motion of a rigid body. The method relies on generating the equations of motion for a rigid body using image space kinematic entities. The boundary value equations that result from the trajectory optimization yield a rigid body motion that is dynamically consistent between the given terminal states.

ACKNOWLEDGEMENT

This work was supported in part by US Army Research Office grant no. DAAL-03-90-G-0005 and by the California Department of Transportation through the AHMCT program.

REFERENCES

Brewer, J. W., 1990, *Feedback Control Theory: An Introduction to the Mathematical and Historical Foundations*, EME 272 Class Notes, Davis, CA.

Bryson, A. E. J., and Ho, Y. C., 1975, *Applied Optimal Control*, Hemisphere, New York.

Kirk, D. E., 1970, *Optimal Control Theory, An Introduction*, Prentice-Hall, Englewood Cliffs, NJ.

Pletinckx, D., 1989, "Quaternion Calculus as a Basic Tool in Computer Graphics," *The Visual Computer*, Vol. 5, pp. 2–13.

Ravani, B., and Ge, Q. J., 1994, "Computer Aided Geometric Design of Motion Interpolants," To appear in ASME Journal of Mechanical Design.

Shoemake, K., 1985, "Animating Rotation with Quaternion Curves," *ACM Siggraph*, Vol. 19, No. 3, pp. 245–254.