Lehigh University
Lehigh Preserve

ATLSS Reports | Civil and Environmental Engineering

12-1-1993

Dynamic System Modelling Techniques Applicable to Force-Controlled Stewart Platform Systems

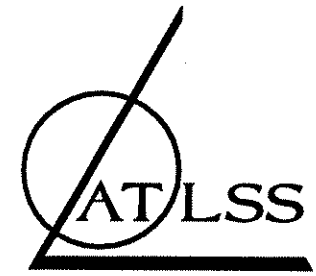
N. Duke Perreira

Janzen Lo

Follow this and additional works at: http://preserve.lehigh.edu/engr-civil-environmental-atlss-reports

Recommended Citation
Perreira, N. Duke and Lo, Janzen, "Dynamic System Modelling Techniques Applicable to Force-Controlled Stewart Platform Systems" (1993). ATLSS Reports. ATLSS report number 93-13:.
http://preserve.lehigh.edu/engr-civil-environmental-atlss-reports/192

This Technical Report is brought to you for free and open access by the Civil and Environmental Engineering at Lehigh Preserve. It has been accepted for inclusion in ATLSS Reports by an authorized administrator of Lehigh Preserve. For more information, please contact preserve@lehigh.edu.



ADVANCED TECHNOLOGY FOR
LARGE
STRUCTURAL SYSTEMS

Lehigh University

Dynamic System Modelling Techniques Applicable to Force-Controlled Stewart Platform Systems

by

N. Duke Perreira
Associate Professor of Mechanical Engineering

Janzen Lo
Research Assistant

ATLSS Report No. 93 -13

December 1993

ATLSS Engineering Research Center
Lehigh University
117 ATLSS Dr., Imbt Laboratories
Bethlehem, PA 18015-4729
(215) 758-3525

An NSF Sponsored Engineering Research Center

OVERVIEW

This two part ATLSS report is an overview of dynamic system modelling techniques directly applicable to linkage and robotic mechanisms. The specific application motivating the study was the simulation and control of a **Stewart Platform.**

The two primary goals of the effort have been to:

Create a single document that overviews all of the available modelling techniques applicable to such systems including the identification of major computer programs such as DADS, ADAMS, and TREETOPS that utilize these techniques and the primary reference sources that provide further technical information. These are the contents of Part II.

Identify from within this list a method or methods that have the potential for application in a new form of force control where models of reaction forces as functions of state variables are used, and identify briefly sample systems that would benefit from such controllers. This information is contained in summary form in Part I and in more detail in Part II.

An appropriate modelling scheme that meets the second goal has been identified and has been applied, as well as a more classical approach, to the simple example of a simple pendulum to show the relative benefits.

PART I -- COMPARISON OF APPROACHES

ABSTRACT

In this Part I, a brief overview of dynamic system modelling procedures is presented through the use of the simple example of a pendulum. It is shown that system equations of motion directly relating state variables to internal constraint forces can be generated through an automatic procedure. Such representations are a requirement for robust force controllers, an as yet not-achieved occurrence.

INTRODUCTION

Dynamic system modelling techniques are currently used to simulate open (serial) and closed (parallel) kinematic chains as well as within model driven controllers. Both areas have been found to be deficient.

Simulators that utilize computer-aided equation generation techniques tend to generate large sets of coupled differential-algebraic equations whose reduction in order and solution are prone to singularities and instabilities.

Robust model driven controllers require dynamic system models that incorporate state variables or their estimates. Since all currently available dynamic system modelers can easily predict position and velocity, these quantities can be easily and robustly controlled. Predictions of internal forces, on the other hand, as functions of position and velocity, the most typical state variables, are not available. Thus the stability and convergence of current force controllers is questionable. The need for force control includes such tasks as: part insertion using robots, metal cutting using machine tools, car suspensions and passenger comfort, utilizing high speed cam-followers and controlling buildings prone to earthquake loading.

Modelling techniques for both rigid and deformable systems are ill tuned to determining constraint forces as explicit functions of the state variables. This is simply a result of earlier research priorities toward minimizing the size of the dynamic system equations for ease of simulation or automated equation generation. Although programs utilizing these approaches can be designed to solve for internal reactions and bearing or constraint forces, they do so with great complexity because they either require solving additional differential or additional algebraic or pairs of coupled differential-algebraic equations that arise because of the additional kinematic constraints required to define the reaction force or they need to apply a residual kineto-static method. These programs and their underlying methods have not been designed to automatically produce the type of equations required for use in controlling reaction forces within intelligent machines.

Of initial concern to any modelling approach is the number of generalized coordinates desired within the dynamic system model. The minimum number is controlled by the degrees of freedom of the system. The maximum practical number is determined by multiplying the number of bodies in the system less one (if a body is held fixed) by either 6 if spatial or 3 if planar. Between these two extremes is a wide range. The specific selection should be made depending upon both simulation and control issues. As such, the model should be reconfigurable by the robust controller. It would use the minimum number of equations to predict the states as well as the internal reaction forces. This minimum number would change depending on the task. Details on how this is to be accomplished both theoretically and in a practical sense is an issue of the research.

COMMON APPROACHES TO DYNAMIC SYSTEM MODELLING

Dynamic system modelling equations can be classified within two major groupings, Vectorial and Analytical Dynamics. Within the vectorial grouping are Newton-Euler Methods, d'Alemberts Principle and, to a point, Virtual Work. Associated with analytical dynamics is LaGrange's form of d'Alemberts Principle, Kane's Equations and utilization of Pseudo Coordinates, LaGrange's Equations and Hamilton's Canonical Equations. Both major groupings may be applied to open and closed chain systems although their methods of approach differ. Variations within the groupings are associated with the use or non-use of LaGrange multipliers, kinematic influence coefficients or Jacobians. Part II of this report uses a "cross-cultural" viewpoint to present and discuss some of the key facets of commonality and differences between these methods. For the example problem given here of a pendulum, the results are so similar that details between the differences are not important. This example was chosen because it is atypical.

Pendulum-A Simple Open Chain

The position and orientation of this single link planar system, whose details are shown in Fig. 1., requires up to three generalized coordinates to describe. The connection between the moving pendulum and the ground or fixed body constrains two of these motions and thus the pendulum has only one degree of freedom.

Simple Pendulum

$$\begin{aligned} \text{number of moving bodies} &= n = 2-1=1 \\ \text{maximum number of generalized coordinates} &= 3*1 = 3 \\ \text{number of joint constraints} &= r = 2*1=2 \\ \text{degrees of freedom} &= m = 3-2 = 1 \end{aligned} \qquad (1)$$

Using standard modelling procedures, a dynamic model of this system may require three differential equations coupled to two algebraic equations for a system of five equations. These equations can be reduced automatically through various procedures discussed in Part II. In that case only one equation would result. Typically the reduction methods must be done numerically and are prone to singularities and convergence problems.

Absolute and Relative Coordinates

The pendulum's location relative to a fixed frame may be located using either absolute or relative coordinates. In an absolute system the position **R** and orientation θ of the pendulum is located relative to the inertial fixed frame. For the pendulum the cartesian absolute coordinates, x, y, are used whereas the angular orientation is defined by the absolute angular coordinates α. The combination of position and orientation is the pose of the pendulum, ν.

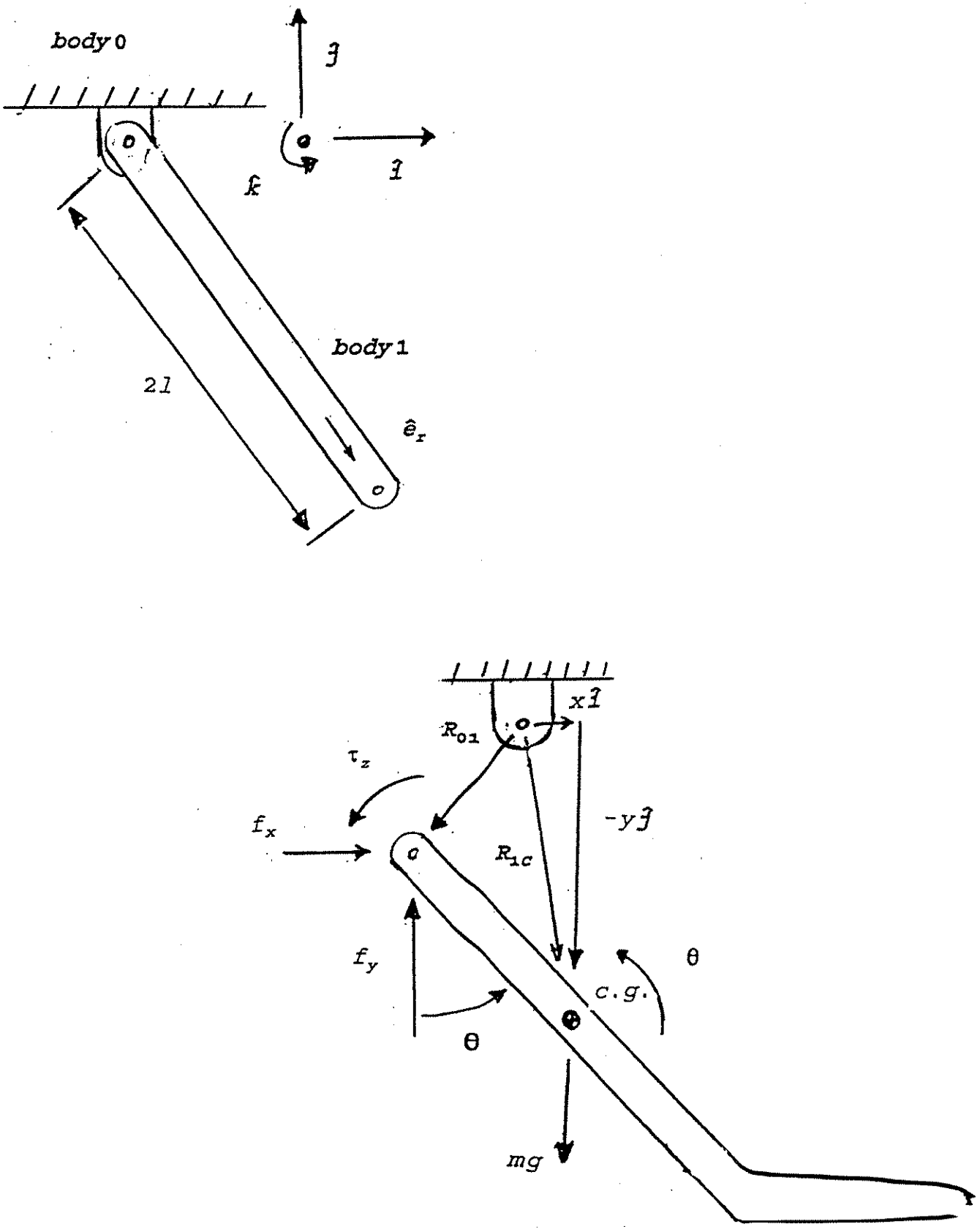


Fig. 1. Pendulum Example

$$\mathbf{v} = \begin{Bmatrix} x \\ y \\ \theta \end{Bmatrix} \tag{2}$$

When using relative coordinates, on the other hand, the position and orientation of a body is related to the position and orientation of its interconnected bodies. Obviously six relative coordinates would thus be required to interrelate the pose of two spatial bodies. When referring to relative coordinates the symbol **q** is typically used. For the pendulum, a degenerative case, a single relative coordinate is required and the relative coordinate is of the same value as the absolute orientation coordinate; the symbol θ is used to represent both.

Joint and Loop Constraints

Typically the relative motion of two bodies is constrained by the type of interconnection. Without loss of generality we assume that each connection between two spatial bodies has one degree of joint freedom and constrains the relative motion of the bodies in five ways. In the planar case two constraints would occur. Typically constraints between links in a kinematic chain are defined in one of two ways, joint or loop type. Joint constraints are usually written in absolute coordinates and relate how the motion of points in one body are restricted by the motion of similar points in the second body due to the particular type of joint. Loop constraints are of two forms, complete loops and partial loops. Partial loop constraints are used to relate a link's absolute coordinates to its relative coordinates. Complete loop equations are usually written exclusively in terms of relative coordinates and relate the requirements on these coordinates for loop closure. For the pendulum a requirement will be the use of a partial loop constraint of the form.

$$\phi = \phi\ (x,\ y,\ \theta) = 0 \tag{3}$$

Most modelling procedures require that the system equations are written in Pfaffin form so that linear algebraic procedures can be utilized. The forms are obtained through the use of first time-derivatives of the constraints. Numerical solution methods used in the simulations also tend to require second derivatives. From Eq. (3) we find

$$\begin{aligned} C(\ q,\ t)\ \dot{q}\ + A(\ q,\ t) &= 0 \\ C\ \ddot{q}\ + \left(\ \dot{C}\dot{q}\ + A\ \right) &= 0 \\ C\ \ddot{q}\ + B &= 0 \end{aligned} \tag{4}$$

For the pendulum there are r=2 constraints and C is of dimension 2 x 3. Of the 3 absolute coordinates, only m=3n-r=3-2=1 is independent.

Applied to the pendulum these become

$$R_{01} = 0 = R_{1c} - l\,\hat{e}_r$$

$$\begin{aligned} x - l\sin\theta &= 0 \\ y = l\cos\theta &= 0 \end{aligned}$$

$$\begin{aligned} \dot{x} - l\,\dot{\theta}\cos\theta &= 0 \\ \dot{y} - l\,\dot{\theta}\sin\theta &= 0 \end{aligned} \tag{5}$$

$$\begin{bmatrix} 1 & 0 & -l\cos\theta \\ 0 & 1 & -l\sin\theta \end{bmatrix} \begin{Bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{Bmatrix} = C\,\dot{\mathbf{v}} = \mathbf{0}$$

$$\begin{bmatrix} 1 & 0 & -l\cos\theta \\ 0 & 1 & -l\sin\theta \end{bmatrix} \begin{Bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{\theta} \end{Bmatrix} + \begin{Bmatrix} l\,\dot{\theta}^2\sin\theta \\ -l\,\dot{\theta}^2\cos\theta \end{Bmatrix} = C\,\ddot{q} + B = \mathbf{0}$$

Newton-Euler, LaGrange Procedures

Through the use of LaGrange multipliers it can be shown that Newton-Euler and LaGrange procedures can lead to essentially the same form of solution. LaGrange multipliers contain trigonometric combinations of constraint forces which can be determined through the use of LaGrange's form of d'Alembert's Principle. Constraint forces appear in the equations as a direct consequence of the choice of generalized coordinates. If one chooses the absolute coordinates as the generalized coordinates and there are numerous interconnections between the links, then numerous constraint equations are required and numerous constraint forces will be determined. The appropriate modelling equation becomes:

$$\begin{aligned} M\,\ddot{\mathbf{v}} + \dot{\mathbf{v}}\,D\,\dot{\mathbf{v}} &= Q^{con} + Q^{ext} \\ C\,\ddot{\mathbf{v}} + B &= 0 \end{aligned} \tag{6}$$

where **M** is a mass matrix and **D** is associated with centripetal and coriolis forces as well as gyroscopic effects. The forcing functions, **Q**, are associated with the constraints, actuators, and body forces. In the case of the constraint forces,

$$Q^{con} = C^T\,\lambda = \frac{\partial \dot{\mathbf{v}}}{\partial \dot{q}} \cdot e^{con} \tag{7}$$

For the pendulum, the generalized constraint forces in terms of LaGrange multipliers become:

$$Q^{con} = C^T \lambda = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -l\cos\theta & -l\sin\theta \end{bmatrix} \begin{Bmatrix} \lambda_1 \\ \lambda_2 \end{Bmatrix} = \begin{Bmatrix} \lambda_1 \\ \lambda_2 \\ -l\cos\theta\ \lambda_1 - l\sin\theta\ \lambda_2 \end{Bmatrix} \tag{8}$$

The second part of Eq. (6) leads to

$$Q^{con} = \begin{bmatrix} \frac{\partial \dot{R}_{1c}}{\partial \dot{x}} & \frac{\partial \omega_{1c}}{\partial \dot{x}} \\ \frac{\partial \dot{R}_{1c}}{\partial \dot{y}} & \frac{\partial \omega_{1c}}{\partial \dot{y}} \\ \frac{\partial \dot{R}_{1c}}{\partial \dot{\theta}} & \frac{\partial \omega_{1c}}{\partial \dot{\theta}} \end{bmatrix} \cdot \begin{Bmatrix} f_{11} \\ \tau_{11} \end{Bmatrix} = \begin{Bmatrix} f_x \\ f_y \\ -f_x\, l\cos\theta - f_y\, l\sin\theta \end{Bmatrix} \tag{9}$$

Equating Eq. (8) and Eq. (9) yields a simple relationship between constraint forces and LaGrange multipliers not characteristic of multibody systems.

$$\begin{aligned} f_x &= \lambda_1 \\ f_y &= \lambda_2 \end{aligned} \tag{10}$$

The equations of motion, Eq. (6), become:

$$\begin{bmatrix} m & 0 & 0 & -1 & 0 \\ 0 & m & 0 & 0 & -1 \\ 0 & 0 & \frac{1}{3}ml^2 & l\cos\theta & l\sin\theta \\ -1 & 0 & l\cos\theta & 0 & 0 \\ 0 & -1 & l\sin\theta & 0 & 0 \end{bmatrix} \begin{Bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{\theta} \\ f_x \\ f_y \end{Bmatrix} = \begin{Bmatrix} 0 \\ -mg \\ \tau_a \\ l\dot{\theta}^2\sin\theta \\ -l\dot{\theta}^2\cos\theta \end{Bmatrix} \tag{11}$$

Although the constraint forces are available through simulation they are not explicit functions of the state variables and their first derivatives.

If one were to use coordinate partitioning or some other scheme discussed in Part II the equations of motion could be rewritten in other forms including the most common:

$$\frac{4}{3}ml^2\,\ddot{\theta} + lmg\,\sin\theta = \tau \tag{12}$$

A kineto-static analysis supplies the required solutions for the constraint forces.

$$\begin{Bmatrix} f_x \\ f_y \end{Bmatrix} = \begin{Bmatrix} m\,\ddot{x} \\ m\,\ddot{y} + mg \end{Bmatrix} = \begin{Bmatrix} ml\ddot{\theta}\,\cos\theta - ml\dot{\theta}^2\,\sin\theta \\ ml\ddot{\theta}\,\sin\theta - ml\dot{\theta}^2\,\cos\theta + mg \end{Bmatrix} \tag{13}$$

Again the constraint forces are not written as functions of the state variables alone.

As indicated in the above two analyses, the number of generalized coordinates utilized is up to the problem analyst. In the first analysis three coordinates were used whereas in the second only one was used. If information on the reaction force is not needed, the second approach is preferred because it is of smaller dimension. But when information on the reaction force is required the choice, in general, is somewhat blurred. The relationship between LaGrange multipliers and constraint forces is not as clear as the example would otherwise indicate and the means of obtaining the reduced set of equations is typically subject to instabilities associated with singularities and stiff-system problems. For now it is simply stated that:

The explicit use of a geometric constraint equation as an algebraic augmentation to the system equations will result in

- a dynamic system model of high order and
- a generalized force which incorporates a force normal to the constraint (surface).

Conversely, the elimination of the geometric constraints through a reduction of generalized coordinates within the system dynamical equations will result in

- a reduced dimension of the dynamical problem to be solved but
- a need to solve an auxiliary or kineto-static problem in the constraints.

What is desired and believed to be at hand is an analysis technique which both

- reduces the dimension of the dynamical problem to be solved, yet
- includes a generalized force which incorporates and easily defines the constraint forces, and
- is reconfigurable depending on the task requirements.

e of Least Constraint

o Newton-Euler methods of vectorial mechanics and me[redacted]ods based on tions to Analytical Mechanics are approaches based on Gau[redacted]'s Principle of [1]. Gauss's principle states that the constrained motion ac[redacted]elerations of a hat the Gaussian function, Γ, is minimized over all solutions wh[redacted]ch satisfy these function is written in terms of the system mass matrix and dif[redacted]rences between nd unconstrained accelerations of the system elements.

$$\Gamma = [\ddot{q} - a(q, \dot{q}, t)]^T M [\ddot{q} - a(q, \dot{q}, t)] \tag{14}$$

ı by Udwadia [2,3] through the use of the Moore-Penrose ge[redacted]ralized inverse ıt forces can be written as explicit functions of the state va[redacted]ables and their solution method requires that a matrix **K** involving a pseudo[redacted]nverse be first

$$K(q, \dot{q}, t) = M^{\frac{1}{2}} \left(C M^{-\frac{1}{2}} \right)^{+} \tag{15}$$

s the pseudo-inverse. Details on pseudo-inverses is given [redacted] Part II. The then become

$$Q_c (q, \dot{q}, t) = - K (B + C M^{-1} Q) \tag{16}$$

these equations belies the fact that in some cases that the pse[redacted]lo-inverse may ain, especially if a symbolic representation is desired.

$$M^{-\frac{1}{2}} = \begin{bmatrix} 1 & 0 & -l\cos\theta \\ 0 & 1 & -l\sin\theta \end{bmatrix} \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & \frac{1}{3}ml^2 \end{bmatrix}^{-\frac{1}{2}} = \begin{bmatrix} \frac{1}{\sqrt{m}} & 0 & -\sqrt{\quad\quad} \cos\theta \\ 0 & \frac{1}{\sqrt{m}} & -\sqrt{\quad\quad} \sin\theta \end{bmatrix} \tag{17}$$

The right pseudo-inverse is non-singular and fairly easy to obtain, that is

$$TMP^{+R} = TMP^T \left[TMP\ TMP^T\right]^{-1} = \frac{\sqrt{m}}{4}\begin{bmatrix} 1+3\sin^2\theta & -3\sin\theta\cos\theta \\ -3\sin\theta\cos\theta & 1+3\sin^2\theta \\ -\sqrt{3}\cos\theta & -\sqrt{3}\sin\theta \end{bmatrix} \tag{18}$$

We then determine **K**

$$K = M^{-1}\left[C\ M^{-\frac{1}{2}}\right]^T = \frac{m}{4}\begin{bmatrix} 1+3\sin^2\theta & -3\sin\theta\cos\theta \\ -3\sin\theta\cos\theta & 1+3\sin^2\theta \\ -l\cos\theta & -l\sin\theta \end{bmatrix} \tag{19}$$

and the constraint forces.

$$Q^{con} = -K(B+CM^{-1}Q) = \begin{Bmatrix} -ml\dot{\theta}^2 + \frac{3\tau}{4l}\cos\theta - \frac{3}{4}mg\cos\theta\sin\theta \\ ml\dot{\theta}^2 + \frac{3\tau}{4l}\sin\theta + \frac{mg}{4}(1+3\cos^2\theta) \\ -\frac{3}{4}\tau - \frac{mgl}{4}\sin\theta \end{Bmatrix} \tag{20}$$

Finally, the system equations become

$$\begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & \frac{1}{3}ml^2 \end{bmatrix}\begin{Bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{\theta} \end{Bmatrix} = \begin{Bmatrix} -ml\dot{\theta}^2 + \frac{3\tau}{4l}\cos\theta - \frac{3}{4}mg\cos\theta\sin\theta \\ ml\dot{\theta}^2 + \frac{3\tau}{4l}\sin\theta - \frac{3}{4}mg\sin^2\theta \\ \frac{\tau}{4} - \frac{mgl}{4}\sin\theta \end{Bmatrix} \tag{21}$$

Comparison of Methods

Comparisons of the resulting equations using standard approaches, Eq. (11) or Eq. (12) and (13), and those using Gauss's Constraint Minimization, Eq. (21), can now be made. Both methods give similar angular motion equations yet yield quite different translation equations. Gauss's method describes both the x and y bearing forces as functions of the state variable and its first derivative whereas the Newton-Euler or LaGrange method requires acceleration information.

In more complex linkages a similar situation occurs; Gauss's method determines the constraint and reaction forces as functions of the state variables and their first derivative whereas the Newton-Euler expressions for constraint and bearing forces come in two versions and their combinations. In one version there are numerous equations and LaGrange multipliers required in the simulation. The multipliers are related to the bearing forces through geometry. In the second version the system equations are written in reduced dimension and the bearing forces are made functions of the state variables as well as body accelerations by means of a kineto-static analysis. The reduced set is somewhat compromised in that it typically must be obtained numerically and is subject to stability and singularity problems.

One of the more exciting possibilities of Gauss's method is the potential recasting of model driven force control algorithms. This now provides a method to obtain explicit force models in terms of the state variables. That is, if one wishes to maintain the vertical or x bearing force on the pendulum to be a particular value or function and also knew the current angular position and velocity, one would explicitly know the required torque to be applied. This approach could be developed, using reference model control concepts, to be very robust even when applied to highly nonlinear and coupled systems.

SUMMARY AND CONCLUSIONS

It has been shown elsewhere that many desired machine operations such as metal removal by lathes, mills or grinders and assembly operations perform better when their (internal) constraint forces are controlled. The more robust nonlinear controllers tend to apply varied forms of model-driven controllers which essentially perform an adaptive exact linearization through feedforward action. In such cases the pattern is clear: define the desired value of a variable to be controlled, measure or determine through a model the current value and control the difference. The robustness of the schemes depends on the accuracy and speed of predicting the variable behavior by means of an empirical, analytical or neural network technique.

Dynamic system modelling most typically utilizes various forms of Analytical or Newtonian dynamics to generate system equations. Often the number of generalized coordinates utilized is more than the actual number of independent degrees of freedom and thus requires a number of constraint equations equal to this difference. The constraint equations are usually appended to the dynamical equations through LaGrange multipliers to form system equations. The multipliers are equal to generalized forces composed of various combinations of the physical constraint forces. Upon proper selection of generalized coordinates, implicit solutions for the constraint forces as functions of time can be determined at a cost of numerical complexity.

It is believed that control strategies utilizing Gauss's Principle of Least Constraint to obtain constraint force representations in terms of state variables should and can now be developed. Near term applications of this particular modelling and control structure to intelligent machinery is anticipated not only for Stewart Platforms but also in active suspensions and automated assembly. In the long term, benefits in active prosthetics and intelligent mechanical devices should be anticipated.

REFERENCES

1. Hofrath & Gauss, C. F., "Uber ein Neues Allgemeines Grundgestz der Mechanik", ***J. Reine Angewandte Mathematik***, Vol 4, pp 232-235, 1829.

2. Udwadia, F. E. and Kalaba, R. E., "A new Perspective on Constrained Motion.", ***Proc. R. Soc. London*** Series A, 1992, pp 407-410.

3. Udwadia, F. E. and Kalaba, R. E., "Lagrangian Mechanics, Gauss's Principle, Quadratic Programming and Generalized Inverses: New Equations of Motion of Non-Holonomically Constrained Discrete Mechanical Systems.", ***Q. Appl. Math***. (in press)

PART II — TECHNICAL DETAILS

ABSTRACT

A technical overview of the dynamic modelling procedures applicable to simulation and control of a Stewart Platform is given. The goal of the overview is to identify a scheme which will directly associate state variables that describe platform motions with internal constraint forces. Such constraint forces occur, for example, during installation of an ATLSS connection. The association of state variables to constraint forces is a requirement for robust intelligent controllers; at present no such controllers exist. The overview looks at the techniques associated with DADS and ADAMS, the most well known simulators, as well as at other software such as TREETOPS.

Modelling procedures studied include:

- Newton-Euler
- LaGrange's Equations with and without LaGrange Multipliers
- LaGrange's Form of d'Alembert's Principle
- Kane's Quasi Coordinates
- Constraint Elimination through Singular Value Decomposition
- Gauss's Principle of Least Constraint.

Similarities and differences between the approaches are discussed as is application for both open and closed kinematic chains.

The techniques are applied to a pendulum for comparison.

INTRODUCTION

Intelligent Machines and Structures are systems composed of rigid or deformable elements interconnected by actuators under intelligent control. The robust controllers utilized have the ability to incorporate information about the environment- and situation-dependent performance goals to modify their own structure in an intelligent manner. Although the generic research and application area of such techniques is very broad, this report addresses the more limited area of dynamic system modelling techniques currently used to simulate open and closed kinematic chains. **The specific application of interest is the installation of ATLSS Connections by a Stewart Platform crane.**

There are three reasons for limiting the study.

1. It has been found that the more robust non-linear controllers require reference models of the system about which to control the actual system motion. In such cases stability and convergence can be proven.

2. Performance goals of machines and structures vary with their use and with time. For example, a robot may at one time be used to move an item from one location in (free) space to another, and position and velocity control of the robot endpoint is important. Whereas, if the robot is performing an insertion task, force and possibly a hybrid force-position control is called for. Similar examples occur in manufacturing operations such as the use of machine tools, in car suspensions, in simple linkages such as cam-followers and in buildings prone to earthquake loading. In the three latter areas techniques such as active vibration control are currently a non-robust version of such systems.

3. It has been found that for most systems the appropriate model does not exist. If one is controlling position, an explicit linkage position model is most beneficial and is available, whereas if one is controlling force, a constraint force model with the explicit motion dependencies enumerated would be best. Consider the case of an active car suspension. One of its purposes is to provide a smooth ride to the passengers. Currently the controllers, due to lack of the appropriate constraint-force models, do this in an indirect way. When the car goes over a bump, what is measured and what is controlled is the height (or acceleration) of the passenger above the average roadbed. What could be measured and controlled is the vertical force the passenger feels or would feel given various suspension linkage motions. At present there is no modelling scheme that efficiently makes such predictions. However, it is the (internal constraint) forces felt by the ATLSS Connection as it is installed that are important.

The goal, therefore, is to find dynamic system modelling procedures which produce families of selectable models that can be used within a re-configurable non-linear control paradigm. The explicit goal of the current effort is to evaluate, compare and improve upon the numerous approaches for generating dynamic system models for both open and closed kinematic chains with particular emphasis being placed upon methods for obtaining constraint-force models and to facilitate their control. Details on the controllers and the associated sensor fusion, instrumentation and implementation are not part of the report, however.

MECHANICAL LINKAGE SYSTEMS

Open and Closed Chain Linkages, Manipulators and Mechanisms

Mechanical linkage systems are composed of interconnected rigid or deformable bodies. The connections may be "point/line" or higher pair connections or "surface" or lower pair connections. The latter includes prismatic, revolute, spherical, flat, helical, or cylindrical connections while the former includes connections commonly found on gears and belts where deformation within the connection is of great extent. The limitation at this time is to lower pairs. The bodies may be interconnected to form open or closed kinematic chains. A closed chain or mechanism is one which can be traversed back to the starting point by moving through the connections and elements without repeat. An open chain is a manipulator which at times is allowed to have a closed chain arrangement. Linkages of both open and closed chain types, and both planar and spatial versions of such chains will be considered here. A spatial Stewart Platform is shown in Figure 1, whereas a planar version of the platform is shown in Figure 2. A portion of the spatial platform, shown in Figure 3, is but an open chain as are portions of the planar platform shown in Figure 4.

These linkages are examined during the remainder of this report. Of concern is the number of generalized coordinates desired within the dynamic system model. The minimum number is controlled by the degree of freedom of the system. The maximum practical number is determined by multiplying the number of bodies in the system less one by either 6 if spatial or 3 if planar. Between these two extremes is a wide range. The selection of the number of generalized coordinates should be made depending upon both simulation and control issues. As such it could be reconfigurable by the robust controller of an intelligent machine. Details on how this is to be accomplished both theoretically and in a practical sense is an issue of the research.

The discussions here are arbitrarily limited to rigid body systems. This is so that some additional terms in the dynamic system equations do not have to be shown. Generally speaking the concepts reviewed apply equally well to deformable systems upon inclusion of the additional terms.

Example Linkages

The **Stewart Platform** [1], Figure 1, is the basic linkage that will be described in order to clarify concepts about numbers of bodies, constraints and degrees of freedom. This platform is composed of an upper and lower platform and six each of upper and lower "pods" for a total of 14 bodies. If the linkage of bodies were not formed, then each body would have 6 spatial degrees of freedom, each definable by a generalized coordinate, for a total of 6*14 or 84 degrees of freedom.

In actuality, a **spatial Stewart Platform** has 6 degrees of freedom; the difference being constraints afforded by the joints and a loss of 6 degrees of freedom upon choosing one of the bodies as a base or ground. The pods are connected to the platforms using either spherical or Hookean (perpendicular revolute mutually perpendicular to the pod axis) joints. With spherical joints, the three relative displacements between the platform and pod are lost and, with the Hookean joint, there is an additional loss of angular rotation about the pod axis. Arbitrarily, the upper platform/pod connections can be selected as spherical joints and the lower platform/pod connections as Hookean joints. The lower to upper pod connections are prismatic joints, each constraining the relative motion of the connected elements in 5 ways. Thus, the resulting system has six degrees of freedom.

Spatial Stewart Platform

$$\begin{gathered}\text{number of moving bodies} = n = 14-1=13 \\ \text{maximum number of generalized coordinates} = 6*n = 78 \\ \text{number of joint constraints} = r = 3*6+4*6+5*6=72 \\ \text{degrees of freedom} = m = 78-72 = 6\end{gathered} \tag{1}$$

The dynamic system model of the platform may thus vary from (a) one which contains 78 differential equations coupled (through LaGrange Multipliers) to 72 algebraic equations where each of the constraint forces is easily determinable through a kineto-static analysis to (b) an alternative model with 6 differential equations and a follow-on kineto-static analysis requiring the solution of 72 coupled algebraic equations. The relative performance of methods at either extreme or in-between is unknown. A goal here is to determine the relative performance.

With a **planar Stewart Platform**, Figure 2, each element requires three as opposed to six generalized coordinates and each of the six revolute as well as the three prismatic pairs requires two constraints. Thus:

Planar Stewart Platform

$$\begin{gathered}\text{number of moving bodies} = n = 8-1=7 \\ \text{maximum number of generalized coordinates} = 3*n = 21 \\ \text{number of joint constraints} = r = 2*6+2*3=18 \\ \text{degrees of freedom} = m = 21-18 = 3\end{gathered} \tag{2}$$

Thus one can choose from models with 21 differential equations coupled to 18 algebraic equations to models with a system of but three differential equations.

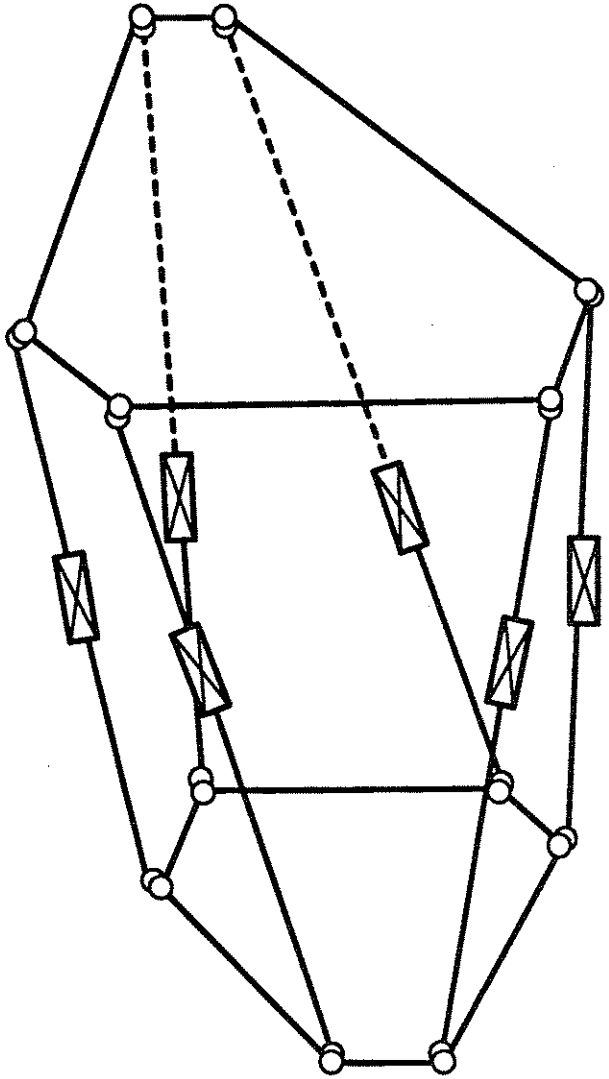


Figure 1 Spatial Stewart Platform

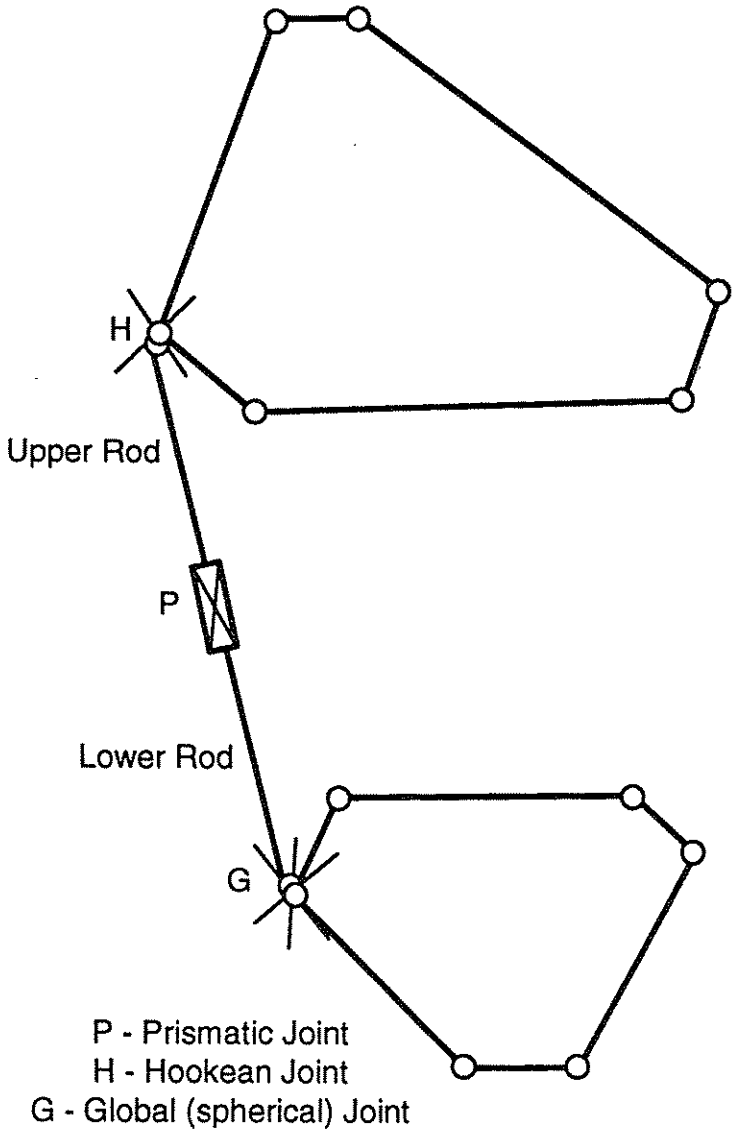


Figure 3 Branch of Spatial Stewart Platform

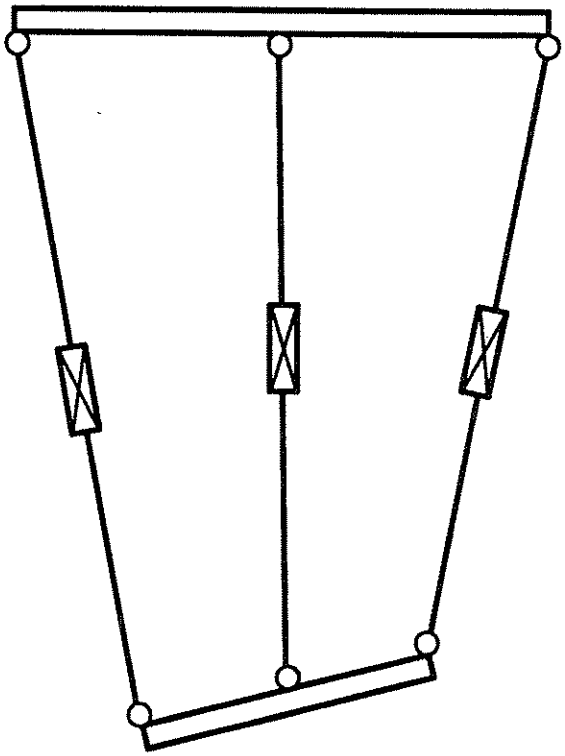


Figure 2 Planar Stewart Platform

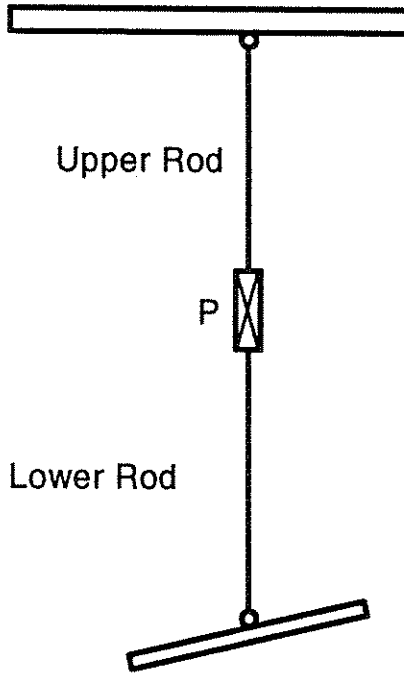


Figure 4 Branch of Planar Stewart Platform

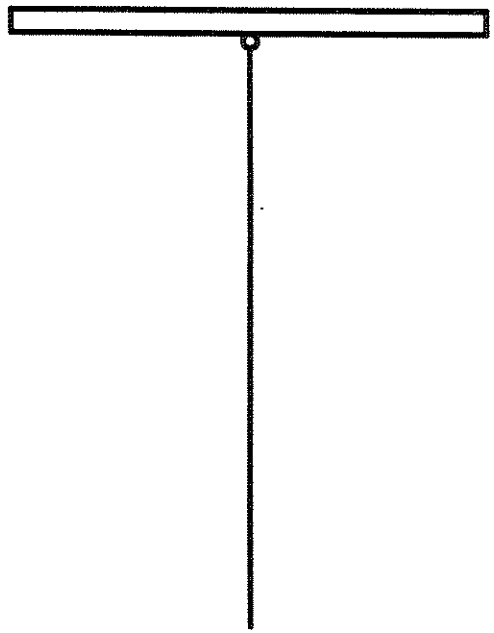


Figure 5 Simple Pendulum

For the open kinematic loop case shown in Figure 3, which is a **branch of a spatial Stewart Platform**, a similar analysis is utilized. The spatial linkage consists of four bodies, a two degree of freedom Hookean joint, a one degree of freedom prismatic joint and a three degree of freedom spherical joint. It has the potential to control all six degrees of freedom of upper platform motion if actuation of all joint axes is accomplished. Thus:

Branch of Spatial Stewart Platform

$$\begin{gathered}\text{number of moving bodies} = n = 4-1=3 \\ \text{maximum number of generalized coordinates} = 6*3 = 18 \\ \text{number of joint constraints} = r = 3*1+4*1+5*1=12 \\ \text{degrees of freedom} = m = 18-12 = 6\end{gathered} \tag{3}$$

The choice of solution method varies from models with 18 differential equations coupled to 12 algebraic equations to a system model with 6 differential equations.

For the planar open chain case in Figure 4, which is a **branch of a planar Stewart Platform**, there is the further simplification:

Branch of Planar Stewart Platform

$$\begin{gathered}\text{number of moving bodies} = n = 4-1=3 \\ \text{maximum number of generalized coordinates} = 3*3 = 9 \\ \text{number of joint constraints} = r = 2*2+2*1=6 \\ \text{degrees of freedom} = m = 9-6 = 3\end{gathered} \tag{4}$$

Thus, there may be dynamic system models with 3 to 9 differential equations and up to 6 constraints.

The somewhat trivial case consisting only of the upper Platform, one Hookean joint, and an upper rod reduces to a simple pendulum, Figure 5.

Simple Pendulum

$$\begin{gathered}\text{number of moving bodies} = n = 2-1=1 \\ \text{maximum number of generalized coordinates} = 3*1 = 3 \\ \text{number of joint constraints} = r = 2*1=2 \\ \text{degrees of freedom} = m = 3-2 = 1\end{gathered} \tag{5}$$

DYNAMIC SYSTEM MODELLING TECHNIQUES

Computer Models

Current methods for dynamic system modelling of both open and closed mechanical linkage systems are ill tuned to determining constraint forces as explicit functions of the state variables. This is simply a result of earlier research priorities toward minimizing the size of the dynamic system equations for ease of simulation or automated equation generation as found in programs such as DADS [2] (Haug/Iowa) and ADAMS [3] (Chace/Michigan) and the more dated IMP [4] (Uicker/Wisconsin) and DRAM/DAMN [5,6] (Chace/Michigan), as well as MEDUSA, VECNET, DYMAC [7] and others, which are programs that were originally designed for rigid links in closed chains. This situation also extends to programs associated with flexible links in open chain configurations such as TREETOPS [8].

Although these programs can be used to solve for internal reaction, bearing or constraint forces, they do so with great complexity. They either require solving an additional or an additional pair of coupled differential-algebraic equations that arise because of the additional kinematic constraint required to define the reaction force or solving for the constraint forces as an aftereffect through a residual kineto-static method. These programs and their underlying methods have not been designed to automatically produce the type of equations required for controlling intelligent machines.

Classification

Dynamic system modelling equations can be classified within two major groupings, vectorial dynamics and analytical dynamics. Within the vectorial grouping are Newton-Euler Methods and d'Alemberts Principle. In the discussion below Virtual Work is also included. Associated with analytical dynamics is LaGrange's form of d'Alemberts Principle (also known as Kane's Equations--particularly when utilizing pseudo coordinates), LaGrange's Equations and Hamilton's Canonical Equations. Both major groupings may be applied to open and closed chain systems although their methods of approach differ. Variations within and between the groupings are associated with the degree of use of LaGrange multipliers, kinematic influence coefficients or Jacobians. In the following a "cross-cultural" viewpoint of some of the key facets of commonality and difference are reviewed to show their limitations in constraint force determination and control. More complete reviews for closed chains are given in the literature by Paul [9] and open chains by Likins [10] and by Hemami [11]. The use of Hamilton's canonical equations is not discussed here except to point out that details of their use are given in [10] and that the method was utilized in the program IMP [4].

Absolute and Relative Coordinates

The ith body within a kinematic chain of n bodies may be located using either an absolute or a relative coordinate system. In an absolute system the position, $\mathbf{R}_i$, and orientation, θ_i of the body are located relative to a fixed inertial frame. The position may be defined by a set of

cartesian absolute coordinates, x_i, y_i, z_i, whereas the angular orientation is defined by a set of absolute angular coordinates, α_i, β_i, and γ_i. The combination of position and orientation is the pose of the body, ν_i. Defining the pose of n bodies in a kinematic chain requires 6 n absolute coordinates.

$$\begin{aligned} \mathbf{v}_i &= \left[\boldsymbol{R}_i \, , \, \boldsymbol{\Theta}_i \right]^T \quad i = 1, \cdots, n \\ \boldsymbol{R}_i &= \boldsymbol{R}_i \, (x_i \, , \, y_i \, , \, z_i) \\ \boldsymbol{\Theta}_i &= \boldsymbol{\Theta}_i \, (\alpha_i \, , \, \beta_i \, , \, \gamma_i) \\ \mathbf{v} &= \left[\mathbf{v}_1^T \, , \, \cdots, \, \mathbf{v}_n^T \right]^T \end{aligned} \tag{6}$$

When using relative coordinates, on the other hand, the position and orientation of a body are related to the position and orientation of a second body. Six relative coordinates are required to interrelate the pose of two spatial bodies. Typically the relative motion of two bodies is constrained by the type of interconnection. Without loss of generality one can assume that each connection between two bodies has one degree of joint freedom and constrains the relative motion in five ways. Although a spherical joint, for example, constrains the three relative translations between the two bodies and does not constrain any of the three relative orientations, an alternative view is to decompose the joint into three co-located and revolute joints with one relative coordinate that is allowed to vary and five which remain fixed. When referring to relative coordinates, q_j, it is the variable coordinate which is being referred. In a kinematic chain composed of r connections each with one degree of freedom, there is therefore:

$$\begin{aligned} q_j \; ; \quad j &= 1, \cdots, r \\ \boldsymbol{q} &= \left[q_1 \, , \, \cdots, \, q_r \right]^T \end{aligned} \tag{7}$$

For a kinematic chain with r joints, r relative coordinates are required. In closed kinematic chains, or when an open chain is in a closed configuration, the number of degrees of freedom of the linkage will be less than the number of joints. In this case it is theoretically possible to partition the relative or joint variables into two sets, independent and dependent.

$$\begin{aligned} \boldsymbol{q} &= \left[q_1 \, , \, \cdots \, , \, q_m \, , \, q_{m+1} \, , \cdots \, , \, q_r \right]^T \\ \boldsymbol{q} &= \left[\boldsymbol{q}^{iT} \, , \, \boldsymbol{q}^{dT} \right]^T \end{aligned} \tag{8}$$

In what follows a notational condensation is used. When discussing absolute coordinates, ν will always be used, and when discussing relative coordinates, q will be used. In some cases, q will also be used to mean generalized coordinates which could be either relative or absolute, but the usage will be clear in the discussion.

Joint and Loop Constraints

Constraints between links in a kinematic chain are defined as either joint or loop constraints. Joint constraints typically are written in absolute coordinates and relate how the motion of points in one body are restricted by the motion of similar points in the second body due to the particular type of joint. Thus, if bodies i and j are interconnected, they are constrained by:

$$\boldsymbol{\Phi}_j = \phi_j \left(\boldsymbol{\nu}_i , \boldsymbol{\nu}_k , t \right) ; \quad j = 1 , \cdots , r \tag{9}$$

Loop constraints are of two forms, complete loops and partial loops. Complete loop equations are typically written exclusively in terms of relative coordinates.

$$\phi_l \left(q , t \right) = 0 \tag{10}$$

There may be up to 6 scalar equations for each complete kinematic loop, but there are only r independent complete loop constraints.

Partial loop constraints are used to relate a particular link's absolute coordinates to the relative coordinates of the linkage chain, q. These constraints can be written as.

$$\boldsymbol{\Phi}_i = \phi_i \left(\boldsymbol{\nu}_i , q , t \right) = 0 ; \quad i = 1, \cdots, n \tag{11}$$

One finds in most cases that the solutions to system dynamic equations requires simultaneous solution of the constraint equations, and most typically of first and second-time derivatives of these equations. To obtain such expressions, first write Eqs. (9)-(11) in terms of generalized coordinates and then take the first time derivative

$$\begin{gathered} \boldsymbol{\Phi} \left(q , t \right) = \left[\phi_1 , \cdots , \phi_r \right]^T = 0 \\ \boldsymbol{\Phi}_q \dot{q} + \boldsymbol{\Phi}_t = 0 \end{gathered} \tag{12}$$

If one assumes that the constraints are of Pfaffin form, that is they include all holonomic constraints and all non-holonomic constraints that can be put in simple form, then Eq. (12) and its time derivative can be easily written in matrix form.

$$C(q, t)\,\dot{q} + A(q, t) = 0$$

$$C\,\ddot{q} + \left(\dot{C}\dot{q} + A \right) = 0 \qquad (13)$$

$$C\,\ddot{q} + B = 0$$

Because there are r constraints, C is of dimension r x n with $r \leq n$. Of the n q's only m=n-r are independent.

One benefit of linear forms such as Eq. (13) is that they can be used to determine the dependent variables in terms of the independent set. To find the dependent velocities or accelerations in terms of the independent velocities or accelerations, partition the appropriate constraint equation, Eq. (13) and solve for the independent set.

$$C\,\dot{q} + A = 0$$

$$\left[C^i \vdots C^d \right] \begin{bmatrix} \dot{q}^i \\ \cdots \\ \dot{q}^d \end{bmatrix} + A = 0 \qquad (14)$$

$$\dot{q}^d = \left[C^d \right]^{-1} \left(A - C^d \dot{q}^i \right)$$

The last of these equations is highly prone to singularities and is thus the root to the numerous problems discussed later. To obtain solutions for the generalized coordinates consistent with the nonlinear constraint equations, the derivative forms of the constraints given above are utilized within iterative numerical procedures. Small errors in q^i are associated with small values of $\dot{q}^i$ but singularities in C^d cause infinite values in its inverse and thus large errors in estimates of q^d. Thus, computer programs that utilize such expressions must overcome this situation.

Other explicit forms for body accelerations due to joint constraints, Eq. (9), or partial loop constraints, Eq. (11), result in, respectively:

$$\ddot{\mathbf{v}}_i = \ddot{\mathbf{v}}_{ik} \left(\mathbf{v}_i , \mathbf{v}_k , \dot{\mathbf{v}}_i , \dot{\mathbf{v}}_k , \ddot{\mathbf{v}}_k \right) \qquad (15)$$

and

$$\ddot{\mathbf{v}} = C(q)\ \ddot{q} + B(q,\dot{q}) \tag{16}$$

VECTORIAL DYNAMICS

From vectorial mechanics and the direct use of Newton's Laws,

$$\begin{aligned} F &= m\ \ddot{R} \\ N &= \dot{H} \end{aligned} \tag{17}$$

Gibbsian vectors and dyadics are commonly referred to as Newton-Euler schemes. These schemes tend to require reconstruction for each new mechanical system and thus are most useful only in understanding the behavior of simple systems.

For comparison, **E** is defined as an augmented matrix of forces and moments and **P** as an augmented matrix of linear and angular momentum. Thus,

$$\begin{aligned} E &= [\ F\ ,\ N\ ]^T \\ \dot{P} &= [\ m\ \ddot{R}\ ,\ \dot{H}\ ]^T \\ E &= \dot{P} \end{aligned} \tag{18}$$

Open Chains

Hooker and Margulies [12] studied the attitude dynamics of n rigid bodies interconnected by revolute and spherical joints having a total of r rotational degrees of freedom. As an alternative to solving 3 n differential equations incorporating 3 n-r constraint torques as well as 3 n-r constraint equations, they were able to eliminate the constraint torques from the dynamical equations by the solution of 6 n-r linear algebraic equations for 3 n angular velocity rates. Hooker [13] was later able to show by means of sets of simple summations and the appropriate projection that the constraint forces could be easily eliminated resulting in r scalar differential equations whose right sides required the solution of only r linear algebraic equations in what was to become known as the Hooker-Margulies-Hooker equations.

Later, these methods and ideas had moved over to non-space applications in a series of papers: Stephanenko and Vukobratovic [14], later with improved notation in Orin et al [15], Walker and Orin [16] who concentrated on computational efficiency, and finally an on-line application by Luh et al [17]. By then the technique was known as the recursive Newton-Euler formulation of the inverse dynamics problem.

In this approach, [12-17], the accelerations of each link in the chain are sequentially determined starting at the base or ground link of the chain resulting in recursive equations of the form:

$$\ddot{\mathbf{v}}_{i+1} = \ddot{\mathbf{v}}_{i+1}(\ddot{\mathbf{v}}_i \, , \, \dot{\mathbf{v}}_i \, , \, \ddot{q}^{\,i}_{i+1} \, , \, \dot{q}^{\,i}_{i+1}) \tag{19}$$

where a free-body force analysis, starting at the end or n-th link, and incorporating both external and internal forces and torques, is performed recursively on each prior link to form a set of inverse dynamic system simulation equations.

$$\begin{aligned} \boldsymbol{e}_i &= \left[\, \boldsymbol{f}_{i,i+1} \, , \, \boldsymbol{n}_{i,i+1} \, \right]^T \\ &= \boldsymbol{e}_i \, (\, \boldsymbol{e}_{i+1} \, , \, \dot{\boldsymbol{P}}_i \,) \end{aligned} \tag{20}$$

The inertial forces and torques on each link within the chain are determined from:

$$\begin{aligned} \dot{\boldsymbol{P}}_i &= \left[\, m_i \, \ddot{\boldsymbol{R}}_i \, , \, \dot{\boldsymbol{H}}_i \, \right]^T \\ &= \left[\, m_i \, \ddot{\boldsymbol{R}}_i \, , \, \boldsymbol{I}_i \, \dot{\boldsymbol{\omega}}_i + \tilde{\boldsymbol{\omega}}_i \, \boldsymbol{I}_i \, \boldsymbol{\omega}_i \, \right]^T \\ &= \dot{\boldsymbol{P}}_i \, (\, \ddot{\mathbf{v}}_i \, , \, \dot{\mathbf{v}}_i \,) \end{aligned} \tag{21}$$

where $\mathbf{I}_i$ is the inertia matrix for body i and

$$\tilde{\boldsymbol{\omega}}_i = \begin{bmatrix} 0 & -\omega_{iz} & \omega_{iy} \\ \omega_{iz} & 0 & -\omega_{ix} \\ -\omega_{iy} & \omega_{ix} & 0 \end{bmatrix} \tag{22}$$

Both constraint and actuator forces and torques are contained within these equations. The actuation torque about a rotational joint or force along a sliding joint are determined by simply taking the appropriate projection.

$$\tau_i = \left\{ \begin{array}{ll} n_{i,i+1} \cdot z_{i-1} , & \textit{if link } i \textit{ is rotational} \\ f_{i,i+1} \cdot z_{i-1} , & \textit{if link } i \textit{ is translational} \end{array} \right\} \tag{23}$$

The remaining forces and torques at the joint are constraint forces and torques.

When only the actuator forces and torques are desired, Eqs. (18)-(23) are combined and rewritten as

$$\tau = M(q^i) \ddot{q}^i + D(q^i , \dot{q}^i) \tag{24}$$

Details on how to efficiently cast the recursive Newton-Euler forms of these equations are given by Walker and Orin [16]. The inertia matrix **M** and the cross coupled inertial forces **D** are both functions of q and thus require continual re-evaluation. For solving the forward dynamics problem Eq. (24) is recast as

$$\ddot{q}^u = M(q^u)^{-1} [\tau - D(q^u , \dot{q}^u)] \tag{25}$$

These techniques for determining (1) joint force and torques, (2) actuator forces and torques, and (3) a dynamic simulation can be further extended for use in closed-chain linkages, as shown by Orin and McGhee [18]. This is accomplished by incorporating the constraints, afforded by the kinematic loops, through augmenting the system dynamic equations. The constraints, via virtual work, are of two forms--position and force. The constraints between a link's absolute position coordinates and defining joint coordinates is given by Eq. (11) and in terms of acceleration constraints by Eq. (16). The latter is simply modified when only the otherwise independent coordinates are used to define the link's absolute position.

$$\ddot{\mathbf{v}} = C(q^i) \ddot{q}^i + B(q^i, \dot{q}^i) \tag{26}$$

The effect of a loop closure force, $\mathbf{e}^{ext}$, is a superposition of additional constraint and actuation forces onto all of the joints within the chain. The additional effect is determined through the use of a family of kinematic influence coefficient matrices $\mathbf{G}_i$. Each relates the known external closure force to the six additional constraint and actuation forces that occur at the appropriate joint, i.

$$e_i^{con} = G_i(\ q\ )\ e^{ext} \tag{27}$$

In cases where the desired joint trajectory is known as is the additional external loop closure force at one point within the chain, the resulting absolute motion and required joint torques can be determined. In such a case, only subsets of the kinematic influence coefficient matrices are used.

Through the use of virtual work, Whitney [19] and Shimano [20] have shown that the same Jacobian which relates infinitesimal motions at a particular location to motions at the joints can be used to relate actuator forces and torques at the joints to effective external forces and torques at the particular location. That is, if one takes the first variation of Eq. (11) while utilizing otherwise independent coordinates, one obtains

$$\delta \mathbf{v} = C(q^i)\ \delta q^i \tag{28}$$

The virtual work at the particular point due to the external load must equal the virtual work at the joints due to the actuator torques and forces required to balance this load

$$\delta W = e^{ext^T}\ \delta \mathbf{v} = \tau^{con^T}\ \delta q^i \tag{29}$$

Substituting Eq. (28) into Eq. (29) and eliminating the independent virtual displacements results in

$$\tau^{con} = C(q^i)^T\ e^{ext} \tag{30}$$

The total actuator torques are the sum of the torques responding to inertial loads, Eq. (24), and the torques due to the external load, Eq. (30). The motion is constrained as in Eq. (26) resulting in:

$$\begin{bmatrix} \tau \\ \ddot{\mathbf{v}} \end{bmatrix} = \begin{bmatrix} M & C^T \\ C & 0 \end{bmatrix} \begin{bmatrix} \ddot{q}^i \\ e^{ext} \end{bmatrix} + \begin{bmatrix} D \\ B \end{bmatrix} \tag{31}$$

The inversion of Eq. (31) is a solution of the forward dynamics problem of a simple closed-loop system. Luh and Zheng [21] have further extended the use of recursive Newton-Euler inverse dynamics algorithm to handle additional cases of closed loops.

A more direct way of accomplishing forward dynamics simulations for open chain linkages than provided by Eqs. (19)-(25), while still utilizing recursive forms of Newton-Euler equations, became possible when Vereshchagin [22], using Gauss's principle of least constraint [23], was able to formulate recursive forms of recursive Newton-Euler forward dynamics algorithms. Featherstone [24], using spatial algebra based on screw calculus, showed that these forward dynamics algorithms could in theory be used for closed chain linkages and Lathrop [25] showed how this could be accomplished by means of a backward propagation. The method required, for an otherwise single loop kinematic chain, that twelve end-point force and acceleration environment interactions be fully characterized in terms of end-point constraints.

Closed Chains

Direct use of Newton's Laws for closed looped chains tends to be application specific, as exemplified in the planar analysis programs MEDUSA [26] and VECNET [27]. In this method, the sums of forces and moments on each of the n links are separated into internal reaction forces and torques between the link elements and external forces and torques including those due to actuation. For generality, the spatial pose notation follows.

$$E_i = [\, F_i \,,\, N_i \,]^T$$

$$= \left[f_i^{ext} + \sum_{j=1}^{p} f_{ij} \,,\, n_i^{ext} + \sum_{j=1}^{p} n_{ij} \right]^T = e_i^{ext} + \sum_{j=1}^{p} e_{ij} \qquad (32)$$

where

$$n_{ij} = r_{ij} \times f_{ij}$$

Equations (18), (21) and (32) are then combined resulting in 6 differential equations, 3 in the planar case, for each link.

$$M_i \, \dot{\mathbf{v}}_i + \dot{\mathbf{v}}_i^T D_i \dot{\mathbf{v}}_i = \sum_{j=1}^{p} e_{ij} + e_i^{ext} \;;\quad i = 1 \,,\, \cdots \,,\, n \qquad (33)$$

In the above programs, joint constraints, similar to Eqs. (9) and (2) are used to supply the remaining needed information. For each planar revolute pair two such joint equations occur. A spatial revolute results in five such scalar equations. Altogether, r scalar joint equations exist. Thus, for a spatial linkage there are 6 n scalar differential equations and r algebraic equations in the accelerations. By using and manipulating these expressions, r pairs of reaction force f_{ij} and moment n_{ij} components occurring in Eqs. (33) are in principle eliminated using algebra. The actual algebraic procedure tends to be case specific and difficult.

The technique described above uses absolute coordinates but can be modified to use relative joint coordinates. Similarly, loop or partial loop constraints as opposed to joint constraint equations be used. See for example; Do [28] who analyzed the inverse dynamics of a spatial Stewart Platform; Hunt [29] who showed that Plucker (relative) coordinates, screw calculus and motor algebra are particularly useful in describing linkage motions; and, Fichter [30] who used the methods to define the kinematic and dynamics of spatial Stewart Platforms. Although details are quite different than those presented here, the end results are dynamic system equations with great similarity.

Virtual Work and d'Alembert's Principle

The use of virtual work and d'Alembert's Principle within the context of Newton-Euler formulations of closed chains was developed by Paul [7, 9, 31] who restricted his analysis to the planar cases. In his procedure he recasts Eq. (18) in light of virtual work and d'Alembert's principle and then sums the equations over all of the links. Again, a spatial notation is used.

$$\sum_{i=1}^{n} \left[\left(\boldsymbol{E}_i - \dot{\boldsymbol{P}}_i \right) \cdot \delta \mathbf{v}_i \right] = 0 \tag{34}$$

He then sets up loop constraint equations similar to Eq. (9) and partial loop constraint equations which relate the pair variables or relative coordinates to each of the body absolute positions similar to those of Eq. (11). By taking the variation of the partial loop constraints and substitution into Eq. 6 the variations $\delta \boldsymbol{\nu}_i$ can be written in terms of the joint variables $\delta \mathbf{q}$. Taking the first and second time derivative of the loop constraint equations, recasting as in Eq. (14) and substituting, leads to the system of equations;

$$\boldsymbol{S}\, \ddot{\boldsymbol{q}}^{\,i} = \boldsymbol{R} \tag{35}$$

Even in the planar case presented here, the expressions used to resolve S and R are rather complex and these resulting equations are highly coupled. R includes all squared-velocity terms such as centripetal and Coriolis terms and, if the technique were further expanded to the spatial case, would include gyroscopic effects. In addition, solving for the constraint forces requires an additional set of coupled algebraic equations and knowledge of the joint accelerations. A variation to the approach for determining bearing reactions of closed linkages while simultaneously keeping the number of system equations small has been recently proposed by Litvin [32].

ANALYTICAL DYNAMICS\LaGRANGIAN MECHANICS

The methods of analytical dynamics either start from utilization of the principle of virtual work including inertial forces leading first through LaGrange's form of d'Alembert's Principle or Kane's Method or directly from minimization concepts. The latter approach being more correctly associated with the analytical dynamics approach and the first being more commonly found in the applications literature. The minimization approach, through the Principle of Least Action, can be used to obtain both LaGrange's and Hamilton's Canonical Equations. The analytical dynamics approach as envisioned by LaGrange [33] and properly embellished in Pars [34] can be easily used for systems with integrable or holonomic constraints. It is problematic with non-integrable constraints having to be individually tailored to the problem. The analytical dynamics approach when applied to Gauss's Integral [23], can be used to form an alternative set of dynamic equations more attuned to the goal of constraint force determination and control.

LaGrange's Equations, Closed Chains, and Relative Coordinates

Chace [35], through the use of LaGrange multipliers, relative coordinates and the following principle of least action:

$$\delta \int_{t_1}^{t_2} \left[T - V - F - \sum_{j=1}^{r} \lambda_j \, \phi_j \right] = 0 \qquad (36)$$

was able to both describe the system motion and determine reaction/constraint forces to which he was interested. This was done through the use of loop constraint functions, as given in Eq. (10), where the relative coordinates one wishes to use to describe the motion are retained and r constraint equations of the partial loop form, as given in Eq. (9), are used. These constraints restrict the motion of similar points on the interconnecting bodies, forcing the d'Alembert multipliers to equal the constraint forces desired. This latter statement is made clear later in the section entitled LaGrange's form of d'Alembert's Principle. The generalized coordinates q_i include r selected absolute coordinates ν_i and dependent relative coordinates as well as m independent relative coordinates q_j. Such application results in LaGrange's Equations with constraints [36] to which the appropriate constraint equations are appended. Solution requires that the constraints be written in Pfaffin form and that the second time-derivative of the constraint equation be similar in form to Eq. (13).

$$\frac{d}{dt}\left(\frac{dT}{d\dot{q}_i}\right) - \frac{dT}{dq_i} = Q_i - \sum_{j=1}^{r} \lambda_j \left(\frac{d\phi_j}{dq_i}\right); \quad i = 1, \cdots, m + r$$

$$\phi_j (q_1, \cdots, q_{m+r}) = 0 ; \quad j = 1, \cdots, r \tag{37}$$

$$C\,\dot{q} + A = 0$$

$$C\,\ddot{q} + B = 0$$

Q_i is a composite force taking into account both the strength of each external force or torque and its relative advantage to the motion defined by q_i.

If it is assumed that no explicit knowledge of reaction forces and torques is desired then the minimal number of dynamical system equations is governed by the minimum number of joint pair variables that would have to be determined. For the spatial Stewart Platform each branch would require (starting from the base platform) a minimum of 2 rotations and 1 translation or a total of 18 coordinates associated with the six branches. With the addition of the six coordinates required for the moving platform 24 dynamic system equations would result. The 18 constraint equations result from the six sets of vector loop equations in position. For the planar Stewart Platform case there are three branches with two degrees of freedom each plus the moving platform with three degrees of freedom giving 9 dynamic system equations. Three vector loop equations for position supply the six required constraint equations.

Uicker [37], almost simultaneously to Chace, was able to formulate the dynamic system equations without requiring LaGrange multipliers by representing the kinetic energy in terms of a minimal set of relative coordinates through the use of homogeneous transformation matrices [38]. This simplification occurred at the expense of creating fairly complex coefficients in the dynamic system equations very similar to Eq. (33).

$$M(q^i)\ \ddot{q}^i + (\dot{q}^i)^T\ D(q^i)\ \dot{q}^i = Q \tag{38}$$

The mass matrix, M, and coupling matrix D are determined by solving first for generalized coordinates consistent with the constraints. Such equations are formed by equating the multiplication of relative link homogeneous transformation matrices with the identity matrix, in what may be called a pose equation, followed by numerical reduction of a fairly large expression. The reduction is typically done through the use of an explicit time differentiation of the pose equation to result in a set of velocity constraint equations that can be used in a numerical algorithm. This must be done at each time step within the simulation. Finally the constraint forces, if needed, are determined by utilizing an axial set of equations determined from a kineto-static analysis typically derived using virtual work ideas presented earlier.

LaGrange's Equations, Closed Chains, and Absolute Coordinates

An alternative to the Uicker and Chace approaches of using relative coordinates within the Lagrangian formulation is to use absolute coordinates as first proposed by Chace [39] and latter by Haug [40] when evaluating machine dynamics of planar systems.

$$\begin{aligned} M_i\ \ddot{\mathbf{v}}_i + \dot{\mathbf{v}}_i\ D_i\ \dot{\mathbf{v}}_i &= C_i^T\ \boldsymbol{\lambda} + e_i^{ext}\ ;\ i = 1, \cdots, n \\ C_j\ \ddot{\mathbf{v}} + B_j &= 0\ \ ;\ \ j = 1, \cdots, r \end{aligned} \tag{39}$$

In Eq. (39) some modifications were made in light of the work of others and so as not to introduce additional notation. For example the partial of the constraint equation with respect to the absolute coordinates has been replaced by the transpose of the first column of the constraint matrix, in accordance with utilizing Eq. (12) and (13). The j constraint equations are usually written in Pfaffin form and their derivatives are utilized in obtaining the solution. The problem with this method is the large number of generalized constraint and dynamic system equations. The benefit is the fairly sparse and constant mass and coupling matrices. Eq. (39) has a similarity with that derived using Newton's Laws and absolute coordinates given by Eq. (33); the difference being the means of including constraint forces and the solution method required. Nikravesh has shown the applicability of Haug's work to spatial mechanisms through the use of Euler parameters [41]; his approach is contained within the above expression.

To solve the dimensionality problem Wehage [42] developed a coordinate partitioning method which reduced the number of equations to the number of degrees of freedom and used the technique in DADS-2D. Other methods, such as those typically utilized in the solution of electrical system simulation to deal with sparsity, were used by Orlandea [43] in the development of ADAMS. In coordinate partitioning as applied to dynamic systems, a generalization of the coordinate reduction methods found in finite element programs is utilized. The absolute coordinates are separated into a set of dependent and independent generalized coordinates and then the technique described in Eq. (14) is used to define the dependent velocities and accelerations in terms of the independent set.

$$\begin{aligned} \dot{\mathbf{v}} &= \begin{bmatrix} \dot{q}^i \\ \cdots \\ \dot{q}^d \end{bmatrix} \\ \dot{q}^d &= \left[\ C^i\ \right]^{-1} \left(\ A\ -\ C^d\ \dot{q}^i\right) \\ \ddot{q}^d &= \left[\ C^i\ \right]^{-1} \left(\ B\ -\ C^d\ \ddot{q}^i\right) \end{aligned} \tag{40}$$

Next the dynamic system equations themselves, the first n+r Eqs. (39), must be partitioned and the dependent coordinates eliminated with the aid of Eq. (40).

Typically, coordinate partitioning is done numerically and because of potential convergence problems associated with solving the constraint equation it may be beneficial to change the choice of independent and dependent coordinates continually. Nikravesh in [44] discusses a particular scheme to accomplish this task and with Chang [45] through feedback control theory methods, was able to reduce convergence problems without the need for partitioning.

Common to each of these methods is the realization that differential equations coupled to algebraic equations are stiff [46], containing widely separated natural frequencies, and require special numerical methods such as those introduced by Gear [47]. In addition, computer codes such as ADAMS and DADS append the constraint equations to the dynamic system equations and cast the results in first-order form. The references above describe the various methods of accomplishing these tasks.

LaGrange's Equations, Open Chains, and Relative Coordinates

A couple years after Uicker's original work of using relative coordinates, homogeneous transformations and LaGrange's Equations were used by Kahn [48,49] to formulate the dynamics of closed spatial linkages. The results were similar to Uicker and Eq. (38); constraint forces were automatically eliminated toward the goal of computational efficiency. A recursive version of this formulation was latter developed by Hollerbach [50] and found to be computationally less efficient than the recursive Newton-Euler approach.

Kinematic Influence Coefficients, Power and LaGrange' Equations

Teasar uses an equivalent scheme to that of Uicker through the use of kinematic influence coefficients. Modrey [51] showed that velocities of any link within a planar closed chain, could be related to first-order influence coefficients, G_{jk}, and input pair variable velocities whereas link accelerations were related to a combination of first-order influence coefficients to input pair variable accelerations and by second-order coefficients, H_{jkl}, to input pair variable velocities. Recasting Modrey's work for use in spatial systems we have the link's absolute positional and orientation, $\boldsymbol{\nu}$ related to the minimal set of relative coordinates of the linkage chain, $\mathbf{q}^u$, and through the use of derivatives we find.

$$\mathbf{v}_i^i = \left[\mathbf{R}_i^i \, , \, \boldsymbol{\Theta}_i^i \right]^T = \mathbf{v}_i^i (\mathbf{q}^i) ; \qquad i = 1, \cdots , n$$

$$\dot{\mathbf{v}}_i^i = \frac{d\mathbf{v}_i^i}{dt} = \sum_{k=1}^{m} G_{ik}^i \, \dot{q}_k^i$$

$$\ddot{\mathbf{v}}_i^i = \sum_{l=1}^{m} \sum_{k=1}^{m} H_{ikl}^i \, \dot{q}_k^i \, \dot{q}_l^i + \sum_{k=1}^{m} G_{ik}^i \, \ddot{q}_k^i \tag{41}$$

where

$$\frac{\partial \mathbf{v}_i^i}{\partial q_k^i} = G_{ik}^i = \frac{\partial \dot{\mathbf{v}}_i}{\partial \dot{q}_k^i} ; \quad \frac{\partial^2 \mathbf{v}_i^i}{\partial q_k^i \, \partial q_l^i} = H_{ikl}^i ; \quad \dot{q}_k^i = \frac{dq_k^i}{dt} ; \quad \ddot{q}_k^i = \frac{d^2 q_k^i}{dt^2}$$

The means for determining the kinematic influence coefficient matrices are discussed in detail by Benedict [52]. An equivalent scheme for obtaining these matrices, as presented by Kane in the utilization of quasi coordinates, is given in a later section.

Benedict points out that the minimal set of generalized coordinates, in the case of closed chains, is defined through the use of Assur groups and their associated poles. Assur groups are sets of interconnected elements that appear to be a structure if their extremities were temporarily held fixed. The group at that instant moves as if it were a single rigid body. The influence coefficients represent how a specific point in the group would move given an infinitesimal motion of one of these extreme links. Benedict notes that the kinematic influence coefficients can be used to recast the kinetic energy in terms of the associated minimal set of generalized coordinates.

$$KE = \frac{1}{2} \sum_{i=1}^{n} \left(m_i \, \dot{\mathbf{R}}_i \cdot \dot{\mathbf{R}}_i + \boldsymbol{\omega}_i \cdot \mathbf{I}_i \cdot \boldsymbol{\omega}_i \right)$$

$$= \frac{1}{2} \sum_{i=1}^{n} \dot{\mathbf{v}}_i \, \mathbf{M}_i^i \, \dot{\mathbf{v}}_i = \frac{1}{2} \sum_{k=1}^{m} \sum_{l=1}^{m} \dot{q}_k^i \, M_{kl}^* \, \dot{q}_l^i \tag{42}$$

where

$$M_{kl}^{i*} = \sum_{i=1}^{n} G_{ki}^i \, M_i \, G_{li}^i$$

In a similar fashion the power associated with the inertia motion can be evaluated.

$$P^{inertia} = \frac{d}{dt}(KE) = \sum_{k=1}^{m} Q_k^{inertia} \dot{q}_k^{\,i}$$

where

$$Q_k^{inertia} = \sum_{l=1}^{m} M_{kl}^{i*} \ddot{q}_l^{\,i} + \frac{1}{2} \sum_{k=1}^{m} \sum_{l=1}^{m} M_{kl}^{i*\prime} \dot{q}_k^{\,u} \dot{q}_l^{\,i} \qquad (43)$$

$$M_{kl}^{i*\prime} = \sum_{i=1}^{n} H_{ki}^{i} M_i G_{li}^{i}$$

As an alternative to the use of LaGrange's equations Benedict noted that a "power balance" analysis could be used to derive dynamic system equations. Actuators are a source/sink of power into the linkage and this power becomes distributed by changes in kinetic energy and energy associated with body forces such as those associated with conservative potential energies ar non-conservative/dissipation energies. In all cases the kinematic influence coefficients are used to relate the power to changes in the generalized coordinates. Since these coordinates are independent the power equation can be separated into m torque equations.

$$P^{actuator} + P^{body} = P^{inertia}$$

$$P^{actuator} = \sum_{k=1}^{m} Q_k^{actuator} \dot{q}_k^{\,i} \; ; \quad P^{body} = \sum_{k=1}^{m} Q_k^{body} \dot{q}_k^{\,i} \qquad (44)$$

$$Q_k^{actuator} + Q_k^{body} = Q_k^{inertia} \; ; \qquad k = 1, \cdots, m$$

In light of Eq. (43), the dynamic system equations have exactly the same form as that found by Uicker, Eq. (38). As such they suffer from the same problem. Benedict's work was later extended by Thomas [53] for use in modelling open chain or serial linkages. Hudgens [54] worked out a detailed example of the technique's use for the case of a spatial Stewart Platform.

LaGrange's form of d'Alemberts Principle

LaGrange's form of d'Alemberts Principle is essentially a shortcut to obtaining results similar to that of LaGrange's equations without the necessity of the manipulations required in first obtaining kinetic energy and then performing the required partial derivatives. Generalizing the approach, as is common [10], there is for the rigid body case;

$$Q_k + Q_k^* = Q_k^{con}; \qquad k = 1, \cdots, m+r$$

$$Q_k = \sum_{j=1}^{n} \boldsymbol{E}_j \cdot \frac{\partial \dot{\mathbf{v}}_j}{\partial \dot{q}_k} = \sum_{j=1}^{n} \boldsymbol{E}_j \cdot G_{jk}\, \hat{\boldsymbol{i}}_{jk}$$

$$Q_k^* = -\sum_{j=1}^{n} \dot{\boldsymbol{P}}_j \cdot \frac{\partial \dot{\mathbf{v}}_j}{\partial \dot{q}_k} = \sum_{j=1}^{n} \dot{\boldsymbol{P}}_j \cdot G_{jk}\, \hat{\boldsymbol{i}}_{jk} \qquad (45)$$

$$Q_k^{con} = \boldsymbol{C}_k^T\, \boldsymbol{\lambda} = \sum_{j=1}^{n} \boldsymbol{e}_j^{con} \cdot \frac{\partial \dot{\mathbf{v}}_j}{\partial \dot{q}_k} = \sum_{j=1}^{n} \boldsymbol{e}_j^{con} \cdot G_{jk}\, \hat{\boldsymbol{i}}_{jk}$$

$$\boldsymbol{C}_s\, \dot{\boldsymbol{q}} + \boldsymbol{A}_s = 0\,; \qquad s = 1, \cdots, r$$

One attribute of the approach is that both $\boldsymbol{E}_j$ and $\dot{\boldsymbol{P}}_j$ are covariant vectors whereas $\frac{\partial \dot{\mathbf{v}}_i}{\partial \dot{q}_k}$ and thus $\hat{\boldsymbol{i}}_{jk}$ are contravariant vectors, the dot product between them being an invariant [55]. In essence the inner product selects that portion of the effort which is in the direction of the motion causing work. The kinematic coefficient matrix is similar to that discussed earlier in Eq. (41) and in fact is equivalent when the generalized coordinates selected in both methods are the same. It is expressed here as pairs of magnitudes along and about unit vectors in the translation and rotations directions associated with the particular generalized coordinate as opposed to components in the inertial $\hat{\boldsymbol{i}}$, $\hat{\boldsymbol{j}}$, $\hat{\boldsymbol{k}}$ directions.

In this, like many of the other methods, one is free to choose the number and meaning of the generalized coordinates and the particular details of the constraint equations as long as the system is fully described; and the total number of degrees of freedom, m, plus the number of constraints r is equal to the number of generalized coordinates. The larger the number of generalized coordinates selected, the larger the number of constraints equations required.

The constraint equations are typically combinations of joint or loop form. The generalized coordinates selection predicates which constraint forces will be determined. Typically the desired practice is to select coordinates so that constraints of no interest are (automatically) not involved in the derivation of the dynamic system equations. To do so requires defining the absolute motion of bodies constrained by such forces to be functions of generalized coordinates that do not have the potential of causing motions in the direction of these forces. In such cases the dot product above is zero. The difficulty comes in making such a selection and then determining the actual "partial loop" relationships used to define the absolute body motion in terms of the generalized coordinates. Also note that no motion in the direction need take place. In fact, no motion will occur in the direction if it is constrained.

Using the spatial Stewart Platform case as an example, it is rather easy to select six generalized coordinates for each link resulting in 78 generalized coordinates and thus 78 differential equations. Since there are only 6 actual degrees of freedom in such a linkage 72 constraint equations are required. These equations can be formed easily as each of the 6 prismatic pairs has 5 constraints, each of the 6 Hooke pairs has 4 constraints and each of the 6 globes has 3 constraints. Thus, 78 differential equations coupled to 72 algebraic equations associated with the constraints are used within the simulation. If, on the other hand, loop constraints associated with each of the 6 branches are utilized - indicating that the pose of the moving platform must be equal to the pose determined through a branch - then only 42 generalized coordinates are necessary to describe the system motion and 36 constraint forces will be determined. This requires that the motion of each body within the platform be defined within partial loop equations that utilize this more limited set of coordinates.

Quasi-coordinates and Kane's Method

A quasi-coordinate method further developed by Kane [56, 57] is typically used when a minimal set of coordinates is desired. In such a case LaGrange multipliers are no longer utilized and are eliminated from Eq. (45). This approach is used in the TREETOPS program. The technique, although commonly referred to as Kane's method is very similar to a procedure originally perfected by Appell [58], first anticipated by Gibbs [59] and used also by Volterra and Boltzmann. A variation on the scheme was also developed by Dirac [60]. Essentially the quasi-coordinate method is a formal scheme that can be used to eliminate dependent coordinates and is commonly used for systems with rotation primarily because finite rotations are not vectors, because they are order dependent and thus non-communitive. Angular velocities on the other hand are communitive and thus make ideal choices for derivatives of quasi-coordinates.

The approach is as follows. First find the opportune choice for quasi-coordinates $\mathbf{u}$ and relate them to a subset of independent generalized coordinates, $\mathbf{q}^n$ by;

$$\begin{aligned} \dot{u} &= W(q^i, t)\,\dot{q}^i + w(q^i, t) \\ \dot{q}^i &= W^{-1}(\dot{u} - w) \end{aligned} \tag{46}$$

where there are m=n-r independent μ. The matrix **W** is of m x m dimension and non-singular whereas **w** is m x 1. **W** and **w** are chosen so that explicit and unique solutions for independent generalized velocity can be found using the second of Eqs. (46). A common choice is to make **W** and **w** identity and null matrices, respectively. To modify Eq. (45) so that the state variables are a minimal set of quasi coordinates as opposed to the non-independent set of generalized coordinates requires that absolute velocity be written in terms of the quasi coordinates. First, relationships of the form

$$\dot{\mathbf{v}}_i = V_i(\, q^i,\, t)\; \dot{u} \;+\; V_{it}\;(\, q^i,\, t) \tag{47}$$

are required so that $\dot{\mathbf{v}}_i$ can be replaced by $\dot{u}$ in Eq. (46). To obtain these relationships note that the second of Eqs. (41) when written as a function of all generalized coordinates, q, as opposed to the minimal set is given by,

$$\begin{aligned}
\dot{\mathbf{v}}_i &= \frac{d\mathbf{v}_i}{dt} = \sum_{k=1}^{n} \frac{\partial \dot{\mathbf{v}}_i}{\partial \dot{q}_k}\, \dot{q}_k + \frac{\partial \mathbf{v}_i}{\partial t} \\
&= \sum_{k=1}^{m} \frac{\partial \dot{\mathbf{v}}_i}{\partial \dot{q}_k}\, \dot{q}_k + \sum_{k=m+1}^{n} \frac{\partial \dot{\mathbf{v}}_i}{\partial \dot{q}_k}\, \dot{q}_k + \frac{\partial \mathbf{v}_i}{\partial t} \\
&= \frac{\partial \dot{\mathbf{v}}_i}{\partial \dot{q}^i}\, \dot{q}^i + \frac{\partial \dot{\mathbf{v}}_i}{\partial \dot{q}^d}\, \dot{q}^d + \frac{\partial \mathbf{v}_i}{\partial t}
\end{aligned} \tag{48}$$

To eliminate the dependent coordinates, $\mathbf{q}^c$, utilize the partitioning technique presented in Eq. (14). To eliminate the dependent coordinates, $\mathbf{q}^u$, utilize the second of Eqs. (46). Substituting and rearranging, there is:

$$\begin{aligned}
\dot{\mathbf{v}}_i = &\left(\left(\frac{\partial \dot{\mathbf{v}}_i}{\partial \dot{q}^i} - \frac{\partial \dot{\mathbf{v}}_i}{\partial \dot{q}^d} \left[\, C^i \,\right]^{-1} \; C^d \right) W^{-1} \right) \dot{u} \\
&- \left(\left(\frac{\partial \dot{\mathbf{v}}_i}{\partial \dot{q}^i} - \frac{\partial \dot{\mathbf{v}}_i}{\partial \dot{q}^d} \left[\, C^i \,\right]^{-1} \; C^d \right) W^{-1} \right) w \\
&+ \frac{\partial \dot{\mathbf{v}}_i}{\partial \dot{q}^d} \left[\, C^i \,\right]^{-1} A \;+\; \frac{\partial \mathbf{v}_i}{\partial t}
\end{aligned} \tag{49}$$

Direct comparison of Eq. (49) with Eq. (47) yields the required values for V_i and V_{it}. Thus, Kane's equations in matrix form become

$$
\begin{aligned}
& Q + Q^* = 0 \\
& Q = \sum_{i=1}^{n} (E_i \cdot V_i) \; ; \qquad Q^* = - \sum_{i=1}^{n} (\dot{P}_i \cdot V_i) \\
& C \, \dot{q} + A = 0 \\
& \dot{u} = W(q^i, t) \, \dot{q}^i + w(q^i, t)
\end{aligned}
\tag{50}
$$

where **Q** and **Q*** are of minimal dimension m and all non-working constraint forces have been eliminated. To examine the power and generality of the approach, take the case where **W** is an identity, **w** is null, and the independent relative coordinates and the absolute coordinates are not explicit functions of time, then Eqs. (41), (47), and (50) lead to

$$
\begin{aligned}
& \dot{u} = \dot{q}^i \; ; \quad C \, \dot{q} = 0 \; ; \quad A = 0 \\
& \dot{\mathbf{v}}_i = V_i(q^i, t) \, \dot{q}^i \; ; \quad V_t (q^i, t) = 0 \; ; \quad \frac{\partial \mathbf{v}}{\partial t} = 0 \\
& G_i = \frac{\partial \dot{\mathbf{v}}_i}{\partial \dot{q}^i} - \frac{\partial \dot{\mathbf{v}}_i}{\partial \dot{q}^d} \left[C^i \right]^{-1} C^d
\end{aligned}
\tag{51}
$$

The latter of these is the expression used by Benedict [52] to determine the kinematic influence coefficients in his special case planar example. For the spatial Stewart Platform the value in the approach is that all unnecessary coordinates can be eliminated leaving the motion of the 6 platform absolute coordinates to be a function of the prismatic pair actuation forces. An earlier utilization of such an approach on Spatial Stewart Platforms has not been found.

Using Kane's method, Huston was able to show in a series of papers, including a summary work [61], how the automatic elimination of nonworking internal constraint forces without the introduction of tedious differentiation or other calculations could be applied in describing the dynamics of complex multi-linked open chain systems.

Constraint Elimination through Pseudo Inverses and Singular Value Decomposition

Although Kane's approach does not utilize LaGrange multipliers, it requires that the generalized velocities be partitioned and a subset of generalized coordinates selected for elimination. Arbitrary selection may result in mathematical singularities due to the nature of Eq. (14). When large motions occur, the constraint equations might reduce three-dimensional loop constraint equations into two dimensional equations at particular locations resulting in singular solutions on the third dimension. Both types of effects also occur when utilizing dynamic equations that incorporate LaGrange multipliers such as Eqs. (39) or (45).

These problems are generic to all of the above solution methods. For example, recall the coordinate partitioning method developed by Wehage for use in DADS-2D [42]. It has been pointed out by Haug [62] that generalized coordinate partitioning suffers from difficulties in stiff systems and has defied implementation with an implicit algorithm. Nikravesh [44] has proposed methods that continually select the coordinate to be eliminated based on a "stiffness metric" and Haug [62] has proposed an alternative numerical integration approach which partitions in an implicit manner to help overcome this problem. Nonetheless, 2n+2r first-order coupled nonlinear differential and algebraic equations are required for the solution.

An alternative scheme utilizing singular value decomposition has been suggested by Singh [63]. The technique can be applied equally as well to system equations otherwise derived from the methods mentioned above. The method employs a partitioning scheme which is very sensitive to the stiffness of the system. Stiff coordinates cause singularities and are thus automatically eliminated.

If a system is unconstrained and its motion is described by a minimal number of generalized coordinates, it has an equations of motion,

$$M(q^i, t)\ \ddot{q}^i = Q\ (\ q^i, \dot{q}^i, t) \tag{52}$$

where M is n x n, positive definite and symmetric. In the presence of constraints onto an otherwise non-constrained system the result is an imposition of as yet unknown generalized constraint forces, Q_c and a resulting constrained motion. The generalized coordinates contain both independent and dependent types.

$$M(q, t)\ \ddot{q} = Q\ (\ q, \dot{q}, t) + Q_c\ (\ q, \dot{q}, t) \tag{53}$$

Assume that the kinematic constraints expressed as in Eq. (12) can be written in Pfaffian form as expressed in Eq. (13). If there are r constraints, the constraint matrix C is of dimension r x n with $r \leq n$. Of the $\mathbf{q}$'s, only m=n-r can be selected to be independent. As seen earlier, coordinate partitioning as used by Haug/Wehage or by Kane allows a dependent set of

coordinates to be written in terms of a dependent set and this in turn allows the dimension of the dynamic equations to be reduced. The problem is that the coordinate partitioning scheme requires that a particular set of coordinates or quasi-coordinates be chosen, and this selection may lead to a singularity problem. An alternative approach could be used if the constraint matrix were invertible.

Inversion of Generalized Matrices. Because the methods for inverting non-square matrices are not well known, this topic is briefly discussed here.

Assume that one wishes to solve the first of Eq. (13) for the generalized velocity knowing that C is of dimension r x n and that A is a column vector. An approach to solve for the velocity is to modify the equation as follows.

$$\begin{aligned} \boldsymbol{C}\,\dot{\boldsymbol{q}} + \boldsymbol{B} &= 0 \\ \boldsymbol{C}^{T}\boldsymbol{C}\,\dot{\boldsymbol{q}} + \boldsymbol{C}^{T}\boldsymbol{B} &= 0 \\ \dot{\boldsymbol{q}} + [\,\boldsymbol{C}^{T}\boldsymbol{C}\,]^{-1}\,\boldsymbol{C}^{T}\boldsymbol{B} &= 0 \end{aligned} \qquad (54)$$

It is seen that the inverse for pre-multiplication or left-side multiplication of C by $\boldsymbol{C}^{+L} = [\boldsymbol{C}^{T}\,\boldsymbol{C}\,]^{-1}\,\boldsymbol{C}^{T}$ leads to an n by n identity matrix. In a similar fashion post or right side multiplication of C by $\boldsymbol{C}^{+R} = \boldsymbol{C}^{T}[\,\boldsymbol{C}\,\boldsymbol{C}^{T}\,]^{-1}$ leads to an r by r identity matrix. These are but two parts of a more generalized inverse known as Moore [64]-Penrose [65] inverse, $\boldsymbol{C}^{+} = \boldsymbol{C}^{+R}\,\boldsymbol{C}\,\boldsymbol{C}^{+L}$. Both the left- and right-side inverses require the inversion of a square matrix. In many cases one of these is singular. However, many problems can be solved if either the left or right inverse is available.

Singular Value Decomposition. In cases where neither is available or if one wishes to use an algorithm which will work even if the matrices are singular, a technique known as singular value decomposition, SVD, can be used to obtain the inverse. Because of the anticipated importance of using various classes of pseudo inverses and SVD to solve mechanical linkage dynamics problems some details are given here. Additional details and stable numerical schemes for performing SVD and associated inverses are given in Golub [66,67] and Klema [68].

The SVD method is in reality an eigenvalue-eigenvector scheme. For example assume that one had a square matrix **D** and wished to determine its eigenvalues λ_i and its eigenvectors V_i. In the eigenvalue-eigenvector problem:

$$D\,V = \lambda^2\,V$$

$$V = [\,V_1\,,\,\cdots\,,\,V_n\,];\quad V_i = \begin{bmatrix} V_{i1} \\ \vdots \\ V_{in} \end{bmatrix};\quad \lambda^2 = \begin{bmatrix} \lambda_1^2 & & 0 \\ & \ddots & \\ 0 & & \lambda_n^2 \end{bmatrix} \tag{55}$$

The solution for the eigenvalues is through use of a determinate and the solution for the eigenvalues is by solving a set of coupled linear equations that incorporate an associated eigenvalue.

$$\begin{aligned} &Det\,(D - \lambda^2) = 0\;;\quad \text{gives} \quad \lambda_1^2 \geq \lambda_i^2 \geq \lambda_n^2 \geq 0 \\ &\left(D - \lambda_i^2\, I\right) V_i = 0\;;\quad \text{gives} \quad V_{i1}\,,\,\cdots\,,\,V_{in} \end{aligned} \tag{56}$$

The eigenvalues by their nature are orthogonal. The arbitrary constant associated with the eigenvalue solution is usually chosen to make the vectors orthonormal. Thus

$$\begin{aligned} &V^T = V^{-1} \\ &V^T D\,V = V^T \lambda^2\,V = \lambda^2 \end{aligned} \tag{57}$$

If one makes no restrictions on the rank of **D**, zero value eigenvalues may occur. To show this, partition the eigenvalue and eigenvector matrices each into two parts naming the partitions, renaming the eigenvalue matrix and naming a partition.

$$D\,V = \Sigma^2\,V$$

$$V = [\,V_{\lambda^2} \;\vdots\; V_0\,];\quad \Sigma^2 = \begin{bmatrix} \lambda_1^2 & 0 & 0 \\ 0 & \lambda_m^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} \lambda^2 & 0 \\ 0 & 0 \end{bmatrix} \tag{58}$$

where the λ^2 matrix is of dimension m by m, m being the rank of **D**.

Now set **D** equal to $\mathbf{C}^T\mathbf{C}$, thus **D** is of dimension n by n and

$$V^T C^T C V = \Sigma^2_{C^TC} \tag{59}$$

V and Σ are also of dimension n by n and

$$\left[\Sigma^{-1}_{C^TC} V^T C^T \right] \left[C V \Sigma^{-1}_{C^TC} \right] = I \tag{60}$$

The bracketed terms are transposes of each other with a unit product making them matrices of orthonormal vectors. The matrices are named **U** and its transpose, $\mathbf{U}^T$ and are of dimension r by r. Thus,

$$\begin{aligned} U &= C V \Sigma^{-1}_{C^TC} \\ C &= U \Sigma_{C^TC} V^T \end{aligned} \tag{61}$$

Noting that the following reduction is true,

$$\left[U \Sigma_{C^TC} V^T \right] \left[V \Sigma^{-1}_{C^TC} U^T \right] = I \tag{62}$$

allows one to define the inverse of **C** given the decomposition of Eq. (61). This is the left side pseudo-inverse of **C**.

$$C^{+L} = V \Sigma^{-1}_{C^TC} U^T \tag{63}$$

The eigenvectors $\mathbf{V}_\lambda$ are found using the non-zero eigenvalues of $\mathbf{C}^T\mathbf{C}$, or λ^2, as in a standard eigenvalue problem. Just as the eigenvector **V** is decomposed into two portions so is the eigenvector **U**. From Eq. (61) it is found that

$$U_\lambda = C V_\lambda \Sigma^{-1}_{C^TC} \tag{64}$$

The remaining portions of **V** and **U** are composed of orthonormal vectors that span the null spaces of **C** and $\mathbf{C}^T$ respectively. That is, those orthonormal vectors that satisfy the following conditions.

$$\begin{aligned} C\, V_\bullet &= 0 \\ C^T\, U_\bullet &= 0 \end{aligned} \tag{65}$$

If **D** had alternatively been set equal to $\mathbf{CC}^T$ and following along as above except switching the meaning of **U** and **V** one would have

$$\begin{aligned} U^T\, C\, C^T\, U &= \lambda^2_{CC^T} \\ \left[\, \Sigma^{-1}_{CC^T}\, U^T\, C \,\right]\left[\, C^T\, U\, \Sigma^{-1}_{CC^T} \,\right] &= I \\ V^T &= \Sigma^{-1}_{CC^T}\, U^T\, C \\ C &= U\, \Sigma_{CC^T}\, V^T \end{aligned} \tag{66}$$

This is exactly as in Eq. (61) with the exception that the non-zero eigenvalues utilized are of a different matrix and are associated with $\mathbf{U}_\lambda$. Thus the right side pseudo-inverse becomes,

$$C^{+R} = V\, \Sigma^{-1}_{CC^T}\, U^T \tag{67}$$

Although the Moore-Penrose inverse can be obtained from these if desired, the left and right inverses is all that is required.

Application of Technique. The technique can now be applied to solving the dynamic system Eq. (53) in light of the constraint Eq. (13). First the generalized velocity consistent with the constraints, Eqs. (13), is determined by

$$\dot{q} = C^+ A + V_\bullet\, \dot{z} \tag{68}$$

for any vector $\dot{z}$. This is easily checked by substituting Eq. (68) into the first of Eqs. (13) and utilizing the first of Eq. (31). Similarly for the generalized accelerations to be consistent with the constraints, the last of Eqs. (13) it is found

$$\ddot{q} = C^{+} B + V_{\bullet} \ddot{z} \tag{69}$$

z is a reduced set of m=n-r independent coordinates. It is preferable to look at these as a form of quasi coordinates and the pseudo-inverse procedure discussed here as a means of removing the arbitrary nature of quasi-coordinate selection presented in Eq. (46) so that both independent and independent coordinates can be utilized. Through Eq. (69) it is found

$$\begin{aligned} \dot{q} &= C^{+} A + V_{\bullet} \dot{z} \\ \dot{z} &= V_{\bullet} \dot{q} - V_{\bullet} C^{+} A \end{aligned} \tag{70}$$

which is of the form given by Eq. (46) except that all the generalized coordinates are used.

Substitution of Eq. (69) into the system eqs, Eq. (53), yields

$$V_{\bullet}^{T} M V_{\bullet} \ddot{z} = V_{\bullet}^{T} Q + V_{\bullet}^{T} Q_{c} - V_{\bullet}^{T} M C^{+} B \tag{71}$$

Recalling from LaGrange's Equations, as in Eqs. (39) and (23), that

$$Q^{c} = C^{T} \lambda_{constraints} \tag{72}$$

and noting that

$$V_{\bullet}^{T} Q^{c} = V_{\bullet}^{T} C^{T} \lambda_{constraints} = [C \; V_{\bullet}]^{T} \lambda_{constraints} = 0 \tag{73}$$

the system equations reduce to a minimal set not containing the numerical problems found in other methods.

$$\begin{aligned} V_{\bullet}^{T} M V_{\bullet} \ddot{z} &= V_{\bullet}^{T} Q - V_{\bullet}^{T} M C^{+} B \\ \dot{q} &= C^{+} A + V_{\bullet} \dot{z} \end{aligned} \tag{74}$$

Results with similar form occur when applying the method directly to Kane's Eqs. [69]. The benefit of the modelling procedure being the automatic selection and grouping of generalized coordinates into a minimal set of pseudo coordinates that can, in principle, be used in numerical integrations in a highly stable manner. The mapping between pseudo-coordinates and generalized coordinates continually changes and since the rank of the constraint matrix will vary with location the number of pseudo-coordinates will similarly change.

Gauss's Principle of Least Constraint

An alternative to methods based on LaGrange's solutions to analytic mechanics are approaches based on Gauss's Principle of Least Constraint [23]. If a system is unconstrained as in Eq. (52) it has a solution

$$\ddot{q} = M^{-1} Q = a(q, \dot{q}, t) \tag{75}$$

If on the other hand the otherwise same system becomes constrained its system equations look like Eq. (53). In this derivation it is assumed that there are r consistent and not necessarily independent constraints of a form very similar but more general than that of Eqs. (13)

$$C(q, \dot{q}, t) \ddot{q} + B(q, \dot{q}, t) = 0 \tag{76}$$

where, **C** is a known r x n constraint matrix and **B** is a known r-vector then. This general class includes holonomic, non-holonomic, scleronomic, rheonomic, catastatic and acatastatic varieties as well as others. Gauss's principle states that the constrained motion accelerations are such that the Gaussian function, Γ, is minimized over all solutions with satisfy these constraints.

$$\Gamma = [\ddot{q} - a(q, \dot{q}, t)]^T M [\ddot{q} - a(q, \dot{q}, t)] \tag{77}$$

Udwadia [70, 71], through the use of the Moore-Penrose generalized inverse, has been able to provide an interesting solution to the allowable motions. Utilizing a matrix K, where

$$K(q, \dot{q}, t) = M^{\frac{1}{2}} \left(C M^{-\frac{1}{2}} \right)^{+} \tag{78}$$

and + indicates any of the above discussed generalized inverse, Udwadia determined that the constraint force could be determined as

$$Q_c (q, \dot{q}, t) = - K (B + C M^{-1} Q) \tag{79}$$

The simplicity of these equations belies the fact that in some cases that the pseudo inverse may be difficult to obtain, especially if a symbolic representation is desired. Refer to Golub and Kahan for numerical computational aspects and to Wolfram [72] for a discussion on Mathematica and symbolic processing aspects.

EXAMPLE APPLICATION

As examples, the utilization of the above techniques is demonstrated for a simple pendulum, an open kinematic chain, and two groupings of methods. In actuality the relative power of the methods would be better demonstrated in a combined analysis of the branch of the Planar Stewart Platform looked upon as either an open or a closed kinematic chain. As such it could be used to represent a cutting tool on a machine surface, an active force-controlled high speed cam follower, a portion of a car suspension, a robotic insertion device or other systems and their associated simulation and control. Inferences made from the simple pendulum problem onto these more complex systems should hold true in practice. The first grouping is associated with LaGrangian or Newton-Euler techniques with and without the use of LaGrange multipliers. The second technique is associated with Gauss's Integral of Constraint. The use of quasi-coordinates and constraint elimination through singular value decomposition is not demonstrated here.

The pendulum consists of two bodies, one fixed and moving as shown in Figure 6. The absolute pose of the moving body is described by three absolute coordinates, (x, y, θ), given that it is planar. Because of the simple nature of this system, the coordinate, θ, can also be viewed as a relative coordinate. A torque motor, a torsional spring and a dashpot at the pivot with the ground are added for generality, as well as disturbance forces near the end of the linkage.

Newton-Euler, LaGrange Methods

The kinematic constraints for the pendulum, written first in vector form, then in scalar and then in both scalar and matrix Pfaffin form become:

$$\boldsymbol{R}_{01} = 0 = \boldsymbol{R}_{1c} - l\,\hat{e}_r$$

$$\begin{aligned} x - l\sin\theta &= 0 \\ y = l\cos\theta &= 0 \end{aligned}$$

$$\begin{aligned} \dot{x} - l\,\dot{\theta}\cos\theta &= 0 \\ \dot{y} - l\,\dot{\theta}\sin\theta &= 0 \end{aligned}$$

$$\begin{bmatrix} 1 & 0 & -l\cos\theta \\ 0 & 1 & -l\sin\theta \end{bmatrix} \begin{Bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{Bmatrix} = \boldsymbol{C}\,\dot{\mathbf{v}} = \mathbf{0} \qquad \textbf{(80)}$$

$$\begin{bmatrix} 1 & 0 & -l\cos\theta \\ 0 & 1 & -l\sin\theta \end{bmatrix} \begin{Bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{\theta} \end{Bmatrix} + \begin{Bmatrix} l\,\dot{\theta}^2\sin\theta \\ -l\,\dot{\theta}^2\cos\theta \end{Bmatrix} = \boldsymbol{C}\,\ddot{\boldsymbol{q}} + \boldsymbol{B} = \mathbf{0}$$

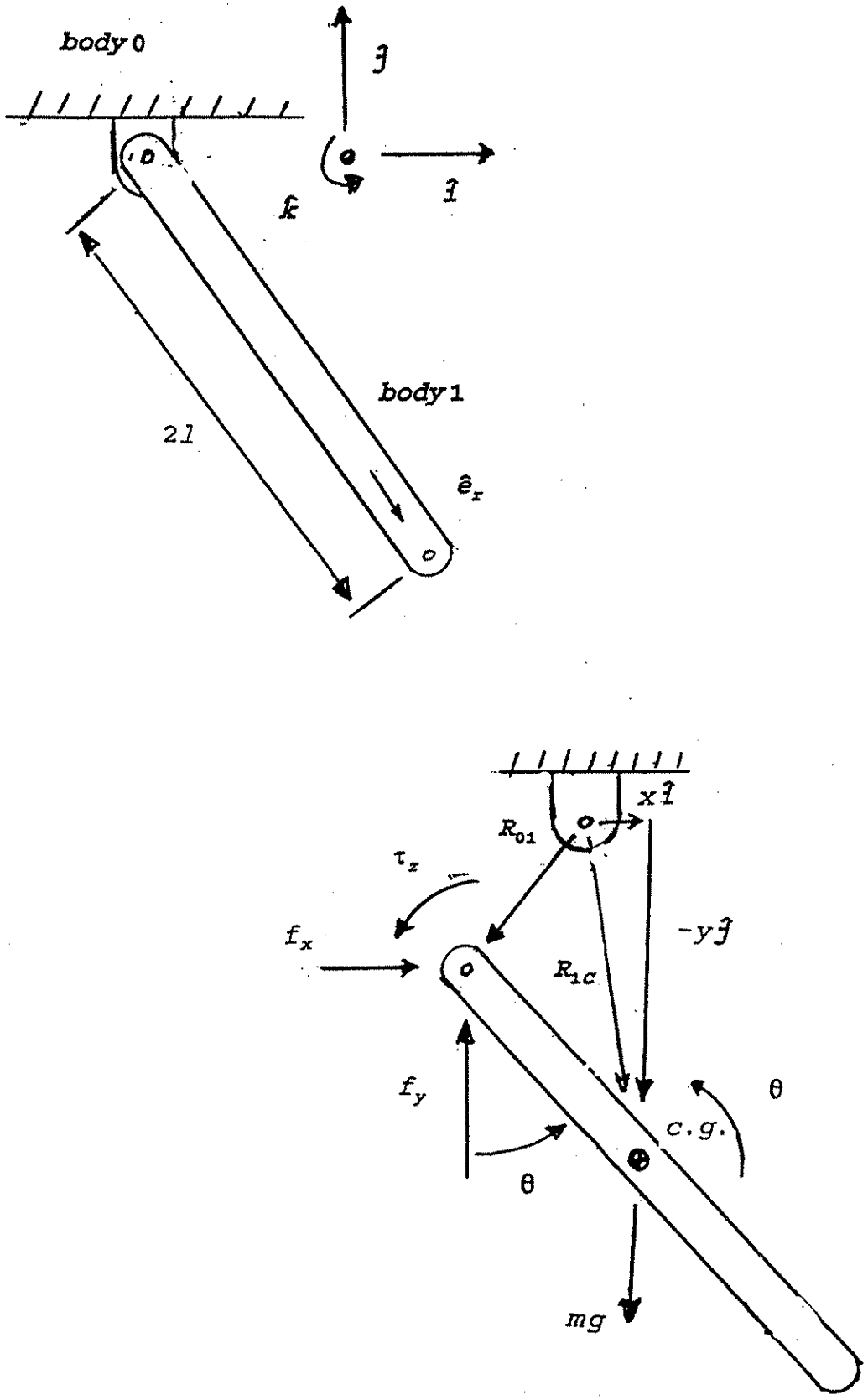


Fig. 6 Pendulum Example

The generalized constraint forces in terms of LaGrange multipliers become

$$Q^{con} = C^T \lambda = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -l\cos\theta & -l\sin\theta \end{bmatrix} \begin{Bmatrix} \lambda_1 \\ \lambda_2 \end{Bmatrix} = \begin{Bmatrix} \lambda_1 \\ \lambda_2 \\ -l\cos\theta\ \lambda_1 - l\sin\theta\ \lambda_2 \end{Bmatrix} \tag{81}$$

Also,

$$Q^{con} = \begin{bmatrix} \frac{\partial \dot{R}_{1c}}{\partial \dot{x}} & \frac{\partial \omega_{1c}}{\partial \dot{x}} \\ \frac{\partial \dot{R}_{1c}}{\partial \dot{y}} & \frac{\partial \omega_{1c}}{\partial \dot{y}} \\ \frac{\partial \dot{R}_{1c}}{\partial \dot{\theta}} & \frac{\partial \omega_{1c}}{\partial \dot{\theta}} \end{bmatrix} \cdot \begin{Bmatrix} f_{11} \\ \tau_{11} \end{Bmatrix} = \begin{Bmatrix} f_x \\ f_y \\ -f_x\, l\cos\theta - f_y\, l\sin\theta \end{Bmatrix} \tag{82}$$

Direct comparison of Eqs. (81) and (82) yields a simple relationship between constraint forces and LaGrange multipliers not characteristic of multibody systems.

$$\begin{aligned} f_x &= \lambda_1 \\ f_y &= \lambda_2 \end{aligned} \tag{83}$$

Noting that Q is the body force vector (which does work),

$$Q = \begin{Bmatrix} 0 \\ -mg \\ \tau \end{Bmatrix} \tag{84}$$

The equations of motion become

$$\begin{bmatrix} m & 0 & 0 & -1 & 0 \\ 0 & m & 0 & 0 & -1 \\ 0 & 0 & \frac{1}{3}ml^2 & l\cos\theta & l\sin\theta \\ -1 & 0 & l\cos\theta & 0 & 0 \\ 0 & -1 & l\sin\theta & 0 & 0 \end{bmatrix} \begin{Bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{\theta} \\ f_x \\ f_y \end{Bmatrix} = \begin{Bmatrix} 0 \\ -mg \\ \tau_a \\ l\dot{\theta}^2 \sin\theta \\ -l\dot{\theta}^2 \cos\theta \end{Bmatrix} \tag{85}$$

Although the constraint forces are available through simulation, they are not explicit functions of the state variables.

If one were to use coordinate partitioning or some other scheme, the equations of motion could be rewritten in other forms including the most obvious:

$$\frac{4}{3}ml^2\,\ddot{\theta} + lmg\,\sin\theta = \tau \tag{86}$$

A kineto-static analysis supplies the required solutions for the constraint forces. These forces are not functions of the state variables.

$$\begin{Bmatrix} f_x \\ f_y \end{Bmatrix} = \begin{Bmatrix} m\,\ddot{x} \\ m\,\ddot{y} + mg \end{Bmatrix} = \begin{Bmatrix} ml\ddot{\theta}\,\cos\theta - ml\dot{\theta}^2\,\sin\theta \\ ml\ddot{\theta}\,\sin\theta - ml\dot{\theta}^2\,\cos\theta + mg \end{Bmatrix} \tag{87}$$

Gauss's Principle of Least Constraint

Since most of the terms have been defined above all that is left is to show some of the intermediate results. Noting that

$$\boldsymbol{TMP} = \boldsymbol{C}\,\boldsymbol{M}^{-\frac{1}{2}} = \begin{bmatrix} 1 & 0 & -l\cos\theta \\ 0 & 1 & -l\sin\theta \end{bmatrix} \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & \frac{1}{3}ml^2 \end{bmatrix}^{-\frac{1}{2}} = \begin{bmatrix} \frac{1}{\sqrt{m}} & 0 & -\sqrt{\frac{3}{m}}\cos\theta \\ 0 & \frac{1}{\sqrt{m}} & -\sqrt{\frac{3}{m}}\sin\theta \end{bmatrix} \tag{88}$$

and that the left pseudo-inverse is singular, the right inverse is determined as:

$$\boldsymbol{TMP}^{+R} = \boldsymbol{TMP}^T\left[\boldsymbol{TMP}\;\boldsymbol{TMP}^T\right]^{-1} = \frac{\sqrt{m}}{4}\begin{bmatrix} 1+3\sin^2\theta & -3\sin\theta\cos\theta \\ -3\sin\theta\cos\theta & 1+3\sin^2\theta \\ -\sqrt{3}\cos\theta & -\sqrt{3}\sin\theta \end{bmatrix} \tag{89}$$

Thus

$$\boldsymbol{K} = \boldsymbol{M}^{-1}\left[\boldsymbol{C}\,\boldsymbol{M}^{-\frac{1}{2}}\right]^T = \frac{m}{4}\begin{bmatrix} 1+3\sin^2\theta & -3\sin\theta\cos\theta \\ -3\sin\theta\cos\theta & 1+3\sin^2\theta \\ -l\cos\theta & -l\sin\theta \end{bmatrix} \tag{90}$$

and the constraint forces become

$$Q^{con} = -K(B+CM^{-1}Q) = \begin{Bmatrix} -ml\dot{\theta}^2 + \frac{3\tau}{4l}\cos\theta - \frac{3}{4}mg\cos\theta\sin\theta \\ ml\dot{\theta}^2 + \frac{3\tau}{4l}\sin\theta + \frac{mg}{4}(1+3\cos^2\theta) \\ -\frac{3}{4}\tau - \frac{mgl}{4}\sin\theta \end{Bmatrix} \tag{91}$$

Finally, the system equations become

$$\begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & \frac{1}{3}ml^2 \end{bmatrix} \begin{Bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{\theta} \end{Bmatrix} = \begin{Bmatrix} -ml\dot{\theta}^2 + \frac{3\tau}{4l}\cos\theta - \frac{3}{4}mg\cos\theta\sin\theta \\ ml\dot{\theta}^2 + \frac{3\tau}{4l}\sin\theta - \frac{3}{4}mg\sin^2\theta \\ \frac{\tau}{4} - \frac{mgl}{4}\sin\theta \end{Bmatrix} \tag{92}$$

Comparisons may be made between Eq. (92) and those of Eq. (86) and (87). They give the same angular motion equations and quite different translational motion. Gauss's method describes both forces as functions of the state variable and its first derivative, whereas the Newton-Euler or LaGrange method also requires acceleration information.

In more complex linkages a similar situation occurs; Gauss's method determines the constraint and reaction forces as functions of the state variables and their first derivative whereas the Newton-Euler expressions for constraint and bearing forces include acceleration terms and are obtained either from an increased size in the system equations or an additional kineto-static analysis.

One strength of Gauss's method is the potential recasting of model-driven force-control algorithms. This provides a method to obtain explicit force models in terms of the state variables. Take, for example, the pendulum problem. Assume one wishes to maintain the vertical or x force on the pendulum to be less than some function. If one knows the current angular position and velocity, one would know the allowable torque and would also know the resulting angular acceleration.

SUMMARY

It has been shown elsewhere that many desired machine operations such as metal removal by lathes, mills or grinders and assembly operations perform better when their (internal) constraint forces are controlled.

The more robust nonlinear controllers tend to apply varied forms of model-driven controllers which essentially perform an adaptive exact linearization through forms of feedforward action. The generality of such schemes is made clearly evident by the direct analogy of the research practice of incorporating neural networks within learning systems to predict the behavior of a modelled system under various loadings. For control purposes, the resulting error between modelled and actual system performance, as measured by weighted deviations of the predicted variable from that actually obtained, is then controlled using spatial control strategies. In all such cases the pattern is clear; predict the variable to be controlled, measure the actual variable and control the difference. The robustness of the scheme comes down to the accuracy and speed of predicting the variable behavior by means of an empirical, analytical or neural network technique.

Dynamic system modelling most typically utilizes various forms of analytical or Newtonian dynamics to generate system equations. Often the number of generalized coordinates utilized is greater than the actual number of independent degrees of freedom and thus requires a number of constraint equations equal to this difference. The constraint equations are usually appended to the dynamical equations through utilization of LaGrange multipliers to form system equations. The multipliers are themselves equal to generalized forces composed of various combinations of the physical constraint forces. With proper selection of generalized coordinates, implicit solutions for the constraint forces as functions of time can be determined at a cost of numerical complexity. For example, if one wishes to determine all of the constraint forces in a system composed of n rigid bodies in 6 space with m interconnections so formed as to generate 6 n-1 constraint equations then a solution of a system of equations in generalized coordinates and LaGrange multipliers incorporating a 12 n-1 by 12 n-1 mass coupling matrix will be required. The solution for the constraint forces will most likely be numerical and will require inversion of the mass matrix at each time step within the integration in order to first find the multipliers and the coordinates. To determine the physical constraint forces, solution of a set of 6 n-1 equations linear in the constraints and multipliers and non-linear in the coordinates is then required. For many practical problems the system is stiff requiring Gear's method, coordinate partitioning, constraint violation stabilization, or combination of other procedure.

FURTHER WORK

Attention should be placed on using Gauss's Principle of Least Constraint to obtain explicit constraint force representations when these are needed. Explicit constraint force representation will allow for alternative non-linear control schemes to those currently employed.

Near term applications of this particular modelling and control structure to intelligent machinery is anticipated in active suspensions and automated assemblies. In the long term, benefits in active prosthetics, intelligent mechanical devices (in particular sensors/actuators), and in improved manufacturing process control for quality are anticipated.

Major tasks suggested for a near term research effort include:

- generalizing the modelling schemes attributed to Vereshchagin and to Udwadia for employing Gauss's Principle for Least Constraint (for ease of symbolic manipulation);
- determining the relationship of dynamic-system equations, using coordinate partitioning by means of SVD, to those determined from Gauss's Principle for Least Constraint.
- properly selecting generalized coordinates to facilitate rapid/accurate prediction schemes;
- determining the appropriate make-up of generalized coordinates within the constraint equations and the appropriate form of constraint equations for various types of lower-order and higher-order mechanical connections;
- selecting and developing the appropriate numerical integration techniques and strategies for equation solutions;
- and, making comparisons for speed, accuracy and simplicity, with results attributed to more classical approaches.

Major tasks suggested for long term research effort include:

- developing a force (impedance) responsive virtual reality system incorporating the imaging and vision processing.

Future research efforts could be centered on applying the technique to various linkage designs including parallel linkages such as Stewart Platforms, cable driven systems, systems of chain elements, intelligent active suspension systems, high speed cam-followers, precision linkages and biomedical applications.

REFERENCES

1. Stewart, D., "A Platform with Six Degrees of Freedom," ***Proc. Inst. Mech. Engr.***, Vol. 180, Pt.1, No. 15, 1965, pp. 371-386.

2. ***DADS User's Manual, Rev. 5.0***, Computer Aided Design Software, Inc., Oakdale Iowa, 1988.

3. ***ADAMS User's Guide***, Mechanical Dynamics, Inc., 555 South Forest, Ann Arbor, Mich., 1981.

4. Uicker, J. J., ***Users's guide for IMP (Integrated Mechanisms Program): a problem oriented language for the computer-aided design and analysis of mechanisms***, 1973.

5. Chace, M. A. and Angell, J.C., ***User's Guide to DRAM (Dynamic Response of Articulated Machinery)***, Design Engineering Computer Aids Laboratory, Department of Mechanical Engineering, University of Michigan, 1973.

6. Chace, M. A. and Smith, D.A., "DAMN-- Digital Computer Program for the Dynamic Analysis of Generalized Mechanical Systems.", ***SAE Trans.***, Vol 80, pp 969-983, 1971.

7. Paul, B. and Hud G., "DYMAC-L User's Manual", ***MEAM Report No. 75-6***, Dept. of Mech Engng., Univ. of Penn, Phila., Sept. 1975.

8. Vander Voort, R. J., and Singh, R. P., "A Pointing Control System Simulation and Generic Model Generator for Multi-Body Flexible Structures," ***Proceedings of the Ninth DARPA Strategic and Space Symposium***, Monterey, Calif., Oct. 1983.

9. Paul, B., "Analytical Dynamics of Mechanisms - A Computer oriented Overview.", ***Mechanism and Machine Theory,*** Vol 10, 1975, pp 481-507.

10. Likins, P.W., "Analytical Dynamics and Nonrigid Spacecraft Simulation," ***NASA Technical Report 32-1593***, JPL, Pasadena California, July 1974.

11. Hemami, H., Jaswa, V.C., McGhee, R.B., "Some Alternative Formulations of Manipulator Dynamics for Computer Simulation Studies," ***Proc. of the 13th Allerton Conference on Circuit and System Theory***, University of Illinois, October, 1975.

12. Hooker, W.W. and Margulies, G., "The Dynamical Attitude Equations for an n-Body Satellite," ***Journal of the Astronautical Sciences***, Vol 12, No. 4, Winter 1965, pp. 123-128.

13. Hooker, W., "A Set of r Dynamical Attitude Equations for an Arbitrary n-Body Satellite Having r Rotational Degrees of Freedom," ***AIAA Journal***, Vol. 8, No. 7, July 1970.

14. Stephanenko, Y. and Vukobratovic, M., "Dynamics of articulated open-chain active mechanisms", ***Math. Biosci.***, Vol. 28, Nos. 1/2, 1976.

15. Orin, D.E., McGhee, R.B., Vukobratovic, M., and Hartoch, G., "Kinematic and Kinetic Analysis of Open-Chain Linkages Utilizing Newton-Euler Methods," ***Mathematical Biosciences***, Vol 43, No. 1/2 Feb 1979, pp.107-130.

16. Walker, N.W. and Orin, D.E., "Efficient Dynamic Computer Simulation of Robotic Mechanisms," ***Proc. of 1982 Joint Automatic Control Conference,*** Charlotsville, Virginia, June, 1981.

17. Luh, J.Y.S., Walker, N.W., Paul, R.P.C., "On-Line Computational Scheme for Mechanical Manipulators," ***Journal of Dynamic Systems, Measurement and Control,*** ASME, Vol. 102, June, 1980, pp. 69-76.

18. Orin, D. E. and McGhee, R.B., "Dynamic Computer Simulation of Robotic Mechanisms", ***Theory and Practice of Robots and Manipulators***, Warsaw, Poland, Sept. 1981.

19. Whitney, D. E., "The Mathematics of Coordinated Control of Prosthetic Arms and Manipulators," ***Journal of Dynamic Systems, Measurement, and Control***, ASME, Dec. 1972, 303-309.

20. Shimano, B. E., "The Kinematic Design and Force Control of Computer Controlled Manipulators", ***AIM 313***, Stanford Artificial Intelligence Laboratory, Stanford University,1978.

21. Luh, J. Y. S.; and Zheng, Yuan-Fang; "Computation of Input Generalized Forces for Robots with Closed Kinematic Chain Mechanism", ***J. of Robotics and Automation***, IEEE, Vol. RA-1, No. 2, June 1985, pp. 95-103.

22. Vereshchagin, A. F.; "Computer Simulation of the Dynamics of Complicated Mechanisms of Robot-Manipulators", ***Engineering Cybernetics***, Vol. 12, No. 6, Nov./Dec. 1974, pp. 65-70.

23. Hofrath & Gauss, C. F., "Uber ein Neues Allgemeines Grundgestz der Mechanik", ***J. Reine Angewandte Mathematik***, Vol 4, pp 232-235, 1829.

24. Featherstone, R., ***Robot Dynamics Algorithms***, Ph.D. Thesis, University of Edinburgh, 1884.

25. Lathrop, R., "Constrained (Closed Loop) Robot Simulation by Local Constraint Propagation", ***IEEE***, 1986.

26. Dix R. C. and Leham T. J., Simulation of the Dynamics of Machinery, ***J. Engng. for Industry***, ASME, Ser. B Vol. 94, 1972, pp. 433-438.

27. Andrews G. C. and Kesaven H. K., "The Vector Network Model: A New Approach to Vector Dynamics", ***Mechanism and Machine Theory***, Vol 10, No 1, 1975, pp. 57-75.

28. Do, W. Q. D. and Yang, D. C. H., "Inverse Dynamic Analysis and Simulation of a Platform Type of Robot." ***Journal of Robotic Systems***, Vol 5, No. 3, 1988. pp. 209-227.

29. Hunt, K. H., ***Kinematic Geometry of Mechanisms***, Oxford Press, Oxford, Great Britain, 1978.

30. Fichter, E. F., "A Stewart Platform Based Manipulator: General Theory and Practical Construction", ***International Journal of Robotics Research***, Vol. 5, No. 2, Summer 1986, pp. 157-182.

31. Paul, B. and Krajcinovic, D., "Computer Analysis of Machines with Planar Motion, Part I-Kinematics, Part 2-Dynamics", ***Journal of Applied Mechanics***, ASME, Sept 1970, pp. 697-702 & 703-712.

32. Litvin, F. L. and Tan, J., Determination of Bearing Reactions of Spatial Linkages and Manipulators", ***Journal of Mechanical Design***, ASME, Vol. 112, June 1990, pp. 168-174.

33. LaGrange, J. L., ***Mecanique Analytique,*** Paris: Mme Ve Courcier, 1811.

34. Pars, L. A., ***A Treatise on Analytical Dynamics***, Ox Bow Press, Connecticut, 1965.

35. Chace, M. A., "Analysis of the Time-Dependence of Multi-Freedom Mechanical Systems in Relative Coordinates", ***Journal of Engineering for Industry***, ASME, Feb 1967, pp 119-125.

36. Chace, M.A., and Bayazitoglu, Y.O., "Development and Application of a Generalized d'Alembert Force for Multi-freedom Mechanical Systems", ***Journal of Engineering for Industry***, ASME, Feb 1971, pp 317-327.

37. Uicker, J. J., "Dynamic Force Analysis of Spatial Linkage", ***Journal of Applied Mechanics***, ASME, June 1967, pp 418-424.

38. Denavit, J. and Hartenburg, R.S., "A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices," ***Journal of Applied Mechanics***, ASME, Vol. 22, 1955, pp. 215-221.

39. Chace, M.A., et al., "Formulation and Numerical Methods in the Computer Evaluation of Mechanical Dynamic Systems," ***International Federation for Theory of Machines and Mechanisms--Third World Congress***, Dubrovnik Kupari, Yugoslavia, Sept. 13-20, 1971.

40. Haug, E.J., Wehage, R., and Barman, N.C., "Design Sensitivity Analysis of Planar Mechanism and Machine Dynamics", ***Journal of Mechanical Design***, ASME, Vol 103, July 1981, pp. 560-570.

41. Nikravesh, P.E., and Chung, I.S., "Application of Euler Parameters to the Dynamic Analysis of Three-Dimensional Constrained Mechanical Systems", ***Journal of Mechanical Design***, ASME, October 1982, Vol. 104, pp. 785-791.

42. Wehage, R. A., and Haug, E.J., "Generalized Coordinate Partitioning for Dimension Reduction in Analysis of Constrained Systems", ***Journal of Mechanical Design***, ASME, Vol. 104, Jan 1982, pp. 247-255.

43. Orlandea, N., Chace, M.A., and Calahan, D.A., "A Sparsity-Oriented Approach to the Dynamic Analysis and Design of Mechanical Systems-Part 1 and Part 2," ***Journal of Engineering for Industry***, ASME, Vol. 99, August 1977, pp. 773-784.

44. Nikravesh, P. E., ***Computer-Aided Analysis of Mechanical System***, Prentice Hall, 1988.

45. Chang, C.O. and Nikravesh, P.E., "An Adaptive Constraint Violation Stabilization Method for Dynamic Analysis of Mechanical Systems", ***Journal of Mechanisms, Transmissions, and Automation in Design,*** ASME, Vol. 107, Dec. 1985, pp. 488-492.

46. Petzold, L., "Differential/Algebraic Equations are not ODEs", SIAM, ***J. Sci. Stat. Comput.***, Vol 3., No. 3, 1982, pp. 367-384.

47. Gear, C. W., and Petzold, L. R., 1984, "ODE Methods for the Solution of Differential/Algebraic Systems," ***J. Num. Anal.***, SIAM, Vol 21, No. 4, pp. 716-728.

48. Kahn, M.E. and Roth B., "The Near-Minimum-Time Control of Open-Loop Articulated Kinematic Chains," ***Journal of Dynamic Systems, Measurement, and Control,*** ASME, Sept 1971, pp. 164-172.

49. Kahn, M.E., "The Near-Minimum-Time Control of Open-Loop Articulated Kinematic Chains," ***AIM-106***, Computer Science Department, Stanford University, December 10, 1969.

50. Hollerbach, J. M., "A Recursive Lagrangian Formulation of Manipulator Dynamics and a Comparative Study of Dynamics Formulation Complexity", ***Transactions on Systems, Man, and Cybernetics***, IEEE, Vol SMC-10, No. 11, Nov. 1980, pp 730-735.

51. Modrey, J., "Analysis of Complex Kinematic Chains with Influence Coefficients", ***Journal of Applied Mechanics***, ASME, Series E, Vol 81, June 1957, pp. 184-188.

52. Benedict, C.E. and Teasar, D., "Model Formulations of Complex Mechanisms with Multiple Inputs: Part 1-Geometry, Part 2-The Dynamic Model", ***Journal of Mechanical Design***, ASME, Vol. 100, Oct. 1978, pp. 747-761.

53. Thomas, M., and Teasar, D., 1982, "Dynamic Modelling of Serial Manipulator Arms," ***Journal of Dynamic Systems, Measurement, and Control,*** ASME, Vol. 104, pp. 218-228.

54. Hudgens, J.C. and Teasar, D., "A Fully Parallel Six Degree-of-Freedom Micromanipulator: Kinematic and Dynamic Model", ***1988 ASME Design Conference-20th Biannual Mechanisms Conference***, Kissiminee, Florida, Sept 25-28, 1988, DE-Vol. 15-3, pp 29-37.

55. Goldstein, J.A., "Partial Differential Equations and Related Topics," Springer-Verlag, New York, 1975.

56. Kane, T.R., "Dynamics of Non-holonomic Systems," ***Journal of Applied Mechanics***, ASME Series E., Vol 83, 1961, pp. 574-578.

57. Kane, T. R. and Wang, C. F., "On the Derivation of Equations of Motion.", ***J. Soc. Indust. Appl. Math.***, Vol. 13, No. 2, June 1965, pp. 487-492.

58. Appell, Paul, 1902.

59. Gibbs, J. Willard, 1882.

60. Dirac, P.A.M., "Lectures on Quantum Mechanics", Belfer Graduate School of Science, Yeshiva University, New York, 1964.

61. Huston, R., and Passerello, C., "Multibody Structural Dynamics Including Translation Between the Bodies", ***Computers & Structures***, Vol. 12, pp. 713-720.

62. Haug, E. J., and Yen, J, "Implicit Numerical Integration of Constrained Equations of Motion Via Generalized Coordinate Partitioning," ***Journal of Mechanical Design***, Vol 114, ASME, June 1992, pp. 296-304.

63. Singh, R.P. and Likins, P.W., "Constraint Elimination in Dynamical Systems", ***Computational Structural Mechanics Workshop***, June 19-21, 1985.

64. Moore, E. H., "On the Reciprocal of the General Algebraic Matrix," Abstract, ***Bull. Am. Math. Soc.***, Vol 26, 1920, pp. 394-395.

65. Penrose, R., "A Generalized Inverse of Matrices", ***Proc. Camb. Phil. Soc.***, Vol. 51, 1955, pp. 406-413.

66. Golub, G. and Reinsch, C., "Singular Value Decomposition and Least Squares Solutions.", ***Numer. Math.***, Vol. 14, 1970, pp. 403-420.

67. Golub, G. and Kahan, W., "Calculating The Singular Values and Pseudo-inverse of a Matrix.", ***J. Numer Anal.*** SIAM, Ser. B, Vol 2, No. 2, 1965, pp. 205-223.

68. Klema, V. C. and Laub, A. J., "The Singular Value Decomposition: Its Computation and Some Applications.", ***Trans. on Automatic Control***, IEEE, Vol. AC-25, No. 2, April 1980, pp. 164-176.

69. Singh, R. P., and Likins, P.W., "Singular Value Decomposition for Constrained Dynamical Systems", ***Journal of Applied Mechanics***, ASME, Vol. 52, No. 4, pp. 943-48, 1987.

70. Udwadia, F. E. and Kalaba, R. E., "A New Perspective on Constrained Motion.", ***Proc. R. Soc. London*** Series A, 1992, pp 407-410.

71. Udwadia, F. E. and Kalaba, R. E., "Lagrangian Mechanics, Gauss's Principle, Quadratic Programming and Generalized Inverses: New Equations of Motion of Non-holonomically Constrained Discrete Mechanical Systems.", ***Q. Appl. Math.*** (in the press)

72. Wolfram, S., ***Mathematica-A System for Doing Mathematics by Computer***, 2nd Ed. Addison-Wesley, Redwood City, Ca, 1991.