Proceedings of DETC'00
ASME 2000 Design Engineering Technical Conferences
and Computers and Information in Engineering Conferences
Baltimore, Maryland, September 10-13, 2000

DETC00/MECH-14067

CONFIGURATION ESTIMATION OF GOUGH-STEWART PLATFORMS USING EXTENDED KALMAN FILTERING

Ernest D. Fasse
Dept. of Aerospace and Mechanical Engineering
The University of Arizona
Tucson, Arizona

Albert J. Wavering
Manufacturing Engineering Laboratory
National Institute of Standards and Technology
Gaithersburg, Maryland

ABSTRACT

This paper develops extended Kalman filtering algorithms for a generic Gough-Stewart platform assuming realistically available sensors such as length sensors, rate gyroscopes, and accelerometers. The basic idea is to extend existing methods for satellite attitude estimation. The nondeterministic methods are meant to be a practical alternative to existing iterative, deterministic methods for real-time estimation of platform configuration.

1 INTRODUCTION

Parallel mechanisms were introduced by Gough and Whitehall in tire-testing equipment (Gough and Whitehall 1962). Later, Stewart (1965) proposed to use a parallel mechanism as a motion base for a flight simulator. This architecture, which is referred to as the "Gough-Stewart platform", can now be found in virtually all modern flight simulators. Parallel mechanisms have also been used in a number of other applications (Hunt 1978; Fichter 1986; Merlet 1994), including robotics.

The direct kinematic problem is the problem of computing the rigid body configuration (position and orientation) of the mobile platform given measurements of articular sensors. The inverse kinematic problem is the problem of computing the articular configurations given a rigid body configuration of the platform. For parallel robots the inverse kinematic problem is straightforward. The direct kinematic problem is difficult. Except for very special cases, e.g. (Bruyninckx 1997b; Bruyninckx 1997a), analytic solutions do not exist. For practical problems one solution approach is to add additional sensors to the links (Tancredi, Teillaud, and Merlet 1995).

It is known with certainty in the planar case that all direct kinematics problems can be transformed to polynomial root-finding problems. This has proven true for all spatial mechanisms that have been analyzed. The equations are typically "horribly complex" (Merlet 1989b). Iterative numerical solution methods are required (Merlet 1989a). For the spatial case there are typically many different solutions.

In general, kinematic algorithms can be classified as deterministic or nondeterministic. Deterministic algorithms assume that all measured or otherwise assumed kinematic variables are known perfectly and that unknown kinematic variables are to be determined exactly. This type of analysis is centuries old and is generally what one associates with "kinematic analysis". Nondeterministic algorithms assume that measurements are imperfect and perhaps insufficient in number to determine exactly the unknown kinematic variables. At any instant one has an estimate of the kinematic variables and the accuracy of the estimate. This estimate is propagated through time and updated as measurements are available.

Nondeterministic algorithms are recent in origin. The best known example of a nondeterministic estimator is the Kalman filter (Kalman 1960; Gelb 1974). Nonlinear extensions of Kalman filtering, in particular the so-called "extended" Kalman filter, have been applied extensively to aerospace applications (Schmidt 1981). Elegant methods exist for satellite attitude estimation (Lefferts, Markley, and Shuster 1982).

A fundamental advantage of Kalman filtering and other nondeterministic methods is their ability to use a wide variety of sensor modalities and measurement frequencies. This is essential in space applications, where bearing measurements from different sensors like magnetometers and star trackers are intermittent and available at different rates. For robotics one might use both articular measurements, which are available frequently, and vision estimates, which are available infrequently. Using inertial sensors like miniature gyroscopes, or other tool motion sensors, one may be able to effectively track changes in the tool position that cannot be observed by articular sensors alone.

Software development is modular because information from each sensor is processed individually. If in practice a sensor fails then the estimate update rule for that sensor is simply not called. In this sense the software is fault tolerant. If the sensor configuration is changed, where a sensor is replaced by one or more other sensors, then only the relevant sensor update routines need be changed. No other kinematic routines need be changed. This makes it easier to modify and maintain software. This is a significant practical benefit and discussed in some detail in Sec. 10.

For parallel, spatial robots geometrical methods of configuration estimation are desirable. Much relevant work has been done in the area of satellite attitude estimation (Schmidt 1981; Lefferts, Markley, and Shuster 1982). Particularly relevant methods are described and extended in this paper. We are not aware of relevant work regarding parallel, spatial robots. Certainly Kalman filter methods have been applied to robot vehicle pose estimation and to a lesser extent manipulator pose estimation (de Graaf 1994). The contribution of this paper is to develop extended Kalman filtering algorithms for a generic Gough-Stewart platform assuming realistically available sensors such as length sensors, rate gyroscopes, and accelerometers. The basic idea is to extend existing methods for satellite attitude estimation. The nondeterministic methods are meant to be a practical alternative to existing iterative, deterministic methods for real-time estimation of platform configuration.

2 RELATED PRIOR WORK

Merlet compares seven methods of numerical, direct kinematic computation (Merlet 1989a). Broadly he classifies algorithms as being either "iterative" or "least squares". First consider iterative algorithms. Let X be a set of variables representing the rigid body configuration of the mobile platform. Let ρ be a set of articular configuration variables (e.g., link lengths). Let $J(X)$ be a Jacobian matrix relating infinitesimal variations of the platform and articular variables, so that $\delta\rho = J(X)\delta X$. This is the so-called inverse Jacobian, which is easy to compute for parallel mechanisms. We are given the actual articular variables, ρ. Let $\hat{X}_k$ be the current, k-th estimate of the platform configuration. Let $\hat{\rho}_k$ be the corresponding set of articular variables, which is easily computed using the inverse kinematics. Let $\Delta X = \hat{X}_{k+1} - \hat{X}_k$ be the difference between the next and current estimates. To first order one requires that

$$(\rho - \hat{\rho}_k) = J(\hat{X}_k)\Delta X \tag{1}$$

This equation can be solved for ΔX, determining the next estimate $\hat{X}_{k+1}$. It is essentially a Newton-Raphson update rule. Merlet considers six different iterative rules, the differences of which are not important for this discussion.

In the least squares method the goal is to minimize the cost function

$$C(\hat{X}) = (\rho - \hat{\rho})^{\mathrm{T}} M (\rho - \hat{\rho}) \tag{2}$$

where M is a symmetric, positive-definite weighting matrix. This equation must be minimized numerically using, for example, steepest descent methods.

Merlet achieved good numerical results using iterative methods, including one geometrical method using the "kinematic Jacobian" ("geometric Jacobian"). Poor results were achieved using the least squares approach. Another approach using iterative polynomial solution was reported in (Merlet 1989b). This method is quite complicated. For the Gough-Stewart platform it requires solving a twentieth-degree polynomial. It has fundamentally better convergence properties than the iterative methods because of the robust numerical stability of polynomial solution methods. Still, it is more complex and was an order of magnitude slower than the iterative method using the kinematic Jacobian.

For all solution methods multiple solutions are a fundamental problem. The Gough-Stewart platform is known to have at most 40 solutions (Faugère and Lazard 1995; Lazard 1992; Raghavan 1991). Some examples are known to have 40 real solutions (Dietmaier 1998).

3 INTRODUCTION TO EXTENDED KALMAN FILTERING

Extended Kalman filtering is a well established technique for estimating the state of nonlinear systems. This section summarizes the continuous-discrete extended Kalman filter equations. It is based on the summary of (Lefferts, Markley, and Shuster 1982) and the extensive treatment of (Gelb 1974).

3.1 Model

The model of the dynamic system is of the form

$$\frac{d}{dt}x(t) = f(x(t),t) + g(x(t),t)w(t) \tag{3}$$

where $x(t)$is the system state and process noise $w(t)$ is a zero-mean, Gaussian white noise process. The mean and covariance are

$$E[w(t)] = 0 \text{ and } E[w(t)w^{\mathrm{T}}(t')] = Q(t)\delta(t-t') \tag{4}$$

where E denotes the expectation and $\delta(t-t')$ is nonzero only for $t = t'$, in which case $\delta(t-t') = 1$. The initial mean and covariance of the state are

$$E[x(t_0)] \equiv \hat{x}(t_0) = x_0 \tag{5}$$
$$E[(x(t_0)-x_0)(x(t_0)-x_0)^{\mathrm{T}}] \equiv P(t_0) = P_0 \tag{6}$$

3.2 Propagation

In between measurements the state and error covariance are propagated as follows. The estimate of state at any instant is defined by a conditional expectation:

$$\hat{x}(t) = E[x(t) \mid \hat{x}(t_0) = x_0] \tag{7}$$

The evolution of this estimate is governed by a differential equation

$$\frac{d}{dt}\hat{x}(t) = E[f(x(t),t)] \equiv \hat{f}(x(t),t) \approx f(\hat{x}(t),t) \tag{8}$$

The latter approximation is a defining feature of the extended Kalman filter.

The state error, and error covariance matrix are defined by

$$\Delta x(t) = x(t) - \hat{x}(t) \text{ and } P(t) = E[\Delta x(t)\Delta x^{\mathrm{T}}(t)] \tag{9}$$

The evolution of the error is governed by

$$\frac{d}{dt}\Delta x(t) = F(t)\Delta x(t) + G(t)w(t) \tag{10}$$

where

$$F(t) \equiv \frac{\partial}{\partial x} f(x,t)\big|_{\hat{x}(t)} \text{ and } G(t) \equiv g(\hat{x}(t),t) \tag{11}$$

The evolution of the error covariance matrix is governed by a Riccati equation:

$$\frac{d}{dt}P(t) = F(t)P(t) + P(T)F^{\mathrm{T}}(t) + G(t)Q(t)G^{\mathrm{T}}(t) \tag{12}$$

3.3 Measurement updates

Between measurements the state estimates and error covariance are propagated according to (8) and (12). When measurements are available the state estimates and error covariance are updated according to the following rules. Measurements are assumed to be a function of state

$$z_k = h(x_k) + v_k \tag{13}$$

where $x_k = x(t_k)$ and measurement noise v_k is a discrete, zero-mean, Gaussian white-noise process

$$E[v_k] = 0 \text{ and } E[v_k v_{t'}^{\mathrm{T}}] = \underline{R}_k \delta_{kk'} \tag{14}$$

where $\delta_{kk'}$ is nonzero only for $k = k'$, in which case $\delta_{kk'} = 1$. The measurement covariance matrix is underlined, $\underline{R}$, to distinguish it from the direction cosine matrix, R.

Let $\hat{x}_k(-)$ and $P_k(-)$ be the state estimate and error covariance immediately before measurement. Let $\hat{x}_k(+)$ and $P_k(+)$ be the state estimate and error covariance immediately following measurement. The minimum variance estimate of x_k is given by

$$\hat{x}_k(+) = \hat{x}_k(-) + K_k[z_k - h(\hat{x}_k(-))] \tag{15}$$

where K_k is the Kalman gain matrix defined by

$$K_k = P_k(-)H_k^{\mathrm{T}}[H_k P_k(-)H_k^{\mathrm{T}} + \underline{R}_k]^{-1} \tag{16}$$

and H_k is the measurement sensitivity matrix defined by

$$H_k \equiv \frac{\partial}{\partial x} h(x)\big|_{\hat{x}_k(-)} \tag{17}$$

The error covariance matrix is updated according to

$$P_k(+) = (I - K_k H_k)P_k(-)(I - K_k H_k)^{\mathrm{T}} + K_k \underline{R}_k K_k^{\mathrm{T}} \tag{18}$$

4 PLATFORM KINEMATICS

The Gough-Stewart platform is a spatial platform acted upon by six linear actuators, as depicted schematically in Fig. 1. The mobile platform and base are depicted as polyhedra. The base attachment point of actuator i is denoted by A_i. The platform attachment point of actuator i is denoted by B_i. The configuration shown is nearly singular.

Let p be the position of any convenient point on the platform. It is expressed in coordinates of the distinguished base

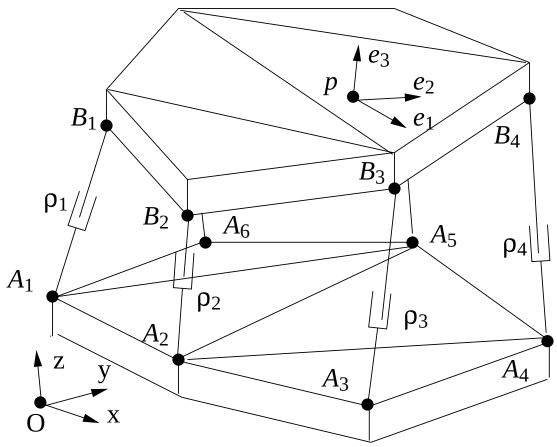

Figure 1. A Gough-Stewart platform

frame. Let e_1, e_2 and e_3 be an orthonormal triplet of unit vectors attached to the platform, again in coordinates of the base frame. Together these unit vectors define a direction cosine matrix $R = [e_1 e_2 e_3]$, which represents the orientation (attitude) of the platform. Matrix R corresponds to matrix A^{T} of Lefferts, Markley, and Shuster (1982). Orientation can also be expressed using Euler parameters (unit quaternions).

4.1 Euler parameters

This section is based primarily on reference (Nikravesh 1988). Rotation matrix R can be associated with an angle of rotation α and axis of rotation u. Let $e_0 = \cos(\alpha/2)$. Let $e = \sin(\alpha/2)u$. Let $q = [e_0; e]$ denote the Euler parameters (unit quaternion) corresponding to R. Scalar e_0 is the scalar part of q. Vector e is the vector part of q.

In general given Euclidean vector v let $\tilde{v} = (v)^{\sim}$ denote the cross-product matrix

$$\tilde{v} = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix} \tag{19}$$

Given Euler parameters q the corresponding rotation matrix is

$$R = \Gamma\Lambda^{\mathrm{T}} \tag{20}$$

where

$$\Gamma(q) = \begin{bmatrix} -e & \tilde{e} + e_0\mathrm{I} \end{bmatrix} \text{ and } \Lambda(q) = \begin{bmatrix} -e & -\tilde{e} + e_0\mathrm{I} \end{bmatrix} \tag{21}$$

In general given Euclidean vector v let matrices v^+ and v^- be defined by

$$v^+ = \begin{bmatrix} 0 & -v^{\mathrm{T}} \\ v & +\tilde{v} \end{bmatrix} \text{ and } v^- = \begin{bmatrix} 0 & -v^{\mathrm{T}} \\ v & -\tilde{v} \end{bmatrix} \tag{22}$$

These matrices have the useful properties that

$$\Gamma^{\mathrm{T}}(q)v = v^+ q \text{ and } \Lambda^{\mathrm{T}}(q)v = v^- q \tag{23}$$

4.2 Rate of change of platform configuration

Let ω be the angular velocity of the platform in platform coordinates. Then the rate of change of the Euler parameters is

$$\dot{q} = \frac{1}{2}\Lambda^{\mathrm{T}}\omega = \frac{1}{2}\omega^- q \tag{24}$$

Let v be the linear velocity of platform point p in platform coordinates. Then the rate of change of p is

$$\dot{p} = Rv \tag{25}$$

5 RATE AND ACCELERATION SENSORS

Assume that the angular velocity of the platform is measured using a three-axis gyroscope. Such gyroscopes are common and relatively inexpensive. Assume that the linear acceleration of point p on the platform is measured using accelerometers. These sensors are not required for state estimation. The point is that they can be incorporated in the method.

Let z_{g} be the gyro measurement, with

$$z_{\mathrm{g}} = \omega + b_{\mathrm{g}} + w_{\mathrm{g1}} \tag{26}$$

where b_{g} is the gyro bias and w_{g1} is the gyro bias noise. The gyro bias noise is a zero-mean Gaussian process with

$$E[w_{\mathrm{g1}}(t)] = 0 \text{ and } E[w_{\mathrm{g1}}(t)w_{\mathrm{g1}}^{\mathrm{T}}(t')] = Q_{\mathrm{g1}}\delta(t-t') \tag{27}$$

The gyro bias is assumed to be driven by a Gaussian process

$$\frac{d}{dt}b_{\mathrm{g}} = w_{\mathrm{g2}} \tag{28}$$

where w_{g2} is the gyro ramp noise. The mean and covariance are assumed to be

$$E[w_{\mathrm{g2}}(t)] = 0 \text{ and } E[w_{\mathrm{g2}}(t)w_{\mathrm{g2}}^{\mathrm{T}}(t')] = Q_{\mathrm{g2}}\delta(t-t') \tag{29}$$

Let z_a be the accelerometer measurement, with

$$z_g = a + R^T g + b_a + w_{a1} \tag{30}$$

where a is the linear acceleration of point p in platform coordinates, g is the acceleration of gravity in base coordinates, b_a is the accelerometer bias, and w_{a1} is the accelerometer bias noise. The accelerometer bias noise is a zero-mean Gaussian process with

$$E[w_{a1}(t)] = 0 \text{ and } E[w_{a1}(t)w_{a1}^T(t')] = Q_{a1}\delta(t-t') \tag{31}$$

The accelerometer bias is assumed to be driven by a Gaussian process

$$\frac{d}{dt} b_a = w_{a2} \tag{32}$$

where w_{a2} is the accelerometer ramp noise. The mean and covariance are assumed to be

$$E[w_{a2}(t)] = 0 \text{ and } E[w_{a2}(t)w_{a2}^T(t')] = Q_{a2}\delta(t-t') \tag{33}$$

6 STATE EQUATIONS

6.1 General state equations

The state is defined to be $x = [q; p; v^p; b_g; b_a]$. The corresponding state equations are

$$\dot{q} = \frac{1}{2}\Lambda^T(z_g - b_g - w_{g1}) \tag{34}$$
$$\dot{p} = Rv^p \tag{35}$$
$$\dot{v}^p = z_a - R^T g - b_a - w_{a1} \tag{36}$$
$$\dot{b}_g = w_{g2} \tag{37}$$
$$\dot{b}_a = w_{a2} \tag{38}$$

The process noise is $w = [w_{g1}; w_{a1}; w_{g2}; w_{a2}]$. The associated covariance is

$$Q = \begin{bmatrix} Q_{g1} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & Q_{a1} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & Q_{g2} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & Q_{a2} \end{bmatrix} \tag{39}$$

The norm of any unit quaternion is constrained to be unity, $q^T q = 1$. From this it follows that $\Delta q^T q \approx 0$ and that $[\hat{q}; 0_{3\times1}; 0_{3\times1}; 0_{3\times1}; 0_{3\times1}]$ is a null vector of the error covariance matrix P. Maintaining this singularity is difficult in practice because of numerical errors. This problem is dealt with in Sec. 8 using a reduced error covariance matrix.

6.2 Linearized state equations

The linearized state equations are

$$F(t) = \begin{bmatrix} F_{\dot{q},q} & 0_{4\times3} & 0_{4\times3} & F_{\dot{q},b_g} & 0_{4\times3} \\ F_{\dot{p},q} & 0_{3\times3} & F_{\dot{p},v^p} & 0_{3\times3} & 0_{3\times3} \\ F_{\dot{v}^p,q} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & F_{\dot{v}^p,b_a} \\ 0_{3\times4} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times4} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \end{bmatrix} \tag{40}$$

where

$$F_{\dot{q},q} = \frac{1}{2}(z_g - \hat{b}_g)^- \tag{41}$$
$$F_{\dot{q},b_g} = -\frac{1}{2}\Lambda^T(\hat{q}) \tag{42}$$
$$F_{\dot{p},q} = 2\Gamma(\hat{q})(\hat{v}^p)^- + 2\hat{v}^p\hat{q}^T \tag{43}$$
$$F_{\dot{p},v^p} = R(\hat{q}) \tag{44}$$
$$F_{\dot{v}^p,q} = -2\Lambda(\hat{q})g^+ - 2g\hat{q}^T \tag{45}$$
$$F_{\dot{v}^p,b_a} = -\mathrm{I}_{e\times3} \tag{46}$$

$$G(t) = \begin{bmatrix} G_{\dot{q},w_{g1}} & 0_{4\times3} & 0_{4\times3} & 0_{4\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & G_{\dot{v}^p,w_{a1}} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & G_{\dot{b}_g,w_{g2}} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & G_{\dot{b}_a,w_{a2}} \end{bmatrix} \tag{47}$$

where

$$G_{\dot{q},w_{g1}} = -\frac{1}{2}\Lambda^T(\hat{q}) \tag{48}$$
$$G_{\dot{v}^p,w_{a1}} = -\mathrm{I}_{3\times3} \tag{49}$$
$$G_{\dot{b}_g,w_{g2}} = \mathrm{I}_{3\times3} \tag{50}$$
$$G_{\dot{b}_a,w_{a2}} = \mathrm{I}_{3\times3} \tag{51}$$

7 CONFIGURATION SENSORS

Two kinds of configuration sensors are considered, vector field sensors and articular sensors.

7.1 Vector field sensors

First we consider a sensor for which the sensor output is a function of a vector field. Let $v^0(\cdot)$ be a vector field expressed in base coordinates. This might be the geomagnetic field, an artificially generated magnetic field, or a directional field towards a beacon. Assume that the field is measured at point p on the platform. The vector field in platform coordinates is given by

$$v^{\mathrm{p}}(p) = R^{\mathrm{T}}(q)v^{0}(p) \tag{52}$$

Let $z = h(v^{\mathrm{p}})$ be the n-dimensional sensor output. We need to compute the corresponding measurement sensitivity matrix. To do this we use the identity

$$\delta(R^{\mathrm{T}})v = (2\Lambda v^{+} + 2vq^{\mathrm{T}})\delta q \tag{53}$$

Using this identity it follows that

$$\begin{aligned}\delta v^{\mathrm{p}} &= \delta(R^{\mathrm{T}})v^{0} + R^{\mathrm{T}}\delta(v^{0}) \\ &= (2\Lambda(v^{0})^{+} + 2v^{0}q^{\mathrm{T}})\delta q + R^{\mathrm{T}}\frac{\partial v^{0}}{\partial p}\delta p\end{aligned} \tag{54}$$

and

$$\delta h = \frac{\partial h}{\partial v^{\mathrm{p}}}(2\Lambda(v^{0})^{+} + 2v^{0}q^{\mathrm{T}}(-))\delta q + \frac{\partial h}{\partial v^{\mathrm{p}}}R^{\mathrm{T}}\frac{\partial v^{0}}{\partial p}\delta p \tag{55}$$

Let

$$H_{\mathrm{q}} = \frac{\partial h}{\partial v^{\mathrm{p}}}\Big|_{\hat{x}(-)}\left[2\Lambda(v^{0})^{+} + 2v^{0}q^{\mathrm{T}}\right]_{\hat{x}(-)} \tag{56}$$

$$H_{\mathrm{p}} = \frac{\partial h}{\partial v^{\mathrm{p}}}\Big|_{\hat{x}(-)}\left[R^{\mathrm{T}}\right]_{\hat{q}(-)}\frac{\partial v^{0}}{\partial p}\Big|_{\hat{p}(-)} \tag{57}$$

The measurement sensitivity matrix is then given by

$$H = \left[H_{\mathrm{q}}\ H_{\mathrm{p}}\ 0_{n\times 3}\ 0_{n\times 3}\ 0_{n\times 3}\right] \tag{58}$$

7.2 Articular sensors

Vector field sensors are common on satellites but uncommon on manipulators. Much more common are articular sensors, which sense displacement of manipulator articulations (joints). We assume that the sensor output can be expressed as a function of the position of a point on the platform. Let

$$p_i = p + Rp_i^{\mathrm{p}} \tag{59}$$

be the position of some point on the platform, so that p_i^{p} is constant. In particular this might be an actuator attachment point. Let $z = h(p_i)$ be the n-dimensional sensor output. Using analysis similar to that of the preceding section it can be shown that

$$\delta h = \frac{\partial h}{\partial p_i}(2\Gamma(p_i^{\mathrm{p}})^{-} + 2p_i^{\mathrm{p}}q^{\mathrm{T}}(-))\delta q + \frac{\partial h}{\partial p_i}\delta p \tag{60}$$

Let

$$H_{\mathrm{q}} = \frac{\partial h}{\partial p_i}\Big|_{\hat{x}(-)}\left[2\Gamma(p_i^{\mathrm{p}})^{-} + 2p_i^{\mathrm{p}}q^{\mathrm{T}}(-)\right]_{\hat{x}(-)} \tag{61}$$

$$H_{\mathrm{p}} = \frac{\partial h}{\partial p_i}\Big|_{\hat{x}(-)} \tag{62}$$

The measurement sensitivity matrix is then given by

$$H = \left[H_{\mathrm{q}}\ H_{\mathrm{p}}\ 0_{n\times 3}\ 0_{n\times 3}\ 0_{n\times 3}\right] \tag{63}$$

In particular consider a length (linear displacement) sensor. Let p_{B_i} be the location of the platform attachment point of linear actuator i. Let p_{A_i} be the location of the base attachment point of actuator i. The sensor output is assumed to be the length of the actuator, ρ_i, which is the distance between the two attachment points:

$$h = \rho_i = \|p_{\mathrm{B}_i} - p_{\mathrm{A}_i}\| = [(p_{\mathrm{B}_i} - p_{\mathrm{A}_i})^{\mathrm{T}}(p_{\mathrm{B}_i} - p_{\mathrm{A}_i})]^{0.5} \tag{64}$$

The matrix of partial derivatives with respect to p_{B_i} is

$$\begin{aligned}\frac{\partial h}{\partial p_{\mathrm{B}_i}} &= [(p_{\mathrm{B}_i} - p_{\mathrm{A}_i})^{\mathrm{T}}(p_{\mathrm{B}_i} - p_{\mathrm{A}_i})]^{-0.5}(p_{\mathrm{B}_i} - p_{\mathrm{A}_i})^{\mathrm{T}} \\ &= \frac{1}{h}(p_{\mathrm{B}_i} - p_{\mathrm{A}_i})^{\mathrm{T}}\end{aligned} \tag{65}$$

8 KALMAN FILTERING USING A REDUCED ERROR COVARIANCE MATRIX

One approach to maintaining singularity of the error covariance matrix is to use a reduced error covariance matrix. This is a straightforward extension of the method described in (Lefferts, Markley, and Shuster 1982). Again let Δq be the estimation error of the Euler parameters. If the estimation error is small then it is approximately given by

$$\Delta q \approx \Lambda^{\mathrm{T}}(\hat{q})\Delta\theta \tag{66}$$

where $\Delta\theta$ is a 3×1 angular error in platform coordinates. The state error is $\Delta x = [\Delta q; \Delta p; \Delta v^{\mathrm{p}}; \Delta b_{\mathrm{g}}; \Delta b_{\mathrm{a}}]$. Let the reduced state error be $\Delta x' = [\Delta\theta; \Delta p; \Delta v^{\mathrm{p}}; \Delta b_{\mathrm{g}}; \Delta b_{\mathrm{a}}]$. Define transformation matrix S as:

$$S = \begin{bmatrix} \Lambda^{\mathrm{T}}(\hat{q}) & 0_{4\times 12} \\ 0_{12\times 3} & \mathrm{I}_{12\times 12} \end{bmatrix} \tag{67}$$

Then $\Delta x = S\Delta x'$. The full error covariance is defined by $P = E[(\Delta x)(\Delta x)^{\mathrm{T}}]$. The reduced error covariance is defined by

$$P' = E[(\Delta x')(\Delta x')^{\mathrm{T}}] \tag{68}$$

The full error covariance matrix can be determined from the reduced error covariance matrix by

$$P = SP'S^{\mathrm{T}} \tag{69}$$

Matrix S is not orthonormal. (It is not even square.) Still it is true that $S^{\mathrm{T}}S = \mathrm{I}_{15\times 15}$, because in general $\Lambda\Lambda^{\mathrm{T}} = \mathrm{I}_{3\times 3}$. It follows that

$$P' = S^{\mathrm{T}}PS \tag{70}$$

It is not difficult to show that

$$S^{\mathrm{T}}\dot{S} = -\dot{S}^{\mathrm{T}}S = \Omega \tag{71}$$

where

$$\Omega = \begin{bmatrix} \frac{1}{2}\tilde{\omega} & 0_{3\times 12} \\ 0_{12\times 3} & 0_{12\times 12} \end{bmatrix} \tag{72}$$

The rate of change of the reduced covariance matrix is then

$$\begin{aligned} \dot{P}' &= \dot{S}^{\mathrm{T}}PS + S^{\mathrm{T}}P\dot{S} + S^{\mathrm{T}}\dot{P}S \\ &= -S^{\mathrm{T}}\dot{S}P' + P'S^{\mathrm{T}}\dot{S} + S^{\mathrm{T}}\dot{P}S \\ &= P'\Omega - \Omega P' + S^{\mathrm{T}}\dot{P}S \end{aligned} \tag{73}$$

Let

$$F' = S^{\mathrm{T}}FS - \Omega \tag{74}$$

$$G' = S^{\mathrm{T}}G \tag{75}$$

Using (12) and (73) it is then easy to show that

$$\dot{P}' = F'P' + P'(F')^{\mathrm{T}} + G'Q(G')^{\mathrm{T}} \tag{76}$$

This is the equation used to propagate the reduced error covariance matrix between measurements.

Next consider the problem of error covariance updates given measurements. Let

$$H' = HS \tag{77}$$

$$K' = P'(H')^{\mathrm{T}}(H'P'(H')^{\mathrm{T}} + \underline{R}) \tag{78}$$

The full and reduced Kalman gain matrices are related by

$$K = SK' \text{ and } K' = S^{\mathrm{T}}K \tag{79}$$

It is straightforward to show that the reduced error covariance update rule is

$$P'(+) = (\mathrm{I} - K'H')P'(-)(\mathrm{I} - K'H')^{\mathrm{T}} + K'\underline{R}(K')^{\mathrm{T}} \tag{80}$$

The state estimate is still updated according to (15) using the Kalman gain defined by (78) and (79).

9 EXAMPLE

This section presents a realistic example illustrating the application of the methods. The example is adapted from one of Fasse and Gosselin (1999). Suppose that the task is to grind an elliptical paraboloid, a common shape for lenses, mirrors, antennas, etc. The kinematics of this task are given in (Fasse and Gosselin 1999).

The assumed platform geometry is shown in Fig. 2. In (Fasse and Gosselin 1999) a special Minimal Symmetric Simplified Manipulator (MSSM) geometry was used for illustration. In this geometry both the base and mobile platform attachment points consist of three pairs of coincident points. Neither the methods described in (Fasse and Gosselin 1999) nor those described here assume any particular symmetries. Pairs of points are chosen to be nearly coincident, both to keep the figure uncluttered and to ensure that the manipulator does not go through any kinematic singularities.

The platform is drawn as a regular hexagonal prism. A spherical grinding tool is mounted to the platform. The elliptical paraboloid is to be ground from below.

The manipulator is controlled using a geometrical impedance controller. Perfect state estimates are assumed for control purposes. A Kalman filter-based state estimator runs simultaneously, but these estimates are not used for control. This was done to allow fair comparison of different estimation algorithms.

Interaction of the grinding tool with the paraboloid was modelled as a frictionless kinematic constraint between the tool tip and the paraboloid surface.

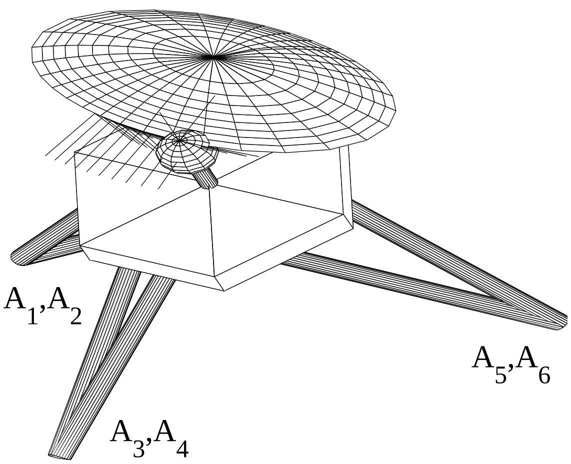

Figure 2. Final configuration of controlled platform

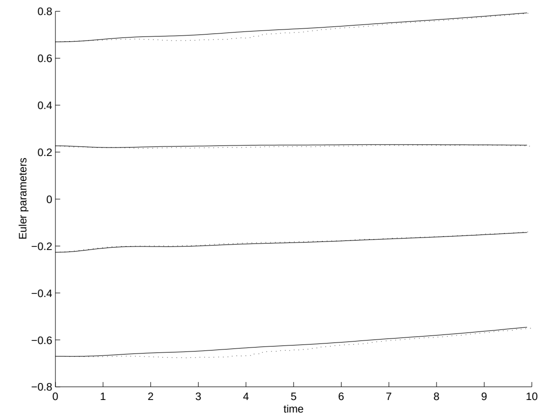

Figure 4. Actual (solid lines) and estimated (dashed lines) Euler parameters

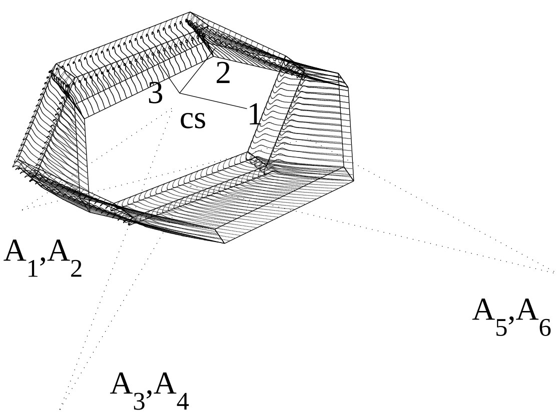

Figure 3. Trajectory of controlled platform using global potential function method

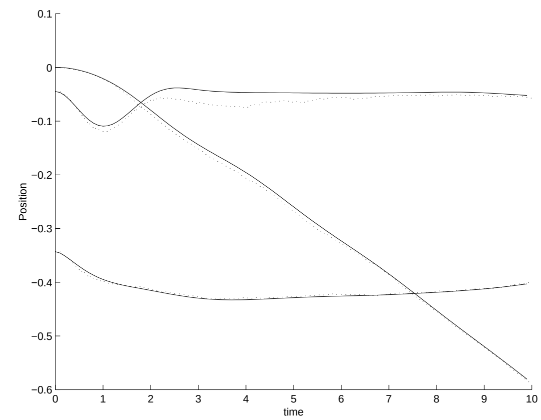

Figure 5. Actual (solid lines) and estimated (dashed lines) position

Figure 3 depicts a wire-frame animation of the platform. Only the platform is shown for clarity. The configurations of the six linear actuators are depicted only at the final configuration of the platform. The final configuration of the platform and actuators along with the paraboloid is shown more realistically in Fig. 2.

Results are shown in Figs. 4 and 5. The actual, simulated variables are graphed with solid lines. The estimated variables are graphed with dashed lines. The length measurements are assumed to have constant errors in addition to time-varying errors, as is realistic. Because of this perfect state estimation is impossible.

One of the features of Kalman filtering in general is that it can be used even when there is insufficient information for a deterministic measurement. Assume, for example, that lengths ρ_5 and ρ_6 are not available for measurement due to a data acquisition failure. It is still possible to estimate the platform configuration, although of course the accuracy of the estimate will be worse. Figure 6 shows an estimate of the position using only four measurements. A deterministic method could not be used in this situation. It would be just as easy to add redundant length measurements, or add measurements from different sensory modalities like vision. Deterministic methods (e.g., based on Newton-Raphson solution of the constraint equations) cannot be used given insufficient or redundant information.

10 MODULARITY OF SOFTWARE DEVELOPMENT

Modularity of software development is of enormous practical benefit as is apparent for satellite systems. A wide variety of bearing sensors are available for satellites including magnetome-

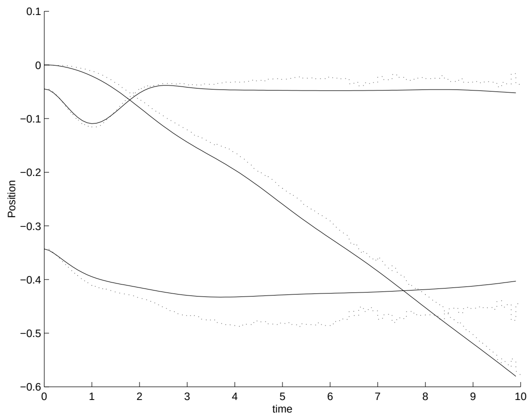

Figure 6. Actual (solid lines) and estimated (dashed lines) position using only four length measurements

ters, sun sensors, horizon sensors and star trackers. The sensors that are available vary from satellite to satellite. For a given satellite the measurements that are available at any instant depend on sensor health and environmental conditions. For example, if the earth occludes the sun then no sun sensor measurements will be available. It is thus of great advantage if not essential to have attitude estimation methods that process measurements individually rather than collectively.

For mechanisms it may not be essential to process measurements individually, but still it can be useful. Suppose that when a mechanism is new it is equipped with eight sensors of one type (e.g., angular encoders) and six sensors of a second type (e.g., length sensors). The following pseudo-code illustrates the possible organization of the configuration estimation code.

```
while(true) {
  [x,P] = propagate(x,P);
  for i=1:8 {
    if data_available(sensor_type1(i))
      [x,P] = update(x,P,sensor_type1(i));
    endif
  }
  for i=1:6 {
    if data_available(sensor_type2(i))
      [x,P] = update(x,P,sensor_type2(i));
    endif
  }
}
```

Functions `data_available` and `update` are assumed to be overloaded for each sensor type. Function `data_available` returns a boolean value depending on if the sensor is healthy and if data is available. If data is available then function `update` updates the state `x` and error covariance matrix `P` using the available data.

Suppose that after five years two of the sensors of the first type are broken and irreplaceable, and that three of the sensors of the second type are broken and prohibitively expensive to replace. Fortunately, two very sophisticated sensors of a third type (e.g., vision-based) have been added. The modified pseudo-code might look like the following:

```
while(true) {
  [x,P] = propagate(x,P);
  for i=1:6 {
    if data_available(sensor_type1(i))
      [x,P] = update(x,P,sensor_type1(i));
    endif
  }
  for i=1:3 {
    if data_available(sensor_type2(i))
      [x,P] = update(x,P,sensor_type2(i));
    endif
  }
  for i=1:2 {
    if data_available(sensor_type3(i))
      [x,P] = update(x,P,sensor_type3(i));
    endif
  }
}
```

The originally written code is unchanged except that there is no check to see if data is available from the broken sensors. Functions `data_available` and `update` have been overloaded for the third sensor type. This would require an in-depth analysis of the third sensor type, but it would not require re-analysis of the entire system.

11 CONCLUSION

Based on limited simulation results we draw the following simple conclusion: Nondeterministic methods can be used to estimate the configuration of parallel manipulators. Whether or not these methods are practical remains to be demonstrated experimentally. Because similar methods are used routinely in satellite attitude estimation we expect that the methods will be practical.

Perhaps the most significant expected practical benefit is modularity of software development and maintenance. Software development is modular because information from each sensor is processed individually. If in practice a sensor fails then the estimate update rule for that sensor is simply not called. In this sense the software is fault tolerant. If the sensor configuration is changed, where a sensor is replaced by one or more other sensors, then only the relevant sensor update routines need be changed. No other kinematic routines need be changed. This makes it easier to modify and maintain software.

ACKNOWLEDGEMENTS

The authors are grateful to the anonymous reviewers for their comments, particularly regarding the number of solutions of general Gough-Stewart platforms.

REFERENCES

Bruyninckx, H. (1997a). The 321-HEXA: A fully-parallel manipulator with closed-form position and velocity kinematics. In *Proc. IEEE ICRA*, pp. 2657–2662.

Bruyninckx, H. (1997b). The analytical forward displacement kinematics of the "31-1" parallel manipulator. In *Proc. IEEE ICRA*, pp. 2956–2960.

Dietmaier, P. (1998). The Stewart-Gough platform of general geometry can have 40 real postures. In *Proc. 6th Int. Symp. on Advances in Robot Kinematics*, pp. 7–16.

Fasse, E. and C. Gosselin (1999). Spatio-geometric impedance control of Gough-Stewart platforms. *IEEE Trans. on Robotics and Automation 15*, 281–288.

Faugère, J. and D. Lazard (1995). The combinatorial classes of parallel manipulators. *Mechanism and Machine Theory 30*, 765–776.

Fichter, E. (1986). A Stewart platform-based manipulator: general theory and practical construction. *Int. J. of Robotics Research 5-2*, 157–182.

Gelb, A. (Ed.) (1974). *Applied Optimal Estimation*. Cambridge, Massachusetts: MIT Press.

Gough, V. and S. Whitehall (1962). Universal tyre test machine. In *Proc. 9th Int. Tech. Congress FISITA*, pp. 117–137.

de Graaf, A. (1994). *On-line measuring systems for a mobile vehicle and a manipulator gripper: a model based approach*. Ph. D. thesis, University of Twente, Enschede, The Netherlands.

Hunt, K. (1978). *Kinematic geometry of mechanisms*. Oxford, United Kingdom: Oxford University Press.

Kalman, R. (1960). A new approach to linear filtering and prediction problems. *Trans. of the ASME, Series D, Journal of Basic Engineering 82*, 35–45.

Lazard, P. (1992). Stewart platform and gröbner basis. In *Proc. 3rd Int. Symp. on Advances in Robot Kinematics*, pp. 136–142.

Lefferts, E., F. Markley, and M. Shuster (1982). Kalman filtering for spacecraft attitude estimation. *J. Guidance 5*(5), 417–429.

Merlet, J.-P. (1989a). Parallel manipulators, part III: Applications. Technical Report 1003, Institut National de Recherche en Informatique et en Automatique (INRIA). In French.

Merlet, J.-P. (1989b). Parallel manipulators, Part IV: Assembly-modes and direct kinematics polynomial formulation. Technical Report 1135, Institut National de Recherche en Informatique et en Automatique (INRIA). In French.

Merlet, J.-P. (1994). Parallel manipulators: State of the art and perspectives. *Advanced Robotics 8*(6), 589–596.

Nikravesh, P. (1988). *Computer-Aided Analysis of Mechanical Systems*. Prentice Hall.

Raghavan, M. (1991). The stewart platform of general geometry has 40 configurations. In *ASME Design and Automation Conf.*, Volume 32-2, pp. 397–402.

Schmidt, S. (1981). The Kalman filter: Its recognition and development for aerospace applications. *J. of Guidance and Control 4*, 4–7.

Stewart, D. (1965). A platform with six degrees of freedom. In *Proc. Instn. Mech. Engrs.*, Volume 180-1(15), pp. 371–386.

Tancredi, L., M. Teillaud, and J.-P. Merlet (1995). Forward kinematics of a parallel manipulator with additional rotary sensors measuring the position of platform joints. In B. Ravani and J.-P. Merlet (Eds.), *Computational Kinematics*, pp. 261–270. Kluwer.