Chen, Yixin, Estimation of a Hexapod's Joint Space Mass-Inertia Matrix,
M.S., Department of Electrical Engineering, December, 1999.

The joint space mass-inertia matrix plays a very important role in the vibration isolation and pointing control of the hexapod. Although it can be calculated from the system parameters of the hexapod, in practice the calculation is laborious and can introduce errors. However, it can also be estimated using the measurements of the payload accelerations and base forces. The estimation problem is equivalent to solving an overdetermined set of linear equations $\mathbf{AX} = \mathbf{B}$ where $\mathbf{A}$, $\mathbf{B}$ are matrices of measurements. The main subtlety here is that $\mathbf{X}$ must be symmetric and positive definite. The least squares based symmetric procrustes method employs the symmetry constraint. But the definiteness of the estimate is not guaranteed. And one of the data matrices $\mathbf{A}$ or $\mathbf{B}$ should be "error" free. The Total Least Squares method can handle the case when "errors" exists in both $\mathbf{A}$ and $\mathbf{B}$. But neither the symmetry nor the positive definite constraints can be embedded into the algorithm. A new method is proposed which can directly take into account both the symmetry and the positive definite constraints. The new method is experimentally compared to the other two methods. Numerical experimental results indicate that the new approach is practical and gives a better estimate.

Contents

List of Figures

Chapter 1

UW's Flexure Jointed Hexapods

Several researchers have developed flexure jointed hexapods for micro-precision applications in which only a very small workspace is required ([1], [2], [3], [4], [5], [6], [7], [8]). UW's hexapods are flexure jointed hexapods. Figure 1.1 shows the structure of one of the UW's two flexure jointed hexapods.

Flexure jointed hexapods are great candidates for micro-precision applications, such as micro-manipulation, laser weapon pointing, optical communications, and remote sensing. They can provide simultaneous six degree-of-freedom active and passive vibration isolation and precision pointing. To avoid the extremely nonlinear micro-dynamics of joint friction and backlash, these hexapods employ flexure joints. A flexure joint bends material to achieve motion, rather than sliding or rolling across two surfaces. This does eliminate friction and backlash, but adds spring dynamics and limits the workspace.

Compared to non-flexure jointed hexapods, flexure jointed hexapods have several distinct characteristics [9]:

- The flexures greatly alter the dynamic behavior.

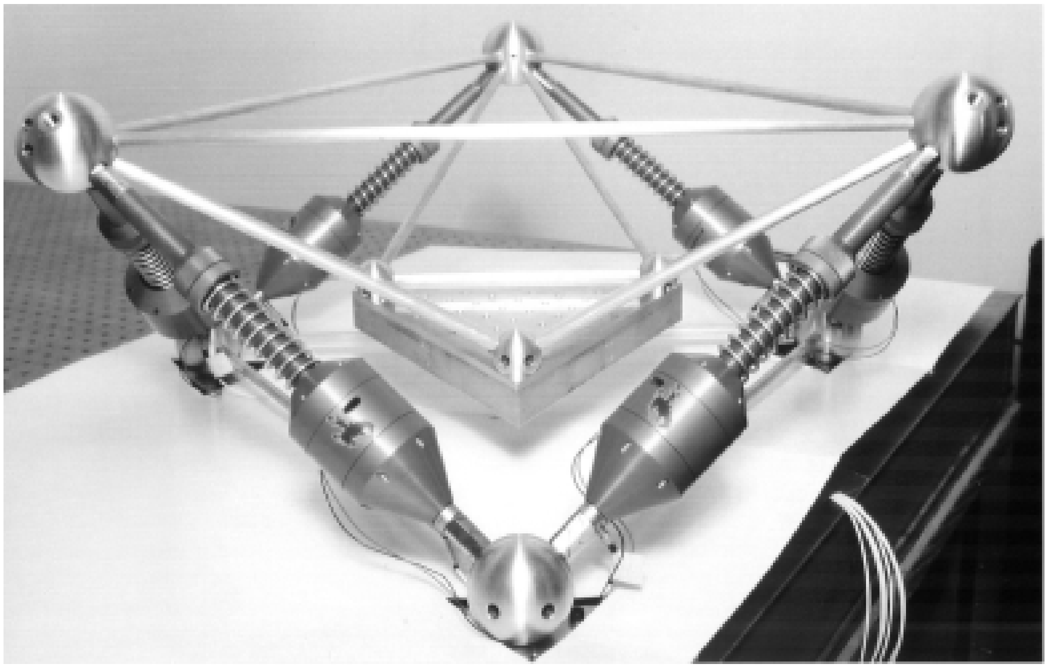

Figure 1.1: One of the UW's two flexure jointed hexapods.

- The base motion is a significant contributor to the overall motion, even when the base is subjected only to ambient seismic vibrations.

- Because the workspace is so small, linearized dynamic models are highly accurate.

Flexure jointed hexapods have been developed to meet two principle needs, depending on what is mounted to the hexapod "box" [10]. Figure 1.2 defines the two general problems. Generally, the "quiet box" problem uses payload acceleration, velocity, or position measurements to control the payload motion. The "dirty box" problem uses base force feedback to minimize the transmission of forces to the base.

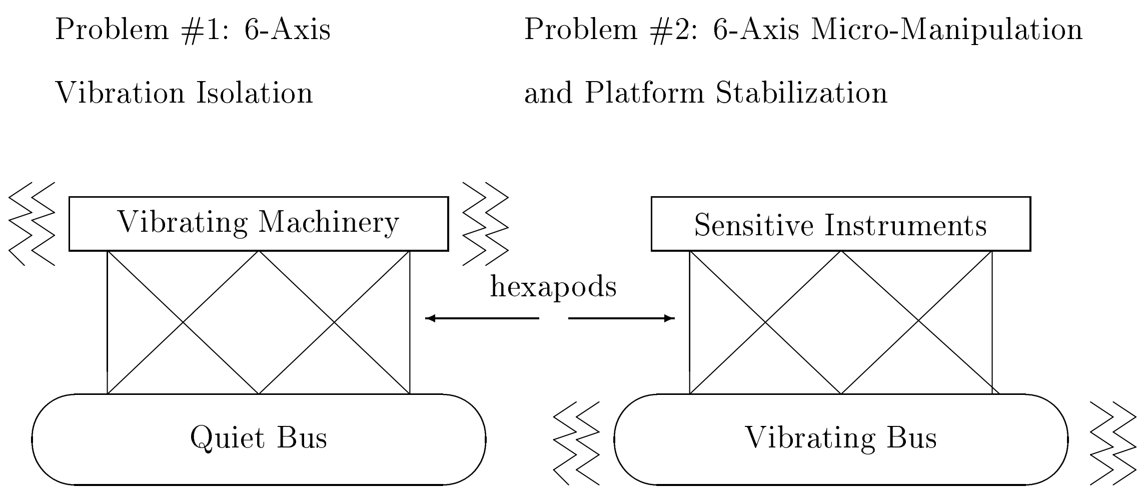

Figure 1.2: Problem #1: Vibrating machinery must be isolated from a precision bus (this is termed the *dirty box* problem because the machinery mounted on the hexapod "box" is mechanically "dirty", i.e. vibrating). Problem #2: A precision payload must be manipulated in the presence of base vibrations and/or exogenous forces (this is termed the *quiet box* problem).

Chapter 2

Dynamic Modeling of Flexure Jointed Hexapods

This chapter first summarizes the dynamic model of flexure jointed hexapods [10], then introduces the definition of the joint space mass-inertia matrix, and briefly describes its physically intrinsic properties.

2.1 Dynamic Model of Flexure Jointed Hexapods

Figure 2.1 illustrates a general flexure jointed hexapod. Like any hexapod, it consists of a base, a payload, and six struts that can change their lengths using the linear actuators inside them. The struts, which have spherical joints at both ends, connect the payload to the base.

In the joint space, the dynamics of a flexure jointed hexapod are written as [10]

$$\vec{f}_b = \vec{f}_m - \mathbf{K}(\vec{l} - \vec{l}_r) - \mathbf{B}\dot{\vec{l}} \tag{2.1}$$

$$\begin{aligned}({}^U_P\mathbf{R}\,{}^P\mathbf{M}_x\,{}^B_P\mathbf{R}^T\mathbf{J}^{-1} + {}^U_B\mathbf{R}\mathbf{J}^T\mathbf{M}_s)\ddot{\vec{l}} + {}^U_B\mathbf{R}\mathbf{J}^T\mathbf{B}\dot{\vec{l}} + {}^U_B\mathbf{R}\mathbf{J}^T\mathbf{K}(\vec{l} - \vec{l}_r) = \\ {}^U_B\mathbf{R}\mathbf{J}^T\vec{f}_m - ({}^U_B\mathbf{R}\mathbf{J}^T\mathbf{M}_s + {}^U_P\mathbf{R}\,{}^P\mathbf{M}_x\,{}^U_P\mathbf{R}^T\mathbf{J}_c\mathbf{J}_B{}^{-1})\ddot{\vec{q}}_s + \vec{\mathcal{F}}_e + \vec{\mathcal{G}} + \vec{\mathcal{C}}\end{aligned} \tag{2.2}$$

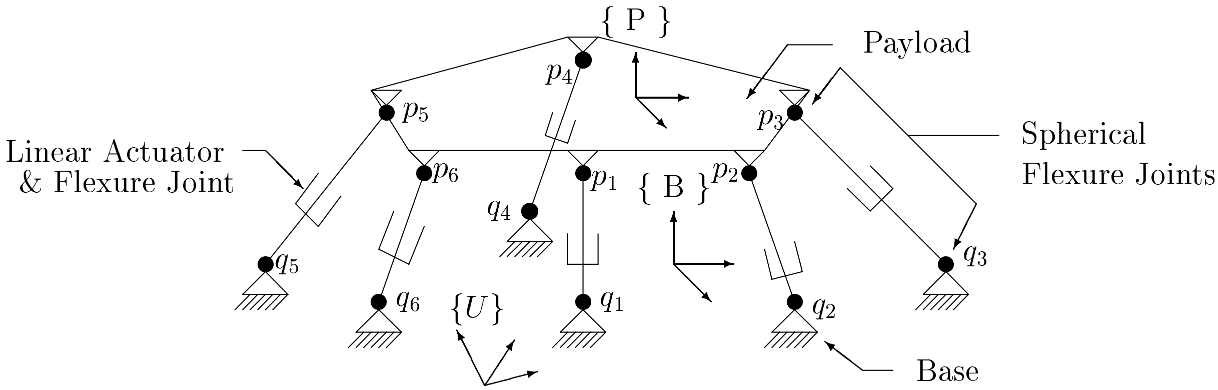

Figure 2.1: A flexure jointed hexapod (or Stewart Platform). $\{P\}$ is a Cartesian coordinate frame located at, and rigidly attached to, the payload's center of mass. $\{B\}$ is the frame attached to the (possibly moving) base, and $\{U\}$ is a Universal inertial frame of reference.

where

- $\mathbf{J}$ is the 6×6 hexapod Jacobian relating payload Cartesian movements, expressed in the $\{P\}$ frame, to strut length changes in the joint space,
- ${}^U_B\mathbf{R}$ is the 6×6 rotation matrix from the base frame, $\{B\}$, to the Universal inertial frame of reference $\{U\}$ (it consists of two identical 3×3 rotation matrices forming a block diagonal 6×6 matrix). Similarly, ${}^B_P\mathbf{R}$ is the rotation matrix from the payload frame to the base frame, and ${}^U_P\mathbf{R} = {}^U_B\mathbf{R}\,{}^B_P\mathbf{R}$,
- $\mathbf{J}_c$ and $\mathbf{J}_B$ are 6×6 Jacobian matrices capturing base motion,
- ${}^P\mathbf{M}_x$ is the 6×6 mass-inertia matrix of the payload, found with respect to the payload frame, $\{P\}$, whose origin is at the hexapod payload's center of mass,

- $\mathbf{M}_s$ is a diagonal 6×6 matrix containing the moving mass of each strut,
- $\mathbf{B}$ and $\mathbf{K}$ are 6×6 diagonal matrices containing the damping and stiffness, respectively, of each strut,
- $\vec{l}$ is the 6×1 vector of strut lengths, and $\vec{l_r}$ is the constant vector of relaxed strut lengths,
- $\vec{f_b}$ is the 6×1 vector of forces exerted at the bottom of the strut,
- $\vec{f_m}$ is the 6×1 vector of strut motor forces,
- $\ddot{\vec{q}}_s$ is a 6×1 vector of base accelerations along each strut plus some Coriolis terms,
- $\vec{\mathcal{F}_e}$ is a 6×1 vector of payload exogenous generalized forces applied at the origin of the $\{P\}$ frame,
- $\vec{\mathcal{C}}$ is a 6×1 vector containing all the Coriolis and centripetal terms,
- $\vec{\mathcal{G}}$ is a 6×1 vector containing all gravity terms.

Since the struts can only move very small distances, the Jacobian ($\mathbf{J}$) and the rotation matrix (${}^B_P\mathbf{R}$) can be considered constant, and Coriolis and Centripetal terms are often negligible.

Note that the base motions play a role in the dynamic model (2.2) both explicitly ($\ddot{\vec{q}}_s$, ${}^U_P\mathbf{R}$, and ${}^U_B\mathbf{R}$) and implicitly (through $\ddot{\vec{l}}$, $\dot{\vec{l}}$, and $\vec{l}$). The relations among payload motion, base motion and strut dynamics can be described as [10]

$$\vec{l} \;=\; \vec{p}_s - \vec{q}_s \tag{2.3}$$

$$\dot{\vec{l}} = \dot{\vec{p}}_s - \dot{\vec{q}}_s \tag{2.4}$$

$$\ddot{\vec{l}} = \ddot{\vec{p}}_s - \ddot{\vec{q}}_s \tag{2.5}$$

where $\vec{p}_s = [\vec{u}_1^T\vec{p}_1, ..., \vec{u}_6^T\vec{p}_6]^T$, $\vec{q}_s = [\vec{u}_1^T\vec{q}_1, ..., \vec{u}_6^T\vec{q}_6]^T$, $\vec{p}_i$ denotes the three dimensional attachment point of the i^{th} strut to the payload and $\vec{q}_i$ denotes the attachment point of the i^{th} strut to the base (Figure 2.1), $\vec{u}_i$ is the unit direction vector of the i^{th} strut ($\vec{p}_i$, $\vec{q}_i$, and $\vec{u}_i$ are expressed in the same coordinate frame).

Substituting (2.3-2.5) into (2.1-2.2) and rearranging terms produces the following dynamic equations

$$\vec{f}_b = \vec{f}_m - \mathbf{K}(\vec{p}_s - \vec{q}_s - \vec{l}_r) - \mathbf{B}(\dot{\vec{p}}_s - \dot{\vec{q}}_s) \tag{2.6}$$

$$\begin{aligned}({}^U_P\mathbf{R}\,{}^P\mathbf{M}_x\,{}^B_P\mathbf{R}^T\mathbf{J}^{-1} + {}^U_B\mathbf{R}\mathbf{J}^T\mathbf{M}_s)\ddot{\vec{p}}_s + {}^U_B\mathbf{R}\mathbf{J}^T\mathbf{B}\dot{\vec{p}}_s + {}^U_B\mathbf{R}\mathbf{J}^T\mathbf{K}\vec{p}_s = {}^U_B\mathbf{R}\mathbf{J}^T\vec{f}_m + \\ ({}^U_P\mathbf{R}\,{}^P\mathbf{M}_x\,{}^B_P\mathbf{R}^T\mathbf{J}^{-1} - {}^U_P\mathbf{R}\,{}^P\mathbf{M}_x\,{}^U_P\mathbf{R}^T\mathbf{J}_c\mathbf{J}_B{}^{-1})\ddot{\vec{q}}_s + {}^U_B\mathbf{R}\mathbf{J}^T\mathbf{B}\dot{\vec{q}}_s + {}^U_B\mathbf{R}\mathbf{J}^T\mathbf{K}(\vec{q}_s + \vec{l}_r) + \\ \vec{\mathcal{F}}_e + \vec{\mathcal{G}} + \vec{\mathcal{C}}\ .\end{aligned} \tag{2.7}$$

For the small movements possible in flexure jointed hexapods, $\mathbf{J}$, $\mathbf{J}_c$, $\mathbf{J}_B$, and ${}^B_P\mathbf{R}$ are all nearly constant. For small base motions, ${}^U_B\mathbf{R}$, ${}^U_P\mathbf{R}$, and $\vec{\mathcal{G}}$ are constant, while $\vec{\mathcal{C}}$ can be neglected because large velocities cannot be attained in the small distance moved. Large base motions can be treated by incorporating ${}^U_B\mathbf{R}$ and feeding forward $\vec{\mathcal{G}}$ and $\vec{\mathcal{C}}$ terms.

By letting the spring compression absorb the static gravity forces (for small base motions) or the static part of gravity forces (for the large base motions), both $\vec{l}_r$ and $\vec{\mathcal{G}}$

terms can be removed. Thus (2.7) can be written as the following equation

$$({}^U_P\mathbf{R}{}^P\mathbf{M}_x{}^B_P\mathbf{R}^T\mathbf{J}^{-1} + {}^U_B\mathbf{R}\mathbf{J}^T\mathbf{M}_s)\ddot{\vec{p}}_s + {}^U_B\mathbf{R}\mathbf{J}^T\mathbf{B}\dot{\vec{p}}_s + {}^U_B\mathbf{R}\mathbf{J}^T\mathbf{K}\vec{p}_s = {}^U_B\mathbf{R}\mathbf{J}^T\vec{f}_m +$$

$$({}^U_P\mathbf{R}{}^P\mathbf{M}_x{}^B_P\mathbf{R}^T\mathbf{J}^{-1} - {}^U_P\mathbf{R}{}^P\mathbf{M}_x{}^U_P\mathbf{R}^T\mathbf{J}_c{\mathbf{J}_B}^{-1})\ddot{\vec{q}}_s + {}^U_B\mathbf{R}\mathbf{J}^T\mathbf{B}\dot{\vec{q}}_s + {}^U_B\mathbf{R}\mathbf{J}^T\mathbf{K}\vec{q}_s + \vec{\mathcal{F}}_e + \Delta\vec{\mathcal{G}} \quad (2.8)$$

where $\Delta\vec{\mathcal{G}}$ is the dynamic part of gravity forces. For small base motions, $\Delta\vec{\mathcal{G}} = \vec{0}$.

Multiplying both sides of (2.8) by $\mathbf{J}^{-T}{}^U_B\mathbf{R}^T$, it can be written as

$$\mathbf{M}_p\ddot{\vec{p}}_s + \mathbf{B}\dot{\vec{p}}_s + \mathbf{K}\vec{p}_s = \vec{f}_m + \mathbf{M}_q\ddot{\vec{q}}_s + \mathbf{B}\dot{\vec{q}}_s + \mathbf{K}\vec{q}_s + \mathbf{J}^{-T}{}^U_B\mathbf{R}^T(\vec{\mathcal{F}}_e + \Delta\vec{\mathcal{G}}) \quad (2.9)$$

where

$$\mathbf{M}_p = \mathbf{J}^{-T}{}^B_P\mathbf{R}{}^P\mathbf{M}_x{}^B_P\mathbf{R}^T\mathbf{J}^{-1} + \mathbf{M}_s \quad (2.10)$$

$$\mathbf{M}_q = \mathbf{J}^{-T}{}^B_P\mathbf{R}{}^P\mathbf{M}_x{}^B_P\mathbf{R}^T\mathbf{J}^{-1} - \mathbf{J}^{-T}{}^B_P\mathbf{R}{}^P\mathbf{M}_x{}^U_P\mathbf{R}^T\mathbf{J}_c{\mathbf{J}_B}^{-1}. \quad (2.11)$$

2.2 Hexapod's Joint Space Mass-Inertia Matrix

The hexapod's joint space mass-inertia matrix $\mathbf{M}_p$ is defined as (2.10)

$$\mathbf{M}_p = \mathbf{J}^{-T}{}^B_P\mathbf{R}{}^P\mathbf{M}_x{}^B_P\mathbf{R}^T\mathbf{J}^{-1} + \mathbf{M}_s \quad (2.12)$$

where $\mathbf{J}$ is the hexapod Jacobian relating payload Cartesian movements to strut length changes in the joint space, ${}^B_P\mathbf{R}$ is the rotation matrix from the payload frame to the base frame, ${}^P\mathbf{M}_x$ is the mass-inertia matrix of the payload found with respect to the payload frame, and $\mathbf{M}_s$ is a diagonal matrix containing the moving mass of each strut. The payload's mass-inertia matrix, ${}^P\mathbf{M}_x$, consists of two blocks, one expressing mass and the other expressing inertial properties of the payload,

$$
\begin{aligned}
{}^{B}\mathbf{M}_x &= {}^{B}_{P}\mathbf{R}\,{}^{P}\mathbf{M}_x\,{}^{B}_{P}\mathbf{R}^T \\
&= {}^{B}_{P}\mathbf{R} \begin{bmatrix} m_p\mathbf{I}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & {}^{c}\mathbf{I} \end{bmatrix} {}^{B}_{P}\mathbf{R}^T \\
&= \begin{bmatrix} m_p\mathbf{I}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & {}^{B}_{P}\mathbf{R}\,{}^{c}\mathbf{I}\,{}^{B}_{P}\mathbf{R}^T \end{bmatrix}
\end{aligned} \tag{2.13}
$$

where m_p is the payload mass, and ${}^{c}\mathbf{I}$ is the symmetric inertia tensor [11] of the payload with respect to the payload frame $\{P\}$. The upper block, $m_p\mathbf{I}_{3\times 3}$, is always diagonal. The lower block, ${}^{B}_{P}\mathbf{R}\,{}^{c}\mathbf{I}\,{}^{B}_{P}\mathbf{R}^T$, is always symmetric.

The joint space mass-inertia matrix ($\mathbf{M}_p$) has two properties:

- It is symmetric.
- It is positive definite.

The first property comes directly from (2.12) and the fact that ${}^{c}\mathbf{I}$ is symmetric. The second property is not obvious. But it can be derived from the intrinsic physical properties of the ${}^{B}\mathbf{M}_x$ matrix. For any rigid body, there exist three orthogonal axes of symmetry with the payload mass distributed symmetrically about these axes. ${}^{B}\mathbf{M}_x$ is then diagonal with nonnegative diagonal entries if $\{B\}$ is selected to coincide with these axes. This implies that ${}^{P}\mathbf{M}_x$ is similar to a positive semi-definite diagonal matrix $\mathbf{D}$, i.e. there exists a rotation matrix ${}^{B}_{P}\mathbf{R}$, which is by definition unitary, such that ${}^{B}_{P}\mathbf{R}\,{}^{P}\mathbf{M}_x\,{}^{B}_{P}\mathbf{R}^T = \mathbf{D}$. The definiteness of ${}^{B}\mathbf{M}_x$ can also be derived from the energy point of view. The kinetic energy of the payload is $\frac{1}{2}\dot{\vec{\chi}}^T\, {}^B\mathbf{M}_x \dot{\vec{\chi}}$, where $\dot{\vec{\chi}}$ is the payload's Cartesian space velocity. The kinetic energy is always greater than or equal to zero. This implies that ${}^B\mathbf{M}_x$ is positive semi-definite. Since $\mathbf{M}_s$ is positive definite, $\mathbf{M}_p$ can be concluded to be positive definite from (2.12).

Substituting (2.6) into (2.9), assuming $\vec{l}_r$ and $\vec{\mathcal{G}}$ terms cancel each other, and solving for $\vec{f}_b$ gives

$$\vec{f}_b = \mathbf{M}_p \ddot{\vec{p}}_s - \mathbf{M}_q \ddot{\vec{q}}_s - \mathbf{J}^{-T}\, {}^U_B\mathbf{R}^T \vec{\mathcal{F}}_e \ . \tag{2.14}$$

where $\vec{f}_b$ is the 6×1 vector of forces exerted at the bottom of the strut, $\ddot{\vec{p}}_s$ is a 6×1 vector of payload accelerations along each strut, $\ddot{\vec{q}}_s$ is a 6×1 vector of base accelerations along each strut, and $\vec{\mathcal{F}}_e$ is a 6×1 vector of payload exogenous generalized forces applied at the origin of the $\{P\}$ frame.

It can be seen from (2.14) that the payload accelerations ($\ddot{\vec{p}}_s$) are related to the base forces ($\vec{f}_b$) by $\mathbf{M}_p$. (2.9) shows that the "modes" of the hexapod are determined by the eigenvalues of the matrix $\mathbf{M}_p{}^{-1}\mathbf{K}$. Thus $\mathbf{M}_p$ plays a very important role in the compensator design for the vibration isolation and pointing control of the flexure jointed hexapod [12].

Note that since the accelerometer can only measure the acceleration along the strut direction, the measured payload accelerations is not $\ddot{\vec{p}}_s$ but $\ddot{\vec{p}}_u$, where $\ddot{\vec{p}}_u$ is a vector of payload accelerations along each strut ($\ddot{\vec{p}}_u = [\vec{u}_1^T \ddot{\vec{p}}_1, \ldots, \vec{u}_6^T \ddot{\vec{p}}_6]^T$). The difference between $\ddot{\vec{p}}_s$ and $\ddot{\vec{p}}_u$ is that $\ddot{\vec{p}}_s = \ddot{\vec{p}}_u + \vec{\mathcal{C}}_1$, where $\vec{\mathcal{C}}_1 = [\dot{\vec{u}}_1^T \dot{\vec{p}}_1, \ldots, \dot{\vec{u}}_6^T \dot{\vec{p}}_6]^T$ is a vector of Coriolis terms. Compared with the magnitude of $\ddot{\vec{p}}_u$, $\vec{\mathcal{C}}_1$ is negligible.

Chapter 3

Estimation of the Joint Space Mass-Inertia Matrix Using the Least Squares Approach

There are two ways of obtaining the joint space mass-inertia matrix $\mathbf{M}_p$. One way is to calculate $\mathbf{M}_p$ from the design parameters of the hexapod. The other is to identify $\mathbf{M}_p$ from measurements. The former method requires exact values of $\mathbf{M}_s$, $\mathbf{M}_x$, and $\mathbf{J}$, which in practice is laborious and can introduce errors. This chapter first formulates the estimation problem, then summarizes a constrained least squares algorithm, proposed by Brock ([14]) and Higham ([15]), which can possibly be used to estimate $\mathbf{M}_p$. A different proof of the algorithm is also given. Finally, a method is introduced for computing the nearest symmetric positive semi-definite matrix (in the Frobenius norm) to an arbitrary real matrix [17]. The results derived in this chapter utilize the analytic qualities of convex sets and convex functions. Some additional background for these properties is included in Appendix A.

3.1 A Constrained Least Squares Problem

Equation (2.14) describes the relationships between the payload accelerations ($\ddot{\vec{p}}_s$) and the base forces ($\vec{f}_b$). If the base is kept stationary and there are no exogenous generalized forces exerted on the payload then $\ddot{\vec{q}}_s = \vec{0}$ and $\vec{\mathcal{F}}_e = \vec{0}$. And use the measured payload acceleration $\ddot{\vec{p}}_u$ instead of $\ddot{\vec{p}}_s$. Thus (2.14) becomes

$$\vec{f}_b = \mathbf{M}_p \ddot{\vec{p}}_u \ . \tag{3.1}$$

This implies that if both the base forces $\vec{f}_b$ and strut accelerations ($\ddot{\vec{p}}_u$) are measured, then $\mathbf{M}_p$ can be estimated. The main subtlety is that $\mathbf{M}_p$ is symmetric and positive definite.

Estimation of the $\mathbf{M}_p$ matrix fits into a class of constrained least squares approximation problems: Find

$$\min_{\mathbf{X} \in P} \|\mathbf{AX} - \mathbf{B}\|_F^2, \quad \mathbf{A}, \mathbf{B} \in \mathcal{R}^{m \times n}, \quad m \geq n, \tag{3.2}$$

where $P \subseteq \mathcal{R}^{n \times n}$, and $\|.\|_F$ denotes the Frobenius (or Euclidean) norm (2.3.2, [13]),

$$\|\mathbf{Y}\|_F = \left(\sum_{i,j} y_{ij}^2 \right)^{1/2} . \tag{3.3}$$

Equation (3.2) can also be written as

$$\min_{\mathbf{X} \in P} \text{Trace} \left\{ (\mathbf{AX} - \mathbf{B})^T (\mathbf{AX} - \mathbf{B}) \right\}, \quad \mathbf{A}, \mathbf{B} \in \mathcal{R}^{m \times n}, \quad m \geq n, \tag{3.4}$$

where $P \subseteq \mathcal{R}^{n \times n}$, and Trace$\{.\}$ denotes the trace of a matrix,

$$\text{Trace}\{\mathbf{Y}\} = \sum_i y_{ii}. \tag{3.5}$$

If $P = \mathcal{R}^{n \times n}$, then (3.2) and (3.4) becomes a standard unconstrained least squares problem, having a well-known solution $\mathbf{X} = \mathbf{A}^{+}\mathbf{B}$, where $\mathbf{A}^{+}$ is the pseudo-inverse of $\mathbf{A}$ [13]. If P is the set of all real symmetric matrices, then the problem is called the "symmetric procrustes problem" [15].

3.2 The Symmetric Procrustes Problem

The symmetric procrustes problem arises in the determination of the strain matrix of an elastic structure: find the symmetric matrix which minimizes the Frobenius (or Euclidean) norm of $\mathbf{AX} - \mathbf{B}$, where $\mathbf{A}$ and $\mathbf{B}$ are given rectangular matrices. In [14], Brock derives the normal equation of this problem. He claims that the solutions of the normal equation are local minimums. As this thesis will show, his proof in incomplete. In [15], Higham derives the same normal equation and the set of minimizers using singular value decomposition, and gives a sufficient condition to get a positive definite or positive semi-definite solution. This section gives a different proof of Brock and Higham's results.

The Symmetric Procrustes Problem can be stated as

$$\min_{\mathbf{X}=\mathbf{X}^T,\, \mathbf{X} \in \mathcal{R}^{n \times n}} \|\mathbf{AX} - \mathbf{B}\|_F^2, \quad \mathbf{A},\, \mathbf{B} \in \mathcal{R}^{m \times n}, \quad m \geq n, \tag{3.6}$$

which is equivalent to minimizing a scalar

$$\alpha = \text{Trace}\left\{(\mathbf{AX} - \mathbf{B})^T(\mathbf{AX} - \mathbf{B})\right\}. \tag{3.7}$$

Employing the fact that $\mathbf{X} = \mathbf{X}^T$, (3.7) can be written as

$$\alpha = \text{Trace}\left\{\mathbf{XA}^T\mathbf{AX} - \mathbf{XA}^T\mathbf{B} - \mathbf{B}^T\mathbf{AX} + \mathbf{B}^T\mathbf{B}\right\}. \tag{3.8}$$

Using the following known identities in matrix calculus

$$\begin{aligned}
\text{Trace}\{\mathbf{PQ}\} &= \text{Trace}\{\mathbf{QP}\}, &(3.9)\\
\text{Trace}\{\mathbf{P}\} + \text{Trace}\{\mathbf{Q}\} &= \text{Trace}\{\mathbf{P}+\mathbf{Q}\}, &(3.10)\\
\text{Trace}\{\mathbf{P}\} &= \text{Trace}\left\{\mathbf{P}^T\right\}, &(3.11)\\
\frac{\partial \text{Trace}\{\mathbf{P}\}}{\partial x} &= \text{Trace}\left\{\frac{\partial \mathbf{P}}{\partial x}\right\}, &(3.12)\\
\left[\frac{\partial \mathbf{P}}{\partial x}\right]^T &= \frac{\partial \mathbf{P}^T}{\partial x}, &(3.13)
\end{aligned}$$

where $\mathbf{P}, \mathbf{Q} \in \mathcal{R}^{n\times n},\ x \in \mathcal{R}$, then the partial derivative of α with respect to x_{ij} is derived as [14]

$$\begin{aligned}
\frac{\partial \alpha}{\partial x_{ij}} &= \text{Trace}\left\{\frac{\partial \mathbf{X}}{\partial x_{ij}}\mathbf{A}^T\mathbf{AX} + \mathbf{XA}^T\mathbf{A}\frac{\partial \mathbf{X}}{\partial x_{ij}} - \frac{\partial \mathbf{X}}{\partial x_{ij}}\mathbf{A}^T\mathbf{B} - \mathbf{BA}^T\frac{\partial \mathbf{X}}{\partial x_{ij}}\right\}\\
&= \text{Trace}\left\{\left[\mathbf{A}^T\mathbf{AX} + \mathbf{XA}^T\mathbf{A} - \mathbf{A}^T\mathbf{B} - \mathbf{BA}^T\right]\frac{\partial \mathbf{X}}{\partial x_{ij}}\right\}, \qquad (3.14)
\end{aligned}$$

where x_{ij} is the ijth entry of the $\mathbf{X}$ matrix. Thus, α is made stationary with respect to each element of $\mathbf{X}$ if $\mathbf{X}$ satisfies the equation

$$\mathbf{A}^T\mathbf{AX} + \mathbf{XA}^T\mathbf{A} = \mathbf{A}^T\mathbf{B} + \mathbf{B}^T\mathbf{A}. \qquad (3.15)$$

Equation (3.15) is the normal equation of the symmetric procrustes problem. It is a special form of Lyapunov equation.

The solutions of (3.15) are called critical points (stationary points). A critical point can be a minimizer, a maximizer, or a saddle point. The following new lemma and theorem prove that the critical points of α are minimizers.

Lemma 3.1 *If $f(\mathbf{X})$ is a real-valued function defined on $\mathcal{R}^{n\times n}$ by*

$$f(\mathbf{X}) = \text{Trace}\left\{(\mathbf{AX}-\mathbf{B})^T(\mathbf{AX}-\mathbf{B})\right\}, \tag{3.16}$$

where $\mathbf{A}$, $\mathbf{B} \in \mathcal{R}^{m\times n}$, then $f(\mathbf{X})$ is convex on $\mathcal{R}^{n\times n}$ and any convex subset of $\mathcal{R}^{n\times n}$. If $\text{Rank}(\mathbf{A}) = n$, *then $f(\mathbf{X})$ is strictly convex on $\mathcal{R}^{n\times n}$ and any convex subset of $\mathcal{R}^{n\times n}$.*

Proof: From Definition A.1 (Appendix A), it is easy to check that $\mathcal{R}^{n\times n}$ is convex.

Let $\mathbf{X} = [\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_n], \mathbf{B} = [\vec{b}_1, \vec{b}_2, \ldots, \vec{b}_n]$, where $\vec{x}_i \in \mathcal{R}^n, \vec{b}_i \in \mathcal{R}^m, i = 1, \ldots, n$. Then

$$\begin{aligned} f(\mathbf{X}) &= \text{Trace}\left\{(\mathbf{AX}-\mathbf{B})^T(\mathbf{AX}-\mathbf{B})\right\} \\ &= \sum_{i=1}^{n}\left[(\mathbf{A}\vec{x}_i-\vec{b}_i)^T(\mathbf{A}\vec{x}_i-\vec{b}_i)\right] \\ &= \left[\mathbf{A}_c\vec{x}-\vec{b}\right]^T\left[\mathbf{A}_c\vec{x}-\vec{b}\right] \\ &= \vec{x}^T{\mathbf{A}_c}^T\mathbf{A}_c\vec{x} - 2\vec{b}^T\mathbf{A}_c\vec{x} + \vec{b}^T\vec{b}, \end{aligned}$$

where $\mathbf{A}_c = I_{n\times n} \otimes \mathbf{A}$, $\otimes$ denotes Kronecker product, $\vec{x} = \left[\vec{x}_1^T, \vec{x}_2^T, \ldots, \vec{x}_n^T\right]^T$, and $\vec{b} = \left[\vec{b}_1^T, \vec{b}_2^T, \ldots, \vec{b}_n^T\right]^T$. So

$$\begin{aligned} \nabla f(\mathbf{X}) &= 2{\mathbf{A}_c}^T\mathbf{A}_c\vec{x} - 2{\mathbf{A}_c}^T\vec{b}, \\ Hf(\mathbf{X}) &= 2{\mathbf{A}_c}^T\mathbf{A}_c. \end{aligned}$$

Since $Hf(\mathbf{X})$ is always positive semi-definite, it follows from Theorem A.4 that $f(\mathbf{X})$ is convex on $\mathcal{R}^{n\times n}$. And from Definition A.2, $f(\mathbf{X})$ is convex on any subset of $\mathcal{R}^{n\times n}$. Moreover, if $\text{Rank}(\mathbf{A}) = n$, then $\text{Rank}(\mathbf{A}_c) = n^2$. So $Hf(\mathbf{X})$ is positive definite, which implies that $f(\mathbf{X})$ is strictly convex on $\mathcal{R}^{n\times n}$. And from Definition A.2, $f(\mathbf{X})$ is strictly convex on any open subset of $\mathcal{R}^{n\times n}$. $\square$

Theorem 3.2 *Define the set of minimizers of* (3.6) *as*

$$M = \left\{\mathbf{X} \in \mathcal{R}^{n\times n} : \mathbf{X} = \mathbf{X}^T \text{ and } f(\mathbf{X}) \leq f(\mathbf{Y}), \text{ for all } \mathbf{Y} = \mathbf{Y}^T, \mathbf{Y} \in \mathcal{R}^{n\times n}\right\}, \quad (3.17)$$

where

$$f(\mathbf{X}) = \text{Trace}\left\{(\mathbf{AX} - \mathbf{B})^T(\mathbf{AX} - \mathbf{B})\right\}.$$

then

1. $\mathbf{X} \in M$ *if and only if* $\mathbf{X} = \mathbf{X}^T$ and $\mathbf{A}^T\mathbf{AX} + \mathbf{XA}^T\mathbf{A} = \mathbf{A}^T\mathbf{B} + \mathbf{B}^T\mathbf{A}$.

2. *The minimizer is unique if and only if* $\text{rank}(\mathbf{A}) = n$.

3. *If* $\text{rank}(\mathbf{A}) = n$ *and* $\mathbf{A}^T\mathbf{B} + \mathbf{B}^T\mathbf{A}$ *is positive (semi-)definite then the unique minimizer* $\mathbf{X}_{LS}$ *is positive (semi-)definite.*

Proof: Define a set, S, of all symmetric matrices as

$$S = \left\{\mathbf{X} \in \mathcal{R}^{n\times n} : \mathbf{X} = \mathbf{X}^T\right\}$$

From Definition A.1, it is easy to check that S is convex. Since S is a subset of $\mathcal{R}^{n\times n}$, so from Lemma 3.1, $f(\mathbf{X})$ is convex on S. Then (1) follows from Corollary A.7 and Theorem A.8.

If $\text{rank}(\mathbf{A}) = n$, then from Lemma 3.1 $f(\mathbf{X})$ is strictly convex on S. From Theorem A.9 the minimizer is unique. And the normal equation (3.15) has a unique solution only if $\text{rank}(\mathbf{A}) = n$. This proves (2).

If $\text{rank}(\mathbf{A}) = n$, then $\mathbf{A}^T\mathbf{A}$ is positive definite, thus the unique solution of the normal equation (3.15) can be written as (p.82, [16])

$$\mathbf{X}_{LS} = \int_0^\infty e^{-\mathbf{A}^T\mathbf{A}t}(\mathbf{A}^T\mathbf{B} + \mathbf{B}^T\mathbf{A})e^{-\mathbf{A}^T\mathbf{A}t}dt, \tag{3.18}$$

which is positive (semi-)definite. This proves (3).□

The last property gives a sufficient condition for the definiteness of $\mathbf{X}_{LS}$. Note that the condition is not necessary. An example in [14] shows that $\mathbf{X}_{LS}$ can be positive definite when $\mathbf{A}^T\mathbf{B} + \mathbf{B}^T\mathbf{A}$ is indefinite. All the above properties can also be proved using a singular value decomposition approach [15].

3.3 Frobenius Norm Positive Approximation

Last section presents a method to find a symmetric matrix $\mathbf{X}$ which "best" describes the linear relationships between $\mathbf{A}$ and $\mathbf{B}$. Here "best" denotes minimizing the Frobenius norm of $\mathbf{AX} - \mathbf{B}$, where $\mathbf{X}$ is positive (semi-)definite if $\mathbf{A}^T\mathbf{B} + \mathbf{B}^T\mathbf{A}$ is positive (semi-)definite.

In many applications (like in our case), the intrinsic relationships between $\mathbf{A}$ and $\mathbf{B}$ are characterized by a symmetric and positive (semi-)definite matrix. But, due to some unknown factors, such as measurement noises and unmodeled dynamics residing in $\mathbf{A}$ and/or $\mathbf{B}$, $\mathbf{A}^T\mathbf{B} + \mathbf{B}^T\mathbf{A}$ may not be positive (semi-)definite. Thus the normal equation (3.15) may yield an indefinite solution although positive (semi-)definiteness is what we really want.

In [17], Higham derives a method of computing the nearest symmetric positive semi-definite matrix in the Frobenius norm to an arbitrary real matrix. The method is summarized as follows:

For any $\mathbf{A} \in \mathcal{R}^{n \times n}$

1. $\mathbf{B} = \frac{\mathbf{A}+\mathbf{A}^T}{2}$.

2. $\mathbf{B} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, the singular value decomposition of $\mathbf{B}$.

3. $\mathbf{H} = \mathbf{V}\mathbf{\Sigma}\mathbf{V}^T$.

4. $\mathbf{X}_F = \frac{\mathbf{B}+\mathbf{H}}{2}$ is the unique positive approximation of $\mathbf{A}$ in the Frobenius norm.

Note that if $\mathbf{A}$ is positive definite, then the above steps can give a symmetric positive definite approximation. If $\mathbf{A}$ is not positive definite, the above steps can only give a symmetric positive semi-definite approximation.

Chapter 4

Estimation of the Joint Space Mass-Inertia Matrix Using the Total Least Squares Approach

This chapter first presents a geometric interpretation of the least squares and total least squares methods, then summarizes the basic algorithms for one-dimensional and multi-dimensional total least squares problems. Finally, a way is proposed to employ the symmetric constraint.

4.1 Some Geometric Motivations

The least squares method and total least squares method are used to solve a set of overdetermined linear equations

$$\mathbf{A}\vec{x} = \vec{b}, \tag{4.1}$$

where $\mathbf{A}$ and $\vec{b}$ are known.

In the classical least squares approach, the measurement data matrix $\mathbf{A}$ is assumed to be free of error. Hence all errors are confined to the observation vector $\vec{b}$. However, in many applications, like in our case, sampling errors, modeling errors, and sometimes human errors make the data matrix **A** inaccurate as well. Total least squares is one of the methods of doing estimation when there are errors in both the observation vector $\vec{b}$ and the data matrix **A**.

To compare the effects of using least squares as opposed to using total least squares, let's first look at a one variable case, solving a set of overdetermined equations

$$\vec{a}x = \vec{b}, \tag{4.2}$$

where $\vec{a} = [a_1, a_2, \ldots, a_m]^T, \vec{b} = [b_1, b_2, \ldots, b_m]^T \in \mathcal{R}^m$ are known, and $x \in \mathcal{R}$.

If both $\vec{a}$ and $\vec{b}$ are free of errors, and the underlying relationship between $\vec{a}$ and $\vec{b}$ is described by (4.2), then there exists an exact solution of (4.2). However, in almost all applications, either $\vec{a}$ or $\vec{b}$, or both of them contain errors. So there is no exact solution of (4.2).

Suppose the errors exist only in $\vec{b}$, then using a least squares approach is appropriate, and the least squares solution of (4.2) is

$$x_{LS} = \frac{\vec{a}^T\vec{b}}{\vec{a}^T\vec{a}}, \tag{4.3}$$

which minimizes the sum of the squared differences $\sum_{i=1}^{m}(b_i - a_i x)^2$.

Suppose the errors exist only in $\vec{a}$, then using a least squares approach is again appropriate, because (4.2) can be written as

$$\frac{\vec{b}}{x} = \vec{a}. \tag{4.4}$$

Thus the least squares solution of (4.4) is

$$x_{LS} = \frac{\vec{b}^T\vec{b}}{\vec{b}^T\vec{a}}, \tag{4.5}$$

which minimizes the sum of the squared differences $\sum_{i=1}^{m}(a_i - \frac{b_i}{x})^2$.

Geometric interpretations of (4.3) and (4.5) are given in Figure 4.1 and Figure 4.2.

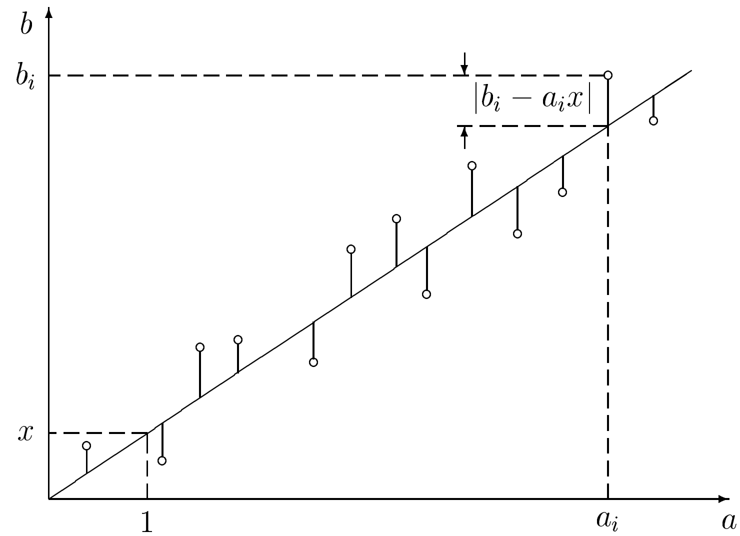

Figure 4.1: Geometric interpretation of one parameter estimation (least squares solution) $\vec{a}x = \vec{b}$ with errors in the measurements $\vec{b}$.

As shown in Figure 4.1, the least squares solution (4.3) minimizes the sum of the squared vertical "errors". And in Figure 4.2, the least squares solution (4.5) minimizes the sum of the squared horizontal "errors".

However, in many applications, both $\vec{a}$ and $\vec{b}$ are measurements containing errors. If the errors are independently and identically distributed with zero mean and common variance, the best estimate $\hat{x}$ of (4.2) is obtained by minimizing the sum of squared distances of the observed points from the fitted line, i.e., $\sum_{i=1}^{m}\frac{(b_i - a_i x)^2}{(1+x^2)}$ [18].

In fact $\hat{x}$ is the total least squares solution of (4.2). Figure 4.3 illustrates the

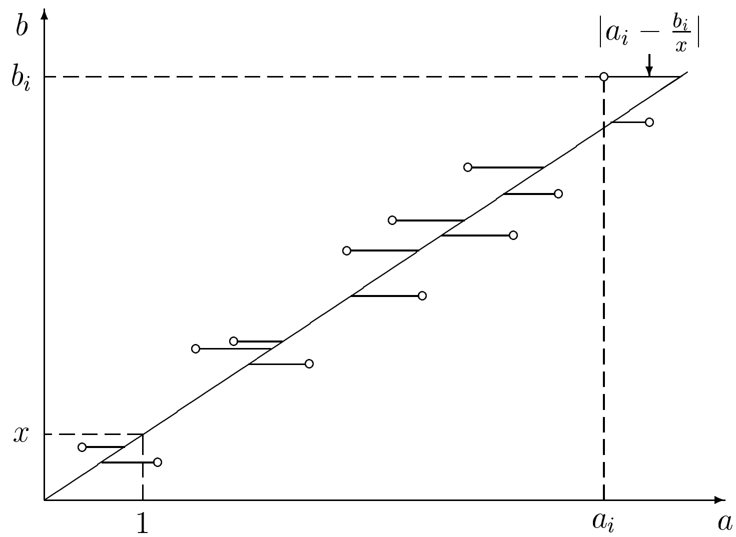

Figure 4.2: Geometric interpretation of one parameter estimation (least squares solution) $\frac{\vec{b}}{x} = \vec{a}$ with errors in the measurements $\vec{a}$.

estimation. The deviations are orthogonal to the fitted line. Therefore, it is the sum of

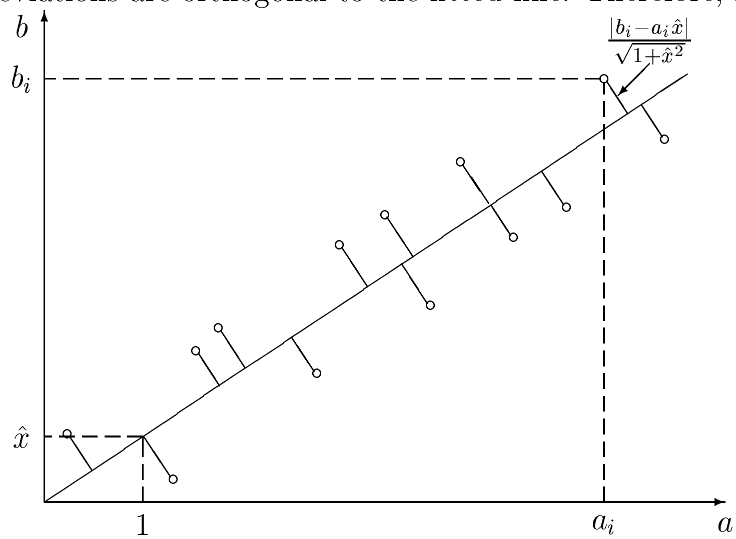

Figure 4.3: Geometric interpretation of one parameter estimation (total least squares solution) with errors in both the measurements $\vec{a}$ and $\vec{b}$.

squares of their lengths that is minimized.

4.2 Solving Total Least Squares Problem

In the field of numerical analysis, the total least squares problem was studied by Golub and Van Loan [19]. It is an alternative form of the least squares method suitable for the case where all data are affected by errors.

A good way to introduce and motivate the total least squares method is to recast the ordinary least squares problem (12.3, [13]).

Definition 4.1 (Ordinary least squares problem) (Definition 2.2, [20]) *Given an overdetermined set of* m *linear equations* $\mathbf{A}\vec{x} = \vec{b}$ *in* n *unknowns* $\vec{x}$*, the least squares problem seeks to*

$$\min_{\vec{b}' \in \mathcal{R}^m} \|\vec{b} - \vec{b}'\|_2 \tag{4.6}$$

$$\text{subject to} \quad \vec{b}' \in \text{Range}(\mathbf{A}). \tag{4.7}$$

Once a minimizing $\vec{b}'$ is found, then any $\vec{x}$ satisfying $\mathbf{A}\vec{x} = \vec{b}'$ is called a least squares solution and $\Delta\vec{b}' = \vec{b} - \vec{b}'$ the corresponding least squares correction. Equation (4.6) and (4.7) are satisfied if $\vec{b}'$ is the orthogonal projection of $\vec{b}$ onto Range($\mathbf{A}$). Thus, the least squares problem amounts to perturbing the observation $\vec{b}$ by a minimum amount $\Delta\vec{b}$ so that $\vec{b}' = \vec{b} - \Delta\vec{b}'$ can be "predicted" by the columns of $\mathbf{A}$.

The underlying assumption here is that the data matrix $\mathbf{A}$ is exactly known, and errors only occur in the observation vector $\vec{b}$. In many applications, this assumption is not realistic since sampling errors, unmodeled dynamics, or measurement noise also affect the data matrix $\mathbf{A}$. As in our case, there are sampling errors, unmodeled dynamics, and measurement noise in both the base forces and the payload accelerations.

One way to take into account the errors in the data matrix $\mathbf{A}$ is to introduce perturbations in $\mathbf{A}$. Under a similar idea as that of the ordinary least squares problem, the total least squares problem is defined as following.

Definition 4.2 (Basic total least squares problem) (Definition 2.3, [20]) *Given an overdetermined set of* m *linear equations* $\mathbf{A}\vec{x} = \vec{b}$ *in* n *unknowns* $\vec{x}$*, the total least squares problem seeks to*

$$\min_{[\hat{\mathbf{A}};\hat{\vec{b}}]\in\mathcal{R}^{m\times(n+1)}} \|[\mathbf{A};\vec{b}] - [\hat{\mathbf{A}};\hat{\vec{b}}]\|_F \tag{4.8}$$

$$\text{subject to} \quad \hat{\vec{b}} \in \text{Range}(\hat{\mathbf{A}}). \tag{4.9}$$

Once a minimizing $[\hat{\mathbf{A}};\hat{\vec{b}}]$ is found, then any $\vec{x}$ satisfying $\hat{\mathbf{A}}\vec{x} = \hat{\vec{b}}$ is called a total least squares solution and $[\Delta\hat{\mathbf{A}};\Delta\hat{\vec{b}}] = [\mathbf{A};\vec{b}] - [\hat{\mathbf{A}};\hat{\vec{b}}]$ the corresponding total least squares correction.

The following two theorems are of great theoretical and practical importance for the total least squares problems. The first theorem is about the singular value decomposition, and the other is about the matrix approximation.

Theorem 4.3 (Singular value decomposition (SVD)) (Theorem 2.5.2, [13]) *If* $\mathbf{A} \in \mathcal{R}^{m\times n}$*, then there exist orthogonal matrices*

$$\mathbf{U} = [\vec{u}_1, \ldots, \vec{u}_m] \in \mathcal{R}^{m\times m} \text{ and } \mathbf{V} = [\vec{v}_1, \ldots, \vec{v}_n] \in \mathcal{R}^{n\times n} \tag{4.10}$$

such that

$$\mathbf{U}^T\mathbf{A}\mathbf{V} = \text{diag}[\sigma_1, \ldots, \sigma_p] \in \mathcal{R}^{m\times n} \quad p = \min(m, n) \tag{4.11}$$

where $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_p \geq 0$.

Theorem 4.4 (Eckart-Young-Mirsky matrix approximation theorem) (Theorem 2.3, [20]) *Let the singular value decomposition of* $\mathbf{A} \in \mathcal{R}^{m\times n}$ *be given by*

$$\mathbf{A} = \sum_{i=1}^{r} \sigma_i \vec{u}_i \vec{v}_i^T \quad \text{with } r = \text{rank}(\mathbf{A}). \tag{4.12}$$

If $k < r$ and $\mathbf{A}_k = \sum_{i=1}^{k} \sigma_i \vec{u}_i \vec{v}_i^T$, *then*

$$\min_{\text{rank}(\mathbf{B})=k} \|\mathbf{A} - \mathbf{B}\|_F = \|\mathbf{A} - \mathbf{A}_k\|_F = \sqrt{\sum_{i=k+1}^{p} \sigma_i{}^2}, \quad p = \min{(m,n)}. \tag{4.13}$$

As pointed out in [21] and [19], the singular value decomposition can be used to solve the total least squares problem. Let's first bring $\mathbf{A}\vec{x} = \vec{b}$ into the following form:

$$[\mathbf{A}; \vec{b}] \begin{bmatrix} \vec{x} \\ \\ -1 \end{bmatrix} = \vec{0}. \tag{4.14}$$

Then, let the singular value decomposition of $[\mathbf{A}; \vec{b}]$ be

$$[\mathbf{A}; \vec{b}] = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \tag{4.15}$$

where

$$\begin{aligned} \mathbf{\Sigma} &= \text{diag}[\sigma_1, \ldots, \sigma_n, \sigma_{n+1}] \in \mathcal{R}^{m\times(n+1)}, \quad \sigma_1 \geq \ldots \geq \sigma_n \geq \sigma_{n+1} \geq 0, \\ \mathbf{U} &= [\vec{u}_1, \ldots, \vec{u}_n, \vec{u}_{n+1}, \ldots, \vec{u}_m] \in \mathcal{R}^{m\times m}, \quad \mathbf{U}\mathbf{U}^T = \mathbf{I}_m, \\ \mathbf{V} &= [\vec{v}_1, \ldots, \vec{v}_n, \vec{v}_{n+1}] \in \mathcal{R}^{(n+1)\times(n+1)}, \quad \mathbf{V}\mathbf{V}^T = \mathbf{I}_{n+1}. \end{aligned} \tag{4.16}$$

If $\sigma_{n+1} \neq 0$, then $[\mathbf{A}; \vec{b}]$ is of rank $n+1$ and the space S generated by the rows of $[\mathbf{A}; \vec{b}]$ coincides with $\mathcal{R}^{n+1}$. There is no nonzero vector in the orthogonal complement of S, hence the set of equations (4.14) have no solution (or are called incompatible). To obtain a solution, the rank of $[\mathbf{A}; \vec{b}]$ must be reduced to n. From the Eckart-Young-Mirsky Theorem 4.4, the best rank n total least squares approximation $[\hat{\mathbf{A}}; \hat{\vec{b}}]$ of $[\mathbf{A}; \vec{b}]$, which minimizes the deviations in variance, is given by

$$[\hat{\mathbf{A}}; \hat{\vec{b}}] = \mathbf{U}\hat{\mathbf{\Sigma}}\mathbf{V}^T \quad \text{with } \hat{\mathbf{\Sigma}} = \text{diag}[\sigma_1, \ldots, \sigma_n, 0] \in \mathcal{R}^{m \times (n+1)}. \tag{4.17}$$

And the minimal total least squares correction given by (4.13) is then

$$\sigma_{n+1} = \min_{\text{rank}([\hat{\mathbf{A}}; \hat{\vec{b}}]) = n} \| [\mathbf{A}; \vec{b}] - [\hat{\mathbf{A}}; \hat{\vec{b}}] \|_F, \tag{4.18}$$

with

$$[\mathbf{A}; \vec{b}] - [\hat{\mathbf{A}}; \hat{\vec{b}}] = [\Delta\mathbf{A}; \Delta\vec{b}] = \sigma_{n+1} \vec{u}_{n+1} \vec{v}_{n+1}^T. \tag{4.19}$$

Now, the approximate equations

$$[\hat{\mathbf{A}}; \hat{\vec{b}}] \begin{bmatrix} \vec{x} \\ -1 \end{bmatrix} = \vec{0} \tag{4.20}$$

are compatible and the solution is $\vec{v}_{n+1}$, as it is in the null space of $[\hat{\mathbf{A}}; \hat{\vec{b}}]$. Thus the total least squares solution is obtained by scaling $\vec{v}_{n+1}$ so that its last entry is -1:

$$\begin{bmatrix} \vec{x}_{TLS} \\ -1 \end{bmatrix} = \frac{-1}{v_{n+1,n+1}} \vec{v}_{n+1}, \tag{4.21}$$

where $v_{n+1,n+1}$ is the $(n+1, n+1)$th entry of matrix $\mathbf{V}$ in (4.16). Note, if $\sigma_{n+1} = 0$, then $[\mathbf{A}; \vec{b}]$ is of rank n, hence $\vec{v}_{n+1}$ is in the null space of $[\mathbf{A}; \vec{b}]$. So the total least squares solution is still given by (4.21).

The following theorem gives conditions for the uniqueness and existence of a total least squares solution.

Theorem 4.5 (Solution of the total least squares problem $\mathbf{A}\vec{x} = \vec{b}$) (Theorem 2.6, [20]) *Suppose the singular value decomposition of* $\mathbf{A}$ *is* $\mathbf{A} = \mathbf{U}'\mathbf{\Sigma}'\mathbf{V}'^T$, *where*

$$\begin{aligned}
\mathbf{\Sigma}' &= \text{diag}[\sigma'_1, \ldots, \sigma'_n] \in \mathcal{R}^{m\times n}, \quad \sigma'_1 \geq \ldots \geq \sigma'_n \geq 0, \\
\mathbf{U}' &= [\vec{u}'_1, \ldots, \vec{u}'_m] \in \mathcal{R}^{m\times m}, \quad \mathbf{U}'\mathbf{U}'^T = \mathbf{I}_m, \\
\mathbf{V}' &= [\vec{v}'_1, \ldots, \vec{v}'_n] \in \mathcal{R}^{n\times n}, \quad \mathbf{V}'\mathbf{V}'^T = \mathbf{I}_n.
\end{aligned}$$

And the singular value decomposition of $[\mathbf{A}; \vec{b}]$ *is given by* (4.15) and (4.16). *If* $\sigma'_n > \sigma_{n+1}$, *then*

$$[\hat{\mathbf{A}}; \hat{\vec{b}}] = \mathbf{U}\hat{\mathbf{\Sigma}}\mathbf{V}^T, \text{ and } \hat{\mathbf{\Sigma}} = \text{diag}[\sigma_1, \ldots, \sigma_n, 0], \tag{4.22}$$

with corresponding total least squares correction matrix

$$[\Delta\hat{\mathbf{A}}; \Delta\hat{\vec{b}}] = [\mathbf{A}; \vec{b}] - [\hat{\mathbf{A}}; \hat{\vec{b}}] = \sigma_{n+1}\vec{u}_{n+1}\vec{v}_{n+1}^T, \tag{4.23}$$

solves the total least squares problem (4.8) *and* (4.9). *And*

$$\vec{x}_{TLS} = -\frac{1}{v_{n+1,n+1}}[v_{1,n+1}, \ldots, v_{n,n+1}]^T \tag{4.24}$$

exists and is the unique solution to $\mathbf{A}\hat{\vec{x}} = \hat{\vec{b}}$. v_{ij} *is the* (i,j)*th entry of matrix* $\mathbf{V}$.

Next, let's consider the multidimensional total least squares problem. Similar to the basic total least squares problem (Definition 4.2), it is defined as

Definition 4.6 (Multidimensional total least squares problem) (Definition 3.1, [20]) *Given an overdetermined set of m linear equations $\mathbf{AX} = \mathbf{B}$, $\mathbf{B} \in \mathcal{R}^{m\times d}$, in $n \times d$ unknowns $\mathbf{X}$. The multidimensional total least squares problem seeks to*

$$\min_{[\hat{\mathbf{A}};\hat{\mathbf{B}}]\in\mathcal{R}^{m\times(n+d)}} \|[\mathbf{A};\mathbf{B}] - [\hat{\mathbf{A}};\hat{\mathbf{B}}]\|_F \tag{4.25}$$

$$\text{subject to} \quad \text{Range}(\hat{\mathbf{B}}) \subseteq \text{Range}(\hat{\mathbf{A}}). \tag{4.26}$$

Once a minimizing $[\hat{\mathbf{A}};\hat{\mathbf{B}}]$ is found, then any $\mathbf{X}$ satisfying

$$\hat{\mathbf{A}}\mathbf{X} = \hat{\mathbf{B}} \tag{4.27}$$

is called a total least squares solution and $[\Delta\hat{\mathbf{A}};\Delta\hat{\mathbf{B}}] = [\mathbf{A};\mathbf{B}] - [\hat{\mathbf{A}};\hat{\mathbf{B}}]$ the corresponding total least squares correction.

Similar to the one dimensional case, first bring (4.27) into the following form:

$$[\mathbf{A};\mathbf{B}]\begin{bmatrix}\mathbf{X}\\ -\mathbf{I}_d\end{bmatrix} = \mathbf{0}_{m\times d}. \tag{4.28}$$

To obtain a solution, the rank of $[\mathbf{A};\mathbf{B}]$ must be reduced to n. First let the singular value decomposition of $[\mathbf{A};\mathbf{B}]$ be

$$[\mathbf{A};\mathbf{B}] = \mathbf{U}^*\mathbf{\Sigma}^*\mathbf{V}^{*T} \tag{4.29}$$

with

$$\begin{aligned}\mathbf{U}^* &= [\mathbf{U}_1;\mathbf{U}_2] \in \mathcal{R}^{m\times m},\ \mathbf{U}_1 = [\vec{u}_1,\ldots,\vec{u}_n],\ \mathbf{U}_2 = [\vec{u}_{n+1},\ldots,\vec{u}_m],\ \mathbf{U}^{*T}\mathbf{U}^* = \mathbf{I}_m,\\ \mathbf{V}^* &= \begin{bmatrix}\mathbf{V}_{11} & \mathbf{V}_{12}\\ \mathbf{V}_{21} & \mathbf{V}_{22}\end{bmatrix} = [\vec{v}_1,\ldots,\vec{v}_{n+d}] \in \mathcal{R}^{(n+d)\times(n+d)},\ \mathbf{V}_{11} \in \mathcal{R}^{n\times n},\ \mathbf{V}_{12} \in \mathcal{R}^{n\times d},\end{aligned}$$

$$\mathbf{V}_{21} \in \mathcal{R}^{d\times n},\ \mathbf{V}_{22} \in \mathcal{R}^{d\times d},\ \mathbf{V}^{*T}\mathbf{V}^{*} = \mathbf{I}_{(n+d)\times(n+d)},$$

$$\mathbf{\Sigma}^{*} = \begin{bmatrix} \mathbf{\Sigma}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_2 \end{bmatrix} = \mathrm{diag}[\sigma_1, \ldots, \sigma_{n+t}],\ t = \min(m-n, d),\ \sigma_1 \geq \ldots \geq \sigma_{n+t} \geq 0,$$

$$\mathbf{\Sigma}_1 = \mathrm{diag}[\sigma_1, \ldots, \sigma_n] \in \mathcal{R}^{n\times n},\ \mathbf{\Sigma}_2 = \mathrm{diag}[\sigma_{n+1}, \ldots, \sigma_{n+t}] \in \mathcal{R}^{(m-n)\times d}.$$

Using the Eckart-Young-Mirsky Theorem 4.4, the best rank n approximation $[\hat{\mathbf{A}}; \hat{\mathbf{B}}]$ of $[\mathbf{A}; \mathbf{B}]$ that minimizes (4.25) is obtained by making the smallest singular values σ_{n+i}, $1 \leq i \leq d$, of $[\mathbf{A}; \mathbf{B}]$ be zero. From (4.29), the lower rank approximation of (4.28) is

$$[\hat{\mathbf{A}}; \hat{\mathbf{B}}] \begin{bmatrix} \mathbf{X} \\ -\mathbf{I}_d \end{bmatrix} = \mathbf{0}_{m\times d} \tag{4.30}$$

with

$$[\hat{\mathbf{A}}; \hat{\mathbf{B}}] = \mathbf{U}_1\mathbf{\Sigma}_1[\mathbf{V}_{11}^T; \mathbf{V}_{21}^T], \tag{4.31}$$

the total least squares approximation of $[\mathbf{A}; \mathbf{B}]$. The corresponding total least squares correction matrix is given by

$$[\Delta\hat{\mathbf{A}}; \Delta\hat{\mathbf{B}}] = \mathbf{U}_2\mathbf{\Sigma}_2[\mathbf{V}_{12}^T; \mathbf{V}_{22}^T] \tag{4.32}$$

and its Frobenius norm

$$\|[\Delta\hat{\mathbf{A}}; \Delta\hat{\mathbf{B}}]\|_F = \|\mathbf{\Sigma}_2\|_F = \sqrt{\sum_{i=1}^{t} \sigma_{n+i}^2}, \quad \text{where} \quad t = \min(m-n, d) \tag{4.33}$$

represents the minimal total least squares correction. The solution of (4.30) is then given by the d right singular vectors $\begin{bmatrix} \mathbf{V}_{12} \\ \mathbf{V}_{22} \end{bmatrix}$, which are in the null space of $[\hat{\mathbf{A}}; \hat{\mathbf{B}}]$. If $\mathbf{V}_{22}$ is nonsingular, the total least squares solution is

$$\begin{bmatrix} \mathbf{X}_{TLS} \\ \\ -\mathbf{I}_d \end{bmatrix} = \begin{bmatrix} \mathbf{V}_{12} \\ \\ \mathbf{V}_{22} \end{bmatrix} [-\mathbf{V}_{22}^{-1}] = \begin{bmatrix} -\mathbf{V}_{12}\mathbf{V}_{22}^{-1} \\ \\ -\mathbf{I}_d \end{bmatrix}. \tag{4.34}$$

The conditions for uniqueness and existence of a total least squares solution for multi-dimensional problems are given in the following theorem.

Theorem 4.7 (Solution of the multidimensional total least squares problem) (Theorem 3.1, [20]) *Suppose* (4.29) *is the singular value decomposition of* $[\mathbf{A}; \mathbf{B}]$. *If* $\sigma_n > \sigma_{n+1}$ *and* $\mathbf{V}_{22}$ *nonsingular, then*

$$[\hat{\mathbf{A}}; \hat{\mathbf{B}}] = \mathbf{U}^* \text{diag}[\sigma_1, \ldots, \sigma_n, 0, \ldots, 0] \mathbf{V}^{*T} = \mathbf{U}_1 \mathbf{\Sigma}_1 [\mathbf{V}_{11}^T; \mathbf{V}_{21}^T], \tag{4.35}$$

with corresponding total least squares correction matrix

$$[\Delta\hat{\mathbf{A}}; \Delta\hat{\mathbf{B}}] = [\mathbf{A}; \mathbf{B}] - [\hat{\mathbf{A}}; \hat{\mathbf{B}}] = \mathbf{U}_2 \mathbf{\Sigma}_2 [\mathbf{V}_{12}^T; \mathbf{V}_{22}^T], \tag{4.36}$$

solves the total least square problem (4.25) *and* (4.26). *And*

$$\mathbf{X}_{TLS} = -\mathbf{V}_{12}\mathbf{V}_{22}^{-1} \tag{4.37}$$

exists and is the unique solution to $\hat{\mathbf{A}}\mathbf{X} = \hat{\mathbf{B}}$.

This section only gives an outline of the most basic total least squares algorithms (Theorem 4.5 and Theorem 4.7). A more detailed discussion about the advanced topics of total least squares (such as nonuniqueness of the solution, sensitivity analysis, and statistical properties) can be found in [20].

4.3 The Symmetric Procrustes Problem and Total Least Squares

Previous section gives two algorithms to solve the one dimensional and multidimensional total least squares problems. But these algorithms are only valid for unconstrained problems.

The joint space mass-inertia matrix is symmetric and positive definite. These constraints should be considered in the total least squares estimation. Unfortunately, unlike the symmetric procrustes method using the least squares approach, the symmetric procrustes problem using the total least squares method is still an open problem.

However, Theorem 4.7 can still be used to calculate a total least squares estimate $\mathbf{X}_{TLS}$. Then the nearest symmetric matrix $\mathbf{X}_S$ in Frobenius norm to $\mathbf{X}_{TLS}$ is given by (P12.6.10, [13])

$$\mathbf{X}_S = \frac{{\mathbf{X}_{TLS}}^T + \mathbf{X}_{TLS}}{2}. \tag{4.38}$$

And the nearest symmetric positive approximation $\mathbf{X}_F$ in Frobenius norm to $\mathbf{X}_{TLS}$ is given by [17]

$$\mathbf{X}_F = \frac{\mathbf{B} + \mathbf{H}}{2} \tag{4.39}$$

where $\mathbf{B} = \frac{{\mathbf{X}_{TLS}}^T + \mathbf{X}_{TLS}}{2}$, $\mathbf{B} = \mathbf{UH}$ is the polar decomposition (4.2.10, [13]) of $\mathbf{B}$. The columns of $\mathbf{U}$ are orthonormal, and $\mathbf{H}$ is a symmetric and positive definite matrix with eigenvalues equal to the singular values of $\mathbf{B}$.

If $\mathbf{X}_{TLS}$ is positive definite, then $\mathbf{X}_S$ and $\mathbf{X}_F$ are symmetric positive definite. If

$\mathbf{X}_{TLS}$ is positive semi-definite, then $\mathbf{X}_S$ and $\mathbf{X}_F$ are symmetric positive semi-definite. If $\mathbf{X}_{TLS}$ is indefinite, then $\mathbf{X}_S$ is symmetric indefinite, and $\mathbf{X}_F$ is symmetric positive semi-definite.

Chapter 5

Estimation of Joint Space Mass-Inertia Matrix Employing Positive Definite Constraint

Although several algorithms have been discussed in Chapter 3 and Chapter 4, none of them really takes into account the positive definite constraint. In the symmetric procrustes problem, the definiteness of the estimate $\mathbf{X}_{LS}$ depends on the definiteness of $\mathbf{A}^T\mathbf{B} + \mathbf{B}^T\mathbf{A}$ (Section 3.2). Using the total least squares method, the estimate $\mathbf{X}_{TLS}$ will not even be symmetric. If what we want is a symmetric positive definite matrix, then Higham's method [17] will not be very useful because it only computes a nearest symmetric positive semi-definite matrix if $\mathbf{X}_{LS}$ and $\mathbf{X}_{TLS}$ are indefinite.

This chapter first introduces a method to employ the positive definite constraint. Then, a necessary condition for the existence and uniqueness of a symmetric positive definite estimate is proven. Finally, a geometric explanation of one parameter estimation is given, which illustrates that the symmetric positive definite estimate seeks to minimize the sum of the areas.

5.1 Symmetric Positive Definite Estimation Problem

Our goal is to solve an overdetermined set of linear equations

$$\mathbf{AX} = \mathbf{B} \tag{5.1}$$

with the constraints

$$\mathbf{X}^T = \mathbf{X} \quad \text{and} \quad \mathbf{X} \text{ being positive definite,} \tag{5.2}$$

where $\mathbf{A},\ \mathbf{B} \in \mathcal{R}^{m\times n}$, $\mathbf{X} \in \mathcal{R}^{n\times n}$.

It is well known that if $\mathbf{X} \in \mathcal{R}^{n\times n}$ is symmetric and positive definite, then it can be written as

$$\mathbf{X} = \mathbf{YY}^T \tag{5.3}$$

where $\mathbf{Y} \in \mathcal{R}^{n\times n}$, $\mathbf{Y}$ is nonsingular. Substituting (5.3) into (5.1), we have

$$\mathbf{AYY}^T = \mathbf{B} \tag{5.4}$$

which can also be written as

$$\mathbf{AY} = \mathbf{BY}^{-T}. \tag{5.5}$$

Note, the constraints (5.2) have been embedded into (5.4) and (5.5).

If (5.1) has a symmetric positive definite solution $\mathbf{X}^*$, then (5.4) and (5.5) can be solved, and all solutions satisfy

$$\mathbf{YY}^T = \mathbf{X}^*. \tag{5.6}$$

Obviously, we are not interested in this case.

If (5.1) fails to have an exact solution or the solutions are not symmetric positive definite, then (5.4) and (5.5) don't have an exact solution. However, by transforming (5.5) into the following optimizing problem, an optimum solution (under certain criterion) can be found.

Definition 5.1 (Symmetric positive definite estimation problem) *Given an over-determined set of m linear equations $\mathbf{AX} = \mathbf{B}$, $\mathbf{B} \in \mathcal{R}^{m\times n}$, in $\frac{n(n+1)}{2}$ unknowns $\mathbf{X}^T = \mathbf{X} \in \mathcal{R}^{n\times n}$, the symmetric positive definite estimation problem seeks to*

$$\min_{\mathbf{Y}\in S} \|\mathbf{AY} - \mathbf{BY}^{-T}\|_F^2 \tag{5.7}$$

where $S = \{\mathbf{Z} \in \mathcal{R}^{n\times n} : \mathbf{Z} \text{ is nonsingular.}\}$. And the symmetric positive definite estimate $\mathbf{X}$ is given by $\mathbf{X} = \mathbf{YY}^T$.

The following theorem gives a necessary condition for a local minimizer for (5.7).

Theorem 5.2 *Suppose that $\mathbf{Y}^* \in S$ is a local minimizer for problem (5.7), then it satisfies the following equation*

$$\mathbf{Y}^*\mathbf{Y}^{*T}\mathbf{A}^T\mathbf{AY}^*\mathbf{Y}^{*T} = \mathbf{B}^T\mathbf{B}. \tag{5.8}$$

Proof: Let $f(\mathbf{Y})$ be a real valued function defined on S by

$$\begin{aligned}
f(\mathbf{Y}) &= \|\mathbf{AY} - \mathbf{BY}^{-T}\|_F^2 \\
&= \text{Trace}\left\{(\mathbf{AY} - \mathbf{BY}^{-T})^T(\mathbf{AY} - \mathbf{BY}^{-T})\right\} \\
&= \text{Trace}\left\{\mathbf{Y}^T\mathbf{A}^T\mathbf{AY} + \mathbf{Y}^{-1}\mathbf{B}^T\mathbf{BY}^{-T}\right\} + c
\end{aligned} \tag{5.9}$$

where $c = -2\,\text{Trace}\left\{\mathbf{A}^T\mathbf{B}\right\}$ is a constant.

Let $g(\mathbf{X},\mathbf{Y})$ be a real valued function defined on $\mathcal{R}^{n\times n}\times\mathcal{R}^{n\times n}$ by

$$g(\mathbf{X},\mathbf{Y}) = \text{Trace}\left\{\mathbf{Y}^T\mathbf{A}^T\mathbf{A}\mathbf{Y} + \mathbf{X}\mathbf{B}^T\mathbf{B}\mathbf{X}^T\right\}. \tag{5.10}$$

Then minimizing $f(\mathbf{Y})$ on S is equivalent to minimizing $g(\mathbf{X},\mathbf{Y})$ on $\mathcal{R}^{n\times n}\times\mathcal{R}^{n\times n}$ with the constraint $\mathbf{X}\mathbf{Y} = \mathbf{I}_n$.

Let $\mathbf{X}$, $\mathbf{Y} \in \mathcal{R}^{n\times n}$, $\mathbf{X} = [\vec{x}_1,\ldots,\vec{x}_n]^T$, $\mathbf{Y} = [\vec{y}_1,\ldots,\vec{y}_n]$, and $\mathbf{\Lambda} = [\lambda_{ij}] \in \mathcal{R}^{n\times n}$. Now it is convenient to introduce the Lagrangian associated with the constrained problem (5.10), defined as

$$L(\mathbf{X},\mathbf{Y},\mathbf{\Lambda}) = g(\mathbf{X},\mathbf{Y}) + \sum_{i=1}^{n}\sum_{j=1}^{n}\lambda_{ij}\left(\vec{x}_i^T\vec{y}_j - \delta_{ij}\right), \tag{5.11}$$

where

$$\delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}.$$

By rearranging terms, (5.11) can be written as

$$L(\mathbf{X},\mathbf{Y},\mathbf{\Lambda}) = \text{Trace}\left\{\mathbf{Y}^T\mathbf{A}^T\mathbf{A}\mathbf{Y} + \mathbf{X}\mathbf{B}^T\mathbf{B}\mathbf{X}^T + \mathbf{\Lambda}^T(\mathbf{X}\mathbf{Y} - \mathbf{I}_n)\right\}. \tag{5.12}$$

Thus the necessary conditions (10.3, [22]) for a local minimum can be expressed in the form

$$\left.\frac{\partial L(\mathbf{X},\mathbf{Y},\mathbf{\Lambda})}{\partial x_{ij}}\right|_{\mathbf{X}^*,\mathbf{Y}^*,\mathbf{\Lambda}^*} = 0 \tag{5.13}$$

$$\left.\frac{\partial L(\mathbf{X},\mathbf{Y},\mathbf{\Lambda})}{\partial y_{ij}}\right|_{\mathbf{X}^*,\mathbf{Y}^*,\mathbf{\Lambda}^*} = 0 \tag{5.14}$$

$$\left.\frac{\partial L(\mathbf{X},\mathbf{Y},\mathbf{\Lambda})}{\partial \lambda_{ij}}\right|_{\mathbf{X}^*,\mathbf{Y}^*,\mathbf{\Lambda}^*} = 0. \tag{5.15}$$

for all $1 \leq i, j \leq n$.

Using (3.9, 3.11, 3.12, 3.13), equations (5.13, 5.14, 5.15) can be written as

$$\begin{aligned}
\text{Trace}\left\{[\mathbf{B}^T\mathbf{B}\mathbf{X}^T + \mathbf{B}^T\mathbf{B}\mathbf{X}^T + \mathbf{Y}\mathbf{\Lambda}^T]\frac{\partial \mathbf{X}}{\partial x_{ij}}\right\}\Bigg|_{\mathbf{X}^*,\mathbf{Y}^*,\mathbf{\Lambda}^*} &= 0 \quad (5.16)\\
\text{Trace}\left\{[\mathbf{Y}^T\mathbf{A}^T\mathbf{A} + \mathbf{Y}^T\mathbf{A}^T\mathbf{A} + \mathbf{\Lambda}^T\mathbf{X}]\frac{\partial \mathbf{Y}}{\partial y_{ij}}\right\}\Bigg|_{\mathbf{X}^*,\mathbf{Y}^*,\mathbf{\Lambda}^*} &= 0 \quad (5.17)\\
\text{Trace}\left\{[\mathbf{X}\mathbf{Y} - \mathbf{I}_n]\frac{\partial \mathbf{\Lambda}^T}{\partial \lambda_{ij}}\right\}\Bigg|_{\mathbf{X}^*,\mathbf{Y}^*,\mathbf{\Lambda}^*} &= 0 \quad (5.18)
\end{aligned}$$

for all $1 \leq i, j \leq n$. This implies

$$\begin{aligned}
2\mathbf{B}^T\mathbf{B}\mathbf{X}^{*T} + \mathbf{Y}^*\mathbf{\Lambda}^{*T} &= \mathbf{0}_n \quad (5.19)\\
2\mathbf{Y}^{*T}\mathbf{A}^T\mathbf{A} + \mathbf{\Lambda}^{*T}\mathbf{X}^* &= \mathbf{0}_n \quad (5.20)\\
\mathbf{X}^*\mathbf{Y}^* - \mathbf{I}_n &= \mathbf{0}_n. \quad (5.21)
\end{aligned}$$

From (5.19) and (5.21) we have

$$\mathbf{\Lambda}^* = -2\mathbf{X}^*\mathbf{B}^T\mathbf{B}\mathbf{X}^{*T}. \quad (5.22)$$

Substituting (5.21) and (5.22) into (5.20) we get

$$\mathbf{Y}^*\mathbf{Y}^{*T}\mathbf{A}^T\mathbf{A}\mathbf{Y}^*\mathbf{Y}^{*T} = \mathbf{B}^T\mathbf{B}. \quad (5.23)$$

This completes the proof. □

The following Corollary gives a necessary condition for the uniqueness of the symmetric positive definite estimate.

Corollary 5.3 *If* Rank($\mathbf{A}$) = Rank($\mathbf{B}$) = n, *and* $\mathbf{Y}^* \in S$ *is a local minimizer for the problem* (5.7), *then the symmetric positive definite estimation problem in* Definition 5.1

has a unique solution

$$\mathbf{X} = \mathbf{Y}^*\mathbf{Y}^{*T} = \mathbf{U}\boldsymbol{\Sigma}_1^{-1}\mathbf{V}\boldsymbol{\Sigma}_2\mathbf{V}^T\boldsymbol{\Sigma}_1^{-1}\mathbf{U}^T \tag{5.24}$$

where,

$$\begin{aligned}
\mathbf{A}^T\mathbf{A} &= \mathbf{U}\boldsymbol{\Sigma}_1\mathbf{U}^T\mathbf{U}\boldsymbol{\Sigma}_1\mathbf{U}^T, \;\; \mathbf{U}^T\mathbf{U} = \mathbf{I}_n, \;\; \boldsymbol{\Sigma}_1 = \mathrm{diag}[\sqrt{\lambda_1}, \ldots, \sqrt{\lambda_n}], \\
& \lambda_1, \ldots, \lambda_n \textit{ are eigenvalues of } \mathbf{A}^T\mathbf{A}, \\
\boldsymbol{\Sigma}_1\mathbf{U}^T\mathbf{B}^T\mathbf{B}\mathbf{U}\boldsymbol{\Sigma}_1 &= \mathbf{V}\boldsymbol{\Sigma}_2\mathbf{V}^T\mathbf{V}\boldsymbol{\Sigma}_2\mathbf{V}^T, \;\; \mathbf{V}^T\mathbf{V} = \mathbf{I}_n, \;\; \boldsymbol{\Sigma}_2 = \mathrm{diag}[\sqrt{\sigma_1}, \ldots, \sqrt{\sigma_n}], \\
& \sigma_1, \ldots, \sigma_n \textit{ are eigenvalues of } \boldsymbol{\Sigma}_1\mathbf{U}^T\mathbf{B}^T\mathbf{B}\mathbf{U}\boldsymbol{\Sigma}_1.
\end{aligned}$$

Proof: From Theorem 5.2, $\mathbf{Y}^*$ satisfies

$$\mathbf{Y}^*\mathbf{Y}^{*T}\mathbf{A}^T\mathbf{A}\mathbf{Y}^*\mathbf{Y}^{*T} = \mathbf{B}^T\mathbf{B}. \tag{5.25}$$

Since Rank($\mathbf{A}$) $= n$, it follows that $\mathbf{A}^T\mathbf{A}$ is symmetric and positive definite. Using square root decomposition, we have

$$\mathbf{A}^T\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}_1\mathbf{U}^T\mathbf{U}\boldsymbol{\Sigma}_1\mathbf{U}^T = \mathbf{U}\boldsymbol{\Sigma}_1\boldsymbol{\Sigma}_1\mathbf{U}^T \tag{5.26}$$

where $\mathbf{U}$ is unitary, $\boldsymbol{\Sigma}_1 = \mathrm{diag}[\sqrt{\lambda_1}, \ldots, \sqrt{\lambda_n}]$ with $\lambda_1, \ldots, \lambda_n$ the eigenvalues of $\mathbf{A}^T\mathbf{A}$. Substitute (5.26) into (5.25), left multiply both sides by $\boldsymbol{\Sigma}_1\mathbf{U}^T$, and right multiply both sides by $\mathbf{U}\boldsymbol{\Sigma}_1$, we have

$$\boldsymbol{\Sigma}_1\mathbf{U}^T\mathbf{Y}^*\mathbf{Y}^{*T}\mathbf{U}\boldsymbol{\Sigma}_1\boldsymbol{\Sigma}_1\mathbf{U}^T\mathbf{Y}^*\mathbf{Y}^{*T}\mathbf{U}\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_1\mathbf{U}^T\mathbf{B}^T\mathbf{B}\mathbf{U}\boldsymbol{\Sigma}_1. \tag{5.27}$$

Since Rank($\mathbf{B}$) $= n$, $\boldsymbol{\Sigma}_1$ and $\mathbf{U}$ are nonsingular. This implies that $\boldsymbol{\Sigma}_1\mathbf{U}^T\mathbf{B}^T\mathbf{B}\mathbf{U}\boldsymbol{\Sigma}_1$ is symmetric and positive definite. Using square root decomposition again, we have

$$\boldsymbol{\Sigma}_1\mathbf{U}^T\mathbf{B}^T\mathbf{B}\mathbf{U}\boldsymbol{\Sigma}_1 = \mathbf{V}\boldsymbol{\Sigma}_2\mathbf{V}^T\mathbf{V}\boldsymbol{\Sigma}_2\mathbf{V}^T \tag{5.28}$$

where $\mathbf{V}$ is unitary, $\mathbf{\Sigma_2} = \text{diag}[\sqrt{\sigma_1}, \ldots, \sqrt{\sigma_n}]$ with $\sigma_1, \ldots, \sigma_n$ the eigenvalues of $(\mathbf{V\Sigma_2V}^T)^2$. Thus (5.27) can be written as

$$\left(\mathbf{\Sigma_1 U}^T\mathbf{Y}^*\mathbf{Y}^{*T}\mathbf{U\Sigma_1}\right)^2 = \left(\mathbf{V\Sigma_2V}^T\right)^2. \tag{5.29}$$

Since $\mathbf{Y}^*\mathbf{Y}^{*T}$ is symmetric and positive definite, and $\mathbf{\Sigma}_1\mathbf{U}^T$ is invertable, this implies that $\mathbf{\Sigma_1 U}^T\mathbf{Y}^*\mathbf{Y}^{*T}\mathbf{U\Sigma_1}$ is symmetric and positive definite. Thus from the property of the square root of a symmetric positive definite matrix (4.2.10, [13]), we have

$$\mathbf{\Sigma_1 U}^T\mathbf{Y}^*\mathbf{Y}^{*T}\mathbf{U\Sigma_1} = \mathbf{V\Sigma_2V}^T. \tag{5.30}$$

Solving (5.30) we have

$$\mathbf{X} = \mathbf{Y}^*\mathbf{Y}^{*T} = \mathbf{U\Sigma_1^{-1}V\Sigma_2V}^T\mathbf{\Sigma_1^{-1}U}^T \tag{5.31}$$

This completes the proof. □

Corollary 5.3 says that if $\text{Rank}(\mathbf{A}) = \text{Rank}(\mathbf{B}) = n$, and we known (5.7) has at least one local minimum, then (5.24) gives the unique symmetric positive definite estimate for (5.1).

5.2 A Geometric Interpretation

Section 4.1 gives us some geometric interpretations of the least squares and total least squares methods. As shown in Figure 4.1 and 4.2, the least squares method seeks to minimize the sum of the squared vertical or horizontal "errors". And Figure 4.3 tells us that using the total least squares method, it is the sum of squared distances of the observed points from the fitted line that is minimized. Both the least squares and total least squares methods have nice geometric meanings. What is the geometric meaning of the optimality criterion in (5.7)?

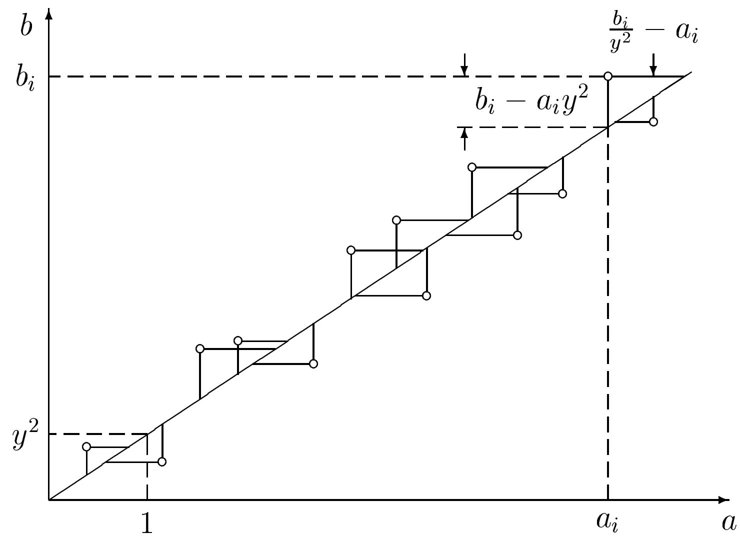

Figure 5.1: Geometric interpretation of symmetric positive definite estimation problem $\vec{a}x = \vec{b}$ with one parameter.

Although it is not obvious, minimizing the optimality criterion in (5.7) is in fact equivalent to minimizing the sum of the "areas" of the error triangles [14], which can be shown by the following equivalent transformations

$$\begin{aligned}
\|\mathbf{A}\mathbf{Y} - \mathbf{B}\mathbf{Y}^{-T}\|_F^2 &= \text{Trace}\left\{(\mathbf{A}\mathbf{Y} - \mathbf{B}\mathbf{Y}^{-T})^T(\mathbf{A}\mathbf{Y} - \mathbf{B}\mathbf{Y}^{-T})\right\} \\
&= \text{Trace}\left\{(\mathbf{Y}^T\mathbf{A}^T - \mathbf{Y}^{-1}\mathbf{B}^T)(\mathbf{A}\mathbf{Y} - \mathbf{B}\mathbf{Y}^{-T})\right\} \\
&= \text{Trace}\left\{\mathbf{Y}(\mathbf{Y}^T\mathbf{A}^T - \mathbf{Y}^{-1}\mathbf{B}^T)(\mathbf{A}\mathbf{Y} - \mathbf{B}\mathbf{Y}^{-T})\mathbf{Y}^{-1}\right\} \\
&= \text{Trace}\left\{(\mathbf{Y}\mathbf{Y}^T\mathbf{A}^T - \mathbf{B}^T)(\mathbf{A} - \mathbf{B}\mathbf{Y}^{-T}\mathbf{Y}^{-1})\right\} \\
&= \text{Trace}\left\{(\mathbf{A}\mathbf{Y}\mathbf{Y}^T - \mathbf{B})^T(\mathbf{A} - \mathbf{B}\mathbf{Y}^{-T}\mathbf{Y}^{-1})\right\}. \qquad (5.32)
\end{aligned}$$

In the one variable case, (5.32) becomes

$$\begin{aligned} \|\vec{a}y - \frac{\vec{b}}{y}\|_2^2 &= (\vec{a}y^2 - \vec{b})^T(\vec{a} - \frac{\vec{b}}{y^2}) \\ &= \sum_{i=1}^{m} \left(a_i y^2 - b_i\right)\left(a_i - \frac{b_i}{y^2}\right) \end{aligned} \qquad (5.33)$$

where $\vec{a} = [a_1, a_2, \ldots, a_m]^T$, $\vec{b} = [b_1, b_2, \ldots, b_m]^T \in \mathcal{R}^m$ are data vectors, and y is the variable to be estimated. The geometric interpretation of (5.33) is given in Figure 5.1. It is the sum of the "areas" of the error triangles that is minimized.

Chapter 6

Numerical Results

In this chapter, three numerical examples of determining a symmetric positive definite matrix are given. The symmetric procrustes method (SP), the total least squares method (TLS), and symmetric positive definite estimation method (SPD) are compared. Supporting MATLAB M-files are contained in Appendix B.

The matrix ($\mathbf{M}_p$) data given below is actual data from one of UW's flexure jointed hexapods. It is calculated from the real design parameters of the hexapod.

$$\mathbf{M}_p = \begin{bmatrix} 4.6881 & 0.1978 & -0.4042 & 1.7981 & -0.4046 & 0.6110 \\ 0.1978 & 4.6881 & 0.6110 & -0.4046 & 1.7981 & -0.4042 \\ -0.4042 & 0.6110 & 4.6881 & 0.1975 & -0.4042 & 1.7984 \\ 1.7981 & -0.4046 & 0.1975 & 4.6881 & 0.6114 & -0.4042 \\ -0.4046 & 1.7981 & -0.4042 & 0.6114 & 4.6881 & 0.1975 \\ 0.6110 & -0.4042 & 1.7984 & -0.4042 & 0.1975 & 4.6881 \end{bmatrix}. \tag{6.1}$$

The relationship between payload accelerations and base forces is

$$\vec{f}_b = \mathbf{M}_p \ddot{\vec{p}}_u \tag{6.2}$$

where $\vec{f}_b = [f_{b1}, \ldots, f_{b6}]^T$, $\ddot{\vec{p}}_u = [\vec{u}_1^T \ddot{\vec{p}}_1, \ldots, \vec{u}_6^T \ddot{\vec{p}}_6]^T$. The measured payload accelerations and base forces are

$$\hat{\ddot{\vec{p}}}_u = \ddot{\vec{p}}_u + \vec{v}_l \quad (6.3)$$

$$\hat{\vec{f}}_b = \vec{f}_b + \vec{v}_f \quad (6.4)$$

where $\vec{v}_l = [v_{l1}, \ldots, v_{l6}]^T$ and $\vec{v}_f = [v_{f1}, \ldots, v_{f6}]^T$ are measurement noises in the payload accelerations and base forces respectively.

Experiment 1: $\ddot{\vec{p}}_u = [\vec{u}_1^T \ddot{\vec{p}}_1, \ldots, \vec{u}_6^T \ddot{\vec{p}}_6]^T$ are independent white noise signals with zero mean and common variance 8.2. v_{li} and v_{fi} are independent white noise signals with zero mean and common variance 0.082.

- Using the symmetric procrustes algorithm (3.15), the least squares estimate is

$$\hat{\mathbf{M}}_{LS} = \begin{bmatrix} 4.6923 & 0.3468 & -0.2365 & 1.7292 & -0.3496 & 0.6795 \\ 0.3468 & 4.9109 & 0.6210 & -0.4481 & 1.8583 & -0.3188 \\ -0.2365 & 0.6210 & 4.4224 & 0.2190 & -0.4299 & 1.5901 \\ 1.7292 & -0.4481 & 0.2190 & 4.5584 & 0.6608 & -0.3114 \\ -0.3496 & 1.8583 & -0.4299 & 0.6608 & 4.7572 & 0.1994 \\ 0.6795 & -0.3188 & 1.5901 & -0.3114 & 0.1994 & 4.4602 \end{bmatrix}. \quad (6.5)$$

  And the estimation error is measured by

$$\text{Estimation Error } = \|\mathbf{M}_p - \hat{\mathbf{M}}_{LS}\|_F = 0.6755. \quad (6.6)$$

- Using the total least squares algorithm (4.37), and casting the result into a symmetric matrix by (4.38), the estimate is

$$\hat{\mathbf{M}}_{TLS} = \begin{bmatrix} 4.8274 & 0.3924 & -0.2547 & 1.7308 & -0.3314 & 0.6634 \\ 0.3924 & 4.9296 & 0.6146 & -0.4683 & 1.8915 & -0.3243 \\ -0.2547 & 0.6146 & 4.4648 & 0.2180 & -0.4287 & 1.6177 \\ 1.7308 & -0.4683 & 0.2180 & 4.5382 & 0.6834 & -0.2978 \\ -0.3314 & 1.8915 & -0.4287 & 0.6834 & 4.7530 & 0.2111 \\ 0.6634 & -0.3243 & 1.6177 & -0.2978 & 0.2111 & 4.4957 \end{bmatrix}. \quad (6.7)$$

And the estimation error is

$$\text{Estimation Error } = \|\mathbf{M}_p - \hat{\mathbf{M}}_{TLS}\|_F = 0.6899. \quad (6.8)$$

- Using the symmetric positive definite estimation algorithm (5.24), the estimate is

$$\hat{\mathbf{M}}_{spd} = \begin{bmatrix} 4.6756 & 0.3219 & -0.2249 & 1.7124 & -0.3855 & 0.6973 \\ 0.3219 & 4.9281 & 0.6110 & -0.4229 & 1.8757 & -0.3527 \\ -0.2249 & 0.6110 & 4.4442 & 0.2110 & -0.3984 & 1.5892 \\ 1.7124 & -0.4229 & 0.2110 & 4.6488 & 0.7056 & -0.3230 \\ -0.3855 & 1.8757 & -0.3984 & 0.7056 & 4.7443 & 0.2255 \\ 0.6973 & -0.3527 & 1.5892 & -0.3230 & 0.2255 & 4.4565 \end{bmatrix}. \quad (6.9)$$

And the estimation error is

$$\text{Estimation Error } = \|\mathbf{M}_p - \hat{\mathbf{M}}_{spd}\|_F = 0.6630. \quad (6.10)$$

Experiment 2: $\ddot{\vec{p}}_u = [\vec{u}_1^T\ddot{\vec{p}}_1, \ldots, \vec{u}_6^T\ddot{\vec{p}}_6]^T$ are independent white noise signals with zero mean and common variance 8.2. v_{li} and v_{fi} are independent white noise signals with zero mean and common variance 0.86.

- Using the symmetric procrustes algorithm (3.15), the least squares estimate is

$$\hat{\mathbf{M}}_{LS} = \begin{bmatrix} 4.3104 & -0.1033 & -0.0144 & 2.1560 & -0.3818 & 1.007 \\ -0.1033 & 5.1528 & 0.8565 & -0.5108 & 2.2474 & -0.5110 \\ -0.0144 & 0.8565 & 3.4603 & 0.1445 & -0.2490 & 1.2144 \\ 2.1560 & -0.5108 & 0.1445 & 4.5493 & 0.2986 & -0.1294 \\ -0.3818 & 2.2474 & -0.2490 & 0.2986 & 4.3539 & -0.2728 \\ 1.0069 & -0.5110 & 1.2144 & -0.1294 & -0.2728 & 4.0349 \end{bmatrix}. \quad (6.11)$$

And the estimation error is

$$\text{Estimation Error } = \|\mathbf{M}_p - \hat{\mathbf{M}}_{LS}\|_F = 2.3610. \quad (6.12)$$

- Using the total least squares algorithm (4.37), and casting the result into a symmetric matrix by (4.38), the estimate is

$$\hat{\mathbf{M}}_{TLS} = \begin{bmatrix} 4.6080 & 0.0809 & -0.3035 & 2.2266 & -0.7014 & 0.9748 \\ 0.0809 & 5.9198 & 0.7359 & -0.5061 & 1.9453 & -0.4148 \\ -0.3035 & 0.7359 & 3.5913 & -0.0697 & -0.1584 & 1.2445 \\ 2.2266 & -0.5061 & -0.0697 & 4.6728 & 0.2298 & -0.3504 \\ -0.7014 & 1.9453 & -0.1584 & 0.2298 & 4.6010 & -0.2161 \\ 0.9748 & -0.4148 & 1.2445 & -0.3504 & -0.2161 & 4.3855 \end{bmatrix}. \quad (6.13)$$

And the estimation error is

$$\text{Estimation Error } = \|\mathbf{M}_p - \hat{\mathbf{M}}_{TLS}\|_F = 2.3007. \quad (6.14)$$

- Using the symmetric positive definite estimation algorithm (5.24), the estimate is

$$\hat{\mathbf{M}}_{spd} = \begin{bmatrix} 4.6242 & -0.0176 & -0.3489 & 2.2192 & -0.4702 & 0.9565 \\ -0.0176 & 5.3939 & 0.7467 & -0.4705 & 2.0333 & -0.5280 \\ -0.3489 & 0.7467 & 3.9466 & 0.0348 & -0.0596 & 1.2329 \\ 2.2192 & -0.4705 & 0.0348 & 4.4798 & 0.4103 & -0.2566 \\ -0.4702 & 2.0333 & -0.0596 & 0.4103 & 4.6034 & -0.2907 \\ 0.9565 & -0.5280 & 1.2329 & -0.2566 & -0.2907 & 4.0921 \end{bmatrix}. \quad (6.15)$$

And the estimation error is

$$\text{Estimation Error } = \|\mathbf{M}_p - \hat{\mathbf{M}}_{spd}\|_F = 1.9689. \quad (6.16)$$

Experiment 3: $\ddot{\vec{p}}_u = [\vec{u}_1^T \ddot{\vec{p}}_1, \ldots, \vec{u}_6^T \ddot{\vec{p}}_6]^T$ are independent white noise signals with zero mean and common variance 8.2. v_{li} and v_{fi} are independent white noise signals with zero mean and common variance 1.28.

- Using the symmetric procrustes algorithm (3.15), the least squares estimate is

$$\hat{\mathbf{M}}_{LS} = \begin{bmatrix} 3.3256 & 0.0243 & -0.3521 & 0.8617 & -0.4398 & 0.0413 \\ 0.0243 & 4.0808 & 0.6554 & -0.2534 & 1.4923 & -0.8605 \\ -0.3521 & 0.6554 & 4.5358 & -0.2603 & -0.7717 & 1.2805 \\ 0.8617 & -0.2534 & -0.2604 & 4.9588 & 1.5125 & -0.5735 \\ -0.4398 & 1.4923 & -0.7717 & 1.5125 & 4.5898 & 0.1853 \\ 0.0413 & -0.8605 & 1.2805 & -0.5735 & 0.1853 & 4.3464 \end{bmatrix}. \quad (6.17)$$

And the estimation error is

$$\text{Estimation Error } = \|\mathbf{M}_p - \hat{\mathbf{M}}_{LS}\|_F = 2.9120. \quad (6.18)$$

- Using the total least squares algorithm (4.37), and casting the result into a symmetric matrix by (4.38), the estimate is

$$\hat{\mathbf{M}}_{TLS} = \begin{bmatrix} 2.8541 & 0.3739 & -1.1554 & 2.1164 & -0.0316 & 0.1311 \\ 0.3739 & 4.0972 & 0.7489 & 0.0802 & 1.4777 & -0.3654 \\ -1.1554 & 0.7489 & 5.2622 & -0.9170 & -0.8636 & 1.3953 \\ 2.1164 & 0.0802 & -0.9170 & 5.8818 & 1.7438 & -0.3611 \\ -0.0316 & 1.4777 & -0.8635 & 1.7438 & 5.0314 & 0.2683 \\ 0.1311 & -0.3654 & 1.3953 & -0.3611 & 0.2683 & 4.5636 \end{bmatrix}. \tag{6.19}$$

And the estimation error is

$$\text{Estimation Error } = \|\mathbf{M}_p - \hat{\mathbf{M}}_{TLS}\|_F = 3.7761. \tag{6.20}$$

- Using the symmetric positive definite estimation algorithm (5.24), the estimate is

$$\hat{\mathbf{M}}_{spd} = \begin{bmatrix} 3.8906 & 0.1994 & -0.5813 & 1.3596 & -0.3519 & 0.3424 \\ 0.1994 & 4.3711 & 0.6745 & -0.0381 & 1.4298 & -0.7983 \\ -0.5814 & 0.6745 & 4.4697 & -0.1437 & -0.6737 & 1.3437 \\ 1.3596 & -0.0381 & -0.1437 & 5.0564 & 1.2486 & -0.4884 \\ -0.3519 & 1.4298 & -0.6737 & 1.2486 & 4.5972 & 0.1782 \\ 0.3424 & -0.7983 & 1.3437 & -0.4884 & 0.1782 & 4.5134 \end{bmatrix}. \tag{6.21}$$

And the estimation error is

$$\text{Estimation Error } = \|\mathbf{M}_p - \hat{\mathbf{M}}_{spd}\|_F = 2.0083. \tag{6.22}$$

Chapter 7

Conclusions and Future Work

An overdetermined set of linear equations $\mathbf{AX} = \mathbf{B}$, with the constraints $\mathbf{X}^T = \mathbf{X}$ and $\mathbf{X}$ being positive definite, can be solved in three possible ways:

1. The Symmetric Procrustes Method (3.15) directly employs the symmetry constraint. But, the definiteness of the solution depends on the definiteness of the data matrix $\mathbf{A}^T\mathbf{B} + \mathbf{B}^T\mathbf{A}$. Since it is a constrained least squares method, one of the data matrices, $\mathbf{A}$ or $\mathbf{B}$, is supposed to be "error" free.

2. The Total Least Squares Method (4.37) takes into account the "errors" in both the data matrix $\mathbf{A}$ and $\mathbf{B}$, but it does not include the symmetry and positive definiteness constraints. Using (4.39), the total least squares estimate can be cast into a nearest symmetric positive definite matrix only if it is positive definite.

3. The new method proposed in Chapter 5 can handle both the symmetry and the positive definiteness constraints. In addition, it gives a better result in the numerical experiments. But, only a necessary condition for the existence and uniqueness of solutions has been proven.

The following is a brief list of future work

1. Find a way to solve the symmetric constrained total least squares problem - symmetric procrustes method using total least squares.

2. Find and prove a sufficient condition for the existence of the solutions of problem (5.7).

3. Investigate the statistical properties of the symmetric positive definite estimation method (5.24).

Bibliography

[1] D. Thayer, J. Vagners, A. Von Flotow, C. Hardham, and K. Scribner, "Six-axis vibration isolation systems using soft actuators and multiple sensors," in *Proceedings of the Annual American Astronautical Society (AAS) Rocky Mountain Guidance and Control Conference,* pp. 497-506, 1998.

[2] J. Sullivan, A. Rahman, R. Cobb, and J. Spanos, "Closed-loop performance of a vibration isolation and suppression system," in *Proceedings of the American Control Conference,* pp. 3974-3978, 1997.

[3] E. H. Anderson, D. J. Leo, and M. D. Holcomb, "Ultraquiet platform for active vibration isolation," in *Proceedings of the SPIE Smart Structure and Materials Conference,* pp. 436-451, 1996.

[4] Z. J. Geng and L. S. Haynes, "Six degree-of-freedom active vibration control using the Stewart platforms," *IEEE Transactions on Control Systems Technology,* vol. 2, pp. 45-53, March 1994.

[5] Z. J. Geng, G. G. Pan, L. S. Haynes, B. K. Wada, and J. A. Garba, "An intelligent control system for multiple degree-of-freedom vibration isolation," *Journal of Intelligent Material Systems and Structures,* vol. 6, pp. 787-800, November 1995.

[6] J. Spanos, Z. Rahman, and G. Blackwood, "A soft 6-axis active vibration isolator," in *Proceedings of the American Control Conference,* pp. 412-416, 1995.

[7] J. E. McInroy, J. F. O'Brien, and G. W. Neat, "Precise, fault tolerant pointing using a Stewart platform," *IEEE/ASME Transactions on Mechatronics,* vol. 4, pp. 91-95, March 1999.

[8] J. F. O'Brien, J. E. McInroy, D. Bodtke, M. Bruch, and J. C. Hamann, "Lessons learned in nonlinear systems and flexible robots through experiments on a 6 legged platform," in *Proceedings of the American Control Conference,* (Philadelphia, PA), pp. 868-872, 1998.

[9] J. E. McInroy and J. C. Hamann, "Design and Control of Flexure Jointed Hexapods," *IEEE Transactions on Robotics and Automation,* Accepted and awaiting publication.

[10] J. E. McInroy, "Dynamic Modeling of Flexure Jointed Hexapods for Control Purposes," *IEEE Conference on Control Applications,* (Kona, Hawaii), pp. 508-513, August 1999. (Long version also submitted to *IEEE/ASME Transactions on Mechatronics.*)

[11] J. J. Craig, *Introduction to Robotics: Mechanics and Control,* Reading, MA: Addison-Wesley, 1986.

[12] Yixin Chen and J. E. McInroy, "Identification and Decoupling Control of Flexure Jointed Hexapods," Submitted to *IEEE International Conference on Robotics and Automation 2000,* (San Francisco, CA).

[13] G. H. Golub and C. F. Van Loan, *Matrix Computations,* 3rd ed., Johns Hopkins University Press, Baltimore, MD, 1996.

[14] J. E. Brock, "Optimal Matrices Describing Linear Systems," *AIAA Journal,* 6(1968), pp. 1292-1296.

[15] N. J. Higham, "The Symmetric Procrustes Problem," *BIT,* 28(1988), pp. 133-143.

[16] J. E. Slotine and W. Li, *Applied Nonlinear Control,* Prentice Hall, Englewood Cliffs, New Jersey 07632, 1991.

[17] N. J. Higham, "Computing a Nearest Symmetric Positive Semidefinite Matrix," *Linear Algebra and its Applications,* 103(1988), pp. 103-118.

[18] S. Van Huffel and J. Vandewalle, "On the Accuracy of Total Least Squares and Least Squares Techniques in the Presence of Errors on All Data," *Automatica,* 25(1989), pp. 765-769.

[19] G. H. Golub and C. F. Van Loan, "An Analysis of the Total Least Squares Problem," *SIAM J. Numer. Anal.,* 17(1980), pp. 883-893.

[20] S. Van Huffel and J. Vandewalle, *The Total Least Squares Problem: Computational Aspects and Analysis,* SIAM Philadelphia, 1991.

[21] G. H. Golub, "Some Modified Matrix Eigenvalue Problems," *SIAM Rev.,* 15(1973), pp. 318-344.

[22] D. G. Luenberger, *Linear and Nonlinear Programming,* Second Edition, Addison-Wesley Publishing Company, 1984.

[23] A. L. Peressini, F. E. Sullivan, and J. J. Uhl, Jr., *The Mathematics of Nonlinear Programming,* Springer-Verlag New York Inc., 1988.

Appendix A

Convex Sets and Convex Functions

Definition A.1 ([23] 2.1.1) *A set C in $\mathcal{R}^n$ is convex if for every $\vec{x}$ and $\vec{y}$ in C, the line segment joining $\vec{x}$ and $\vec{y}$ also lies in C.*

Definition A.2 ([23] 2.3.2) *Suppose that $f(\vec{x})$ is a real-valued function defined on a convex set C in $\mathcal{R}^n$. Then:*

1. *the function $f(\vec{x})$ is convex on C if*

$$f(\lambda\vec{x} + [1-\lambda]\vec{y}) \leq \lambda f(\vec{x}) + [1-\lambda]f(\vec{y}) \tag{A.1}$$

   for all $\vec{x}$, $\vec{y}$ in C and all λ with $0 \leq \lambda \leq 1$;

2. *the function $f(\vec{x})$ is strictly convex on C if*

$$f(\lambda\vec{x} + [1-\lambda]\vec{y}) < \lambda f(\vec{x}) + [1-\lambda]f(\vec{y}) \tag{A.2}$$

   for all $\vec{x}$, $\vec{y}$ in C with $\vec{x} \neq \vec{y}$ and all λ with $0 < \lambda < 1$. If the inequalities in *the above definitions are reversed, we obtain the definitions of concave and strictly concave functions.*

Definition A.3 *Suppose that $f(\vec{x})$ is a real-valued function for which all second partial derivatives of $f(\vec{x})$ exist on a subset D of $\mathcal{R}^n$. The gradient ∇f of $f(\vec{x})$ is the n-vector*

$$\nabla f = \left[\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \cdots, \frac{\partial f}{\partial x_n}\right]^T, \tag{A.3}$$

while the Hessian Hf of $f(\vec{x})$ is the symmetric $n \times n$-matrix

$$Hf = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \cdots & \cdots & \cdots & \cdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix} \tag{A.4}$$

Theorem A.4 ([23] 2.3.7) *Suppose that $f(\vec{x})$ has continuous second partial derivatives on some open convex set C in $\mathcal{R}^n$. If the Hessian $Hf(\vec{x})$ of $f(\vec{x})$ is positive semi-definite (resp. positive definite) on C, then $f(\vec{x})$ is convex (resp. strictly convex) on C.*

Theorem A.5 ([23] 2.3.10) *If $f_1(\vec{x}), \ldots, f_k(\vec{x})$ are convex functions on a convex set C in $\mathcal{R}^n$, then*

$$f(\vec{x}) = f_1(\vec{x}) + f_2(\vec{x}) + \cdots + f_k(\vec{x}) \tag{A.5}$$

is convex. Moreover, if at least one $f_i(\vec{x})$ is strictly convex on C, then the sum $f(\vec{x})$ is strictly convex.

Theorem A.6 ([23] 2.3.5) *Suppose that $f(\vec{x})$ has continuous first partial derivatives on a convex set C in $\mathcal{R}^n$. Then:*

1. *the function* $f(\vec{x})$ *is convex if and only if*

$$f(\vec{x}) + \nabla f(\vec{x}) \cdot (\vec{y} - \vec{x}) \leq f(\vec{y}) \tag{A.6}$$

for all $\vec{x}$, $\vec{y}$ *in* C;

2. *the function* $f(\vec{x})$ *is strictly convex on* C *if and only if*

$$f(\vec{x}) + \nabla f(\vec{x}) \cdot (\vec{y} - \vec{x}) < f(\vec{y}) \tag{A.7}$$

for all $\vec{x}$, $\vec{y}$ *in* C *with* $\vec{x} \neq \vec{y}$.

Corollary A.7 ([23] 2.3.6) *If* $f(\vec{x})$ *is a convex function with continuous first partial derivatives on some convex set* C*, then any critical point of* $f(\vec{x})$ *in* C *is a global minimizer of* $f(\vec{x})$.

Theorem A.8 ([23] 1.2.3) *Suppose that* $f(\vec{x})$ *is a real-valued function for which all first partial derivatives of* $f(\vec{x})$ *exist on a subset* D *of* $\mathcal{R}^n$*. If* $\vec{x}^*$ *is an interior point of* D *that is a local minimizer of* $f(\vec{x})$*, then* $\vec{x}^*$ *is a critical point of* $f(\vec{x})$*, that is,* $\nabla f(\vec{x}^*) = \vec{0}$.

Theorem A.9 ([23] 2.3.4) *Any local minimizer of a convex function* $f(\vec{x})$ *defined on a convex subset* C *of* $\mathcal{R}^n$ *is also a global minimizer. Any local minimizer of a strictly convex function* $f(\vec{x})$ *defined on a convex set* C *in* $\mathcal{R}^n$ *is the unique strict global minimizer of* $f(\vec{x})$ *on* C.

Appendix B

Matlab M-files

```
%Simulation Code to generate the measurements of
%payload accelerations and base forces.
clear all;
m = 12; %m – the number of samples
n = 6; %n – the number of variables
nm = 3.9;
%nm = 1 – the variance of the noise is 0.082
%nm = 3.2 – the variance of the noise is 0.86
%nm = 3.9 – the variance of the noise is 1.28
%Mprel is the ideal joint space mass-inertia matrix.
Mprel = [4.6881 0.1978 -0.4042 1.7981 -0.4046 0.6110;
         0.1978 4.6881 0.6110  -0.40461.7981  -0.4042;
```

```
        -0.40420.6110 4.6881  0.1975 -0.4042 1.7984;
        1.7981 -0.40460.1975  4.6881 0.6114  -0.4042;
        -0.40461.7981 -0.4042 0.6114 4.6881  0.1975;
        0.6110 -0.40421.7984  -0.40420.1975  4.6881];

%Generate the payload accelerations lddot
lddot = 10.*(rand([m,n]) - 0.5.*ones([m,n]));
%Add noise to the measurement of lddotm
lddotm = lddot + nm*(rand([m,n])-0.5.*ones([m,n]));
%Generate the base forces fb
fb = lddot * Mprel;
%Add noise to the measurement of fbm
fbm = fb + nm*(rand([m,n])-0.5.*ones([m,n]));
%Save data
save simdata m n nm Mprel lddot lddotm fb fbm;
```

Symmetric Procrustes Problem.

```
clear all;

%Load the simulation data

load simdata;

%Calculate symmetric estimate of Mprel

Mestsp = lyap(lddotm'*lddotm, -1.*(lddotm'*fbm+fbm'*lddotm));

ERROR = norm(Mprel-Mestsp,'fro');
```

Total Least Square Estimate.

```
clear all;

%Load the simulation data

load simdata;

%Singular value decomposition of [lddotm fbm]

[U,S,V] = svd([lddotm fbm]);

%V=[V11 V12; V21 V22]

V12 = V(1:n,n+1:2*n);

V22 = V(n+1:2*n,n+1:2*n);

%Calculate symmetric estimate of Mptls

Mesttls = -1.*V12*inv(V22);

%Calculate the Frobenius Norm of Mprel-Mesttls

ERROR1 = norm(Mprel-Mesttls,'fro');

%Cast Mesttls into symmetric matrix

Mesttlss = 0.5.*(Mesttls' + Mesttls);

%Calculate the Frobenius Norm of Mprel-Mesttlss

ERROR2 = norm(Mprel-Mesttlss,'fro');
```

Symmetric Positive Definite Estimation Problem.

```
clear all;

%Load the simulation data

load simdata;

%Calculate positive definite matrices P and Q

P = lddotm'*lddotm;

Q = fbm'*fbm;

%Calculate the squart root of P

[U,D1] = eig(P);

Sigma1 = sqrt(D1);

%Calculate the squart root of Sigma1*U'*Q*U*Sigma1

[V,D2] = eig(Sigma1*U'*Q*U*Sigma1);

Sigma2 = sqrt(D2);

%Calcualte the symmetric and positive definite estimate Mestspd

Mestspd = U*inv(Sigma1)*V*Sigma2*V'*inv(Sigma1)*U';

%Calculate the Frobenius Norm of Mprel-Mestspd

ERROR = norm(Mprel-Mestspd,'fro');
```