Chapter | three

The Basics of Analytical Mechanics, Optimization, Control and Estimation

making tapes together, discs together,
sweating for applause
they read basically to and for
each other.

Charles Bukowski (1920–1994)

In this chapter, we review the basics of *analytical mechanics* as applied to orbit theory, optimization, control and estimation. We start by presenting some notions in Lagrangian and Hamiltonian mechanics, including *canonical transformations*. We introduce the *Delaunay variables* and show their applications in *Brouwer's satellite theory*. This material is followed by a discussion of static optimization, control, and filtering methods, to be applied in subsequent chapters to the development of relative spacecraft control and navigation algorithms.

3.1 LAGRANGIAN AND HAMILTONIAN MECHANICS

Lagrangian and Hamiltonian mechanics are two main approaches to mechanics that constitute a generalization of the classical Newtonian mechanics. The rationale behind the Lagrangian and Hamiltonian formalisms, often termed analytical mechanics, is related to *variational principles*, which we will briefly discuss in this section. Hamiltonian mechanics is based on the energy concept, and is directly related to symmetry, one of the most powerful tools in modern physics. The subject of Lagrangian and Hamiltonian mechanics is vast. The reader is referred to, e.g., Marsden and Ratiu [71] for a rigorous treatment of these subjects.

Spacecraft Formation Flying; ISBN: 9780750685337

Lagrangian and Hamiltonian mechanics have played a fundamental role in astrodynamics during the past two and a half centuries. Hamiltonian mechanics was applied to some of the most fundamental astrodynamical theories, such as Brouwer's artificial satellite theory [59,66], to be discussed in Chapters 7 and 8. Our interest in Lagrangian and Hamiltonian mechanics stems from the important applications of these methods to the development of spacecraft formation flying models. We will subsequently use Lagrangian and Hamiltonian methods in Chapter 4, where we show how to develop nonlinear models of relative motion using analytical mechanics; in Chapter 5, where we show how to derive linear relative motion models based on energy methods; and in Chapter 8, where we develop perturbation theories.

Lagrangian mechanics deals with a *configuration space*, $\mathbb{Q}$, parameterized by a set of *generalized coordinates*, $\mathbf{q} \in \mathbb{Q}$, and *generalized velocities*, $\dot{\mathbf{q}}$. The structure $T_s(\mathbb{Q}) = \mathbb{Q} \times \mathbb{Q}$ is called the *tangent space*. Coordinates on $T_s(\mathbb{Q})$ are then $(\mathbf{q}, \dot{\mathbf{q}})$. The *Lagrangian*, $\mathcal{L} : T_s(\mathbb{Q}) \times \mathbb{R} \to \mathbb{R}$, is a function of time and the generalized coordinates and velocities, and is defined as the kinetic minus the potential energy of the system:

$$\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}, t) = \mathcal{K} - \mathcal{U} \tag{3.1}$$

Hamilton's variational principle states that

$$\delta \int_a^b \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}, t) dt = 0 \tag{3.2}$$

Using the calculus of variations, Eq. (3.2) leads to the *Euler–Lagrange equations*

$$\frac{\mathrm{d}}{\mathrm{d}t} \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} - \frac{\partial \mathcal{L}}{\partial \mathbf{q}} = \mathbf{0} \tag{3.3}$$

To use the Hamiltonian formalism, one introduces the *conjugate momenta*, defined as

$$\mathbf{p} = \frac{\partial \mathcal{L}}{\partial \dot{\mathbf{q}}} \tag{3.4}$$

and then adopts the change of variables $(\mathbf{q}, \dot{\mathbf{q}}) \to (\mathbf{q}, \mathbf{p})$. The *Legendre transformation* provides a connection between the *Hamiltonian*, $\mathcal{H}$, and the Lagrangian:

$$\mathcal{H}(\mathbf{q}, \mathbf{p}, t) = \mathbf{p}^T \dot{\mathbf{q}} - \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}, t) \tag{3.5}$$

The structure $T_c(\mathbb{Q})$, having coordinates $(\mathbf{q}, \mathbf{p})$ is referred to as the *cotangent space*, so that $\mathcal{H} : T_c(\mathbb{Q}) \times \mathbb{R} \to \mathbb{R}$. Using the change of variables and the Legendre transformation leads to *Hamilton's equations*:

$$\dot{\mathbf{q}} = \frac{\partial \mathcal{H}}{\partial \mathbf{p}}$$

$$\dot{\mathbf{p}} = -\frac{\partial \mathcal{H}}{\partial \mathbf{q}} \tag{3.6}$$

3.2 THE DELAUNAY ELEMENTS

The Hamiltonian formalism can be applied to astrodynamical problems using the Delaunay canonical elements. These elements are obtained when one defines the transformation

$$l = M \tag{3.7a}$$
$$g = \omega \tag{3.7b}$$
$$\hbar = \Omega \tag{3.7c}$$
$$L = \sqrt{\mu a} \tag{3.7d}$$
$$G = \sqrt{\mu a(1 - e^2)} \tag{3.7e}$$
$$H = G \cos i \tag{3.7f}$$

In this case $(l, g, \hbar)$ become generalized coordinates, and (L, G, H) are the conjugate momenta.[1] Given a nominal Hamiltonian, $\mathcal{H}_0$, representing the two-body Keplerian motion, a perturbing Hamiltonian, $\mathcal{H}_1$, and the total Hamiltonian, $\mathcal{H} = \mathcal{H}_0 + \varepsilon \mathcal{H}_1$, the Delaunay variables satisfy the following Hamilton equations (cf. Eqs. (3.6)):

$$\dot{l} = \frac{\partial \mathcal{H}}{\partial L}, \quad \dot{g} = \frac{\partial \mathcal{H}}{\partial G}, \quad \dot{\hbar} = \frac{\partial \mathcal{H}}{\partial H} \tag{3.8}$$

$$\dot{L} = -\frac{\partial \mathcal{H}}{\partial l}, \quad \dot{G} = -\frac{\partial \mathcal{H}}{\partial g}, \quad \dot{H} = -\frac{\partial \mathcal{H}}{\partial \hbar} \tag{3.9}$$

Note that G is the orbital angular momentum and H is the corresponding polar component. The term $\mathcal{H}_0$ is given by

$$\mathcal{H}_0 = -\frac{\mu^2}{2L^2} \tag{3.10}$$

The consequences of $\mathcal{H}_0$ being only a function of L, for the two-body problem are that all the momenta and the coordinates g and $\hbar$ are constants and l varies linearly with time.

These results are no longer valid in the presence of the perturbing Hamiltonian. The Delaunay variables lend themselves to canonical transformations, which render the Hamiltonian system easily integrable; this is the basis for Brouwer's satellite theory [66]. We discuss such transformations in Section 3.3, before introducing Brouwer's satellite theory. We will also use the Delaunay formalism in Chapter 8 to derive perturbation mitigation methods.

[1] We will use the notation $\hbar$ instead of the customary h so as to avoid confusion with the orbital angular momentum which is denoted by H.

3.3 CANONICAL TRANSFORMATIONS

In general, the Hamiltonian is a function of both $\mathbf{q}$ and $\mathbf{p}$. It is advantageous in many applications to determine transformations of the generalized coordinates and momenta such that the new variables $(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}})$ also satisfy Eq. (3.6), but with a different Hamiltonian, $\widetilde{\mathcal{H}}(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}}, t)$, having some special properties. Such a transformation is called a *canonical transformation*, and it is a vast subject matter in itself. There are many advantages of finding such transformations; for example, $\widetilde{\mathcal{H}} = \widetilde{\mathcal{H}}(\widetilde{\mathbf{p}}, t)$ leads to the constants of motion $\widetilde{\mathbf{p}}$ (cf. Eq. (3.10)). We will restrict our discussion to the special case of *autonomous dynamical systems*, for which it can be shown that a canonical transformation must satisfy the Hamiltonian invariance property Ref. [59]

$$\widetilde{\mathbf{q}}^T \mathrm{d}\widetilde{\mathbf{p}} - \mathbf{q}^T \mathrm{d}\mathbf{p} = \mathrm{d}\left[\widetilde{\mathbf{q}}^T \widetilde{\mathbf{p}} - W\right] \tag{3.11}$$

where $W(\widetilde{\mathbf{q}}, \mathbf{p})$ is called a *generating function* and d stands for the differential. The generating function can be of various types, but the one being discussed here depends on the new generalized coordinates and the old momentum variables such that

$$\widetilde{\mathbf{p}} = \frac{\partial W(\widetilde{\mathbf{q}}, \mathbf{p})}{\partial \widetilde{\mathbf{q}}} \tag{3.12a}$$

$$\mathbf{q} = \frac{\partial W(\widetilde{\mathbf{q}}, \mathbf{p})}{\partial \mathbf{p}} \tag{3.12b}$$

Given $(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}})$, Eqs. (3.12) can be solved for $(\mathbf{q}, \mathbf{p})$ if W is known. A variety of transformations can be generated by selecting different generating functions.

It is usually a difficult task to obtain an explicit generating function to meet the desired objectives for the transformation, since equations of the type (3.12) are implicit. However, the task is simplified for systems whose Hamiltonians can be represented as

$$\mathcal{H}(\mathbf{q}, \mathbf{p}) = \mathcal{H}_0(\mathbf{q}, \mathbf{p}) + \epsilon \mathcal{H}_1(\mathbf{q}, \mathbf{p}) + 0.5\, \epsilon^2 \mathcal{H}_2(\mathbf{q}, \mathbf{p}) + \mathcal{O}(\epsilon^3) \tag{3.13}$$

where ϵ is a small non-dimensional parameter and the unperturbed Hamiltonian $\mathcal{H}_0$ is that of an integrable system, such as the two-body problem. The perturbation parameter ϵ, for example, can be related to J_2. There are two common methods for determining the canonical transformations of such systems: von Zeipel's method, which expands the perturbed Hamiltonian into a Taylor series [66], and Hori's method [72], which uses the Lie series expansion [73]. The von Zeipel method involves a generating function of the mixed type, as used in Eqs. (3.12). On the other hand, Hori's method determines a generating function of the new coordinates and momenta. The generating function is expanded as a series in ϵ:

$$W(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}}) = W_1(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}}) + \epsilon W_2(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}}) + \mathcal{O}(\epsilon^2) \tag{3.14}$$

and the transformed Hamiltonian, also expanded as

$$\widetilde{\mathcal{H}}(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}}) = \widetilde{\mathcal{H}}_0(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}}) + \epsilon \widetilde{\mathcal{H}}_1(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}}) + \mathcal{O}(\epsilon^2) \tag{3.15}$$

The Lie theorem provides a relationship between the Hamiltonian and the generating function as follows:

$$\mathcal{H}(\mathbf{q}, \mathbf{p}) = \mathcal{H}_0(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}}) + \epsilon \{\mathcal{H}, W\} + 0.5\, \epsilon^2 \{\{\mathcal{H}, W\}, W\} + \mathcal{O}(\epsilon^3) \tag{3.16}$$

where $\{\cdot, \cdot\}$ is the Poisson bracket, defined by Eq. (2.93).

The coefficients of the like powers of ϵ can be equated between Eqs. (3.15) and (3.16) to obtain a series of equations, the first two of which are

$$\widetilde{\mathcal{H}}_0(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}}) = \mathcal{H}_0(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}}) \tag{3.17a}$$

$$\widetilde{\mathcal{H}}_1(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}}) = \mathcal{H}_1(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}}) + \{\mathcal{H}_0(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}}), W_1\} \tag{3.17b}$$

Thus, $\widetilde{\mathcal{H}}_0$ is obtained by substituting the new variables $(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}})$ for the respective old variables $(\mathbf{q}, \mathbf{p})$ in $\mathcal{H}_0$. Since Eq. (3.17b) is a single equation in two unknowns, $\widetilde{\mathcal{H}}_1$ and W_1, it can be solved in a variety of ways. Generally, the secular part of $\mathcal{H}_1(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}})$ is absorbed into $\widetilde{\mathcal{H}}_1(\widetilde{\mathbf{q}}, \widetilde{\mathbf{p}})$ and the periodic part is absorbed into the solution for W_1 via quadratures. This process is called averaging. The detailed treatments of the Brouwer and Hori methods are given in Vinti [65].

A modified version of Hori's method has also been implemented for a higher-order satellite theory by Coffey and Deprit [74] with the help of a special-purpose symbolic algebra program.

3.4 BROUWER THEORY

Brouwer [66] developed his satellite theory based on the conversion of the mean Delaunay variables into the corresponding osculating variables, a process known as the *Brouwer transformation*. Brouwer writes Eq. (3.13) for the perturbed Hamiltonian with $\epsilon = -J_2$. Our development of the Brouwer theory uses only the J_2 terms. The higher-order geopotential terms are ignored. The canonical variables used are the Delaunay variables, introduced in Section 3.2. The first term of the Hamiltonian is as defined by Eq. (3.10), and the second term is

$$\mathcal{H}_1 = \frac{\mu^4 R_e^2}{4L^6} \left(\frac{a}{r}\right)^3 \left[\left(3\frac{H^2}{G^2} - 1\right) + 3\left(1 - \frac{H^2}{G^2}\right)\cos\theta\right] \tag{3.18}$$

where r is the orbit radius and θ is the argument of latitude. Note that $\mathcal{H}$ is a function of all the variables except $\hbar$; consequently, none of them will be constant except H, the polar component of the angular momentum.[2] Using Lie

[2] With J_2 as the only perturbation, the right ascension, $\hbar$, does not appear; however, it would appear at the second-order if the time-dependent terms of the geopotential were included.

series [73] or the von Zeipel method as done by Brouwer [66], the Hamiltonian is averaged first with respect to the mean anomaly l to remove the short-periodic (of the order of the orbit period) terms, and then with respect to the argument of perigee g to remove the long-periodic (approximately one order higher than the orbit period) terms. In the remainder of this development, the notation $\bar{(\cdot)}$ will be dropped and the elements $(l, g, \hbar, L, G, H)$ will be assumed to be mean elements. The algebra is minimized if dimensionless variables are used, so distances will be normalized by the Earth's radius, and time will be normalized by the mean motion of a satellite at one Earth radius. This gives $\mu = 1$. In the dimensionless variables, the averaged Hamiltonians become

$$\bar{\mathcal{H}}_0 = -\frac{1}{2L^2} \tag{3.19}$$

$$\bar{\mathcal{H}}_1 = -\frac{1}{4L^6}\left(\frac{L}{G}\right)^3\left(1 - 3\frac{H^2}{G^2}\right) \tag{3.20}$$

The first-order long-periodic and short-periodic generating functions of Brouwer are

$$W^{(lp)} = -\left(\frac{1}{32G^3}\right)\left(1 - \frac{G^2}{L^2}\right)\left(1 - 5\frac{H^2}{G^2}\right)^{-1} \times \left(1 - 16\frac{H^2}{G^2} + 15\frac{H^4}{G^4}\right)\sin 2g \tag{3.21}$$

and

$$W^{(sp)} = -\frac{1}{4G^3}\left(1 - 3\frac{H^2}{G^2}\right)(f - l + e\sin f) + \frac{3}{8G^3}\left(1 - \frac{H^2}{G^2}\right) \times \left[\sin(2f + 2g) + e\sin(f + 2g) + \frac{e}{3}\sin(3f + 2g)\right] \tag{3.22}$$

For the sake of completeness and for reference in Chapter 7, the second-order averaged Hamiltonian is also given below:

$$\bar{\mathcal{H}}_2 = -\frac{15}{64L^{10}}\left(\frac{L}{G}\right)^5\left[\left(1 - \frac{18}{5}\frac{H^2}{G^2} + \frac{H^4}{G^4}\right) + \frac{4}{5}\left(\frac{L}{G}\right)\left(1 - 6\frac{H^2}{G^2} + 9\frac{H^4}{G^4}\right) - \left(\frac{L}{G}\right)^2\left(1 - 2\frac{H^2}{G^2} - 7\frac{H^4}{G^4}\right)\right] \tag{3.23}$$

Note that after averaging, the Hamiltonian is a function of only the momenta, i.e., the angles are ignorable. Consequently, the angle rates and momenta are

constant. As will be shown in Chapter 8, this property makes the design of formations in mean element space easier than that with osculating elements. In the osculating space, the angles are not linear in time and the momenta are not constant. Hence, finding the initial values of the elements for a specific formation is much more difficult than when using mean elements.

Brouwer's transformation in its original form accepts mean elements as inputs and produces the osculating elements as the outputs. The exact inverse transformation is nontrivial to write, since it is nonlinear. However, as we are dealing with a perturbed system with a small perturbation parameter, the first-order approximate inverse transformation can be obtained by replacing J_2 by $-J_2$ in the Brouwer transformation and treating the osculating elements as the inputs and the mean elements as the outputs. This is very convenient, but it may not be acceptable for accurate simulations. If such is the case, then an iterative solution to the inverse transformation may be pursued.

The first-order transformation between the mean and osculating nonsingular orbital elements [75] presented in Chapter 7 forms an important part of the material presented in this book and it is based on generating functions obtained by Brouwer. The detailed expressions of this transformation are provided in Appendix E. Lyddane's modification [76] of Brouwer theory to eliminate singularities associated with small eccentricities and inclinations can be found in Ref. [29]. We present an example of the forward and inverse Brouwer transformation next.

Example 3.1 (*Mean-to-osculating transformation*). *Let the given mean elements be:*

$$\begin{aligned} &\bar{a} = 7100 \text{ km}, \quad \bar{\theta} = 0 \text{ rad}, \quad \bar{i} = 70^\circ \\ &\bar{q}_1 = 0.05, \quad \bar{q}_2 = 0.05, \quad \bar{\Omega} = 45^\circ \end{aligned} \tag{3.24}$$

Obtain the osculating elements.

The first-order Brouwer transformation results in the following osculating elements:

$$\begin{aligned} &a = 7109.31795 \text{ km}, \quad \theta = 0.00005 \text{ rad}, \quad i = 1.22196 \text{ rad} \\ &q_1 = 0.05063, \quad q_2 = 0.05003, \quad \Omega = 0.78547 \text{ rad} \end{aligned} \tag{3.25}$$

Note that there is approximately a 10 km difference between the mean and osculating semimajor axis values.

Example 3.2 (*Osculating-to-mean transformation*). *Now consider the osculating elements of the previous example as given and obtain the mean elements via the first-order inverse transformation.*

The first-order inverse transformation provides the following mean elements:

$$\begin{aligned} &\bar{a} = 7099.996055 \text{ km}, \quad \bar{\theta} = 0.000008 \text{ rad}, \quad \bar{i} = 1.221731 \text{ rad} \\ &\bar{q}_1 = 0.0500006, \quad \bar{q}_2 = 0.04999994, \quad \bar{\Omega} = 0.7853984 \text{ rad} \end{aligned} \tag{3.26}$$

Note that there are minor differences between the mean elements obtained for this example and their respective counterparts from the previous example.

3.4.1 Osculating-to-mean Iterative Solution

If the mean-to-osculating transformation is represented as a nonlinear mapping

$$\mathbf{oe} = \mathbf{oe}(\overline{\mathbf{oe}}) \tag{3.27}$$

then an iterative procedure can be devised to solve Eq. (3.27) for $\overline{\mathbf{oe}}$, given $\mathbf{oe}$. This procedure will require an initial guess for $\overline{\mathbf{oe}}$ and the information on the Jacobian $D = \partial\mathbf{oe}/\partial\overline{\mathbf{oe}}$. The matrix D is provided in Appendix F.

3.5 CONSTRAINED STATIC OPTIMIZATION

In cases where the control of relative motion is applied using impulsive thrusters, the relative dynamics and the associated kinematic constraints may be described by algebraic equations. In this case, derivation of relative spacecraft control methods can be based on simple *static optimization* procedures; by static optimization we mean that differential equations are absent. We will elaborate on this topic in Section 4.3. The purpose of the following discussion is to introduce a method for solving constrained static optimization problems using the formalism of *Lagrange multipliers*.

To that end, consider the following static optimization problem:

$$\begin{aligned} &\text{minimize} \quad \mathcal{J}(\mathbf{x}) \\ &\text{s. t.} \\ &g_k(\mathbf{x}) = 0, \quad k = 1, \ldots, m \end{aligned} \tag{3.28}$$

In Eq. (3.27), $\mathcal{J}$ is the *objective function*, $\mathbf{x}$ is an n-dimensional vector of optimization variables, $[x_1, x_2, \ldots, x_n]^T$, and g_k are m equality constraints.[3] Now, we define the Lagrangian, $\mathcal{L}$, as follows:

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\Lambda}) = \mathcal{J}(\mathbf{x}) + \sum_{k=1}^{m} \lambda_k g_k(\mathbf{x}) \tag{3.29}$$

where $\boldsymbol{\Lambda} = [\lambda_1, \lambda_2, \ldots, \lambda_m]^T$, $m < n$, is a vector of Lagrange multipliers. If $\mathcal{J}$ and the g_k have continuous first partial derivatives and the gradients of the g_k do not vanish on the domain of $\mathcal{J}$ (the domain of $\mathcal{J}$ is assumed an open set containing all points satisfying the constraints), then the stationary points of the Lagrangian are determined by

$$\nabla_{\mathbf{x}}\mathcal{L} = \mathbf{0} \tag{3.30}$$

$$\nabla_{\boldsymbol{\Lambda}}\mathcal{L} = \mathbf{0} \tag{3.31}$$

[3] The method of Lagrange multipliers is generalized by the Karush–Kuhn–Tucker conditions [77], which can also take into account inequality constraints.

Equivalently, the stationary points of the Lagrangian must satisfy

$$\nabla \mathcal{L} = \mathbf{0} \tag{3.32}$$

Equation (3.32) gives $m + n$ unique equations for the unknowns $\mathbf{x}^\star$ and $\mathbf{\Lambda}^\star$, constituting the solutions for the optimization parameters and Lagrange multipliers at the critical point.

To determine whether some stationary point $\{\mathbf{x}^\star, \mathbf{\Lambda}^\star\}$ is a minimum, one must examine the *Hessian* [4] of the Lagrangian, also referred to as the *bordered Hessian*. If there exist vectors $\mathbf{x}^\star \in \mathbb{R}^n$ and $\mathbf{\Lambda}^\star \in \mathbb{R}^m$ such that

$$\nabla \mathcal{L}(\mathbf{x}^\star, \mathbf{\Lambda}^\star) = \mathbf{0} \tag{3.33}$$

and if, for twice continuously-differentiable $\mathcal{J}$ and g_k,

$$(-1)^m \det \begin{bmatrix} \dfrac{\partial^2 \mathcal{L}(\mathbf{x}^\star, \mathbf{\Lambda}^\star)}{\partial x_1 \partial x_1} & \cdots & \dfrac{\partial^2 \mathcal{L}(\mathbf{x}^\star, \mathbf{\Lambda}^\star)}{\partial x_1 \partial x_p} & \dfrac{\partial g_1(\mathbf{x}^\star)}{\partial x_1} & \cdots & \dfrac{\partial g_m(\mathbf{x}^\star)}{\partial x_1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \dfrac{\partial^2 \mathcal{L}(\mathbf{x}^\star, \mathbf{\Lambda}^\star)}{\partial x_p \partial x_1} & \cdots & \dfrac{\partial^2 \mathcal{L}(\mathbf{x}^\star, \mathbf{\Lambda}^\star)}{\partial x_p \partial x_p} & \dfrac{\partial g_1(\mathbf{x}^\star)}{\partial x_p} & \cdots & \dfrac{\partial g_m(\mathbf{x}^\star)}{\partial x_p} \\ \dfrac{\partial g_1(\mathbf{x}^\star)}{\partial x_1} & \cdots & \dfrac{\partial g_1(\mathbf{x}^\star)}{\partial x_p} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \dfrac{\partial g_m(\mathbf{x}^\star)}{\partial x_1} & \cdots & \dfrac{\partial g_m(\mathbf{x}^\star)}{\partial x_p} & 0 & \cdots & 0 \end{bmatrix} > 0 \tag{3.34}$$

for $p = m + 1, \ldots, n$, then $\mathcal{J}$ has a local minimum at $\mathbf{x}^\star$ such that

$$g_k(\mathbf{x}^\star) = 0, \quad k = 1, \ldots, m \tag{3.35}$$

3.6 CONTROL LYAPUNOV FUNCTIONS

Our interest in *Control Lyapunov Functions* (CLFs) stems from their application to relative spacecraft control, to be discussed in Chapter 10. We consider autonomous nonlinear systems of the form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) \tag{3.36}$$

to provide an elementary treatment of the CLF approach. It is assumed that the system of Eq. (3.36) satisfies the property

$$\mathbf{f}(\mathbf{0}) = \mathbf{0}, \tag{3.37}$$

[4] Named after the 19th century German mathematician Ludwig Otto Hesse.

i.e., the origin is an equilibrium point. The stability of the system of Eq. (3.36) at the origin can be verified by using *Lyapunov's theorem* [78], which can be stated as follows.

Theorem 3.1 (*Lyapunov's theorem [78]*). *If $V(\mathbf{x})$ is a continuously differentiable, positive definite function, defined in a domain $\mathbb{D}$ containing the origin (i.e., $V(\mathbf{0}) = 0$ and $V(\mathbf{x}) > 0$ in $\mathbb{D}$, except at $\mathbf{x} = 0$) and furthermore,*

$$\dot{V}(\mathbf{0}) = 0 \quad \textit{and} \quad \dot{V}(\mathbf{x}) < 0; \quad \mathbf{x} \neq \mathbf{0} \tag{3.38}$$

then the equilibrium point $\mathbf{x} = \mathbf{0}$ of Eq. (3.36) is asymptotically stable and $V(\mathbf{x})$ is a Lyapunov function.

For our discussion, Eq. (3.36) can be considered to be the representation of a dynamical system under the action of a feedback control law, $\mathbf{u} = \mathbf{u}(\mathbf{x})$. The CLF approach begins with the selection of a positive definite Lyapunov test function and the feedback control law is determined to satisfy Eq. (3.38) along the trajectories of Eq. (3.36), thus ensuring closed-loop stability. The choice of a CLF is not unique, but in many cases it may resemble the energy or the Hamiltonian of a closely-related open-loop (unforced) system.

3.7 LINEAR QUADRATIC REGULATION

As mentioned in the Chapter 1, the *Linear Quadratic Regulator* (LQR) is an optimal control approach based on a linear approximation of a dynamical system of the form

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u} + \mathbf{N}(\mathbf{x}) \tag{3.39}$$

where A and B are matrices of appropriate dimensions and $\mathbf{N}$ denotes the effects due to nonlinearities and unmodeled dynamics. The control law for $\mathbf{u}$ is obtained in this approach by ignoring the term $\mathbf{N}$ in Eq. (3.39) and minimizing the following *performance index*:

$$\mathcal{J} = \frac{1}{2}\int_0^{t_f} (\mathbf{x}^T Q\mathbf{x} + \mathbf{u}^T R\mathbf{u})dt \tag{3.40}$$

where t_f is the final time and $Q \geq 0$ and $R > 0$ are, respectively, the state and control *weight matrices*. For an autonomous system, constant weight matrices, and $t_f \to \infty$, the minimization of Eq. (3.40) is achieved by the following control law:

$$\mathbf{u} = -K\mathbf{x} \tag{3.41}$$

where $K = R^{-1}B^T S\mathbf{x}$ and S satisfies the *Algebraic Riccati Equation* (ARE)

$$SA + A^T S - SBR^{-1}B^T S + Q = 0 \tag{3.42}$$

The solution to the ARE is positive definite if the pair (A, B) is controllable and the pair $(A, Q^{\frac{1}{2}})$ is observable [79]. Positive definiteness of S guarantees closed-loop stability, i.e., asymptotic stability of the system

$$\dot{\mathbf{x}} = (A - BK)\mathbf{x} \tag{3.43}$$

Numerical solvers of the ARE are standard in software packages such as MATLAB®.

The *Discrete-time Linear Quadratic Regulator* (DLQR) formulation is a discretized version of the continuous LQR, and is defined as follows: Minimize

$$\mathcal{J} = \sum_{i=0}^{N-1} \mathbf{x}^T(i) Q \mathbf{x}(i) + \mathbf{u}^T(i) R \mathbf{u}(i) \tag{3.44}$$

where N indicates the total number of time steps, and $\mathbf{x}(i)$ and $\mathbf{u}(i)$ are, respectively, the state and control vectors at the ith time instant, subject to the state equation

$$\mathbf{x}(i+1) = A_d \mathbf{x}(i) + B_d \mathbf{u}(i) \tag{3.45}$$

As for the continuous-time, autonomous problem, a discrete-time algebraic Riccati equation results when N is large. The discrete-time Riccati equation is

$$A_d^T S A_d - S - A_d^T S B_d R^{-1} B_d^T S A_d + Q = 0 \tag{3.46}$$

and the feedback control law is obtained in the form

$$\mathbf{u}(i) = -K\mathbf{x}(i) = -R^{-1} B_d^T S \mathbf{x}(i) \tag{3.47}$$

3.8 KALMAN FILTERING

In 1960, Robert Kalman introduced a new approach for minimum mean-square error filtering that used state-space methods [80]. The *Kalman Filter* (KF) is a recursive scheme that propagates a current estimate of a state and the error covariance matrix of that state forward in time. The filter optimally blends the new information introduced by the measurements with old information embodied in the prior state with a Kalman gain matrix. The gain matrix balances uncertainty in the measurements with the uncertainty in the dynamics model. The KF is guaranteed to be the optimal filter (in the sense of minimizing the 2-norm-squared of the estimation error) for a linear system with linear measurements [81]. However, few systems can be accurately modeled with linear dynamics. Shortly after its inception, improvements on the Kalman filter to handle nonlinear systems were proposed. One of the most popular choices, the *Extended Kalman Filter* (EKF), was applied to relative navigation filters in LEO [43]. We will demonstrate how to use the EKF for relative spacecraft state estimation in Chapter 12.

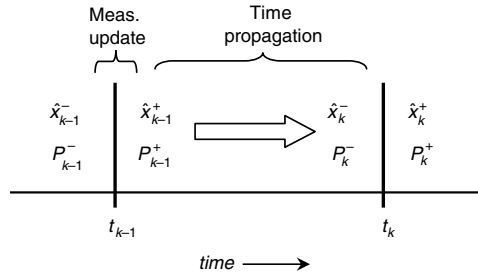

FIGURE 3.1 A Kalman filter process.

The discrete EKF is as a state estimator for systems whose state dynamics model, measurement model, or both may be nonlinear, as in Eqs. (3.48) and (3.53) [81]. The dynamics model provides the equations to propagate $\hat{\mathbf{x}}_k$, the estimate of the state $\mathbf{x}$ at time k, to time step $k+1$, producing $\hat{\mathbf{x}}_{k+1}$. The measurement model then incorporates the new sensor information to update this estimate, updating the a priori estimate $\hat{\mathbf{x}}_{k+1}^-$ to the a posteriori estimate, $\hat{\mathbf{x}}_{k+1}^+$. This process is illustrated in Fig. 3.1.

The continuous state $\mathbf{x}$ is governed by the dynamics

$$\dot{\mathbf{x}}(t_k) = \mathbf{f}(\mathbf{x}, \mathbf{u}, t_k) + \mathbf{w}(t_k) \tag{3.48}$$

where $\mathbf{u}$ is a known control input, and $\mathbf{w}(t)$ is an additive white noise that models the error accumulated by uncertainty in the dynamics during the time step. The power spectral density of this zero mean, white noise process is

$$Q = E[\mathbf{w}(t)\,\mathbf{w}(t)^T] \tag{3.49}$$

To proceed, linear expressions for the dynamics and measurement equations must be formed. In general, this requires knowledge of the probability density function [81], but the EKF approximates the nonlinear function by expanding it in a Taylor series, at each time step, about the current estimate,

$$F_k = \left.\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\right|_{\mathbf{x}=\hat{\mathbf{x}}_k} \tag{3.50}$$

The dynamics are discretized with time step Δt by forming the state transition matrix,

$$\Phi_k = e^{F_k \Delta t} \tag{3.51}$$

The cumulative effect of the white noise process $\mathbf{w}(t)$ over the time step is captured in the discrete process noise covariance matrix

$$Q_k = \int_0^{\Delta t} e^{F_k \tau} Q (e^{F_k \tau})^T \mathrm{d}\tau \tag{3.52}$$

The vector of measurements, $\mathbf{y}$,

$$\mathbf{y} = \mathbf{h}(\mathbf{x}, t) + \mathbf{v}_k \tag{3.53}$$

is modeled as a nonlinear function of the state and time, with an additive white noise process $v(t)$ that accounts for uncertainty in the sensors and their models. The measurement noise covariance matrix is defined by

$$R_k = E[\mathbf{v}_k \, \mathbf{v}_k^T] \tag{3.54}$$

The nonlinear measurement equation is also linearized about the current estimate,

$$H_k = \left. \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right|_{\mathbf{x}=\hat{\mathbf{x}}_k^-} \tag{3.55}$$

Because approximations must be made in the linearization, the EKF is a suboptimal filter, in the sense that its stability and performance are not guaranteed. Fortunately, the dynamics of orbital motion are fairly simple, and the EKF can have very good performance in space navigation applications. The discrete, linear representation of the system dynamics are

$$\mathbf{x}_k = \Phi_{k-1}\mathbf{x}_{k-1} + \mathbf{w}_{k-1} + \mathbf{u}_{k-1} \tag{3.56}$$

The confidence in the current estimate is captured in the state error covariance matrix, P,

$$P_k = \mathrm{E}[\tilde{\mathbf{x}}_k \tilde{\mathbf{x}}_k^T] = \mathrm{E}[(\hat{\mathbf{x}}_k - \mathbf{x}_k)(\hat{\mathbf{x}}_k - \mathbf{x}_k)^T] \tag{3.57}$$

where $\tilde{\mathbf{x}}_k = \hat{\mathbf{x}}_k - \mathbf{x}_k$ is the estimation error. The first step in the EKF involves propagating the state and error covariance forward in time. Equation (3.56), with zero process noise, is used to propagate the state estimate. The error covariance is propagated forward using

$$P_k^- = \Phi_{k-1} P_{k-1}^+ \Phi_{k-1}^T + Q_{k-1} \tag{3.58}$$

An alternate approach to the time propagation step involves using the nonlinear dynamics equations to propagate the state. A 4th-order Runge–Kutta integration scheme uses the nonlinear state dynamics equation

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{f}(\hat{\mathbf{x}}(t), \mathbf{u}(t)) \quad \text{for } t = t_{k-1} \rightarrow t_k \tag{3.59}$$

to find $\hat{\mathbf{x}}_k$. The state covariance is still propagated with Eq. (3.58), so the state transition matrix Φ_{k-1} must be calculated regardless of whether the linear or nonlinear state propagation is chosen.

The second step of the filter uses the measurement equation to update the a priori state $\hat{\mathbf{x}}_k^-$ to the a posteriori state $\hat{\mathbf{x}}_k^+$. When a measurement becomes

available, the new information provided by the measurement and the previous information captured in the state estimate are combined to form an updated state estimate. The Kalman gain K is the blending gain matrix that is used to weight the importance of the old and new information. The optimum gain matrix is formulated by minimizing the trace of the a posteriori state error covariance matrix P_k^+, which essentially minimizes the estimation error vector at each time step [81]. The terms in the gain matrix equation include the previous state estimate, the linearized measurement matrix, and the expected noise of the new measurements,

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1} \tag{3.60}$$

The nonlinear measurement equation is used to update the state estimate

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- - K_k(\mathbf{y}_k - \mathbf{h}_k(\hat{\mathbf{x}}_k^-)) \tag{3.61}$$

Note that the computation of the gain matrix K_k requires the linear measurement matrix H_k. The covariance is updated after the measurement with

$$P_k^+ = (I - K_k H_k) P_k^- (I - K_k H_k)^T + K_k R_k K_k^T \tag{3.62}$$

which is the Joseph form of the covariance update whose inherent symmetry makes it numerically stable [82].

3.9 THE UNSCENTED KALMAN FILTER

The development of the EKF propagation and update equations requires ignoring terms which may not be small for many nonlinear systems [81,83]. The inherent linearization in the process typically introduces significant biases in the estimation results. Although the EKF has been widely used for many years, experience has shown that it is only reliable for systems that are almost linear on the time scale of the update intervals [84]. In contrast, the *Unscented Kalman Filter* (UKF) does not require the linearization of any nonlinear functions.[5] Instead, the UKF uses a set of points, called *sigma points*, that are distributed around the current estimate. These sigma points are chosen so that their mean matches the current estimate and the covariance of the distribution of sigma points matches the current covariance of the estimate [87]. The often-stated premise of this approach is "it is easier to approximate a Gaussian distribution than it is to approximate an arbitrary nonlinear function or transformation" [87].

During the propagation step of the filter, each sigma point is propagated forward through the actual nonlinear dynamics equation. After the propagation, the set of sigma points is condensed back to a single state estimate using a weighted sum of the propagated sigma points. The propagated state covariance is also set to be the calculated covariance of the propagated sigma points.

[5] Ref. [85] presents an opposing view, countered by Ref. [86].

Similarly, the measurement update step requires updating the set of sigma points using the nonlinear measurement equation. The updated sigma points are then condensed back into a single state and their covariances used for the updated state estimate covariance.

While the EKF handles the nonlinearities by approximating them (i.e., by linearization) in the measurement update and time propagation steps, the UKF approximates the *distribution* of $\hat{\mathbf{x}}$, with the sigma points. The mean and covariance of the original state estimate are represented precisely in the distribution of the sigma points. The mean and covariance of the propagated states (i.e., either time update or measurement update) are correct to second order as well [87], which means that the UKF calculates the mean with more accuracy than the EKF, and the covariance with accuracy of at least the EKF [87]. In fact, Theorem 2 of Ref. [84] strengthens this statement to *the prediction algorithm introduces errors in estimating the mean and covariance at the fourth and higher orders in the Taylor series.*

3.9.1 The standard form of the UKF

The original form of the UKF requires augmenting the state with process and measurement noise variables. However, as shown in the equations included in the next subsection, the number of sigma points required is determined by the length of the state (or augmented state). Since the nonlinear state propagation and measurement update is performed for each sigma point, a larger state can require many more calculations. Fortunately, if the measurement and process noises are purely additive, the standard form can be reduced to what is called the additive form of the UKF (UKF-A), which does not require state augmentation [88]. The UKF-A has a smaller state and a reduced computational burden.

In the relative orbital navigation problem, detailed in Chapter 12, the process and measurement noises filter are additive, so the reduced additive form can be used for this application. The additive form of the UKF is presented in the following subsection.

3.9.2 The additive form of the UKF

The additive form of the UKF (UKF-A) is used for systems whose process and measurement noises are purely additive, as in the case discussed in Chapter 12. This form is preferred over the standard form of the UKF because it has a smaller state vector, resulting in fewer sigma points and less computation. Similar to the standard form, the UKF-A uses the nonlinear dynamics and measurement equations and employs a set of sigma points in each time propagation and measurement update step.

At each step, $2n + 1$ sigma points are required, where n indicates the length of the state vector. The sigma points that are used for the time propagation and measurement update steps are chosen to have a cumulative mean and standard deviation identical to the prior estimate. The sigma points are formed by adding and subtracting scaled columns of the matrix square root of the covariance

matrix to the original state estimate. Adding these $2n$ columns to the original state vector produces the required $2n+1$ sigma points [83,88]. This procedure is shown below in the presentation of the UKF-A algorithm.

Several weights and constants are used in the UKF algorithm. The set of scalar weights used to recombine the sigma points into the a posteriori state mean and covariance estimates, $\{W_i\}$, are defined by

$$W_0^{(m)} = \frac{\lambda_{\text{UKF}}}{n+\lambda_{\text{UKF}}} \tag{3.63}$$

$$W_0^{(c)} = \frac{\lambda_{\text{UKF}}}{n+\lambda_{\text{UKF}}} + 1 - \alpha_{\text{UKF}}^2 + \beta_{\text{UKF}} \tag{3.64}$$

$$W_i^{(m)} = \frac{1}{2(n+\lambda_{\text{UKF}})}, \quad i = 1, \ldots, 2n \tag{3.65}$$

$$W_i^{(c)} = \frac{1}{2(n+\lambda_{\text{UKF}})}, \quad i = 1, \ldots, 2n \tag{3.66}$$

The scaling parameter λ_{UKF} is $\lambda_{\text{UKF}} = \alpha_{\text{UKF}}^2(n+\kappa) - n$. The related parameter, $\gamma = \sqrt{n+\lambda_{\text{UKF}}}$, is used in the sigma point calculation. The parameter α_{UKF}, set between 10^{-4} and 1, determines the spread of points around the original estimate. The parameter β_{UKF} influences how much the prior estimate of the covariance is weighted and is typically set to 2 for state estimation problems.

The algorithm for the UKF-A is presented in Table 3.1. In several places, the nonlinear dynamics equation $\mathbf{f}$ or measurement equations $\mathbf{h}$ is applied to a matrix of vectors, producing matrix output. First the sigma points are formed around the $\hat{\mathbf{x}}_{k-1}$ state estimate (Eq. (3.69)). The sigma points are propagated (Eq. (3.70)) and weighted sums of the sigma points are used to calculate propagated state estimate and covariance, $\hat{\mathbf{x}}_k^-$ and P_k^- (Eqs. (3.71) and (3.72)). A new set of sigma points is calculated for the measurement update step (Eq. (3.73)). The measurement equation is used to create an expected measurement for each of the sigma points (Eq. (3.74)), which is then condensed with a weighted sum to create an overall, composite expected measurement, $\hat{\mathbf{y}}_k^-$ (Eq. (3.75)). Covariance matrices are calculated with the expected measurements (Eqs. (3.76) and (3.77)), and a Kalman gain matrix is formed (Eq. (3.78)). Finally, the updated state and covariance, $\hat{\mathbf{x}}_k$ and P_k, are calculated (Eqs. (3.79) and (3.80)) [88].

As a practical note, it was observed that numerical errors would sometimes introduce very small imaginary components when the matrix square root of the covariance was computed. This tends to disrupt the algorithm, so preventative measures should be included.

Another form of the UKF, called the square root form (UKF-S), has been proposed as an alternative to the additive form. The square root form makes use of *Cholesky factorization*, and in some applications may increase numerical stability and performance. A comparison of the UKF-A and the UKF-S for the relative spacecraft navigation problem is discussed in Chapter 12.

Table 3.1 UKF-A algorithm

1. The UKF-A is initialized with

$$\hat{\mathbf{x}}_0 = E[\mathbf{x}_0] \tag{3.67}$$
$$P_0 = E[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T] \tag{3.68}$$

For $k \in \{1, \ldots, \infty\}$,

2. Calculate sigma points:

$$\mathcal{X}_{k-1} = \left[\hat{\mathbf{x}}_{k-1},\ \hat{\mathbf{x}}_{k-1} + \gamma\sqrt{P_{k-1}},\ \hat{\mathbf{x}}_{k-1} - \gamma\sqrt{P_{k-1}}\right] \tag{3.69}$$

3. Propagate the sigma points with the nonlinear dynamics equation $\hat{\mathbf{x}}_{k+1} = \mathbf{f}(\hat{\mathbf{x}}_k)$ and find the covariance:

$$\mathcal{X}^*_{k|k-1} = \mathbf{f}(\mathcal{X}_{k-1}, \mathbf{u}_{k-1}) \tag{3.70}$$
$$\hat{\mathbf{x}}_k^- = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{X}^*_{i,k|k-1} \tag{3.71}$$
$$P_k^- = \sum_{i=0}^{2n} W_i^{(c)} (\mathcal{X}^*_{i,k|k-1} - \hat{\mathbf{x}}_k^-)(\mathcal{X}^*_{i,k|k-1} - \hat{\mathbf{x}}_k^-)^T + Q \tag{3.72}$$

4. Create a new set of sigma points and, with the nonlinear function measurement equation $\hat{\mathbf{y}}_\mathbf{k} = \mathbf{h}(\hat{\mathbf{x}}_\mathbf{k})$, create expected measurements:

$$\mathcal{X}_{k|k-1} = \left[\hat{\mathbf{x}}_k^-,\ \hat{\mathbf{x}}_k^- + \gamma\sqrt{P_k^-},\ \hat{\mathbf{x}}_k^- - \gamma\sqrt{P_k^-}\right] \tag{3.73}$$
$$\mathcal{Y}_{k|k-1} = \mathbf{h}(\mathcal{X}_{k|k-1}) \tag{3.74}$$
$$\hat{\mathbf{y}}_k^- = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{Y}_{i,k|k-1} \tag{3.75}$$

5. Create the Kalman gain and perform a measurement update:

$$P_{\tilde{\mathbf{y}}_k \tilde{\mathbf{y}}_k} = \sum_{i=0}^{2n} W_i^{(c)} (\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k^-)(\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k^-)^T + R \tag{3.76}$$
$$P_{x_k y_k} = \sum_{i=0}^{2n} W_i^{(c)} (\mathcal{X}_{i,k|k-1} - \hat{\mathbf{x}}_k^-)(\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}_k^-)^T \tag{3.77}$$
$$K_k = P_{x_k y_k} P^{-1}_{\tilde{\mathbf{y}}_k \tilde{\mathbf{y}}_k} \tag{3.78}$$
$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + K_k(\mathbf{y}_k - \hat{\mathbf{y}}_k^-) \tag{3.79}$$
$$P_k = P_k^- - K_k P_{\tilde{\mathbf{y}}_k \tilde{\mathbf{y}}_k} K_k^T \tag{3.80}$$

3.9.3 The square root form of the UKF

Van der Merwe and Wan introduced the Square Root form of the UKF (UKF-S) [89] to handle some of the numerical issues referred to in the previous subsection. The computational burden of the UKF-S is expected to be similar to that of the UKF-A. However, the UKF-S may provide improved numerical stability and it also guarantees that the covariance matrix will be positive semi-definite, which is required for filter stability.

The main innovation introduced in the UKF-S is the use of the Cholesky factor, S, of the covariance matrix. The Cholesky factor, S, is initialized taking

Table 3.2 UKF-S algorithm

1. The UKF-S is initialized with

$$\hat{\mathbf{x}}_0 = E[\mathbf{x}_0] \tag{3.81}$$
$$S_0 = \texttt{chol}\left(E[(\mathbf{x}_0 - \hat{\mathbf{x}}_0)(\mathbf{x}_0 - \hat{\mathbf{x}}_0)^T]\right) \tag{3.82}$$

For $k \in \{1, \ldots, \infty\}$,

2. Calculate sigma points

$$\mathcal{X}_{k-1} = [\,\hat{\mathbf{x}}_{k-1},\ \ \hat{\mathbf{x}}_{k-1} + \gamma S_k,\ \ \hat{\mathbf{x}}_{k-1} - \gamma S_k\,] \tag{3.83}$$

3. Propagate sigma points using nonlinear dynamics equation and find covariance

$$\mathcal{X}^*_{k|k-1} = \mathbf{f}(\mathcal{X}_{k-1}, \mathbf{u}_{k-1}) \tag{3.84}$$
$$\hat{\mathbf{x}}^-_k = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{X}^*_{i,k|k-1} \tag{3.85}$$
$$S^-_k = \texttt{qr}\left(\left[\sqrt{W_1^{(c)}}(\mathcal{X}^*_{1:2n,k|k-1} - \hat{\mathbf{x}}^-_k),\ \ \sqrt{Q}\right]\right) \tag{3.86}$$
$$S^-_k = \texttt{cholupdate}\left(S^-_k,\ \ \mathcal{X}^*_{0,k} - \hat{\mathbf{x}}^-_k,\ \ W_0^{(c)}\right) \tag{3.87}$$

4. Create new set of sigma points and, using the nonlinear measurement equation, and perform the Cholesky update

$$\mathcal{X}_{k|k-1} = [\,\hat{\mathbf{x}}^-_k,\ \ \hat{\mathbf{x}}^-_k + \gamma S^-_k,\ \ \hat{\mathbf{x}}^-_k - \gamma S^-_k\,] \tag{3.88}$$
$$\mathcal{Y}_{k|k-1} = \mathbf{h}(\mathcal{X}_{k|k-1}) \tag{3.89}$$
$$\hat{\mathbf{y}}^-_k = \sum_{i=0}^{2n} W_i^{(m)} \mathcal{Y}_{i,k|k-1} \tag{3.90}$$
$$S_{\tilde{y}_k} = \texttt{qr}\left(\left[\sqrt{W_1^{(c)}}(\mathcal{Y}_{1:2n,k} - \hat{\mathbf{y}}_k),\ \ \sqrt{R}\right]\right) \tag{3.91}$$
$$S_{\tilde{y}_k} = \texttt{cholupdate}\left(S_{\tilde{y}_k},\ \ \mathcal{Y}_{0,k} - \hat{\mathbf{y}}_k,\ \ W_0^{(c)}\right) \tag{3.92}$$

5. Create the Kalman gain and perform the measurement update

$$P_{x_k y_k} = \sum_{i=0}^{2n} W_i^{(c)} (\mathcal{X}_{i,k|k-1} - \hat{\mathbf{x}}^-_k)(\mathcal{Y}_{i,k|k-1} - \hat{\mathbf{y}}^-_k)^T \tag{3.93}$$
$$K_k = (P_{x_k y_k} / S^T_{\tilde{y}_k}) / S_{\tilde{y}_k} \tag{3.94}$$
$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}^-_k + K_k(\mathbf{y}_k - \hat{\mathbf{y}}^-_k) \tag{3.95}$$
$$U = K_k\, S_{\tilde{y}_k} \tag{3.96}$$
$$S_k = \texttt{cholupdate}(S^-_k, U, -1) \tag{3.97}$$

the matrix square root of the initial P with a Cholesky factorization. Three linear algebra techniques are recommended by van der Merwe for use in the UKF-S [89]:

- **QR decomposition** The QR decomposition factors a matrix A into the product of an orthogonal matrix, Q, and an upper triangular matrix, R, so $A^T = QR$. The upper triangular part of R is the transpose of the Cholesky factor of a matrix P, defined such that $P = AA^T$. The MATLAB® function `qr` returns the *lower* Cholesky factor. The transpose of the output of `qr` should be used to get the *upper* Cholesky

factor required in this algorithm.

- **Cholesky factor updating** Consider a matrix $P = AA^T$ that has a Cholesky factor of S. If the matrix P is updated so that $P^+ = P \pm \sqrt{\nu}\mathbf{u}\mathbf{u}^T$, the Cholesky factor of the updated P^+ is found using the `cholupdate` MATLAB® function, $\texttt{chol}(P^+) = \texttt{cholupdate}(S, \mathbf{u}, \pm\nu)$. When $\mathbf{u}$ is a matrix, the function performs a Cholesky update for each of the columns of $\mathbf{u}$.
- **Efficient least squares** The MATLAB® function "/" finds the solution to $(AA^T)\mathbf{v} = A^T\mathbf{b}$ using an efficient QR decomposition method.

These three techniques were incorporated into the Square Root Unscented Kalman Filter used in Chapter 12. The UKF-S algorithm is summarized in Table 3.2.

SUMMARY

We presented some key tools of analytical mechanics, optimization, control and estimation, which supplement the introductory material on astrodynamics presented in Chapter 2. The tools provided in the chapter will be extensively used in subsequent discussions. Hamiltonian and Lagrangian mechanics will be used for perturbation modeling, relative motion modeling, mean-to-osculating element conversion using the Brouwer transformation, and averaging; optimization theory, including the concept of bordered Hessians, will be used for developing optimal formation-keeping maneuvers; Linear Quadratic Regulators and Control Lyapunov Functions will be used to control spacecraft formations; and Kalman filtering will be extensively applied to the problem of relative navigation.