Robotics

Stewart Nash

February 21, 2025

Contents

Chapter 1

Dynamics

1.1 Kinematics

Take a vector with components $\mathbf{P} = (a_x, b_y, c_z)$. A scale factor w can be added (to the matrix form) to give

$$\mathbf{P} = \begin{bmatrix} P_x \\ P_y \\ P_z \\ w \end{bmatrix} \tag{1.1}$$

where $(a_x, b_y, c_z) = (P_x/w, P_y/w, P_z/w)$. A direction vector can be represented by a scale factor of zero ($w = 0$).

A universe reference frame is represented by $F_{x,y,z}$ and a moving frame is represented by $F_{n,o,a}$ where the letters n, o, and a come from the words normal, orientation and approach. Relative to the gripper, the z-axis is the approach axis by which the gripper approaches an object. The orientation with which the gripper frame approaches the part is the orientation axis. The normal-axis or x-axis is normal to both. A fourth vector which gives the location of a frame relative to a reference frame can be added to the vectors representing the components of the n-, o-, and a-axes to give a homogeneous matrix representation of this relative frame

$$F = \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.2}$$

Pre-multiplying the frame matrix by the transformation matrix will yield the new location of the frame.

Rotation matrices about the x-, y- and z-axes are given by

$$
\begin{aligned}
\mathrm{rot}(x,\theta) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \\
\mathrm{rot}(y,\theta) &= \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \\
\mathrm{rot}(z,\theta) &= \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}
\end{aligned}
\tag{1.3}
$$

Denoting the transformation of frame R relative to frame U (universe) as ${}^{U}T_R$, denoting the p relative to the frame R as ${}^{R}p = p_{noa}$, and denoting p relative to frame U as ${}^{U}p = p_{xyz}$ we have

$$
{}^{U}p = {}^{U}T_R \times {}^{R}p \tag{1.4}
$$

1.2 Differential Motion

1.3 Dynamic Analysis

The Lagrangian is given by $L = K - P$ where K and P are the kinetic and potential energy of a system, respectively. If F_i is the summation of all external forces for a linear motion and T_i is the summation of all external forces for a linear motion and T_i is the summation of all external torques for a rotational motion, then

$$
\begin{aligned}
F_i &= \frac{\partial}{\partial t}\left(\frac{\partial L}{\partial \dot{x}_i}\right) - \frac{\partial L}{\partial x_i} \\
T_i &= \frac{\partial}{\partial t}\left(\frac{\partial L}{\partial \dot{\theta}_i}\right) - \frac{\partial L}{\partial \theta_i}
\end{aligned}
\tag{1.5}
$$

The equation of motion for a 2-DOF system is given by

$$
\begin{bmatrix} T_i \\ T_j \end{bmatrix} = \begin{bmatrix} D_{ii} & D_{ij} \\ D_{ji} & D_{jj} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_i \\ \ddot{\theta}_j \end{bmatrix} + \begin{bmatrix} D_{iii} & D_{ijj} \\ D_{jii} & D_{jjj} \end{bmatrix} \begin{bmatrix} \dot{\theta}_i \\ \dot{\theta}_j \end{bmatrix} + \begin{bmatrix} D_{iij} & D_{iji} \\ D_{jij} & D_{jji} \end{bmatrix} \begin{bmatrix} \dot{\theta}_i\dot{\theta}_j \\ \dot{\theta}_j\dot{\theta}_i \end{bmatrix} + \begin{bmatrix} D_i \\ D_j \end{bmatrix} \tag{1.6}
$$

Where the coefficient D_{ii} is the effective inertia at joint i, such that an acceleration at joint i causes a torque at joint i equal to $D_{ii}\ddot{\theta}_i$, and the coefficient D_{ij} is the coupling inertia between joints i and j such that an acceleration at joint i or j causes a torque at joint j or i equal to $D_{ij}\ddot{\theta}_j$ or $D_{ji}\ddot{\theta}_i$. $D_{ijj}\dot{\theta}_j^2$ terms represent centripetal forces acting at joint i due to a velocity at joint j. All terms with $\dot{\theta}_i\dot{\theta}_j$ represent Coriolis accelerations and, when multiplied by corresponding inertias, represent Coriolis forces. D_i represents gravity forces at joint i.

1.4 Trajectory Planning

Chapter 2

Controls

2.1 State Variable Representation

Powerful tools from matrix algebra can be used to solve sets of first-order differential equations. That is why it is sometimes helpful to transform a system described by nth-order differential equations into a system of first-order differential equations.

2.1.1 Systems Modeled by Linear Differential Equations

Consider a system represented by a nth-order, single-input linear constant coefficient differential equation

$$\sum_{i=0}^{n} a_i \frac{d^i y}{dt^i} = u \tag{2.1}$$

This equation can be replaced by n first-order differential equations

$$\begin{cases} \dfrac{dx_k}{dt} = x_{k+1},\ 1 < k < n \\ \dfrac{dx_n}{dt} = \dfrac{1}{a_n}\left[\displaystyle\sum_{k=0}^{n-1} a_k x_{k+1}\right] + \dfrac{1}{a_n} u \end{cases} \tag{2.2}$$

where $x_1 \equiv y$ and $i, k \in \mathbb{W}$. This can be written as a matrix equation

$$\begin{bmatrix} \frac{dx_1}{dt} \\ \frac{dx_2}{dt} \\ \vdots \\ \frac{dx_n}{dt} \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -\frac{a_0}{a_n} & -\frac{a_1}{a_n} & -\frac{a_2}{a_n} & \cdots & -\frac{a_{n-1}}{a_n} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \frac{1}{a_n} \end{bmatrix} u \tag{2.3}$$

or

$$\frac{d\mathbf{x}}{dt} = A\mathbf{x} + \mathbf{b}u \tag{2.4}$$

A multi-input-multi-output (MIMO) system can be represented by

$$\begin{bmatrix} \frac{dx_1}{dt} \\ \frac{dx_2}{dt} \\ \vdots \\ \frac{dx_n}{dt} \end{bmatrix} \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1r} \\ b_{21} & b_{22} & \cdots & b_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{nr} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_r \end{bmatrix} \tag{2.5}$$

or

$$\frac{d\mathbf{x}}{dt} = A\mathbf{x} + B\mathbf{u} \tag{2.6}$$

where $\mathbf{u}$ is an r-vector of input functions.
Let $\mathbf{\Phi}$ be the $n \times n$ *transition matrix* of the differential equation given above which is described by the matrix equation

$$\frac{d\mathbf{\Phi}}{dt} = A\mathbf{\Phi} \tag{2.7}$$

If $\mathbf{\Phi}(0) = I$ (initial condition) then $\mathbf{\Phi}(t) = e^{At}$ where

$$e^{At} = \sum_{n=0}^{\infty} \frac{A^n t^n}{n!} \tag{2.8}$$

The solution to (2.6) on the interval $0 \le t < \infty$ is given by

$$\mathbf{x}(t) = e^{At}\mathbf{x}(0) + \int_0^t e^{A(t-\tau)} B\mathbf{u}(\tau)\, d\tau \tag{2.9}$$

2.1.2 Systems Modeled by Constant Coefficient Linear Difference Equations

An n-th order (linear constant-coefficient) difference equation is given by

$$\sum_{i=0}^{n} a_i y(k+i) = \sum_{i=0}^{m} b_i u(k+i) \tag{2.10}$$

Define a shift operator by the equation

$$Z[y(k)] \equiv y(k+1) \tag{2.11}$$

The n-th order linear constant-coefficient difference equation

$$y(k+n) + \sum_{i=0}^{n-1} a_i y(k+i) = u(k) \tag{2.12}$$

can be written as

$$(Z^n + \sum_{i=0}^{n-1} a_i Z^i)[y(k)] = u(k) \tag{2.13}$$

The characteristic equation of this difference equation is

$$Z^n + \sum_{i=0}^{n-1} a_i Z^i = 0 \tag{2.14}$$

2.2 Signal Flow Graphs

2.3 Nyquist Analysis and Design

2.3.1 Mapping

Let us consider a complex variable $s = \sigma + j\omega$. We will denote a complex transfer function of s as $P(s)$. Let us also consider a complex variable $z = \mu + j\nu$ and denote a discrete-time (system) complex transfer function of z as $P(z)$. For the first variable and transfer function we create two graphs: (1) the s-plane which has $j\omega$ on the ordinate and σ on the abscissa, (2) the $P(s)$-plane which has $\operatorname{Im} P$ on the ordinate and $\operatorname{Re} P$ on the abscissa. The function P maps points of the s-plane into the $P(s)$-plane. Similarly, $P(z)$ is a mapping or transformation from the z-plane to the $P(z)$-plane. For Nyquist stability plots, the locus of points in the s-plane which are chosen to map is called the Nyquist path. A polar plot is constructed in the $P(s)$-plane by taking $s = 0 + j\omega$.

2.4 Root Locus Analysis and Design

2.5 Bode Analysis and Design

2.6 Miscellaneous Topics

2.6.1 Non-linear Control Systems

2.6.2 Controllability and Observability

2.6.3 State Feedback

2.6.4 Random Inputs

2.6.5 Optimal Control Systems

2.6.6 Adaptive Control Systems

Chapter 3

Image Processing