Summary of
TTK4130 Modeling and Simulation

Morten Fyhn Amundsen

NTNU

May 19, 2015

Contents

1 Introduction

I can't be bothered to write much here. You know what you're doing.

1.1 Todo

- Force and torque matrices $\boldsymbol{F}_{bc}^{b}$.
- p. 533

2 Things that ought to be obvious

Eigenvalues Values of λ such that: $\det \lambda \mathbf{I} - \boldsymbol{A} = 0$.

Eigenvectors Vectors $\mathbf{v}$ such that: $(\boldsymbol{A} - \lambda \mathbf{I})\mathbf{v} = 0$.

Differentiation of x^2: $\frac{\mathrm{d}x^2}{\mathrm{d}t} = 2\dot{x}x$

Relation of velocity to ang. vel.: $v = r\omega$

Matrix transposes: $(\boldsymbol{A} + \boldsymbol{B})^{\mathrm{T}} = \boldsymbol{A}^{\mathrm{T}} + \boldsymbol{B}^{\mathrm{T}}, \quad (\boldsymbol{A}\boldsymbol{B})^{\mathrm{T}} = \boldsymbol{B}^{\mathrm{T}}\boldsymbol{A}^{\mathrm{T}}$

Inverse of 2×2 matrix:

$$\boldsymbol{A}^{-1} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} = \frac{1}{\det \boldsymbol{A}} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$$

3 Energy-flow modelling (19)

A modular, rather than monolithic approach to modelling a dynamic system. Modules are connected with one *potential* and one *flow* variable, similar to their physical connection. At a connection, potentials must be equal, and flows must sum to zero.

Domain	Potential	Flow	Result
Translation	Velocity [m/s]	Force [N]	Power [W]
Rotation	Angular velocity [/s]	Torque [Nm]	Power [W]
Electrical	Voltage [V]	Current [A]	Power [W]
Magnetic	Mag.mot. force [A]	Mag. flux rate [V]	Power [W]
Hydraulic	Pressure [Pa]	Volume flow rate [m^3/s]	Power [W]
Thermal	Temperature [K]	Heat flow rate [J/Ks]	Power [W]
Chemical	Chem. potential [J/mol]	Molar flow rate [mol/s]	Power [W]

4 Energy (*Lyapunov*) functions and passivity (46)

Energy functions describe the change of internal 'energy' in a system. *Decreasing energy implies stability.*

The time derivative of the energy function is

$$\dot{V} = \frac{\partial V}{\partial t} + \frac{\partial V}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}, \mathbf{u}, t). \tag{1}$$

Passivity describes whether a system produces energy to its surroundings. (Example: Power in a resistor: $P = ui = Ri^2 \implies P \geq 0 \implies$ Passive!)

- Connected stable systems are not always stable...
- Connected passive systems are always passive, and therefore stable!

A system with input u and output y is passive if

$$\int_0^t y(\tau)u(\tau)\,\mathrm{d}\tau \geq -E_0 \;\; \forall \;\; t \geq 0 \tag{2}$$

for all inputs.

4.1 Positive realness of a transfer function (56)

A system is passive iff its transfer function is positive real.

Definition: The t.f. $H(s)$ (rational or not) is positive real if:

1. $H(s)$ is analytic $\;\forall\; \Re[s] > 0$
2. $H(s)$ is real for all positive and real s.
3. $\Re[H(s)] \geq 0 \;\forall\; \Re[s] > 0$.

But this is easier to use:

Theorem: A rational, proper t.f. $H(s)$ is positive real iff:[1]

1. $H(s)$ has no poles in the right half plane.
2. $\Re[H(\jmath\omega)] \geq 0 \;\; \forall \;\; \omega$ such that $\jmath\omega$ is not a pole of $H(s)$.
3. If $\jmath\omega_0$ is a pole in $H(s)$, it is a simple pole, and the residual in $s = \jmath\omega_0$ is positive and real, i.e.,

$$\operatorname*{Res}_{s=\jmath\omega_0} H(s) = \lim_{s \to \jmath\omega_0} (s - \jmath\omega_0) H(s) > 0$$

(And some stuff about poles at infinity.)

5 Simulation (509)

5.1 Notation (517)

$$\underbrace{\underbrace{\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y}, t)}_{\text{ODE}}, \quad \mathbf{y}(t_0) = \mathbf{y}_0}_{\text{IVP}} \tag{3}$$

Given this, we want to approximate $\mathbf{y}(t)$.

5.2 Error (517)

The *local solution* $\mathbf{y}_L(t_n; t)$ is the exact solution of (3) with initial condition $\mathbf{y}_n$ at time t_n:

$$\dot{\mathbf{y}}_L(t_n; t) = \mathbf{f}[\mathbf{y}_L(t_n; t)], \quad \mathbf{y}_L(t_n; t_n) = \mathbf{y}_n \tag{4}$$

I.e., if the system really is in state $\mathbf{y}_n$ at time t_n, the local solution is the true behaviour.

This lets us define the *local error*: the difference between the computed solution and the local solution at t_{n+1}:

$$\mathbf{e}_{n+1} = \mathbf{y}_{n+1} - \mathbf{y}_L(t_n; t_{n+1}) \tag{5}$$

The *global error* is the difference between the computed solution and the exact solution at t_{n+1}:

$$\mathbf{E}_{n+1} = \mathbf{y}_{n+1} - \mathbf{y}(t_{n+1}) \tag{6}$$

[1] if and only if

5.3 Order of a one-step method (517)

A method is of order p if $\mathbf{e}_{n+1} = O(h^{p+1})$.

Given the IVP in (3) and a one-step method (of step length h)

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h\phi(\mathbf{y}_n, t_n), \tag{7}$$

if then

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h\mathbf{f}(\mathbf{y}_n, t) + \frac{h^2}{2}\frac{\mathrm{d}\mathbf{f}(\mathbf{y}_n, t)}{\mathrm{d}t} + \ldots + \frac{h^p}{p!}\frac{\mathrm{d}^{p-1}\mathbf{f}(\mathbf{y}_n, t)}{\mathrm{d}t^{p-1}} + O(h^{p+1}) \tag{8}$$

holds, the local error is $O(h^{p+1})$ and the method is of order p.

5.4 Linearisation (518)

The linearisation of (3) around $\mathbf{y}^*$ is

$$\Delta\dot{\mathbf{y}} = \boldsymbol{J}\Delta\mathbf{y}, \tag{9}$$

where $\boldsymbol{J}$ is

$$\boldsymbol{J} = \left.\frac{\partial\mathbf{f}(\mathbf{y}, t)}{\partial\mathbf{y}}\right|_{\mathbf{y}=\mathbf{y}^*} = \left\{\left.\frac{\partial f_i(\mathbf{y}, t)}{\partial y_j}\right|_{\mathbf{y}=\mathbf{y}^*}\right\}. \tag{10}$$

5.5 Stability (531)

5.5.1 Scalar test system

Given the scalar test system

$$\dot{y} = \lambda y \tag{11}$$

we get

$$y_{n+1} = R(h\lambda)y_n \tag{12}$$

where $R(h\lambda)$ is the stability function of a numerical method.

5.5.2 Requirement for stability

The numerical solution $\Delta\mathbf{y}_n$ of the linearised system is stable if:

$$|R(h\lambda_i)| \leq 1 \tag{13}$$

for all eigenvalues λ_i of $\boldsymbol{J}$.

5.6 Explicit Runge-Kutta methods (ERK) (526)

Given the IVP in (3) and the generic one-step method in (7), the general ERK is

$$\begin{aligned}
\mathbf{k}_1 &= \mathbf{f}(\mathbf{y}_n,\ \ t_n) \\
\mathbf{k}_2 &= \mathbf{f}(\mathbf{y}_n + ha_{21}\mathbf{k}_1,\ \ t_n + c_2 h) \\
\mathbf{k}_3 &= \mathbf{f}(\mathbf{y}_n + h(a_{31}\mathbf{k}_1 + a_{32}\mathbf{k}_2),\ \ t_n + c_3 h) \\
&\ \vdots \\
\mathbf{k}_\sigma &= \mathbf{f}(\mathbf{y}_n + h(a_{\sigma 1}\mathbf{k}_1 + \ldots + a_{\sigma,\sigma-1}\mathbf{k}_{\sigma-1}),\ \ t_n + c_\sigma h) \\
\mathbf{y}_{n+1} &= \mathbf{y}_n + h(b_1\mathbf{k}_1 + \ldots + b_\sigma\mathbf{k}_\sigma).
\end{aligned} \tag{14}$$

The weights can be arranged in a *Butcher array*:

$$\begin{array}{c|ccccc}
0 & & & & & \\
c_2 & a_{21} & & & & \\
c_3 & a_{31} & a_{32} & & & \\
\vdots & \vdots & \vdots & \ddots & & \\
c_\sigma & a_{\sigma 1} & a_{\sigma 2} & \ldots & a_{\sigma,\sigma-1} & \\
\hline
 & b_1 & b_2 & \ldots & b_{\sigma-1} & b_\sigma
\end{array}
\qquad \text{or} \qquad
\begin{array}{c|c}
\mathbf{c} & \boldsymbol{A} \\
\hline
 & \mathbf{b}^{\mathrm{T}}
\end{array} \tag{15}$$

Where σ is the number of stages.

5.6.1 ERK stability

The stability function of an ERK is

$$R_E(h\lambda) = \det\left[\mathbf{I} - \lambda h\left(\boldsymbol{A} - \mathbf{1}\mathbf{b}^{\mathrm{T}}\right)\right] \tag{16}$$

where $\boldsymbol{A}$ and $\mathbf{b}$ are given by the Butcher array.

For ERKs of order $p = \sigma \leq 4$:

$$R_E(h\lambda) = 1 + h\lambda + \cdots + \frac{h^p\lambda^p}{p!} \tag{17}$$

Valid for Euler's method, Modified Euler, RK4, and at least one other. These are the methods that are actually used.

5.6.2 Euler's method (Forward Euler)

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h\mathbf{f}(\mathbf{y}_n, t_n) \iff \begin{array}{c|c} 0 & \\ \hline & 1 \end{array} \tag{18}$$

5.6.3 Improved Euler

$$\begin{aligned}\mathbf{k}_1 &= \mathbf{f}(\mathbf{y}_n, t_n) \\ \mathbf{k}_2 &= \mathbf{f}(\mathbf{y}_n + h\mathbf{k}_1, t_n + h) \\ \mathbf{y}_{n+1} &= \mathbf{y}_n + \frac{h}{2}(\mathbf{k}_1 + \mathbf{k}_2)\end{aligned} \iff \begin{array}{c|cc} 0 & & \\ 1 & 1 & \\ \hline & 1/2 & 1/2 \end{array} \tag{19}$$

5.6.4 Modified Euler (Explicit midpoint)

$$\begin{aligned}\mathbf{k}_1 &= \mathbf{f}(\mathbf{y}_n, t_n) \\ \mathbf{k}_2 &= \mathbf{f}(\mathbf{y}_n + \frac{h}{2}\mathbf{k}_1, t_n + \frac{h}{2}) \\ \mathbf{y}_{n+1} &= \mathbf{y}_n + h\mathbf{k}_2\end{aligned} \iff \begin{array}{c|cc} 0 & & \\ 1/2 & 1/2 & \\ \hline & 0 & 1 \end{array} \tag{20}$$

5.6.5 Heun's method

$$\begin{array}{c|ccc} 0 & & & \\ 1/3 & 1/3 & & \\ 2/3 & 0 & 2/3 & \\ \hline & 1/4 & 0 & 3/4 \end{array} \tag{21}$$

5.6.6 Runge-Kutta 4

$$\begin{array}{c|cccc} 0 & & & & \\ 1/2 & 1/2 & & & \\ 1/2 & 0 & 1/2 & & \\ 1 & 0 & 0 & 1 & \\ \hline & 1/6 & 2/6 & 2/6 & 1/6 \end{array} \tag{22}$$

5.7 Implicit Runge-Kutta methods (IRK) (534)

Defined as

$$\begin{aligned}\mathbf{k}_1 &= \mathbf{f}(\mathbf{y}_n + h(a_{11}\mathbf{k}_1 + \ldots + a_{1\sigma}\mathbf{k}_\sigma),\ \ t_n + c_1 h) \\ &\vdots \\ \mathbf{k}_\sigma &= \mathbf{f}(\mathbf{y}_n + h(a_{\sigma 1}\mathbf{k}_1 + \ldots + a_{\sigma\sigma}\mathbf{k}_\sigma),\ \ t_n + c_\sigma h) \\ \mathbf{y}_{n+1} &= \mathbf{y}_n + h(b_1\mathbf{k}_1 + \ldots + b_\sigma\mathbf{k}_\sigma)\end{aligned} \tag{23}$$

$$\begin{array}{c|ccccc} c_1 & a_{11} & a_{12} & \dots & \dots & a_{1\sigma} \\ c_2 & a_{21} & a_{22} & \dots & \dots & a_{1\sigma} \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ c_\sigma & a_{\sigma 1} & a_{\sigma 2} & \dots & a_{\sigma,\sigma-1} & a_{\sigma\sigma} \\ \hline & b_1 & b_2 & \dots & b_{\sigma-1} & b_\sigma \end{array} \quad \text{or} \quad \begin{array}{c|c} \mathbf{c} & \boldsymbol{A} \\ \hline & \mathbf{b}^{\mathrm{T}} \end{array} \tag{24}$$

IRKs efficiently solve stiff systems (i.e. systems with a large spread in eigenvalues). ERKs do not. Stability is generally the motivation for using IRKs, not accuracy.

5.7.1 IRK stability

The stability of an IRK is given by

$$R(h\lambda) = 1 + \lambda h \mathbf{b}^{\mathrm{T}}(\mathbf{I} - h\lambda \boldsymbol{A})^{-1}\mathbf{1} \tag{25}$$

$$= \frac{\det\left(\mathbf{I} - \lambda h\left[\boldsymbol{A} - \mathbf{1}\mathbf{b}^{\mathrm{T}}\right]\right)}{\det(\mathbf{I} - \lambda h \boldsymbol{A})} \tag{26}$$

5.7.2 Implicit Euler (Radau IIA)

$$\begin{array}{rcl} \mathbf{k}_1 & = & \mathbf{f}(\mathbf{y}_n + h\mathbf{k}_1, t_{n+1}) \\ \mathbf{y}_{n+1} & = & \mathbf{y}_n + h\mathbf{k}_1 \end{array} \iff \begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array} \tag{27}$$

This is stable for all eigenvalues *outside* a unit circle with center $(1, 0)$!

5.8 A- and L-stability (546)

A-stability: A method is A-stable if $|R(\lambda h)| \leq 1 \;\forall\; \Re(\lambda) \leq 0$.

This implies that the method is stable for all stable test systems. Thus, it is also stable for systems with very fast dynamics compared to h, but note that aliasing can occur. High frequency oscillations appear in the solution as oscillations slower than the Nyquist frequency $\frac{\pi}{h}$.

Note that no explicit methods are A-stable.

L-stability: A method is L-stable if it is A-stable, and $\lim_{\omega\to\infty} |R(\jmath\omega h)| = 0$ for all systems $\dot{y} = \lambda y$ where $\lambda = \jmath\omega$.

L-stable methods dampen out the inaccurate, fast dynamics that can occur with A-stable methods.

5.9 Padé-approximations (548)

The local solution of the test system $\dot{y} = \lambda y$ over a time step, and the numerical solution of the same system, is

$$\dot{y}_L(t_n; t_{n+1}) = e^{\lambda h} y_n \tag{28}$$

$$y_{n+1} = R(\lambda h) y_n \tag{29}$$

The accuracy of the numerical solution depends on how well $R(\lambda h)$ approximates the exponential function.

A Padé approximation $P_m^k(s)$ is the best rational approximation of e^s with a numerator of order k and a denominator of order m.

6 Rotation matrices (218)

6.1 Vectors (209)

The vector

$$\vec{v} = v_1\vec{a_1} + v_2\vec{a_2} + v_3\vec{a_3} \tag{30}$$

has 'built in' information about its frame of reference (it is *coordinate free*). However, the coordinate vector

$$\mathbf{u}^a = \begin{pmatrix} u_1 & u_2 & u_3 \end{pmatrix}^{\mathrm{T}} \tag{31}$$

does not, and we must indicate which frame of reference it is given in (frame a in this case).

6.2 Skew-symmetric form (211)

$$\mathbf{u}^\times := \begin{pmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{pmatrix} \tag{32}$$

6.3 Vector cross product (211)

$$\mathbf{w} = \mathbf{u}^\times \mathbf{v} \iff \vec{w} = \vec{u} \times \vec{v} \tag{33}$$

Alternatively:

$$\mathbf{w} = \mathbf{u}^\times \mathbf{v} = \begin{pmatrix} u_2v_3 - u_3v_2 \\ u_3v_1 - u_1v_3 \\ u_1v_2 - u_2v_1 \end{pmatrix} \tag{34}$$

6.4 Properties of the rotation matrix (219)

Notation: $\mathbf{v}^a = \begin{pmatrix} v_1^a & v_2^a & v_3^a \end{pmatrix}^{\mathrm{T}}$ is vector $\mathbf{v}$ given in the coordinates of a.

$$
\begin{aligned}
\mathbf{v}^a &= \boldsymbol{R}_b^a \mathbf{v}^b \\
\boldsymbol{R}_a^b &= (\boldsymbol{R}_b^a)^{-1} = (\boldsymbol{R}_b^a)^{\mathrm{T}} \\
\boldsymbol{R}_c^a &= \boldsymbol{R}_b^a \boldsymbol{R}_c^b \\
(\mathbf{u}^b)^\times &= \boldsymbol{R}_a^b (\mathbf{u}^a)^\times \boldsymbol{R}_b^a
\end{aligned}
\tag{35}
$$

Definition: A matrix $\boldsymbol{R}$ is a rotation matrix iff $\boldsymbol{R} \in SO(3)$:

$$
SO(3) = \{\boldsymbol{R} \mid \boldsymbol{R} \in \mathbb{R}^{3\times 3},\ \boldsymbol{R}^{\mathrm{T}}\boldsymbol{R} = \mathbf{I},\ \det \boldsymbol{R} = 1\}
\tag{36}
$$

6.5 Simple rotations (221)

A *simple rotation* is a rotation about a fixed axis. Rotation matrices for rotation around the x, y, and z axes, respectively, are as follows:

$$
\boldsymbol{R}_x(\phi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{pmatrix}
\tag{37}
$$

$$
\boldsymbol{R}_y(\theta) = \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix}
\tag{38}
$$

$$
\boldsymbol{R}_z(\psi) = \begin{pmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix}
\tag{39}
$$

6.6 Euler angles (224)

A parametrisation of rotation about three axes. Three parameters ψ, θ, ϕ. Singularities exist.

Roll, pitch, yaw: A rotation ψ about the z-axis, then θ about the (rotated) y-axis, then ϕ about the (also rotated) x-axis.

$$
\begin{aligned}
\boldsymbol{R}_b^a &= \boldsymbol{R}_z(\psi)\boldsymbol{R}_y(\theta)\boldsymbol{R}_{\underline{x}}(\phi) \\
&= \begin{pmatrix} \mathrm{c}\,\psi\,\mathrm{s}\,\theta & \mathrm{c}\,\psi\,\mathrm{s}\,\theta\,\mathrm{s}\,\phi - \mathrm{s}\,\psi\,\mathrm{c}\,\phi & \mathrm{s}\,\psi\,\mathrm{s}\,\phi + \mathrm{c}\,\psi\,\mathrm{s}\,\theta\,\mathrm{c}\,\phi \\ \mathrm{s}\,\psi\,\mathrm{c}\,\theta & \mathrm{c}\,\psi\,\mathrm{c}\,\phi + \mathrm{s}\,\psi\,\mathrm{s}\,\theta\,\mathrm{s}\,\phi & \mathrm{s}\,\psi\,\mathrm{s}\,\theta\,\mathrm{c}\,\phi - \mathrm{c}\,\psi\,\mathrm{s}\,\phi \\ -\mathrm{s}\,\theta & \mathrm{c}\,\theta\,\mathrm{s}\,\phi & \mathrm{c}\,\theta\,\mathrm{c}\,\phi \end{pmatrix}
\end{aligned}
\tag{40}
$$

Classical Euler angles: A rotation ψ about the z-axis, then θ about the (rotated) y-axis, then ϕ about the (also rotated) z-axis.

$$\begin{aligned}\boldsymbol{R}_b^a &= \boldsymbol{R}_z(\psi)\boldsymbol{R}_y(\theta)\boldsymbol{R}_{\underline{z}}(\phi) \\ &= \begin{pmatrix} \mathrm{c}\,\psi\,\mathrm{c}\,\theta\,\mathrm{c}\,\phi - \mathrm{s}\,\psi\,\mathrm{s}\,\phi & -\mathrm{c}\,\psi\,\mathrm{c}\,\theta\,\mathrm{s}\,\phi - \mathrm{s}\,\psi\,\mathrm{c}\,\phi & \mathrm{c}\,\psi\,\mathrm{s}\,\theta \\ \mathrm{s}\,\psi\,\mathrm{c}\,\theta\,\mathrm{c}\,\phi + \mathrm{c}\,\psi\,\mathrm{s}\,\phi & \mathrm{c}\,\psi\,\mathrm{c}\,\phi - \mathrm{s}\,\psi\,\mathrm{c}\,\theta\,\mathrm{s}\,\phi & \mathrm{s}\,\psi\,\mathrm{s}\,\theta \\ -\mathrm{s}\,\theta\,\mathrm{c}\,\phi & \mathrm{s}\,\theta\,\mathrm{s}\,\phi & \mathrm{c}\,\theta \end{pmatrix}\end{aligned} \tag{41}$$

6.7 Angle-axis description of rotation (226)

Describes a rotation by an axis of rotation $\mathbf{k}$ about which we move by an angle θ (four parameters k_1, k_2, k_3, θ).

$$\boldsymbol{R}_b^a\mathbf{k} = \mathbf{k}, \quad \mathbf{k}^a = \mathbf{k}^b = \mathbf{k} \tag{42}$$

With this method, the rotation matrix becomes

$$\boldsymbol{R}_b^a = \cos\theta\mathbf{I} + \sin\theta(\mathbf{k}^a)^\times + (1-\cos\theta)\mathbf{k}^a(\mathbf{k}^a)^\mathrm{T} \tag{43}$$

6.8 Euler parameters (231)

The parameters

$$\eta = \cos\frac{\theta}{2}, \quad \boldsymbol{\epsilon} = \mathbf{k}\sin\frac{\theta}{2} \tag{44}$$

lead to the rotation matrix

$$\boldsymbol{R}_e(\eta, \boldsymbol{\epsilon}) = \mathbf{I} + 2\eta\boldsymbol{\epsilon}^\times + 2\boldsymbol{\epsilon}^\times\boldsymbol{\epsilon}^\times. \tag{45}$$

This is nice, because:

- No singularities!
- No trigonometry!
- $\eta^2 + \vec{\epsilon}\cdot\vec{\epsilon} = 1$: Easy to normalise (avoid roundoff errors).

6.9 Homogenous transformation matrices (223)

The position and orientation of frame b relative to frame a is given by the homogenous transformation matrix

$$\boldsymbol{T}_b^a = \begin{pmatrix} \boldsymbol{R}_b^a & \mathbf{r}_{ab}^a \\ \mathbf{0}^\mathrm{T} & 1 \end{pmatrix} \in SE(3) \tag{46}$$

where $\mathbf{r}^a_{ab}$ is the position of frame b relative to frame a, expressed in the coordinates of frame a. (Position as in position of the origin.) The set $SE(3)$ is defined as:

$$SE(3) = \left\{ \boldsymbol{T} \mid \boldsymbol{T} = \begin{pmatrix} \boldsymbol{R} & \mathbf{r} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{pmatrix}, \;\; \boldsymbol{R} \in SO(3), \;\; \mathbf{r} \in \mathbb{R}^3 \right\} \tag{47}$$

6.10 Angular velocity (239)

This is super funky and magical: The vector $\boldsymbol{\omega}^a_{ab}$ defined by satisfying

$$(\boldsymbol{\omega}^a_{ab})^\times = \dot{\boldsymbol{R}}^a_b (\boldsymbol{R}^a_b)^{\mathrm{T}} \tag{48}$$

turns out to represent the angular velocity of b relative to a.

Then, the kinematic differential equations for the rotation matrix is given in two forms

$$\begin{aligned} \dot{\boldsymbol{R}}^a_b &= (\boldsymbol{\omega}^a_{ab})^\times \boldsymbol{R}^a_b \\ \dot{\boldsymbol{R}}^a_b &= \boldsymbol{R}^a_b (\boldsymbol{\omega}^b_{ab})^\times \end{aligned} \tag{49}$$

Example (See (37) for the definition of $\boldsymbol{R}_x$.)

$$\dot{\boldsymbol{R}}_x(\phi)\boldsymbol{R}_x^{\mathrm{T}}(\phi) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & -\mathrm{s}\,\phi & -\mathrm{c}\,\phi \\ 0 & \mathrm{c}\,\phi & -\mathrm{s}\,\phi \end{pmatrix} \dot{\phi} \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ 0 & \mathrm{c}\,\phi & \mathrm{s}\,\phi \\ 0 & -\mathrm{s}\,\phi & \mathrm{c}\,\phi \end{pmatrix}}_{\boldsymbol{R}_x^{\mathrm{T}}(\phi)} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -\dot{\phi} \\ 0 & \dot{\phi} & 0 \end{pmatrix} \tag{50}$$

which gives $\boldsymbol{\omega}_x = \begin{pmatrix} \dot{\phi} & 0 & 0 \end{pmatrix}^{\mathrm{T}}$. Similarly, $\boldsymbol{\omega}_y = \begin{pmatrix} 0 & \dot{\theta} & 0 \end{pmatrix}^{\mathrm{T}}$ and $\boldsymbol{\omega}_z = \begin{pmatrix} 0 & 0 & \dot{\psi} \end{pmatrix}^{\mathrm{T}}$

6.10.1 Simple rotation

Angular velocity $\vec{\omega}_{ab}$ about the axis of rotation $\vec{k}$ is:

$$\vec{\omega}_{ab} = \dot{\theta}\vec{k} \tag{51}$$

6.10.2 Composite rotation

Angular velocity of a composite rotation matrix $\boldsymbol{R}^a_d = \boldsymbol{R}^a_b \boldsymbol{R}^b_c \boldsymbol{R}^c_d$ is

$$\vec{\omega}_{ad} = \vec{\omega}_{ab} + \vec{\omega}_{bc} + \vec{\omega}_{cd} \tag{52}$$

which on coordinate form is

$$\begin{aligned}\boldsymbol{\omega}_{ad}^{a} &= \boldsymbol{\omega}_{ab}^{a} + \boldsymbol{\omega}_{bc}^{a} + \boldsymbol{\omega}_{cd}^{a} \\ &= \boldsymbol{\omega}_{ab}^{a} + \boldsymbol{R}_{b}^{a}\boldsymbol{\omega}_{bc}^{b} + \boldsymbol{R}_{b}^{a}\boldsymbol{R}_{c}^{b}\boldsymbol{\omega}_{cd}^{c}\end{aligned} \tag{53}$$

6.10.3 Differentiation of vectors

$$\dot{\mathbf{u}}^{a} = \boldsymbol{R}_{b}^{a}\left[\dot{\mathbf{u}}^{b} + (\boldsymbol{\omega}_{ab}^{b})^{\times}\mathbf{u}^{b}\right] \tag{54}$$

$$\frac{{}^{a}\mathrm{d}}{\mathrm{d}t}\vec{u} = \frac{{}^{b}\mathrm{d}}{\mathrm{d}t}\vec{u} + \vec{\omega}_{ab} \times \vec{u} \tag{55}$$

7 Kinematic differential equations (244)

Rotational deviation cannot be described by subtraction. Instead, it is described by

$$\widetilde{\boldsymbol{R}}_{a} := \boldsymbol{R}\boldsymbol{R}_{d}^{\mathrm{T}} \implies \boldsymbol{R} = \widetilde{\boldsymbol{R}}_{a}\boldsymbol{R}_{d}, \quad \widetilde{\boldsymbol{R}}_{a} \in SO(3) \tag{56}$$

where $\boldsymbol{R} = \boldsymbol{R}_{b}^{a}$ is the orientation and $\boldsymbol{R}_{d}$ is the desired orientation.

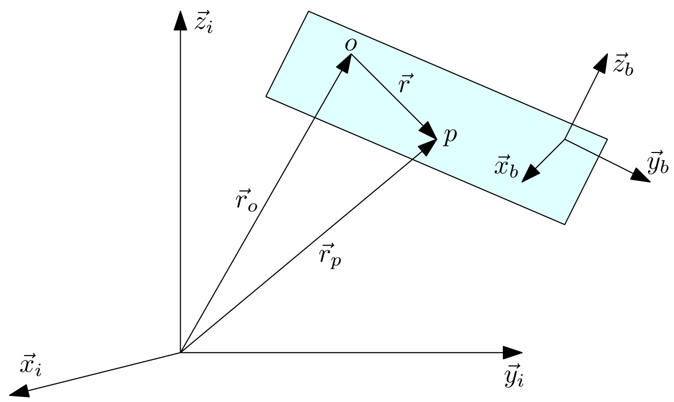
Then the kinematic differential equations for the deviation are

$$\widetilde{\boldsymbol{\omega}}^{a} = \boldsymbol{\omega}^{a} - \boldsymbol{\omega}_{d}^{a} \tag{57}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\widetilde{\boldsymbol{R}}_{a} = \left(\widetilde{\boldsymbol{\omega}}^{a}\right)^{\times}\widetilde{\boldsymbol{R}}_{a} \tag{58}$$

7.1 Rigid body kinematics (259)

7.1.1 Configuration



The orientation $\boldsymbol{R}_b^i$ of a rigid body and the position $\vec{r}_o$ of a point o in the body, both in relation to a reference frame i, define its configuration. The position of any point p in the body is given by

$$\vec{r}_p = \vec{r}_o + \vec{r}. \tag{59}$$

In the reference frame, $\vec{r}$ is

$$\mathbf{r}^i = \boldsymbol{R}_b^i \mathbf{r}^b \tag{60}$$

7.1.2 Velocity

Definition:

$$\vec{v}_o := \frac{{}^i\mathrm{d}}{\mathrm{d}t}\vec{r}_o, \quad \vec{v}_p := \frac{{}^i\mathrm{d}}{\mathrm{d}t}\vec{r}_p \tag{61}$$

Alternatively:

$$\vec{v}_p = \vec{v}_o + \frac{{}^b\mathrm{d}}{\mathrm{d}t}\vec{r} + \vec{\omega}_{ib} \times \vec{r} \tag{62}$$

7.1.3 Acceleration

Definition (translation):

$$\vec{a}_o := \frac{{}^i\mathrm{d}^2}{\mathrm{d}t^2}\vec{r}_o, \quad \vec{a}_p := \frac{{}^i\mathrm{d}^2}{\mathrm{d}t^2}\vec{r}_p \tag{63}$$

Definition (rotation):

$$\vec{\alpha}_{ib} := \frac{{}^i\mathrm{d}^2}{\mathrm{d}t^2}\vec{\omega}_{ib} = \frac{{}^b\mathrm{d}^2}{\mathrm{d}t^2}\vec{\omega}_{ib} \tag{64}$$

In terms of accelerion, ang. acceleration and velocities:

$$\begin{aligned}\underbrace{\vec{a}_p}_{\text{Acceleration of } p} &= \underbrace{\vec{a}_o}_{\text{Acceleration of } o} + \underbrace{\frac{{}^b\mathrm{d}^2}{\mathrm{d}t^2}\vec{r}}_{\text{Second derivative of } \vec{r} \text{ in } b} \\ &+ \underbrace{2\vec{\omega}_{ib} \times \frac{{}^b\mathrm{d}}{\mathrm{d}t}\vec{r}}_{\text{Coriolis acceleration}} + \underbrace{\vec{\alpha}_{ib} \times \vec{r}}_{\text{Transversal acceleration}} + \underbrace{\vec{\omega}_{ib} \times (\vec{\omega}_{ib} \times \vec{r})}_{\text{Centripetal acceleration}}\end{aligned} \tag{65}$$

Alternatively:

$$\vec{a}_p = \frac{{}^b\mathrm{d}}{\mathrm{d}t}\vec{v}_o + \vec{\omega}_{ib} \times \vec{v}_o + \frac{{}^b\mathrm{d}^2}{\mathrm{d}t^2}\vec{r} + 2\vec{\omega}_{ib} \times \frac{{}^b\mathrm{d}}{\mathrm{d}t}\vec{r} + \vec{\alpha}_{ib} \times \vec{r} + \vec{\omega}_{ib} \times (\vec{\omega}_{ib} \times \vec{r}) \tag{66}$$

As well as:

$$\vec{v}_p = \vec{v}_o + \vec{\omega}_{ib} \times \vec{r}, \quad \vec{r} \text{ fixed in } b \tag{67}$$

7.2 EoM for rigid body (269)

$$\begin{aligned}\vec{F}_{bc} &= m\vec{a}_c \\ \vec{T}_{bc} &= \vec{M}_{b/c} \cdot \vec{\alpha}_{ib} + \vec{\omega}_{ib} \times (\vec{M}_{b/c} \cdot \vec{\omega}_{ib})\end{aligned} \tag{68}$$

where

- $\vec{F}_{bc}$ is the force on body b acting through the centre of mass.
- $\vec{T}_{bc}$ is the torque, or moment about the centre of mass.
- $\vec{M}_{b/c}$ is the inertia dyadic of b about c.

8 Lagrangian dynamics (313)

8.1 Lagrange versus Newton-Euler (313)

Newton-Euler	**Lagrange**
Vectors	Algebra
Forces and moments	Energy and work
Must consider all forces	Forces of constraint eliminated
Somewhat complicated	Easier to do by hand
Suitable for computers	Less suitable for computers

Newton-Euler is based on Newton's (second) law and it's exention to rotational dynamics. Lagrangian equations of motion are instead based on algebraic operations on energy expressions, and are better suited for things related to energy conservation and passivity.

Lagrangean methods make sense when there are guiding forces involved.

8.2 Lagrange EoM (315)

The Lagrangian is

$$\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}, t) = T(\mathbf{q}, \dot{\mathbf{q}}, t) - U(\mathbf{q}) \tag{69}$$

and the EoMs are

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}_i}\right) - \frac{\partial \mathcal{L}}{\partial q_i} = \tau_i \tag{70}$$

where τ_i is the generalised actuator force.

9 Reynolds' transport theorem (413)

Time-variant volume:

$$\frac{\mathrm{d}}{\mathrm{d}t}\iiint_{V_c(t)} \phi(\mathbf{x},t)\,\mathrm{d}V = \iiint_{V_c(t)} \frac{\partial \phi(\mathbf{x},t)}{\partial t}\,\mathrm{d}V + \iint_{\partial V_c(t)} \phi \vec{v}_c \cdot \vec{n}\,\mathrm{d}A \tag{71}$$

Material volume:

$$\frac{\mathrm{d}}{\mathrm{d}t}\iiint_{V_m(t)} \phi(\mathbf{x},t)\,\mathrm{d}V = \iiint_{V_m(t)} \frac{\partial \phi(\mathbf{x},t)}{\partial t}\,\mathrm{d}V + \iint_{\partial V_m(t)} \phi \vec{v} \cdot \vec{n}\,\mathrm{d}A \tag{72}$$

Material derivative:

$$\frac{\mathrm{D}}{\mathrm{D}t}\iiint_{V_c(t)} \phi(\mathbf{x},t)\,\mathrm{d}V := \iiint_{V_c(t)} \frac{\partial \phi(\mathbf{x},t)}{\partial t}\,\mathrm{d}V + \iint_{\partial V_c(t)} \phi \vec{v} \cdot \vec{n}\,\mathrm{d}A \tag{73}$$

Differential formulation, material form:

$$\frac{\mathrm{D}}{\mathrm{D}t}\iiint_{V_c(t)} \phi(\mathbf{x},t)\,\mathrm{d}V = \iiint_{V_c(t)} \frac{\mathrm{D}\phi(\mathbf{x},t)}{\mathrm{D}t} + \phi(\mathbf{x},t)\left(\vec{\nabla} \cdot \vec{v}\right)\mathrm{d}V \tag{74}$$

Differential formulation, divergence form:

$$\frac{\mathrm{D}}{\mathrm{D}t}\iiint_{V_c(t)} \phi(\mathbf{x},t)\,\mathrm{d}V = \iiint_{V_c(t)} \frac{\partial \phi(\mathbf{x},t)}{\partial t} + \vec{\nabla} \cdot \left(\phi(\mathbf{x},t)\vec{v}\right)\mathrm{d}V \tag{75}$$

Integral formulation:

$$\frac{\mathrm{d}}{\mathrm{d}t}\iiint_{V_f} \phi(\mathbf{x},t)\,\mathrm{d}V = \frac{\mathrm{D}}{\mathrm{D}t}\iiint_{V_f} \phi(\mathbf{x},t)\,\mathrm{d}V - \iint_{\partial V_f} \phi \vec{v} \cdot \vec{n}\,\mathrm{d}A \tag{76}$$

10 Mass balance (417)

Integral formulation:

$$\frac{\mathrm{d}}{\mathrm{d}t}\iiint_{V_f} \rho(\mathbf{x},t)\,\mathrm{d}V = -\iint_{\partial V_f} \rho \vec{v} \cdot \vec{n}\,\mathrm{d}A \tag{77}$$

Differential formulation:

$$\frac{\partial \rho(\mathbf{x},t)}{\partial t} + \vec{\nabla} \cdot (\rho(\mathbf{x},t)\vec{v}) = 0 \tag{78}$$

11 Some mechanics (141)

Valve equation (141)

$$q = C_d A \sqrt{\frac{2}{\rho} \Delta p} \tag{79}$$

Bulk modulus (151)

$$\frac{\mathrm{d}\rho}{\rho} = \frac{\mathrm{d}p}{\beta} \tag{80}$$

Spring

$$F = -kx \tag{81}$$

$$E_{pot} = \frac{1}{2} k x^2 \tag{82}$$