Geometric Integration for Simulating Coordinate-Free State-Space Rigid Body Dynamics

This repo demonstrates a "geometric integrator", i.e., a numerical integrator that preserves certain geometric properties of the space in which integration is performed. This repo uses the Modern Robotics code library and much of the notation comes from *Modern Robotics* by Kevin Lynch and Frank Park. I intend this repo to serve as an extension of certain topics in that text, mainly section 8.2.2 (twist-wrench formulation of the dynamics of a single rigid body).

1 Equations of Motion

In this repo, I am using the geometric integrator proposed by Crouch and Grossman (J. Nonlinear Sci., 1993) to simulate the motion of a rigid body in 3D space. The body has 6 degrees of freedom (DOF), but in order to avoid the pitfalls associated with choosing a set of 6 generalized coordinates, I am using a coordinate-free approach by working directly with the 4x4 homogeneous transformation matrix that represents the body's configuration in space.

Denote this transformation matrix $T_{\mathrm{wb}} \in SE(3)$, where $\{\mathrm{w}\}$ is the world frame and $\{\mathrm{b}\}$ is the body frame, i.e., a frame oriented along the body's principal axes of inertia and whose origin is located at the body's center of mass (COM). The special Euclidean group $SE(3)$ is a *Lie group*—a kind of differentiable manifold that has special geometry that is not preserved by generic numerical integration algorithms like 4th-order Runge-Kutta, which operate on the n-dimensional Euclidean space $\mathbb{R}^n$. Thus, a different approach is required to faithfully simulate coordinate-free rigid body dynamics. To do this, Crouch and Grossman propose a modified 4th-order Runge-Kutta scheme, which I'll call CG4.

Represented as a system of first-order ordinary differential equations (ODEs), the equations of motion for a rigid body in space are

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} T_{\mathrm{wb}}(t) \\ \mathcal{V}_{\mathrm{b}}(t) \end{bmatrix} = \begin{bmatrix} T_{\mathrm{wb}}(t)\left[\mathcal{V}_{\mathrm{b}}(t)\right] \\ \mathcal{K}\left(t, T_{\mathrm{wb}}(t), \mathcal{V}_{\mathrm{b}}(t)\right) \end{bmatrix}, \tag{1}$$

where

$$\mathcal{K}\left(t, T_{\mathrm{wb}}(t), \mathcal{V}_{\mathrm{b}}(t)\right) = \mathcal{G}_{\mathrm{b}}^{-1}\left(\left[\mathrm{ad}_{\mathcal{V}_{\mathrm{b}}(t)}\right]^{\top}\mathcal{G}_{\mathrm{b}}\mathcal{V}_{\mathrm{b}}(t) + \mathcal{F}_{\mathrm{b}}(t)\right). \tag{2}$$

In these equations,

- $T_{\mathrm{wb}} \in SE(3)$ is the configuration of the body's COM frame $\{\mathrm{b}\}$ relative to the inertial world frame $\{\mathrm{w}\}$.
- $[\mathcal{V}_{\mathrm{b}}] \in \mathfrak{se}(3)$ is the matrix representation of the body twist $\mathcal{V}_{\mathrm{b}} \in \mathbb{R}^6$, where $\mathcal{V}_{\mathrm{b}} = \left[\omega_{\mathrm{b}}^{\top}, v_{\mathrm{b}}^{\top}\right]^{\top}$, where $\omega_{\mathrm{b}} \in \mathbb{R}^3$ and $v_{\mathrm{b}} \in \mathbb{R}^3$ are the angular and linear velocity, respectively, represented in the COM frame $\{\mathrm{b}\}$.
- $\mathcal{G}_{\mathrm{b}} \in \mathbb{R}^{6\times 6}$ is the body's inertia tensor; $\mathcal{G}_{\mathrm{b}}$ is a diagonal matrix with $(\mathcal{I}_{xx}, \mathcal{I}_{yy}, \mathcal{I}_{zz}, \mathfrak{m}, \mathfrak{m}, \mathfrak{m})$ along the diagonal, where $\mathcal{I}_{xx}$, $\mathcal{I}_{yy}$, and $\mathcal{I}_{zz}$ are the inertias about the principal axes and $\mathfrak{m}$ is the body's mass.
- $\mathcal{F}_{\mathrm{b}} \in \mathbb{R}^6$ is the net external wrench on the body, expressed in the COM frame. $\mathcal{F}_{\mathrm{b}} = \left[m_{\mathrm{b}}^{\top}, f_{\mathrm{b}}^{\top}\right]^{\top}$, where $m_{\mathrm{b}} \in \mathbb{R}^3$ and $f_{\mathrm{b}} \in \mathbb{R}^3$ are the moment and force, respectively, in the COM frame.
- $\left[\mathrm{ad}_{\mathcal{V}_{\mathrm{b}}(t)}\right]$ is a 6x6 matrix used to express the rate of change of spatial momentum $\mathcal{P}_{\mathrm{b}} = \mathcal{G}_{\mathrm{b}}\mathcal{V}_{\mathrm{b}}$ due to the twist $\mathcal{V}_{\mathrm{b}}$.

In order to numerically integrate the equations of motion, it often helps to express them as a system of first-order ODEs, as we have done above. However, we must be careful about the topology of the state space we create when we do this. The body configuration T_{wb} is an element of $SE(3)$ and the body twist $\mathcal{V}_{\mathrm{b}}$ is an element of $\mathbb{R}^6$, so the pair $(T_{\mathrm{wb}}, \mathcal{V}_{\mathrm{b}})$ is an element of the space $SE(3) \times \mathbb{R}^6$. Because $SE(3)$ is not globally like $\mathbb{R}^6$, it makes sense to treat them separately when numerically integrating the equations of motion. This is exactly what the Crouch-Grossman geometric integrator does.

2 Crouch-Grossman Algorithm for Integration on SE(3)

This description of the CG4 algorithm comes from Park et al. (IROS 2004), a paper that does a nice job explaining the Crouch-Grossman and Munthe-Kass integrators. The algorithm is as follows:

Given time t_k, timestep h, configuration $T_{\mathrm{wb}}(t_k)$ and twist $\mathcal{V}_{\mathrm{b}}(t_k)$,

$$T_{\mathrm{wb}}\left(t_{k+1}\right) = T_{\mathrm{wb}}\left(t_k\right) \prod_{i=1}^{s} \exp\left(hb_i \left[\mathcal{V}_{\mathrm{b}}^{(i)}\right]\right), \tag{3a}$$

$$\mathcal{V}_{\mathrm{b}}\left(t_{k+1}\right) = \mathcal{V}_{\mathrm{b}}(t_k) + \sum_{i=1}^{s} hb_i \mathcal{K}^{(i)}, \tag{3b}$$

where, for $i = 1, 2, \cdots, s$,

$$\mathcal{K}^{(i)} = \mathcal{K}\left(t_k + hc_i, T_{\mathrm{wb}}^{(i)}, \mathcal{V}_{\mathrm{b}}^{(i)}\right), \text{ where} \tag{4a}$$

$$T_{\mathrm{wb}}^{(i)} = T_{\mathrm{wb}}\left(t_k\right) \prod_{j=1}^{i-1} \exp\left(ha_{ij} \left[\mathcal{V}_{\mathrm{b}}^{(j)}\right]\right) \tag{4b}$$

$$\mathcal{V}_{\mathrm{b}}^{(i)} = \mathcal{V}_{\mathrm{b}}\left(t_k\right) + \sum_{j=1}^{i-1} ha_{ij} \mathcal{K}^{(j)}. \tag{4c}$$

Thus, the Crouch-Grossman integrator uses the product of matrix exponentials to integrate the configuration forward in time and uses simple addition to integrate the twist forward in time, thereby respecting the distinct geometry of $SE(3)$ and $\mathbb{R}^6$. It's worth noting that the product of exponentials is performed left-to-right as i (or j) increases; for example, you'd have

$$T_{\mathrm{wb}}\left(t_k\right) \exp\left(hb_1 \left[\mathcal{V}_{\mathrm{b}}^{(1)}\right]\right) \exp\left(hb_2 \left[\mathcal{V}_{\mathrm{b}}^{(2)}\right]\right) \cdots \exp\left(hb_s \left[\mathcal{V}_{\mathrm{b}}^{(s)}\right]\right).$$

In this algorithm, s represents the number of stages, and the coefficients $a \in \mathbb{R}^{s \times s}$, $b \in \mathbb{R}^s$, $c \in \mathbb{R}^s$ are weights. Park et al. give the coefficients for a 5-stage ($s = 5$) 4th-order integrator. These coefficients are listed in `cg4.m`.

3 Transformation Matrices

The underlying problem, and the reason CG4 exists, is that the sum of two transformation matrices is not itself a transformation matrix. For example, one might be (naively) tempted to apply Euler integration to update T:

$$T(t_{k+1}) = T(t_k) + \dot{T}(t_k)\Delta t, \quad \text{(INCORRECT!)}$$

but **DON'T DO THIS!** The new transformation matrix, $T(t_{k+1})$, will not be an element of $SE(3)$. The correct way to update T is via the *matrix exponential*:

$$T(t_{k+1}) = T(t_k) \exp\left([\mathcal{V}]\right), \quad \text{(CORRECT!)} \tag{5}$$

where $[\mathcal{V}] \in \mathfrak{se}(3)$ (the *Lie algebra* of the Lie group $SE(3)$) is the *twist*, related to $\dot{T}$ by

$$[\mathcal{V}] = T^{-1}\dot{T}, \tag{6}$$

where $[\mathcal{V}]$ denotes the representation of $\mathcal{V} \in \mathbb{R}^6$ as an element of $\mathfrak{se}(3)$:

$$[\mathcal{V}] = \begin{bmatrix} [\omega] & v \\ 0 & 0 \end{bmatrix}, \tag{7}$$

where $\omega \in \mathbb{R}^3$, $v \in \mathbb{R}^3$, and where $[\omega]$ is the skew-symmetric matrix representation of ω:

$$[\omega] = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}. \tag{8}$$