Spatial Vector Algebra

The Easy Way to do Rigid Body Dynamics

Roy Featherstone
Dept. Information Engineering, RSISE
The Australian National University
A concise vector notation for describing rigid–body velocity, acceleration, inertia, etc., using 6D vectors and tensors.

- fewer quantities
- fewer equations
- less effort
- fewer mistakes
Velocity

Spatial velocity:

\[ \hat{\mathbf{v}} = \begin{bmatrix} \omega \\ \mathbf{v}_O \end{bmatrix} \]

\( \mathbf{v}_O = \mathbf{v}_P + \mathbf{OP} \times \omega \)
Acceleration

. . . is the rate of change of velocity:

\[ \dot{\mathbf{a}} = \frac{d}{dt} \dot{\mathbf{v}} = \begin{bmatrix} \dot{\omega} \\ \dot{\mathbf{v}_o} \end{bmatrix} \]

but this is \textit{not} the linear acceleration of any point in the body!
Force

force $\mathbf{f}$ through $P$: $\hat{\mathbf{f}} = \begin{bmatrix} \overrightarrow{OP} \times \mathbf{f} \\ \mathbf{f} \end{bmatrix}$

pure couple $\tau$: $\hat{\mathbf{f}} = \begin{bmatrix} \tau \\ 0 \end{bmatrix}$
Rigid Body Inertia

**mass:** $m$

**CoM:** $P$

**inertia at CoM:** $I^*$

**spatial inertia tensor:**

$$\hat{I} = \begin{bmatrix} I & H \\ H^T & M \end{bmatrix}$$

where

$M = m \mathbf{1}$

$H = m \overrightarrow{OP} \times$

$I = I^* - m \overrightarrow{OP} \times \overrightarrow{OP} \times$
Operations on Spatial Quantities

- Composition of velocities

If $\hat{v}_A = \text{velocity of body } A$

$\hat{v}_B = \text{velocity of body } B$

$\hat{v}_{BA} = \text{relative velocity of B w.r.t. A}$

Then $\hat{v}_B = \hat{v}_A + \hat{v}_{BA}$
Composition of accelerations

If $\hat{a}_A$ = acceleration of body A
$\hat{a}_B$ = acceleration of body B
$\hat{a}_{BA}$ = acceleration of B w.r.t. A

Then $\hat{a}_B = \hat{a}_A + \hat{a}_{BA}$

Look, no Coriolis term!
• Composition of forces

If forces \( \hat{f}_1 \) and \( \hat{f}_2 \) both act on the same body then their resultant is

\[
\hat{f}_{tot} = \hat{f}_1 + \hat{f}_2
\]

• Composition of inertias

If two bodies with inertias \( \hat{I}_A \) and \( \hat{I}_B \) are connected together then the inertia of the composite body is

\[
\hat{I}_{tot} = \hat{I}_A + \hat{I}_B
\]
Mathematical Structure

spatial vectors inhabit two vector spaces:

- \( \mathbb{M}^6 \) — motion vectors
- \( \mathbb{F}^6 \) — force vectors

with a scalar product defined between them

\[
\mathbf{m} \cdot \mathbf{f} = \text{work}
\]

\[\text{“•”} : \mathbb{M}^6 \times \mathbb{F}^6 \rightarrow \mathbb{R}\]
Bases

If \( \{d_1, \ldots, d_6\} \) is an arbitrary basis on \( \mathbb{M}^6 \) then there exists a unique basis \( \{e_1, \ldots, e_6\} \) on \( \mathbb{F}^6 \) satisfying

\[
d_i \cdot e_j = \begin{cases} 
0 & : i \neq j \\
1 & : i = j 
\end{cases}
\]

In this basis, the scalar product of two coordinate vectors is

\[
m \cdot f = [m]^T [f]
\]
Plücker Coordinates

A Cartesian coordinate frame $O_{xyz}$ defines twelve basis vectors:

$d_{Ox}, d_{Oy}, d_{Oz}, d_x, d_y, d_z$:
- rotations about the $Ox$, $Oy$ and $Oz$ axes,
- translations in the $x$, $y$ and $z$ directions

$e_x, e_y, e_z, e_{Ox}, e_{Oy}, e_{Oz}$:
- couples in the $yz$, $zx$ and $xy$ planes, and
- forces along the $Ox$, $Oy$ and $Oz$ axes
Equations like \( \hat{\mathbf{v}} = \begin{bmatrix} \omega \\ \mathbf{v}_O \end{bmatrix} \) and \( \hat{\mathbf{f}} = \begin{bmatrix} \tau_O \\ \mathbf{f} \end{bmatrix} \)

really mean

\[
\hat{\mathbf{v}} = \omega_x \mathbf{d}_{Ox} + \omega_y \mathbf{d}_{Oy} + \omega_z \mathbf{d}_{Oz} + \\
+ \mathbf{v}_{Ox} \mathbf{d}_x + \mathbf{v}_{Oy} \mathbf{d}_y + \mathbf{v}_{Oz} \mathbf{d}_z
\]

\[
\hat{\mathbf{f}} = \tau_{Ox} \mathbf{e}_x + \tau_{Oy} \mathbf{e}_y + \tau_{Oz} \mathbf{e}_z + \\
+ f_x \mathbf{e}_{Ox} + f_y \mathbf{e}_{Oy} + f_z \mathbf{e}_{Oz}
\]
Equation of Motion

\[ f = \frac{d}{dt}(Iv) = Ia + v \times Iv \]

- \( f \) = net force acting on a rigid body
- \( I \) = inertia of rigid body
- \( v \) = velocity of rigid body
- \( Iv \) = momentum of rigid body
- \( a \) = acceleration of rigid body
Example 1: Robot Kinematics

\[ \mathbf{v}_i = \mathbf{v}_{i-1} + \mathbf{s}_i \dot{q}_i \quad (\mathbf{v}_0 = 0) \]
\[ \mathbf{a}_i = \mathbf{a}_{i-1} + \mathbf{s}_i \ddot{q}_i + \mathbf{s}_i \dot{q}_i \quad (\mathbf{a}_0 = 0) \]

\[ \mathbf{v}_i, \mathbf{a}_i \] link velocity and acceleration

\[ \dot{q}_i, \ddot{q}_i, s_i \] joint velocity, acceleration & axis
Example 2: Inverse Dynamics

(Calculate the joint torques $Q_i$ that will produce the desired joint accelerations $\ddot{q}_i$.)

\[ v_i = v_{i-1} + s_i \dot{q}_i \quad (v_0 = 0) \]
\[ a_i = a_{i-1} + s_i \dot{q}_i + s_i \ddot{q}_i \quad (a_0 = 0) \]
\[ f_i = f_{i+1} + I_i a_i + v_i \times I_i v_i \quad (f_{n+1} = f_{ee}) \]
\[ Q_i = s_i^T f_i \]

(The Recursive Newton–Euler Algorithm)