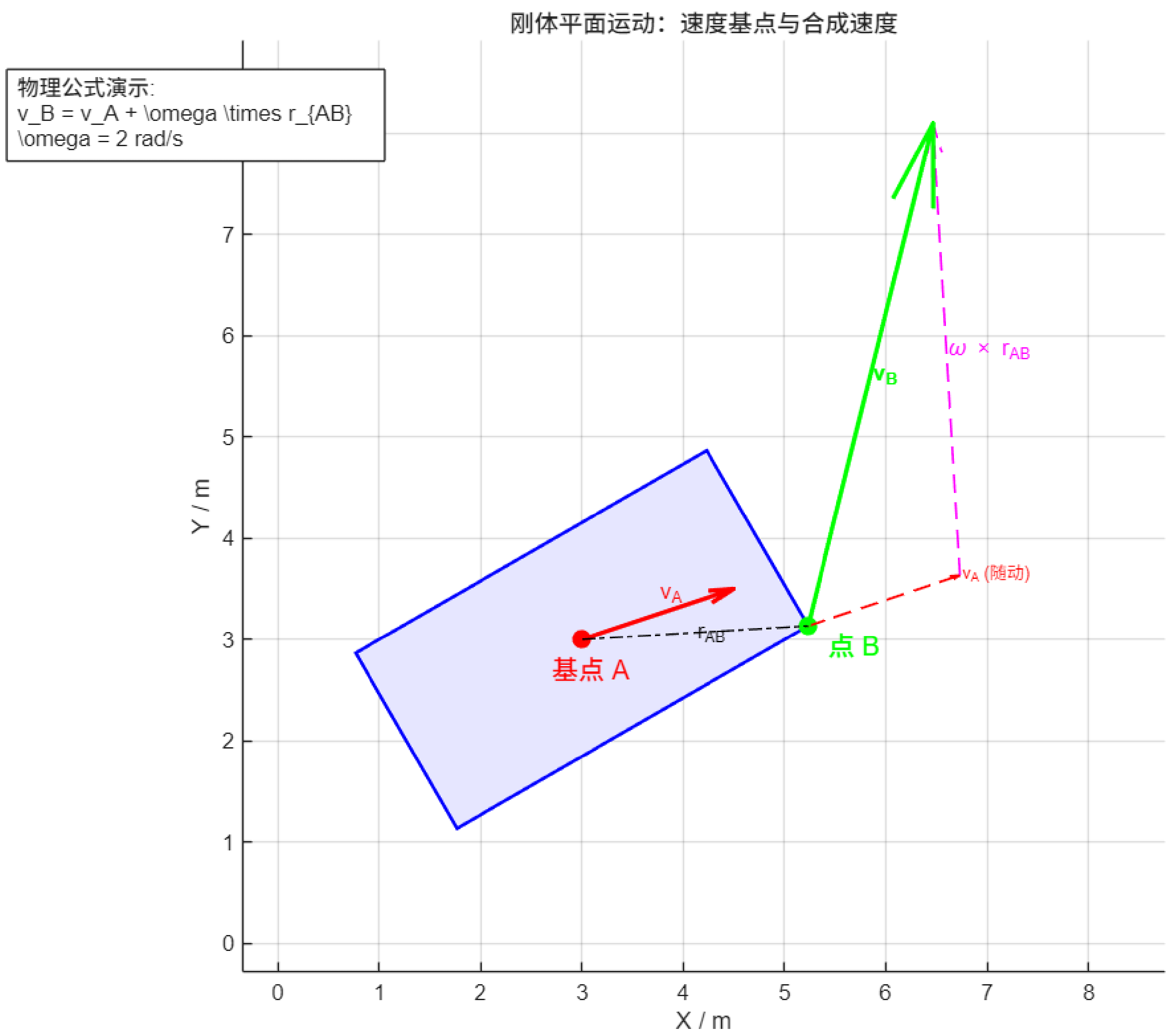
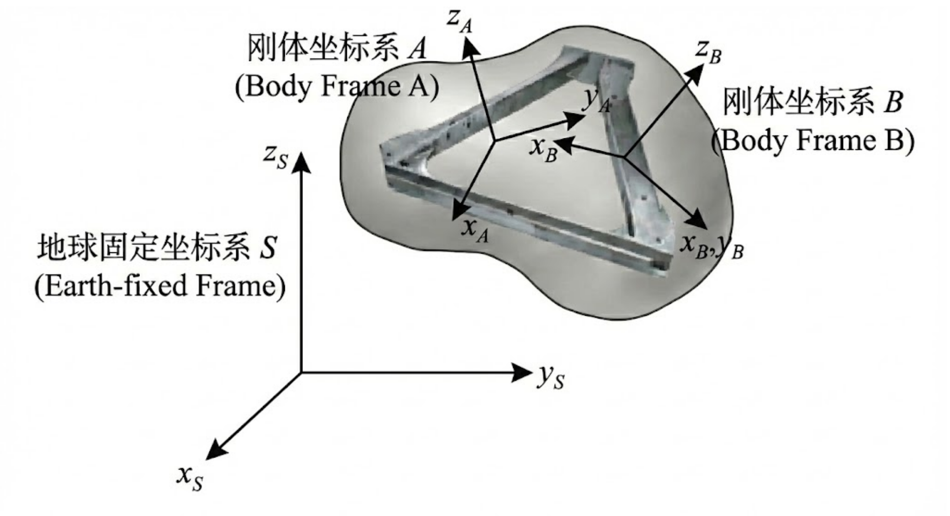
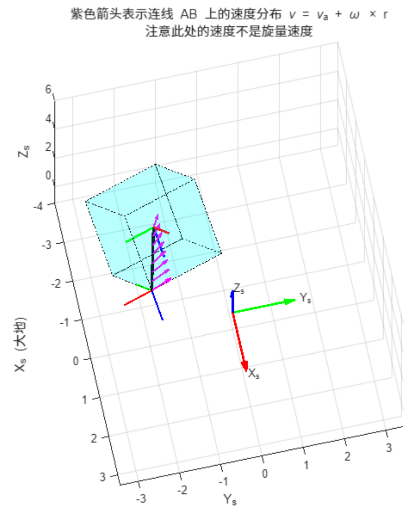
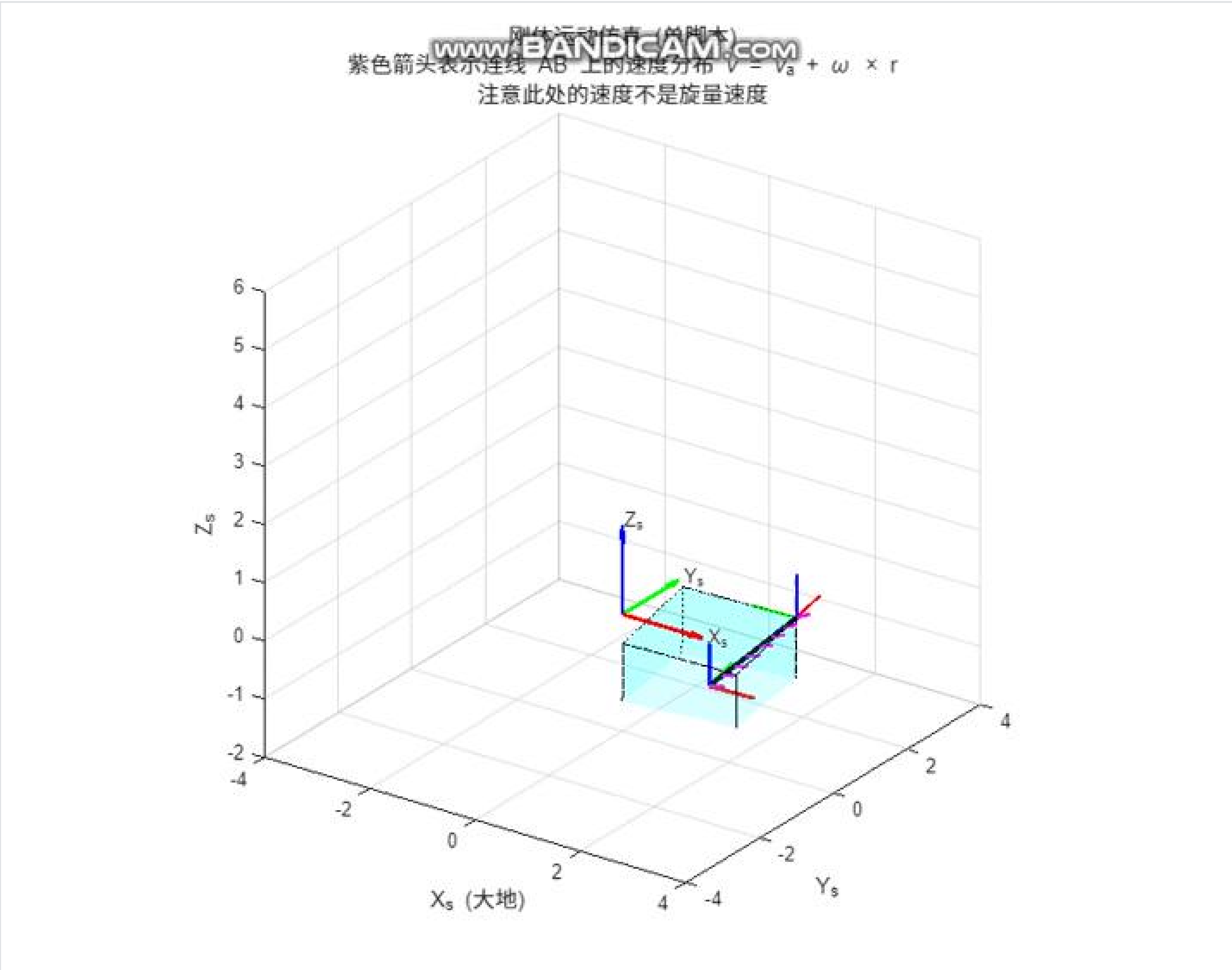
旋量的伴随变换2

> **速度基点 (Velocity Reference Point)**：
>
> 简单来说，它的核心逻辑是：**“当刚体在空间中进行自由运动时，刚体上所有点的角速度 ω 是相同的，但线速度 v 却因点而异。”**
>
> 对于一个在空间中做一般运动的刚体，为了用一组向量（6个数）唯一地描述它的运动状态，我们必须**人为指定一个点**，以该点的线速度来代表整个刚体的平动部分。这个被选定的点，就是**速度基点**。
>
> - **角速度 ω**：是全刚体共享的，与基点选择无关。
> - **线速度 v**：完全取决于你选哪个点作为基点。
>
> 
>







1. 根据微分运动学推导伴随变换矩阵

$^{bb}\mathcal{V}_{sb}$ 表示，以刚体上的b点作为速度基点，且算的速度是b相对于坐标{s}的速度，最终这个(赝)矢量在{b}上的表示。两个上标b，一个表示在{b}系下表示，另外一个表示是刚体上b点的速度。下标表示是b相对于s求导，并且速度基点选在b点（一个b对应了两层含义）。

${}^{aa}\mathcal{V}_{sb}$ 表示，表示刚体上与 $\{a\}$ 原点重合的那一点相对于 $\{s\}$ 的速度（这个速度有很多种表达方式，由下标b决定，它的速度基点选在b点），数值在 $\{a\}$ 系下表示。

这个分别就是书上定义的 $\mathcal{V}_b = \mathrm{Ad}_{T_{ba}} \mathcal{V}_a$

根据旋量的**定义**（定义是人驯服具有群结构的流形空间的精妙结论）：

♥

$$ {}^{bb}v_{sb} = {}^{b}R_s {}^{s}\dot{p}_{sb} \tag{1} $$

$$ {}^{aa}v_{sb} = {}^{a}R_s({}^{s}\dot{p}_{sb} + {}^{s}\omega_{sb} \times (-{}^{s}p_{ab})) = {}^{a}R_s {}^{s}\dot{p}_{sa} = {}^{aa}v_{sa} \tag{2} $$

1. 角速度变换（上半部分）

角速度作为自由向量，其变换仅取决于旋转矩阵：

${}^{b}\omega_{sb} = {}^{b}R_a {}^{a}\omega_{sa} \implies {}^{a}\omega_{sa} = {}^{a}R_b {}^{b}\omega_{sb}$

线速度变换（下半部分）

我们将公式 (1) 代入公式 (2)：

${}^{aa}v_{sb} = {}^{a}R_s(\underbrace{{}^{s}R_b {}^{bb}v_{sb}}_{{}^{s}\dot{p}_{sb}} + {}^{s}\omega_{sb} \times (-{}^{s}p_{ab}))$

利用旋转矩阵的分配律展开：

${}^{aa}v_{sb} = ({}^{a}R_s {}^{s}R_b)^{bb}v_{sb} + {}^{a}R_s({}^{s}\omega_{sb} \times (-{}^{s}p_{ab}))$

由于 ${}^{a}R_s {}^{s}R_b = {}^{a}R_b$，第一项变为 ${}^{a}R_b {}^{bb}v_{sb}$。

对于第二项，利用恒等式 $R(u \times v) = (Ru) \times (Rv)$：

${}^{a}R_s({}^{s}\omega_{sb} \times (-{}^{s}p_{ab})) = ({}^{a}R_s {}^{s}\omega_{sb}) \times ({}^{a}R_s(-{}^{s}p_{ab}))$

已知：

- ${}^{a}R_s {}^{s}\omega_{sb} = {}^{a}\omega_{sb}$ （即刚体角速度在 a 系下的表达）
- ${}^{a}R_s(-{}^{s}p_{ab}) = -{}^{a}p_{ab} = {}^{a}p_{ba}$ （即 a 指向 b 的矢量在 a 系下的表达）

代回原式：

${}^{aa}v_{sb} = {}^{a}R_b {}^{bb}v_{sb} + {}^{a}\omega_{sb} \times {}^{a}p_{ba}$

为了整理成矩阵形式，我们将叉乘写成反对称矩阵 $[{}^{a}p_{ba}]$ 并交换顺序（注意 $u \times v = -v \times u$）：

${}^{aa}v_{sb} = {}^{a}R_b {}^{bb}v_{sb} - [{}^{a}p_{ba}]{}^{a}\omega_{sb}$

再代入角速度关系 ${}^{a}\omega_{sb} = {}^{a}R_b {}^{b}\omega_{sb}$：

${}^{aa}v_{sb} = -[{}^{a}p_{ba}]{}^{a}R_b {}^{b}\omega_{sb} + {}^{a}R_b {}^{bb}v_{sb}$

3. 整理成伴随矩阵

将上述结果组合成块矩阵形式：

$$\begin{bmatrix} {}^a\omega_{sb} \\ {}^{aa}v_{sb} \end{bmatrix} = \begin{bmatrix} {}^aR_b & 0 \\ -[{}^ap_{ba}]^aR_b & {}^aR_b \end{bmatrix} \begin{bmatrix} {}^b\omega_{sb} \\ {}^{bb}v_{sb} \end{bmatrix}$$

只需对上述矩阵求逆。利用伴随矩阵的性质 $Ad_{T^{-1}} = (Ad_T)^{-1}$，且考虑到坐标系对称性（将 a, b 互换）：

$$\begin{bmatrix} {}^b\omega_{sb} \\ {}^{bb}v_{sb} \end{bmatrix} = \begin{bmatrix} {}^bR_a & 0 \\ [{}^bp_{ba}]^bR_a & {}^bR_a \end{bmatrix} \begin{bmatrix} {}^a\omega_{sb} \\ {}^{aa}v_{sb} \end{bmatrix}$$

> **注意：** 这里的 $[{}^bp_{ba}]$ 是 $\{a\}$ 的原点在 $\{b\}$ 系下的坐标（即 b 指向 a 的矢量在 b 系下的表达）。

更简单的推导方式：

$$\begin{bmatrix} {}^b\omega_{sb} \\ {}^{bb}v_{sb} \end{bmatrix} = \begin{bmatrix} {}^bR_a & 0 \\ [{}^bp_{ba}]^bR_a & {}^bR_a \end{bmatrix} \begin{bmatrix} {}^a\omega_{sb} \\ {}^{aa}v_{sb} \end{bmatrix}$$

按矩阵乘开：

$${}^{bb}v_{sb} = ([{}^bp_{ba}]^bR_a)^a\omega_{sb} + {}^bR_a{}^{aa}v_{sb}$$

(2)带入上式第二项：

$${}^bR_a{}^{aa}v_{sb} = {}^bR_a\,({}^aR_s({}^s\dot{p}_{sb} + {}^s\omega_{sb} \times (-{}^sp_{ab})))$$

$${}^bR_a{}^{aa}v_{sb} = {}^bR_s{}^s\dot{p}_{sb} + {}^bR_s({}^s\omega_{sb} \times (-{}^sp_{ab}))$$

(1)再带入上式：

$${}^bR_a{}^{aa}v_{sb} = {}^{bb}v_{sb} + ({}^bR_s{}^s\omega_{sb}) \times ({}^bR_s(-{}^sp_{ab}))$$

$${}^bR_a{}^{aa}v_{sb} = {}^{bb}v_{sb} + {}^b\omega_{sb} \times (-{}^bp_{ab})$$

由于 $-{}^bp_{ab} = {}^bp_{ba}$（即从 b 指向 a 的矢量），上式变为：

$${}^bR_a{}^{aa}v_{sb} = {}^{bb}v_{sb} + {}^b\omega_{sb} \times {}^bp_{ba}$$

上式带入 ${}^{bb}v_{sb} = ([{}^bp_{ba}]^bR_a)^a\omega_{sb} + {}^bR_a{}^{aa}v_{sb}$

对于等式右边有：

$$([{}^bp_{ba}]^bR_a)^a\omega_{sb} + {}^{bb}v_{sb} + {}^b\omega_{sb} \times {}^bp_{ba} = [{}^bp_{ba}]^b\omega_{sb} + {}^{bb}v_{sb} + {}^b\omega_{sb} \times {}^bp_{ba} = {}^{bb}v_{sb}$$

所以 ${}^{bb}\mathcal{V}_{sb} = \mathrm{Ad}_{T_{ba}}{}^{aa}\mathcal{V}_{sb}$

2. 从移轴变换到伴随变换

${}^{sb}\mathcal{V}_{sb} = \begin{bmatrix} I & 0 \\ -[{}^sp_{ab}] & I \end{bmatrix} {}^{sa}\mathcal{V}_{sb}$ 如何从此式推导出 ${}^{bb}\mathcal{V}_{sb} = \begin{bmatrix} {}^bR_a & 0 \\ [{}^bp_{ba}]^bR_a & {}^bR_a \end{bmatrix} {}^{aa}\mathcal{V}_{sb}$ 此式

1. 符号说明与问题设定

采用以下旋量记法：

- ${}^{xy}\mathcal{V}_{zb}$ ：物体b相对于参考系z的速度旋量，**表达在x系**，且速度矩的**参考点选在y点**
- ${}^{bb}\mathcal{V}_{sb}$ ：物体b相对于空间s的速度旋量，表达在b系，速度矩的参考点为b
- ${}^{aa}\mathcal{V}_{sb}$ ：物体b相对于空间s的速度旋量，表达在a系，参考点为a

已知在同一坐标系{s}下的移轴变换公式：

$$
{}^{sb}\mathcal{V}_{sb} = \begin{bmatrix} I & 0 \\ -[{}^{s}p_{ab}] & I \end{bmatrix} {}^{sa}\mathcal{V}_{sb}
$$

其中 ${}^{s}p_{ab}$ 是从点a指向点b的向量在{s}系中的坐标，$[v]$ 表示向量 v 的叉乘矩阵。

目标：推导出跨坐标系的伴随变换公式：

$$
{}^{bb}\mathcal{V}_{sb} = \begin{bmatrix} {}^{b}R_a & 0 \\ [{}^{b}p_{ba}]{}^{b}R_a & {}^{b}R_a \end{bmatrix} {}^{aa}\mathcal{V}_{sb}
$$

其中 ${}^{b}p_{ba}$ 是从点b指向点a的向量在{b}系中的坐标，${}^{b}R_a$ 为从{a}系到{b}系的旋转矩阵。

---

2. 建立坐标系间的旋转变换关系

同一物理旋量在不同坐标系中的表达通过旋转矩阵关联：

1. b系到s系的旋转变换：

物体b的速度旋量从{b}系表达转换到{s}系表达：

$$
{}^{sb}\mathcal{V}_{sb} = \begin{bmatrix} {}^{s}R_b & 0 \\ 0 & {}^{s}R_b \end{bmatrix} {}^{bb}\mathcal{V}_{sb} \tag{1}
$$

2. a系到s系的旋转变换：

物体b的速度旋量从{a}系表达转换到{s}系表达：

$$
{}^{sa}\mathcal{V}_{sb} = \begin{bmatrix} {}^{s}R_a & 0 \\ 0 & {}^{s}R_a \end{bmatrix} {}^{aa}\mathcal{V}_{sb} \tag{2}
$$

其中 ${}^{s}R_b$ 、${}^{s}R_a$ 分别为{b}系、{a}系到{s}系的旋转矩阵。

---

3. 代入移轴公式并重新组合

将式(1)和式(2)代入已知的移轴公式：

$$
\begin{bmatrix} {}^{s}R_b & 0 \\ 0 & {}^{s}R_b \end{bmatrix} {}^{bb}\mathcal{V}_{sb} = \begin{bmatrix} I & 0 \\ -[{}^{s}p_{ab}] & I \end{bmatrix} \begin{bmatrix} {}^{s}R_a & 0 \\ 0 & {}^{s}R_a \end{bmatrix} {}^{aa}\mathcal{V}_{sb}
$$

为解出 ${}^{bb}\mathcal{V}_{sb}$ ，左乘 $\begin{bmatrix} {}^sR_b & 0 \\ 0 & {}^sR_b \end{bmatrix}^{-1}$ ：

$${}^{bb}\mathcal{V}_{sb} = \begin{bmatrix} {}^sR_b & 0 \\ 0 & {}^sR_b \end{bmatrix}^{-1} \begin{bmatrix} I & 0 \\ -[{}^sp_{ab}] & I \end{bmatrix} \begin{bmatrix} {}^sR_a & 0 \\ 0 & {}^sR_a \end{bmatrix} {}^{aa}\mathcal{V}_{sb}$$

利用旋转矩阵的性质 $R^{-1} = R^T$ ，记 ${}^bR_s = ({}^sR_b)^T$ ：

$${}^{bb}\mathcal{V}_{sb} = \begin{bmatrix} {}^bR_s & 0 \\ 0 & {}^bR_s \end{bmatrix} \begin{bmatrix} I & 0 \\ -[{}^sp_{ab}] & I \end{bmatrix} \begin{bmatrix} {}^sR_a & 0 \\ 0 & {}^sR_a \end{bmatrix} {}^{aa}\mathcal{V}_{sb} \tag{3}$$

---

4. 矩阵乘法化简

第一步：计算后两个矩阵的乘积

$$\begin{bmatrix} I & 0 \\ -[{}^sp_{ab}] & I \end{bmatrix} \begin{bmatrix} {}^sR_a & 0 \\ 0 & {}^sR_a \end{bmatrix} = \begin{bmatrix} {}^sR_a & 0 \\ -[{}^sp_{ab}]{}^sR_a & {}^sR_a \end{bmatrix}$$

第二步：左乘第一个矩阵

$$\begin{bmatrix} {}^bR_s & 0 \\ 0 & {}^bR_s \end{bmatrix} \begin{bmatrix} {}^sR_a & 0 \\ -[{}^sp_{ab}]{}^sR_a & {}^sR_a \end{bmatrix} = \begin{bmatrix} {}^bR_s^s R_a & 0 \\ -{}^bR_s[{}^sp_{ab}]{}^sR_a & {}^bR_s^s R_a \end{bmatrix}$$

利用旋转矩阵的链式法则 ${}^bR_s^sR_a =^b R_a$ ，得到：

$${}^{bb}\mathcal{V}_{sb} = \begin{bmatrix} {}^bR_a & 0 \\ -{}^bR_s[{}^sp_{ab}]{}^sR_a & {}^bR_a \end{bmatrix} {}^{aa}\mathcal{V}_{sb} \tag{4}$$

---

5. 关键项化简

重点化简左下角项 $X = -{}^bR_s[{}^sp_{ab}]{}^sR_a$ 。

利用叉乘矩阵与旋转矩阵的交换性质：

对于任意旋转矩阵 R 和向量 v ，有恒等式：

$$R[v] = [Rv]R$$

证明：由 $[Rv] = R[v]R^T$ ，右乘 R 得 $[Rv]R = R[v]R^TR = R[v]$ 。

应用此性质，取 $R =^b R_s$ ， $v =^s p_{ab}$ ：

$${}^bR_s[{}^sp_{ab}] = [{}^bR_s^sp_{ab}]{}^bR_s = [{}^bp_{ab}]{}^bR_s$$

其中 ${}^bp_{ab} =^b R_s^sp_{ab}$ 是向量 p_{ab} 在{b}系中的坐标。

代入 X ：

$$X = -[{}^bp_{ab}]{}^bR_s^sR_a = -[{}^bp_{ab}]{}^bR_a$$

处理向量方向：

由定义：

- ${}^b p_{ab}$ ：从a指向b的向量在{b}系中坐标
- ${}^b p_{ba}$ ：从b指向a的向量在{b}系中坐标

显然 ${}^b p_{ba} = -{}^b p_{ab}$ ，故：

$X = -[{}^b p_{ab}]{}^b R_a = -[-{}^b p_{ba}]{}^b R_a = [{}^b p_{ba}]{}^b R_a$

---

6. 最终结果

将化简后的 X 代回式(4)，得到：

$$\boxed{{}^{bb}\mathcal{V}_{sb} = \begin{bmatrix} {}^b R_a & 0 \\ [{}^b p_{ba}]{}^b R_a & {}^b R_a \end{bmatrix} {}^{aa}\mathcal{V}_{sb}}$$

---

7. 物理意义解读

1. **移轴变换**：公式 ${}^{sb}\mathcal{V}_{sb} = \begin{bmatrix} I & 0 \\ -[{}^s p_{ab}] & I \end{bmatrix} {}^{sa}\mathcal{V}_{sb}$ 描述了**在同一坐标系{s}中**，将速度旋量的参考点从a移动到b时，旋量坐标的变化。
2. **伴随变换**：推导出的公式 ${}^{bb}\mathcal{V}_{sb} = \begin{bmatrix} {}^b R_a & 0 \\ [{}^b p_{ba}]{}^b R_a & {}^b R_a \end{bmatrix} {}^{aa}\mathcal{V}_{sb}$ 描述了**同时改变参考点和表达坐标系**的复合变换：
   - 左上角 ${}^b R_a$ ：处理坐标系从{a}到{b}的旋转
   - 左下角 $[{}^b p_{ba}]{}^b R_a$ ：同时处理参考点从a到b的移动，以及坐标系的旋转
3. **数学本质**：推导中的关键技巧是利用了旋转矩阵与叉乘矩阵的相似变换性质 $R[v]R^T = [Rv]$ ，这使得几何关系在不同坐标系中保持协调。

动画代码

可以修改轨迹、环绕速度观察速度矢量箭头。

也可放到Gemini3里，让它去绘制实时的空间系{s}旋量、物体系{a}旋量、物体系{b}旋量

代码块

```
%% 刚体自由运动及速度场可视化（单脚本版）
% 包含：大地坐标系(S)、刚体中心(A)、刚体角点(B)、AB连线及速度场
clc; clear; close all;

```

```matlab
%% 1. 初始化绘图环境
figure('Color', 'white', 'Name', '刚体运动与速度场', 'NumberTitle', 'off');
axis equal; grid on; hold on;
xlabel('X_s (大地)'); ylabel('Y_s'); zlabel('Z_s');
title({'刚体运动仿真 (单脚本)', '紫色箭头表示连线 AB 上的速度分布 \nu = \nu_a + \omega \times r'});
view(35, 25);
xlim([-4 4]); ylim([-4 4]); zlim([-2 6]); % 稍微调整视野适应向上运动
set(gca, 'FontSize', 10);

%% 2. 绘制大地坐标系 S (静态)
% 红色X, 绿色Y, 蓝色Z
quiver3(0,0,0, 1.5,0,0, 0, 'r', 'LineWidth', 2, 'MaxHeadSize', 0.5);
quiver3(0,0,0, 0,1.5,0, 0, 'g', 'LineWidth', 2, 'MaxHeadSize', 0.5);
quiver3(0,0,0, 0,0,1.5, 0, 'b', 'LineWidth', 2, 'MaxHeadSize', 0.5);
text(1.6,0,0, 'X_s'); text(0,1.6,0, 'Y_s'); text(0,0,1.6, 'Z_s');

%% 3. 定义刚体数据
L = 1; % 半边长
% 局部顶点 (8个点)
local_verts = [ -L -L -L;  L -L -L;  L  L -L; -L  L -L;
                -L -L  L;  L -L  L;  L  L  L; -L  L  L ]';
% 面连接关系
faces = [1 2 3 4; 2 6 7 3; 4 3 7 8; 5 8 7 6; 1 5 6 2; 1 4 8 5];

% A系(中心) 和 B系(角点) 的局部偏移
offset_a_local = [0; 0; 0];
offset_b_local = [L; L; L];
% B系相对于刚体的局部旋转 (绕Z轴转45度)
theta_b = deg2rad(90);
R_b_local = [cos(theta_b) -sin(theta_b) 0; sin(theta_b) cos(theta_b) 0; 0 0 1];

%% 4. 初始化图形句柄 (创建空对象，稍后在循环中更新)

% 4.1 刚体 (透明立方体)
h_cube = patch('Vertices', local_verts', 'Faces', faces, ...
               'FaceColor', 'cyan', 'FaceAlpha', 0.15, ...
               'EdgeColor', 'k', 'LineStyle', ':');

% 4.2 坐标系 A (随动，显示在刚体中心)
h_ax = line([0 0], [0 0], [0 0], 'Color', 'r', 'LineWidth', 2);
h_ay = line([0 0], [0 0], [0 0], 'Color', 'g', 'LineWidth', 2);
h_az = line([0 0], [0 0], [0 0], 'Color', 'b', 'LineWidth', 2);

% 4.3 坐标系 B (随动，显示在刚体角点)
h_bx = line([0 0], [0 0], [0 0], 'Color', 'r', 'LineWidth', 1.5);
h_by = line([0 0], [0 0], [0 0], 'Color', 'g', 'LineWidth', 1.5);
```

```matlab
h_bz = line([0 0], [0 0], [0 0], 'Color', 'b', 'LineWidth', 1.5);

% 4.4 连接线 AB
h_line_ab = line([0 0], [0 0], [0 0], 'Color', 'k', 'LineWidth', 2.5);

% 4.5 速度场箭头 (Quiver对象)
% 在AB线上取点绘制速度矢量
h_quiver = quiver3(0,0,0,0,0,0, 'Color', 'm', 'LineWidth', 1.5, 'AutoScale', 'off');

%% 5. 动画仿真循环
dt = 0.05;
t_total = 20;

% 用于数值差分计算速度的变量
T_prev = [0;0;0];
R_prev = eye(3);
is_first_frame = true;

for t = 0:dt:t_total
    %% --- A. 运动学计算 (位置 T 和 姿态 R) ---
    % 1. 平动: 螺旋上升
    T_curr = [2*sin(0.5*t); 2*cos(0.5*t); 0.3*t - 2];

    % 2. 转动: 欧拉角随时间变化
    yaw = 0.5 * t;    % 绕Z
    pitch = 0.2 * t; % 绕Y
    roll = 0.3 * t;  % 绕X

    R_z = [cos(yaw) -sin(yaw) 0; sin(yaw) cos(yaw) 0; 0 0 1];
    R_y = [cos(pitch) 0 sin(pitch); 0 1 0; -sin(pitch) 0 cos(pitch)];
    R_x = [1 0 0; 0 cos(roll) -sin(roll); 0 sin(roll) cos(roll)];
    R_curr = R_z * R_y * R_x; % 当前姿态矩阵

    %% --- B. 动力学计算 (速度 v 和 角速度 omega) ---
    if is_first_frame
        v_linear = [0;0;0];
        omega = [0;0;0];
        is_first_frame = false;
    else
        % 线速度 v = dP/dt
        v_linear = (T_curr - T_prev) / dt;

        % 角速度 omega 计算: [omega]x = R_dot * R'
        R_dot = (R_curr - R_prev) / dt;
        W_skew = R_dot * R_curr';
        omega = [W_skew(3,2); W_skew(1,3); W_skew(2,1)];
```

```
    end
    % 更新历史状态
    T_prev = T_curr;
    R_prev = R_curr;

    %% --- C. 更新图形对象 ---

    % 1. 更新立方体顶点
    % Global_Verts = R * Local_Verts + T
    verts_global = R_curr * local_verts + repmat(T_curr, 1, 8);
    set(h_cube, 'Vertices', verts_global');

    % 2. 计算 A 系和 B 系的原点和基向量
    % A系 (中心)
    Origin_A = T_curr + R_curr * offset_a_local;
    Axes_A = R_curr * eye(3) * 0.8; % 轴长0.8

    % B系 (角点)
    Origin_B = T_curr + R_curr * offset_b_local;
    R_B_global = R_curr * R_b_local; % B系姿态 = 刚体姿态 * B相对刚体姿态
    Axes_B = R_B_global * eye(3) * 0.8; % 轴长0.6

    % 3. 绘制/更新坐标轴 (手动更新6条线)
    % A系更新
    set(h_ax, 'XData', [Origin_A(1), Origin_A(1)+Axes_A(1,1)], 'YData', [Origin_A(2), Origin_A(2)+Axes_A(2,1)], 'ZData', [Origin_A(3), Origin_A(3)+Axes_A(3,1)]);
    set(h_ay, 'XData', [Origin_A(1), Origin_A(1)+Axes_A(1,2)], 'YData', [Origin_A(2), Origin_A(2)+Axes_A(2,2)], 'ZData', [Origin_A(3), Origin_A(3)+Axes_A(3,2)]);
    set(h_az, 'XData', [Origin_A(1), Origin_A(1)+Axes_A(1,3)], 'YData', [Origin_A(2), Origin_A(2)+Axes_A(2,3)], 'ZData', [Origin_A(3), Origin_A(3)+Axes_A(3,3)]);

    % B系更新
    set(h_bx, 'XData', [Origin_B(1), Origin_B(1)+Axes_B(1,1)], 'YData', [Origin_B(2), Origin_B(2)+Axes_B(2,1)], 'ZData', [Origin_B(3), Origin_B(3)+Axes_B(3,1)]);
    set(h_by, 'XData', [Origin_B(1), Origin_B(1)+Axes_B(1,2)], 'YData', [Origin_B(2), Origin_B(2)+Axes_B(2,2)], 'ZData', [Origin_B(3), Origin_B(3)+Axes_B(3,2)]);
    set(h_bz, 'XData', [Origin_B(1), Origin_B(1)+Axes_B(1,3)], 'YData', [Origin_B(2), Origin_B(2)+Axes_B(2,3)], 'ZData', [Origin_B(3), Origin_B(3)+Axes_B(3,3)]);

    % 4. 更新连线 AB
    set(h_line_ab, 'XData', [Origin_A(1) Origin_B(1)], ...
```

```
                 'YData', [Origin_A(2) Origin_B(2)], ...
                 'ZData', [Origin_A(3) Origin_B(3)]);

    % 5. 计算并更新速度场 (核心部分)
    % 在 AB 线段上取 8 个采样点
    num_samples = 8;
    s_param = linspace(0, 1, num_samples);
    P_samples = zeros(3, num_samples);
    V_samples = zeros(3, num_samples);

    for k = 1:num_samples
        % 当前采样点位置 (线性插值)
        P_k = Origin_A + s_param(k) * (Origin_B - Origin_A);
        P_samples(:, k) = P_k;

        % 速度计算: v_k = v_a + omega x r_ak
        r_ak = P_k - Origin_A; % 从转动中心指向采样点的向量
        v_rot = cross(omega, r_ak);
        V_samples(:, k) = v_linear + v_rot;
    end

    % 更新 Quiver 数据
    scale = 0.3; % 缩小箭头比例，避免太长
    set(h_quiver, 'XData', P_samples(1,:), 'YData', P_samples(2,:), 'ZData', P_samples(3,:), ...
                  'UData', V_samples(1,:)*scale, ...
                  'VData', V_samples(2,:)*scale, ...
                  'WData', V_samples(3,:)*scale);

    drawnow;
    pause(0.01); % 如果电脑太快可以取消注释这行
end
```