

Supplementary Material for Spatial-Temporal Motion Control via Composite Cam-follower Mechanisms

This supplementary material is composed of four parts. The first part presents an experiment to show that the follower motion axis \mathbf{v}_s should not deviate too much from its default direction (see Section 4.1 and Figure 7 in the paper). The second part provides details about the kinematic modeling of camMechs (see Section 4.2 in the paper). The third part provides details about our approach to model dynamics of a given camMech (see Section 4.3 in the paper). The last part presents an experiment to determine a suitable number of control points for the unknown function $s(t)$ represented as a cubic spline (see Section 6.1 in the paper).

1 Follower-support Joint Orientation

For each class of camMech, the orientation of the follower-support joint is defined by a vector \mathbf{v}_s , called the follower motion axis. By default, \mathbf{v}_s is parallel with one of the major axes of the camMech's local frame, as shown in Figure 1(a). To support more flexible follower motion, \mathbf{v}_s can deviate from its default direction by an angle α ; see Figure 1(b-g). In our design, we prefer a small deviation angle α . This is because a larger deviation angle α will likely lead to worse dynamic performance of the camMech.

To support this statement, we use our dynamic modeling approach (see Section 4.2 in the paper) to compute a required motor torque to drive each of the seven camMechs for a whole motion period. To ensure a fair comparison, we assume that: 1) the external load forces for all the seven camMechs are the same; 2) the cams in the

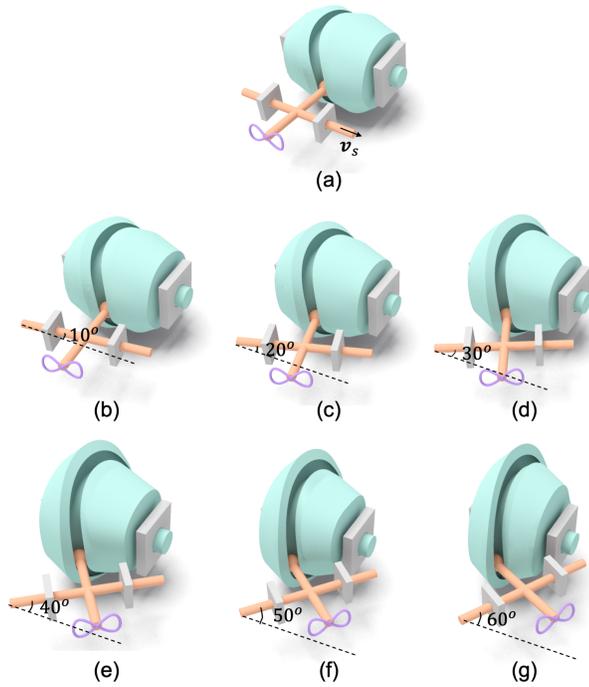


Figure 1: (a) The follower motion axis \mathbf{v}_s is parallel with the cam axis for camMech_ITIR by default. (b-g) The follower motion axis \mathbf{v}_s deviates from its default direction (see the dashed line) by an angle α of 10, 20, 30, 40, 50, 60 degrees, respectively.

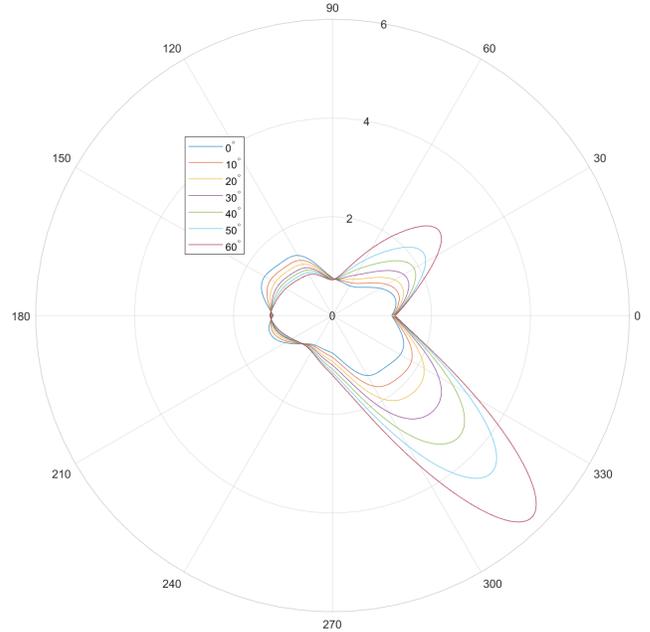


Figure 2: The motor torque $|\tau_c(t)|$ required to drive the cam in each of the seven camMechs in Figure 1 for a whole motion period. The correspondence between the torque curves and the camMechs in Figure 1 is based on the value of the deviation angle α .

camMechs rotate uniformly with the same speed. Figure 2 visualizes the required motor torque $|\tau_c(t)|$, $t \in [0, T]$, for all the camMechs, from which we can see that the average torque and the maximum torque increase when the deviation angle α becomes larger. Hence, we prefer a small deviation angle $\alpha < \alpha_{\text{thres}} = 45^\circ$ in our design, and choose $\alpha = 0^\circ$ if conditions permit.

2 Kinematic Modeling

In the paper, we have formulated a general representation for the kinematic equation of all the camMechs (i.e., Equation 3 in the paper):

$$G(\mathbf{T}_c(t) \mathbf{C}(s), \mathbf{C}(0)) = 0 \quad (1)$$

In the following, we provide a list of Equation 1 with concrete expression for each class of camMechs, where $\mathbf{T}_c(t)$ is a 4x4 rotation matrix that represents the cam's transformation, $\mathbf{C}(s)$ is the pitch curve of the cam, \mathbf{p}_s and \mathbf{v}_s represent the position and orientation of the follower-support joint respectively.

In camMech_IT:

$$\|(\mathbf{T}_c(t) \mathbf{C}(s) - \mathbf{C}(0)) \times \mathbf{v}_s\| = 0 \quad (2)$$

In camMech_IO:

$$\begin{aligned} \|(\mathbf{T}_c(t) \mathbf{C}(s) - \mathbf{p}_s)\| - \|(\mathbf{C}(0) - \mathbf{p}_s)\| &= 0 \\ (\mathbf{T}_c(t) \mathbf{C}(s) - \mathbf{p}_s) \cdot \mathbf{v}_s &= 0 \end{aligned} \quad (3)$$

In camMech_1R:

$$\begin{aligned} \|(\mathbf{T}_c(t) \mathbf{C}(s) - \mathbf{p}_s)\| - \|(\mathbf{C}(0) - \mathbf{p}_s)\| &= 0 \\ (\mathbf{T}_c(t) \mathbf{C}(s) - \mathbf{C}(0)) \cdot \mathbf{v}_s &= 0 \end{aligned} \quad (4)$$

In camMech_2R:

$$\| \mathbf{T}_c(t) \mathbf{C}(s) - \mathbf{p}_s \| = \| \mathbf{C}(0) - \mathbf{p}_s \| \quad (5)$$

In camMech_2T:

$$(\mathbf{T}_c(t) \mathbf{C}(s) - \mathbf{C}(0)) \cdot \mathbf{v}_s = 0 \quad (6)$$

In camMech_1T1R:

$$\|(\mathbf{T}_c(t) \mathbf{C}(s) - \mathbf{p}_s) \times \mathbf{v}_s\| = \|(\mathbf{C}(0) - \mathbf{p}_s) \times \mathbf{v}_s\| \quad (7)$$

3 Dynamic Modeling

We take camMech_2R as an example to illustrate how our dynamic modeling approach (see Section 4.3 in the paper) computes a torque $\tau_c(t)$ required to drive the cam such that the cam can rotate uniformly while the follower can move with the known accelerations (i.e., $\mathbf{a}_f(t)$ and $\alpha_f(t)$) under the workload $\mathbf{F}_e(t)$. For simplicity, we omit time t in the below symbols.

Follower dynamics. Denote the follower's weight and moment of inertia as m_f and I_f respectively. I_f is an anisotropy tensor. Denote the acceleration of gravity as \mathbf{g} . In summary, the follower experiences the following forces that can drive its motion: gravity $\mathbf{F}_g = m_f \mathbf{g}$, workload \mathbf{F}_e , and supporting force \mathbf{N} ; see Figure.8 in paper. Based on Newton's Second Law for translation and rotation of the follower, we have

$$\begin{aligned} F_f(\mathbf{N}; \mathbf{F}_g, \mathbf{F}_e) &= m_f \mathbf{a}_f \\ M_f(\mathbf{N}, \mathbf{p}_N; \mathbf{F}_g, \mathbf{F}_e) &= I_f \alpha_f \end{aligned} \quad (8)$$

where \mathbf{F}_g , m_f , and I are constants while all other variables are functions with respect to time t . Note that only camMech_1T1R requires to formulate both F_f and M_f while others only need to formulate either one.

Particularly, Eq.8 in camMech_2R:

$$(\mathbf{p}_N - \mathbf{p}_s) \times \mathbf{N} + (\mathbf{p}_g - \mathbf{p}_s) \times \mathbf{F}_g + (\mathbf{p}_e - \mathbf{p}_s) \times \mathbf{F}_e = I_f \alpha_f$$

At a given time t , the values of $\mathbf{p}_g(t)$, $\mathbf{p}_e(t)$, and $\alpha_f(t)$ can be obtained from the kinematic modeling.

Denote the velocity of the follower ball center $\mathbf{p}_p(t)$ as $\mathbf{v}_p(t)$, which can be calculated in the kinematic modeling. The relative velocity between the cam and the follower at point $\mathbf{p}_p(t)$ is a known variable: $\mathbf{v}_r = \boldsymbol{\omega}_c \times \mathbf{p}_p - \mathbf{v}_p$, where $\boldsymbol{\omega}_c(t)$ is the cam's angular velocity. Note that the direction of \mathbf{v}_r is identical with the tangent of the pitch curve at point $\mathbf{p}_p(t)$. Due to the way that we model the groove surface by sweeping the follower ball along the pitch curve, the supporting force \mathbf{N} should be perpendicular to \mathbf{v}_r :

$$\mathbf{N} \perp \mathbf{v}_r \quad (9)$$

The contact point \mathbf{p}_N is always on the follower ball surface and its actual location determines the direction of the supporting force \mathbf{N} :

$$(\mathbf{p}_p - \mathbf{p}_N)/R = \mathbf{N}/\|\mathbf{N}\| \quad (10)$$

where R is the radius of the follower ball. Based on Equations 8–10, we can solve instantaneous \mathbf{N} and \mathbf{p}_N at any time t .

3D cam dynamics. Denote the cam's moment of inertia as I_c , which is a scalar since the axis of rotation is fixed. Based on Newton's Second Law for rotation of the 3D cam, we have

$$M_c(\tau_c; \mathbf{N}, \mathbf{p}_N) = I_c \alpha_c \quad (11)$$

Gravity will not create any torque for the cam as we assume the cam centroid locates at its rotation axis. τ_c can be obtained by solving Equation 11 as it is the only unknown variable in the equation.

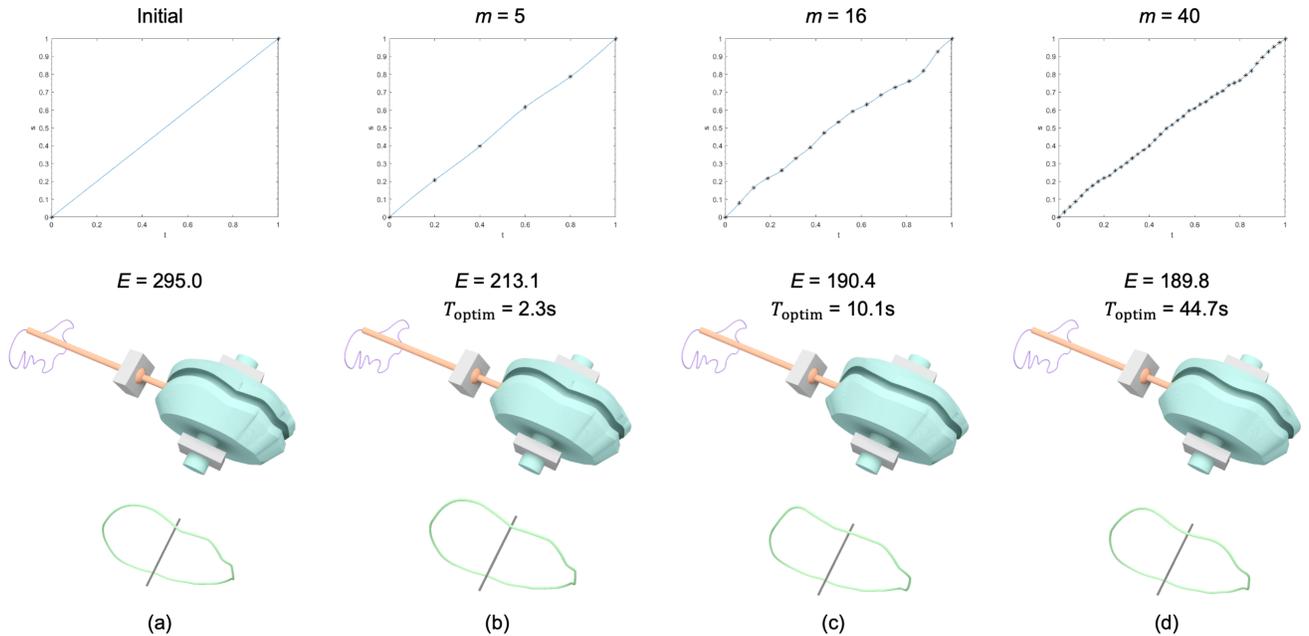


Figure 3: Starting from (a) the same initial camMech, we use our optimization-based approach to design (b-d) three camMechs, each of which represents the unknown function $s(t)$ using a cubic spline with 5, 16, and 40 control points, respectively. From top to bottom: the initial and optimized function $s(t)$, where control points in (b-d) are colored in black; the corresponding camMech design, energy function value E , and the optimization time T_{optim} ; and the corresponding pitch curve.

4 Control points of $s(t)$

In Section 6.1 of the paper, we choose to represent the unknown function $s(t)$ as a cubic spline, and initialize it as $s(t) = t$. We choose 10 to 30 control points according to the target curve's shape complexity. In the following, we present an experiment to show that the range of 10 to 30 control points of $s(t)$ is sufficient for our optimization.

In this experiment, we used our approach in Section 6.1 of the paper to design three camMech_2R for realizing a T-REX curve. When designing these three camMechs, the function $s(t)$ is represented as a cubic spline with m control points, where $m = 5, 16,$ and 40 respectively; see Figure 3(b-d). All these three designs take the same camMech with $s(t) = t$ as an initialization of the optimization; see Figure 3(a). During the optimization, we only search for the control points of the spline curve $s(t)$ while keeping the other design parameters (i.e., distance L and angle θ) fixed.

Figure 3(top) shows the initial and optimized spline curves. Figure 3(middle) shows each resulting camMech design as well as its energy function value E and the optimization time T_{optim} . When the number of control points m is increased from 5 to 16, there is a significant drop in the energy value (i.e., from 213.1 to 190.4). However, when the number of control points m is further increased from 16 to 40, the drop in the energy value is very small (from 190.4 to 189.8), at the cost of significantly more computational time for solving the optimization (from 10.1 seconds to 44.7 seconds). Hence, we choose 10 to 30 control points for the unknown function $s(t)$ since it allows to get a good enough optimization result with reasonable computational cost.