Proceedings of the ASME 2026
International Design Engineering Technical Conferences &
Computers and Information in Engineering Conference
IDETC/CIE 2026
August 23–26, 2026, Houston, TX

IDETC2026-194810

TUNABLE KINEMATIC SMOOTHNESS FOR GENERATIVE MECHANISM SYNTHESIS

Diana Bolanos[1,*], Mohammadmehdi Ataei[2], Kosa Goucher-Lambert[1]

[1]University of California, Berkeley, CA, USA
[2]Autodesk Research, Toronto, ON, Canada

ABSTRACT

Mechanism synthesis is well established for path, motion, and function generation tasks, with recent generative approaches achieving state-of-the-art curve-following accuracy. Yet geometric path fidelity captures only one dimension of mechanism quality. In many engineering applications, the kinematic character of the coupler trajectory, how smoothly and uniformly the end-effector moves through its arc, is equally consequential. A mechanism that traces the desired path but exhibits velocity spikes or high jerk may be unsuitable wherever motion regularity directly affects performance. We present a post-generation selection framework that introduces kinematic smoothness as a tunable design objective atop a pretrained generative synthesis model, requiring no retraining. Given a target coupler curve, the framework generates a pool of candidate mechanisms via two complementary sampling strategies: affine input-space sampling and crank prefix conditioning. Each candidate is then scored on geometric fidelity and a composite speed-invariant smoothness loss. A user-specified weighting then combines the two objectives, exposing the Pareto trade-off between path accuracy and kinematic smoothness. The framework operates entirely at inference time and generalizes to any synthesis method that produces multiple simulable candidates. Evaluated across 56,000 candidate mechanisms spanning 500 validation samples, combining both sampling strategies reduces median smoothness loss by 91% relative to an unguided baseline, while the balanced preset reduces smoothness loss to roughly one-third of the curve-only value at a cost of only 0.24 additional DTW units.

Keywords: mechanism synthesis, kinematics, user-centered design

1. INTRODUCTION

Planar linkage mechanisms remain among the most widely used motion-generation devices in mechanical engineering, appearing in applications ranging from robotic grippers and medical devices to automotive suspensions and manufacturing automation. The classical path synthesis problem (given a desired coupler curve, find the linkage dimensions that produce it) is notoriously difficult due to its mixed discrete-continuous search space and the many-to-one mapping from mechanism parameters to output trajectories. Recent advances in deep generative modeling have introduced promising alternatives to traditional numerical synthesis [1–3]. In our recent work, we present MechaFormer [4], a model that treats mechanism design as a conditional sequence generation task, producing a complete mechanism topology and joint geometry in a single forward pass conditioned on a B-spline representation of the target curve, achieving state-of-the-art path-following accuracy as measured by Dynamic Time Warping (DTW) distance [5].

However, curve-following fidelity is only one dimension of mechanism quality. In engineering practice, designers routinely impose qualitative requirements on how a mechanism moves, not just where it goes. A coupler point intended for a haptic feedback device must traverse its path without transmitting velocity irregularities to the user. Precision manufacturing machinery cannot tolerate speed spikes that degrade positional accuracy. These requirements reflect what the design literature terms *semantic attributes* - qualitative dimensions of design quality that are consequential for downstream use yet absent from standard evaluation metrics [6, 7]. Whereas prior work on semantic attributes has focused on perceptual qualities such as comfort or expressiveness, kinematic smoothness occupies a distinctive position on this spectrum: it originates as a qualitative designer judgment ("this motion should be smooth") yet can be modeled using a simulation-derived formalization. Current generative synthesis approaches evaluate candidates almost exclusively on geometric fidelity, leaving this dimension of design intent unaddressed.

In this paper, we address this gap with a post-generation selection framework that introduces kinematic smoothness as a tunable design objective without retraining the generative model. Given a target curve, the framework generates a diverse pool of

*Corresponding author: dbolanos@berkeley.edu

candidate mechanisms via two complementary sampling strategies: affine input-space sampling and crank prefix conditioning. Each candidate is then scored on geometric fidelity and a composite speed-invariant smoothness loss, and selects the mechanism whose weighted combination of scores best reflects the designer's intent. The framework operates entirely at inference time. Affine sampling and the selection pipeline generalize to any synthesis method that produces multiple simulable candidates, and the smoothness objective can be replaced with any simulable property of the coupler trajectory.

We evaluate the framework on 500 randomly selected validation samples. Compared to a random baseline that selects one candidate uniformly from the same pool, weighted selection consistently outperforms unguided sampling for every strategy. Combining both sampling strategies into a combined pool of 112 candidates, the smooth preset reduces median smoothness loss by 91% relative to the random baseline (0.294 to 0.026), while the balanced preset achieves less than half the smoothness loss of curve-only selection within 0.24 DTW units of curve-optimal path fidelity.

2. RELATED WORK

2.1. Semantic Attributes in Engineering Design

Engineering design artifacts are evaluated not only against functional performance requirements but also against qualitative attributes such as smoothness, elegance, or comfort that capture intangible dimensions of design quality relevant to downstream use [7, 8]. Formalizing these attributes as computable design objectives is challenging because they are often subjective, context-dependent, and not directly encoded in standard simulation outputs. Nevertheless, a growing body of work has demonstrated that qualitative attributes can be systematically integrated into computational design workflows.

In the domain of human-robot interaction, Desai et al. [6] demonstrated that expressive motion qualities such as emotional character can be mapped from subjective human ratings to parameterized motion attributes through data-driven regression, enabling semantic-level design of robot behaviors. This work established that qualitative attributes can serve as first-class design objectives even when they are inherently perceptual and subjective. More recently, Nandy and Goucher-Lambert [7] showed that abstract semantic attributes of physical products, such as how comfortable a mug feels to hold, can be embedded into parameterized designs through interactive preference learning, and that individualized adaptation improves alignment between computational outputs and human perception. Complementary behavioral work by Nandy et al. [9] demonstrated that the psycholinguistic properties of attribute descriptors themselves shape design outcomes, suggesting that how an attribute is specified is as consequential as what it specifies.

When multiple attributes must be satisfied simultaneously, candidate selection becomes a multi-objective problem. Previous work [10] illustrated this in the context of deployable space arrays, where origami source patterns were evaluated against competing metrics using decision matrices with tunable weights to navigate trade-offs among incommensurable criteria. This weighted selection approach reflects a general principle. When no candidate dominates all objectives simultaneously, designer intent must be encoded through the relative weighting of objectives, and the selection framework should enable designers to explore the trade-off landscape before committing to a single candidate selection.

Our work applies this principle to planar mechanism synthesis. Kinematic smoothness represents a qualitative attribute of mechanism output that is not captured by standard path accuracy metrics, requires simulation to evaluate, and trades off against geometric fidelity in a way that varies with application context. By defining a computable smoothness metric and embedding it in a weighted multi-objective selection framework, this paper extends the attribute-conditioned design paradigm to the kinematic synthesis domain.

2.2. Smoothness Characterization Across Domains

Kinematic smoothness has been independently formalized across several engineering and scientific disciplines, each developing metrics suited to its application context. This section surveys representative approaches that motivate and contextualize the composite metric introduced in Section 3.4.

2.2.1. Biomechanics. Quantifying movement smoothness became a clinical priority with the growth of robot-assisted rehabilitation, where therapists require objective measures of motor recovery. Flash and Hogan [11] first demonstrated that healthy reaching movements minimize integrated squared jerk, establishing jerk as the physiological cost function for voluntary motion and providing a foundational justification for jerk-based smoothness penalties. Hogan and Sternad [12] later developed the log dimensionless jerk (LDLJ), a scale-invariant form that enables comparison across movements of different amplitude and duration. Balasubramanian et al. [13] introduced the spectral arc length (SPARC), which measures smoothness in the frequency domain as the arc length of the normalized speed spectrum. SPARC has since been validated across multiple systematic reviews as a robust and bias-resistant smoothness metric [14].

2.2.2. Robotics. Industrial and collaborative robotics evaluate end-effector trajectory quality through velocity profile statistics, since speed irregularities directly affect manufacturing tolerances, cycle time, and human comfort in co-manipulation tasks. The speed coefficient of variation (CV), defined as the ratio of speed standard deviation to mean speed, quantifies how far a velocity profile deviates from constant motion [15]. This quantity maps directly to the CV_s term in our composite metric. The speed metric (SM), defined as the ratio of mean to peak speed, provides a complementary bound on worst-case speed profiles [13].

2.2.3. Computer Graphics. The curve fairness literature offers geometric analogs to kinematic smoothness that are instructive precisely because of where they fall short. Moreton and Séquin [16] formalized curve quality through variational functionals, where elastic energy minimizes the arc-length integral of squared curvature $\int \kappa^2 \, ds$, penalizing overall bending, while their proposed minimum variation curves (MVC) instead minimize $\int (d\kappa/ds)^2 \, ds$, penalizing abrupt changes in curvature. The MVC functional is the geometric analog of jerk. Just as jerk-based metrics penalize rapid changes in velocity, MVC penalizes rapid changes in curvature, and the two criteria share the same variational motivation.

However, geometric fairness and kinematic smoothness are not equivalent. A coupler curve with low curvature variation can still be traversed with highly non-uniform speed. Meaning, if a mechanism dwells at one region of the path and rushes through another, the resulting motion is kinematically rough regardless of how geometrically smooth the curve appears. This decoupling arises because geometric metrics characterize the *shape* of a path, whereas kinematic metrics characterize how the path is *parameterized in time*. A trajectory traversed at constant speed is arc-length parameterized, but most linkage mechanisms produce paths that deviate substantially from arc-length parameterization, with instantaneous speed varying considerably over the operating arc.

2.2.4. Mechanism Design. A common mechanism quality metric is the minimum transmission angle $\mu_{\min}$ [17, 18], defined as the minimum acute angle between the coupler and follower links over the full operating arc. $\mu_{\min}$ governs the fraction of coupler force converted to useful output torque: as $\mu_{\min}$ approaches zero, the mechanism approaches a singular configuration and force transmission becomes mechanically inefficient. Standard practice requires $\mu_{\min} \geq 40$–$45°$ to ensure adequate torque throughout the cycle [19].

However, $\mu_{\min}$ characterizes force transmission feasibility, not kinematic motion quality. A mechanism that satisfies the transmission angle criterion throughout its arc may still exhibit highly non-uniform coupler speed. As a result, the coupler point can dwell, accelerate sharply, and decelerate within a single cycle while $\mu_{\min}$ remains well above the design threshold. This decoupling arises because the transmission angle is defined by the instantaneous geometry of the linkage, whereas speed uniformity depends on how that geometry evolves over time relative to the input motion. Thus, two mechanisms with identical $\mu_{\min}$ values can differ in their velocity profiles and jerk, yet both pass the standard quality criterion. The smoothness metric introduced in this paper addresses this orthogonal dimension of mechanism quality directly, quantifying the kinematic character of the output motion rather than the mechanical feasibility of force transmission.

2.3. Data-Driven Mechanism Synthesis

The kinematic path synthesis problem consists of finding a planar linkage whose coupler point traces a desired curve. This is notoriously difficult due to its mixed discrete-continuous search space and the many-to-one mapping from mechanism parameters to output trajectories [17, 18]. Classical analytical methods, including Burmester theory and precision-point synthesis, produce closed-form solutions for simple topologies and a small number of prescribed positions, but break down for continuous curves or richer linkage families [19]. Numerical optimization approaches cast synthesis as error minimization within a fixed topology, but are sensitive to initialization and prone to local minima in the non-convex landscape.

The availability of large-scale mechanism datasets has catalyzed a shift toward data-driven synthesis. Nobari et al. [20] introduced the LINKS dataset of 100 million single-DOF planar linkage mechanisms paired with coupler curves, while Nurizada et al. [21] released a curated dataset of 3 million four-, six-, and eight-bar mechanisms and reported evaluation metrics that have since become common benchmarks in the field.

Building on these resources, several generative approaches have been proposed. Nurizada et al. [1] used a conditional β-VAE to generate diverse four-bar mechanisms from coupler curves, though reconstruction accuracy remains limited for high-precision tasks. Fogelson et al. [2] proposed a reinforcement learning framework for generating high-order linkage graphs, achieving competitive performance but at substantial computational cost. We also contribute to recent advancements in automated kinematic design by reframing synthesis as conditional sequence generation [4]. In our work, a Transformer encoder-decoder translates a B-spline curve representation into a domain-specific language string that simultaneously encodes topology and quantized joint coordinates in a single forward pass. Combined with affine sampling and optional local optimization, MechaFormer achieves a median DTW of 0.887, substantially outperforming prior learning-based methods.

Recently, Purwar and Ge [3] have surveyed this emerging landscape and noted that data-driven methods have increasingly enabled simultaneous type and dimensional synthesis and can produce diverse, defect-free designs in real time. Most related to the present work, Lin et al. [22] demonstrated that a VAE conditioned on color-encoded trajectory speed can synthesize mechanisms that match both a target curve shape and a prescribed velocity profile. However, this approach requires the designer to specify the desired speed distribution as an input and necessitates a purpose-trained architecture. In contrast, the framework presented here introduces smoothness as a post-generation *quality criterion* rather than a design specification, operates entirely at inference time on any pretrained synthesis model, and exposes the trade-off between path fidelity and kinematic smoothness without requiring the designer to prescribe a target velocity profile.

3. METHODS

We propose a post-generation selection framework (Fig. 1) that augments MechaFormer with motion-quality awareness. Given a target coupler curve C^*, the framework (i) generates a diverse pool of candidate mechanisms via two complementary sampling strategies that intervene at different points in the generation pipeline – affine input-space sampling transforms the target curve before inference, while crank prefix conditioning biases the decoder during generation, (ii) simulates each candidate to obtain its coupler trajectory, (iii) scores each candidate on both geometric fidelity and kinematic smoothness, and (iv) selects the candidate whose weighted combination of scores best matches the designer's intent. These candidates are evaluated under a two-objective criterion consisting of geometric fidelity to the target curve and kinematic smoothness of the resulting motion. The framework operates entirely at inference time and requires no retraining or fine-tuning of any component.

3.1. MechaFormer and Simulation

MechaFormer treats planar mechanism synthesis as a conditional sequence generation problem. A target coupler curve is encoded as a cubic B-spline with $K = 64$ control points and

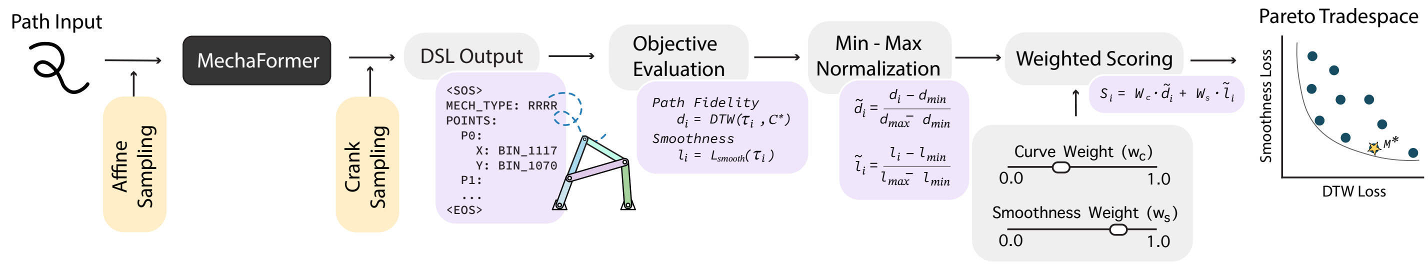

FIGURE 1: Smoothness-aware selection pipeline. A target coupler curve is processed using one or both sampling strategies to generate a pool of candidate mechanisms. Each candidate is then evaluated on path fidelity (DTW) and kinematic smoothness ($\mathcal{L}_{smooth}$). After min–max normalization, the two objectives are combined using user-specified weights to select a mechanism from the resulting Pareto trade-off space.

passed to a Transformer encoder–decoder, which autoregressively generates a domain-specific language (DSL) string specifying both mechanism topology - selected from 24 kinematically valid types spanning four-bar and six-bar linkages - and the quantized joint coordinates of all free joints. This unified representation produces a complete mechanism in a single forward pass.

Given a generated mechanism

$$\hat{m} = (\tau, J, c),$$

its coupler trajectory τ is obtained from the deterministic forward kinematic simulator which integrates the kinematic structure definitions for all 24 topology types. Simulation is computationally inexpensive, taking under 1 ms per mechanism, making it practical to evaluate large candidate pools.

Geometric fidelity is measured using Dynamic Time Warping (DTW) [5] distance between the simulated coupler trajectory and the target curve. A score between 2 and 3 indicates moderate curve approximation, and a score less than 2 indicates satisfactory performance. To enable fair comparison across curves of different scales, DTW distance is normalized by the RMS variance of the input curve. MechaFormer is trained exclusively to minimize this DTW loss and therefore provides no explicit control over the kinematic character of the resulting motion. Two mechanisms with identical DTW scores may differ substantially in speed uniformity and jerk. The smoothness metric introduced in Section 3.4 addresses this limitation. Sections 3.2 and 3.3 first describe how a diverse candidate pool is sampled and generated.

3.2. Affine Transformation Sampling

A single MechaFormer query produces one mechanism candidate. To obtain a diverse pool suitable for multi-objective selection, we exploit MechaFormer's canonical frame normalization to sample a local neighborhood of rigid-body placements in the SE(2) workspace. Because the model operates on curves normalized to a fixed frame, applying an affine transformation to the input curve before inference and inverting it afterward yields a kinematically distinct candidate that traces the same geometric path from a different base placement and orientation in the workspace.

We define an affine transformation grid $\mathcal{G}$ consisting of all combinations of $N_r = 8$ rotations and $N_t = 7$ translations:

$$\mathcal{G} = \{(R_\theta, \mathbf{t}) \mid \theta \in \Theta,\ \mathbf{t} \in \mathbf{T}\}.$$

Algorithm 1 Smoothness-Weighted Mechanism Selection

Require: Target control points $\mathbf{P}$; weights $w_c, w_s \geq 0,\ w_c + w_s = 1$
Require: Affine grid $\mathcal{G}$; crank grid $\mathcal{H}$
Ensure: Selected mechanism m^* with trajectory τ^*

$C^* \leftarrow \text{BSpline}(\mathbf{P})$
$\mathcal{R} \leftarrow \emptyset$
▷ **Phase 1: Candidate generation**
$\mathcal{R} \leftarrow \text{AffineTransformation}(\mathbf{P}, \mathcal{G})$
$\mathcal{R} \leftarrow \mathcal{R} \cup \text{CrankGeneration}(\mathbf{P}, \mathcal{H})$
▷ **Phase 2: Multiobjective scoring**
for all $(\hat{m}_i, \tau_i) \in \mathcal{R}$ **do**
  $d_i \leftarrow \text{DTW}(\tau_i, C^*)$
  $\ell_i \leftarrow \mathcal{L}_{\text{smooth}}(\tau_i)$
end for
▷ **Phase 3: Weighted selection**
$\tilde{d}_i \leftarrow \dfrac{d_i - \min_j d_j}{\max_j d_j - \min_j d_j} \quad \forall i$
$\tilde{\ell}_i \leftarrow \dfrac{\ell_i - \min_j \ell_j}{\max_j \ell_j - \min_j \ell_j} \quad \forall i$
$s_i \leftarrow w_c\, \tilde{d}_i + w_s\, \tilde{\ell}_i \quad \forall i$
$m^* \leftarrow \arg\min_i\ s_i$
return $m^*,\ \tau^*,\ \{(d_i, \ell_i)\}$

The rotation set covers the full circle at equal increments:

$$\Theta = \{0^\circ, 45^\circ, 90^\circ, 135^\circ, 180^\circ, 225^\circ, 270^\circ, 315^\circ\}.$$

The translation set forms a structured sparse grid in normalized coordinates:

$$\mathbf{T} = \{(0,0),\ (\pm 0.3, 0),\ (0, \pm 0.3),\ (0.3, 0.3),\ (-0.3, -0.3)\}.$$

This set covers the origin, four axis-aligned offsets, and two diagonal offsets at ± 0.3 in the normalized workspace.

For each $(R_\theta, \mathbf{t}) \in \mathcal{G}$, the transformed input curve

$$\tilde{C} = R_\theta C^* + \mathbf{t}$$

is passed to MechaFormer. The resulting mechanism and its simulated trajectory are mapped back to the original frame using the inverse transform

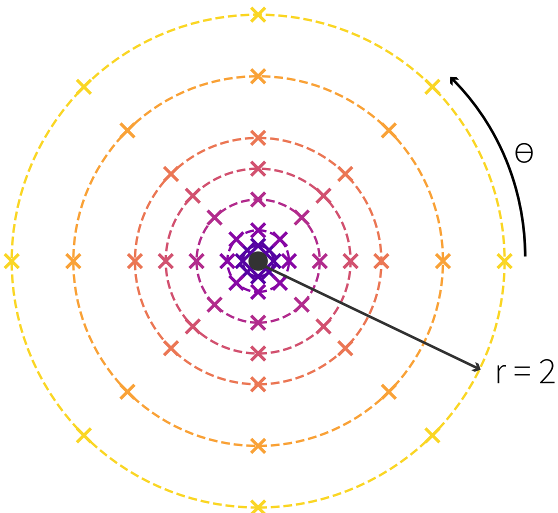

FIGURE 2: Crank sampling grid $\mathcal{H}$. Each cross marks one prefix location $\mathbf{p}(r, \theta)$; color encodes radius $r \in \mathcal{R}$ from smallest (dark) to largest (light). The ground pivot is fixed at the origin.

$$\tau = R_\theta^\top(\tilde{\tau} - \mathbf{t}).$$

This process yields a pool of

$$|\mathcal{G}| = 56$$

candidate mechanisms for each query curve.

3.3. Crank Sampling

While affine sampling intervenes before MechaFormer receives its input, crank sampling intervenes during decoding. MechaFormer's autoregressive generation follows a fixed prediction order: the mechanism type token is predicted first, followed by joint coordinates in sequence. Both can be supplied as prefix tokens before decoding begins. The mechanism type may be fixed by the designer – for example, to restrict generation to RRRR four-bar mechanisms – or left to MechaFormer to predict freely as its first decision. The crank pin, the first predicted joint, can similarly be fixed to a target position, with all remaining joints predicted freely conditioned on that prefix. The prefix therefore acts as a soft constraint: the model may shift the crank pin to the nearest kinematically feasible position, but generation is strongly biased toward mechanisms whose crank geometry is consistent with the supplied location.

We parameterize the crank-pin prefix as a point on a circle of radius r centered at the ground pivot, at angle θ measured from the positive x-axis:

$$\mathbf{p}(r, \theta) = \big(r\cos\theta,\ r\sin\theta\big).$$

The sampling grid $\mathcal{H}$ is defined over $N_r = 7$ radii and $N_\theta = 8$ angles, giving $|\mathcal{H}| = 56$ candidates per query curve, matching the affine pool size and enabling direct comparison. The radius set spans crank lengths in the normalized workspace,

$$\mathcal{R} = \{0.125,\ 0.25,\ 0.5,\ 0.75,\ 1.0,\ 1.5,\ 2.0\},$$

Algorithm 2 Crank Candidate Generation

Require: Target control points $\mathbf{P}$; crank grid $\mathcal{H} = \{\mathbf{p}(r, \theta)\}$
Ensure: Candidate pool $\mathcal{R}$

$\mathcal{R} \leftarrow \emptyset$
▷ **Prefix-conditioned candidate generation**
for all $\mathbf{p}(r, \theta) \in \mathcal{H}$ **do**
  $\hat{m} \leftarrow$ MechaFormer($\mathbf{P}$, `prefix=`$\mathbf{p}(r, \theta)$) ▷ Fix crank position; predict remaining joints freely
  $\tau \leftarrow$ Simulate($\hat{m}$) ▷ Kinematic simulation
  if generation or simulation failed **then**
    continue
  end if
  $\mathcal{R} \leftarrow \mathcal{R} \cup \{(\hat{m}, \tau)\}$
end for
return $\mathcal{R}$

---

and the angle set covers the full circle at equal increments,

$$\Theta = \{0^\circ, 45^\circ, 90^\circ, 135^\circ, 180^\circ, 225^\circ, 270^\circ, 315^\circ\},$$

matching the rotation set used in affine sampling (Section 3.2). Unlike affine candidates, crank candidates require no inverse transform: simulation and scoring are performed directly in the normalized frame. The sampling grid $\mathcal{H}$ is illustrated in Fig. 2 and a detailed description of the process is described in Algorithm 2.

3.4. Speed-Invariant Smoothness Metric

We quantify motion quality with a composite loss that is invariant to operating speed, making it applicable across mechanisms of different scales and input velocities.

Given a simulated coupler trajectory $\mathbf{p} = \{(x_t, y_t)\}_{t=1}^{T}$, we form the kinematic chain via finite differences:

$$\begin{aligned} \mathbf{v}_t &= \mathbf{p}_{t+1} - \mathbf{p}_t, \quad t = 1, \ldots, T-1, \\ \mathbf{a}_t &= \mathbf{v}_{t+1} - \mathbf{v}_t, \quad t = 1, \ldots, T-2, \\ \mathbf{j}_t &= \mathbf{a}_{t+1} - \mathbf{a}_t, \quad t = 1, \ldots, T-3, \end{aligned}$$

with instantaneous speed $s_t = \|\mathbf{v}_t\|_2$ and mean speed $\bar{s} = \frac{1}{T-1}\sum_{t=1}^{T-1} s_t$. From these we define two dimensionless submetrics normalized by $\bar{s}$:

$$\mathrm{CV}_s = \frac{\sigma(s)}{\bar{s}}, \tag{1}$$

$$\hat{J} = \frac{\mathrm{RMS}(\|\mathbf{j}_t\|_2)}{\bar{s}}. \tag{2}$$

CV_s is the coefficient of variation of speed: it equals zero for perfectly constant-velocity motion and grows as the instantaneous speed deviates from its cycle mean, penalizing the acceleration and deceleration that characterize kinematically rough mechanisms. $\hat{J}$ measures higher-order roughness through the root-mean-square (RMS) of the jerk magnitude over the simulated arc, normalized by $\bar{s}$ to preserve speed invariance:

$$\hat{J} = \frac{\sqrt{\frac{1}{T-3}\sum_{t=1}^{T-3}\|\mathbf{j}_t\|_2^2}}{\bar{s}}. \tag{3}$$

Using RMS rather than the peak or mean of $\|\mathbf{j}_t\|_2$ integrates roughness uniformly over the full motion arc, avoiding sensitivity to isolated impulse events while still penalizing sustained jerkiness. This is analogous to the minimum-jerk criterion in trajectory planning [11], which minimizes the integral of squared jerk; here we instead use the RMS as a dimensionless scalar compatible with the weighted scalarization in Eq. 4. Together, CV_s captures first-order non-uniformity (uneven speed) and $\hat{J}$ captures third-order non-uniformity (jerk), so their combination penalizes both classes of kinematic roughness. Both components have independent empirical grounding - speed coefficient of variation is widely used in rehabilitation robotics to assess movement quality [15], and jerk-based penalties originate in the minimum-jerk model of voluntary human movement [11].

$$\mathcal{L}_{\text{smooth}} = \alpha\,\mathrm{CV}_s + (1-\alpha)\,\hat{J}, \qquad \alpha \in [0,1] \tag{4}$$

The parameter α controls the relative emphasis on speed non-uniformity versus jerk. Because both CV_s and $\hat{J}$ are dimensionless ratios normalized by $\bar{s}$, they share a common scale without further adjustment, and we adopt the default $\alpha = 0.5$ throughout this study. This equal weighting treats first-order non-uniformity (uneven speed) and third-order non-uniformity (uneven jerk) as equally undesirable in the absence of application-specific prior knowledge. In practice, α can be adjusted to suit domain requirements: a lower value emphasizes jerk reduction for vibration-sensitive applications, while a higher value prioritizes speed uniformity for constant-velocity tasks. Because both terms are ratios with respect to $\bar{s}$, $\mathcal{L}_{\text{smooth}}$ measures the *shape* of the velocity profile rather than its magnitude. A mechanism that traces the same path at twice the speed is scored identically to the original, provided it does so with the same kinematic regularity.

3.5. User-Defined Weights and Candidate Selection

Each candidate mechanism $\hat{m}_i$ is evaluated using two scalar objectives: path fidelity and kinematic smoothness. Path fidelity is measured using the DTW distance between the simulated coupler trajectory τ_i and the target curve C^*,

$$d_i = \mathrm{DTW}(\tau_i, C^*),$$

while kinematic smoothness is quantified using the smoothness loss

$$\ell_i = \mathcal{L}_{\text{smooth}}(\tau_i).$$

Because the two objectives have different units and dynamic ranges, they are normalized across the candidate pool using min–max scaling,

$$\tilde{d}_i = \frac{d_i - \min_j d_j}{\max_j d_j - \min_j d_j}, \qquad \tilde{\ell}_i = \frac{\ell_i - \min_j \ell_j}{\max_j \ell_j - \min_j \ell_j}.$$

A composite score is then computed using a convex combination

$$\mathcal{S}_i = w_c \tilde{d}_i + w_s \tilde{\ell}_i, \qquad w_c + w_s = 1,$$

and the selected mechanism is

$$\hat{m}^* = \arg\min_i \mathcal{S}_i.$$

Normalization is performed relative to the generated candidate pool. As a result, the weights (w_c, w_s) express the *relative* importance of the two objectives given the variability present among the candidates rather than a fixed global trade-off. If the candidate mechanisms exhibit little variation in DTW error, for example, the normalized path term $\tilde{d}_i$ contributes limited discriminative signal and selection is naturally driven by smoothness. This behavior allows the selector to adapt automatically to the informative dimension of the candidate pool.

To support common design intents, three preset weight configurations are provided:

Preset	w_c	w_s	Design intent
Curve	1.0	0.0	Maximize path fidelity
Balanced	0.5	0.5	Equal emphasis on both objectives
Smooth	0.0	1.0	Maximize kinematic smoothness

In addition to selecting the minimum-score mechanism, the framework computes the Pareto front using the raw objective values. This provides a view of the achievable trade-off between path fidelity and kinematic smoothness for a given query curve and allows designers to select alternative Pareto-optimal mechanisms if desired. The details of this procedure are laid out in Algorithm 1.

4. EXPERIMENTS AND RESULTS

4.1. Objective Space Analysis

Figures 3 and 4 show the objective spaces produced by affine and crank sampling respectively, both generated from Sample 596. Each point represents one candidate mechanism. Lower values are preferred on both axes. The dashed line traces the Pareto front of non-dominated solutions. Three annotated candidates correspond to the arg min selection under each preset, curve ($w_c = 1.0,\ w_s = 0.0$), balanced ($w_c = 0.5,\ w_s = 0.5$), and smooth ($w_c = 0.0,\ w_s = 1.0$).

The affine study spans a wide range of both objectives. Candidate **a** achieves near-zero DTW but at high smoothness cost ($\mathcal{L}_{\text{smooth}} \approx 0.57$); candidate **b** lies near the Pareto elbow with low scores on both objectives ($\mathcal{L}_{\text{smooth}} \approx 0.07$); and candidate **c** achieves the lowest smoothness loss in the pool at substantial path cost (DTW ≈ 10). The Pareto front exposes a clear trade-off between curve-following fidelity and kinematic regularity, with no candidate simultaneously minimizing both objectives. The kinematic profiles of candidates **a**, **b**, and **c** are examined in Section 4.2.

For the crank study, the mechanism type was fixed to RRRR prior to generation. This reflects a common design scenario in which the engineer has already committed to a four-bar revolute topology and wishes to explore dimensional diversity within that family. Alternatively, the mechanism type prefix can be omitted, allowing MechaFormer to select the topology autoregressively as its first prediction. The three annotated candidates (**d**, **e**, **f**) follow the same preset labeling as the affine plot. Compared to the affine pool, crank candidates spread across a wider range of DTW error. The two Pareto fronts occupy complementary regions of

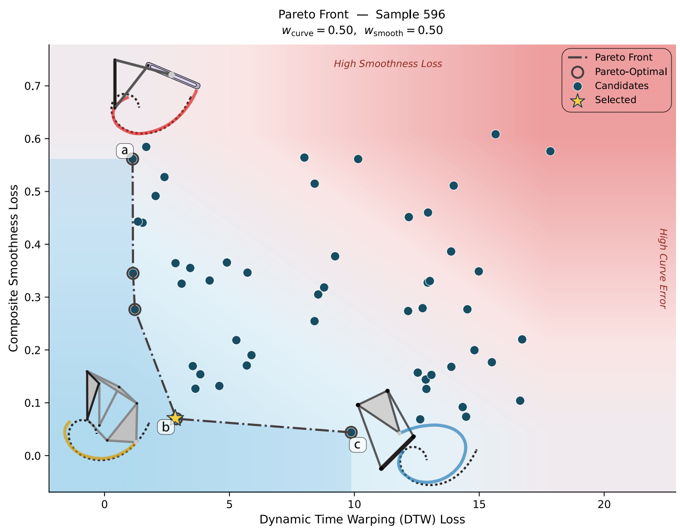

FIGURE 3: Objective space for 56 candidates generated from Sample 596. Each point represents one candidate mechanism; the dashed line traces the Pareto front. Three annotated candidates illustrate the structure of the trade-off: (a) achieves near-zero DTW but high smoothness loss, (b) lies near the Pareto elbow with low scores on both objectives, and (c), selected under the smooth preset minimizes smoothness loss at a substantial cost in path fidelity. The star marks the selected candidate.

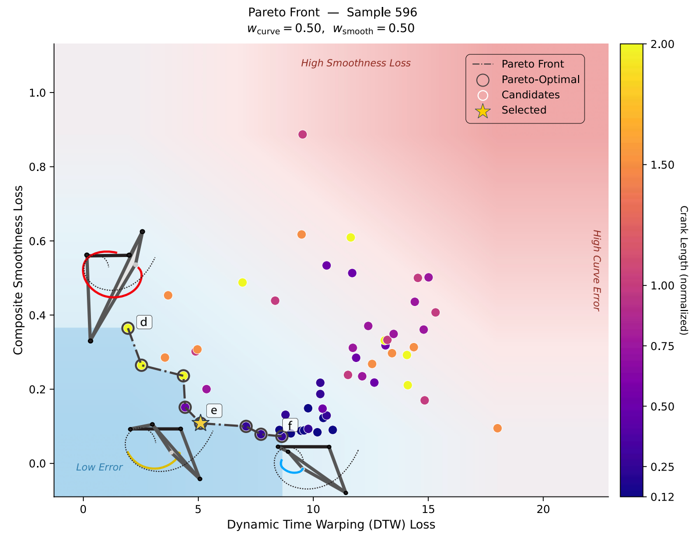

FIGURE 4: Objective space for 56 crank candidates generated from Sample 596. Color encodes the normalized crank radius $r \in \mathcal{R}$. Candidates d, e, and f mark selections under the curve, balanced, and smooth presets respectively.

the objective space: affine candidates obtain lower DTW values while crank candidates extend further along the smoothness axis at higher path cost. This separation confirms that the two strategies explore distinct regions of the design space, motivating their combination in the aggregate evaluation of Section 4.3.

4.2. Kinematic Motion Analysis

Figure 5 examines the kinematic behavior of candidates **a**, **b**, and **c** based on affine transformation sampling. The top row shows each mechanism's coupler trajectory with velocity arrows sampled at equal time intervals. Arrow color encodes normalized speed $s_t/\bar{s}$, so arrow spacing and color jointly indicate where the mechanism moves slowly (tightly spaced and dark arrows) or rapidly (widely spaced and bright arrows). The bottom row plots $s_t/\bar{s}$ versus the fraction of simulated arc using the same color scale, directly linking the spatial and temporal views.

Candidate **a**'s speed profile uses a separate vertical scale to accommodate its large excursion, whereas candidates **b** and **c** share a common scale for direct comparison. The horizontal axis represents normalized simulation progress independently rescaled to [0, 1] for each mechanism; the three profiles are therefore not time-synchronized.

Candidate **a** illustrates the kinematic cost of prioritizing path fidelity. Its trajectory progresses slowly through most of the arc, with tightly clustered, dark arrows, before accelerating sharply near the end of the cycle and reaching approximately 7× the mean speed. This behavior is consistent with its high composite smoothness loss ($\mathcal{L}_{\text{smooth}} \approx 0.57$) and would make the mechanism unsuitable for applications requiring consistent end-effector velocity. Candidates **b** and **c** maintain speed within roughly 20% of the mean throughout their arcs, with candidate **c** slightly flatter than **b**. This matches their lower smoothness losses ($\mathcal{L}_{\text{smooth}} \approx 0.07$ and $\mathcal{L}_{\text{smooth}} \approx 0.04$, respectively). The spread of each speed profile in the bottom row corresponds directly to the coefficient of variation of speed CV_s, one of the two components of $\mathcal{L}_{\text{smooth}}$.

Together with Fig. 3, these profiles make the Pareto trade-off physically concrete, showing that curve-focused selection yields a mechanism that closely traces the target path but with strongly non-uniform velocity, whereas smoothness-oriented selection produces near-constant speed at a measurable cost in path fidelity.

While these profiles confirm that the trade-off is physically meaningful for a single example, we now move on to exploring whether the framework produces consistent improvements across diverse target curves.

4.3. Aggregate Evaluation

To assess whether the selection framework generalizes beyond the single illustrative example of Section 4.1, we evaluate all four conditions - random baseline, curve, balanced, and smooth - across 500 randomly selected validation samples from the benchmark dataset. For each sample we generate candidates under three strategies: a 56-candidate affine pool, a 56-candidate crank-length pool, and a combined pool that merges both (112 candidates), totaling 56,000 candidate evaluations. The full evaluation completed in under one hour on a 64-core server. The random baseline draws one candidate uniformly at random from the relevant pool. Table 1 reports median DTW and median $\mathcal{L}_{\text{smooth}}$ for each strategy and selection condition. Within each strategy, the random condition establishes the performance floor for unguided sampling from that pool. As such, selection gain should be interpreted relative to each pool's own random draw.

The results present three main findings. First, weighted selection improves over random sampling within every strategy:

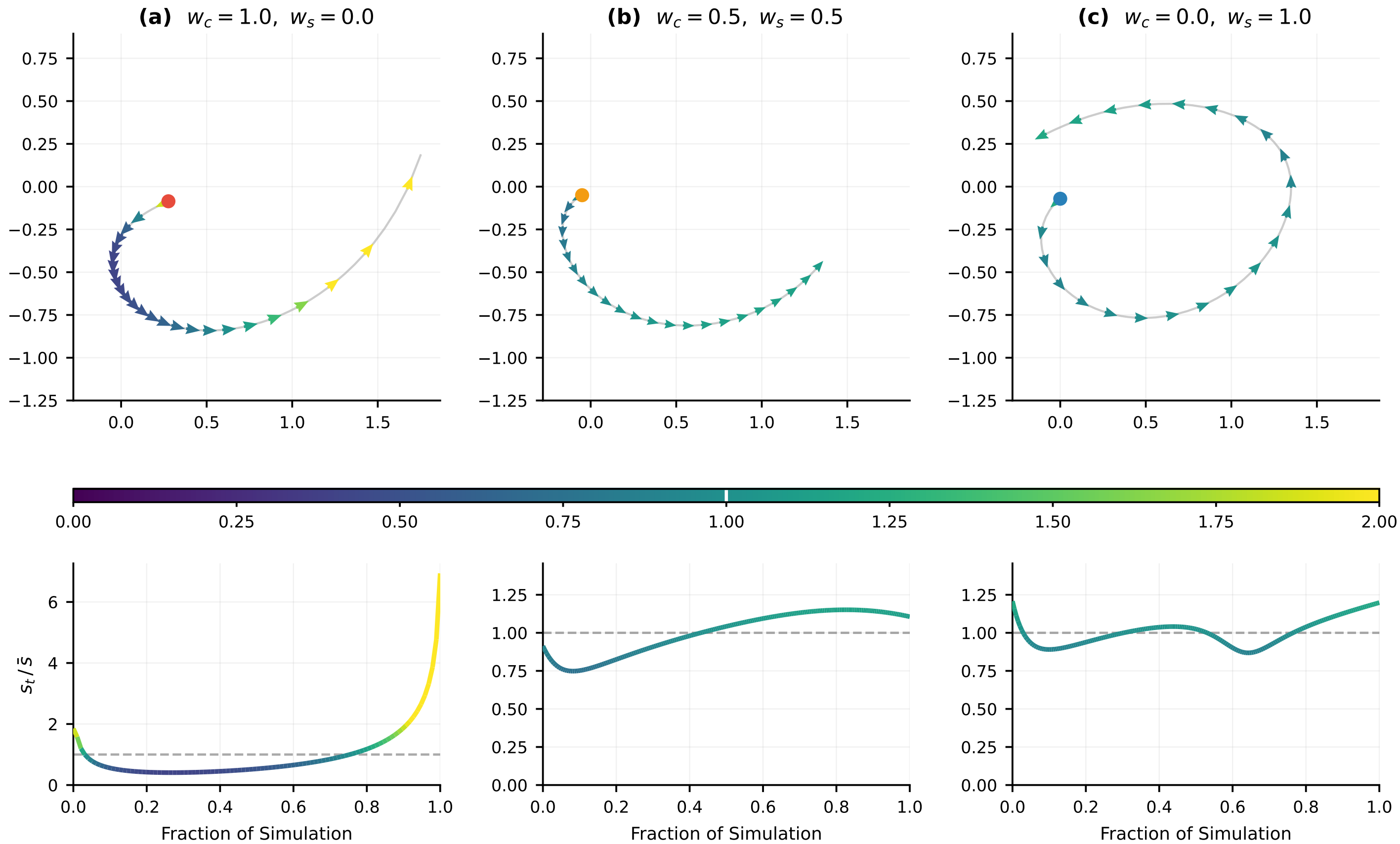

FIGURE 5: Kinematic profiles for candidates (a), (b), and (c) from Fig. 3. Top row: coupler trajectories with velocity arrows sampled at equal time intervals; arrow color encodes normalized speed. Bottom row: normalized speed versus fraction of full simulated arc. Candidate a exhibits a sharp acceleration spike, while candidates b and c maintain speed within roughly 20% of the mean throughout. Note that candidate (a) uses a separate vertical scale.

curve-optimized selection reduces median DTW by 82%, 86%, and 87% relative to the affine, crank, and combined random baselines respectively, confirming that the scoring and selection pipeline is effective regardless of how candidates are generated.

Second, crank-length sampling alone underperforms affine sampling on path fidelity across all presets (curve: 1.415 vs. 1.016 DTW), but produces a distinct distribution of candidates: the crank random baseline achieves lower median smoothness loss than affine random (0.249 vs. 0.311) at substantially higher DTW cost (10.009 vs. 5.562), consistent with the complementary objective-space coverage observed in Fig. 4.

Third, and most importantly, the combined pool demonstrates that merging the two strategies adds value beyond pool size alone. Despite starting from a worse random baseline on both metrics – reflecting the larger and more heterogeneous candidate set – combined curve selection achieves the best median DTW of any condition (0.961), improving on affine curve selection (1.016) by 0.05 units. The combined smooth preset achieves the lowest median smoothness loss across all conditions (0.026), a 91% reduction relative to its own random baseline. These results confirm that the affine and crank-length pools occupy complementary regions of the objective space, and that weighted selection can exploit this diversity to simultaneously improve both path fidelity and motion smoothness.

5. DISCUSSION

The central result of this work is that a pretrained generative synthesis model already encodes sufficient kinematic diversity to support attribute-conditioned design. Thus, the challenge is not generating diverse mechanisms, but making that diversity accessible and navigable. Affine sampling exploits the model's frame normalization to expose diversity through input-space transformations, while crank prefix conditioning exposes a complementary dimension of diversity through constrained autoregressive decoding. That two structurally different conditioning strategies produce candidates in different regions of the objective space suggests that MechaFormer has internalized a rich representation of mechanism space that can be probed from multiple directions without any modification to the underlying model. The emergence of candidates in distinct regions of the objective space under the two strategies suggests that the model has internalized structure in mechanism space beyond what is strictly necessary to optimize path fidelity. As generative synthesis methods mature, the key design challenge increasingly lies not in *generating good candidates*, but in *selecting among the candidates the model already knows how to produce*.

TABLE 1: Median DTW and smoothness loss ($\mathcal{L}_{\text{smooth}}$) with interquartile range across selection conditions (N=500 validation samples).

Strategy	Condition	Median DTW↓	IQR	Median $\mathcal{L}_{\text{smooth}}$↓	IQR
Affine	Random	5.562	[3.25, 10.37]	0.311	[0.212, 0.551]
	Curve	1.016	[0.52, 1.92]	0.290	[0.168, 0.469]
	Balanced	1.576	[0.85, 2.93]	0.089	[0.045, 0.156]
	Smooth	4.864	[2.65, 9.02]	0.030	[0.013, 0.066]
Crank	Random	10.009	[5.86, 18.66]	0.249	[0.170, 0.442]
	Curve	1.415	[0.73, 2.68]	0.248	[0.143, 0.401]
	Balanced	2.076	[1.12, 3.85]	0.104	[0.052, 0.182]
	Smooth	7.577	[4.12, 14.06]	0.042	[0.019, 0.092]
Combined	Random	7.642	[4.47, 14.25]	0.294	[0.200, 0.521]
	Curve	**0.961**	[0.49, 1.82]	0.283	[0.164, 0.458]
	Balanced	1.199	[0.65, 2.23]	0.088	[0.044, 0.154]
	Smooth	6.188	[3.37, 11.48]	**0.026**	[0.012, 0.057]

The smoothness objective introduced here instantiates the attribute-conditioned design paradigm in the kinematic synthesis domain. Prior work has demonstrated that qualitative attributes, such as comfort, expressiveness, and elegance, can serve as design objectives in product and robot design [6, 7, 9]. The present results extend this to mechanism motion quality, showing that $\mathcal{L}_{\text{smooth}}$ provides sufficient discriminative signal to navigate the Pareto front between path fidelity and kinematic regularity. The 91% reduction in median smoothness loss under the smooth preset, and the balanced preset's recovery of roughly one-third that gain while remaining within 0.24 DTW units of curve-optimal selection, suggest that the trade-off space is navigable rather than a binary choice between path accuracy and motion quality. This distinguishes the present approach from methods like Lin et al. [22], which require the designer to prescribe a target speed distribution as input and necessitate a purpose-trained architecture. Here, smoothness emerges as a post-generation selection criterion rather than a design specification, requiring no modification to the underlying model. Whether $\mathcal{L}_{\text{smooth}}$ aligns with human perceptual judgments of mechanism motion quality remains an open question. Prior work in biomechanics [13] and robotics [15] provides empirical grounding for its components, but direct validation against designer preferences in mechanism contexts has not been established.

Across all presets, crank sampling yields worse path fidelity than affine sampling, but it populates a different region of the objective space, both in the single-sample analysis and in the aggregate results. This indicates that sampling strategy is itself a design choice: affine sampling is better when path fidelity is the main priority, whereas crank sampling is more useful for exploring a broader range of kinematic behaviors within a fixed topology. Combining both produces the strongest overall results, achieving the best curve-optimal DTW (0.961) and the best smooth-preset smoothness loss (0.026), which suggests that the two strategies are complementary rather than redundant. More broadly, this points to the value of expanding the candidate pool with additional sampling strategies in future work.

The effectiveness of each strategy follows from the structure it exploits. Affine sampling works because MechaFormer operates on curves normalized to a canonical frame. Applying a rigid-body transformation before inference and inverting it afterward produces a kinematically distinct mechanism that is invisible to the model's learned representation, thus exploring a different region of the design space while preserving the geometric relationship between input and output. This is most effective when the model's output is sensitive to input placement, which is the common case for the nonlinear mapping from curve to mechanism, and least effective when multiple placements converge to the same mechanism family. Crank prefix conditioning works differently. Fixing the first predicted joint biases the decoder toward mechanism geometries that would otherwise receive low probability under unconstrained sampling, surfacing candidates the model can produce but would rarely generate on its own. This is most effective for topologies where the crank geometry strongly constrains the remaining linkage, such as four-bar mechanisms, and less effective for higher-order topologies where a single joint prefix leaves many degrees of freedom unconstrained.

The current framework represents a conservative instantiation of the prefix-conditioning strategy. The aggregate study allowed MechaFormer to select mechanism type freely, but the same prefix approach could in principle be applied across all 24 supported topology types. More directly, affine transformations and crank prefix conditioning could be combined by applying rotations, translations, and scaling to the input curve while simultaneously conditioning the decoder on a crank-pin position prefix, thus producing a substantially denser and more diverse candidate pool than either strategy alone. Beyond sampling coverage, two aspects of the selection pipeline should be addressed. Because min-max normalization is applied within each candidate pool, the weights (w_c, w_s) express relative importance given the variability present among the generated candidates rather than a fixed global trade-off, and should not be interpreted as absolute exchange rates between objectives. The jerk term $\hat{J}$ is additionally sensitive to trajectory sampling density, since it is computed from finite differences of simulator output; coarsely sampled trajectories may exhibit artificially elevated values. Finally, although the internal weighting α between CV_s and $\hat{J}$ is exposed as a tunable parameter, the default equal weighting has not been validated against human perceptual judgments across application domains and remains an open study for future evaluation.

All experiments in this study use MECHAFORMER as the underlying generative model. The selection framework itself is model-agnostic in that it operates on simulated trajectories rather than on the generator's architecture or learned representations, and any generator that produces candidate mechanisms with simulable coupler paths can serve as the upstream source. Affine sampling, which transforms the input curve before inference, is directly applicable to any model that accepts a coupler curve as input. Crank prefix conditioning is more tightly coupled to MECHAFORMER's autoregressive decoding and would require adaptation for architectures with different generation interfaces, such as VAE-based methods that lack a sequential prediction order. Validating the framework on a second generator remains future work.

6. CONCLUSION

This paper introduced an attribute-conditioned selection framework for planar mechanism synthesis that treats kinematic smoothness as a tunable design objective alongside geometric path fidelity. Given a target coupler curve, the framework generates a pool of candidate mechanisms via two complementary sampling strategies: affine input-space sampling and crank-length prefix conditioning. Each candidate is evaluated on a composite speed-invariant smoothness loss $\mathcal{L}_{\text{smooth}}$ and DTW distance, and the mechanism whose weighted score best reflects the designer's intent is selected. Because the full procedure requires no retraining or fine-tuning, the selection framework generalizes to any simulable property of the coupler trajectory.

Across 500 validation samples comprising 56,000 candidate evaluations, the two sampling strategies occupy complementary regions of the objective space, and their combination consistently outperforms either strategy in isolation. The combined smooth preset reduces median smoothness loss by 91% relative to the random baseline, whereas the combined balanced preset attains less than one-third the smoothness loss of curve-only selection while remaining within 0.24 DTW units of curve-optimal path fidelity. The fact that the combined pool achieves the best curve-optimal DTW (0.961) despite originating from a less favorable random baseline than the affine-only pool (7.642 vs. 5.562) indicates that the observed improvement arises from complementary coverage of the objective space rather than from pool size alone.

This work demonstrates that qualitative kinematic attributes can be incorporated into generative mechanism synthesis as post-generation selection criteria, and that inference-time conditioning strategies can surface latent diversity in pretrained models without any modification to the underlying architecture. As generative synthesis methods continue to mature, frameworks of this kind can serve as a practical bridge between application-specific design intent and the increasingly capable models behind them.

REFERENCES

[1] Nurizada, Anar, Lyu, Zhijie and Purwar, Anurag. "Path generative model based on conditional β-variational auto encoder for four-bar mechanism design." *Journal of Mechanisms and Robotics* Vol. 17 No. 6 (2025): p. 061004.

[2] Fogelson, Mitchell B, Tucker, Conrad and Cagan, Jonathan. "GCP-HOLO: Generating high-order linkage graphs for path synthesis." *Journal of Mechanical Design* Vol. 145 No. 7 (2023): p. 073303.

[3] Purwar, Anurag and Ge, Qiaode Jeffrey. "Advancements in Kinematic Synthesis: Data-Driven Methods for Mechanism Design." *Journal of Computing and Information Science in Engineering* Vol. 25 No. 12 (2025): p. 120810.

[4] Bolanos, Diana, Ataei, Mohammadmehdi and Jayaraman, Pradeep Kumar. "MechaFormer: Sequence Learning for Kinematic Mechanism Design Automation." *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 40. 24: pp. 19773–19780. 2026.

[5] Zhang, Zheng, Tavenard, Romain, Bailly, Adeline, Tang, Xiaotong, Tang, Ping and Corpetti, Thomas. "Dynamic time warping under limited warping path length." *Information Sciences* Vol. 393 (2017): pp. 91–107.

[6] Desai, Ruta, Anderson, Fraser, Matejka, Justin, Coros, Stelian, McCann, James, Fitzmaurice, George and Grossman, Tovi. "Geppetto: Enabling semantic design of expressive robot behaviors." *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*: pp. 1–14. 2019.

[7] Nandy, Ananya and Goucher-Lambert, Kosa. "Exploring the effectiveness of interactive preference learning for adapting designs to abstract semantic attributes." *Journal of Mechanical Design* Vol. 148 No. 4 (2026): p. 041702.

[8] Krippendorff, Klaus and Butter, Reinhart. "Product semantics: Exploring the symbolic qualities of form." *Innovation* Vol. 3 No. 40 (1984): pp. 4–9.

[9] Nandy, Ananya, Van, Monica, Li, Jonathan, Goucher-Lambert, Kosa, Klenk, Matthew and Hakimi, Shabnam. "Semantic properties of word prompts shape design outcomes: understanding the influence of semantic richness and similarity." *International Conference on-Design Computing and Cognition*: pp. 241–258. 2024. Springer.

[10] Bolanos, Diana, Ynchausti, Collin and Magleby, Spencer P. "A preliminary approach to select an origami source pattern for deployable space arrays." *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Vol. 86281: p. V007T07A070. 2022. American Society of Mechanical Engineers.

[11] Flash, Tamar and Hogan, Neville. "The coordination of arm movements: an experimentally confirmed mathematical model." *Journal of neuroscience* Vol. 5 No. 7 (1985): pp. 1688–1703.

[12] Hogan, Neville and Sternad, Dagmar. "Sensitivity of smoothness measures to movement duration, amplitude, and arrests." *Journal of motor behavior* Vol. 41 No. 6 (2009): pp. 529–534.

[13] Balasubramanian, Sivakumar, Melendez-Calderon, Alejandro, Roby-Brami, Agnes and Burdet, Etienne. "On the analysis of movement smoothness." *Journal of neuroengineering and rehabilitation* Vol. 12 No. 1 (2015): p. 112.

[14] Melendez-Calderon, Alejandro, Shirota, Camila and Balasubramanian, Sivakumar. "Estimating movement smoothness from inertial measurement units." *Frontiers in bioengineering and biotechnology* Vol. 8 (2021): p. 558771.

[15] Rohrer, Brandon, Fasoli, Susan, Krebs, Hermano Igo, Hughes, Richard, Volpe, Bruce, Frontera, Walter R, Stein,

Joel and Hogan, Neville. “Movement smoothness changes during stroke recovery.” *Journal of neuroscience* Vol. 22 No. 18 (2002): pp. 8297–8304.

[16] Moreton, Henry P and Séquin, Carlo H. “Functional optimization for fair surface design.” *Acm Siggraph Computer Graphics* Vol. 26 No. 2 (1992): pp. 167–176.

[17] Sandor, George N and Erdman, Arthur G. *Mechanism design: analysis and synthesis*. Vol. 1. Prentice-Hall New Delhi (1984).

[18] Norton, Robert L and Han, Jianyou. *Design of machinery*. Vol. 4. McGraw-Hill Science/Engineering/Math (2007).

[19] Freudenstein, Ferdinand. “An analytical approach to the design of four-link mechanisms.” *Transactions of the American Society of Mechanical Engineers* Vol. 76 No. 3 (1954): pp. 483–489.

[20] Nobari, Amin Heyrani, Srivastava, Akash, Gutfreund, Dan, Xu, Kai and Ahmed, Faez. “Link: Learning joint representations of design and performance spaces through contrastive learning for mechanism synthesis.” *arXiv preprint arXiv:2405.20592* (2024).

[21] Nurizada, Anar, Dhaipule, Rohit, Lyu, Zhijie and Purwar, Anurag. “A dataset of 3M single-DOF planar 4-, 6-, and 8-bar linkage mechanisms with open and closed coupler curves for machine learning-driven path synthesis.” *Journal of Mechanical Design* Vol. 147 No. 4 (2025): p. 041702.

[22] Lin, Jiong, Ning, Jialong, Goldfeder, Judah and Lipson, Hod. “Creative synthesis of kinematic mechanisms.” *arXiv preprint arXiv:2510.03308* (2025).