XLVII JORNADAS DE AUTOMÁTICA
CÓRDOBA 2026

Jornadas de Automática

Mollification for efficient curve smoothing with formal guarantees

Alfredo González-Calvin[a,*], Juan F. Jiménez[a]

[a]Dpto. de Arquitectura de Computadores y Automática, Universidad Complutense de Madrid, Plaza de las Ciencias, Madrid, España.

To cite this article: González-Calvin, Alfredo, Jiménez, Juan F. 2026. Mollification for efficient curve smoothing with formal guarantees. Jornadas de Automática, 47. https://doi.org/10.17979/ja-cea.2026.47.13652

Abstract

This work presents a new alternative to polynomial splines for trajectory and path generation. We present a method that, given a non-differentiable trajectory/path, generates an infinitely differentiable one in a single step—requiring small computational resources—and can be made as close to the original as desired, both pointwise and on compact subsets of the trajectory's domain. These methods—*mollification* and its improvement, *directional mollification*—possess $O(n)$ numerical complexity, can inherit certain geometric properties from splines (such as confinement to the convex hull of the points defining a curve), and allow for simple analytical curvature bounding for curves described as polygonal chains. We demonstrate the feasibility of these methods through a set of experiments in which the smooth curve adapts to the vehicle's dynamics.

Keywords: Path Planning, Trajectory tracking, Autonomous mobile robots, Computational methods, Guidance navigation and control

1. Introduction

In the fields of robotics and autonomous systems, motion execution is typically organized into a hierarchical structure of interrelated problems. Path planning first identifies a feasible route through the robot's configuration space—often as a sparse sequence of waypoints—using algorithms such as A* or sampling-based methods like RRT (LaValle, 2006). Subsequently, path generation transforms this abstract plan into a smooth, continuous geometric path or time-parameterized trajectory. This stage is critical because it acts as the bridge to the downstream control layer. In *path following* scenarios, the curve must be geometrically smooth (C^2, G^2 or higher) to ensure well-defined tangents and curvatures for spatial error computation (Yao et al., 2021). In *trajectory tracking*, the curve must additionally respect the robot's dynamic limits to remain physically executable (Siciliano et al., 2009).

Traditional path generation relies heavily on spline-based methods, including cubic B-splines (Lau et al., 2009; Berglund et al., 2010) and Bézier curves (Yang and Sukkarieh, 2010). While these provide local control and compact parametric descriptions (Piegl and Tiller, 2012; Farin, 2002), enforcing exact interpolation of waypoints while maintaining high smoothness often requires knot insertion or degree elevation (Faraway et al., 2007). Furthermore, these methods may produce cumbersome paths if waypoint spacing is irregular (Meek and Walton, 1992). Alternative approaches like Gaussian Process Regression (Rasmussen and Williams, 2006) and kernel smoothing (Hastie et al., 2009) offer C^∞ smoothness but sacrifice exact interpolation and incur $O(n^3)$ computational complexity. Recent advancements in Pythagorean-hodograph (PH) curves, provide elegant closed-form curvature and arc-length parameterization (Farouki et al., 2017; Bay et al., 2023), yet they often rely on rigid algebraic constraints that can limit flexibility across arbitrary data distributions.

In this paper, we explore the application of mollification—the convolution of a non-smooth function with a compactly supported C^∞ kernel (Evans, 2022)—as a computationally efficient and analytically rigorous alternative to traditional spline-based methods for path generation. While the mathematical foundations of mollifiers are well-established in the study of partial differential equations and signal processing (Hohage et al., 2024), their geometrical properties and utility in robotics remain unexploited. We show that mollifiers are uniquely suited for the inexpensive generation of trajectories

*Correspondence author: alfredgo@ucm.es

and paths, offering a robust balance between geometric flexibility and formal analytical guarantees, while being highly applicable to path following in mobile robotics, trajectory generation in CNC machining and computer-aided geometric design.

Specifically, we present two approaches to path smoothing. We first analyze the *conventional mollification*, or in short, mollification, which ensures that the generated curve remains within the convex hull of the original polygonal chain—a property highly desirable for safety-critical applications. However, to address the common requirement of exact waypoint navigation, we also present the *directional mollification* operator. This formulation preserves the C^∞ smoothness and numerical stability of conventional mollification while ensuring the resulting curve passes exactly through every original vertex (waypoint or knot) of the path. In Section 2, we establish the primary theoretical results of both conventional and directional mollifiers, explicitly relating their mathematical properties—such as uniform convergence and closed-form curvature bounds—to the requirements of path generation. In Section 3, we provide a series of experimental validations demonstrating the applicability and performance of these techniques in mobile robotics scenarios. Finally, we offer concluding remarks and directions in Section 4.

1.1. Notation

We say that a function $f = (f_1, \ldots, f_m) : \mathbb{R}^n \to \mathbb{R}^m$ is k-times continuously differentiable with $k \in \mathbb{N}$ if each $f_i : \mathbb{R}^n \to \mathbb{R}$ is k-times continuously differentiable. We denote the set of k-times continuously differentiable functions as $f \in C^k(\mathbb{R}^n, \mathbb{R}^m)$. If the derivative of all orders exists for each component, we say that f is smooth and denote it as $f \in C^\infty(\mathbb{R}^n, \mathbb{R}^m)$. If $f : \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R}^n \to \mathbb{R}$ are functions, we denote as $f * g : \mathbb{R}^n \to \mathbb{R}$ their convolution when it exists. By a parametric path or trajectory, we mean a function $f : [a, b] \to \mathbb{R}^n$ with $-\infty \leq a < b \leq \infty$ which is at least locally integrable (i.e., each component belongs to $L^1_{\mathrm{loc}}(\mathbb{R})$), and we denote its length (in the Euclidean norm) by $\mathcal{L}(f, a, b)$. Given a set A, we denote by $\mathrm{co}(A)$ its convex hull. By $\mathscr{D}(\mathbb{R}^n)$ we mean the set of C^∞ functions with compact support in $\mathbb{R}^n$.

2. Main results of mollification

2.1. Mollifiers

Mollifiers are C^∞ functions which have compact support and converge—via a parameter—to the Dirac delta distribution. The compact support property allows for an easy numerical integration and numerical convolution with other functions, since the domain of integration is bounded. Moreover, the Dirac delta distribution property, allows—via convolution—to approximate any (locally integrable) function, as close as desired, using a parameter. Let us present a mollifier formally.

Definition 1 (Mollifier). Let $\varphi \in \mathscr{D}(\mathbb{R}^n)$ and for $\varepsilon > 0$ define $\varphi_\varepsilon := \frac{1}{\varepsilon^n}\varphi \circ \frac{\mathrm{id}}{\varepsilon}$. We call φ a mollifier if it satisfies:

1. $\int_{\mathbb{R}^n} \varphi = 1$, and
2. in the distributional sense $\lim_{\varepsilon\to 0} \varphi_\varepsilon = \delta$, where δ is the Dirac delta distribution.

Example 1. Let $\varphi : \mathbb{R} \to [0, \infty)$ be the function

$$\varphi(x) = \begin{cases} c_1 \exp\left(\frac{-1}{1-x^2}\right), & |x| < 1, \\ 0, & |x| \geq 1 \end{cases}, \tag{1}$$

where $c_1 > 0$ is a normalization constant that ensures $\int_{\mathbb{R}} \varphi d\lambda = 1$. Clearly $\mathrm{supp}\,\varphi = \overline{(-1, 1)} = [-1, 1]$ and $\mathrm{supp}\,\varphi_\varepsilon = [-\varepsilon, \varepsilon]$.It is not difficult to show that the requirements of Definition 1 hold for this function, so indeed the set of mollifiers is nonempty.

We present the main properties of mollifiers, considered as approximate units.

Theorem 1 ((Evans, 2022, Appendix C, Theorem 7)). *Let $f \in L^p_{\mathrm{loc}}(\mathbb{R}^n, \mathbb{R})$ with $p \in [1, \infty]$, $\varphi \in \mathscr{D}(\mathbb{R}^n)$ be a mollifier, and $\varepsilon > 0$. The following three statements hold:*

1. *$f * \varphi_\varepsilon \in C^\infty(\mathbb{R}^n)$ and for any $n \in \mathbb{N}$ we have that $\partial^\alpha(\varphi_\varepsilon * f) = (\partial^\alpha\varphi_\varepsilon) * f$, for all multi index $\alpha \in \mathbb{N}^n_0$.*
2. *$\varphi_\varepsilon * f \to f$ pointwise almost everywhere as $\varepsilon \to 0^+$.*
3. *If f is continuous then $f * \varphi_\varepsilon \to f$ as $\varepsilon \to 0^+$ on compact subsets of $\mathbb{R}^n$.*

Note that, for trajectory/path generation, the previous results are of special interest. Indeed, while traditional methods using polynomial splines also ensure the second and third conditions of the theorem—usually by degree elevation—the C^∞ condition cannot be achieved by using these methods for it would require an infinite order polynomial spline (Schumaker, 2007), which is, in practice, not feasible. Nevertheless, the first statement ensures that a C^∞ function is obtained with just a single step while also being numerically easy to compute, because due to the compact support of the mollifier it exists a well-defined bounded domain of integration. Moreover, also note that computing the derivative of the mollified curve is straightforward because the mollifier is previously known, and it does not require the derivative of the original function to exist at any point. Finally note that, through the parameter $\varepsilon > 0$ the mollified curve can be made as close as desired to the original one, so it can fit the dynamics of the system in question—via a relationship with the curvature of the curve an the parameter $\varepsilon > 0$—while still resembling the original curve.

2.2. Geometrical properties and curvature guarantees

We sum up the results presented in González-Calvin et al. (2025) using a single theorem.

Theorem 2 (González-Calvin et al. (2025)). *Let $f \in L^p_{\mathrm{loc}}(\mathbb{R}^n, \mathbb{R})$ for some $p \in [1, \infty]$, $\varphi \in \mathscr{D}(\mathbb{R}^n, \mathbb{R})$ be a mollifier, and define $F_\varepsilon := f * \varphi_\varepsilon$.*

1. *If f is convex (resp. concave) and $\varphi \geq 0$ then $F_\varepsilon \geq f$ and F_ε is convex (resp. $F_\varepsilon \leq f$ and F_ε is concave) for any $\varepsilon > 0$. If f is quasiconvex and $\varphi \geq 0$ then F_ε is also quasiconvex, for all $\varepsilon > 0$.*
2. *If $f(x) = \langle a, x\rangle + c$ with $c \in \mathbb{R}$, $a \in \mathbb{R}^n$ and φ is an even function whose support is $\mathrm{supp}\,\varphi = \overline{B}(0, 1)$ then it holds that $F_\varepsilon = f$ for all $\varepsilon > 0$.*

3. *If $n = 1$ and f is monotone, then F_ε is monotone for all $\varepsilon > 0$.*

Moreover, if $f \in C([a,b], \mathbb{R}^n)$ with $-\infty < a < b < \infty$ and $\varphi \in \mathscr{D}(\mathbb{R})$ is a non negative mollifier whose support is $[-1,1]$ (as in Equation 1), and we define $F_\varepsilon := (f_i * \varphi_\varepsilon)_{i=1}^n$ we have that

1. *$\mathcal{L}(F_\varepsilon) \le \mathcal{L}(f)$ for all $\varepsilon > 0$, where $\mathcal{L} : C(\mathbb{R}, \mathbb{R}^n) \to \mathbb{R}$ is the length of the curve f, which has been extended as done in (González-Calvin et al., 2025, Lemma 7 and Theorem 8), computed with respect to any norm in $\mathbb{R}^n$.*
2. *$F_\varepsilon([a,b]) \subset \mathrm{co}(f([a,b]))$, for all $\varepsilon > 0$.*

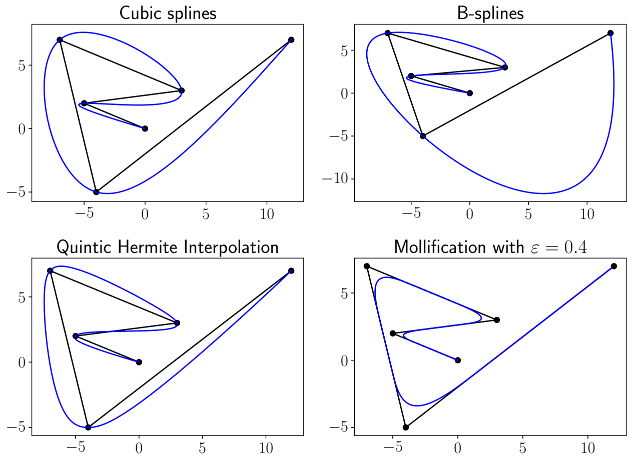

Figure 1: Comparison of different approaches for path generation. The blue line represents the path generated from the linearly interpolated sequence of points shown in black. The mollifier used is the one presented in Example 1 with $\varepsilon = 0.4$.

Let us break down carefully the main applications of the previous theorem for trajectory/path generation. Indeed, the second statement of Theorem 1 states that mollification is invariant to straight lines. That is, in the plane or in $\mathbb{R}^3$, geodesics are invariant under mollification. Therefore, if a curve is a polygonal chain (i.e., a curve created by joining consecutive points with straight lines), the mollification is a curve contained in the convex hull of the points that define the polygonal chain, and, except in neighborhoods—with length 2ε in the parameter—of the non-differentiable points, it is equal to the original curve. See Figure 1 for a comparison between the mollification of a polygonal chain and polynomial spline methods. The computational complexity of the mollification, should be below the one required by polynomial splines under the same conditions; that is, C^∞, curvature bounds and waypoint intersection. Indeed, if we degrade the conditions of the curve continuity and do not pay attention to the curvature upper bounds, it could be possible to find polynomial splines that are more computationally efficient.

Moreover, while the first statement provides a geometrical invariance under mollification, note also that the length of a mollified path is shorter than the length of the original curve. Finally, the monotone invariance statement ensures that the Gibbs phenomenon does not occur in mollification. In addition, it can be seen that there are zones in which the original and mollified curves coincide for a polygonal chain. For that, we need the actual definition of a polygonal chain.

Definition 2. Let $p \in \mathbb{N}_0$ with $p \ge 2$ and let $(P_0, \dots, P_p)_{i=0}^p \subset \mathbb{R}^n \times \cdots \times \mathbb{R}^n$ be a set of ordered points called knots or vertices. We define the $p+1$ knots/vertices polygonal chain in $\mathbb{R}^n$ to be the function $f : [0,p] \to \mathbb{R}^n$, expressed as

$$f(t) := P_{r-1} + (P_r - P_{r-1})(t - (r-1)), \tag{2}$$

where $t \in [r-1, r]$ and $r \in \{1, \dots, p\}$. To be able to convolve the previous function, we define its extension $\bar{f}$ as $\bar{f} : \mathbb{R} \to \mathbb{R}^n$,

$$\bar{f}(t) = \begin{cases} P_0 + (P_1 - P_0)t & t \le 1 \\ f(t) & t \in [1,p] \\ P_{p-1} + (P_p - P_{p-1})(t - (p-1)) & t \ge p \end{cases}, \tag{3}$$

for any $t \in \mathbb{R}$. Note that this function is continuous, hence locally integrable, and differentiable almost everywhere with a locally integrable derivative.

We can now present the regions in which the original and mollified curves coincide (González-Calvin et al., 2026, Theorem 6).

Theorem 3. *Let f be a polygonal chain as in Definition 2 which has already been extended (that is $f = \bar{f}$ in Definition 2). Let $\varphi \in \mathscr{D}(\mathbb{R})$ be an even, non negative mollifier with support $[-1,1]$. Suppose $\frac{1}{2} - \varepsilon > 0$, let $r \in \{1, \dots, p\}$ and define $V_r := [r-1+\varepsilon, r-\varepsilon]$. Then it holds that $f|_{V_r} = F_\varepsilon|_{V_r}$.*

Thus, the previous proposition justifies the bottom right picture of Figure 1, as it can be seen in the zones where the mollification of the curve is equal to the original polygonal chain.

In addition, an explicit, easy-to-compute upper bound on the curvature for a three-point polygonal chain can be found in (González-Calvin et al., 2025, Equation 8). If we define $\tilde{P}_1 := P_1 - P_0$ and $\tilde{P}_2 := P_2 - P_1$, the upper bound on the curvature can be written as

$$\frac{1}{\varepsilon} \|\varphi\|_\infty \|\tilde{P}_2 \wedge \tilde{P}_1\|_2 M(\tilde{P}_1, \tilde{P}_2), \tag{4}$$

where

$$M(\tilde{P}_1, \tilde{P}_2) := \begin{cases} \frac{1}{\|\tilde{P}_1 \bar{s} + (1-\bar{s})\tilde{P}_2\|_2^3}, & 0 \le \bar{s} \le 1 \\ \max\left\{ \frac{1}{\|\tilde{P}_1\|_2^3}, \frac{1}{\|\tilde{P}_2\|_2^3} \right\}, & \text{otherwise} \end{cases}.$$

It can be thought that this upper bound is limiting, but is it not as long as $\varepsilon \in (0, \frac{1}{2})$. Indeed, if that is the case, using Theorem 3, the mollified curve coincides with the original curve at least in neighboring sets in the middle of each segment (cf. Figure 1). Therefore, the computation on the upper bound on the curvature can be done pairwise, that is, for each pair of segments.

Nevertheless, while the mollified curve can be made arbitrary close to the original polygonal chain via Theorem 1 (letting $\varepsilon \to 0$), as can be seen from the bottom right picture in Figure 1, the mollified curve does not intersect the points that define the trajectory. This is a consequence of the convex hull confinement of the mollified curve, as stated in Theorem 1. For this purpose, that is, to create a curve with the same convergence and smoothness properties of mollifiers that intersect the points of a polygonal chain, we introduce the directional mollification operator.

2.3. *Intersecting points of a polygonal chain: The directional mollification operator*

To solve the waypoint intersection problem, while at the same time keeping the smoothness, convergence, and ease of computation of mollification, we introduce the *directional mollification* operator. To do so, we need to present first the definition of the generalized directional derivative, see (Jahn, 2007) for a complete treatment of these types of derivatives.

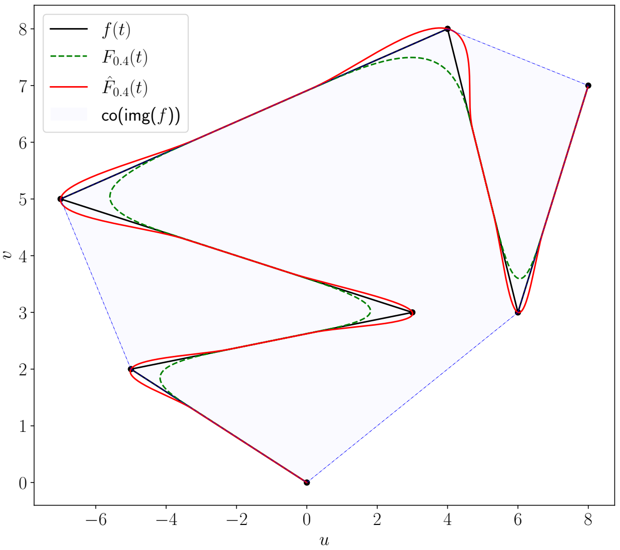

Figure 2: Comparison of mollification techniques for a polygonal chain (black): the conventional approach (green) ensures convex hull confinement (shaded blue) but does not intersect vertices. In contrast, the directional mollification (red) maintains C^∞ smoothness while achieving exact vertex intersection and providing explicit curvature estimates. The value 0.4 stand for the parameter ε used both in the conventional and directional mollification.

Definition 3. Let $f : \mathbb{R}^n \to \mathbb{R}^m$ be a function. Given $x, h \in \mathbb{R}^n$ the directional derivative of f at x in the direction of h is defined as

$$f^\circ(x)(h) := \lim_{t \to 0^+} \frac{f(x + th) - f(x)}{t},$$

provided the limit exists. If the limit exists for all $h \in \mathbb{R}^n$, then f is said to be directionally differentiable at x.

We can now present the definition of the *directional mollification*, which consists of the sum of the previous mollification, called from now on, *conventional mollification*, and a new term, which we call the *directional derivative term*.

Definition 4 (Directional mollification). Let $p \in [1, \infty]$, $f \in L^p_{\text{loc}}(\mathbb{R}^n, \mathbb{R})$ and $\varphi \in \mathscr{D}(\mathbb{R}^n)$ be a mollifier. Let f be directionally differentiable at any point, and for any $x \in \mathbb{R}^n$ let $f^\circ_x \in L^q_{\text{loc}}(\mathbb{R}^n, \mathbb{R})$ for some $q \in [1, \infty]$. For $\varepsilon > 0$, we define the directional mollification of f, $\widehat{F}_\varepsilon : \mathbb{R}^n \to \mathbb{R}$ as $\widehat{F}_\varepsilon(x) := F_\varepsilon(x) + D_\varepsilon(x)$, i.e.,

$$\widehat{F}_\varepsilon(x) := \underbrace{(f * \varphi_\varepsilon)(x)}_{F_\varepsilon(x)} + \underbrace{\int_{\mathbb{R}^n} f^\circ(s)(x - s)\varphi_\varepsilon(x - s)ds}_{D_\varepsilon(x)},$$

which is the sum of the mollification and D_ε which we call the directional derivative term.

We can now state the main properties of directional mollification. These can be found in (González-Calvin et al., 2026).

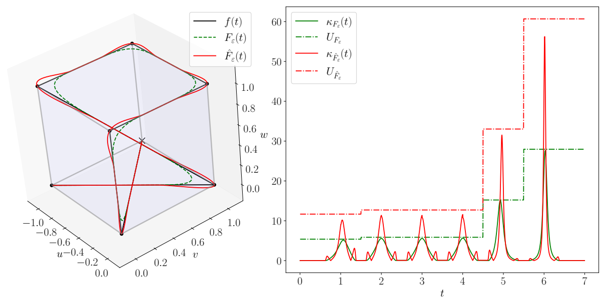

Figure 3: Representation of an eight-point polygonal chain in $\mathbb{R}^3$, and the curvature and curvature bounds of the conventional and directional mollifications.

Theorem 4. *Let $f : \mathbb{R}^n \to \mathbb{R}$ satisfy the assumptions of Definition 4, $\varphi \in \mathscr{D}(\mathbb{R}^n)$ be any non negative mollifier whose support is $\overline{B}(0, 1)$, and let $\varepsilon > 0$. For the directional mollification of f, $\widehat{F}_\varepsilon : \mathbb{R}^n \to \mathbb{R}$, the following statements are true.*

1. *If f is differentiable almost everywhere, (that is, ∇f exists almost everywhere on $\mathbb{R}^n$), and $\partial_i f \in L^{q_i}_{\text{loc}}(\mathbb{R}^n)$ for some $q_i \in [1, \infty]$ for all $i \in \{1, \ldots, n\}$, then $D_\varepsilon \in C^\infty(\mathbb{R}^n)$ and $D_\varepsilon = \sum_{i=1}^n \partial_i f * \pi_i \varphi_\varepsilon$ where $x = (x_1, \ldots, x_n) \in \mathbb{R}^n \mapsto \pi_i(x) = x_i$ is the projection of the i-th component of a point in $\mathbb{R}^n$. Moreover, for any multi-index $\alpha \in \mathbb{N}^n_0$, $\partial^\alpha D_\varepsilon = \sum_{i=1}^n \partial_i f * \partial^\alpha(\pi_i \varphi_\varepsilon)$, where $\partial^\alpha(\pi_i \varphi_\varepsilon) = \sum_{\beta \le \alpha} \partial^\beta \pi_i \partial^{\alpha - \beta} \varphi_\varepsilon$.*

2. *If f is locally Lipschitz, $\widehat{F}_\varepsilon \to f$ as $\varepsilon \to 0^+$ pointwise. If, in addition, f is Lipschitz, the convergence is uniform in compact subsets of $\mathbb{R}^n$. Moreover, if for each $x \in \mathbb{R}^n$, $\sup_{u \in \bar{B}(0,1)} |f^\circ(x - u)(u)| = M_x < \infty$ then $\lim_{\varepsilon \to 0^+} \widehat{F}_\varepsilon = f$ pointwise. Finally, if $\sup_{x \in \mathbb{R}^n} M_x < \infty$ then $\widehat{F}_\varepsilon \to f$ as $\varepsilon \to 0^+$ in compact subsets of $\mathbb{R}^n$.*

3. *If f is convex then $\widehat{F}_\varepsilon \le f$, and $D_\varepsilon \le 0$.*

4. *If $f(x) = \langle a, x \rangle + c$ with $c \in \mathbb{R}$, $a \in \mathbb{R}^n$, and φ is an even function, then it holds that $\widehat{F}_\varepsilon = f$ for all $\varepsilon > 0$.*

As can be seen, the main properties of mollification, that is, smoothness, convergence, and ease of computation, are kept invariant by the directional mollification, with the exception of asking f to be Lipschitz continuous to have convergence on compact subsets. Nevertheless, note that this is always true for polygonal chains. Moreover, note that straight lines are again invariant under this new operator. Thus, returning to the polygonal chain curve, we can finally state the main results.

Theorem 5. *Let f be a polygonal chain as in Definition 2 which has already been extended (that is $f = \bar{f}$ in Definition 2). Let $\varphi \in \mathscr{D}(\mathbb{R})$ be an even, non negative mollifier with support $[-1, 1]$. The following statements hold.*

- *If $\varepsilon \in (0,1)$ it holds that $\widehat{F}_\varepsilon(k) = f(k)$ for $k \in \{0,2,\ldots,p\}$. Moreover, if $\frac{1}{2}-\varepsilon > 0$, let $r \in \{1,\ldots,p\}$ and define $V_r := [r-1+\varepsilon, r-\varepsilon]$. Then it holds that $f|_{V_r} = F_\varepsilon|_{V_r} = \widehat{F}_\varepsilon|_{V_r}$.*

- *Let the polygonal chain consist of three points. Then an upper bound on the curvature κ, for the mollifier of Equation (1) can be found to be*

$$|\kappa(t)| \leq \frac{2}{\varepsilon}\varphi(0)\frac{\|\tilde{P}_2 \wedge \tilde{P}_1\|_2}{\|\bar{s}\tilde{P}_1 + (1-\bar{s})\tilde{P}_2\|_2^3}, \quad \forall t \in \mathbb{R},$$

where $\bar{s} = \frac{\langle \tilde{P}_2-\tilde{P}_1, \tilde{P}_2\rangle}{\|\tilde{P}_2-\tilde{P}_1\|_2^2}$ and $\tilde{P}_1 := P_1 - P_0$ and $\tilde{P}_2 := P_2 - P_1$. That is, twice the conventional mollification.

See Figures 2 and 3 for a representation of the directional mollification of a polygonal chain in two and three dimensions, respectively, and its comparison with the conventional mollification. Since both methods create the same curve on the same neighboring sets (cf. Theorems 3 and 5) a switch between any of the two curves can be made without any discontinuity. This allows for a versatile smooth curve generator, which can be chosen for each pair of segments to be confined in the convex hull, or intersecting the waypoint, and in both cases with smoothness and convergence guarantees. Moreover, the previous Theorem and Equation (4) provide curvature bounds for each type of mollification, and its comparisons can be found in Figure 3. The left picture represents in black the polygonal chain curve f confined in a unit volume cube in $\mathbb{R}^3$, shown in blue. It also represents its conventional F_ε and directional mollifications $\widehat{F}_\varepsilon$ in dashed green and solid red, respectively. In all mollifications, the used mollifier is the one presented in Equation (1), with $\varepsilon = 0.4$. Right picture represents in green the curvature of the conventional mollified curve F_ε, and as a dashed green plot, the curvature upper bounds κ_{F_ε} for each pair of segment as follows. In the set $[0, 1.5]$ the curvature bound is computed using Equation (4) for the polygonal chain consisting of the points P_0, P_1 and P_2. In the set $[k+0.5, k+1.5]$ the curvature bound is computed using Equation (4) for the polygonal chain consisting of the points P_k, P_{k+1} and P_{k+2} for $k \in \{1,\ldots,p-1\}$. The solid and red dashed plots represent, respectively, the curvature of $\widehat{F}_\varepsilon$, and its curvature upper bounds $\kappa_{\widehat{F}_\varepsilon}$ computed in the same manner as for the conventional mollification, but using the formula presented in Theorem 5.

3. Applications in mobile robotics

Our experimental platform, shown in Figure 4, is a rover modelled as a unicycle, built around a Matek F765-Wing autopilot with an STM32 microcontroller, integrated IMUs, and support for GNSS, compass, and radio receivers—specifically a Matek M10Q-5883, a Futaba 7008SB receiver, and a Zigbee Xbee telemetry radio. The entire system runs on the open-source Paparazzi UAV framework (Hattenberger et al., 2014), which handles autonomous operation, telemetry, and real-time communication with the Ground Control Station (GCS). Through the GCS, the user can issue high-level commands and adjust waypoints on the fly, while the onboard microcontroller recomputes the mollified path in real-time whenever a point or parameter is modified, as illustrated in Figure 5.

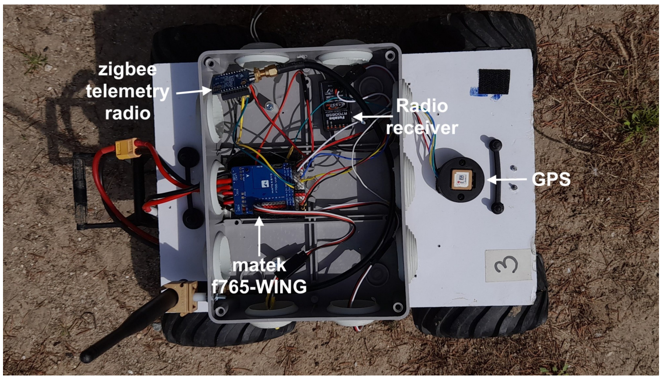

Figure 4: Rover vehicle and hardware used during the experiments.

3.0.1. Experimental data

We show our logs in Figure 6, where we created a polygonal chain in $\mathbb{R}^2$. As noted above, any Paparazzi user can create such trajectories by simply moving points in the ground control station before or during the experiment in real time. The experiment uses a curve similar to that in Figure 5. 5.

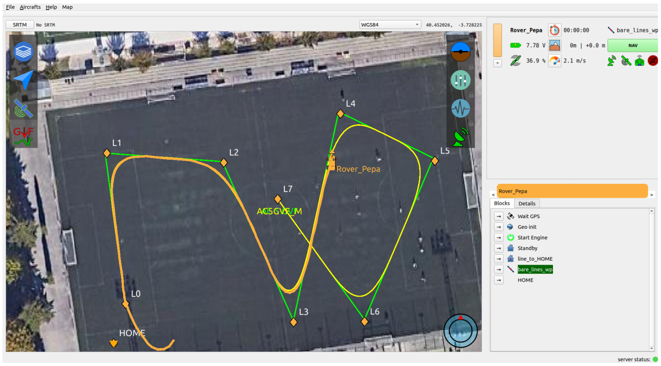

Figure 5: Capture of an experiment using Paparazzi GCS. The original trajectory f is shown in green, created by linearly interpolating the points $L_i \in \mathbb{R}^2$, $i \in \{0,\ldots,7\}$. The yellow curve represents the mollification of the original trajectory where φ is as in Example 1 and $\varepsilon = 0.5$. The orange line represents the trajectory described by the vehicle.

The experimental objective is as follows: Given a desired non-differentiable path f, mollify it with parameter $\varepsilon > 0$ to obtain $F_\varepsilon = (f * \varphi_\varepsilon)$, where ε is a function of vehicle speed limited by maximum allowed curvature. To demonstrate potential applications, we use a linear relationship between maximum allowed curvature and speed. For speed $v > 0$, with R_{min}, R_{max}, and v_{max} denoting minimum radius, maximum radius, and maximum speed, respectively, the allowed radius of curvature is $R(v) := R_{min} + \frac{v}{v_{max}}(R_{max} - R_{min})$. Note this is merely illustrative; curvature and speed need not be linearly related in practice. We are going to consider a polygonal chain curve, as shown in Figure 6.

The adaptation of the curve to vehicle dynamics operates as follows: at the initial time t_0, we set $\varepsilon_1 = 0.5$ to generate the first mollified curve $F_{\varepsilon_1} = f * \varphi_{\varepsilon_1}$, where f is the original curve and φ is as in Example 1. As the vehicle advances and reaches each segment midpoint at times $\{t_i\}_{i=1}^6$, the speed is measured and used to compute the maximum allowed curvature, from which the minimum permissible ε_{i+1} is determined via (4). This generates a family of six parameters $\{\varepsilon_i\}_{i=1}^6$ and corresponding mollified curves $\{F_{\varepsilon_i}\}_{i=1}^6$ that dynamically adapt to vehicle constraints. As shown in Figure 6, the resulting paths remain close to the original trajectory where curvature permits, e.g., segments four and five, while strongly constraining the curve when necessary, e.g., the last two segments. These experiments validate the practical viability of our approach on real, affordable hardware.

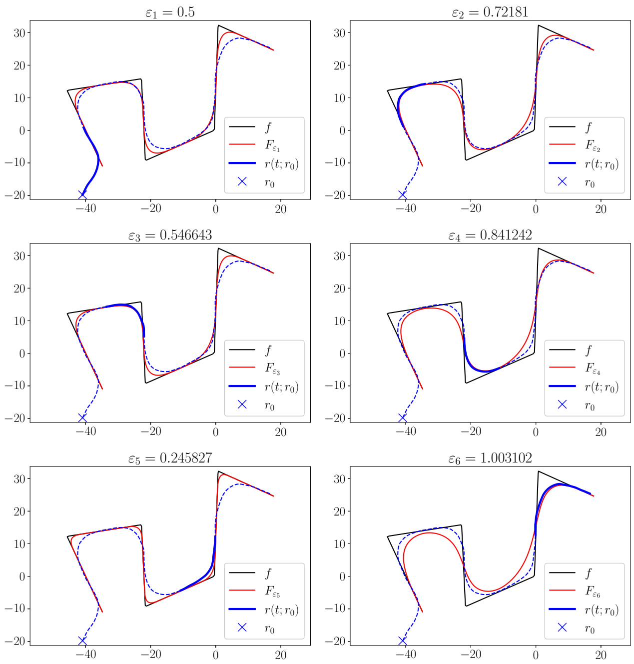

Figure 6: Real experiment of path following of a mollified non-differentiable path with a rover vehicle. The original non-differentiable path f is in black, the mollified family trajectories $\{F_{\varepsilon_i}\}_{i=1}^6$ to be followed at each *stage* are in solid red, its initial point r_0 is denoted as the blue cross, the position of the vehicle for its corresponding $\varepsilon_i > 0$ at times $t \in [t_{i-1}, t_i]$ in a thick solid blue line. Finally, the complete trajectory of the vehicle $r(t; r_0)$ throughout all the stages is the dashed blue line.

4. Conclusions

This work demonstrates that mollification operators offer a mathematically rigorous and computationally efficient alternative to polynomial splines for robotic path generation. By providing C^∞ smoothness, closed-form curvature bounds, and waypoint preservation through the directional variant, these techniques enable real-time, dynamic trajectory adaptation on resource-constrained hardware, as the experimental results validate. Experimental results validate the practical viability of the framework, establishing a robust bridge between high-level planning and low-level control.

Acknowledgments

The research leading to these results was supported by the FPU program of the Ministry of science, innovation and universities of Spain, and by iRoboCity2030-CM, Robótica Inteligente para Ciudades Sostenibles (TEC-2024/TEC-62), funded by the Programas de Actividades I+D en Tecnologías en la Comunidad de Madrid.

References

Bay, T., Cattiaux-Huillard, I., Romani, L., Saini, L., 2023. On g1 and g2 hermite interpolation by spatial algebraic-trigonometric pythagorean hodograph curves with polynomial parametric speed. Applied Mathematics and Computation 458, 128240.
URL: `https://www.sciencedirect.com/science/article/pii/S0096300323004095`
DOI: `https://doi.org/10.1016/j.amc.2023.128240`

Berglund, T., Brodnik, A., Jonsson, H., Staffanson, M., Söderkvist, I., 2010. Planning smooth and obstacle-avoiding B-spline paths for autonomous mining vehicles. IEEE Transactions on Automation Science and Engineering 7 (1), 167–172.

Evans, L. C., 2022. Partial differential equations. Vol. 19. American mathematical society.

Faraway, J. J., Reed, M. P., Wang, J., 09 2007. Modelling Three-Dimensional Trajectories by Using Bézier Curves with Application to Hand Motion. Journal of the Royal Statistical Society Series C: Applied Statistics 56 (5), 571–585.
URL: `https://doi.org/10.1111/j.1467-9876.2007.00592.x`
DOI: `10.1111/j.1467-9876.2007.00592.x`

Farin, G. E., 2002. Curves and surfaces for CAGD: a practical guide. Morgan Kaufmann.

Farouki, R. T., Gentili, G., Giannelli, C., Sestini, A., Stoppato, C., 2017. A comprehensive characterization of the set of polynomial curves with rational rotation-minimizing frames. Advances in Computational Mathematics 43 (1), 1–24.

González-Calvin, A., Jiménez, J. F., de Marina, H. G., 2025. Efficient generation of smooth paths with curvature guarantees by mollification.
URL: `https://arxiv.org/abs/2512.13183`

González-Calvin, A., Jiménez, J. F., de Marina, H. G., 2026. Directional mollification for knot-preserving c^∞ smoothing of polygonal chains with explicit curvature bounds.
URL: `https://arxiv.org/abs/2603.21831`

Hastie, T., Tibshirani, R., Friedman, J., 2009. The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd Edition. Springer, New York.

Hattenberger, G., Bronz, M., Gorraz, M., 2014. Using the paparazzi uav system for scientific research. In: IMAV 2014, international micro air vehicle conference and competition 2014. pp. pp–247.

Hohage, T., Maréchal, P., Simar, L., Vanhems, A., 2024. A mollifier approach to the deconvolution of probability densities. Econometric Theory 40 (2), 320–359.

Jahn, J., 2007. Introduction to the theory of nonlinear optimization. Springer.

Lau, B., Sprunk, C., Burgard, W., 2009. Kinodynamic motion planning for mobile robots using splines. In: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, pp. 2427–2433.

LaValle, S. M., 2006. Planning Algorithms. Cambridge University Press, Cambridge, UK.
URL: `http://planning.cs.uiuc.edu/`

Meek, D. S., Walton, D. J., 1992. Approximation of discrete data by G1 arc splines. Computer-Aided Design 24 (6), 301–306.

Piegl, L., Tiller, W., 2012. The NURBS book. Springer Science & Business Media.

Rasmussen, C. E., Williams, C. K. I., 2006. Gaussian Processes for Machine Learning. MIT Press, Cambridge, MA.

Schumaker, L., 2007. Spline functions: basic theory. Cambridge university press.

Siciliano, B., Sciavicco, L., Villani, L., Oriolo, G., 2009. Robotics: modelling, planning and control. Springer.

Yang, K., Sukkarieh, S., 2010. An analytical continuous-curvature path-smoothing algorithm. IEEE Transactions on Robotics 26 (3), 561–568.

Yao, W., de Marina, H. G., Lin, B., Cao, M., 2021. Singularity-free guiding vector field for robot navigation. IEEE Transactions on Robotics 37 (4), 1206–1221.
DOI: `10.1109/TRO.2020.3043690`