Numerical Reparametrization of Rational Parametric Plane Curves

Sonia Pérez-Díaz\textsuperscript{a}, Li-Yong Shen\textsuperscript{b,*}

\textsuperscript{a}Dpto. de Física y Matemáticas, Universidad de Alcalá, E-28871 Madrid, Spain
\textsuperscript{b}School of Mathematical Sciences, University of CAS, Beijing, China

Abstract
Proper reparametrization is a basic simplifying process for rational parameterized curves. There are complete results proposed for the curves with exact coefficients but few papers discuss the situations with numerical coefficients which have many practical applications. In this paper, we deal with mathematical objects that are assumed to be given approximately. The approximate improper index is firstly defined. Then, we provide some important properties concerning the approximate improper index and the numerical reparametrization. Finally, we propose the numerical reparametrization algorithm for rational parametric plane curves, and the error bound is carefully discussed.

Keywords: Rational Curve, Approximate Improper, Proper Reparametrization

1. Introduction
Rational parametric curves and surfaces usually come from the original designing of engineering, geometric modeling and computer aided design (CAD). A natural question arises on how to simplify the parametrization of curves and surfaces, by which we mean finding rational functions with degrees as small as possible. There are several motivations for this simplification (see e.g. \[20\]). First, parametrizations of smaller degrees can be represented with less data; second, implicitization is easier when the degree is smaller; at last, it is easier to find rational curves of smaller degrees on the given surface.

In the simplification, the proper reparametrization plays an important role if the given parametrization is improper. Hence, the study of proper reparametrization has been concerned by some authors such as \[4, 13, 18, 22, 24\]. In particular, for rational parametric curves, the problem is well studied. Lüroth’s theorem (see \[28\]) has shown that there always exist proper reparametrizations constructively. Several efficient proper reparametrization algorithms based on Lüroth’s theorem can be found in \[10, 21\]. Here, the problem was discussed in symbolic consideration, that is, with exact coefficients as rational numbers.

*Corresponding author

Email addresses: sonia.perez@uah.es (Sonia Pérez-Díaz), shenly@amss.ac.cn (Li-Yong Shen)

Preprint submitted to Elsevier May 14, 2013
Nevertheless, in many practical applications, for instance in the frame of computer aided geometric design, these approaches tend to be insufficient, since in practice most of data objects are given or become approximate. As a consequence of this phenomenon, there has been an increasing interest for the development of hybrid symbolic-numeric algorithms and approximate algorithms. Approximate algorithms have been developed for some applied numerical topics, such as, approximate parametrization of algebraic curves and surfaces \[14, 15, 17\], approximate greatest common divisor (gcd) \[2, 3, 6, 11, 12, 30\], finding zeros of multivariate systems \[6\] and factoring polynomials \[5, 9\].

For a given improper rational parametrization, the perturbed one is proper in exact consideration but it could be nearly improper. This state of affairs is unexpected in real applications, but few papers discussed the problem of properly reparametrizing a given parametric curve with perturbed float coefficients. This fact motivates us to consider the problem in numerical way. As we know, only a heuristic algorithm was proposed to properly reparametrize a perturbed improper parametric curve (see \[21\]). However, no step was given to detect whether a numerical curve is improper within a tolerance. And there was no error analysis. In symbolic considerations, the tracing index is used to determine the properness of a parametrization of a plane algebraic curve (see \[22, 27\]). Essentially, it is the cardinality of a generic fibre of the parametrization. From the geometric point of view, the tracing index measures the number of times that a parametrization traces a curve over the algebraic closure of the ground field. In this paper, we extend the concept to the numerical situations that is, the approximate improper index is expected to be the number of parameter value mapped in a neighborhood to a generic point of a given plane curve. This gives the theoretical foundation for our further discussion.

In these conditions, we review the symbolic algorithm of reparametrization for algebraic plane curves presented in \[18\], and generalize it for the numerical case. For this purpose, we first define the approximate improper index and afterwards, we introduce the notion of the equivalence of two numerical rational parametric curves. The followed structure is similar to the symbolic situation, but the discussions are quite different. Some important properties are generalized to the numerical situation. Moreover, as the necessary work for the numerical discussion, the relation between the reparameterized and the original curve is subtly analyzed. As the error control, the approximate reparameterized curve obtained is restricted in the offset region of the original one (and reciprocally). In the studies and computations, we use the computation of approximate gcd and univariate resultants.

The paper is organized as follows. At first, the symbolic algorithm of proper reparameterization presented in \[18\] is briefly reviewed (see Section 2). In Section 3, the definition of approximate improper index ($\epsilon$index) and $\epsilon$-numerical reparametrization are proposed. In addition, we construct the $\epsilon$-numerical reparametrization, and we prove that it is $\epsilon$-proper. Furthermore, we discuss the relation between the reparameterized curve and the input one. In Section 4, the numerical algorithm is given as well as some examples. Finally, we conclude with Section 5, where we propose topics for further study.
2. Symbolic Algorithm of Reparametrization for Curves

Before describing the method for the approximate case, and for reasons of completeness, in this section we briefly review some notions and the algorithmic approach to symbolically reparametrize curves presented in [18]. Let $\mathbb{C}$ be the field of the complex numbers, and $\mathcal{C}$ a rational plane algebraic curve over $\mathbb{C}$. A parametrization $\mathcal{P}$ of $\mathcal{C}$ is proper if and only if the map

$$\mathcal{P} : \mathbb{C} \rightarrow \mathcal{C} \subset \mathbb{C}^2; t \mapsto \mathcal{P}(t)$$

is birational, or equivalently, if for almost every point on $\mathcal{C}$ and for almost all values of the parameter in $\mathbb{C}$ the mapping $\mathcal{P}$ is rationally bijective. The notion of properness can also be stated algebraically in terms of fields of rational functions. In fact, a rational parametrization $\mathcal{P}$ is proper if and only if the induced monomorphism $\phi_P$ on the fields of rational functions

$$\phi_P : \mathbb{C}(\mathbb{C}) \rightarrow \mathbb{C}(\mathbb{C}); R(x, y) \mapsto R(\mathcal{P}(t))$$

is an isomorphism. Therefore, $\mathcal{P}$ is proper if and only if the mapping $\phi_P$ is surjective, that is, if and only if $\phi_P(\mathbb{C}(\mathbb{C})) = \mathbb{C}(\mathcal{P}(t)) = \mathbb{C}(t)$. Thus, Lüroth’s theorem implies that any rational curve over $\mathbb{C}$ can be properly parametrized (see [1], [22], [25]). Furthermore, given an improper parametrization, in [18], [21] it is shown how to compute a new parametrization of the same curve being proper.

Intuitively speaking, we say that $\mathcal{P}$ is proper if and only if $\mathcal{P}(t)$ traces $\mathcal{C}$ only once. In this sense, we may generalize the above notion by introducing the notion of tracing index of $\mathcal{P}(t)$. More precisely, we say that $k \in \mathbb{N}$ is the tracing index of $\mathcal{P}(t)$, and we denote it by $\text{index}(\mathcal{P})$, if all but finitely many points on $\mathcal{C}$ are generated, via $\mathcal{P}(t)$, by $k$ parameter values; i.e. $\text{index}(\mathcal{P})$ represents the number of times that $\mathcal{P}(t)$ traces $\mathcal{C}$. Hence, the birationality of $\phi_P$, i.e. the properness of $\mathcal{P}(t)$, is characterized by tracing index 1 (for further details see [22]).

In the following, we outline the algorithm developed in [18] that computes a rational proper reparametrization of an improperly parametrized algebraic plane curve. The algorithm is valid over any field, and it is based on the computation of polynomial gcds and univariate resultants.

For reasons of completeness, we summarize some properties of the resultant that will be used through the paper. To start with, we represent the univariate resultant of two polynomials $A, B \in \mathbb{C}[x_1, \ldots, x_n, t]$ as $\text{Res}_t(A, B)$. It holds that $\text{Res}_t(A, B) \in \mathbb{C}[x_1, \ldots, x_n]$, and $\text{Res}_t(A, B) = 0$ if and only if $A, B$ have a common factor (depending on $t$). In addition, the resultant is contained in the ideal generated by its two input polynomials, and hence if $A(\alpha, b) = B(\alpha, b) = 0$ where $\alpha = (\alpha_1, \ldots, \alpha_n)$, then $\text{Res}_t(A, B)(\alpha) = 0$. Reciprocally, if $\text{Res}_t(A, B)(\alpha) = 0$, then $\text{lc}(A, t)(\alpha) = \text{lc}(B, t)(\alpha) = 0$ ($\text{lc}(A, t)$ denotes the leading coefficient of $A$ w.r.t $t$) or there exists $b \in \mathbb{C}$ such that $A(\alpha, b) = B(\alpha, b) = 0$ (for more details see for instance Chapter 3 in [7], or Sections 5.8 and 5.9 in [26]).

Additionally, we remind the reader the following specialization resultant property that will be used for our purposes: if $\alpha \in \mathbb{C}^n$ is such that $\deg_t(\varphi_\alpha(A)) = \deg_t(A)$, and $\deg_t(\varphi_\alpha(B)) = \deg_t(B)$, then $\text{Res}_t(A, B)(\alpha) = 0$.\]
deg(B) − k then,

\[ \varphi_\alpha(\text{Res}_t(A, B)) = \varphi_\alpha(\text{lc}(A, t))^k\text{Res}_t(\varphi_\alpha(A), \varphi_\alpha(B)), \]

where \( \varphi_\alpha \) is the natural evaluation homomorphism

\[ \varphi_\alpha : \mathbb{C}[x_1, \ldots, x_n, t] \rightarrow \mathbb{C}[x_1, \ldots, x_n, t] ; \ A(x_1, \ldots, x_n, t) \mapsto A(\alpha_1, \ldots, \alpha_n, t) \]

(see Lemma 4.3.1, pp.96 in [29]).

Finally, given \( R(t) = r_1(t)/r_2(t) \in \mathbb{C}(t) \), where \( \gcd(r_1, r_2) = 1 \), we define \( \deg(R) \) as the maximum of \( \deg(r_1) \) and \( \deg(r_2) \).

---

**Symbolic Algorithm Reparametrization for Curves.**

**Input:** a rational affine parametrization \( \mathcal{P}(t) = (p_{1,1}(t)/p_{1,2}(t), p_{2,1}(t)/p_{2,2}(t)) \in \mathbb{C}(t)^2 \), with \( \gcd(p_{1,1}, p_{1,2}) = 1, i = 1, 2 \), of a plane algebraic curve \( \mathcal{C} \).

**Output:** a rational proper parametrization \( \mathcal{Q}(t) \in \mathbb{C}(t)^2 \) of \( \mathcal{C} \), and a rational function \( R(t) \in \mathbb{C}(t) \setminus \mathbb{C} \) such that \( \mathcal{P}(t) = \mathcal{Q}(R(t)) \).

1. Compute \( H_j(t, s) = p_{j,1}(t)p_{j,2}(s) - p_{j,1}(s)p_{j,2}(t) \), \( j = 1, 2 \).
2. Determine the polynomial \( S(t, s) = \gcd(H_1(t, s), H_2(t, s)) = C_m(t)s^m + \cdots + C_0(t) \).
3. If \( \deg(S) = 1 \), RETURN \( \mathcal{Q}(t) = \mathcal{P}(t) \), and \( R(t) = t \). Otherwise go to Step 4.
4. Consider a rational function \( R(t) = \frac{C_i(t)}{C_j(t)} \in \mathbb{C}(t) \), such that \( C_j(t), C_i(t) \) are two of the polynomials obtained in Step 2 such that \( \gcd(C_j, C_i) = 1 \), and \( C_jC_i \notin \mathbb{C} \).
5. For \( k = 1, 2 \), compute the polynomials

\[ L_k(s, x_k) = \text{Res}_t(G_k(t, x_k), sC_j(t) - C_i(t)) = (q_{k,2}(s)x_k - q_{k,1}(s))^{\deg(R)}, \]

where \( G_k(t, x_k) = x_kp_{k,2}(t) - p_{k,1}(t) \).
6. RETURN \( \mathcal{Q}(t) = (q_{1,1}(t)/q_{1,2}(t), q_{2,1}(t)/q_{2,2}(t)) \in \mathbb{C}(t)^2 \), and \( R(t) = C_i(t)/C_j(t) \).

**Remark 1.** It is proved that \( \text{index}(\mathcal{P}) = \deg(S) \) (see Theorem 2 in [22]). In addition, for all but finitely many values \( \alpha \) of the variable \( s \), \( \deg(S) = \deg(\gcd(H_1(t, \alpha), H_2(t, \alpha))) \) (see Lemma 4 and Subsection 3.1 in [22]).

In the following example, we illustrate Symbolic Algorithm Reparametrization for Curves.

**Example 1.** Let \( \mathcal{C} \) be the rational curve defined by the parametrization

\[ \mathcal{P}(t) = \left( \frac{p_{1,1}(t)}{p_{1,2}(t)}, \frac{p_{2,1}(t)}{p_{2,2}(t)} \right) = \]

\[ \left( \frac{p_{1,1}(t)}{p_{1,2}(t)}, \frac{p_{2,1}(t)}{p_{2,2}(t)} \right) = \]
Now, we determine \( S(t,s) \) where
\[
H_1(t,s) = \frac{91t^4 - 19t^6 + 31t^5 - 26t^6 s^3 - 49t^6 s^2 + 26t^5 s^3 - 31t^4 s^2 + 91t^4 s^3 - 11t^5 s^2 - 195t^2 s^5 + 195t^2 s^3 + 234t s^3 + 19s^5 + 31s^5 - 107s^4 + 39s^3 - 216s^2 - 54ts^2 + 12ts^4 + 66ts^6 + 6t^2 s^4 + 54t^2 s^6 + 11t^2 s^5 - 91t^3 s^4 - 234t^3 s + 26t^3 s^6 - 26t^3 s^5 - 12t^4 s + 27t^4 s^6 + t^4 s^5 - 27t^6 s^4 - 66t^6 s - 8t^6 s^5 - t^5 s^4 - 6t^5 s + 8t^5 s^6. \\
H_2(t,s) = 36t^2 + 24t^3 - 8t^6 - 5t^5 - 5t^6 s^3 - 9t^6 s^2 - 4t^5 s^3 - 8t^4 s^3 - 36t^3 s^2 + 36t^2 s^3 - s^6 + 4s^5 + 8s^4 - 24s^3 - 36s^2 + 9t^2 s^6 + 8t^3 s^4 + 5t^3 s^6 + 4t^3 s^5 - 2t^4 s^6 + 2t^6 s^4 + t^6 s^5 - t^5 s^6.
\]

Now, we determine \( S(t,s) \). We obtain
\[
S(t,s) = C_0(t) + C_1(t)s + C_2(t)s^2 + C_3(t)s^3,
\]
where \( C_0(t) = -6t - 2t^2 + t^3 \), \( C_1(t) = 3t^3 + 6 \), \( C_2(t) = t^3 + 2 \), and \( C_3(t) = -t^2 - 3t - 1 \).

Since \( \deg_1(S) > 1 \), we go to Step 4 of the algorithm, and we consider
\[
R(t) = \frac{C_3(t)}{C_2(t)} = \frac{-t^2 - 3t - 1}{t^3 + 2}.
\]
Note that \( \gcd(C_2, C_3) = 1 \). Now, we compute the polynomials
\[
L_1(s, x_1) = \text{Res}_t(G_1(t, x_1), sC_2(t) - C_3(t)) = -961(-3x_1 + sx_1 + 1 - 3s + s^2)^3,
\]
\[
L_2(s, x_2) = \text{Res}_t(G_2(t, x_2), sC_2(t) - C_3(t)) = -961(-x_2 - 1 + s + s^2)^3,
\]
where \( G_i(t, x_i) = x_i p_i(t) - p_{i,1}(t) \), \( i = 1, 2 \) (see Step 5). Finally, in Step 6, the algorithm outputs the proper parametrization \( Q(t) \), and the rational function \( R(t) \)
\[
Q(t) = \left( \frac{-1 - 3t + t^2}{t - 3}, -1 + t + t^2 \right), \quad R(t) = \frac{-t^2 - 3t - 1}{t^3 + 2}.
\]

3. The Problem of Numerical Reparametrization for Curves

The problem of numerical reparametrization for curves can be stated as follows: given the field \( \mathbb{C} \) of complex numbers, a tolerance \( \epsilon > 0 \), and a rational parametrization
\[
\mathcal{P}(t) = (p_{1,1}(t)/p_{1,2}(t), p_{2,1}(t)/p_{2,2}(t)) \in \mathbb{C}(t)^2,
\]
of an algebraic plane curve \( \mathcal{C} \) that is approximate improper (see Definition 1), find a rational parametrization \( Q(t) \in \mathbb{C}(t)^2 \) of an algebraic plane curve \( \mathcal{D} \), and a rational function \( R(t) \in \mathbb{C}(t) \setminus \mathbb{C} \) such that \( Q \) is an \( \epsilon \)-proper reparametrization of \( \mathcal{D} \) (see Definition 3).
In this section, the input and output are not assumed to be exact as in Section 2. Instead, we deal with mathematical objects that are given approximately, probably because they proceed from an exact data that has been perturbed under some previous measuring process or manipulation. Note that, in many practical applications, for instance in the frame of computer aided geometric design, most of data objects are given or become approximate.

In this new situation, the idea is to adapt the algorithm in Section 2 as follows. We consider a rational parametrization $\mathcal{P}(t) \in \mathbb{C}(t)^2$ of an algebraic plane curve $\mathcal{C}$. Note that because of a previous measuring process or manipulation, the parametrization $\mathcal{P}$ is assumed to be given approximately. Afterwards, one computes the polynomials introduced in Step 1 of the symbolic algorithm presented in Section 2.

At this point, in Step 2 of the symbolic algorithm, since we are working with mathematical objects that are assumed to be given approximately, we have to compute the approximate $\epsilon$gcd, denoted by $\epsilon$gcd, instead of the gcd (note that the gcd of two not exact input polynomials is always 1). There are different $\epsilon$gcd algorithms proposed for inexact polynomials (see for instance, [2, 3, 6, 11, 12, 30]). Some typical algorithms of univariate polynomials are included in the mathematical softwares, for example, Maple provides some $\epsilon$gcd algorithms in the package SNAP. We here introduce the $\epsilon$gcd algorithm for a pair of univariate numeric polynomials by using QR factoring. It is implemented in Maple as the function $\text{QRGCD}$. The $\text{QRGCD}(f, g, x, \epsilon)$ function returns univariate numeric polynomials $u, v, d$ such that $d$ is an $\epsilon$gcd for the input polynomials $\langle f(x), g(x) \rangle$. With high probability, the output polynomials satisfy $\|uf + vg - d\|_2 < \epsilon\|f, g, u, v, d\|_2$, $\|f - df_1\|_2 < \epsilon\|f\|_2$, and $\|g - dg_1\|_2 < \epsilon\|g\|_2$, where the polynomials $f_1$ and $f_2$ are cofactors of $f$ and $g$ with respect to the divisor $d$.

Under these conditions, in the following we deal with the notion of approximate improper index of $\mathcal{P}$. This notion generalizes the concept of tracing index (see Section 2) to the numerical situations.

**Definition 1.** We define the approximate improper index of $\mathcal{P}$, and we denote it $\epsilon\text{index}(\mathcal{P})$, as

$$\epsilon\text{index}(\mathcal{P}) = \deg_t(S_{\epsilon\text{PP}}).$$

$\mathcal{P}$ is said to be approximate improper or $\epsilon$-improper if $\epsilon\text{index}(\mathcal{P}) > 1$. Otherwise, it is said to be approximate proper or $\epsilon$-proper.

Note that in the symbolic situation, one can get the tracing index with probability one, by counting the common solutions for a specialized $s_0$ (see Remark 1). Similarly, for the numerical situation, we can fix $s = s_0 \in \mathbb{C}$ as a specialization and find the $\epsilon$gcd for two univariate polynomials $H_1^{\text{PP}}(t, s_0)$ and $H_2^{\text{PP}}(t, s_0)$, under tolerance $\epsilon$. Hence, we first can find the approximate improper index by the specialization and the univariate $\epsilon$gcd computation and then, we can recover an $\epsilon$gcd defined by the polynomial $S_{\epsilon\text{PP}}(t, s)$. More precisely, $S_{\epsilon\text{PP}}(t, s)$ can be found from several $S_{\epsilon\text{PP}}(t, s_k), k = 1, \ldots, n$, whose degrees equal to the approximate improper index. The polynomial $S_{\epsilon\text{PP}}(t, s)$ can be computed using least
Given two polynomials $A$, $B$ in $\mathbb{C}[t, s]$ with $\|A\| = \|B\| = 1$, we say that $A \approx B$, if $\|A - B\| \leq \epsilon$, and $\deg_t(A) = \deg_t(B)$, $\deg_s(A) = \deg_s(B)$. Furthermore, we say that $A(t, r(t)) \approx_{\epsilon} 0$ where $r(t) \in \mathbb{C}(t)$, if $\|\text{num}(A(t, r(t)))\| \leq \epsilon\|A\|$.

Let $\epsilon > 0$ be a given tolerance, and let

$$P(t) = (p_1(t), p_2(t)) = \left(\frac{p_{1,1}(t)}{p_{1,2}(t)}, \frac{p_{2,1}(t)}{p_{2,2}(t)}\right) \in \mathbb{C}(t)^2, \quad \epsilon\gcd(p_{j,1}, p_{j,2}) = 1, \quad j = 1, 2$$

be a rational parametrization of a given plane algebraic curve $C$. We remind that $P$ is expected to be given with perturbed float coefficients. We assume that $\text{index}(P) = 1$. Observe that we are working numerically and then, with probability almost one $\deg_t(S) = 1$, where $S$ is the polynomial introduced in Section 2. Otherwise, if $\text{index}(P) > 1$, we may apply Symbolic Algorithm Reparametrization for Curves in Section 2.

We also consider the polynomials

$$S^P_\epsilon(t, s) = \epsilon\gcd(H^P_1, H^P_2), \quad H^P_j(t, s) = p_{j,1}(t)q_{j,2}(s) - q_{j,1}(s)p_{j,2}(t), \quad j = 1, 2,$$
where $s$ is a new variable, and
\[
Q(t) = (q_1(t), q_2(t)) = \left( \frac{q_{1,1}(t)}{q_{1,2}(t)}, \frac{q_{2,1}(t)}{q_{2,2}(t)} \right) \in \mathbb{C}(t)^2, \quad \text{gcd}(q_{j,1}, q_{j,2}) = 1, \ j = 1, 2
\]
a rational parametrization of a new plane curve. In these conditions, we say that $\mathcal{P}(t) \sim_{\epsilon} Q(r(t))$ if $S_{\epsilon}^{\mathcal{P}Q}(t, r(t)) \approx_{\epsilon} 0$, where $r(t) \in \mathbb{C}(t)$ (see Definition 2).

Observe that since $\text{gcd}(p_{j,1}, p_{j,2}) = 1, \ j = 1, 2$, then $S_{\epsilon}^{\mathcal{P}Q}(t, s) \in \mathbb{C}[t, s] \setminus \mathbb{C}[s]$. Similarly, since $\text{gcd}(q_{j,1}, q_{j,2}) = 1, \ j = 1, 2$, we also get that $S_{\epsilon}^{\mathcal{P}Q}(t, s) \in \mathbb{C}[t, s] \setminus \mathbb{C}[t]$.

Through the paper, we assume that $\mathcal{P}(t) \not\sim_{\epsilon} (a, b) \in \mathbb{C}^2$ (see Remark 2, statement 1).

**Remark 2.** Observe that:

1. Since $\mathcal{P}(t) \not\sim_{\epsilon} (a, b) \in \mathbb{C}^2$, we have that $\deg(S_{\epsilon}^{\mathcal{P}Q}(t, s)) \geq 1$. Indeed: note that $H_{\epsilon}^{\mathcal{P}Q}(t, s) \approx_{\epsilon} (t - s)N_j(t, s)$, where $N_j \in \mathbb{C}[t, s], \ j = 1, 2$. It holds that $N_j \neq 0, \ j = 1, 2$; otherwise, $S_{\epsilon}^{\mathcal{P}Q}(t, s) \approx_{\epsilon} 0$, and in particular $S_{\epsilon}^{\mathcal{P}Q}(t, s_0) \approx_{\epsilon} 0$ for $s_0 \in \mathbb{C}$ satisfying that $p_{j,2}(s_0)p_{2,2}(s_0) \neq 0$. Then, $\mathcal{P}(t) \sim_{\epsilon} \mathcal{P}(s_0) \in \mathbb{C}^2$ which is impossible, and thus $N_j \neq 0, \ j = 1, 2$. Hence, $S_{\epsilon}^{\mathcal{P}Q}(t, s) \approx_{\epsilon} (t - s)N(t, s)$, where $N \in \mathbb{C}[t, s] \setminus \{0\}$.
2. $\text{index}(\mathcal{P}) = 1$ if and only if $S_{\epsilon}^{\mathcal{P}Q}(t, s) \approx_{\epsilon} (t - s)$.
3. Clearly the notion of approximate improper index generalizes the notion of tracing index. In particular, if $\text{index}(\mathcal{P}) = 1$ then $\text{index}(\mathcal{P}) = 1$.

Now, we are ready to introduce the notions of $\epsilon$-numerical reparametrization and $\epsilon$-proper reparametrization.

**Definition 3.** Let $\mathcal{P}(t) = (p_1(t), p_2(t)) \in \mathbb{C}(t)^2$ be a rational parametrization of a given plane curve $\mathcal{C}$. We say that a parametrization $Q(t) = (q_1(t), q_2(t)) \in \mathbb{C}(t)^2$ is an $\epsilon$-numerical reparametrization of $\mathcal{P}(t)$ if there exists $R(t) = M(t)/N(t) \in \mathbb{C}(t) \setminus \mathbb{C}$, with $\text{gcd}(M, N) = 1$, such that $\mathcal{P} \sim_{\epsilon} Q(R)$. In addition, if $\text{index}(Q) = 1$, then we say that $Q$ is an $\epsilon$-proper reparametrization of $\mathcal{P}$.

Using Definition 3, and the notations introduced above, we obtain some theorems where some properties characterizing numerical reparametrizations are proved. We start with the following proposition.

**Proposition 1.** Let $Q(t) = (q_1(t), q_2(t)) \in \mathbb{C}(t)^2, q_j = q_{j,1}/q_{j,2}, \ \text{gcd}(q_{j,1}, q_{j,2}) = 1, \ j = 1, 2$ be such that $Q(t) \not\sim_{\epsilon} (a, b) \in \mathbb{C}^2$. Let $R(t) = M(t)/N(t) \in \mathbb{C}(t) \setminus \mathbb{C}$, with $\text{gcd}(M, N) = 1$. Then, up to constants in $\mathbb{C} \setminus \{0\}$, it holds that
\[
S_{\epsilon}^{Q(R)}(t, s) \approx_{\epsilon} \text{num}(S_{\epsilon}^{Q(R)}(R(t), R(s)) ).
\]
Proof. From the definition of $S_e^Q(t, s)$, there are $M_1, M_2 \in \mathbb{C}[t, s]$ that satisfy
\[ H_j^Q(t, s) \approx_e S_e^Q(t, s)M_j(t, s), \quad j = 1, 2, \quad \text{and} \quad \gcd(M_1, M_2) = 1 \] (see Definition 2). Now, taking into account the definition of $S_e^{Q(R)Q(R)}$, one gets that
\[ S_e^{Q(R)Q(R)}(t, s) = \gcd(H_1^{Q(R)Q(R)}(t, s), H_2^{Q(R)Q(R)}(t, s)) = \gcd(\num(H_1^Q(R(t), R(s))), \num(H_2^Q(R(t), R(s)))) \] (2)
Thus, from the above equalities, one deduces that
\[ S_e^{Q(R)Q(R)}(t, s) \overset{(2)}{=} \gcd(\num(H_1^Q(R(t), R(s))), \num(H_2^Q(R(t), R(s)))) \overset{(1)}{=} \epsilon \gcd(\num(S_e^Q(R(t), R(s)))) M(t, s), \]
where
\[ M(t, s) := \gcd(\num(M_1(R(t), R(s))), \num(M_2(R(t), R(s)))) \).
Since $\gcd(M_1, M_2) = 1$, we have that $M(t, s) = 1$, and we conclude that
\[ S_e^{Q(R)Q(R)}(t, s) \approx_e \num(S_e^Q(R(t), R(s))). \]

In the following, we consider $P(t) = (p_1(t), p_2(t)) \in \mathbb{C}(t)^2$ the input rational parametrization of the given plane curve $C$. Let $Q(t) \in \mathbb{C}(t)^2$ be an $\epsilon$-numerical reparametrization of $P(t)$, and $P \sim_\epsilon Q(R)$ where $R(t) = M(t)/N(t) \in \mathbb{C}(t)\setminus\mathbb{C}$ (see Definition 3). In these conditions, we have the following results.

**Theorem 1.** $Q$ is $\epsilon$-proper if and only if $\epsilon\text{index}(P) = \deg(R)$.

Proof. If $\epsilon\text{index}(Q) = 1$, from Proposition 1 and Remark 2 (statement 2), one deduces that
\[ S_e^{PP}(t, s) \approx_e \num(S_e^Q(R(t), R(s))) \approx_e \num(R(t) - R(s)) = M(t)N(s) - M(s)N(t). \]
Therefore, $\epsilon\text{index}(P) = \deg_e(S_e^{PP}) = \deg(R)$ (see Definitions 1 and 2). Reciprocally, from Proposition 1, we have that
\[ S_e^{PP}(t, s) \approx_e \num(S_e^Q(R(t), R(s))) \]
which implies that $\deg_e(S_e^{PP}) = \deg_e(S_e^Q)\deg(R)$ (see Definition 2). Therefore, if $\epsilon\text{index}(P) = \deg(R)$, then $\deg_e(S_e^Q) = 1$ and thus $Q$ is $\epsilon$-proper (see Definition 1).
Corollary 1. It holds that $\epsilon\text{index}(P) = \epsilon\text{index}(Q)\deg(R)$.

**Proof.** Reasoning as in proof of Theorem 1, one deduces that $\deg_t(S^\epsilon_{PP}) = \deg_t(S^\epsilon_{QQ})\deg(R)$. Thus, from Definition 1, we conclude that $\epsilon\text{index}(P) = \epsilon\text{index}(Q)\deg(R)$.

Corollary 2. $S^\epsilon_{PP}(t, s) \approx_\epsilon \text{num}(R(t) - R(s)) = M(t)N(s) - M(s)N(t)$, if and only if $Q$ is $\epsilon$-proper.

**Proof.** If $\epsilon\text{index}(Q) = 1$, reasoning as in proof of Theorem 1, one deduces that $S^\epsilon_{PP}(t, s) \approx_\epsilon \text{num}(R(t) - R(s))$. Reciprocally, if $S^\epsilon_{PP}(t, s) \approx_\epsilon \text{num}(R(t) - R(s))$, we get that $\epsilon\text{index}(P) = \deg_t(S^\epsilon_{PP}) = \deg(R)$ (see Definition 2). Thus, from Corollary 1, we conclude that $Q$ is $\epsilon$-proper.

Construction of the Rational Function $R(t)$

In the following, we construct a rational function $R(t) \in \mathbb{C}(t) \setminus \mathbb{C}$, such that there exists an $\epsilon$-proper reparametrization of $P$. That is, there exists $Q$ such that $P \sim_\epsilon Q(R)$ and $Q$ is $\epsilon$-proper (see Theorem 2 and Corollary 3). Hence, we are addressing the existence of the $\epsilon$-proper reparameterization.

For this purpose, we first write

$$S^\epsilon_{PP}(t, s) = C_m(t)s^m + C_{m-1}(t)s^{m-1} + \cdots + C_0(t). \quad (3)$$

This polynomial is computed from the input parametrization $P$, and then it is known. Furthermore, taking into account Corollary 2, we have that

$$S^\epsilon_{PP}(t, s) \approx_\epsilon \text{num}(R(t) - R(s)),$$

where $R(t) = M(t)/N(t) \in \mathbb{C}(t) \setminus \mathbb{C}$ is the unknown rational function we are looking for. That is, we look for $R$ satisfying the above condition.

In the symbolic situation, Lemma 3 in [18] states that, up to constants in $\mathbb{C} \setminus \{0\}$,

$$\text{num}\left(\frac{C_i(t)}{C_j(t)} - \frac{C_i(s)}{C_j(s)}\right) = C_m(t)s^m + C_{m-1}(t)s^{m-1} + \cdots + C_0(t),$$

where $C_i, C_j$ are such that $C_iC_j \not\in \mathbb{C}$, and $\gcd(C_i, C_j) = 1$.

Therefore, the unknown rational function, $R(t) \in \mathbb{C}(t) \setminus \mathbb{C}$, can be constructed as

$$R(t) = \frac{C_i(t)}{C_j(t)} \in \mathbb{C}(t) \setminus \mathbb{C},$$

where $C_i$ and $C_j$ are from (3) satisfying that:
Proof. First, we observe that above. Then, we have that $S^\epsilon(t, s) \approx_\epsilon \text{num}(R(t) - R(s))$.

Construction and Properties of the $\epsilon$-Numerical Reparametrization $Q(t)$

In the following, we consider the rational function $R(t) = \frac{C_i(t)}{C_j(t)} \in \mathbb{C}(t) \setminus \mathbb{C}$ computed as above. Then, we have that $S^\epsilon(t, s) \approx_\epsilon \text{num}(R(t) - R(s))$. Hence, from Corollary 2, if $Q$ is such that $\mathcal{P} \sim_\epsilon Q(R)$, then $Q$ is $\epsilon$-proper.

In Theorem 2, where we show how to compute the $\epsilon$-numerical reparametrization $Q$ of $\mathcal{P}$.

**Theorem 2.** For $k = 1, 2$, let

$$L_k(s, x_k) = \text{Res}_t(G_k(t, x_k), sC_j(t) - C_i(t)), \text{ where } G_k(t, x_k) = x_kp_{k,2}(t) - p_{k,1}(t).$$

If

$$L_k(s, x_k) = (x_kq_{k,2}(s) - q_{k,1}(s))^\ell + \epsilon^tW_k(s, x_k), \quad \|\text{num}(W_k(R, p_k))\| \leq \|H_k^{PQ}\|^\ell, \quad k = 1, 2,$$

where $\ell = \text{deg}(R)$, and $\text{gcd}(q_{k,1}, q_{k,2}) = 1$, then $Q(s) = \left(\frac{q_{1,1}(s)}{q_{1,2}(s)}, \frac{q_{2,1}(s)}{q_{2,2}(s)}\right)$ is an $\epsilon$-numerical reparametrization of $\mathcal{P}$.

**Proof.** First, we observe that $L_k \neq 0$ (otherwise, $G_k$ and $sC_j(t) - C_i(t)$ have a common factor depending on $t$, which is impossible because $\text{gcd}(C_i, C_j) = 1$). In addition, it holds that $\text{deg}_{x_k}(L_k) = \text{deg}(R)$. Indeed, since

$$L_k(s, x_k) = \text{Res}_t(G_k(t, x_k), sC_j(t) - C_i(t)),$$

we get that, up to constants in $\mathbb{C}(s) \setminus \{0\}$,

$$L_k(s, x_k) = \prod_{\{\alpha | sC_j(\alpha) - C_i(\alpha) = 0\}} G_k(\alpha, x_k),$$

(see Sections 5.8 and 5.9 in [26]), and thus

$$\text{deg}_{x_k}(L_k) = \text{deg}_t(sC_j(t) - C_i(t))\text{deg}_{x_k}(G_k(t, x_k)) = \text{deg}(R).$$

In addition, from $\text{deg}_{x_k}(L_k) = \text{deg}(R)$, we deduce that $\text{deg}_{x_k}(W_k) \leq \ell$. In fact, since we are working numerically, we may assume w.l.o.g that $\text{deg}_{x_k}(W_k) = \ell$.

Now, taking into account the properties of the resultant (see Section 2), one has that

$$0 = L_k(R(t), p_k(t)) = (p_k(t)q_{k,2}(R(t)) - q_{k,1}(R(t)))^\ell + \epsilon^tW_k(R(t), p_k(t)).$$
Then,
\[
\num(H_k^{\mathcal{Q}}(t, R(t)))^\ell = \varepsilon^\ell e_k(t), \quad \text{where} \quad e_k = -W_k(R(t), p_k(t))p_{k,2}(t)^\ell C_{j}^{\text{deg}(q_k)}, \quad k = 1, 2.
\]

Since \( \deg_{x_k}(L_k) = \ell \), and \( \deg_{x_k}(W_k) = \ell \deg(q_k) \) (see Corollary 4), one has that \( e_k = -\num(W_k(R(t), p_k(t))) \in \mathbb{C}[t] \) (i.e. the denominator of \( W_k(R(t), p_k(t)) \) is canceled with \( p_{k,2}(t)^\ell C_{j}(t)^{\text{deg}(q_k)} \)). Therefore, from \( \num(H_k^{\mathcal{Q}}(t, R(t)))^\ell = \varepsilon^\ell e_k(t) \), we get that
\[
\| \num(H_k^{\mathcal{Q}}(t, R(t))) \|^\ell = \varepsilon^\ell \| e_k \| \leq \varepsilon^\ell \| H_k^{\mathcal{Q}} \|^\ell,
\]
which implies that \( H_k^{\mathcal{Q}}(t, R(t)) \approx \varepsilon \) (see Definition 2). Thus, \( S_{\varepsilon}^{\mathcal{Q}}(t, R(t)) \approx \varepsilon \), and then \( \mathcal{P}(t) \approx_{\varepsilon} \mathcal{Q}(R(t)) \).

\[\text{Remark 3. From the proof of Theorem 2, we have that} \quad \deg_{x_k}(L_k) = \deg_{x_k}(W_k) = \deg(R).\]

\[\text{Remark 4. If the tolerance in Theorem 2 changes (that is, instead } \varepsilon \text{ we have } \tau, \text{ Theorem 2 holds. More precisely, if} \]
\[L_k(s, x_k) = (x_kq_{k,2}(s) - q_{k,1}(s))^\ell + \tau^\ell W_k(s, x_k), \quad \| \num(W_k(R, p_k)) \| \leq \| H_k^{\mathcal{Q}} \|^\ell,
\]

\[\text{where } \ell = \deg(R) \text{ and } \gcd(q_{k,1}, q_{k,2}) = 1, \text{ then } \mathcal{Q}(s) = \left( \frac{q_{1,1}(s)}{q_{1,2}(s)}, \frac{q_{2,1}(s)}{q_{2,2}(s)} \right) \text{ is an } \tau \text{-numerical reparametrization of } \mathcal{P}.\]

\[\text{Corollary 3. Let } \mathcal{Q} \text{ be the } \varepsilon \text{-numerical reparametrization of } \mathcal{P} \text{ computed in Theorem 2. It holds that } \mathcal{Q} \text{ is } \varepsilon \text{-proper.}\]

\[\text{PROOF. Since } R(t) = \frac{C(t)}{C_j(t)} \in \mathbb{C}(t) \setminus \mathbb{C} \text{ is such that } S_{\varepsilon}^{\mathcal{P}}(t, s) \approx \varepsilon \num(R(t) - R(s)), \text{ and } \mathcal{Q} \text{ is an } \varepsilon \text{-numerical reparametrization of } \mathcal{P} \text{ (see Theorem 2), from Corollary 2, we conclude that } \mathcal{Q} \text{ is } \varepsilon \text{-proper.}\]

\[\text{Remark 5. Corollaries 1 and 3 imply that } \ell = \varepsilon \text{-index}(\mathcal{P}), \text{ where } \ell = \deg(R) \text{ is introduced in Theorem 2.}\]

In the following, we deduce some properties concerning the \( \varepsilon \)-numerical reparametrization computed in Theorem 2.

\[\text{Corollary 4. It holds that} \]
\[\deg(\mathcal{P}) = \deg(\mathcal{Q})\deg(R).\]
Proof. First, we observe that \( \deg_s(W_k) = \deg(p_k) \), for \( k = 1, 2 \). Indeed, since
\[
L_k(s, x_k) = \text{Res}_t(G_k(t, x_k), sC_j(t) - C_i(t)),
\]
we get that, up to constants in \( \mathbb{C}(x_k) \setminus \{0\} \),
\[
L_k(s, x_k) = \prod_{\{\beta_\ell | G_k(\beta_\ell, x_k) = 0\}} sC_j(\beta_\ell) - C_i(\beta_\ell),
\]
(see Sections 5.8 and 5.9 in [26]), and thus
\[
\deg_s(L_k) = \deg_s(sC_j(t) - C_i(t))\deg_t(G_k(t, x_k)) = \deg(p_k).
\]
Since we are working numerically, we may assume w.l.o.g that \( \deg_s(W_k) = \deg_s(L_k) \). On the other side, from Theorem 2, we have that
\[
L_k(s, x_k) = (x_k q_{k,2}(s) - q_{k,1}(s))^\ell + \ell^\ell W_k(s, x_k), \quad k = 1, 2.
\]
Since we are working numerically, we may assume w.l.o.g that
\[
\deg_s(W_k) = \deg_s((x_k q_{k,2}(s) - q_{k,1}(s))^\ell) = \ell \deg(q_k).
\]
Therefore, \( \ell \deg(q_k) = \deg(p_k) \), \( k = 1, 2 \), which implies that
\[
\deg(P) = \deg(Q) \ell = \deg(Q) \deg(R)
\]
(from Theorem 2, we have that \( \ell = \deg(R) \)). □

Corollary 5. Under the conditions of Theorem 2, it holds that
\[
\text{Res}_t(p_{k,2}(t), sC_j(t) - C_i(t)) = q_{k,2}(s)^\ell + \ell^\ell b_k(s), \quad b_k \in \mathbb{C}[s], \quad k = 1, 2.
\]
Proof. From Theorem 2 and Corollary 4, we have that
\[
L_k(s, x_k) = (x_k q_{k,2}(s) - q_{k,1}(s))^\ell + \ell^\ell W_k(s, x_k), \quad k = 1, 2,
\]
and \( \deg_s(W_k) = \ell \deg(q_k) = \deg(p_k) \). Let \( L_k^*(s, x_k, x_3) \) be the homogeneous form of the polynomial \( L_k(s, x_k) \) w.r.t. the variables \( x_1 \) and \( x_2 \). Using the specialization resultant property (see Section 2), we deduce that
\[
L_k^*(s, x_k, x_3) = \text{Res}_t(x_k p_{k,2}(t) - x_3 q_{k,1}(t), sC_j(t) - C_i(t)) =
\]
\[
= (x_k q_{k,2}(s) - x_3 q_{k,1}(s))^\ell + \ell^\ell b_k(s)x_k^\ell + \ell^\ell U^*(s, x_k, x_3),
\]
where \( U^*(s, x_k, x_3) \) denotes the homogeneous form of the polynomial \( W_k(s, x_k) - b_k(s)x_k^\ell \in (\mathbb{C}[s])x_k \), and \( b_k \) is the leading coefficient of \( W_k \) w.r.t \( x_k \) that is, the coefficient of \( W_k \) w.r.t \( x_k^\ell \) (see Remark 3). Hence, from the specialization resultant property (see Section 2), we get that
\[
L_k^*(s, 1, 0) = \text{Res}_t(p_{k,2}(t), sC_j(t) - C_i(t)) = q_{k,2}(s)^\ell + \ell^\ell b_k(s), \quad k = 1, 2.
\]
□
Corollary 6. Under the conditions of Theorem 2, it holds that

1. The rational function \( q_k(s) = q_{k,1}(s)/q_{k,2}(s) \) could be obtained by simplifying the root in the variable \( x_k \) of the polynomial \( \frac{\partial^{\ell-1} L_k(x_k)}{\partial^{\ell-1} x_k} \), \( k = 1, 2 \).

2. The rational function \( q_k(s) = q_{k,1}(s)/q_{k,2}(s) \) could be obtained by simplifying the rational function \( \frac{-\text{coeff}(L_k, x_k, \ell-1)/\ell}{\text{coeff}(L_k, x_k, \ell)} \), \( k = 1, 2 \).

**Proof.** In order to prove statement 1, we write

\[
L_k(s, x_k) = (x_k q_{k,2}(s) - q_{k,1}(s))^{\ell} + \ell^\ell W_k(s, x_k), \quad k = 1, 2,
\]

where \( \deg_s(W_k) = \ell \deg(q_k) = \deg(p_k) \) (see Corollary 4). Thus,

\[
\frac{\partial^{\ell-1} L_k}{\partial^{\ell-1} x_k}(s, x_k) = \ell! x_k q_{k,2}(s)^{\ell} - \ell! q_{k,1}(s) q_{k,2}(s)^{\ell-1} + \ell! x_k b_k(s) - (\ell - 1)! a_k(s),
\]

where \( b_k(s) \) is the coefficient of \( W_k \) w.r.t \( x_k^\ell \) (see Remark 3), and \( a_k(s) \) is the coefficient of \( W_k \) w.r.t \( x_k^{\ell-1} \). The root in the variable \( x_k \) of this polynomial is

\[
\frac{q_{k,1}(s) q_{k,2}(s)^{\ell-1} + \ell^\ell a_k(s)/\ell}{q_{k,2}(s)^{\ell} + \ell^\ell b_k(s)}, \quad k = 1, 2.
\]

Statement 2 is obtained from statement 1 and using that, for \( k = 1, 2 \),

\[
\text{coeff}(L_k, x_k^\ell) = q_{k,2}(s)^{\ell} + \ell^\ell b_k(s), \quad \text{and} \quad \text{coeff}(L_k, x_k^{\ell-1}) = -\ell q_{k,2}(s)^{\ell-1} q_{k,1}(s) - \ell^\ell a_k(s),
\]

where \( \text{coeff}(\text{pol}, \text{var}) \) denotes de coefficient of a polynomial \( \text{pol} \) w.r.t. the variable \( \text{var} \).

In the following, we consider the parametrization obtained by applying Corollary 6. More precisely,

\[
\tilde{Q}(s) = (\tilde{q}_1, \tilde{q}_2) = \left( \frac{\tilde{q}_{1,1}}{\tilde{q}_{1,2}}, \frac{\tilde{q}_{2,1}}{\tilde{q}_{1,2}} \right) = \left( \frac{q_{1,1}(s) q_{1,2}(s)^{\ell-1} + \ell^\ell a_1(s)/\ell}{q_{2,2}(s)^{\ell} + \ell^\ell b_1(s)}, \frac{q_{2,1}(s) q_{2,2}(s)^{\ell-1} + \ell^\ell a_2(s)/\ell}{q_{2,2}(s)^{\ell} + \ell^\ell b_2(s)} \right).
\]

We observe that \( \tilde{Q} \) can be further simplified by removing the approximate gcd from the numerator and denominator. For instance, one may use \( QRGCD \) algorithm to compute an approximate gcd of two univariate polynomials (see more details before Definition 1). The simplification of \( \tilde{Q}(s) \) provides the rational parametrization \( Q(s) = \left( \frac{q_{1,1}(s)}{q_{1,2}(s)}, \frac{q_{2,1}(s)}{q_{1,2}(s)} \right) \).

**Relation between the Input Curve and the Output Curve**

Let \( C \) be the input curve defined by the parametrization \( P = (p_{1,1}/p_{1,2}, p_{2,1}/p_{1,2}) \), with index(\( P \)) = 1. Since \( P \) is expected to be given with perturbed float coefficients, we may assume w.l.o.g that gcd\( (p_{k,1}, p_{1,2}) = 1 \), for \( k = 1, 2 \).
Let \( \tilde{D} \) be the output curve defined by the parametrization \( \tilde{\mathbf{Q}} = \left( \frac{\tilde{q}_{1,1}}{\tilde{q}_{1,2}}, \frac{\tilde{q}_{2,1}}{\tilde{q}_{1,2}} \right) \) such that \( \gcd(\tilde{q}_{k,1}, \tilde{q}_{1,2}) = 1 \), for \( k = 1, 2 \). We may assume w.l.o.g that \( \text{index}(\tilde{\mathbf{Q}}) = 1 \). Observe that we are working numerically and then, with probability almost one \( \deg_{\mathbb{S}}(S) = 1 \), where \( S \) is the polynomial introduced in Section 2.

Note that from Corollary 5,

\[
\text{Res}_{\mathbb{S}}(p_{k,2}(t), sC_j(t) - C_i(t)) = q_{k,2}(s)^\ell + \epsilon^\ell b_k(s) = \tilde{q}_{k,2}(s), \quad k = 1, 2
\]

which implies that if \( p_{1,2} = p_{2,2} \), then \( \tilde{q}_{1,2} = \tilde{q}_{2,2} \). In addition, we may assume w.l.o.g that \( \deg(p_{k,1}) = \deg(p_{1,2}) \) and \( \deg(\tilde{q}_{k,1}) = \deg(\tilde{q}_{1,2}) \), for \( k = 1, 2 \) (otherwise, one may apply both parametrizations a birational parameter transformation). Thus,

\[
\deg(p_1) = \deg(p_2) = \deg(p_{1,1}) = \deg(p_{2,1}) = \deg(p_{1,2}), \quad \text{and}
\]

\[
\deg(\tilde{q}_1) = \deg(\tilde{q}_2) = \deg(\tilde{q}_{1,1}) = \deg(\tilde{q}_{2,1}) = \deg(\tilde{q}_{1,2}).
\]

Under these conditions, in the following theorem we prove that \( \deg(f) = \deg(h) \), where \( f \in \mathbb{C}[x_1, x_2] \) is the polynomial defining implicitly the curve \( C \) and \( h \in \mathbb{C}[x_1, x_2] \) is the polynomial defining implicitly the curve \( \tilde{D} \).

**Theorem 3.** The curves \( C \) and \( \tilde{D} \) have the same degree.

**Proof.** First, taking into account that

\[
\deg(p_{1,1}) = \deg(p_{2,1}) = \deg(p_{1,2}), \quad \deg(\tilde{q}_{1,1}) = \deg(\tilde{q}_{2,1}) = \deg(\tilde{q}_{1,2}),
\]

and applying Theorem 6.3.1 in [23], one has that all the infinity points of both curves are reachable by the corresponding projective parametrizations (at most, one affine point is not reachable by the parametrizations). In addition, if \((s_0, w_0)\) is such that \( \tilde{q}_{1,2}(s_0, w_0) = 0 \), where

\[
\tilde{\mathbf{Q}}^*(s, w) = (\tilde{q}_{1,1}(s, w), \tilde{q}_{2,1}(s, w), \tilde{q}_{1,2}(s, w))
\]

is the projective parametrization of the projective curve \( \tilde{D}^* \), then \( \tilde{\mathbf{Q}}^*(s_0, w_0) \) generates an infinity point of \( \tilde{D}^* \). Observe that since \( \deg(\tilde{q}_{1,1}) = \deg(\tilde{q}_{2,1}) = \deg(\tilde{q}_{1,2}) \), it holds that \( \tilde{q}_{1,2}(s_0, w_0) = 0 \) if and only if \( \tilde{q}_{1,2}(s_0) = 0 \). Thus, the homogeneous form of maximum degree of \( h \) is given as

\[
h_{r}(x_1, x_2) = \prod_{\{s_i \mid \tilde{q}_{1,2}(s_i) = 0\}} (x_2\tilde{q}_{1,1}(s_i) - \tilde{q}_{2,1}(s_i)x_1).
\]

Since we are working with approximate mathematical objects, we may assume w.l.o.g that the polynomial \( \tilde{q}_{1,2} \) does not have multiple roots, and then \( r = \deg(h) = \deg(\tilde{q}_{1,2}) \). In addition, since

\[
\text{Res}_{\mathbb{S}}(p_{1,2}(t), sC_j(t) - C_i(t)) = \tilde{q}_{1,2}(s),
\]

...
(see Corollary 5), we get that, up to constants in \( C(x_k) \setminus \{0\} \),

\[
\tilde{q}_{1,2}(s) = \prod_{\{\gamma | p_{1,2}(\gamma) = 0\}} sC_j'(\gamma t) - C_i(\gamma t),
\]

(see Sections 5.8 and 5.9 in [26]), and thus \( \deg(\tilde{q}_{1,2}) = \deg(sC_j(t) - C_i(t))\deg(p_{1,2}) = \deg(p_{1,2}) \). Hence, \( r = \deg(h) = \deg(p_{1,2}) \).

Reasoning similarly with the projective parametrization of the projective input curve \( C^* \), we have that all the infinity points of \( C^* \) are reachable by the projective parametrization

\[
\mathcal{P}^*(t, w) = (p_{1,1}(t, w), p_{2,1}(t, w), p_{1,2}(t, w)).
\]

Hence, the homogeneous form of maximum degree of the polynomial \( f \) is given as

\[
f_d(x_1, x_2) = \prod_{\{t_i | p_{1,2}(t_i) = 0\}} (x_2p_{1,1}(t_i) - p_{2,1}(t_i)x_1).
\]

Since we are working with approximate mathematical objects, we may assume w.l.o.g that the polynomial \( p_{1,2} \) does not have multiple roots, and then \( d = \deg(f) = \deg(p_{1,2}) \). Therefore, we conclude that \( d = \deg(f) = \deg(h) = \deg(p_k) = \deg(\tilde{q}_k) \).

\[
\square
\]

**Remark 6.** Observe that \( x_k, k = 1, 2 \) does not divides \( f_d \). That is, \( f_d(1, 0) \neq 0 \), and \( f_d(0, 1) \neq 0 \). Indeed, if \( f_d(1, 0) = 0 \) (similarly if \( f_d(0, 1) = 0 \)), one gets that there exists \( t_i \in C \) such that \( p_{1,2}(t_i) = 0 \) and \( p_{2,1}(t_i) = 0 \). This is impossible, because we have assumed that \( \gcd(p_{1,2}, p_{2,1}) = 1 \).

Reasoning similarly one may prove that that \( x_k, k = 1, 2 \) does not divides \( h_d \).

Using Theorem 3, in the following theorem we prove that the curves \( C \) and \( \tilde{D} \) have the same behavior at infinity. More precisely, we show that the homogeneous form of maximum degree of \( h \) is equal to the homogeneous form of maximum degree of \( f \).

**Theorem 4.** The implicit equation defining the curves \( C \) and \( \tilde{D} \) have the same homogeneous form of maximum degree, and hence both curves have the same points at infinity.

**Proof.** First, by applying Theorem 8 in [22], one has that

\[
f(x_1, x_2) = \Res_i(G_1(t, x_1), G_2(t, x_2), \text{where } G_k(t, x_k) = xkp_{1,2}(t) - p_{k,1}(t), k = 1, 2.
\]

We recall that we are assuming that \( \text{index}(\mathcal{P}) = 1 \). Now, we consider the polynomials

\[
L_k(s, x_k) = \Res_i(G_k(t, x_k), sC_j(t) - C_i(t)), k = 1, 2,
\]

introduced in Theorem 2. By Corollary 6, we have that

\[
L_k(s, x_k) = x_k^{\ell} \tilde{q}_{1,2} - \ell x_k^{\ell-1} \tilde{q}_{k,1} + A_k(s, x_k), \quad \deg_{x_k}(A_k) \leq \ell - 2, \quad k = 1, 2.
\]
Let us prove that there exists a non empty open subset $\Omega \subset \mathbb{C}^2$, such that for every $q \in \Omega$ with $f(q) = 0$, it holds that $R(q) = 0$, where $R(x_1, x_2) := \text{Res}_s(L_1, L_2)$. Thus, one would deduce that $f$ divides $R$. Indeed, first we observe that $R \neq 0$ because does not exist any factor depending on $s$ that divides $L_k$ (note that $\gcd(q_{1,2}, q_{k,1}) = 1$). Now, let

$$\Omega = \{ q \in \mathbb{C}^2 \mid \text{lc}(G_1, t)(q)\text{lc}(G_2, t)(q)D_2(q)C_j(\mathcal{P}^{-1}(q)) \neq 0 \},$$

where $D_1(x_1, x_2)/D_2(x_1, x_2) = \mathcal{P}^{-1}(x_1, x_2)$ (note that $\text{index}(\mathcal{P}) = 1$ and then, there exists the inverse of $\mathcal{P}$ in $\mathbb{C}[x_1, x_2] \setminus \mathbb{C}$). Observe that $\Omega$ is a non empty open subset of $\mathbb{C}^2$ since

$$\text{lc}(G_1, t)(x_1)\text{lc}(G_2, t)(x_2)D_2(x_1, x_2)C_j(\mathcal{P}^{-1}(x_1, x_2)) \neq 0.$$  

Now, let $q = (x_1^0, x_2^0) \in \Omega$ be such that $f(q) = 0$ (note that $\mathcal{C}$ and $\mathbb{C}^2 \setminus \Omega$ intersect at finitely many points). Since $\text{lc}(G_j, t)(q) \neq 0$, $j = 1, 2$, by the resultant property (see Section 2), there exists $t_0 \in \mathbb{C}$ such that $G_k(t_0, x_k^0) = 0$, $k = 1, 2$. In addition, since $q \in \Omega$, one has that there exists $s_0 \in \mathbb{C}$ such that $s_0 C_j(t_0) - C_i(t_0) = 0$ (note that $t_0 = \mathcal{P}^{-1}(q)$, and $C_j(t_0) \neq 0$). Then, since $L_k(s, x_k) = \text{Res}_s(G_k(t, x_k), sC_j(t) - C_i(t))$, we get that $L_k(s_0, x_k^0) = 0$, $k = 1, 2$. Hence, by the specialization of the resultant property (see Section 2), we deduce that

$$R(q) = \text{Res}_s(L_1(s, x_1), L_2(s, x_2))(q) = \text{Res}_s(L_1(s, x_1^0), L_2(s, x_2^0)) = 0.$$  

Thus,

$$R(x_1, x_2) = f(x_1, x_2)m(x_1, x_2), \quad m \in \mathbb{C}[x_1, x_2].$$

Since we are working with approximate mathematical objects, we may assume w.l.o.g that $\text{deg}_{(x_1, x_2)}(R) = \text{deg}_s(L_1)\text{deg}_s(L_2)$ (see Sections 5.8 and 5.9 in [26]). Then, if we homogenize the above equation with respect to the variables $x_1$ and $x_2$, we get that

$$R^*(x_1, x_2, x_3) := \text{Res}_s(L_1^*(s, x_1, x_3), L_2^*(s, x_2, x_3)) = F(x_1, x_2, x_3)M(x_1, x_2, x_3),$$

where $F, M \in \mathbb{C}[x_1, x_2, x_3]$ are the homogenization of $f, m$, respectively, with respect to the variables $x_1$ and $x_2$, and

$$L_k^*(s, x_k, x_3) = x_k^\ell \tilde{q}_{1,2} - \ell x_k^{\ell-1} x_3 \tilde{q}_{k,1} + x_3^2 A_k(s, x_k, x_3), \quad \text{deg}_{(x_3, x_3)}(A_k) = \ell - 2, \quad k = 1, 2.$$  

is the homogenization of $L_k$ with respect to $x_1$ and $x_2$. Observe that $x_3$ does not divide to $M$ because $\text{deg}_{(x_1, x_2)}(R) = \text{deg}_s(L_1)\text{deg}_s(L_2)$.

Now, we consider the system defined by the polynomials

$$L_1^* = (x_1^\ell \tilde{q}_{1,2} + x_3^2 A_1(s, x_k, x_3)) + x_3(-\ell x_1^{\ell-1} \tilde{q}_{1,1}), \quad L_2^* = (x_2^\ell \tilde{q}_{1,2} + x_3^2 A_2(s, x_k, x_3)) + x_3(-\ell x_2^{\ell-1} \tilde{q}_{2,1}).$$

Observe that the two equations are independent. Thus, solving from $L_1^* = 0$, we have that $x_3 = (x_1^\ell \tilde{q}_{1,2} + x_3^2 A_1(s, x_k, x_3))/(-\ell x_1^{\ell-1} \tilde{q}_{1,1})$. We substitute it in $L_2^*$, and we obtain the following equivalent system

$$L^*(s, x_1, x_2, x_3) = \tilde{q}_{1,2}(s)x_1^{\ell-1} x_2^{\ell-1}(-\tilde{q}_{2,1}(s)x_1 + \tilde{q}_{1,1}(s)x_2) + x_3^2 B(s, x_1, x_2, x_3), \quad B \in \mathbb{C}[s, x_1, x_2, x_3].$$
Thus, \[ R^*(x_1, x_2, x_3) = \text{Res}_s(L_1^*(s, x_1, x_3), L^*(s, x_2, x_3)) = F(x_1, x_2, x_3)M(x_1, x_2, x_3). \]

Using the property of specialization of the resultants, we consider \( x_3 = 0 \) in the above equality, and we get that (we remind that \( x_3 \) does not divide to \( M \))

\[
\text{Res}_s(\tilde{q}_{1,2}(s), x_1^{\ell-1}x_2^{\ell-1}(-\tilde{q}_{2,1}(s)x_1 + \tilde{q}_{1,1}(s)x_2)) = x_1^{\deg(\tilde{q}_{1,2})(\ell-1)} x_2^{\deg(\tilde{q}_{1,2})(\ell-1)} \text{Res}_s(\tilde{q}_{1,2}(s), (-\tilde{q}_{2,1}(s)x_1 + \tilde{q}_{1,1}(s)x_2)) = f_d(x_1, x_2)m_x(x_1, x_2),
\]

where \( f_d, m_x \) are the homogeneous form of maximum degree of \( F, M \), respectively.

On the other side, by applying Theorem 8 in [22], one also has that

\[
h(x_1, x_2)^{\text{index}(\tilde{Q})} = \text{Res}_t(\tilde{G}_1(t, x_1), \tilde{G}_2(t, x_2)), \quad \text{where} \quad \tilde{G}_k(t, x_k) = x_k\tilde{q}_{1,2}(t) - \tilde{q}_{k,1}(t), \quad k = 1, 2.
\]

We recall that \( \text{index}(\tilde{Q}) = 1 \). Since we are working with approximate mathematical objects, similarly as above we may assume that \( \deg_{\{x_1, x_2\}}(h) = \deg_t(\tilde{G}_1)\deg_t(\tilde{G}_2) \). Then, if we homogenize the above equation with respect to the variables \( x_1 \) and \( x_2 \), we get that

\[
H(x_1, x_2, x_3) = \text{Res}_t(\tilde{G}_1^*(t, x_1, x_3), \tilde{G}_2^*(t, x_2, x_3)), \quad \text{where} \quad \tilde{G}_k^*(t, x_k, x_3) = x_k\tilde{q}_{1,2}(t) - \tilde{q}_{k,1}(t)x_3,
\]

and \( H \) is the homogenization of \( h \) with respect to the variables \( x_1 \) and \( x_2 \). Observe that \( x_3 \) does not divide to \( H \) because \( \deg_{\{x_1, x_2\}}(h) = \deg_t(\tilde{G}_1)\deg_t(\tilde{G}_2) \).

Now, reasoning as above, we have that the system defined by the polynomials \( \tilde{G}_1^* \) and \( \tilde{G}_2^* \) is equivalent to the system defined by \( \tilde{G}_1^* \) and the polynomial \( \tilde{G}^* = -\tilde{q}_{2,1}(s)x_1 + \tilde{q}_{1,1}(s)x_2 \). Thus,

\[
H(x_1, x_2, x_3) = \text{Res}_t(\tilde{G}_1^*(t, x_1, x_3), \tilde{G}^*(t, x_1, x_2)).
\]

Using the property of specialization of the resultants, we consider \( x_3 = 0 \) in the above equality, and we get that (observe that \( x_3 \) does not divide \( H \))

\[
\text{Res}_s(\tilde{q}_{1,2}(s), -\tilde{q}_{2,1}(s)x_1 + \tilde{q}_{1,1}(s)x_2) = h_d(x_1, x_2),
\]

where \( h_d \) is the homogeneous form of maximum degree of \( H \) (we recall that \( d = \deg(f) = \deg(h) \), see Theorem 3). Thus, since

\[
f_d(x_1, x_2)m(x_1, x_2) = x_1^{\deg(\tilde{q}_{1,2})(\ell-1)} x_2^{\deg(\tilde{q}_{1,2})(\ell-1)} \text{Res}_s(\tilde{q}_{1,2}(s), (-\tilde{q}_{2,1}(s)x_1 + \tilde{q}_{1,1}(s)x_2)) = x_1^{\deg(\tilde{q}_{1,2})(\ell-1)} x_2^{\deg(\tilde{q}_{1,2})(\ell-1)} h_d(x_1, x_2),
\]

and \( x_k, \ k = 1, 2 \), does not divides \( f_d \) (see Remark 6), we conclude that \( h_d = f_d \). Hence, \( C \) and \( D \) have the same homogeneous form of maximum degree, and then both curves have the same degree and the same points at infinity. \( \square \)
As we stated above, parametrization $\tilde{Q}$ should be further simplified to obtain the searched parametrization $Q$ (note that by Theorem 3, $\deg(\tilde{q}_k) = \deg(p_k)$, and from Corollary 4, we look for $Q$ such that $\ell \deg(q_k) = \deg(p_k)$). However, when we simplify $\tilde{Q}$, the curve $\tilde{D}$ defined by $\tilde{Q}$ changes (the infinity points are not exactly the same because the numerical simplification). Observe that this is the expected situation because that in fact, the input parametrization $P$ and the output parametrization $Q$ have different degrees (see Corollary 4).

In order to analyze the behavior at affine points, we study the closeness of the curves $C$ and $D$, where $D$ is the curve defined by the simplified parametrization $Q = \left( \frac{q_{1,1}}{q_{1,2}}, \frac{q_{2,1}}{q_{2,2}} \right)$ (note that by Corollary 3, $\text{index}(Q) = 1$), and $C$ is defined by $P = \left( \frac{p_{1,1}}{p_{1,2}}, \frac{p_{2,1}}{p_{2,2}} \right)$. For this purpose, we first assume that $\deg(p_{i,1}) = \deg(p_{i,2})$ and $\deg(q_{i,1}) = \deg(q_{i,2})$, for $i = 1, 2$ (otherwise, one may apply both parametrizations a birational parameter transformation). In addition, let $\|p\| = \max\{\|p_{1,1}\|, \|p_{2,1}\|, \|p_{1,2}\|, \|p_{2,2}\|\}$, and $\|q\| = \max\{\|q_{1,1}\|, \|q_{2,1}\|, \|q_{1,2}\|, \|q_{2,2}\|\}$.

Finally, we also assume that Theorem 2 holds and then, $Q$ is an $\epsilon$-proper reparametrization of $P$ (see Corollary 3). If Theorem 2 does not hold, one applies Remark 4, and then $Q$ an $\tau$-proper reparametrization of $P$. In this case, the formula obtained in Theorem 5 is the same but it involves $\tau$ instead $\epsilon$.

Under these conditions, in order to analyze the behavior at affine points, we restrict us to an interval where the parametrizations $P$ and $Q$ are well defined. Thus, the general strategy we follow is to show that almost any affine real point on $D$ is at small distance of an affine real point on $C$ (and reciprocally).

For this purpose, we consider the interval $I = (d_1, d_2) \subset \mathbb{R}$ satisfying that for all $t_0 \in I$, there exists $M \in \mathbb{N}$ such that $|q_{i,2}(R(t_0))| \geq M$, and $|p_{i,2}(t_0)| \geq M$, $i = 1, 2$. Note that we can decompose $\mathbb{R}$ as union of finitely many intervals, $I_j, j = 1, \ldots, n$, satisfying the above condition (that is, we consider intervals when no roots of the denominators of the parametrizations appear; see [19]). Then, we may reason similarly as we do in Theorem 5 for each interval $I_j, j = 1, \ldots, n$ considered above.

**Theorem 5.** The following statements hold:

1. Let $I = (d_1, d_2) \subset \mathbb{R}$, and $M \in \mathbb{N}$ such that for every $t_0 \in I$, it holds that $|q_{i,2}(R(t_0))| \geq M$, and $|p_{i,2}(t_0)| \geq M$ for $i = 1, 2$. Let $d = \max\{|d_1|, |d_2|\}$. Then, for every $t_0 \in I$,
   \[ |p_i(t_0) - q_i(R(t_0))| \leq 2/M^2 \epsilon C \|p\| \|q\|, \quad i = 1, 2, \]

   where
   \[
   C = \begin{cases} 
   \frac{d^{\deg(P)+1}}{(d-1)^{1/\ell}}, & \text{if } d > 1 \\
   \frac{1}{(1-d)^{1/\ell}}, & \text{if } d < 1 \\
   \ell^{1/\ell \deg(P)}^{1/\ell}, & \text{if } d = 1.
   \end{cases}
   \]
2. \( C_{\ell \in I} \) is contained in the offset region of \( D_{s \in J} \) at distance \( 4\sqrt{2}/M^2\epsilon C \|p\|\|q\| \), where \( J = R(I) \).

3. \( D_{s \in J} \) is contained in the offset region of \( C_{\ell \in I} \) at distance \( 4\sqrt{2}/M^2\epsilon C \|p\|\|q\| \), where \( J = R(I) \).

**Proof.** First, we observe that statement (1) implies statements (2) and (3). For this purpose, we note that for almost all affine real point \( Q \in D \) there exists an affine real point \( P \in C \) such that

\[
\|P - Q\| \leq 2\sqrt{2}/M^2\epsilon C \|p\|\|q\|.
\]

Indeed, using statement (1), we have that

\[
\|P - Q\|_2 = \sqrt{(p_1(t_0) - q_1(R(t_0)))^2 + (p_2(t_0) - q_2(R(t_0)))^2} \leq \sqrt{(2/M^2\epsilon C\|p\|\|q\|)^2 + (2/M^2\epsilon C\|p\|\|q\|)^2} \leq 2\sqrt{2}/M^2\epsilon C \|p\|\|q\|.
\]

Now, reasoning as in Section 2.2 in [8], we deduce statements (2) and (3).

Now, we prove statement (1). For this purpose, from the proof of Theorem 2, we have that

\[
H_i^PQ(t, R(t))^\ell = (p_{i,1}(t)q_{i,1}(R(t)) - q_{i,1}(R(t))p_{i,2}(t))^\ell = \epsilon_i^\ell e_i(t),
\]

where

\[
e_i(t) = -\text{num}(W_i(R(t), p_i(t))) = e_{i,0} + e_{i,1}t + \ldots + e_{i,n_i}t^{n_i} \in \mathbb{C}[t],
\]

and

\[
\|e_i\| = \|\text{num}(W_i(R, p_i))\| \leq \|H_i^PQ\|\ell.
\]

In addition, since \( e_i(t) = -\text{num}(W_i(R(t), p_i(t))) \), we have that \( n_i = \text{deg}(e_i) \leq \ell\text{deg}(P) \) for \( i = 1, 2 \). Indeed, since \( \text{deg}_{x_i}(W_i) = \ell \) (see Remark 3), we deduce that

\[
\text{deg}(e_i) \leq \max\{\text{deg}(R)\text{deg}_{x_i}(W_i), \ell\text{deg}(P)\} \leq \max\{\ell\text{deg}(P), \ell\text{deg}(P)\} = \ell\text{deg}(P).
\]

In these conditions, for every \( t_0 \in I, \) if \( d \neq 1, \) it holds that

\[
|H_i^PQ(t_0, R(t_0))^\ell| = \epsilon_i^\ell |e_i(t_0)| \leq \epsilon_i^\ell \|H_i^PQ\|\ell (|e_{i,0}| + |e_{i,1}|t_0 + \ldots + |e_{i,n_i}|t_0^{n_i}) \leq \epsilon_i^\ell \|H_i^PQ\|\ell (1 + d + \ldots + d^{n_i}) = \epsilon_i^\ell \|H_i^PQ\|\ell d^{n_i+1}/d - 1, \quad i = 1, 2.
\]

(4)

If \( d = 1, \) then

\[
|H_i^PQ(t_0, R(t_0))^\ell| \leq \epsilon_i^\ell \|H_i^PQ\|\ell (1 + |t_0| + \ldots + |t_0|^{n_i}) \leq \epsilon_i^\ell \|H_i^PQ\|\ell (1 + 1 + \ldots + 1) = \epsilon_i^\ell \|H_i^PQ\|\ell n_i.
\]

(5)

Therefore, we conclude that

1. If \( d > 1, \) from (4), and taking into account that \( |g_{i,2}(R(t_0))| \geq M, \) and \( |p_{i,2}(t_0)| \geq M \)

\[
\text{for } i = 1, 2, \text{ we obtain that}
\]

\[
|p_i(t_0) - q_i(R(t_0))| = \frac{|H_i^PQ(t_0, R(t_0))|}{|g_{i,2}(R(t_0))p_{i,2}(t_0)|} \leq 1/M^2\epsilon \|H_i^PQ\|\ell d^{\text{deg}(P)+1}/(d - 1)^{1/\ell} \leq
\]

\[
1/M^2 \epsilon \|H_i^PQ\|\ell d^{\text{deg}(P)+1} / (d - 1)^{1/\ell}.
\]

20
2. If $d < 1$, from (4), and taking into account that $1 - d_{n+1} < 1$, and $|q_{i,2}(R(t_0))| \geq M$, and $|p_{i,2}(t_0)| \geq M$ for $i = 1, 2$, we obtain that

$$|p_i(t_0) - q_i(R(t_0))| = \frac{|H^PQ_i(t_0, R(t_0))|}{|q_{i,2}(R(t_0))|p_{i,2}(t_0)|} \leq \frac{1}{M^2 \epsilon \|H^PQ_i\| \frac{1}{(1 - d)^{1/\ell}}.}
$$

3. If $d = 1$, from (5), and taking into account that $|q_{i,2}(R(t_0))| \geq M$, and $|p_{i,2}(t_0)| \geq M$ for $i = 1, 2$, we obtain that

$$|p_i(t_0) - q_i(R(t_0))| = \frac{|H^PQ_i(t_0, R(t_0))|}{|q_{i,2}(R(t_0))|p_{i,2}(t_0)|} \leq \frac{1}{M^2 \epsilon \|H^PQ_i\| (\ell \deg(P))^{1/\ell}.}
$$

Finally, we note that

$$\|H^PQ_i\| = \|p_{i,1}(t)q_{i,2}(s) - q_{i,1}(s)p_{i,2}(t)\| \leq 2 \|p\| \|q\|.
$$

□

From Theorem 5, we deduce the following corollary:

**Corollary 7.** Under the conditions of Theorem 5, it holds that:

1. If $d \geq 2$, then $C \leq \epsilon^{\deg(P)+1}$.
2. If $1 < d < 2$, then $C \leq 2^{\deg(P)+1}$.

**Proof.** If $d \geq 2$, we use Theorem 5, and we get that $C \leq \frac{\epsilon^{\deg(P)+1}}{(d-1)^{1/\ell}} \leq \epsilon^{\deg(P)+1}$.

If $1 < d < 2$,

$$|H^PQ_i(t_0, R(t_0))| \leq \epsilon^{\ell} \|H^PQ_i\|^\ell (1 + |t_0| + \ldots + |t_0|^n) \leq 

\epsilon^{\ell} \|H^PQ_i\|^\ell (1 + 2 + \ldots + 2^n) \leq \epsilon^{\ell} \|H^PQ_i\|^\ell 2^{n_i+1}, \quad i = 1, 2.
$$

Thus,

$$|p_i(t_0) - q_i(R(t_0))| = \frac{|H^PQ_i(t_0, R(t_0))|}{|q_{i,2}(R(t_0))|p_{i,2}(t_0)|} \leq \frac{1}{M^2 \epsilon \|H^PQ_i\| 2^{\deg(P)+1}} \leq 

2/M^2 \epsilon \|p\| \|q\| 2^{\deg(P)+1}. \quad \square
$$

4. Numeric Algorithm of Reparametrization for Curves

In this section, we apply the results obtained in Section 3 to derive an algorithm that computes an $\epsilon$-proper reparametrization of a given approximate improper parametrization of a plane curve. We outline this approach, and we illustrate it with two examples.
Numeric Algorithm Reparametrization for Curves.

**INPUT:** a tolerance $\epsilon > 0$, and a rational parametrization $P(t) = \left( \frac{p_{1,1}(t)}{p_{1,2}(t)}, \frac{p_{2,1}(t)}{p_{2,2}(t)} \right) \in \mathbb{C}(t)^2$, with $\text{gcd}(p_{1,i}, p_{2,i}) = 1, i = 1, 2$, of a plane algebraic curve $\mathcal{C}$.

**OUTPUT:** a rational parametrization $Q(t) = \left( \frac{q_{1,1}(t)}{q_{1,2}(t)}, \frac{q_{2,1}(t)}{q_{2,2}(t)} \right) \in \mathbb{C}(t)^2$, with $\text{gcd}(q_{1,i}, q_{2,i}) = 1$ for $i = 1, 2$, and such that $\text{index}(Q) = 1$ and $P \sim_{\epsilon} Q(R)$, where $R(t) \in \mathbb{C}(t) \setminus \mathbb{C}$.

1. Compute the polynomials $H^P_k(t, s) = p_{k,1}(t)p_{k,2}(s) - p_{k,1}(s)p_{k,2}(t), \ k = 1, 2$.

2. Compute
   
   $$S^P_\ell(t, s) = \varepsilon \text{gcd}(H^P_1(t, s), H^P_2(t, s)) \approx_\varepsilon C_m(t)s^m + \cdots + C_0(t),$$

   and $\text{index}(P) = \deg_\varepsilon(S^P_\ell)$ (see Definition 1).

3. If $\text{index}(P) = 1$, RETURN $Q(t) = P(t)$, and $R(t) = t$. Otherwise go to Step 4.

4. Consider $R(t) = \frac{C(t)}{C(t)} \in \mathbb{C}(t)$, such that $C_j(t), C_i(t)$ are two of the polynomials obtained in Step 2 satisfying that $C_jC_i \notin \mathbb{C}$, $\text{gcd}(C_j, C_i) = 1$, and $S^P_\ell(t, s) \approx_\varepsilon \text{num}(R(t) - R(s))$.

5. For $k = 1, 2$, compute the polynomials (see Theorem 2)
   
   $$L_k(s, x_k) = \text{Res}_t(G_k(t, x_k), sC_j(t) - C_i(t)), \text{ where } G_k(t, x_k) = x_kp_{k,2}(t) - p_{k,1}(t).$$

6. For $k = 1, 2$, compute the root in the variable $x_k$ of the polynomial $\frac{\partial^{-l}L_k(t, s, x_k)}{\partial^{-l}x_k}$ (see Corollary 6), where $l = \deg(R) = \text{index}(P)$ (see Remark 5). Let $q_k(t) = \tilde{q}_{k,1}(t)/\tilde{q}_{k,2}(t)$ be this root, and let $\tilde{Q}(t) = (\tilde{q}_{1,1}(t)/\tilde{q}_{1,2}(t), \tilde{q}_{2,1}(t)/\tilde{q}_{2,2}(t)) \in \mathbb{C}(t)^2$.

7. Simplify $\tilde{Q}(t)$ (see Remark 7). Let
   
   $$Q(t) = \left( \frac{q_{1,1}(t)}{q_{1,2}(t)}, \frac{q_{2,1}(t)}{q_{2,2}(t)} \right) \in \mathbb{C}(t)^2, \ \text{gcd}(q_{k,1}, q_{k,2}) = 1, \ k = 1, 2,$$

   be the simplified parametrization. Check whether the following equality holds
   
   $$L_k(s, x_k) = (x_kq_{k,2}(s) - q_{k,1}(s))^\ell + \varepsilon^\ell W_k(s, x_k), \ \ |\text{num}(W_k(R, p_k))| \leq \|H^P_\ell\|^\ell$$

   (see Theorem 2). If it does not hold, use Remark 4 and compute $\tau$.

8. RETURN $Q, R$, and the message “$Q$ is an $\varepsilon$-proper reparametrization of $P$” (or “$Q$ is an $\tau$-proper reparametrization of $P$”, if Remark 4 is applied).

**Remark 7.** In the $\text{gcd}$ computation, one may use the SNAP package included in Maple. This package is based on [2], [3], [6] and [12]. For simplification of $Q$ in Step 7, we remove an $\text{gcd}$ from its numerator and denominator under the given tolerance $\epsilon$ (see more details before Definition 1).
In the following, we illustrate Numeric Algorithm Reparametrization for Curves with two examples.

**Example 2.** Let the tolerance \( \epsilon = 0.2 \), and the rational curve \( C \) defined by the parametrization (see Figure 1)

\[
\mathcal{P}(t) = \left( \frac{p_{1,1}(t)}{p_{1,2}(t)}, \frac{p_{2,1}(t)}{p_{2,2}(t)} \right) = \left( \frac{1.999t^2 + 3.999t + 2.005 - 0.003t^4 + 0.001t^3}{2.005 + 0.998t^4 + 6.004t^2 + 3.997t}, \frac{0.001 - 0.998t^4 - 4.003t^3 - 5.996t^2 - 4.005t}{2.005 + 0.998t^4 + 6.004t^2 + 3.997t} \right).
\]

![Figure 1: Parametrization \( \mathcal{P} \)](image)

Using the SNAP package included in Maple, one has that \( \gcd(p_{j,1}, p_{j,2}) = 1, j = 1, 2. \) In these conditions, we apply the algorithm. In Step 1, we compute the polynomials

\[
H_1^{PP}(t, s) = 8030025t^2 + 2007005t^4 + 8022005t^3 + 4010t - 8022005s^3 - 2007005s^4 - 8030025s^2 + 2013014s^2t^4 + 7993994s^2t^3 - 16019993s^2t + 4002993st^4 + 16000001st^3 + 16019993st^2 - 2013014t^2s^4 - 7993994t^2s^3 - 4002993t^4 - 16000001t^3 + 13004s^3t^4 - 13004s^4t^3,
\]

\[
H_2^{PP}(t, s) = 12027984t^2 + 2001988t^4 + 8030017t^3 - 8034022s + 8034022t - 8030017s^3 - 2001988s^4 - 12027984s^2 + 7984s^2t^4 + 38020s^2t^3 + 80008s^2t - 7984st^4 - 28019st^3 - 80008st^2 - 7984t^2s^4 - 38020t^2s^3 + 7984ts^4 + 28019ts^3 - 998ts^4 + 998ts^3.
\]

Now, we compute the polynomial \( S_\epsilon^{PP} \). We have,

\[
S_\epsilon^{PP}(t, s) \approx_\epsilon C_0(t) + C_1(t)s + C_2(t)s^2,
\]

where \( C_0(t) = 35906(5704t + 11637), \) \( C_1(t) = 2202769t - 417838122, \) and \( C_2(t) = -207010593. \) Then, \( \epsilon\text{index}(\mathcal{P}) = \deg_1(S_\epsilon^{PP}) = 2 \) (see Definition 1).
In these conditions, we go to Step 4 of the algorithm, and we consider

\[ R(t) = \frac{C_0(t)}{C_1(t)} = \frac{35906t (5704t + 11637)}{2202769 t - 417838122}. \]

Now, we determine the polynomials

\[ L_1(s, x_1) = \text{Res}_t(G_1(t, x_1), sC_1(t) - C_0(t)) = -0.1657789791 x - 0.3230498883 s \\
+ 0.08367752306 x^2 + 0.3167465668 s^2 + 0.000002969181206 s^4 + 0.002036068597 s^3 \\
- 1.0 xs^2 + 0.6643167904 xs + 0.08210145599 + 0.6582430776 xs^3 + 0.002017625640 xs^4 \\
+ 0.6863872982 x^2 s^2 - 0.3412653300 x^2 s - 0.6904125796 x^2 s^3 + 0.3426060897 x^2 s^4, \]

\[ L_2(s, x_2) = \text{Res}_t(G_2(t, x_2), sC_1(t) - C_0(t)) = 0.002306209600 y - 0.004604780069 s \\
+ 0.06075995785 y^2 + 0.2546086397 s^2 + 0.2487745087 s^4 - 0.4986772157 s^3 - 1.0 y s^3 \\
- 0.2511168983 y s + 0.7530028104 y s^2 - 0.00001165343550 + 0.4975477850 y s^4 \\
+ 0.4983998305 y^2 s^2 - 0.2477997234 y^2 s - 0.5013226695 y^2 s^3 + 0.2487732763 y^2 s^4, \]

where \( G_k(t, x_k) = x_k p_{k,2}(t) - p_{k,1}(t), \ k = 1, 2 \) (see Step 5). In Step 6, we compute the root in the variable \( x_k \) of the polynomial \( \frac{dL_k}{dx_k}(s, x_k) \) (see Corollary 6). We get the rational parametrization (see Figure 2): \( \tilde{Q}(t) = \)

\[
\begin{pmatrix}
1.131266574 t^2 - 0.002282472444 t^4 - 0.7515193794 t + 0.1875402176 - 0.7446483912 t^3 \\
0.1893231696 + 1.552974014 t^2 - 0.7721241210 t - 1.562081346 t^3 + 0.7751576345 t^4 \\
-0.003592982342 - 0.7751595544 t^4 + 1.557960014 t^3 + 0.3912300866 t - 1.173148269 t^2 \\
0.1893231696 + 1.552974014 t^2 - 0.7721241210 t - 1.562081346 t^3 + 0.7751576345 t^4
\end{pmatrix}.
\]

Figure 2: Parametrization \( P \) (black curve) and Parametrization \( \tilde{Q} \) (red curve)
Finally, we simplify $\tilde{Q}$ by removing certain $\epsilon$-gcds (see Remark 7). In this example, we return the simplification of the parametrization $\tilde{Q}$ by using the SNAP package included in Maple. We get the $\epsilon$-numerical reparametrization (see Figure 3):

$$Q(t) = \left(\frac{q_{1.1}(t)}{q_{1.2}(t)}, \frac{q_{2.1}(t)}{q_{2.2}(t)}\right) =$$

$$\left(\frac{-0.00139214373770521 t^2 - 0.455587113115768 t + 0.2308045658878748}{0.472790306463932 t^2 - 0.475516806696674 t + 0.233345983511073}, \frac{-0.472791477433681 t^2 + 0.47300190895789 t - 0.00421763512489261}{0.472790306463932 t^2 - 0.475516806696674 t + 0.233345983511073}\right).$$

One may check that the equality

$$L_k(s, x_k) = (x_k q_{k,2}(s) - q_{k,1}(s))^\ell + \epsilon^\ell W_k(s, x_k), \|\text{num}(W_k(R, p_k))\| \leq \|H_k^PQ\|^{\ell}, \ k = 1, 2$$

holds. Then, “$Q$ is an $\epsilon$-proper reparametrization of $P$” (see Theorem 2 and Corollary 3).

In Figure 3, we can see the relation between parametrization $\tilde{Q}$ and parametrization $Q$. They are almost cover each other.

In the following, we analyze the error in the computation, by using Theorem 5. For this purpose, taking into account the assumptions introduced before Theorem 5, we consider $I = (-10, 10)$. Thus, $d = 10$. Let $M \in \mathbb{N}$ be such that for every $t_0 \in I$, it holds that $|q_{i,2}(R(t_0))| \geq M$, and $|p_{i,2}(t_0)| \geq M$, for $i = 1, 2$. We have that $M = 1007.628025$. Then, by Theorem 5, we deduce that

$$C = \frac{d^{\deg(P)+1}}{(d-1)^{1/\ell}} = 33333.33,$$

In Figure 4, we can see the relation between parametrization $\tilde{Q}$ and parametrization $Q$. They are almost cover each other.
and for every $t_0 \in I$, it holds that
\[ |p_i(t_0) - q_i(R(t_0))| < 2/M^2 \epsilon C \|p\|\|q\| = 0.03749, \quad i = 1, 2, \]
where $\|p\| = \max \{\|p_{1,1}\|, \|p_{2,1}\|, \|p_{1,2}\|, \|p_{2,2}\|\} = 6.004$, and $\|q\| = \max \{\|q_{1,1}\|, \|q_{2,1}\|, \|q_{1,2}\|, \|q_{2,2}\|\} = 0.4755$.

Example 3. Let the tolerance $\epsilon = 0.01$, and the rational curve $C$ defined by the parametrization
\[ P(t) = \left( \frac{p_{1,1}(t)}{p_{1,2}(t)}, \frac{p_{2,1}(t)}{p_{2,2}(t)} \right) = \left( \frac{3t^6 + 4.001t^3 + 1.99}{7t^6 + 7.001t^3 + 1}, \frac{t^6 + 0.0001t}{7t^6 + 7.001t^3 + 1} \right). \]
Using the SNAP package, one has that $\text{egcd}(p_{j,1}, p_{j,2}) = 1$, $j = 1, 2$. In these conditions, we apply the algorithm and, in Step 1, we compute the polynomials
\[ H_1^{PP}(t, s) = -7.004s^6t^3 - 10.93s^6 + 7.004s^3t^6 - 9.93099s^3 + 10.93t^6 + 9.93099t^3, \]
\[ H_2^{PP}(t, s) = 7.001s^6t^3 + s^6 + 0.0007st^6 + 0.0007001st^3 + 0.0001s - 7.001s^3t^6 - t^6 - 0.0007ts^3 - 0.0001t, \]
Now, we compute the polynomial $S_{\epsilon}^{PP}$. We have,
\[ S_{\epsilon}^{PP}(t, s) \approx \epsilon C_0(t) + C_1(t)s + C_2(t)s^2 + C_3(t)s^3, \]
where
\[ C_0(t) = \frac{559}{4610898} t^3 + \frac{1}{61231746870} t + \frac{31}{24584294} t^2; \]
\[
C_1(t) = \frac{528652565}{4367968227972} t^2 + \frac{676064579}{536152992199734} t - \frac{1}{61231746870},
\]
\[
C_2(t) = -\frac{2571167547}{106220786065948} t - \frac{55}{21808761}, \quad C_3 = -\frac{23}{112128382}.
\]

Then, \( \ell := \text{eindex}(P) = \deg_\epsilon(S^{PP}) = 2 \) (see Definition 1).

In these conditions, we apply Step 4 of the algorithm, and we consider
\[
R(t) = \frac{C_0(t)}{C_3(t)} = -t^3 - \frac{256161}{1901585916685} t - \frac{71468919}{6871310173} t^2.
\]

In Steps 5 and 6 of the algorithm, we determine the polynomials \( L_k(s, x_k) \), and we compute the root in the variable \( x_k \) of the polynomial \( \frac{\partial L_k}{\partial x_k}(s, x_k), k = 1, 2 \). We get the rational parametrization \( \tilde{Q}(t) \). We simplify it, and we return the \( \epsilon \)-numerical reparametrization
\[
\tilde{Q}(t) = \left(\frac{q_{1,1}(t)}{q_{1,2}(t)}, \frac{q_{2,1}(t)}{q_{2,2}(t)}\right) = \left(\frac{0.7498146927 t^2 - 1.0 t + 0.4973778072}{1.749567617 t^2 - 1.749810875 t + 0.2499421551}, \frac{0.2499382310 t^2 + 0.0000009544277226 t + 0.000001948317819}{1.749567617 t^2 - 1.749810875 t + 0.2499421551}\right).
\]

One may check that the equality of Theorem 2 does not hold. However, Remark 4 holds taking \( \tau = 0.04 \). Then, \( \tilde{Q} \) is an \( \tau \)-proper reparametrization of \( P \). In Figure 5, we plot the input curve and the output curve.

In the following, we analyze the error in the computation, by using Theorem 5. For this purpose, we consider \( I = (0, 2) \). Thus, \( d = 2 \). Let \( M \in \mathbb{N} \) be such that for every \( t_0 \in I \), it holds that \( |q_{i,2}(R(t_0))| \geq M \), and \( |p_{i,2}(t_0)| \geq M \), for \( i = 1, 2 \). We have that \( M = 1000 \). Then, by Theorem 5, we deduce that
\[
C = \frac{d^{\deg(P)+1}}{(d-1)^{1/\ell}} = 128,
\]
and for every \( t_0 \in I \), it holds that
\[
|p_i(t_0) - q_i(R(t_0))| < 2/M^2 \tau C \|p\| \|q\| = 0.0001254443616, \quad i = 1, 2,
\]
where \( \|p\| = 7.001 \), and \( \|q\| = 1.749810875 \).
5. Conclusion

The paper focuses on the problem of numerical proper reparametrization which has both theoretical and practical background. Based on the results on the symbolic situation (see [18]), we achieve the expected properties and algorithm. For a given numerical curve, we can determine whether it is approximate improper with respect to a given precision and, in the affirmative case, an $\epsilon$-proper reparametrization can be found. More important, the reparameterized curve obtained always lies in the certain offset region of the input one (and reciprocally). As a natural but more difficult problem, we would like to consider the problem of numerical proper reparametrization for rational space curves and surfaces.

References

[1] Abhyankar, S., Bajaj, C., (1988). *Automatic Parametrization of Rational curves and Surfaces III: Algebraic Plane Curves*. Computer Aided Geometric Design. Vol. 5, pp. 321–390.
[2] Beckermann, B., Labahn, G., (1998). *A fast and numerically stable Euclidean-like algorithm for detecting relatively prime numerical polynomials*. Journal of Symbolic Computation. Vol. 26, pp. 691-714.
[3] Beckermann, B., Labahn, G., (1998). *When are two numerical polynomials relatively prime?* Journal of Symbolic Computation. Vol. 26, pp. 677-689.
[4] Chionh E.-W., Gao X.-S., Shen L.-Y. (2006). *Inherently improper surface parametric supports*. Computer Aided Geometric Design. Vol. 23(8), pp. 629-639.
[5] Corless, R.M., Giesbrecht, M.W., Kotsireas, I.S., van Hoeij, M., Watt, S.M. (2001). *Towards Factoring Bivariate Approximate Polynomials*. Proc. ISSAC 2001. pp. 85–92.
[6] Corless, R.M., Watt, S. M., Zhi, L (2004). *QR Factoring to compute the GCD of Univariate Approximate Polynomials*. IEEE Transactions on Signal Processing. Vol. 52(12), 3394-3402.
[7] Cox, D.A., Little, J., O’Shea, D. (1998). *Using Algebraic Geometry*. Graduate Texts in Mathematics, 185. Springer–Verlag.
[8] Farouki, R.T., Rajan, V.T. (1988). *On the numerical condition of algebraic curves and surfaces. 1: Implicit equations*. Computer Aided Geometric Design. Vol. 5, pp. 215-252.
[9] Galligo A., Rupprech D. (2002). *Irreducible decomposition of curves*. Journal of Symbolic Computation. Vol.33, pp. 661C677.
[10] Gao X.-S., Chou S.-C. (1992). *Implicitization of rational parametric equations*. Journal of Symbolic Computation. Vol. 14(5), pp. 459–470.

[11] Kaltofen, E., May, J.P., Yang, Z., Zhi, L. (2008). *Approximate factorization of multivariate polynomials using singular value decomposition*. Journal of Symbolic Computation, Vol.(43)5, pp. 359-376.

[12] Karmarkar, N., Lakshman, Y.N., (1996). *Approximate polynomial greatest common divisors and nearest singular polynomials*. ISSAC 1996, pp. 35-39. ACM Press.

[13] Pérez-Díaz S., Schicho J., Sendra J.R. (2002). *Properness and inversion of rational parametrizations of surfaces*. Appl. Algebra Eng. Commun. Comput. Vol. 13(1), pp. 29–51.

[14] Pérez-Díaz S., Sendra J.R., Sendra J. (2004). *Parametrizations of Approximate Algebraic Curves by Lines*. Theoretical Computer Science on Algebraic - Numeric Algorithms. Vol. 315/2-3, pp. 627-650.

[15] Pérez-Díaz S., Sendra J.R., Sendra J. (2005). *Parametrizations of Approximate Algebraic Surfaces by Lines*. Computer Aided Geometric Design. Vol. 22(2). pp. 147-181.

[16] Pérez-Díaz S., Sendra J.R., Sendra J. (2006). *Distance Bounds of e-Points on Hypersurfaces*. Theoretical Computer Science. Vol 359. N. 1-3. pp. 344-368.

[17] Pérez-Díaz S., Rueda S.L., Sendra J.R., Sendra J. (2010). *Approximate parametrization of plane algebraic curves by linear systems of curves*. Computer Aided Geometric Design. Vol 27. pp. 212-31.

[18] Pérez-Díaz S. (2006). *On the Problem of Proper Reparametrization for Rational Curves and Surfaces*. Computer Aided Geometric Design. Vol. 23(4), pp. 771-795.

[19] Sendra J.R., Villarino C. (2007). *Finite Piecewise Polynomial Parametrization of Plane Rational Algebraic Curves*. Applicable Algebra in Engineering, Communication and Computing. Vol 18(1-2), pp. 91-105.

[20] Sendra J.R., Winkler F. (2001). *Tracing Index of Rational Curve Parametrizations*. Computer Aided Geometric Design. Vol. 3, pp. 67-75.

[21] Shen L.-Y., Chionh E.-W., Gao X.-S., Li J. (2011). *Proper reparametrization for inherently improper unirational varieties*. Journal of Systems Sciences and Complexity. Vol. 24 (2), pp. 367-380.

[22] van Hoeij M., (1994). *Computing Parametrizations of Rational Algebraic Curves*. In J. von zur Gathen (ed) Proc. ISSAC 1994, pp. 187-190.

[23] van der Waerden B.L. (1970). *Algebra I and II*. Springer-Verlag, New York.

[24] van der Waerden B.L. (1973). *Einfüllung in Die Algebraischen Geometrie*. Springer Verlag, Berlin.

[25] Walker R. (1950). *Algebraic Curves*. Princeton University Press.

[26] Winkler, F. (1996). *Polynomials Algorithms in Computer Algebra*. Wien New York: Springer-Verlag.

[27] Zeng, Z. and Dayton, B.H. *The approximate GCD of inexact polynomials part II: a multivariate algorithm*. In Proc. ISSAC 2004. pp. 320-327.