Using differential invariants to detect projective equivalences and symmetries of rational 3D curves

Uğur Gözütok\textsuperscript{a},* \ Hüsni Anıl Çoban\textsuperscript{a}, Yasemin Sağiroğlu\textsuperscript{a}, Juan Gerardo Alcázar\textsuperscript{b}

\textsuperscript{a}Department of Mathematics, Karadeniz Technical University, Turkey
\textsuperscript{b}Departamento de Física y Matemáticas, Universidad de Alcalá, E-28871 Madrid, Spain

Abstract

We present a new approach using differential invariants to detect projective equivalences and symmetries between two rational parametric 3D curves properly parametrized. In order to do this, we introduce two differential invariants that commute with Möbius transformations, which are the transformations in the parameter space associated with the projective equivalences between the curves. The Möbius transformations are found by first computing the gcd of two polynomials built from the differential invariants, and then searching for the Möbius-like factors of this gcd. The projective equivalences themselves are easily computed from the Möbius transformations. In particular, and unlike previous approaches, we avoid solving big polynomial systems. The algorithm has been implemented in Maple\textsuperscript{™} (2021), and evidences of its efficiency as well as a comparison with previous approaches are given.

Keywords: Projective equivalences, projective symmetries, rational curves, differential invariants

1. Introduction.

Detecting projective and affine equivalences implies recognizing whether or not two objects are the same in a certain setup, i.e. up to certain type of deformations. Also, finding the symmetries of an object is important in order to understand its shape, and also to efficiently visualize and store the information regarding the object. For these reasons, these questions have been treated in fields like Computer Vision, Computer Graphics, Computer Aided Geometric Design and Pattern Recognition. Several studies addressing the problem are, for instance, (Bokeloh et al., 2009; Brass and Knauer, 2004; Huang and Cohen, 1996; Lebmeir and Richter-Gebert, 2009; Lebmeir, 2009); a more comprehensive review can be found in (Alcázar et al., 2015).

In recent years several papers (Alcázar, 2014; Alcázar et al., 2014a,b, 2015, 2019a,b; Hauer et al., 2019; Bizzarri et al., 2020a; Hauer and Jüttler, 2018; Bizzarri et al., 2021; Jüttler et al., 2022) have pursued these problems for rational curves and surfaces, using tools from Algebraic Geometry and Computer Algebra. In the case of curves, the main idea behind these approaches is the fact that projective or affine equivalences between the curves, and symmetries as a particular case, have a corresponding transformation in the parameter domain which must be a Möbius transformation whenever the curves are properly, i.e. birationally, parametrized. Thus, the usual approach is to compute the Möbius transformations, and derive the equivalences themselves from there.

For projective equivalences, the algorithms in (Hauer and Jüttler, 2018; Bizzarri et al., 2020b) follow this strategy and compute the Möbius transformations by solving a polynomial system which is increasingly big as the degree of the curves involved in the computation grows. Solving these polynomial system implies using Gröbner bases, which results in higher complexity. In this paper we use a different approach following

*Corresponding author

Email addresses: ugurgozutok@ktu.edu.tr (Uğur Gözütok), hacoban@ktu.edu.tr (Hüsni Anıl Çoban)
yasiroglu@ktu.edu.tr (Yasemin Sağiroğlu), juange.alcazar@uah.es (Juan Gerardo Alcázar)
the idea in (Alcázar et al., 2015), where the classical curvature and torsion, two well-known differential invariants, are used to compute the symmetries of a space rational curve. In (Alcázar et al., 2015) the Möbius transformations are derived as special factors of a gcd of two polynomials, computed from the curvature and torsion functions. On one hand, this has the advantage of working with smaller polynomials, since taking the gcd already reduces the degree of the polynomial one has to analyze. On the other hand, one avoids solving polynomial systems by using factoring instead.

In a similar way, in this paper we present a strategy for 3D space rational curves that also pursues the Möbius transformation first. However, in order to compute them we introduce two differential invariants, which we call projective curvatures, that allow us to obtain the Möbius transformations using an analogous procedure to that in (Alcázar et al., 2015), i.e. using gcd computing and factoring over the reals, without sorting to polynomial system solving. The projective curvatures are constructed by using ideas from differential invariant theory (Olver, 1986; Dolgachev, 2003; Olver, 1995; Mansfield, 2010). To this aim, we first present four differential invariants that completely characterize projective equivalence but that, however, are not well suited for computation, because they do not commute with Möbius transformations. From here, we develop two more differential invariants, the projective curvatures, that do commute with Möbius transformations, and we characterize projective equivalence between the curves using these curvatures. The experimentation carried out in Maple® (2021) shows that our approach is efficient and works better than (Hauer and Jüttler, 2018; Bizzarri et al., 2020b) as the degree of the curves grow.

The structure of the paper is the following. In Section 2 we provide the necessary background on rational curves and differential invariants. The main results behind the algorithm are developed in Section 3, where we introduce several differential invariants to finally derive the projective curvatures, and the theorems relating them to the projective equivalences between the curves. The algorithm itself is provided in Section 4. We present the results of the experimentation carried out in Maple® (2021) in Section 5, where a comparison with the results in (Hauer and Jüttler, 2018; Bizzarri et al., 2020b) is also given. Finally, we close with our conclusion in Section 6. Several technical results and technical proofs are deferred to two appendixes, so as to improve the reading of the paper.

Acknowledgements

The first three authors are partially supported by the project FAY-2021-9648 of Scientific Research Projects Unit, Karadeniz Technical University. The first author would like to thank TUBITAK (The Scientific and Technological Research Council of Turkey) for their financial supports during his doctorate studies. Juan G. Alcázar is supported by the grant PID2020-113192GB-I00 (Mathematical Visualization: Foundations, Algorithms and Applications) from the Spanish MICINN, and is also a member of the Research Group ASYNACS (Ref. ccee2011/r34). Juan G. Alcázar and Uğur Gözütköç were also supported by TUBITAK for a short research visit to Yıldız Technical University in Istanbul (Turkey).

2. Preliminaries.

2.1. General notions and assumptions.

For the sake of comparability, in general we will follow the notation in (Hauer and Jüttler, 2018). Thus, let $C_1$ and $C_2$ be two parametric rational curves embedded in the three real projective space $\bar{\mathbb{P}}^3$. The points $x \in \bar{\mathbb{P}}^3$ are represented by $x = (x_0, x_1, x_2, x_3)^T$, where the $x_i$ are real numbers and correspond to the homogeneous coordinates of $x$. In particular, whenever $\lambda \neq 0$, the vectors $x$ and $\lambda x$ represent the same point in $\bar{\mathbb{P}}^3$. The curves are $C_1$ and $C_2$ defined by means of parametrizations

$$p : P^1(\mathbb{R}) \to C_1 \subset \bar{\mathbb{P}}^3,$$

$$q : P^1(\mathbb{R}) \to C_2 \subset \bar{\mathbb{P}}^3,$$

$$(t_0, t_1) \mapsto p(t_0, t_1) = (p_0(t_0, t_1), p_1(t_0, t_1), p_2(t_0, t_1), p_3(t_0, t_1)),

(t_0, t_1) \mapsto q(t_0, t_1) = (q_0(t_0, t_1), q_1(t_0, t_1), q_2(t_0, t_1), q_3(t_0, t_1)),$$

where $P^1(\mathbb{R})$ denotes the real projective line. The components of each curve are homogeneous polynomials of degree $n$,
\[ p_i(t_0, t_1) = \sum_{j=0}^{n} c_{j,i} t_0^{n-j} t_1^j \text{ and } q_i(t_0, t_1) = \sum_{j=0}^{n} c'_{j,i} t_0^{n-j} t_1^j, \]

with \( i \in \{0, 1, 2, 3\} \), and \( c_{j,i}, c'_{j,i} \in \mathbb{R} \). Additionally, we denote

\[ c_j = (c_{j,0}, c_{j,1}, c_{j,2}, c_{j,3})^T, \quad c'_j = (c'_{j,0}, c'_{j,1}, c'_{j,2}, c'_{j,3})^T, \]

which will be referred to as the \textit{coefficient vectors} of the curves.

Furthermore, we make the following assumptions on \( C_1 \) and \( C_2 \); we will refer later to these hypotheses as \textit{hypotheses (i-iv)}.

(i) The parametrizations \( p \) and \( q \) defining \( C_1 \) and \( C_2 \) are \textit{proper}, i.e. birational, so that almost all points in \( C_i \) are generated by one element of \( P^1(\mathbb{R}) \). It is well-known that every rational curve can be reparametrized to obtain a proper parametrization (see Sendra et al., 2008).

(ii) The parametrizations \( p \) and \( q \) are in reduced form, i.e.,

\[ \gcd(p_0(t_0, t_1), p_1(t_0, t_1), p_2(t_0, t_1), p_3(t_0, t_1)) = \gcd(q_0(t_0, t_1), q_1(t_0, t_1), q_2(t_0, t_1), q_3(t_0, t_1)) = 1. \]

(iii) Both parametrizations \( p \) and \( q \) have the same degree \( n \). Notice that since regular projective transformations preserve the degree, the degree of projectively equivalent curves must be equal. Furthermore, we assume \( n \geq 4 \).

(iv) None of the \( C_i \) is contained in a hyperplane. Consequently, the matrices \((c_{j,k}), (c'_{j,k})\) formed by the coefficient vectors \( c_j \) and \( c'_j \) have rank 4 (Hauer and Jüttler, 2018).

\textbf{Remark 1.} Notice that because of these assumptions, the coefficient vectors \( c_0 \) and \( c'_0 \) cannot be identically zero.

A \textit{projectivity} is a mapping \( f \) defined in \( \mathbb{P}^3 \) such that

\[ f : \mathbb{P}^3 \rightarrow \mathbb{P}^3 : \mathbf{x} \mapsto f(\mathbf{x}) = M \cdot \mathbf{x}, \]

where \( M = (m_{ij})_{0 \leq i,j \leq 3} \) is a non-singular \( 4 \times 4 \) matrix. Note that if \( m_{00} \neq 0 \), \( m_{01} = m_{02} = m_{03} = 0 \), then \( f \) is an affine transformation. Then we have the following definition.

\textbf{Definition 2.} Two curves \( C_1 \) and \( C_2 \) are said to be \textit{projectively equivalent} if there exists a regular projectivity \( f \) such that \( f(C_1) = C_2 \). A curve \( C \) has a \textit{projective symmetry} if there exists a non-trivial regular projectivity \( f \) such that \( f(C) = C \).

It is well-known (Sendra et al., 2008; Hauer and Jüttler, 2018; Bizzarri et al., 2020b) that any two proper parametrizations of a rational curve are related by a linear rational transformation

\[ \varphi : P^1(\mathbb{R}) \rightarrow P^1(\mathbb{R}), \quad (t_0, t_1) \mapsto \varphi(t_0, t_1) = (at_0 + bt_1, ct_0 + dt_1), \]

with \( ad - bc \neq 0 \). The mapping \( \varphi \) is called a \textit{Möbius transformation}. This fact is essential to prove the following result, which is used in (Hauer and Jüttler, 2018; Bizzarri et al., 2020b).

\textbf{Theorem 3.} Two rational curves \( C_1, C_2 \) properly parametrized by \( p \) and \( q \) are projectively equivalent if and only if there exist a non-singular \( 4 \times 4 \) matrix \( M \) and a Möbius transformation \( \varphi(t_0, t_1) = (at_0 + bt_1, ct_0 + dt_1) \) with \( ad - bc \neq 0 \) such that

\[ Mp = q(\varphi). \]
2.2. Differential Invariants.

Classical invariant theory (Kung and Rota, 1984; Olver, 1999; Dolgachev, 2003; Dieudonné and Carrell, 1970) deals with the actions of transformation groups on varieties. Determining functions that do not change, i.e. are invariant, under the action of a given transformation group is at the core of the theory. In more detail, let $G$ be a transformation group, let $V$ be a vector space over $K$ and let $f$ be a function on $V$. We say that $f$ is an absolute invariant with respect to the action of $G$ on $V$ if for all $g \in G$ and $x \in V$,

$$f(g \cdot x) = f(x).$$

Furthermore, we say that $x, y \in V$ are $G$-equivalent if there exists $g \in G$ such that

$$g \cdot x = y.$$

It is clear that if $x, y \in V$ are $G$-equivalent then for any absolute invariant $f$ we have $f(x) = y$. Thus, absolute invariants are useful in order to detect $G$-equivalence. In the same context, we say that $f$ is a relative invariant if

$$f(g \cdot x) = \lambda f(x)$$

with $0 \neq \lambda \in K$. Relative invariants may give rise to absolute invariants: if $f_1, f_2$ are two relative invariants with the same $\lambda$, then $f_1 f_2$ is an absolute invariant. As an example, let $\|w_1 \ldots w_n\|$ denote the determinant of the vectors $w_i \in \mathbb{R}^n$, $i = 1, \ldots, n$, and let $M \in M_{n \times n}$ be a matrix whose determinant $|M|$ is nonzero. It is well-known that

$$\|M w_1 \ldots M w_n\| = |M| \|w_1 \ldots w_n\|. \quad (2)$$

Now if $G$ is the special orthogonal group $SL(n)$, whose elements satisfy that $|M| = 1$, the determinant $\|w_1 \ldots w_n\|$ is an absolute invariant. However, if $G$ is the general linear group $GL(n)$, whose elements verify that $|M| \neq 0$, the determinant $\|w_1 \ldots w_n\|$ is just a relative invariant, although the quotient of two such determinants is an absolute invariant. In the rest of the paper, and unless we specify it, whenever we speak about a differential invariant, it will be understood that it is an absolute differential invariant.

In a similar manner, a differential invariant (Olver, 1986; Dolgachev, 2003; Olver, 1995; Mansfield, 2010), absolute or relative, is an invariant with respect to the action of a Lie group on a space that involves derivatives. For instance, if we set $G$ to be the group of space rigid motions and $V$ the set of regular parametrizations $[x(t), y(t), z(t)]$ of affine space curves, the classical curvature $\kappa$ and torsion $\tau$ are well-known differential invariants in the Euclidean geometry.

In our case, the group $G$ we are interested in is the group of projectivities in $\mathbb{E}^3$, and the vector space $V$ corresponds to the homogeneous parametrizations of curves in $\mathbb{E}^3$. In this setting the notion of $G$-equivalence coincides with that of projective equivalence in Definition 2. Now let $u = u(t_0, t_1), v = v(t_0, t_1)$ be two homogeneous parametrizations such that $u = M \cdot v$, therefore $G$-equivalent, and let us denote

$$u_{k,l}^{t_0,t_1} = \frac{\partial^k}{\partial t_0^k} \frac{\partial^l}{\partial t_1^l} u(t_0, t_1); \quad (3)$$

similarly for $v$. If $u = M \cdot v$, then $u_{k,l}^{t_0,t_1} = M v_{k,l}^{t_0,t_1}$ for any choice $k, l \in \mathbb{Z}^+ \cup \{0\}$. Thus, for instance, because of Eq. (2) the functions

$$\|u_{k}^{t_0}, u_{l}^{t_1}, u_{k}^{t_2}, u_{l}^{t_2}\|, \|u_{k}^{t_0}, u_{l}^{t_1}, u_{k}^{t_2}, u_{l}^{t_2}\| \quad (4)$$

are relative invariants, and their quotient

$$I_1(u) = \|u_{k}^{t_0}, u_{l}^{t_1}, u_{k}^{t_2}, u_{l}^{t_2}\| \quad \|u_{k}^{t_0}, u_{l}^{t_1}, u_{k}^{t_2}, u_{l}^{t_2}\| \quad (5)$$

is an absolute invariant, implying that $I_1(u) = I_1(v)$. If we have several absolute differential invariants $I_i, i = 1, \ldots, m$, and our goal is to check whether or not two given parametrizations $u, v$ are projectively equivalent, then $I_i(u) = I_i(v)$ for $i = 1, \ldots, m$ provide necessary conditions for equivalence. A good choice of invariants can ensure that these conditions are also sufficient, as it happens with the curvature and torsion for the case of rigid motions and curves in space.
3. A new method to detect projective equivalence.

In this section we consider two curves $C_1, C_2$, defined by homogeneous parametrizations $p$ and $q$, satisfying the assumptions in Subsection 2.1.

3.1. Overall strategy.

As in previous approaches, our strategy takes advantage of Theorem 3, and proceeds by first computing the Möbius transformation $\varphi$ in the statement of Theorem 3. If no such transformation is found, the curves $C_1, C_2$ are not projectively equivalent. Otherwise, the matrix $M$ defining the projectivity between $C_1, C_2$ is determined from $\varphi$; we will see that in our case this just amounts to performing a matrix multiplication.

The main ideas in our approach, which we will develop in order later, are the following.

(A) Principal differential invariants. We start with four absolute differential invariants, that we denote $I_i$, $i \in \{1, 2, 3, 4\}$ and refer to as principal invariants. These invariants are defined as quotients of certain determinants, each one a relative invariant itself. The fact that the $I_i$ are absolute invariants follows from the property for determinants shown in Eq. (2). From Theorem 3, if $p, q$ correspond to projectively equivalent curves then $M \cdot p = q \circ \varphi$, with $\varphi$ a Möbius transformation. By the definition of a differential invariant, we have

$$I_i(p) = I_i(q \circ \varphi)$$

for $i \in \{1, 2, 3, 4\}$. These are necessary conditions for $C_1, C_2$ to be projectively equivalent, and will be revealed to be also sufficient.

The equations that stem from Eq. (6) would give rise to a polynomial system in the parameters of $\varphi$. However, this system has a high order, and therefore solving it implies a high computational cost that we want to avoid.

(B) Projective curvatures. In order to avoid solving a big polynomial system, we will derive, from the $I_i$, two more absolute differential invariants $\kappa_1, \kappa_2$ that we will refer to as projective curvatures. Since $\kappa_1, \kappa_2$ are also differential invariants, $\kappa_1, \kappa_2$ do satisfy that

$$\kappa_i(p) = \kappa_i(q \circ \varphi)$$

for $i = 1, 2$. But the advantage of the $\kappa_i$ is that while in general $I_i(q \circ \varphi) \neq I_i(q) \circ \varphi$, the $\kappa_i$ do satisfy $\kappa_i(q \circ \varphi) = \kappa_i(q) \circ \varphi$. By taking together the two relationships in Eq. (7) for $i = 1, 2$ we can find the whole $\varphi$ as a special quadratic factor of the gcd of two polynomials built from the $\kappa_i$. Thus, to compute $\varphi$ we just need gcd computing and factoring, and we do not need to solve any polynomial system. This idea was inspired by the strategy in Alcázar et al. (2015) to compute the symmetries of a rational space curve, where the classical curvature and torsion are used in a similar way.

(C) Projective equivalences. Once $\varphi$ is obtained, the nonsingular matrix $M$ defining the projective equivalence can be computed. This just requires performing matrix multiplications involving the parametrizations $p$ and $q \circ \varphi$.

3.2. Principal Differential Invariants.

In the rest of the paper, we will use the notation $\|w_1, \ldots, w_n\|$ for the determinant of $n$ vectors $w_i \in \mathbb{R}^n$, and the notation $[w_1, \ldots, w_n]$ for the $n \times n$ matrix whose columns are the $w_i$. Additionally, we will use the notation for partial derivatives introduced in Eq. (3).

In order to motivate our principal invariants, we consider first two homogeneous parametrizations $u(t_0, t_1), v(t_0, t_1)$ of curves of degree $n$ in $\mathbb{P}^3$ defining two projectively equivalent curves, such that

$$M \cdot u(t_0, t_1) = v(t_0, t_1)$$

(8)
where $M$ represents a projectivity; notice that because of Theorem 3, what we are pursuing is exactly Eq. (8), with $\mathbf{u} := p$, and $\mathbf{v} := q \circ \varphi$ (where $\varphi$ is unknown). Now let $D(\mathbf{u})(t_0, t_1), D(\mathbf{v})(t_0, t_1)$ be the matrices defined as

$$ D(\mathbf{u}) = [u_{t_0}, u_{t_1}, u_{t_0}^2, u_{t_1}^3], \quad D(\mathbf{v}) = [v_{t_0}, v_{t_1}, v_{t_0}^2, v_{t_1}^3]. $$

(9)

Because of Eq. (8), one can see that $M \cdot D(\mathbf{u}) = D(\mathbf{v})$. Assume that the determinant of $D(\mathbf{u})$ is not identically zero, i.e. $\|u_{t_0} u_{t_1} u_{t_0}^2 u_{t_1}^3\|$ does not vanish identically; later, in see Lemma 4, we will see that this holds in our case. Then $M = D(\mathbf{v})(D(\mathbf{u}))^{-1}$. Differentiating $D(\mathbf{v})(D(\mathbf{u}))^{-1}$ with respect to $t_k$, $k = 0, 1$, we get that

$$ \frac{\partial(D(\mathbf{v}) \cdot (D(\mathbf{u}))^{-1})}{\partial t_k} = \frac{\partial D(\mathbf{v})}{\partial t_k} \cdot (D(\mathbf{u}))^{-1} + D(\mathbf{v}) \cdot \frac{\partial(D(\mathbf{u}))^{-1}}{\partial t_k} $$

$$ = \frac{\partial D(\mathbf{v})}{\partial t_k} \cdot (D(\mathbf{u}))^{-1} - D(\mathbf{v})(D(\mathbf{u}))^{-1} \cdot \frac{\partial D(\mathbf{u})}{\partial t_k} \cdot (D(\mathbf{u}))^{-1} $$

$$ = D(\mathbf{v}) \cdot \left( (D(\mathbf{v}))^{-1} \cdot \frac{\partial D(\mathbf{v})}{\partial t_k} - (D(\mathbf{u}))^{-1} \cdot \frac{\partial D(\mathbf{u})}{\partial t_k} \right) \cdot (D(\mathbf{u}))^{-1}. $$

Since $M = D(\mathbf{v})(D(\mathbf{u}))^{-1}$, we get that the matrices defined by the derivatives at the left-hand side of the above expression are identically zero, and therefore that

$$ (D(\mathbf{u}))^{-1} \cdot \frac{\partial D(\mathbf{u})}{\partial t_k} = (D(\mathbf{v}))^{-1} \cdot \frac{\partial D(\mathbf{v})}{\partial t_k} $$

(10)

for $k = 0, 1$.

To find our principal differential invariants, we need a closer look at the matrices $U_k, V_k$, defined as

$$ U_k = (D(\mathbf{u}))^{-1} \cdot \frac{\partial D(\mathbf{u})}{\partial t_k}, \quad V_k = (D(\mathbf{v}))^{-1} \cdot \frac{\partial D(\mathbf{v})}{\partial t_k}. $$

(11)

3.2.1. Demonstrating $U_k = V_k$.

In this subsection, let us show $U_k = V_k$ for $k = 0, 1$.

First let us show that for all $k = 0, 1$,

$$ (D(\mathbf{u}))^{-1} \cdot \frac{\partial D(\mathbf{u})}{\partial t_k} = (D(\mathbf{v}))^{-1} \cdot \frac{\partial D(\mathbf{v})}{\partial t_k}, $$

(12)

where $(D(\mathbf{u}))^{-1}$ denotes the inverse matrix of $D(\mathbf{u})$.

Let $(D(\mathbf{u}))^{-1} \cdot \frac{\partial D(\mathbf{u})}{\partial t_k} = U_k$ for an unknown matrix $U_k$, $k = 0, 1$. Then $D(\mathbf{u}) \cdot U_k = \frac{\partial D(\mathbf{u})}{\partial t_k}$. Denote the $j$th column of the matrix $D(\mathbf{u})$ by $D(\mathbf{u})_j$, and similarly denote the $j$th column of the matrix $U_k$ by $U^k_j$ for $1 \leq j \leq 4$. So we have 8 systems each of which corresponds to a pair of the values $j$ and $k$, namely

$$ D(\mathbf{u}) \cdot U^k_j = \frac{\partial D(\mathbf{u})_j}{\partial t_k}, $$

(13)

for $k = 0, 1$ and $1 \leq j \leq 4$. The system corresponding to each pair is linear in the components of $U^k_j$. On the other hand, since the coefficient matrix $D(\mathbf{u})$ of each system is non singular ($\Delta(\mathbf{u}) \neq 0$), each system
has only one solution. The solution to the systems are

\[
U^j_k = \begin{bmatrix}
\| \partial D(u)_j \|_{u^1_0 u^2_0 u^3_0} \\
\| u^0_0 u^1_0 u^2_0 u^3_0 \| \\
\| u^0_0 \partial D(u)_j \|_{u^1_0 u^2_0 u^3_0} \\
\| u^0_0 u^1_0 \partial D(u)_j \|_{u^2_0 u^3_0} \\
\| u^0_0 u^1_0 u^3_0 \|_{u^2_0 u^3_0} \\
\| u^0_0 u^1_0 \partial D(u)_j \|_{u^3_0} \\
\| u^0_0 u^1_0 u^2_0 \|_{u^3_0} \\
\| u^0_0 u^1_0 u^2_0 u^3_0 \|
\end{bmatrix}. 
\]  

(14)

Using Euler’s homogeneous function theorem, we conclude that for \( k = 0 \) and \( j < 4 \),

\[
U^1_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \quad U^2_0 = \begin{bmatrix} n^{-1} \\ -\frac{t_1}{t_0} \\ 0 \\ 0 \\ 1 \end{bmatrix}, \quad U^3_0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}.
\]

(15)

and for \( k = 1 \) and \( j < 4 \),

\[
U^1_1 = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \quad U^2_1 = \begin{bmatrix} -\frac{(n-1)t_0}{t_1} \\ \frac{n-1}{t_0} \\ \frac{t_0}{t_1} \\ 0 \\ 1 \end{bmatrix}, \quad U^3_1 = \begin{bmatrix} 0 \\ 0 \\ n^{-1} \\ -\frac{t_1}{t_0} \\ \frac{t_1}{t_0} \end{bmatrix}.
\]

(16)

It is seen that for \( k = 0, 1 \) and \( j < 4 \), \( U^j_k \) do not depend on \( u \). Now for \( k = 0 \) and \( j = 4 \), we have \( \frac{\partial D(u)_4}{\partial t_0} = \frac{\partial u^3_0}{\partial t_0} = u^3_0 \). It is obtained

\[
U^4_0 = \begin{bmatrix}
\| u^0_0 u^1_0 u^2_0 u^3_0 \| \\
\| u^0_0 u^1_0 u^2_0 u^3_0 \| \\
\| u^0_0 u^1_0 u^2_0 u^3_0 \| \\
\| u^0_0 u^1_0 u^2_0 u^3_0 \| \\
\| u^0_0 u^1_0 u^2_0 u^3_0 \| \\
\| u^0_0 u^1_0 u^2_0 u^3_0 \| \\
\| u^0_0 u^1_0 u^2_0 u^3_0 \|
\end{bmatrix}.
\]

(17)
Similarly, for \( k = 1 \) and \( j = 4 \), we have \[
\frac{\partial D(u)}{\partial t_4} = \frac{\partial u_{t, j}^4}{\partial t_1} = \frac{u_{t, j}^4 - u_{t, j}^3}{t_1} = u_{t, j}^4 - \frac{u_{t, j}^3}{t_1} = \frac{n - 3}{t_1} u_{t, j}^3 - \frac{t_0}{t_1} u_{t, j}^3.
\]
It is obtained
\[
U_1^4 = \begin{bmatrix}
\left\| \frac{n - 3}{t_1} u_{t, j}^3 - \frac{t_0}{t_1} u_{t, j}^3 \right\| \\
\left\| \frac{u_{t, j}^3}{t_1} - \frac{t_0}{t_1} u_{t, j}^3 \right\| \\
\left\| \frac{u_{t, j}^3}{t_1} - \frac{u_{t, j}^3}{t_1} \right\| \\
\left\| u_{t, j}^3 - \frac{u_{t, j}^3}{t_1} \right\| \\
\left\| u_{t, j}^3 - \frac{u_{t, j}^3}{t_1} \right\|
\end{bmatrix} = \begin{bmatrix}
-\frac{t_0}{t_1} \left\| u_{t, j}^4 - u_{t, j}^3 \right\| \\
-\frac{t_0}{t_1} \left\| u_{t, j}^4 - u_{t, j}^3 \right\| \\
-\frac{t_0}{t_1} \left\| u_{t, j}^4 - u_{t, j}^3 \right\| \\
-\frac{t_0}{t_1} \left\| u_{t, j}^4 - u_{t, j}^3 \right\| \\
-\frac{t_0}{t_1} \left\| u_{t, j}^4 - u_{t, j}^3 \right\|
\end{bmatrix}.
\] (18)

Now let \((D(v))^{-1} \cdot \frac{\partial D(v)}{\partial k} = V_k\) for an unknown matrix \(V_k\), \(k = 0, 1\). Then \(D(v) \cdot V_k = \frac{\partial D(v)}{\partial k}\). Denote the \(j\)th column of the matrix \(D(v)\) by \(D(v)_j\), and similarly denote the \(j\)th column of the matrix \(V_k\) by \(V_k^j\) for \(1 \leq j \leq 4\). Similarly, we again have 8 systems each of which corresponds to a pair of the values \(j\) and \(k\) for \(v\). For the solutions \(V_k^j\) we have \(U_{k, j}^4 = V_k^j\) for all \(k\) and \(j < 4\), since \(U_{k, j}^4\) and \(V_k^j\) do not depend on the parametrizations. In addition, similar operations lead to

\[
V_0^4 = \begin{bmatrix}
\left\| v_{t, j}^4 - v_{t, j}^3 \right\| \\
\left\| v_{t, j}^4 - v_{t, j}^3 \right\| \\
\left\| v_{t, j}^4 - v_{t, j}^3 \right\| \\
\left\| v_{t, j}^4 - v_{t, j}^3 \right\| \\
\left\| v_{t, j}^4 - v_{t, j}^3 \right\|
\end{bmatrix}
\] (19)

and

\[
V_1^4 = \begin{bmatrix}
\left\| \frac{v_{t, j}^4}{t_1} - \frac{v_{t, j}^3}{t_1} \right\| \\
\left\| \frac{v_{t, j}^4}{t_1} - \frac{v_{t, j}^3}{t_1} \right\| \\
\left\| \frac{v_{t, j}^4}{t_1} - \frac{v_{t, j}^3}{t_1} \right\| \\
\left\| \frac{v_{t, j}^4}{t_1} - \frac{v_{t, j}^3}{t_1} \right\| \\
\left\| \frac{v_{t, j}^4}{t_1} - \frac{v_{t, j}^3}{t_1} \right\|
\end{bmatrix}.
\] (20)
we call projective curvatures $n^{(i-iv)}$ in Subsection 2.1. Then

$\Delta(\cdot)$

Lemma 4. Theorem 5.

2.1. Then we have the following result, related to the previous discussion.

\[ I_1(u) := \frac{A_1(u)}{\Delta(u)}, I_2(u) := \frac{A_2(u)}{\Delta(u)}, I_3(u) := \frac{A_3(u)}{\Delta(u)}, I_4(u) := \frac{A_4(u)}{\Delta(u)}. \]

(21)

Then the following four rational expressions, which are the expressions arising in Eq. (??), will be referred to, from now on, as the principal invariants:

\[ A_1(u) := \|u_{i_0} \; u_{i_1} \; u_{i_2} \; u_{i_3}\|, A_2(u) := \|u_{i_0} \; u_{i_1} \; u_{i_2} \; u_{i_3}\|, A_3(u) := \|u_{i_0} \; u_{i_1} \; u_{i_2} \; u_{i_3}\|, A_4(u) := \|u_{i_0} \; u_{i_1} \; u_{i_2} \; u_{i_3}\|. \]

(22)

The following lemma, proven in Appendix A, guarantees that under our hypotheses the principal invariants are well-defined.

Lemma 4. Let $C$ in $\mathbb{P}^3$ be a rational algebraic curve properly parametrized by $u$, satisfying the hypotheses (i-iv) in Subsection 2.1. Then $\Delta(u)$ is not identically zero.

Now let us go back to our curves $C_1, C_2$, defined by homogeneous parametrizations $p$ and $q$ of degree $n \geq 4$, as in Subsection 2.1. We recall that we are assuming that $p, q$ satisfy hypotheses (i-iv) in Subsection 2.1. Then we have the following result, related to the previous discussion.

Theorem 5. Let $C_1, C_2$ be two rational algebraic curves properly parametrized by $p, q$ satisfying hypotheses (i-iv). Then $C_1, C_2$ are projectively equivalent if and only if there exists a M"obius transformation $\varphi$ such that

\[ I_i(p) = I_i(q \circ \varphi) \]

(23)

for $i \in \{1, 2, 3, 4\}$.

Proof. The implication ($\Rightarrow$) follows from Theorem 3 and the discussion at the beginning of this subsection. So let us focus on ($\Leftarrow$). Let $u := p, v := q \circ \varphi$. Since by hypothesis $I_i(u) = I_i(v)$ for $i = 1, 2, 3, 4$, taking Eq. (17)-(20) into account we have $U_k = V_k$ for $k = 0, 1$, so Eq. (10) holds for $k = 0, 1$; notice that since the determinants of $D(u), D(v)$ are, precisely, $\Delta(u), \Delta(v)$, by Lemma 4 the inverses $D(u)^{-1}, D(v)^{-1}$ exist. Therefore, the matrix $D(v) \cdot (D(u))^{-1}$ is a constant nonsingular matrix $M$. Thus, $M \cdot D(u) = D(v)$, so $M \cdot u_{t_0} = v_{t_0}$ and $M \cdot u_{t_1} = v_{t_1}$. Using Euler’s Homogeneous Function Theorem, we have

\[ n v = t_0 v_{t_0} + t_1 v_{t_1} = t_0 M \cdot v_{t_0} + t_1 M \cdot v_{t_1} = M \cdot (t_0 u_{t_0} + t_1 u_{t_1}) = n M \cdot u, \]

so $M \cdot u = v$. □

The relationships in Eq. (23) lead to a polynomial system in the parameters of the M"obius transformation $\varphi$. However, this system has a high degree. Because of this, we will derive other differential invariants, that we call projective curvatures. This is done in the next subsection.
3.3. Projective curvatures.

A first question when examining the relationships in Eq. (23) is how the \( I_i(q) \) change when \( q \) is composed with \( \varphi \). Writing \( \varphi(t_0, t_1) = (a t_0 + b t_1, c t_0 + d t_1) = (u, v) \), calling \( \delta = ad - bc \neq 0 \) and using the Chain Rule, we get that

\[
v^4 I_1(q \circ \varphi) = c^3(n-1)(n-2)(n-3)(3v + dt_1) + c^2(n-1)(n-2)\delta t_1(2v + dt_1) I_4(q \circ \varphi) - c(n-1)\delta^2 t_1^2(v + dt_1) I_4(q \circ \varphi) + \delta t_1^4(dI_1(q) \circ \varphi - b I_2(q) \circ \varphi)
\]

(24)

\[
v^4 I_2(q \circ \varphi) = -c^4(n-1)(n-2)(n-3)t_1 - c^3(n-1)(n-2)\delta t_1^2 I_4(q) \circ \varphi + c^2(n-1)\delta^2 t_1^2 I_4(q) \circ \varphi + \delta^3 t_1^4(a I_2(q) \circ \varphi - c I_1(q) \circ \varphi)
\]

(25)

\[
v^2 I_3(q \circ \varphi) = -6c^2(n-2)(n-3) - 3c(n-2)\delta t_1 I_4(q) \circ \varphi + \delta^2 t_1^2 I_3(q) \circ \varphi
\]

(26)

\[
v I_4(q \circ \varphi) = 4c(n-3) + \delta t_1 I_4(q) \circ \varphi.
\]

(27)

From these expressions we see that the \( I_i \) do not commute with \( \varphi \), i.e. in general \( I_i(q \circ \varphi) \neq I_i(q) \circ \varphi \). We aim to find invariants that do commute with \( \varphi \). In order to do that, we substitute the equations (24)-(27) into the relationships in Eq. (23), and we get

\[
v^4 I_1(p)(t_0, t_1) = c^3(n-1)(n-2)(n-3)(3v + dt_1) + c^2(n-1)(n-2)\delta t_1(2v + dt_1) I_4(q(u, v)) - c(n-1)\delta^2 t_1^2(v + dt_1) I_4(q(u, v)) + \delta t_1^4(dI_1(q) \circ \varphi - b I_2(q) \circ \varphi)
\]

(28)

\[
v^4 I_2(p)(t_0, t_1) = -c^4(n-1)(n-2)(n-3)t_1 - c^3(n-1)(n-2)\delta t_1^2 I_4(q(u, v)) + c^2(n-1)\delta^2 t_1^2 I_4(q(u, v)) + \delta^3 t_1^4(a I_2(q) \circ \varphi - c I_1(q) \circ \varphi)
\]

(29)

\[
v^2 I_3(p)(t_0, t_1) = -6c^2(n-2)(n-3) - 3c(n-2)\delta t_1 I_4(q(u, v)) + \delta^2 t_1^2 I_3(q(u, v))
\]

(30)

\[
v I_4(p)(t_0, t_1) = 4c(n-3) + \delta t_1 I_4(q(u, v)).
\]

(31)

We want to eliminate the parameters \( a, b, c, d \) from these equations, something that we can interpret in terms of polynomial ideals. Indeed, let us write \( J_i = I_i(q)(u, v) \) for \( i \in \{1, 2, 3, 4\} \), and let us denote the \( I_i(p) \) by just \( I_i \). Then after clearing denominators, the equations (28)-(31) generate a polynomial ideal \( \mathcal{I} \) of

\[\mathbb{R}[a, b, c, d, t_1, v, I_1, I_2, I_3, I_4, J_1, J_2, J_3, J_4].\]

Eliminating \( a, b, c, d \) from equations (28)-(31) amounts to finding elements in the elimination ideal

\[\mathcal{I}^* = \mathcal{I} \cap \mathbb{R}[t_1, v, I_1, I_2, I_3, I_4, J_1, J_2, J_3, J_4].\]

3.3.1. Eliminating variables.

In our case, this can be done by hand, without using Gröbner bases; the process consists of several easy, but lengthy, following substitutions and manipulations.

Now we are ready to eliminate the coefficients \( a, b, c, d \) from the above system to find a system such that \( \{ k_i(p) = k_i(q)(u, v), k_i(p) = k_i(q)(u, v) \} \). We first try to eliminate \( a, b, d \) from (28) and (29), then multiplying (28) by \( t_0 \) and (29) by \( -t_1 \) and summing the results, we have

\[
v^4 I_0(p)(t_0, t_1) = 4 c^3(n-1)(n-2)(n-3)t_1v + 3c^2(n-1)(n-2)\delta t_1^2 v I_4(q)(u, v) - 2c(n-1)\delta^2 t_1^2 v I_3(q)(u, v) + \delta^3 t_1^4 I_0(q)(u, v),
\]

(32)

where \( I_0(p)(t_0, t_1) = t_1 I_1(p)(t_0, t_1) - t_0 I_2(p)(t_0, t_1) \).

Again to eliminate \( a \) from (29), we use \( av - cu = \delta \). Substituting \( a = \frac{\delta + cu}{v} \) in (29), it is obtained
\[v^5 I_2(p)(t_0, t_1) = -c^4(n-1)(n-2)(n-3)t_1 v - c^3(n-1)(n-2)\delta t^2_1 v I_4(q)(u, v) + c^2(n-1)\delta^2 t^4_1 v J_3(q)(u, v) - c\delta^3 t^4_1 I_0(q)(u, v) + \delta^4 t^5_1 I_2(q)(u, v).\] (33)

We know that the coefficients of the Möbius transformation satisfy \(\delta = ad - bc \neq 0\). Then there is a number \(s \neq 0\) such that \(\delta s = 1\). Now we use this in the equations (28)-(31) in order to eliminate the parameters \(\delta, c\). Thus multiplying the equations (31), (30), (32), (33) by \(s, s^2, s^3, s^4\), respectively, we have

\[se I_4(p)(t_0, t_1) = 4(cs)(n-3) + t_1 J_4(q)(u, v)\] (34)
\[s^2 e^2 I_3(p)(t_0, t_1) = -6(cs)^2(n-2)(n-3) - 3(cs)(n-2)t_1 J_4(q)(u, v) + t^2_1 I_2(q)(u, v)\] (35)
\[s^3 e^4 I_0(p)(t_0, t_1) = 4(cs)^3(n-1)(n-2)(n-3)t_1 v + 3(cs)^2(n-1)(n-2)t^2_1 v I_4(q)(u, v) - 2(cs)(n-1)t^2_1 v J_3(q)(u, v) + t^4_1 I_0(q)(u, v)\] (36)
\[s^4 e^6 I_2(p)(t_0, t_1) = -c^4(n-1)(n-2)(n-3)t_1 v - (cs)^3(n-1)(n-2)t^2_1 v I_4(q)(u, v) + (cs)^2(n-1)t^2_1 v J_3(q)(u, v) - (cs)t^4_1 I_0(q)(u, v) + t^5_1 I_2(q)(u, v).\] (37)

From now on unless otherwise stated explicitly, we denote \(I_1(q)(u, v)\) by \(J_1(q)\) and drop \((t_0, t_1)\) from the function \(I_1(p)(t_0, t_1)\) for the sake of shortening of the equations. We are ready to eliminate \(c\) from the above equations. Let us get \(cs\) from first equation and write as

\[cs = \frac{se I_4(p) - t_1 J_4(q)}{4(n-3)}.\]

Substituting \(cs\) in the equations (35), (36), (37) and using Theorem 17, we have

\[s^2 = \frac{t^2_1 (8(n-3) J_3(q) + 3(n-2) J^2_2(q))}{v^5 (8(n-3) I_3(p) + 3(n-2) I^2_2(p))},\] (38)
\[s^3 = \frac{t^3_1 (8(n-3)^2 J_0(q) + 4(n-1)(n-3) t_1 J_3(q) J_4(q) + (n-1)(n-2) t_1 J^2_1(q))}{v^6 (8(n-3)^2 J_0(p) + 4(n-1)(n-3) t_1 J_3(p) J_4(p) + (n-1)(n-2) t_1 J^2_1(p))},\] (39)
\[s^4 = \frac{t^4_1 (256(n-3)^3 J_2(q) + 64(n-3)^2 J_0(q) J_4(q) + 16(n-1)(n-3) t_1 J_3(q) J^2_2(q) + 3(n-1)(n-2) t_1 J^3_1(q))}{v^8 (256(n-3)^3 J_2(p) + 64(n-3)^2 J_0(p) J_4(p) + 16(n-1)(n-3) t_1 J_3(p) J^2_2(p) + 3(n-1)(n-2) t_1 J^3_1(p))},\] (40)

respectively.

Eliminating \(s\) by equating the cube of (38) and the square of (39) it is obtained

\[\frac{(8(n-3)^2 J_0(p) + 4(n-1)(n-3) t_1 J_3(p) J_4(p) + (n-1)(n-2) t_1 J^2_1(p))^2}{t^4_1 (8(n-3) J_3(p) + 3(n-2) J^2_2(p))^3} = \frac{(8(n-3) J_0(q) + 4(n-1)(n-3) t_1 J_3(q) J_4(q) + (n-1)(n-2) t_1 J^2_1(q))^2}{v^2 (8(n-3) J_3(q) + 3(n-2) J^2_2(q))^3}.\] (41)

And by equating the square of (38), and (40) it is obtained

\[\frac{256(n-3)^3 I_2(p) + 64(n-3)^2 J_0(p) J_4(p) + 16(n-1)(n-3) t_1 J_3(p) J^2_2(p) + 3(n-1)(n-2) t_1 J^3_1(p)}{t^4_1 (8(n-3) J_3(p) + 3(n-2) J^2_2(p))^2} = \frac{256(n-3)^3 J_2(q) + 64(n-3)^2 J_0(q) J_4(q) + 16(n-1)(n-3) t_1 J_3(q) J^2_2(q) + 3(n-1)(n-2) t_1 J^3_1(q)}{v (8(n-3) J_3(q) + 3(n-2) J^2_2(q))^2}.\] (42)
Eventually we get
\[
\frac{(8n-3)^2I_0(p) + 4(n-1)(n-3)t_1I_3(p)I_4(p) + (n-1)(n-2)t_1I_2^2(p))}{t_1^2(8n-3)I_3(p) + 3(n-2)I_2^2(p))}\]
\[= \frac{(8n-3)J_0(q) + 4(n-1)(n-3)t_1J_3(q)J_4(q) + (n-1)(n-2)t_1J_2^2(q))}{v^2(8n-3)J_3(q) + 3(n-2)J_2^2(q))},
\]
(43)
and
\[
\frac{256(n-3)^3I_2(p) + 64(n-3)^2I_0(p)I_4(p) + 16(n-1)(n-3)t_1I_3(p)I_2^2(p) + 3(n-1)(n-2)t_1I_2^4(p)}{t_1(8n-3)I_3(p) + 3(n-2)I_2^2(p))}\]
\[= \frac{256(n-3)^3J_2(q) + 64(n-3)^2J_0(q)J_4(q) + 16(n-1)(n-3)t_1J_3(q)J_2^2(q) + 3(n-1)(n-2)t_1J_2^4(q)}{v(8n-3)J_3(q) + 3(n-2)J_2^2(q))},
\]
(44)
where
\[I_0(p) = t_1I_1(p) - t_0I_2(p), \quad J_0(q) = vJ_1(q) - uJ_2(q).
\]
Notice that Eq. (43) and (44) have a very special structure: if we examine the right-hand side and the left-hand side of each of these equations, we detect the same function but evaluated at \((t_0, t_1)\), at the left, and at \((u, v)\), at the right, where \(u, v\) are the components of the M"obius function. This motivates our definition of the following two functions, that we call projective curvatures:

\[
\kappa_1(p) = \frac{(8n-3)^2I_0(p) + 4(n-1)(n-3)t_1I_3(p)I_4(p) + (n-1)(n-2)t_1I_2^2(p))}{t_1^2(8n-3)I_3(p) + 3(n-2)I_2^2(p))}
\]
\[\kappa_2(p) = \frac{256(n-3)^3I_2(p) + 64(n-3)^2I_0(p)I_4(p) + 16(n-1)(n-3)t_1I_3(p)I_2^2(p) + 3(n-1)(n-2)t_1I_2^4(p)}{t_1(8n-3)I_3(p) + 3(n-2)I_2^2(p))},
\]
(45)

**Remark 6.** Notice that there are additional possibilities for projective curvatures, other than \(\kappa_1, \kappa_2\) in Eq. (45). What we really want are elements in the ideal \(I'\) which correspond to the subtraction of the evaluations of a certain rational function at \(t_1, I_1, I_2, I_3, I_4\) and at \(v, J_1, J_2, J_3, J_4\), respectively. We do not have yet a complete theoretical explanation of why the ideal \(I'\) contains such elements. This probably requires further look into the theory of differential invariants.

The next result follows directly from Eq. (43) and (44).

**Lemma 7.** Let \(C\) be a rational algebraic curve properly parametrized by \(p\) satisfying hypotheses (i-iv) and let \(\varphi(t_0, t_1) = (a_{t_1} + t_1c_0 + dt_1)\) be a M"obius transformation with \(ad - bc \neq 0\). The following equalities hold.

i. \(\kappa_1(p \circ \varphi) = \kappa_1(p) \circ \varphi\),

ii. \(\kappa_2(p \circ \varphi) = \kappa_2(p) \circ \varphi\).

The fact that \(\kappa_1, \kappa_2\) are well defined follows from the following result, which is proven in Appendix B. In fact, in Appendix B we prove a stronger result which implies this lemma, namely that the invariants \(I_i\), \(i \in \{1, 2, 3, 4\}\), are algebraically independent.

**Lemma 8.** The denominators in \(\kappa_1, \kappa_2\) do not identically vanish, and therefore \(\kappa_1, \kappa_2\) are well defined.

Now we are ready to present our main result, that characterizes the projective equivalences of rational 3D curves in terms of the rational invariant functions \(\kappa_1\) and \(\kappa_2\).

**Theorem 9.** Let \(C_1, C_2\) be two rational algebraic curves properly parametrized by \(p, q\) satisfying hypotheses (i-iv). Then \(C_1, C_2\) are projectively equivalent if and only if there exists a M"obius transformation \(\varphi(t_0, t_1) = (a_{t_1} + t_1c_0 + dt_1) = (u, v)\) satisfying the following equations

\[
\kappa_1(p)(t_0, t_1) = \kappa_1(q)(u, v)
\]
\[\kappa_2(p)(t_0, t_1) = \kappa_2(q)(u, v),
\]
(46)
and such that \(D(q \circ \varphi)(D(p))^{-1}\) is a constant matrix \(M\). Furthermore, \(f(x) = M \cdot x\) is a projective equivalence between \(C_1, C_2\).
constant, then $\kappa_1$ from the discussion at the beginning of Subsection 3.2, $M = D(q \circ \varphi)(D(p))^{-1}$. By Theorem 5, $I_i(p) = I_i(q \circ \varphi)$ for $i \in \{1, 2, 3, 4\}$, and therefore Eq. (43) and (44) hold. The rest follows from the definition of $\kappa_1, \kappa_2$. $(\Leftarrow)$ From the proof of the implication $\Leftarrow$ in Theorem 5, if $D(q \circ \varphi)(D(p))^{-1} = M$, with $M$ constant, then $M \cdot p = q \circ \varphi$, so $f(x) = M \cdot x$ is a projective equivalence between $C_1, C_2$. \hfill \Box

4. The algorithm.

In this section we will see how to turn the result in Theorem 9 into an algorithm to detect projective equivalence. In order to do this, first we write

$$\begin{align*}
\kappa_1(p)(t_0, t_1) &= \frac{U(t_0, t_1)}{V(t_0, t_1)} \\
\kappa_2(p)(t_0, t_1) &= \frac{Y(t_0, t_1)}{Z(t_0, t_1)}, \\
\kappa_1(q)(t_0, t_1) &= \frac{\bar{U}(t_0, t_1)}{\bar{V}(t_0, t_1)} \\
\kappa_2(q)(t_0, t_1) &= \frac{\bar{Y}(t_0, t_1)}{\bar{Z}(t_0, t_1)},
\end{align*}$$

(48)

where $U, V, Y, Z$ and $\bar{U}, \bar{V}, \bar{Y}, \bar{Z}$ are homogeneous polynomials such that $\gcd(U, V) = 1$, $\gcd(Y, Z) = 1$, $\gcd(\bar{U}, \bar{V}) = 1$ and $\gcd(\bar{Y}, \bar{Z}) = 1$. From Theorem 9 we know that if the curves are projectively equivalent, then

$$\kappa_1(p)(t_0, t_1) - \kappa_1(q)(u, v) = 0, \quad \kappa_2(p)(t_0, t_1) - \kappa_2(q)(u, v) = 0$$

(50)

where $\varphi(t_0, t_1) = (at_0 + bt_1, ct_0 + dt_1) = (u, v)$. Clearing the denominators of these equations, we define two homogeneous polynomials $E_1$ and $E_2$ in $t_0, t_1, u, v$,

$$\begin{align*}
E_1(t_0, t_1, u, v) &= U(t_0, t_1)\bar{V}(u, v) - V(t_0, t_1)\bar{U}(u, v) \\
E_2(t_0, t_1, u, v) &= Y(t_0, t_1)\bar{Z}(u, v) - Z(t_0, t_1)\bar{Y}(u, v).
\end{align*}$$

(51)

We are interested in the common factors of $E_1$ and $E_2$. Thus, let us write

$$G(t_0, t_1, u, v) := \gcd(E_1(t_0, t_1, u, v), E_2(t_0, t_1, u, v)).$$

(53)

Finally, for an arbitrary Möbius transformation $\varphi(t_0, t_1) = (at_0 + bt_1, ct_0 + dt_1) = (u, v)$, $ad - bc \neq 0$, we say that

$$F(t_0, t_1, u, v) = u(ct_0 + dt_1) - v(at_0 + bt_1)$$

(54)

is the associated Möbius-like factor. Notice that the condition $ad - bc \neq 0$ guarantees that $F$ is irreducible.

Then we have the following result, which follows from Bezout’s Theorem.

**Theorem 10.** Let $C_1, C_2$ be two rational algebraic curves properly parametrized by $p, q$ satisfying hypotheses (i-in), and let $G$ be as in Eq. (53). If $C_1$ and $C_2$ are projectively equivalent then there exists a Möbius-like factor $F$ such that $F$ divides $G$.

Thus, in order to compute the Möbius transformation $\varphi$, we just need to compute the polynomial $G(t_0, t_1, u, v)$ in Eq. (53), factor it, and look for the Möbius-like factors. In general we need to factor over the reals, which can be efficiently done with the command `Factors` in Maple™ (2021). Once the $\varphi$ are found, we check whether or not $D(q \circ \varphi)(D(p))^{-1}$ is constant in the affirmative case, $M = D(q \circ \varphi)(D(p))^{-1}$ defines a projectivity between the curves. For this last part, it is computationally cheaper to compute $D((q \circ \varphi)(t_0))(D(p(t_0)))^{-1}$ for some $t_0 \in \mathbb{R}$, and then check whether or not $M p = q \circ \varphi$ holds.

Therefore, we get the following algorithm, Prj3D, to check whether or not two given rational curves are projectively equivalent. In order to execute the algorithm, we need that not both $\kappa_1, \kappa_2$ are constant. We conjecture that the space curves with both $\kappa_1$ constant may be related to $W$-curves (Sasaki, 1936), but at this point we must leave this case out of our study.
The projective curvatures are, in this case, the computation of $\kappa_1, \kappa_2$ are constant.

**Output:** Either the list of Möbius transformations and projectivities, or the warning: “The curves are not projectively equivalent.”

**Example 11.** Consider the curves given by the rational parametrizations

$$ p(t_0, t_1) = \left( \frac{(t_0 - t_1)^4 + 16t_0^4 - 8t_0^2(t_0 - t_1) + 4t_0^3(t_0 - t_1)^2}{2t_0(t_0 - t_1)((t_0 - t_1)^2 + 4t_0^3)} \right), \quad q(t_0, t_1) = \left( \frac{(t_0 - t_1)^4 + 16t_0^4}{2(t_0 - t_1)^3 t_0^2} \right). $$

The projective curvatures are, in this case,

$$ \kappa_1(p)(t_0, t_1) = \frac{(17t_0^4 - 4t_0^3 t_1 + 6t_0^2 t_1^2 - 4t_0 t_1^3 + t_1^4)^2}{384t_0^2 (t_0 - t_1)^2 (t_0^3 - 2t_0 t_1 + t_1^2)} $$

$$ \kappa_2(p)(t_0, t_1) = \frac{273a^8 - 72a^7 t_1 + 124a^6 t_1^2 - 120a^5 t_1^3 + 86a^4 t_1^4 - 56a^3 t_1^5 + 28a^2 t_1^6 - 8a t_1^7 + t_1^8}{96 (t_0^2 - 2t_0 t_1 + t_1^2) t_0^4} $$

Thus we get

$$ E_1 = 384 \left( 17t_0^4 - 4t_0^3 t_1 + 6t_0^2 t_1^2 - 4t_0 t_1^3 + t_1^4 \right)^2 \left( u^2 - 4u^3 + 6u^2 v - 4u v^3 + v^4 \right) $$

$$ E_2 = 96 \left( 17t_0^4 - 4t_0^3 t_1 + 6t_0^2 t_1^2 - 4t_0 t_1^3 + t_1^4 \right)^2 \left( u^2 - 4u^3 + 6u^2 v - 4u v^3 + v^4 \right) $$

$$ (u^2 - 2uv + v^2)^2 u^4 - 96 \left( t_0^2 - 2t_0 t_1 + t_1^2 \right)^2 t_0^4 \left( t_0^2 - 2t_0 t_1 + t_1^2 \right)^2 t_0^4 $$

$$ (273a^8 - 72a^7 t_1 + 124a^6 t_1^2 - 120a^5 t_1^3 + 86a^4 t_1^4 - 56a^3 t_1^5 + 28a^2 t_1^6 - 8a t_1^7 + t_1^8) $$

The computation of $G = \gcd(E_1, E_2)$ yields

$$ G(t_0, t_1, u, v) = (t_0^2 - ut_1) (3t_0 u + t_0 v + ut_1 - t_1 v) (5t_0 u - t_0 v - ut_1 + t_1 v) (2t_0 u - t_0 v - ut_1) $$

$$ (2t_0 u - 2t_0^2 uv + t_0^2 v^2 - 2u^2 t_0 t_1 + u^2 t_1^2) $$

$$ (17t_0^2 u^2 - 2t_0^2 uv + t_0^2 v^2 - 2u^2 t_0 t_1 + 4t_0 t_1 uv - 2t_0 t_1 v^2 + u^2 t_1^2 - 2t_1^2 uv + t_1^2 v^2) $$

Factoring $G$, we get the following Möbius-like factors:
\[ f_1 = t_0 u - \frac{1}{2} t_0 v - \frac{1}{2} t_1 u \]
\[ f_2 = t_0 u + \frac{1}{3} t_0 v + \frac{1}{3} t_1 u - \frac{1}{3} t_1 v \]
\[ f_3 = t_0 v - t_1 u \]
\[ f_4 = t_0 u - \frac{1}{5} t_0 v - \frac{1}{5} t_1 u + \frac{1}{5} t_1 v, \]

which correspond to the following four Möbius transformations

\[ \varphi_1(t_0, t_1) = (t_0, 2t_0 - t_1), \quad \varphi_2(t_0, t_1) = (-t_0 + t_1, 3t_0 + t_1), \]
\[ \varphi_3(t_0, t_1) = (t_0, t_1), \quad \varphi_4(t_0, t_1) = (t_0 - t_1, 5t_0 - t_1). \]

For \( i \in \{1, 2, 3, 4\} \), the product \( D(q(\varphi_i))D(p)^{-1} \) yields a constant matrix \( M_i \), so we get four projectivities \( f(x) = M_i x \) between the curves defined by \( p \) and \( q \) corresponding to

\[
M_1 = \begin{pmatrix}
1 & -1 & 1 & 0 \\
0 & 0 & 0 & -1 \\
0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0
\end{pmatrix}, \quad M_2 = \begin{pmatrix}
16 & -16 & 16 & 0 \\
0 & 0 & 0 & 16 \\
0 & 0 & 16 & 0 \\
0 & 16 & 0 & 0
\end{pmatrix}, \\
M_3 = \begin{pmatrix}
1 & -1 & 1 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & -1 & 1 \\
0 & 1 & 0 & 0
\end{pmatrix}, \quad M_4 = \begin{pmatrix}
16 & -16 & 16 & 0 \\
0 & 0 & 0 & -16 \\
0 & 0 & -16 & 0 \\
0 & 16 & 0 & 0
\end{pmatrix}.
\]

5. Implementation and Performance.

The algorithm Prj3D was implemented in the computer algebra system Maple™ (2021), and was tested on a PC with a 3.6 GHz Intel Core i7 processor and 32 GB RAM. In order to factor the gcd we use Maple™ (2021)'s AFactors function, since in general we want to factor over the reals. We want to explicitly mention that the Maple™ (2021) command AFactors works very well in practice. In fact, in our experimentation we observed that most of the time is spent computing the gcd of the polynomials \( E_1 \) and \( E_2 \). Technical details, examples and source codes of the procedures are provided in the first author’s personal website (Gözütoğ, 2021).

In this section, first we provide tables and examples to compare the performance of our algorithm with the algorithms in (Hauer and Jüttler, 2018; Bizzarri et al., 2020b). Then we provide a more detailed analysis of our own implementation.

We recall that the bitsize \( \tau \) of an integer \( k \) is the integer \( \tau = \lceil \log_2 k \rceil + 1 \). If the bitsize of an integer is \( \tau \), then the number of digits of the integer could be calculated by the formula \( d = \lceil \log_\tau \rceil + 1 \), where \( d \) is the number of digits. By an abuse of notation, in this section we have used \( \tau \) for representing the maximum bitsize of the coefficients of the components of the parametrization corresponding to a curve.

5.1. Comparison of the Results.

To the best of our knowledge, there are two simple and efficient algorithms to detect the projective equivalences of 3D rational curves (Hauer and Jüttler, 2018; Bizzarri et al., 2020b). Although their methods differ, in both cases the authors rely on Gröbner bases to solve a polynomial system on the coefficients of the Möbius transformations corresponding to the equivalences. Thus, in both methods most of the time is spent computing the Gröbner basis of the system, which is considerably large. In contrast, our method does not require to solve any polynomial system. Instead, our algorithm computes the Möbius-like factors by factoring a polynomial of small degree, compared to the degrees in the polynomials involved in the methods.
(Hauer and Jüttler, 2018; Bizzarri et al., 2020b). The reason is that the polynomial that we need to factor is a gcd of two polynomials where the projective curvatures $\kappa_1$ and $\kappa_2$ are involved.

In order to compare our results with those in (Hauer and Jüttler, 2018; Bizzarri et al., 2020b), we provide two tables, Table 2 and Table 3, with the timing $t_h$ corresponding to the so-called “reduced method” in (Hauer and Jüttler, 2018), the timing $t_b$ corresponding to Bizzarri et al. (2020b), and the timing $t_{our}$ corresponding to our algorithm. We consider both projective equivalences and symmetries. Since Bizzarri et al. (2020b) provide no implementation or tests in their paper, we implemented this algorithm in Maple™ (2021) to compare with our own, and the timings $t_b$ we are including are the timings obtained with this implementation. For (Hauer and Jüttler, 2018) we just reproduce the timings in their paper, taking into account that the machine in (Hauer and Jüttler, 2018) is similar to ours. We understand that the comparison is unfair because (Hauer and Jüttler, 2018) uses Singular to compute Gröbner bases, but perhaps this same fact, i.e. not using the power of Singular, that we do not need because we do not compute any Gröbner basis, may partially compensate for this unfairness. The results in Table 2 and Table 3 show that as the degree of the parametrizations grow, the timings for our algorithm grow much less that the timings for (Hauer and Jüttler, 2018; Bizzarri et al., 2020b), in accordance with the fact that Gröbner bases have an exponential complexity.

Let us present the results corresponding to Table 2. The parametrizations used in this table are given in Table 1; the first three are taken from (Hauer and Jüttler, 2018). Here we have highlighted in blue the best timing for each example. One may notice that our method always beats Bizzarri et al. (2020b), while (Hauer and Jüttler, 2018) is better for the first two examples, of small degree.
Table 2: CPU time in seconds for projective symmetries and equivalences for the curves represented by the parametrizations

| Degree | Parametrization |
|--------|-----------------|
| 4      | \( \left( t_0^4 + t_1^4 \right) \) |
|       | \( t_0 t_1 + b \cdot t_1 \) |
|        | \( t_0^2 t_1^2 \) |
|        | \( t_0^3 t_1^3 \) |
| 6      | \( \left( 125 t_0^6 + 450 t_0^4 t_1^2 + 690 t_0^2 t_1^4 + 576 t_1^6 \right) \) |
|        | \( + 276 t_0^6 + 724 t_1^6 + 84 t_1^8 \) |
|        | \( - 27 t_0^6 \cdot 54 t_0^4 t_1^2 - 36 t_0^2 t_1^4 - 8 t_1^6 \) |
|        | \( 64 t_0^6 + 288 t_0^4 t_1^2 + 528 t_0^2 t_1^4 + 504 t_1^6 \) |
|        | \( + 264 t_0^6 + 724 t_1^6 + 84 t_1^8 \) |
|        | \( + 216 t_0^6 + 216 t_0^4 t_1^2 + 168 t_0^2 t_1^4 + 60 t_1^6 \) |
|        | \( + 8 t_1^8 \) |
| 8      | \( \left( 625 t_0^8 + 3000 t_0^6 t_1^2 + 6400 t_0^4 t_1^4 + 7920 t_1^6 \right) \) |
|        | \( + 6216 t_0^6 + 31683 t_1^6 \) |
|        | \( + 102341 t_1^6 \) |
|        | \( + 192 t_1^6 \) |
|        | \( + 16 t_1^8 \) |
|        | \( - 20276 t_0^6 - 8392 t_0^4 t_1^2 - 143344 t_0^2 t_1^4 \) |
|        | \( - 12768 t_1^6 - 59680 t_1^8 \) |
|        | \( - 10560 t_0^6 + 22144 t_1^6 \) |
|        | \( + 128 t_1^6 \) |
|        | \( + 16 t_1^8 \) |
|        | \( 1664 t_0^6 + 7744 t_0^4 t_1^2 + 16288 t_0^2 t_1^4 \) |
|        | \( + 17049 t_1^6 + 9472 t_1^8 \) |
|        | \( + 3392 t_1^8 \) |
|        | \( + 7040 t_1^6 + 64 t_1^8 \) |
|        | \( + 40 t_1^8 \) |
|        | \( + 1080 t_0^6 \) |
|        | \( + 1080 t_0^4 t_1^2 \) |
|        | \( + 480 t_0^2 t_1^4 \) |
|        | \( + 80 t_1^6 \) |
| 9      | \( \left( \right) \) |
|        | \( \left( \right) \) |
| 10     | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
| 11     | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
| 12     | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |
|        | \( \left( \right) \) |

Table 1: Parametrizations of the curves considered in Section 5.1

| Degree | \( t_b \) | \( t_h \) | \( t_{our} \) |
|--------|---------|---------|------------|
| 4      | 0.344   | 0.703   | 0.01       |
| 6      | 4.391   | 2.547   | 0.06       |
| 8      | 3.094   | 2.500   | 0.78       |
| 9      | 1.140   | 1.000   | 0.016      |
| 10     | 14.750  | 10.000  | 0.422      |
| 11     | 31.625  | 21.172  | 0.421      |
| 12     | 40.313  | 41.437  | 0.625      |

Table 2: CPU time in seconds for projective symmetries and equivalences for the curves represented by the parametrizations in Table 1

Now let us introduce Table 3. In this table we test random curves with a fixed bitsize \( 3 < \tau < 4 \) (coefficients are ranges between \( -10 \) and \( 10 \)) as in (Hauer and Jüttler, 2018). The first six examples are taken from (Hauer and Jüttler, 2018). Again we have highlighted in blue the best timing between the methods in (Bizzarri et al., 2020b), (Hauer and Jüttler, 2018) and ours. Our method is only beaten in the first example, of degree 4. For higher degrees not only our algorithm is better, but the growing of the
timings is much slower.

| Deg. | Symm. | Eqvl. | Symm. | Eqvl. | Symm. | Eqvl. | Symm. | Eqvl. |
|------|-------|-------|-------|-------|-------|-------|-------|-------|
| 4    | 0.390 | 0.400 | 0.04  | 0.4   | 0.687 | 0.860 |       |       |
| 5    | 0.110 | 0.172 | 1.6   | 1.6   | 0.015 | 0.016 |       |       |
| 6    | 0.234 | 0.359 | 8.4   | 1.2   | 0.047 | 0.031 |       |       |
| 7    | 0.610 | 1.047 | 37    | 8.6   | 0.187 | 0.063 |       |       |
| 8    | 1.579 | 2.546 | 150   | 310   | 0.125 | 0.110 |       |       |
| 9    | 4.844 | 4.969 | 670   | 1700  | 0.297 | 0.343 |       |       |
| 10   | 10.439| 10.484|       |       | 0.496 | 0.391 |       |       |
| 11   | 22.265| 22.438|       |       | 0.625 | 0.453 |       |       |
| 12   | 42.625| 42.797|       |       | 0.906 | 0.547 |       |       |

Table 3: CPU time in seconds for projective equivalences and symmetries of random curves with fixed bitsize \((3 < \tau < 4)\)

5.2. Further Tests.

The tables given in this subsection are provided to better understand the performance of our method and to assist performance testing of similar studies in the future. These tables list timings for homogeneous curve parametrizations with various degrees \(m\) and coefficients with bitsizes at most \(\tau\).

5.2.1. Projective Equivalences and Symmetries of Random Curves.

In order to generate projectively equivalent curves, we apply the following non-singular matrix and Möbius transformation to a random parametrization \(q\) of degree \(n\) and bitsize \(\tau\).

\[
M = \begin{pmatrix}
1 & -1 & 1 & 0 \\
0 & 0 & 0 & -1 \\
0 & 0 & -1 & 0 \\
0 & 1 & 0 & 0
\end{pmatrix}, \quad \varphi(t_0, t_1) = (-t_0 + t_1, 2t_0).
\]

Thus, taking \(p = Mq(\varphi)\), we run \(\text{Prj3D}(p, q)\) to get the results for projective equivalences, shown in Table 4, and \(\text{Prj3D}(q, q)\) for the results in Table 5 (symmetries); since \(q\) is randomly generated, in general only the trivial symmetry is expected. Looking at Table 4 and Table 5 one observes a smooth increase in the timings for \(n \geq 5\); however \(n = 4\) has, comparatively, higher timings because for degree four curves the homogeneous polynomials \(E_1\) and \(E_2\) have more redundant common factors than with higher degrees.

| \(t\) | \(\tau = 4\) | \(\tau = 8\) | \(\tau = 16\) | \(\tau = 32\) | \(\tau = 64\) | \(\tau = 128\) | \(\tau = 256\) |
|------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| 4    | 0.703        | 0.641        | 1.500        | 3.140        | 4.640        | 10.281       | 85.989       |
| 6    | 0.062        | 0.062        | 0.047        | 0.063        | 0.110        | 0.203        | 0.531        |
| 8    | 0.109        | 0.125        | 0.140        | 0.172        | 0.969        | 1.469        | 3.578        |
| 10   | 0.343        | 0.531        | 0.250        | 0.344        | 1.000        | 2.203        | 6.000        |
| 12   | 0.641        | 0.718        | 0.891        | 0.860        | 2.063        | 3.078        | 10.719       |
| 14   | 0.890        | 1.188        | 1.313        | 1.641        | 2.922        | 5.719        | 15.704       |
| 16   | 1.218        | 1.172        | 1.593        | 1.875        | 3.437        | 7.484        | 23.828       |
| 18   | 1.797        | 1.844        | 2.313        | 2.688        | 5.656        | 9.890        | 32.985       |
| 20   | 2.344        | 2.125        | 3.281        | 4.219        | 7.297        | 14.203       | 46.282       |
| 22   | 2.985        | 3.609        | 4.203        | 5.391        | 8.781        | 18.062       | 65.000       |
| 24   | 4.125        | 4.672        | 4.859        | 6.344        | 11.110       | 20.954       | 74.766       |

Table 4: CPU times in seconds for projective equivalences of random curves with various degrees \(m\) and bitsizes at most \(\tau\)
5.2.3. Projective Equivalences of Non-equivalent Curves.

In the last table that we present here, Table 7, we generate both curves randomly, so in general no projective equivalence is expected. Table 7 shows the computation times for non-equivalent random curves with various degrees \( m \) and bitsizes at most \( \tau \).

| \( t \)  | \( \tau = 4 \) | \( \tau = 8 \) | \( \tau = 16 \) | \( \tau = 32 \) | \( \tau = 64 \) | \( \tau = 128 \) | \( \tau = 256 \) |
|-------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| 4     | 0.688         | 1.438         | 0.797         | 2.078         | 3.125         | 9.891         | 219.750       |
| 6     | 0.110         | 0.016         | 0.047         | 0.062         | 0.093         | 0.172         | 0.531         |
| 8     | 0.110         | 0.109         | 0.438         | 0.625         | 0.250         | 0.593         | 2.344         |
| 10    | 0.344         | 0.235         | 0.562         | 0.781         | 1.047         | 2.297         | 5.203         |
| 12    | 0.547         | 0.750         | 0.812         | 1.609         | 2.281         | 3.547         | 9.281         |
| 14    | 0.688         | 0.922         | 1.672         | 1.546         | 3.297         | 5.172         | 17.531        |
| 16    | 1.297         | 1.609         | 1.828         | 2.672         | 5.219         | 7.360         | 22.156        |
| 18    | 2.047         | 1.750         | 2.156         | 3.281         | 6.797         | 10.907        | 34.718        |
| 20    | 2.562         | 2.281         | 3.516         | 4.687         | 8.856         | 13.906        | 45.093        |
| 22    | 3.375         | 3.469         | 4.735         | 5.500         | 11.609        | 16.859        | 57.500        |
| 24    | 4.093         | 4.703         | 5.391         | 7.375         | 12.781        | 22.343        | 75.469        |

Table 7: CPU times in seconds for projective symmetries (central inversion) of random curves with various degrees \( m \) and bitsizes at most \( \tau \).

5.2.2. Projective Symmetries of Random Curves with Central Inversion.

To analyze the effect of an additional non-trivial symmetry, we considered random parametrizations

\[
p(t_0, t_1) = (p_0(t_0, t_1), p_1(t_0, t_1), p_2(t_0, t_1), p_3(t_0, t_1))
\]

with a symmetric \( p_0(t_0, t_1) \) and an anti-symmetric triple \( p_1(t_0, t_1), p_2(t_0, t_1) \) and \( p_3(t_0, t_1) \) of the same even-degree \( m \) and with bitsize at most \( \tau \), i.e. of the form

\[
p_0(t_0, t_1) = c_0 + t_0^{m-1} t_1 + \ldots + c_{m-1} t_0^{m-1} t_1 + t_1
\]

\[
p_1(t_0, t_1) = c_0 + t_0^{m-1} t_1 + \ldots + c_{m-1} t_0^{m-1} t_1 - c_1 t_0^m
\]

with \( c_{m-1} = 0 \) for all \( i \in \{1, 2, 3\} \). Since \( p(t_1, t_0) = (p_0(t_0, t_1), -p_1(t_0, t_1), -p_2(t_0, t_1), -p_3(t_0, t_1)) \), such homogeneous parametric curves are invariant under a central inversion with respect to the origin.

Table 6 lists the timings to detect projective symmetries (central inversions, in this case) of random curves with various degrees \( m \) and bitsizes at most \( \tau \). As expected, one can see that the computation times remain within the same magnitude order with respect to previous tables.

| \( t \)  | \( \tau = 4 \) | \( \tau = 8 \) | \( \tau = 16 \) | \( \tau = 32 \) | \( \tau = 64 \) | \( \tau = 128 \) | \( \tau = 256 \) |
|-------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| 8     | 0.078         | 0.078         | 0.078         | 0.093         | 0.141         | 0.500         | 1.109         |
| 10    | 0.234         | 0.172         | 0.313         | 0.453         | 0.781         | 1.609         | 4.516         |
| 12    | 0.360         | 0.516         | 0.531         | 0.704         | 1.282         | 3.031         | 8.140         |
| 14    | 0.625         | 0.812         | 0.703         | 1.078         | 2.062         | 5.000         | 13.032        |
| 16    | 0.921         | 1.047         | 1.203         | 1.397         | 3.125         | 7.172         | 20.109        |
| 18    | 1.329         | 1.250         | 1.578         | 2.516         | 4.969         | 9.141         | 28.922        |
| 20    | 1.765         | 1.922         | 2.282         | 3.390         | 5.594         | 14.000        | 39.125        |

Table 6: CPU times in seconds for projective symmetries (central inversion) of random curves with various degrees \( m \) and bitsizes at most \( \tau \).

5.2.3. Projective Equivalences of Non-equivalent Curves.

In the last table that we present here, Table 7, we generate both curves randomly, so in general no projective equivalence is expected. Table 7 shows the computation times for non-equivalent random curves with various degrees \( m \) and bitsizes at most \( \tau \). As expected, the timings are faster than those of Table 4, Table 5, Table 6. The reason is that in most cases the gcd \( G \) is constant and therefore the algorithm finishes earlier.
Table 7: CPU times in seconds for non-equivalent random curves with various degrees \( m \) and bitsizes at most \( \tau \)

| \( t \) | \( \tau = 4 \) | \( \tau = 8 \) | \( \tau = 16 \) | \( \tau = 32 \) | \( \tau = 64 \) | \( \tau = 128 \) | \( \tau = 256 \) |
|-------|--------|--------|--------|--------|--------|--------|--------|
| 4     | 0.016  | 0.015  | 0.016  | 0.015  | 0.015  | 0.015  | 0.015  |
| 6     | 0.094  | 0.329  | 0.031  | 0.047  | 0.034  | 0.046  | 0.688  |
| 8     | 0.062  | 0.062  | 0.078  | 0.094  | 0.110  | 0.829  | 0.328  |
| 10    | 0.313  | 0.141  | 0.157  | 0.453  | 0.796  | 0.328  | 0.656  |
| 12    | 0.281  | 0.250  | 0.718  | 0.297  | 0.391  | 0.937  | 1.343  |
| 14    | 0.547  | 0.625  | 0.391  | 0.703  | 0.953  | 1.344  | 2.500  |
| 16    | 0.922  | 0.547  | 0.969  | 0.609  | 1.031  | 1.937  | 3.595  |
| 18    | 1.062  | 1.046  | 1.047  | 1.313  | 1.500  | 2.922  | 4.266  |
| 20    | 1.438  | 1.375  | 1.312  | 1.890  | 2.344  | 3.594  | 6.359  |
| 22    | 2.109  | 1.704  | 1.609  | 2.187  | 3.219  | 4.453  | 8.891  |
| 24    | 1.719  | 2.343  | 2.391  | 2.782  | 4.234  | 6.735  | 11.078 |

5.2.4. Effect of the Bitsize and Degree on the Algorithm.

Our implementation provides solutions can deal with curves of degree 24 and bitsize 256 at the same time. When we attempt to solve the problem for higher degrees and bitsizes at the same time, the computer runs out of memory. However, by fixing the bitsize or degree we are able to go further and explore the limits of the method. This way we can check the effect of increasing the degree or the bitsize. Here we present the results of two different tests on random homogeneous parametrizations, one for a fixed bitsize and one for a fixed degree. In these tests the second parametrization is obtained by applying a projective transformation and a Möbius transformation to the first, random, parametrization. For the first test we fix the bitsize at 4, and increase the degree up to 128; for the second test, we fix the degree at 8, and increase the bitsize up to \( 2^{12} \). The results are shown in Figure 1; Figure 1a exhibits log plots of CPU times against the degree, and Figure 1b exhibits non-log plots of CPU times against the coefficient bitsizes. The data were analysed using the \texttt{PowerFit} function of the \texttt{Statistics} package of \texttt{Maple™} (2021). Thus, as a function of the degree \( m \), the CPU time \( t \) satisfies

\[
t \sim \alpha m^\beta, \quad \alpha \approx 2.0 \times 10^{-4}, \quad \beta \approx 3.1, \quad (55)
\]

and as a function of the bitsize \( \tau \), the CPU time \( t \) satisfies

\[
t \sim \alpha \tau^\beta, \quad \alpha \approx 5.7 \times 10^{-2}, \quad \beta \approx 0.6. \quad (56)
\]
6. Conclusion and Future Work.

We have presented a new approach to the problem of detecting projective equivalences of space rational curves by using projective differential invariants. The method is inspired in the ideas developed in (Alcázar et al., 2015) for computing symmetries of 3D space rational curves. The method proceeds by introducing two invariants, called projective curvatures, so that the projectivities between the curves are derived after computing the Möbius-like factors of two polynomials built from the projective curvatures. From an algorithmic point of view, it only requires gcd computing and factoring of a polynomial of relatively small degree, and therefore differs from previous approaches, where big polynomial systems were used. The experimentation carried out with Maple™ (2021) shows that the method is efficient and works better than previous approaches as the degree of the curves involved in the computation grow. Furthermore, the method seems to be generalizable to rational surfaces and hypersurfaces, with a similar strategy. We intend to pursue these generalizations in the future.

Additionally, the method opens several interesting theoretical questions. A first question is the geometric interpretation of the curvatures introduced in this paper, as well as a study of the curves where these curvatures are constant, which is a particular case that the algorithm in this paper cannot deal with. A more general question is a complete theoretical justification of the existence of invariants with the required properties, i.e. which commute with the transformation in the parameter space (in the case of this paper, Möbius transformations), as well as a development of the method for more general transformation groups and varieties.

References

Alcázar, J.G., 2014. Efficient detection of symmetries of polynomially parametrized curves. Journal of Computational and Applied Mathematics 255, 715–724. doi:10.1016/j.cam.2013.06.041.
Alcázar, J.G., Díaz-Toca, G.M., Hermoso, C., 2019a. The problem of detecting when two implicit plane algebraic curves are similar. International Journal of Algebra and Computation 29, 775–793. doi:10.1142/S0218196719500279.
Alcázar, J.G., Hermoso, C., Muntingh, G., 2014a. Detecting symmetries of rational plane and space curves. Computer Aided Geometric Design 31, 199–209. doi:10.1016/j.cagd.2014.02.004.
Appendix A. Proof of Lemma 4.

Proof. (of Lemma 4) Using the notation in Section 2 for the parametrization $u$, we get that

$$\mathbf{u}(t_0, t_1) = A \cdot \begin{bmatrix} t_1^n t_0^{-1} \\ \vdots \\ t_1 t_0^{n-1} \\ t_0^n \\ t_0^1 \end{bmatrix},$$

where $A$ is the $4 \times (n + 1)$ coefficient matrix corresponding to $u$. Differentiating (A.1) we have

$$\mathbf{u}_t(t_0, t_1) = A \cdot T_0, \quad T_0 = \begin{bmatrix} 0 \\ t_1^{n-1} \\ \vdots \\ (n-1)t_1 t_0^{n-2} \\ nt_0^{n-1} \end{bmatrix}$$

$$\mathbf{u}_{t_1}(t_0, t_1) = A \cdot T_1, \quad T_1 = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ (n-1)t_1^{n-2} t_0 \\ nt_1^{n-1} \\ (n-1)t_1^n t_0 \end{bmatrix}$$

$$\mathbf{u}_{t_2}(t_0, t_1) = A \cdot T_2, \quad T_2 = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ (n-1)(n-2)t_1 t_0^{n-3} \\ n(n-1)t_0^{n-2} \end{bmatrix}$$

$$\mathbf{u}_{t_3}(t_0, t_1) = A \cdot T_3, \quad T_3 = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ (n-1)(n-2)(n-3)t_1 t_0^{n-4} \\ n(n-1)(n-2)t_0^{n-3} \end{bmatrix}$$

Thus

$$D(u) = [A \cdot T_0 \ A \cdot T_1 \ A \cdot T_2 \ A \cdot T_3] = A \cdot T,$$

where $T = [T_0 \ T_1 \ T_2 \ T_3]$.

Now since by hypothesis $C$ is not contained in a hyperplane, \(\text{rank}(A) = 4\). Additionally, since \(n \geq 4\) we also have \(\text{rank}(T) = 4\). But then the product $A \cdot T$ must also have full rank, and therefore \(\Delta(u) = |A \cdot T|\) cannot be identically zero.

Appendix B. Projective curvatures are well defined.

In this appendix we prove that the principal invariants $I_i$, $i \in \{1, 2, 3, 4\}$, introduced in Subsection 3.2 are algebraically independent, and, as a consequence, that the projective curvatures $\kappa_1$ and $\kappa_2$ introduced in Subsection 3.3 are well defined.

The following technical results regarding the properties of the principal invariants will be later used to prove the algebraic independence of the $I_i$. 23
Lemma 12. Let $C$ be a rational algebraic curve of degree $n$ properly parametrized by $p(t_0, t_1) = (p_0(t_0, t_1), p_1(t_0, t_1), p_2(t_0, t_1), p_3(t_0, t_1))$ satisfying hypotheses (i-iv). Then $t_1$ is a factor of $\Delta(p)$.

Proof. Since the degree of $C$ is $n$, we can write

$$p_k(t_0, t_1) = \sum_{r=0}^{n} a_{r,k} t_0^{n-r} t_1^r, \quad 0 \leq k \leq 3.$$ 

The partial derivatives of order $i$ of these polynomials with respect to $t_0$ are

$$\frac{\partial^i p_k}{\partial t_0^i}(t_0, t_1) = \sum_{r=0}^{n-i} \frac{(n-r)!}{(n-i)!} a_{r,k} t_0^{n-r-i} t_1^r,$$

with $i \in \{1, 2, 3\}$.

Additionally,

$$\Delta(p) = \|p_0, p_1, p_2, p_3\| = \left| \begin{array}{cccc}
\frac{\partial p_0}{\partial t_0}(t_0, t_1) & \frac{\partial^2 p_0}{\partial t_1^2}(t_0, t_1) & \frac{\partial^3 p_0}{\partial t_1^3}(t_0, t_1) & \frac{\partial^3 p_0}{\partial t_1^3}(t_0, t_1) \\
\frac{\partial^2 p_0}{\partial t_1^2}(t_0, t_1) & \frac{\partial^3 p_0}{\partial t_1^3}(t_0, t_1) & \frac{\partial^3 p_0}{\partial t_1^3}(t_0, t_1) & \frac{\partial^3 p_0}{\partial t_1^3}(t_0, t_1) \\
\frac{\partial^2 p_0}{\partial t_1^2}(t_0, t_1) & \frac{\partial^3 p_0}{\partial t_1^3}(t_0, t_1) & \frac{\partial^3 p_0}{\partial t_1^3}(t_0, t_1) & \frac{\partial^3 p_0}{\partial t_1^3}(t_0, t_1) \\
\frac{\partial^2 p_0}{\partial t_1^2}(t_0, t_1) & \frac{\partial^3 p_0}{\partial t_1^3}(t_0, t_1) & \frac{\partial^3 p_0}{\partial t_1^3}(t_0, t_1) & \frac{\partial^3 p_0}{\partial t_1^3}(t_0, t_1) \\
\end{array} \right| .$$

In order to compute $\Delta(p)$, we expand the above determinant by the first column,

$$\Delta(p) = \frac{\partial p_0}{\partial t_0}(t_0, t_1) \begin{vmatrix} \frac{\partial p_1}{\partial t_1}(t_0, t_1), & \frac{\partial^2 p_1}{\partial t_1^2}(t_0, t_1), & \frac{\partial^3 p_1}{\partial t_1^3}(t_0, t_1) \\
\frac{\partial p_2}{\partial t_1}(t_0, t_1), & \frac{\partial^2 p_2}{\partial t_1^2}(t_0, t_1), & \frac{\partial^3 p_2}{\partial t_1^3}(t_0, t_1) \\
\frac{\partial p_3}{\partial t_1}(t_0, t_1), & \frac{\partial^2 p_3}{\partial t_1^2}(t_0, t_1), & \frac{\partial^3 p_3}{\partial t_1^3}(t_0, t_1) \\
\end{vmatrix} - \frac{\partial p_1}{\partial t_0}(t_0, t_1) \begin{vmatrix} \frac{\partial p_0}{\partial t_1}(t_0, t_1), & \frac{\partial^2 p_0}{\partial t_1^2}(t_0, t_1), & \frac{\partial^3 p_0}{\partial t_1^3}(t_0, t_1) \\
\frac{\partial p_2}{\partial t_1}(t_0, t_1), & \frac{\partial^2 p_2}{\partial t_1^2}(t_0, t_1), & \frac{\partial^3 p_2}{\partial t_1^3}(t_0, t_1) \\
\frac{\partial p_3}{\partial t_1}(t_0, t_1), & \frac{\partial^2 p_3}{\partial t_1^2}(t_0, t_1), & \frac{\partial^3 p_3}{\partial t_1^3}(t_0, t_1) \\
\end{vmatrix}$$

$$+ \frac{\partial p_2}{\partial t_0}(t_0, t_1) \begin{vmatrix} \frac{\partial p_0}{\partial t_1}(t_0, t_1), & \frac{\partial^2 p_0}{\partial t_1^2}(t_0, t_1), & \frac{\partial^3 p_0}{\partial t_1^3}(t_0, t_1) \\
\frac{\partial p_1}{\partial t_1}(t_0, t_1), & \frac{\partial^2 p_1}{\partial t_1^2}(t_0, t_1), & \frac{\partial^3 p_1}{\partial t_1^3}(t_0, t_1) \\
\frac{\partial p_3}{\partial t_1}(t_0, t_1), & \frac{\partial^2 p_3}{\partial t_1^2}(t_0, t_1), & \frac{\partial^3 p_3}{\partial t_1^3}(t_0, t_1) \\
\end{vmatrix} - \frac{\partial p_3}{\partial t_0}(t_0, t_1) \begin{vmatrix} \frac{\partial p_0}{\partial t_1}(t_0, t_1), & \frac{\partial^2 p_0}{\partial t_1^2}(t_0, t_1), & \frac{\partial^3 p_0}{\partial t_1^3}(t_0, t_1) \\
\frac{\partial p_1}{\partial t_1}(t_0, t_1), & \frac{\partial^2 p_1}{\partial t_1^2}(t_0, t_1), & \frac{\partial^3 p_1}{\partial t_1^3}(t_0, t_1) \\
\frac{\partial p_2}{\partial t_1}(t_0, t_1), & \frac{\partial^2 p_2}{\partial t_1^2}(t_0, t_1), & \frac{\partial^3 p_2}{\partial t_1^3}(t_0, t_1) \\
\end{vmatrix} .$$

Now let us consider the cofactors of the elements in the first column of each of these four determinants:

$$\frac{\partial^2 p_k}{\partial t_0^2}(t_0, t_1) \frac{\partial^3 p_k}{\partial t_1^3}(t_0, t_1) = \frac{\partial^3 p_k}{\partial t_1^3}(t_0, t_1) \frac{\partial^2 p_k}{\partial t_0^2}(t_0, t_1) =$$

$$= \sum_{r=0}^{n-2} \sum_{s=0}^{n-3} \frac{(n-r)!}{(n-r-2)!} \frac{(n-s)!}{(n-s-3)!} a_{r,k} a_{s,l} t_0^{2n-(r+s+5)} t_1^{r+s},$$

$$- \sum_{r=0}^{n-2} \sum_{s=0}^{n-3} \frac{(n-r)!}{(n-r-2)!} \frac{(n-s)!}{(n-s-3)!} a_{s,k} a_{r,l} t_0^{2n-(r+s+5)} t_1^{r+s},$$

$$- \sum_{r=0}^{n-2} \sum_{s=0}^{n-3} \frac{(n-r)!}{(n-r-2)!} \frac{(n-s)!}{(n-s-3)!} (a_{s,k} a_{r,l} - a_{r,k} a_{s,l}) t_0^{2n-(r+s+5)} t_1^{r+s},$$

where $0 \leq k < l \leq 3$. One can easily see that $a_{s,k} a_{r,l} - a_{r,k} a_{s,l} = 0$ whenever $r = s$. Thus, whenever $r+s > 0$ the factor $t_1$ is present in each cofactor, and therefore $t_1$ is a factor of $\Delta(p)$. 

□
The following lemma follows directly from properties of determinants. Here we recall the definition of the invariants $I_i$ introduced in Subsection 3.2.

$$I_1(p) := \frac{A_1(p)}{\Delta(p)}, \quad I_2(p) := \frac{A_2(p)}{\Delta(p)}, \quad I_3(p) := \frac{A_3(p)}{\Delta(p)}, \quad I_4(p) := \frac{A_4(p)}{\Delta(p)}.$$  \hfill (B.1)

**Lemma 13.** Let $C$ be a rational algebraic curve properly parametrized by $p$ satisfying hypotheses (i-iv). Then

$$A_1(p)_{t_0} = A_{51}(p) - \frac{n - 1}{t_1}A_2(p)$$

$$A_2(p)_{t_0} = A_{52}(p)$$

$$A_3(p)_{t_0} = A_{53}(p) + \frac{t_0}{t_1}A_2(p) - A_1(p)$$

$$A_4(p)_{t_0} = A_{54}(p) - A_3(p),$$

where

$$A_{51}(p) = \|p_{05} p_{16} p_{26} p_{36}\|,$$

$$A_{52}(p) = \|p_{06} p_{16} p_{26} p_{36}\|,$$

$$A_{53}(p) = \|p_{06} p_{16} p_{26} p_{36}\|,$$

$$A_{54}(p) = \|p_{06} p_{16} p_{26} p_{36}\|.$$

The next lemma is the standard bracket syzygy in the classical invariant theory (Olver, 1999).

**Lemma 14.** Let $x_0, x_1, \ldots, x_n, y_1, y_2, y_3, \ldots, y_n \in \mathbb{E}^n$. Then

$$\|x_1 \ldots x_n\| \|x_0 y_2 \ldots y_n\| - \|x_0 \ldots x_n\| \|x_1 y_2 \ldots y_n\| - \ldots - \|x_1 \ldots x_{n-1} x_0\| \|x_n y_2 \ldots y_n\| = 0. \hfill (B.2)$$

The next result introduces new relationships between the numerators of some of the invariants $I_i$, and the polynomials introduced in the statement of Lemma 13.

**Lemma 15.** Let $C$ be a rational algebraic curve properly parametrized by $p$ satisfying hypotheses (i-iv). Then we get that

$$\Delta(p_{t_0}) = \frac{n - 1}{t_1}A_2(p)$$

$$A_1(p_{t_0}) = \frac{n - 1}{t_1} \left( I_3(p) A_{52}(p) - I_2(p) A_{53}(p) \right) - \frac{t_0}{t_1} \left( I_2(p) A_{51}(p) - I_1(p) A_{52}(p) \right)$$

$$A_2(p_{t_0}) = I_2(p) A_{51}(p) - I_1(p) A_{52}(p)$$

$$A_3(p_{t_0}) = \frac{n - 1}{t_1} \left( I_4(p) A_{52}(p) - I_2(p) A_{54}(p) \right).$$

**Proof.** We prove the lemma only for $A_1(p_{t_0})$; the proofs for the other equalities are similar. By definition, we have $A_1(p_{t_0}) = \|p_{05} p_{16} p_{26} p_{36}\|$. Using Euler’s Homogeneous Function Theorem, we get that

$$A_1(p_{t_0}) = \frac{n - 1}{t_1} \|p_{05} p_{16} p_{26} p_{36}\| - \frac{t_0}{t_1} \|p_{05} p_{16} p_{26} p_{36}\|.$$

Let us apply Lemma 14 to the vectors $p_{t_0}, p_{t_0} p_{16} p_{26} p_{36}$ and $p_{t_0} p_{t_0} p_{16} p_{26} p_{36}$. Eliminating the zero determinants, we have

$$\|p_{05} p_{16} p_{26} p_{36}\| \|p_{16} p_{26} p_{36}\| - \|p_{16} p_{26} p_{36}\| \|p_{05} p_{16} p_{26} p_{36}\| = 0.$$  

By the definitions of $A_2(p), A_3(p), A_52(p), A_53(p), \Delta(p)$, we obtain

$$\|p_{05} p_{16} p_{26} p_{36}\| \Delta(p) = A_2(p) A_53(p) - A_3(p) A_52(p).$$
This yields, by Lemma 4,
\[ \|p_3^p p_0^p p_5^p\| = I_2(p) A_{5,3}(p) - I_3(p) A_{5,2}(p). \] (B.3)

Again applying Lemma 14 to the vectors \( p_0^p, p_1^p, p_3^p, p_5^p \) and \( p_1^p, p_3^p, p_5^p \) and eliminating the zero determinants, we have
\[ \|p_0^p p_3^p p_5^p\| - \|p_0^p p_3^p p_5^p\| + \|p_0^p p_3^p p_5^p\| = 0. \]

By the definitions of \( A_1(p), A_2(p), A_{5,1}(p), A_{5,2}(p), \Delta(p) \), we obtain
\[ \|p_3^p p_0^p p_5^p\| \Delta(p) = A_2(p) A_{5,1}(p) - A_1(p) A_{5,2}(p). \]

This yields, by Lemma 4,
\[ \|p_3^p p_0^p p_5^p\| = I_2(p) A_{5,1}(p) - I_1(p) A_{5,2}(p). \] (B.4)

Combining (B.3) and (B.4), we conclude that
\[ A_1(p_0^p) = \frac{n-1}{t_1} (I_3(p) A_{5,2}(p) - I_2(p) A_{5,3}(p)) - \frac{t_0}{t_1} (I_2(p) A_{5,1}(p) - I_1(p) A_{5,2}(p)). \]

\[ \square \]

**Lemma 16.** Let \( C \) be a rational algebraic curve properly parametrized by \( p \) satisfying hypotheses (i-iv), and \( c_j, j = 0, ..., n \) be the coefficient vectors of \( p \). If at least one of the polynomials \( A_1(p), A_2(p), A_3(p), A_4(p), \Delta(p) \) depend on \( t_0 \), then \( c_0 = 0 \).

**Proof.** We proceed by contradiction to prove that if neither of \( A_1(p), A_2(p), A_3(p), A_4(p), \Delta(p) \) depend on \( t_0 \), then \( c_0 = 0 \); since this cannot happen by Remark 1, the statement follows.

So let us assume that the \( A_i(p) \) and \( \Delta(p) \) do not depend on \( t_0 \). In order to show that under this assumption \( c_0 = 0 \) we use induction on \( n \). Recall that hypothesis (iii) assumes that \( n \geq 4 \). Now for \( n = 4 \), we have
\[ p(t_0, t_1) = \left( \sum_{i=0}^{4} c_{i} t_0^{4-i} t_1^i, \sum_{i=0}^{4} c_{i} t_0^{4-i} t_1^i, \sum_{i=0}^{4} c_{i} t_0^{4-i} t_1^i, \sum_{i=0}^{4} c_{i} t_0^{4-i} t_1^i \right), \]

where the \( c_{i}, 0 \leq i \leq 4 \) are the components of the vectors \( c_j, 0 \leq j \leq 3 \). Now we can compute the determinant \( \Delta(p) \) as
\[ \Delta(p) = \text{-}192\|c_3 c_4 c_5 c_0\|t_0 t_1^3 - 192\|c_2 c_4 c_5 c_0\|t_0^4 + 288\|c_3 c_4 c_1 c_0\|t_0^4 t_1 - 384\|c_3 c_4 c_2 c_0\|t_0 t_1^2 + 96\|c_3 c_4 c_2 c_1\|t_0^4. \]

Since by assumption \( \Delta(p) \) does not depend on \( t_0 \), we get
\[ \|c_3 c_4 c_5 c_0\| \neq 0, \|c_3 c_4 c_2 c_0\| \neq 0, \|c_3 c_4 c_1 c_0\| = 0, \|c_2 c_4 c_1 c_0\| = 0, \|c_2 c_3 c_1 c_0\| = 0. \]

Since \( \|c_3 c_4 c_2 c_0\| = 0 \) and \( \|c_3 c_4 c_2 c_1\| \neq 0 \), there are scalars \( \lambda_1, \lambda_2, \lambda_3 \) such that \( c_0 = \lambda_1 c_3 + \lambda_2 c_4 + \lambda_3 c_2 \). Substituting \( c_0 = \lambda_1 c_3 + \lambda_2 c_4 + \lambda_3 c_2 \) in the vanishing determinants we conclude that \( \lambda_1 = \lambda_2 = \lambda_3 = 0 \), i.e., \( c_0 = 0 \).

Assume that the lemma holds for \( n \). Let us show that for a parametrization \( p \) with degree \( n + 1 \) and coefficient vectors \( c_j, j = 0, ..., n+1 \), we also have \( c_0 = 0 \). Consider the parametrization \( q = p_{t_0} \) with degree \( n \) and coefficient vectors \( c'_j, j = 0, ..., n \). Since by assumption \( A_i(p), i \in \{1,2,3\} \) and \( \Delta(p) \) do not depend on \( t_0 \), we have
\[ A_1(p) = k_1 t_1^{4n-6}, A_2(p) = k_2 t_1^{4n-6}, A_3(p) = k_3 t_1^{4n-5}, \Delta(p) = k_0 t_1^{4n-3}. \] (B.5)
where \( k_0, k_1, k_2, k_3 \) are constants. The equation (B.3) and Lemma 13 yield

\[
A_{5,1}(p) = nk_2 t_1^{4n-7}, \quad A_{5,2}(p) = 0, \quad A_{5,3}(p) = k_1 t_1^{4n-6} - k_2 t_0 t_1^{4n-7}, \quad A_{5,4}(p) = k_3 t_1^{4n-5}.
\]  

(B.6)

By using Lemma 15 and the equations (B.5) and (B.6), we obtain that \( A_i(q) \), \( i \in \{1, 2, 3\} \) and \( \Delta(q) \) do not depend on \( t_0 \), so \( c'_0 = 0 \). However, one can easily see that \( c_j = (n+1-j)c'_j \) for all \( j \in \{0,\ldots,n\}, \) and therefore \( c_0 = 0 \). \( \square \)

Now we are ready to prove that the principal invariants are algebraically independent.

**Theorem 17.** Let \( C \) be a rational algebraic curve properly parametrized by \( p = (p_1,p_2,p_3,p_4) \) satisfying hypothesis (i-iv). The projective differential invariants \( I_1(p), I_2(p), I_3(p), I_4(p) \) of \( p \) are algebraically independent.

**Proof.** Let us assume that \( I_1(p) = \frac{A_1(p)}{\Delta(p)}, \quad I_2(p) = \frac{A_2(p)}{\Delta(p)}, \quad I_3(p) = \frac{A_3(p)}{\Delta(p)}, \quad I_4(p) = \frac{A_4(p)}{\Delta(p)} \) are algebraically dependent. It follows that the homogeneous polynomials \( \Delta(p), A_1(p), A_2(p), A_3(p), A_4(p) \) in \( t_0, t_1 \) are algebraically dependent. Thus, the Jacobian matrix

\[
J(p) = \begin{bmatrix}
\frac{\partial \Delta(p)}{\partial t_0} & \frac{\partial \Delta(p)}{\partial t_1} \\
\frac{\partial A_1(p)}{\partial t_0} & \frac{\partial A_1(p)}{\partial t_1} \\
\frac{\partial A_2(p)}{\partial t_0} & \frac{\partial A_2(p)}{\partial t_1} \\
\frac{\partial A_3(p)}{\partial t_0} & \frac{\partial A_3(p)}{\partial t_1} \\
\frac{\partial A_4(p)}{\partial t_0} & \frac{\partial A_4(p)}{\partial t_1}
\end{bmatrix}
\]

has rank 1. Note that the total degrees of the homogeneous polynomials \( \Delta(p), A_1(p), A_2(p), A_3(p), A_4(p) \) are \( 4n-7, 4n-10, 4n-10, 4n-9, 4n-8 \), respectively, where \( n \geq 4 \) is the degree of \( p \). Using Euler’s Homogeneous Function Theorem to eliminate the partial derivative with respect to \( t_1 \) in the second column of \( J(p) \), we can write \( J(p) \) as

\[
J(p) = \begin{bmatrix}
\frac{\partial \Delta(p)}{\partial t_0} & \frac{\partial \Delta(p)}{\partial t_1} \\
\frac{\partial A_1(p)}{\partial t_0} & \frac{\partial A_1(p)}{\partial t_1} \\
\frac{\partial A_2(p)}{\partial t_0} & \frac{\partial A_2(p)}{\partial t_1} \\
\frac{\partial A_3(p)}{\partial t_0} & \frac{\partial A_3(p)}{\partial t_1} \\
\frac{\partial A_4(p)}{\partial t_0} & \frac{\partial A_4(p)}{\partial t_1}
\end{bmatrix}.
\]

Applying elementary operations by columns, we reach the matrix

\[
\tilde{J}(p) = \begin{bmatrix}
\frac{\partial \Delta(p)}{\partial t_0} & \frac{\partial \Delta(p)}{\partial t_1} \\
\frac{\partial A_1(p)}{\partial t_0} & \frac{\partial A_1(p)}{\partial t_1} \\
\frac{\partial A_2(p)}{\partial t_0} & \frac{\partial A_2(p)}{\partial t_1} \\
\frac{\partial A_3(p)}{\partial t_0} & \frac{\partial A_3(p)}{\partial t_1} \\
\frac{\partial A_4(p)}{\partial t_0} & \frac{\partial A_4(p)}{\partial t_1}
\end{bmatrix}.
\]
which must also have rank 1. Because of this, \( \frac{4n - 8}{t_1} \frac{\partial \Delta(p)}{\partial t_0} A_4(p) - \frac{4n - 7}{t_1} \Delta(p) \frac{\partial A_4(p)}{\partial t_0} = 0 \). Solving this differential equation yields \( \Delta(p)^{4n-8} = h(t_1) A_4(p)^{4n-7} \), where \( h \) is an arbitrary function of \( t_1 \). But since the degrees of the homogeneous polynomials \( \Delta(p)^{4n-8} \) and \( A_4(p)^{4n-7} \) are the same, \( h \) must be a constant function, say \( h(t_1) = c \). By definition, \( \frac{\partial \Delta(p)}{\partial t_0} = A_4(p) \). Therefore we have a differential equation

\[ \Delta(p)^{4n-8} = c \left( \frac{\partial \Delta(p)}{\partial t_0} \right)^{4n-7} \]

Solving this equation, we get \( \Delta(p)(t_0, t_1) = (c_1 t_0 + g(t_1))^{4n-7} \), where \( g \) is an arbitrary function of \( t_1 \) and \( c_1 \) is a constant. We know that \( \Delta(p) \) is a homogeneous polynomial in \( t_0, t_1 \) with a total degree \( 4n - 7 \), so \( g \) must be of the form \( g(t_1) = c_2 t_1 \). On the other hand, according to Lemma 12, \( t_1 \) must be a factor of \( \Delta(p) \). Thus we have \( c_1 = 0 \), and in this case, there is a constant \( r_0 \) such that \( \Delta(p) = r_0 t_1^{4n-7} \).

Again, since \( J(p) \) has rank 1, the following equations also hold

\[ \frac{4n - 10}{t_1} \frac{\partial \Delta(p)}{\partial t_0} A_1(p) - \frac{4n - 7}{t_1} \Delta(p) \frac{\partial A_1(p)}{\partial t_0} = 0 \]  \( \text{(B.7)} \)

\[ \frac{4n - 10}{t_1} \frac{\partial \Delta(p)}{\partial t_0} A_2(p) - \frac{4n - 7}{t_1} \Delta(p) \frac{\partial A_2(p)}{\partial t_0} = 0 \]  \( \text{(B.8)} \)

\[ \frac{4n - 9}{t_1} \frac{\partial \Delta(p)}{\partial t_0} A_3(p) - \frac{4n - 7}{t_1} \Delta(p) \frac{\partial A_3(p)}{\partial t_0} = 0 \]  \( \text{(B.9)} \)

Using the fact that \( \Delta(p) \) does not depend on \( t_0 \) and the equations (B.7), (B.8), (B.9), we deduce that \( A_i(p) \) for \( i \in \{1, 2, 3\} \) do not depend on \( t_0 \). But this contradicts Lemma 16, and therefore the \( I_i \) are algebraically independent.

And we can finally prove Lemma 8.

Proof. (of Lemma 8) Assume that \( I_3(p) \) and \( I_4(p) \) are not identically zero. Then because of Theorem 17, the expression \( 8(n - 3) I_3(p) + 3(n - 2) I_4^2(p) \) in the denominator of \( \kappa_1, \kappa_2 \) cannot be identically zero. So let us assume that both \( I_3(p) \) and \( I_4(p) \) are identically zero. Then, by the definition of the principal invariants, \( A_3(p) \) and \( A_4(p) \) are both identically zero. Since \( A_3(p) = \| p_{i_1} p_{i_2} p_{i_3}^2 \| = 0 \) and \( A_4(p) = \| p_{i_1} p_{i_2}^2 p_{i_3}^2 \| = 0 \), we conclude that both of \( p_{i_1}^2 p_{i_2} \) and \( p_{i_3}^2 p_{i_2} \) are linear combinations of \( p_{i_1}, p_{i_2}, p_{i_3} \). Substituting these linear combinations in \( A_1(p) \) and \( A_2(p) \) yields that both \( A_1(p) \) and \( A_2(p) \) are identically zero too. But this again contradicts Theorem 17.