SPECIAL ISSUE PAPER

Solver-free optimal control for linear dynamical switched system by means of geometric algebra

Anna Derevianko | Petr Vašík

Brno University of Technology, Brno, Czech Republic

Correspondence
Petr Vašík, Brno University of Technology, Brno, Czech Republic.
Email: Petr.Vasik@vutbr.cz

Communicated by: E. Hitzer

Funding information
Vysoké Učení Technické v Brně, Grant/Award Number: FSI-S-20-6187

An algorithm for finding a control of a linear switched system by means of Geometric Algebra is designed. More precisely, we develop a switching path searching algorithm for a two-dimensional linear dynamical switched system with a non-singular matrix whose integral curves are formed by two sets of centralized ellipses. It is natural to represent them as elements of Geometric Algebra for Conics and construct the switching path by calculating switching points, i.e., intersections and contact points. For this, we use symbolic algebra operations or, more precisely, the wedge and inner products that are realizable by sums of products in the coordinate form. Therefore, no numerical solver to the system of equations is needed. Indeed, the only operation that may bring in an inaccuracy is vector normalization, i.e., square root calculation. The resulting switching path is formed by pieces of ellipses that are chosen, respectively, from the two sets of integral curves. The switching points are either intersections in the first or final step of our algorithm, or contact points. This choice guarantees the optimality of the switching path with respect to the number of switches. Two examples are provided to demonstrate the search for the intersections of the conics and, consequently, a description is presented of the construction of a switching path in both cases.

KEYWORDS
Clifford algebra, controllability, geometric algebra, switched system

MSC CLASSIFICATION
15A66, 37N35

1 | INTRODUCTION

Switched systems form a special case of hybrid dynamical systems with discrete and continuous dynamics. They are widely applied when a real system cannot be described by one single model. Numerous examples are given by engineering systems of electronics, power systems, traffic control, and others. Since the 1990s, researching the stability of switched systems has become very popular; see, e.g., Vidyasagar¹ and Sun and Sam Ge.² A particular case of linear switched systems can be found in Colaneri.³ More recent literature about switched systems includes publications by Colaneri,³ Lin et al,⁴ and Jiang and Wang⁵; the question of stability remains widely studied until today.

In the sequel, we use the power of Geometric Algebra for Conics (GAC) to control a $2 \times 2$ linear dynamical switched system with non-singular matrices, where we exploit the fact that their control paths form a set of centralized ellipses.
and, for the control of a switched system, it is enough to find the point of a switch, i.e., the intersection of two ellipses. Classically, this leads to a system of quadratic equations which is simple to solve numerically, but a certain inaccuracy is involved, i.e., a wrong control path is chosen. This error increases with the increasing number of switches due to inaccurate initial conditions. Therefore, we introduce an algorithm for searching for intersections of ellipses with no solver needed, which, together with a straightforward way of conic scaling, leads to a control with a minimal number of switches. In this sense, our control is optimal.

If we restrict to the above-described class of switched systems, their (optimal) control by means of Geometric Algebra for Conics (GAC) can be designed. This is an efficient geometric tool to handle both conics and their transformations as elements of a particular Clifford algebra. Also, and this is particularly exploited by our approach, intersections and contact points of conics may be obtained by simple operations in GAC, namely, by the wedge and inner product with a great advantage that they can be expressed as sums of products, and thus, no numerical solver is needed. Our algorithm uses not only GAC but also its subalgebra called Compass Ruler Algebra (CRA), which is a conformal model of a two-dimensional Euclidean space with circles as intrinsic geometric primitives.

Our main idea is based on a recent concept of GAC, although there have been other constructions; see, e.g., Easter and Hitzer. Since the concept in Hrdina et al is rather new, we refer to Hrdina et al for better insight into the description of objects and their manipulation in GAC. Generally, our main observation is that, in GAC, you can express the four intersecting points of two specific conics without knowing their precise coordinates and, consequently, you can construct a degenerate conic of two intersecting lines containing the intersections of the conics. The lines can be separated by means of standard linear algebra operations. One either continues with a geometric approach and calculates the intersection with a circle in CRA, see Example 3, or puts the line equation together with a quadratic equation of a conic to find their intersections analytically, see Example 1.

Our results are demonstrated on examples of specific switched systems. The output is provided of our implementation in Python using a module clifford for symbolic GAC operations.

## 2 | STATE OF THE ART

Nowadays, the most popular approach to searching for a control of a switched system is connected with Lyapunov functions, which requires sophisticated algorithmic structures and precise criteria for checking the validity of the method used in the particular problem. The ordinary Lyapunov function is used to test whether a given dynamical system is stable (more precisely, asymptotically stable) but does not provide any information about controllability. A particular specificity of switched systems lies in the interaction between the continuous variable and the discrete state, which is not present in the standard control systems. In his publications, Liberzon and Tempo address the problems of stability and control for particular types of switched systems using the analytical approach, i.e., Lyapunov function and Brockett’s condition for asymptotic stability by continuous feedback and controllability. Stabilization of switched positive regular linear systems by state-dependent switching was considered in Ding and Liu, and an anti-bump switching control problem was introduced in Han et al.

The issue of the optimal control has also been addressed several times. The most popular are the problems of time or distance optimality. For example, finding the time-optimal control for a dynamical system was considered by Nasir, while for switched systems, an analogical problem was considered in Seidman, where the author constructs a minimizing sequence and uses the compactness property for finding a subsequence that minimizes the cost functional. Another way of optimization is the construction of a switching path with a minimal amount of switches, which is of our particular interest. This is clearly achieved if the switch points are order one contact points; i.e., consequent ellipses are circumscribed. We are not aware of any similar construction in literature.

For switched systems of 2 × 2 dimensions, it is possible to construct an analytical solution using classical methods. Thus, if \( A_1 = \begin{pmatrix} 0 & 1 \\ -\alpha & 0 \end{pmatrix} \), i.e., a 2 × 2 matrix with purely imaginary eigenvalues \( \pm \sqrt{\alpha} \), then the solution to the system

\[
\dot{x}(t) = Ax(t)
\]
can be written in the form
\[ x(t) = \gamma_1 \sin \left( \sqrt{\alpha_1} t \right) + \gamma_2 \cos \left( \sqrt{\alpha_1} t \right), \]
\[ y(t) = \sqrt{\alpha_1} \gamma_1 \sin \left( \sqrt{\alpha_1} t \right) - \sqrt{\alpha_1} \gamma_2 \cos \left( \sqrt{\alpha_1} t \right), \quad \gamma_1, \gamma_2 \in \mathbb{R}. \]
\[ \dot{x}(t) = A(t) x(t) \]

After parameter elimination, one gets the solution in the form of the ellipse polar equation
\[
\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1.
\]

Also, in this specific case, one can write the solution directly in the form
\[
\frac{x^2}{a} + \frac{y^2}{c^2} = 1.
\]

The procedure for finding a switched control then proceeds numerically, especially the intersection calculations. However, in the case of general type complex eigenvalues (i.e., the rotated case), the parameter elimination is harder, and, consequently, the properties of the ellipses are not easy to obtain. For 2 \times 2 systems, one can also use the nullcline method, where there is clear separation of timescales between the variables. By the use of nullclines, the trajectories for each subsystem can be found; however, in the case of non axis-aligned conics, it becomes much more complicated for software implementation.

A classical procedure for scaling an ellipse is performed by stretching its semi axis. To do this, we have to work with the polar equation of the conic, which, in the case of a rotated ellipse, becomes more complex. For instance, the Gröbner basis procedure has to be employed and implemented.

One should also pay attention to the existing methods for finding these intersections: In a Euclidean space, the problem of finding the intersection points of two conics is reduced to solving a system of quadratic equations using numerical methods.

3 | GEOMETRIC ALGEBRA

By Geometric Algebra (GA), we mean a Clifford algebra with a Euclidean space (of arbitrary dimension) embedded in such a way that the intrinsic geometric primitives as well as their transformations are viewed as elements of a single vector space or, more precisely, multivectors. This concept was introduced by Hestenes and has been used in many mathematical and engineering applications since; see, e.g., Gonzalez-Jimenez et al. and Hrdina et al.

A great calculational advantage of GA is that geometric operations such as intersections, tangents, and distances are linear functions and, therefore, their calculation is efficient. To demonstrate this, we refer to Perwass for the basics of geometric algebras, especially for conformal representation of a Euclidean space. Indeed, a three-dimensional Euclidean space is represented in Clifford algebra $\mathcal{C}(4,1)$, and the consequent geometric algebra is often denoted as $G_{4,1}$ (or $G_3$) with spheres of all types as geometric primitives and Euclidean transformations at hand; see, e.g., Dorst et al. In the sequel, we also use the two-dimensional subalgebra $G_{3,1}$ (or $G_2$) called a Compass Ruler Algebra (CRA), which is an analogue of $G_{4,1}$ for a two-dimensional Euclidean space.

Let us now recall the generalization of $G_{4,1}$, i.e., Geometric Algebra for Conics (GAC), proposed by Perwass to generalize the concept of (two-dimensional) conformal geometric algebra $G_{3,1}$. Let us stress that we use the notation of Hrdina et al. In the usual basis $\bar{n}, e_1, e_2, n$, the embedding of a plane into $G_{3,1}$ is given by

\[
(x, y) \mapsto \bar{n} + xe_1 + ye_2 + \frac{1}{2}(x^2 + y^2)n,
\]

where $e_1, e_2$ form the Euclidean basis and $\bar{n}$ and $n$ stand for a specific linear combination of additional basis vectors $e_3, e_4$ with $e_3^2 = 1$ and $e_4^2 = -1$, giving them the meaning of the coordinate origin and infinity, respectively. Hence, the objects representable by vectors in $G_{3,1}$ are linear combinations of $1, x, y, x^2 + y^2$, i.e., circles, lines, point pairs, and points. To cover
general conics as well, two terms have to be added: \( \frac{1}{2}(x^2 - y^2) \) and \( xy \). It turns out that two new infinities are needed for that as well as their two corresponding counterparts (Witt pairs). Thus, the resulting dimension of the space generating the appropriate geometric algebra is eight.

Analogously to CGA and to the notation in Perwass,\(^{24}\) we denote the corresponding basis elements as

\[
\hat{n}_+ , \hat{n}_-, \hat{n}_x, e_1, e_2, n_+, n_-, n_x. \tag{1}
\]

This notation suggests that the basis elements \( e_1, e_2 \) play the usual role of a standard basis of the Euclidean plane, while the null vectors \( \hat{n}, n \) represent the origin and infinity, respectively. Note that there are three orthogonal “origins” \( \hat{n} \) and three corresponding orthogonal “infinities” \( n \). In basis (1), a point of the plane \( x \in \mathbb{R}^2 \) defined by \( x = xe_1 + ye_2 \) is embedded using the operator \( C : \mathbb{R}^2 \rightarrow Cones \subset \mathbb{R}^{5,3} \), which is defined by

\[
C(x, y) = \hat{n}_+ + xe_1 + ye_2 + \frac{1}{2}(x^2 + y^2)n_+ + \frac{1}{2}(x^2 - y^2)n_- + xyn_x. \tag{2}
\]

The image \( Cone \) is an analogue of the conformal cone. In fact, it is a two-dimensional real projective variety determined by five homogeneous polynomials of degree 1 and 2.

**Definition 1.** Geometric Algebra for Conics (GAC) is the Clifford algebra \( Cl_{5,3} \) together with the embedding (2) in the basis (1).

Note that, except for the last two terms, the embedding (2) is the embedding of the plane into the two-dimensional conformal geometric algebra \( G_{5,1} \).

Let us recall that the invertible algebra elements are called versors and they form a group, the Clifford group, and that conjugations with versors give transformations intrinsic to the algebra. Namely, if the conjugation with a \( G_{5,3} \) versor \( R \) preserves the set \( Cone \), i.e., for each \( x \in \mathbb{R}^2 \), there exists such a point \( \tilde{x} \in \mathbb{R}^2 \) that

\[
RC(x)\tilde{R} = C(\tilde{x}), \tag{3}
\]

where \( \tilde{R} \) is the reverse of \( R \); then, \( x \rightarrow \tilde{x} \) induces a transformation \( \mathbb{R}^2 \rightarrow \mathbb{R}^2 \), which is intrinsic to GAC. See Hrdina et al\(^{6}\) to find that the conformal transformations are intrinsic to GAC.

The representable objects can be found by examining the inner product of a vector and an embedded point. An inner representation of a conic in GAC can be defined as the vector

\[
Q_I = \tilde{v}^+ \hat{n}_+ + \tilde{v}^- \hat{n}_- + \tilde{v}^x \hat{n}_x + \tilde{v}^1 e_1 + \tilde{v}^2 e_2 + \tilde{v}^+ n_+. \tag{4}
\]

The type of a given unknown conic can be read off its matrix representation, which in our case, for a conic given by vector (4), reads

\[
Q = \begin{pmatrix}
-\frac{1}{2}(\tilde{v}^+ + \tilde{v}^-) & -\frac{1}{2} \tilde{v}^x & \frac{1}{2} \tilde{v}^1 \\
-\frac{1}{2} \tilde{v}^x & -\frac{1}{2}(\tilde{v}^+ - \tilde{v}^-) & \frac{1}{2} \tilde{v}^2 \\
\frac{1}{2} \tilde{v}^1 & \frac{1}{2} \tilde{v}^2 & -\tilde{v}^+ 
\end{pmatrix}. \tag{5}
\]

The internal parameters of a conic and its position and orientation in the plane are determined from the matrix (5). Hence, all this can be determined from the GAC vector \( Q_I \) by means of the inner product.

The classification of conics is well known. The non-degenerate conics are of three types, the ellipse, hyperbola, and parabola. Now, we present the vector form (4) appropriate to the simplest case, i.e., an axis-aligned ellipse \( E_I \) with its center in the origin and semi-axes \( a, b \in R^+ \). The correctness can be verified easily by multiplying its vector by an embedded point, which means the application of (2). The corresponding GAC vector is of the form

\[
E_I = (a^2 + b^2)\hat{n}_+ + (a^2 - b^2)\hat{n}_- a^2 b^2 n_+. \tag{6}
\]
Remark 1. Note that a line in GAC is not an intrinsic primitive and, thus, one may understand it as a CRA object. Therefore, its IPNS representation has the same form as in CRA,\(^8\)

\[ n_1 e_1 + n_2 e_2 + d n_+ , \]

where \( n = (n_1, n_2) \) is the normal vector and \( d \) is the distance from the coordinate origin. The OPNS representation of a line passing through two points \( P_1 \) and \( P_2 \) is\(^6\) of the form

\[ P_1 \wedge P_2 \wedge n_+ \wedge n_- \wedge n_\infty . \]

Regarding the transformations, our algorithm uses explicitly just isotropic scaling, which is non-Euclidean, but can be computed in the same way. Indeed, the scaling is generated by a bivector and the action of this bivector on a GAC element is given by conjugation. To show the precise coordinate form and all the possibilities, the following proposition is recalled.\(^6\)

### 4 | INTERSECTIONS AND CONTACT POINTS IN GAC

This section provides a procedure for intersecting two conics, particularly, ellipses with a common center in the coordinate origin but in a general mutual position otherwise. Moreover, a system of circumscribed ellipses is considered, and a procedure is shown for detecting the first-order contact points, i.e., the points in Figure 1 where the ellipses touch with an identical first-order derivative. Again, the contribution of GAC lies in avoiding the use of a solver, which leads to accuracy improvement.

Let us first describe some differences to CRA or its three-dimensional version CGA (Conformal Geometric Algebra). A crucial difference lies in the type of objects that are intrinsic to the respective structures. For CGA (CRA), spheres (circles) are the geometric primitives that may be represented by specific elements. Taking into account that lines and planes are spheres with infinite radii and a point pair is a one-dimensional sphere, one receives all the geometric primitives for Euclidean geometry. Moreover, their intersections still remain such objects. Indeed, an intersection of two spheres or two circles are circles or point pairs, respectively. Therefore, intersections that are realized by a wedge of IPNS representations remain the representatives of Euclidean primitives intrinsic to CGA (CRA). On the contrary, the situation is different in GAC. Even if we restrict to the case of co-centric ellipses, their intersection is a “four point” (or a point pair or no proper point), which has no meaning in the sense of conic-sections. Indeed, a planar conic is uniquely generated by five points. This leads to an algorithm that may be used for co-centric conics (all types). On the other hand, the algorithm is still geometric-based and may be realized by a sequence of simple operations in GAC, i.e. there is no numerical solver involved.

#### 4.1 | Intersections

Let us now present the procedure for computing the intersections of two co-centric ellipses, i.e., the set up according to Figure 1. Note that it may be assumed, without loss of generality, that the ellipses have four intersection points. Other cases are not of our interest and would be recognized by the form of a GAC element representing no conic. Furthermore,
it may be assumed that the centers of the ellipses are situated at the coordinate origin; otherwise, the whole picture can be translated in GAC to fulfill this assumption.

We start by taking two IPNS representations of ellipses \( E_1 \) and \( E_2 \) and wedge them. The result corresponds to the common points of both geometric primitives. This is a standard operation intrinsic to any geometric algebra. In our case, we receive an IPNS representative of a four-point \( E_1 \wedge E_2 = P_1 \wedge P_2 \wedge P_3 \wedge P_4 \) as in Figure 1. Therefore, as the next step, we construct a degenerate conic, more precisely, a pair of intersecting lines \((E_1 \wedge E_2)^* \wedge \mathbf{n}_+\), where \( \mathbf{n}_+ \) represents the origin of the Euclidean coordinates and, therefore, the common ellipse center, and \((E_1 \wedge E_2)^*\) is the OPNS representation of the four point.

Now, we need to decompose the pair of lines into two separate single lines. First, the matrix \( Q \) of its quadratic form by (5) is constructed. Note that \( Q \) is a symmetric singular matrix. To decompose a degenerate conic, we follow an algorithm described in Richter-Gebert. 27 Let us recall the algorithm just to present that all the operations involved are the sums and products in the determinant calculations. The only numerical inaccuracy may be imported by a square root calculation.

Indeed, to decompose a pair of intersecting lines into two distinct lines, we have to find a skew-symmetric matrix \( P \) formed by parameters \( \lambda, \mu, \) and \( \tau \) such that \( N = Q + P \) is of rank 1. Thus, in our case, we have

\[
N = \begin{pmatrix}
q_{11} & q_{12} & q_{13} \\
q_{21} & q_{22} & q_{23} \\
q_{31} & q_{32} & q_{33}
\end{pmatrix} + \begin{pmatrix}
0 & \tau & -\mu \\
-\tau & 0 & -\lambda \\
\mu & \lambda & 0
\end{pmatrix}.
\] (7)

The rank condition reads that every \( 2 \times 2 \) submatrix determinant must vanish. Thus, the necessary conditions for the parameters \( \lambda, \mu, \) and \( \tau \) are

\[
\tau^2 = -\left| \begin{array}{cc}
q_{11} & q_{12} \\
q_{21} & q_{22}
\end{array} \right|, \quad \mu^2 = -\left| \begin{array}{cc}
q_{11} & q_{13} \\
q_{31} & q_{33}
\end{array} \right|, \quad \lambda^2 = -\left| \begin{array}{cc}
q_{22} & q_{23} \\
q_{32} & q_{33}
\end{array} \right|.
\]

This determines the parameters \( \lambda, \mu, \) and \( \tau \) up to their sign. In a general case, to get the precise values of \( \lambda, \mu, \) and \( \tau \), one can take a nonzero column of the matrix dual to \( Q \) and divide it with a specific factor; see Richter-Gebert. 27 If the lines are passing through the origin, the division may be omitted, and, thus, only the dual matrix, i.e., nine determinants of order 2, has to be calculated; see Richter-Gebert. 27

By taking an arbitrary nonzero row and a nonzero column of the matrix \( N \), we get the coefficients of the respective separated lines. We shall now recall that a single line represents no conic and, therefore, it is not a geometric primitive intrinsic to GAC. Yet it is understood as an element of subalgebra CRA, i.e., a model of two-dimensional Euclidean space formed by Clifford algebra \( Cl(3,1) \); see Hildenbrand. 8

**Example 1.** To construct an ellipse, we need the semi-axis lengths \( a, b \) center coordinates \( c_1, c_2 \) and the angle of rotation \( \theta \). Let us consider two ellipses, \( Ell_1 \) and \( Ell_2 \), with parameters \( (a, b, c_1, c_2, \theta) = (2, 4, 0, 0, 0) \) and \( (4, 2, 0, 0, \frac{\pi}{6}) \), respectively; see Figure 2A.

Their IPNS representations will then be of the form

\[
Ell_1 = \mathbf{n}_+ + \frac{3}{5} \mathbf{n}_- - \frac{16}{5} \mathbf{n}_+
\]

![FIGURE 2](https://example.com/figure2.png)  Setting of Example 1: (A) ellipse setting; (B) pair of lines [Colour figure can be viewed at wileyonlinelibrary.com]
and
\[ Ell2 = \hat{n}_+ - \frac{3}{10} \hat{n}_- - \frac{3\sqrt{3}}{10} \hat{n}_x - \frac{16}{5} n_+. \]

If transformed to OPNS, the intersections become four vectors. Therefore, their representation corresponds to a wedge of four GAC points. By wedging the origin represented by \( \hat{n}_+ \), we receive an OPNS representation of a degenerate conic, more precisely, of a pair of intersecting lines. Their IPNS form is
\[-\frac{72}{25} \hat{n}_- - \frac{24\sqrt{3}}{25} \hat{n}_x.\]

The type of the conic may be easily checked using their matrix form
\[
\begin{pmatrix}
\frac{36}{25} & -\frac{12\sqrt{3}}{25} & 0 & -\frac{12\sqrt{3}}{25} & 0 & 0 & 0 & 0 \\
-\frac{12\sqrt{3}}{25} & \frac{36}{25} & 0 & \frac{36}{25} & 0 & 0 & 0 & 0 \\
0 & 0 & -\frac{36}{25} & -\frac{36}{25} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -\frac{36}{25} & -\frac{36}{25} & 0 & 0 & 0
\end{pmatrix},
\]

After normalization, the equation of this conic is \( x^2 - y^2 + \frac{24\sqrt{3}}{3} xy = 0 \). Thus, we have a pair of lines containing all four intersections of the ellipses and the origin; see Figure 2B.

Let us provide all the necessary inputs for the procedure of line separation (7) in the same notation:
\[
\begin{pmatrix}
\frac{36}{25} & -\frac{12\sqrt{3}}{25} & 0 & -\frac{12\sqrt{3}}{25} & 0 & 0 & 0 & 0 \\
-\frac{12\sqrt{3}}{25} & \frac{36}{25} & 0 & \frac{36}{25} & 0 & 0 & 0 & 0 \\
0 & 0 & -\frac{36}{25} & -\frac{36}{25} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -\frac{36}{25} & -\frac{36}{25} & 0 & 0 & 0
\end{pmatrix} + \begin{pmatrix}
0 & -\frac{12\sqrt{3}}{25} & 0 & 0 & 0 & 0 & 0 & 0 \\
-\frac{12\sqrt{3}}{25} & \frac{36}{25} & 0 & \frac{36}{25} & 0 & 0 & 0 & 0 \\
0 & 0 & -\frac{36}{25} & -\frac{36}{25} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -\frac{36}{25} & -\frac{36}{25} & 0 & 0 & 0
\end{pmatrix},
\]
i.e., \( \mu = 0, \tau = -\frac{24\sqrt{3}}{25}, \lambda = 0 \). Therefore, the matrix of the pair of lines has the form
\[
\begin{pmatrix}
\frac{36}{25} & -\frac{12\sqrt{3}}{25} & 0 & 0 \\
-\frac{12\sqrt{3}}{25} & \frac{36}{25} & 0 & 0 \\
0 & 0 & -\frac{36}{25} & -\frac{36}{25} \\
0 & 0 & 0 & -\frac{36}{25}
\end{pmatrix},
\]

and, thus, the separated lines may be easily derived according to the first (nonzero) row and column. After normalization, we receive
\[
\frac{1}{2} x + \frac{\sqrt{3}}{2} y = 0, \text{ and } \frac{\sqrt{3}}{2} x - \frac{1}{2} y = 0.
\]

It is clear that they are perpendicular, which has been expected due to the symmetries given by the setting.

Thus, we get a system of quadratic (ellipse) and linear (line) equations that does not require the use of a solver. In our case,
\[
\begin{align*}
x + \sqrt{3} y &= 0, \\
x^2 + \frac{y^2}{4} &= 1,
\end{align*}
\]

for \( x = -\sqrt{3} y, 13y^2 - 16 = 0 \) and \( y = \pm \frac{4\sqrt{13}}{13}, x = \mp \frac{4\sqrt{39}}{13} \). So the intersection points are \( \left[ \frac{4\sqrt{39}}{13}, \frac{4\sqrt{13}}{13} \right], \left[ -\frac{4\sqrt{39}}{13}, \frac{4\sqrt{13}}{13} \right] \).

In the same way, by using the line \( x + \sqrt{3} y = 0 \), we get the following points: \( \left[ \frac{4\sqrt{39}}{13}, \frac{4\sqrt{13}}{13} \right], \left[ -\frac{4\sqrt{39}}{13}, -\frac{4\sqrt{13}}{13} \right] \).

The above procedure can be applied to different types of co-centered conics with four intersection points. Note that if a conic is constructed in GAC by wedging points, it is necessary to work with very high coordinate precision, because even a small inaccuracy in the coordinates of points to be wedged may lead to a different conic or to a conic of a different type. Furthermore, after the wedge, the normalization factor has to be determined for correct parameter extraction.

**Example 2.** Figure 3 demonstrates the output of the Python code for different types of co-centered conics with four intersection points. It shows that our considerations are valid not only for ellipses but for arbitrary pairs of non-degenerate co-centered conics; see Loučka and Vašík\(^{28} \) for proofs.
Example 3. In the case of axes-aligned ellipses, a more geometric approach can be applied. Given two ellipses \( Ell1 \) and \( Ell2 \) with parameters \( (a, b, c_1, c_2, \theta) = (2, 4, 0, 0, 0) \) and \( (4, 2, 0, 0, 0) \), respectively, we determine their IPNS representations according to (6) in the form

\[
Ell1 = \bar{n}_+ + \frac{3}{5} \bar{n}_- - \frac{16}{5} n_+
\]

and

\[
Ell1 = \bar{n}_+ - \frac{3}{5} \bar{n}_- - \frac{16}{5} n_+.
\]

The intersecting points lie on a circle \( C \) that may be constructed by \( (Ell1 \wedge Ell2)^* \wedge \bar{n}_+ \), and thus, it may be represented by a GAC element

\[
C = \frac{6}{5} \bar{n}_+ - \frac{96}{5} n_+;
\]

i.e., its standard equation will be

\[
x^2 + y^2 - \frac{32}{5} = 0.
\]

Then, we can construct a pair of intersecting lines \( (Ell1 \wedge Ell2)^* \wedge \bar{n}_+ \) with the IPNS representation

\[
\frac{6}{5} \bar{n}_- - \frac{96}{25} n_+,
\]

i.e., of the equation (after normalization) \(-x^2 + y^2 = 0\).

The line decomposition procedure, although not necessary in this particular case, will lead to a pair of lines \( y = x \) and \( y = -x \). As CRA elements, they are of the form \( l_1 = \frac{-\sqrt{2}}{2} e_1 + \frac{\sqrt{2}}{2} e_2 \) and \( l_2 = \frac{\sqrt{2}}{2} e_1 + \frac{\sqrt{2}}{2} e_2 \), respectively. Then, it is enough to calculate the intersections \( C \wedge l_1 \) and \( C \wedge l_2 \) to get two point pairs \( P_1, P_2 \) in CRA. Consequently, a procedure for a point pair decomposition must be applied in the form

\[
p_{i1} = \frac{-\sqrt{P_{i1} \cdot P_i} + P_i}{n_+ \cdot P_i}, \quad p_{i2} = \frac{\sqrt{P_{i1} \cdot P_i} + P_i}{n_+ \cdot P_i} \text{ for } i = 1, 2.
\]

In this very simple case, we receive the CRA points

\[
\bar{n}_+ \pm \frac{4\sqrt{5}}{5} e_1 \pm \frac{4\sqrt{5}}{5} e_2 \pm \frac{16}{5} n^+,
\]

which means that the points of intersections are of the form \( \left[ \pm \frac{4\sqrt{5}}{5}, \pm \frac{4\sqrt{5}}{5} \right] \).

Note that, in GAC, the authors are not aware of any universal procedure for 4-point or point pair decomposition.
4.2 Contact points

As mentioned above, by contact points, we understand first order contact points, i.e., points where two curves have identical first order derivative. We shall describe how to receive a set of contact points for a given system of co-centered ellipses. Such system is formed, as shown in Figure 4, beginning with two intersecting ellipses $E_1$ and $E_2$. Then, the ellipse $E'_2$ is constructed from $E_2$ just by scaling in such a way that $E'_2$ is circumscribed to $E_1$; i.e., they have two contact points. Then, the ellipse $E'_1$ would be constructed from $E_1$ so that it would be circumscribed to $E'_2$, etc.

**Proposition 1.** Given a system of co-centric ellipses as in Figure 4, the contact points form a pair of intersecting lines. Furthermore, keeping the notation of 4, one of these lines is the axis of lines $p_1$ and $p_2$ denoted as $1_2(p_1 + p_2)$, and, similarly, the other line is the axis of the lines $p_3$ and $p_4$ in Figure 4.

**Proof.** Taking into account that the ellipses are co-centric and their symmetry properties, it is obvious that 4 points of their intersection form a parallelogram with diagonals passing through the common center. The line $1_2(p_1 + p_2)$ is the middle line of the parallelogram passing through the common center $S$ of the ellipses. Note that the notation $1_2(p_1 + p_2)$ for the axes of $p_1$ and $p_2$ is exactly the way to calculate this line in CRA. Indeed, this is true for the IPNS representations of $p_1$ and $p_2$. □

**Proposition 2.** Given a system of co-centric ellipses as in Figure 4, a scalor transforming an ellipse $E_2$ to $E'_2$ may be calculated as $SP = \frac{|SK'|}{|SK|}$, where $K$ and $K'$ are the intersection points of the ellipses $E_2$ and $E_1$ with the line $1_2(p_1 + p_2)$ and $S$ is the common center of the ellipses.

**Proof.** Provided that $K'$ is the contact point of ellipses $E'_2$ and $E_1$, scaling ellipse $E_2$ until it touches $E_1$ means scaling the length section $SK$ until it reaches the length of $SK'$. Therefore, $SP = \frac{|SK'|}{|SK|}$, where $S$ denotes the common center of the ellipses. □

**Remark 2.** The transformation of $E_2$ to $E'_2$ is then given in GAC by a scalor. In the case of co-centered ellipses with the center in the coordinate system origin, we may just multiply the semi-axis lengths by the scaling factor.

5 | CONTROLLABLE $2 \times 2$ LINEAR SWITCHED DYNAMICAL SYSTEMS

Let us briefly recall the basic terminology in the switched systems theory. By a switched system, we mean the following system:

$$\dot{x}(t) = f_{\sigma(t)}(x(t)), \quad x(0) = x_0,$$

where $x \in \mathbb{R}^m$ is called a continuous state, $\sigma$ stands for a discrete state with values from an index set $M := \{1, \ldots, n\}$, and $f_{\sigma(t)}$, for $\sigma(t) \in M$, are given vector fields.

The behavior of the dynamical system is regulated by a switching signal. Namely, at specific time moments, i.e., for $t = \tau_1, \ldots, \tau_n$, the system changes its setting from $\sigma(\tau_i)$ to $\sigma(\tau_{i+1})$; hence, the trajectory of the system, starting from $t = \tau_i$, is given by the vector field $f_{\sigma(\tau_{i+1})}$ instead of $f_{\sigma(\tau_i)}$. Based on the publications on switched systems, switching times can be random or given by a law. In the sequel, we consider a different formulation of the problem; i.e., the switching signal...
is under our control. To guarantee that there exists a path connecting two arbitrary points, let us recall the following definition.

**Definition 2.** We say that the switched system

\[ \dot{x}(t) = f_{\sigma(t)}(x(t)), \quad x(0) = x_0 \]

is controllable if, for any two points \( A, B \) from the state space, there exists a switching signal generating a continuous path from \( A \) to \( B \).

The above definition corresponds to the concept of controllability for control systems of the form

\[ \dot{x} = f(x, u), \quad x(0) = x_0, \]

where the control \( u(t) \) plays the role of a switching signal.

Particularly, linear switched systems, \(^3\) of the form

\[ \dot{x}(t) = A_{\sigma(t)}x(t), \quad x(0) = x_0 \neq 0, \]

where \( A_1 \ldots A_n \) are given matrices, are of our interest.

More precisely, the case of \( 2 \times 2 \) matrices with both subsystems having pure imaginary eigenvalues is studied. This case has already been considered in Derevianko and Korobov, \(^{30}\) and the main difference lies in using GAC as a suitable space for geometric operations with the ellipses, for elementary notions see Section 3.

First, consider the problem of spring pendulum oscillation under the condition of absence of external and friction forces

\[ \ddot{x} = -kx, \]

with a switchable stiffness coefficient \( k > 0 \), which changes the value from \( k_1 \) by joining and removing an additional spring with a stiffness coefficient \( k_2 \). Two cases can be considered. If the springs are connected in parallel, the parameter \( k \) of the system switches between \( k = k_1 \) and \( k = k_1 + k_2 \). If the connection is series, the parameter \( k \) of the system switches between \( k = k_1 \) and \( k = \frac{k_1 k_2}{k_1 + k_2} \).

Let us rewrite the differential equation of the pendulum oscillations as a switched system:

\[ \dot{x}(t) = A_i x(t), \quad A_i \in \text{Mat}_2(\mathbb{R}), \quad i = 1, 2. \]

Without loss of generality, let us assume that we start and end with the first system \( i = 1 \). Suppose that two nonzero points (initial \( A(x_1, y_1) \) and final \( B(x_2, y_2) \)) are given.

By rewriting the coordinates of \( x \) as \( (x, y) \), we get that the solutions of the system

\[ \dot{x}(t) = A_1(x(t)) \]

can be determined in the form

\[
\begin{align*}
    x(t) &= \gamma_1 \sin \left( \sqrt{\alpha_1} t \right) + \gamma_2 \cos \left( \sqrt{\alpha_1} t \right), \\
    y(t) &= \sqrt{\alpha_1} \gamma_1 \sin \left( \sqrt{\alpha_1} t \right) - \sqrt{\alpha_1} \gamma_2 \cos \left( \sqrt{\alpha_1} t \right). \quad \gamma_1, \gamma_2 \in \mathbb{R}.
\end{align*}
\]

The trajectories for the system with pure imaginary eigenvalues are given by ellipses. If \( \text{Tr}A_i = 0 \), i.e., the case of the spring pendulum without damping, for example,

\[ A_i = \begin{bmatrix} 0 & 1 \\ -a_i & 0 \end{bmatrix}, \quad a_i \in \mathbb{R}^+, \]

we have an axis-aligned ellipse, while, if \( \text{Tr}A_i \neq 0 \), then the ellipses are rotated and the given switched system is equivalent to the equation describing the oscillatory system with damping. In this case, the rotation angle can be calculated by means
of the conic matrix (5) in terms of geometric algebra:

$$\theta = \begin{cases} \arctan \left( \frac{1}{q_{12}} \left( q_{22} - q_{11} - \sqrt{(q_{11} - q_{22})^2 + q_{12}^2} \right) \right), & q_{12} \neq 0 \\ 0, & q_{12} = 0, \ q_{11} < q_{22} \\ \frac{\pi}{2}, & q_{12} = 0, \ q_{11} > q_{22} \end{cases}$$

6 | ALGORITHM FOR A SWITCHING PATH CONSTRUCTION

In the following, we describe the algorithm for finding a control of a switched system, i.e., finding a path composed of the systems’ integral curves from the initial point \( A \) to the endpoint \( B \) such that the number of switches is minimal. Consider the case \( n = 2 \), meaning that only two systems are included, and both the starting and the final ellipse belong to the same family. To apply the GAC based calculations, it is necessary to get the exact GAC form of the representatives of both families of ellipses. Thus, the system of ODEs is solved numerically (e.g., by the Runge-Kutta method) with the initial condition at the starting point \( A \). This will give us a set of points representing the initial ellipse. After applying the GAC conic fitting algorithm, we get the ellipse in IPNS representation. Note that, according to Loučka and Vašík, the algorithm may be further specified by prescribing the resulting ellipse to be axis-aligned and with its center placed in the origin. This makes the initial trajectories very precise.

**Algorithm 1** Algorithm for a switching path construction

1. Get \( A, B \), the starting and final point, respectively, i.e. get their conformal embedding \( C(A), C(B) \) to GAC, (2).
2. Find the IPNS representation of the initial ellipse \( E^1_1 \) by conic fitting algorithm. Denote its semiaxis by \( a \) and \( b \).
3. Find the final ellipse \( E_f \) by Algorithm 2.
4. Find the first intermediate ellipse \( E^1_2 \) by Algorithm 3. Note that the lower index indicates to which system the ellipse belongs.
5. If \( E_f \cap E^1_2 \neq \emptyset \), then find the intersection points of all ellipses, get the path from \( A \) to \( B \) by choosing the nearest point with respect to the path evolution. This will switch to the final ellipse.
6. If \( E_f \cap E^1_2 = \emptyset \), then calculate the scaling parameter \( SP \) according to Proposition 2. By scaling \( E^1_1, E^1_2 \) using \( SP \), get a new pair of circumscribed ellipses

$$E^{i+1}_2 := \text{scale}(E^1_2, SP), \quad E^{i+1}_1 := \text{scale}(E^1_1, SP).$$

**Algorithm 2** Final ellipse construction

**Inputs:** Two co-centric ellipses, where one is the scaled copy of the other

**Output:** Final ellipse \( E_f \)

1. Construct a line \( l \) passing through the points \( n^+ \) and \( e_2 \):

$$l = e_2 \wedge n^+ \wedge n^+ \wedge n_\wedge \wedge n_\wedge.$$

2. Find the intersection points \( C = E_1 \cap l, \ B = E_2 \cap l \) of the line and both ellipses, i.e. solve a quadratic equation in a Euclidean space.
3. The scale parameter between the ellipses is

$$SP = \frac{\| n^+ \cdot C(B) \|}{\| n^+ \cdot C(C) \|}.$$

4. Construct the final ellipse: \( E_f = S_+ S_- s_x E^1_1 s_x \tilde{s}_x \tilde{s}_- \).
As a result, the algorithm composed of the above algorithms provides a sequence of switching points as well as a sequence of trajectories in GAC. For an example of the resulting path, see Figure 6. We will demonstrate the functionality of the whole algorithm on the following two examples, which form a generalization of the system from Coppel. 31, p. 6

The following steps are numerical with respect to the steps of Algorithm 1,

**Example 4.** an oscillatory system without damping. Consider the switched system \( \dot{x} = A_t x \), where \( A_1 = \begin{pmatrix} 0 & 1 \\ -2 & 0 \end{pmatrix} \), \( A_2 = \begin{pmatrix} 0 & 1 \\ 1 & 2 \end{pmatrix} \). We choose the starting point \((2, 5)\), and we need to find the way to another chosen point \((12, 22)\).

1. Starting point \( A = (2, 5) \), final point \( B = (12, 22) \).
2. The IPNS representation of the initial ellipse and its characteristics are (Python output)

\[
\begin{align*}
-(2.45181^\text{e3}) - (0.03553^\text{e4}) - (2.23861^\text{e6}) + (0.0353^\text{e7}) \\
text{ellipse:} \\
S = [0.0, 0.0] \\
a = 5.744562666461688 \\
b = 4.062019202259119 \\
teta = 89.9999999996051\\n\end{align*}
\]

3. The IPNS representation of the final ellipse and its characteristics are

\[
\begin{align*}
-(11.12057^\text{e3}) - (0.00751^\text{e4}) - (11.07552^\text{e6}) + (0.00751^\text{e7}) \\
text{ellipse:} \\
S = [0.0, 0.0] \\
a = 27.184554438097937 \\
b = 19.22238278669092 \\
teta = 89.9999999996051\\n\end{align*}
\]

4. The IPNS representation of the first intermediate ellipse and its characteristics are

\[
\begin{align*}
-(3.392^\text{e3}) + (0.02513^\text{e4}) - (3.24125^\text{e6}) - (0.02513^\text{e7}) \\
text{ellipse:} \\
S = [0.0, 0.0] \\
a = 8.124038404526583 \\
b = 5.744562646460197 \\
teta = 0.0\\n\end{align*}
\]

5. Scaling parameter \( SP = 2.0 \)

6. The set of the ellipses used can be seen in Figure 5, while the set of the switching points is 
\{\( (0; -5.74456), (8.12404; 0), (0; 11.48913), (16.24808; 0), (0; -22.97825), (23.2054167141; 12.86501593890354) \)\}.
This is a result of Python code written in module clifford according to the algorithm in Section 6. Note that the red points in Figure 5, left, form a set of points generated by the Runge-Kutta method and you can see the fitted conic, too.

**Example 5.** Consider the switched system $\dot{x} = A_t x$, where $A_1 = \begin{pmatrix} 0 & 1 \\ -2 & 0 \end{pmatrix}$, $A_2 = \begin{pmatrix} 1 & 1 \\ -2 & 1 \end{pmatrix}$. The initial point is (2, 5), and we need to find a path to the point (30, 22). Both of the matrices have pure imaginary eigenvalues, so the system is switching between ellipses. Ellipses of the second family are rotated. That means that the second subsystem describes one of the cases of the oscillatory system with damping. These can also produce other types of conics, e.g., spirals, but that case is not the point of our interest. The set of the switching points is $\{(-2.88653912573; -3.697011208397), (-4.23042514649; 8.76807759024), (5.98270815467; -7.662511450902), (-8.76807759024; 18.1729216252), (-15.88149952097; -12.39990012429), (-11.14111993673; 43.01897176660)\}$, and the switching path calculated in Python module clifford can be seen in Figure 6.

## 7 Conclusion and Discussion

A novel algorithm for finding the optimal control of a switched dynamical systems with purely imaginary eigenvalues of both matrices has been proposed. Namely, we have constructed a switching path consisting of circumscribed ellipses with switching points exactly at the contact points, which guaranteed a minimal number of switching points. The whole construction was implemented in Python module clifford using its standard commands and functions for Geometric Algebra for Conics (GAC). The main advantage of GAC consists in the possibility of simple representation of transformed objects (for example rotated and scaled ellipses) together with the possibility of effective circumscribed ellipse construction. Let us stress that our geometric approach eliminates the need for any solver and, therefore, it minimizes numerical errors.
We demonstrated a complete geometric procedure for two families of axis-aligned ellipses in Examples 3 and 4, where we demonstrated symbolic and Python calculations, respectively. In addition, we used the fact that GAC contains a two-dimensional conformal geometric algebra, in which our calculations were completed. This case corresponds to an oscillatory switched system without damping. Our approach also applies to damped systems, where the integral curves are formed by rotated ellipses, i.e. non-axis-aligned, which we demonstrated in Examples 1 and 5. Even in this case, no solver was needed because, in the system of two quadratic equations describing the intersections of the ellipses, we replaced an ellipse equation by a line equation, which reduced the degree and allowed an analytic solution. Note that both approaches exploit the elegance of conic manipulation in GAC by constructing a pair of lines containing the intersecting points and circumscribed ellipses simply calculated by GAC scaling with a factor determined according to Proposition 2.

Let us point out that even the preparation of the initial trajectories is highly geometric. Fitting a conic with prescribed properties in GAC eliminates an error in the numerical solution to our switched system. Indeed, all the trajectories will be precisely of a given type, i.e., co-centered and axis-aligned. The consequent GAC transformations do not change these properties and do not input any numerical errors. Indeed, the only place for a rounding error is the calculation of fractions and square roots because all operations in GAC may be converted to sums of products, and thus, only the problem of computer representation of numbers is involved, meaning that one has to check that no two numbers with very different magnitudes are summed.

Our algorithm generates a switching path that is optimal with respect to the number of switching points. Indeed, by constructing circumscribed ellipses, we minimize the number of trajectories involved and, therefore, the number of switching points. Finally, let us note that we applied our algorithm on systems that are controllable; i.e., the existence of a trajectory connecting the initial and final points is guaranteed.

ACKNOWLEDGEMENTS

The authors were supported by a grant of Brno University of Technology, No. FSI-S-20-6187.

CONFLICT OF INTEREST

This work does not have any conflict of interest.

ORCID

Petr Vašík https://orcid.org/0000-0001-6725-7378

REFERENCES

1. Vidyasagar M. *Nonlinear System Analysis*. 2nd ed. New Jersey: Prentice Hall, Eaglewood Cliffs; 1993.
2. Sun Z, Sam Ge S. *Stability Theory of Switched Dynamical Systems*. New York: Springer-Verlag London; 2011.
3. Colaneri P. *Analysis and Control of Linear Switched Systems*. Italy: Politecnico di Milano; 2018.
4. Lin Y, Sontag ED, Wang Y. A smooth converse Lyapunov theorem for robust stability. *SIAM J Control Optim*. 1996;34(1):124-160.
5. Jiang Z-P, Wang Y. A converse Lyapunov theorem for discrete-time systems with disturbances. *Syst Control Lett*. 2002;45(1):49-58.
6. Hrdina J, Návrat A, Vašík P. Geometric algebra for conics. *Adv Appl Clifford Algebras*. 2018;28(66).
7. Hildenbrand D. *Foundations of Geometric Algebra Computing*. Springer Science and Business Media; 2013.
8. Hildenbrand D. *Introduction to geometric algebra computing*. USA: Chapman and Hall/CRC; 2018.
9. Easter RB, Hitzer E. Double conformal geometric algebra. *Adv Appl Clifford Algebras*. 2017;27(3):2175-2199.
10. Hrdina J, Návrat A, Vašík P. Conic fitting in geometric algebra setting. *Adv Appl Clifford Algebras*. 2019;29(72).
11. Lyapunov A. *General Problem of Motion Stability*. Kharkov; 1898.
12. Liberzon D. *Switching in Systems and Control*. Boston: Birkhauser; 2003.
13. Ding X, Liu X. On stabilizability of switched positive linear systems under state-dependent switching. *Appl Math Comput*. 2017;307:92-101.
14. Han Y, Zhao Y, Wang P. Finite-time rate anti-bump switching control for switched systems. *Appl Math Comput*. 2021;401:126086.
15. Nasir UA. *Dynamic Systems and Control With Applications*. Canada: University of Ottawa; 2006.
16. Seidman T. Optimal control of a diffusion/reaction/switching system. *Evol Equ Control Theory [Electronic Only]*. 2013;4:723.
17. Hankins MJ, Nagy T, Kiss IZ. Methodology for a nullcline-based model from direct experiments: Applications to electrochemical reaction models. *Comput Math Appl*. 2013;65(10):1633-1644. Grasping Complexity.
18. Izhikevich EM. *Dynamical Systems in Neuroscience*. London: Massachusetts Institute of Technology; 2007.
20. Lazard D. Gröbner Bases, Gaussian Elimination and Resolution of Systems of Algebraic Equations. Berlin, Heidelberg: Springer Berlin Heidelberg; 1983.
21. Hestenes D. Space-Time Algebra. New York: Gordon and Breach; 1966.
22. Gonzalez-Jimenez L, Carbajal-Espinosa O, Loukianov A, Bayro-Corrochano E. Robust pose control of robot manipulators using conformal geometric algebra. Adv Appl Clifford Algebras. 2014;24(2):533-552.
23. Hrdina J, Návrat A, Vašík P, Matoušek R. Geometric algebras for uniform colour spaces. Math Meth Appl Sci. 2018;41:4117-4130.
24. Perwass C. Geometric Algebra With Applications in Engineering. Springer Verlag; 2009.
25. Dorst L, Fontijne D, Mann S. Geometric Algebra for Computer Science: An Object-Oriented Approach to Geometry. Rev. ed. Burlington, Mass.: Morgan Kaufmann Publishers, Morgan Kaufmann series in computer graphics; 2007.
26. Lounesto P. Clifford Algebra and Spinors. 2nd ed. Cambridge: CUP; 2006.
27. Richter-Gebert J. Perspectives on Projective Geometry. Berlin, Heidelberg: Springer Berlin Heidelberg; 2011.
28. Loučka P, Vašík P. Algorithms for multi-conditioned conic fitting in geometric algebra for conics. Advances in Computer Graphics - 38th Computer Graphics International Conference, CGI 2021, Virtual Event, September 6-10, 2021, Proceedings, Lecture Notes in Computer Science, vol. 13002: Springer; 2021:645-657.
29. Byrtus R, Derevianko A, Vašík P, Hildenbrand D, Steinmetz C. On specific conic intersections in GAC and symbolic calculations in gaalopweb. Adv Appl Clifford Algebras. 2022;32:1-21.
30. Derevianko AI, Korobov VI. Controllability of the given switched linear system of special type. Visnyk of V.N.Karazin Kharkiv Natl Univ Ser Math, Appl Math Mechanics. 2019;89:93-101.
31. Coppel W. Stability and Asymptotic Behavior of Differential Equations. Boston: D. C. Heath and Company; 1965.

How to cite this article: Derevianko A, Vašík P. Solver-free optimal control for linear dynamical switched system by means of geometric algebra. Math. Meth. Appl. Sci. 47 (2024), 1274-1288, DOI 10.1002/mma.8752.