



Research article

Dynamic construction of a family of octic curves as geometric loci

Thierry Dana-Picard^{1,*} and Tomás Recio²

¹ Department of Mathematics, Jerusalem College of Technology, Havaad Haleumi Street 21, Jerusalem 9116011, Israel

² Universidad Antonio de Nebrija, C/ Santa Cruz de Marcenado 27, 28015 Madrid, Spain

* **Correspondence:** Email: ndp@jct.ac.il; Tel: +97226751056; Fax: +97226751285.

Abstract: We explore the construction of curves of degree 8 (oactics) appearing as geometric loci of points defined by moving points on an ellipse and its director circle. To achieve this goal we develop different computer algebra methods, dealing with trigonometric or with rational parametric representations, as well as through implicit polynomial equations, of the given curves. Finally, we highlight the involved mathematical or computational issues arising when reflecting on the outputs obtained in each case.

Keywords: plane curves; geometric locus; octic curve; interactive construction; dynamic geometry; GeoGebra; symbolic computation; Maple; implicitization

Mathematics Subject Classification: 14H50, 51M15, 13P10

1. Introduction

In this paper we report on our experimentation, carried out by merging a dynamic geometric system (DGS) and a computer algebra system (CAS), dealing with plotting the geometric locus described by a certain family of octic curves, that can be motivated as a mathematical construct related to modeling planetary orbits. Actually, our final goal is to exhibit the challenging mathematical issues that are involved in this quite simple context, showing the need for DGS/CAS cooperation, and the rising of yet unsolved mathematical questions, dealing with the interrelation of specialization (required sometimes to simplify input) and elimination, or with the appearance in some circumstances of “strange” components in the locus plot.

Plane curves have been studied for centuries, and large catalogues are available, either printed or online. Numerous curves are presented as stand-alone examples, such as the Witch of Agnesi, but there exist also families of curves, depending on one parameter or more. Even so, curves known as stand-alone appear also in families. For example, the Bernoulli lemniscate appears also as a specialization of

Cassini ovals. The curves, and the families of curves, may be defined through implicit, or parametric, or polar equations, or in a concurrent way, via more than one of these tools: e.g. the aforementioned example has both a trigonometric parametric presentation and an implicit equation. The latter is a polynomial equation of degree 3, making the curve an algebraic curve. Actually it is a rational cubic. There exist also large families of curves, for which the parametric equations or the implicit equation contain a parameter. Famous examples are epicycloids, hypocycloids, epitrochoids, hypotrochoids, given by trigonometric parametric equations.

Two processes in reversed directions may be at work, but not always simultaneously:

- If a plane curve C is given by a parametric presentation, then implicitization might be a core issue. For example, if algebraic manipulation transforms the given parametrization into a rational one, a presentation with polynomials is afforded. These polynomials generate an ideal in a polynomial ring over the field of coefficients. Using methods based on Gröbner bases and elimination (see [4]), an implicit equation may be derived for the curve C . This has been extensively used in [6]. We must mention that in some cases, even with a powerful CAS and a powerful computer, implicitization revealed impossible, not for theoretical but for technical reasons [11, 13].
- Deriving a rational parametrization from an implicit equation of a curve C is a non trivial issue. In some instances, as the one studied in [5] it is possible, but not in most cases. For a general presentation of the problem and available methods with a Computer Algebra approach, we refer to [18].

Technology is a useful tool to explore plane curves. Indeed, visualization of a curve in a screen may be quite faithful, as the mathematical situation under study is a 2D situation. Moreover, a Dynamic Geometry System (DGS) offers two main dynamical features:

- Dragging of points [17]: the display can be changed continuously using the mouse. Here the changes may not be uniform, as the process is interactive.
- Slider bars: the construction of an object depending on a parameter can be analyzed when the parameter changes in a uniform way, i.e. receiving values in a pre-defined arithmetic sequence.

The two features offer different kinds of man-and-machine interaction. Moreover, some Dynamic Geometry Systems include the direct computation and/or display of geometric loci, involving both “mover” and “tracer” points, see [1] for a summary description of loci computation in DGS. But the obtained display does not have to be, in general, topologically reliable, as achieving such requirement involves complex symbolic computation algorithms, see the recent work [12] and the references therein, pointing out to over 30 years of research on this issue. Computing the topology of families of curves depending on a parameter – as it is often the case when dealing with locus – is even a more challenging issue, addressed, for example, in [2, 3], but its applicability in our context is very limited, since the kind of curves we will work with here is not included in the first reference, and, for profiting from the second one we would have to handle implicit equations, something that often turns out to be too involved.

Indeed, let us point out that the algebraic manipulations requested to switch between parametric equations and implicit equations may be performed by hand just in very simple cases, but they are

generally performed using the Gröber package implemented in a Computer Algebra System (CAS). At the beginning, CAS and DGS were conceived as different systems. Finding ways towards a dialog, a collaboration between the two kinds of systems has been explored for a long time; see [15, 16, 9]. Some time ago, a CAS has been implemented into the GeoGebra DGS system [14]. Nevertheless, it is sometimes not powerful enough for implicitization, as we will show in subsection 2.2.

Therefore, in what follows, we use both GeoGebra and Maple for the exploration of algebraic varieties in \mathbb{R}^2 and in \mathbb{R}^3 . Of course, when working in 3D, visualization rises specific problems, and in this case our main contribution here is devoted to algebraic work, whence to the CAS, as in [7].

In [8], we described a simple model of planetary orbits and mathematical constructs based on them, namely the trajectory of the midpoint of two planets. This midpoint has no true physical meaning, but its dynamics can be a trigger to study some epi and hypocycloids, letting some parameters vary according the precision of the orbital data in use. An extension has been made also when the two mobile objects are moving in opposite directions (a situation which occurs when a satellite is launched on a retrograde orbit). The movements of the two planets have different angular velocity.

The present paper is another contribution to the construction of plane curves given by trigonometric parametric equations. This time, we consider (depending on a parameter k) an ellipse $x^2 + k^2y^2 = 1$, its director circle, and one point on each with the same angular velocity. In the next section 2 we approach the locus of the corresponding midpoints, in two different ways: using a trigonometric parametrization of the ellipse and director circle, and using a rational parametrization of the same curves, and then implicitizing. Figure 4 shows already some surprising facts about the path followed by the locus point in terms of the values of the parameter u . On the other hand subsection 2.2 deals already with some more involved (some of them not yet well understood) issues, dealing with the existence of three absolutely irreducible components for each octic, with only one of them coinciding with the one displayed in the previous subsection.

Aiming to clarify this situation, section 3 starts with a strict polynomial equation setting, through the implicit equations of the ellipse and director circle, but recalling that they depend on a parameter k . Now the construction of the implicit equation of the locus for each particular value of k can be achieved through different alternatives, such as including k as a variable in the input ideal, computing the locus, and then specializing the result for specific values of k ; or specializing the input ideal for values of k and then obtaining the locus. We will check that, in this particular case, both approaches lead to the same, absolutely irreducible, equation of degree 8 in the locus coordinates (with k also of degree 8): the octic.

The paper ends with a short summary of the different results and open problems that this apparently simple context has yield.

2. An algorithmic exploration leading to the discovery of an octic curve

2.1. The parametric construction

We consider an ellipse \mathcal{E} whose equation is $x^2 + k^2y^2 = 1$, where k is a positive real parameter. Its director circle \mathcal{D} has equation $x^2 + y^2 = 1 + 1/k^2$. Recall that the director circle is the orthoptic curve of the ellipse, i.e. the geometric locus of points through which passes a pair of tangents to \mathcal{E} which are perpendicular. This can be easily checked graphically using the DGS, as in Figure 1. The slider enables an automated check that the construct is valid for any $k > 0$, and dragging the point A along

the circle \mathcal{D} shows perpendicularity of the tangents from any point A .

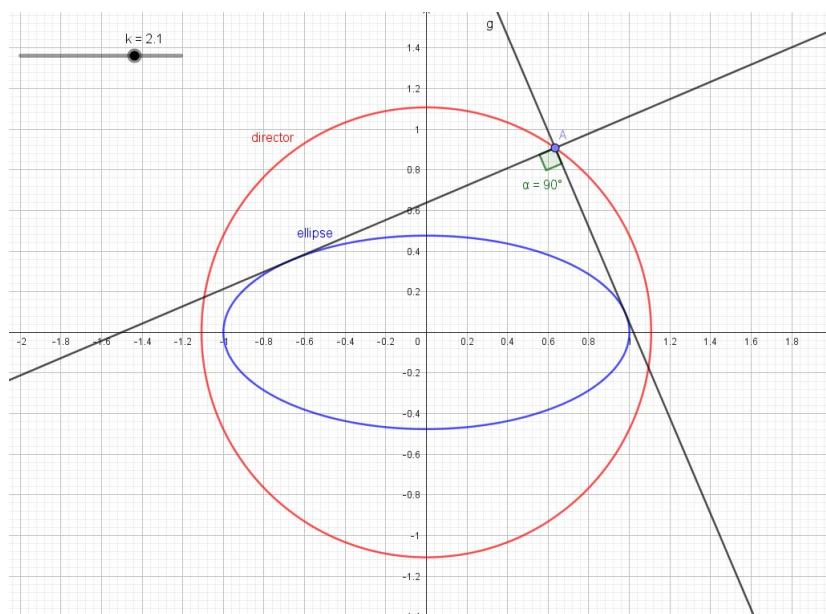


Figure 1. An ellipse and its director circle.

A general point E on \mathcal{E} has coordinates

$$(x_E, y_E) = \left(\cos u, \frac{1}{k} \sin u \right), \quad u \in \mathbb{R}. \quad (2.1)$$

Let O be the origin. The line (OE) intersects the circle \mathcal{D} at 2 points F and G whose coordinates are as follows:

$$\begin{cases} x_F = \sqrt{\frac{(k^2+1)\cos^2 u}{k^2 \cos^2 u - \cos^2 u + 1}} \\ y_F = \frac{\sin u}{(k \cos u)} \sqrt{\frac{(k^2+1)\cos^2 u}{1+(k^2-1)\cos^2 u}} \end{cases} \quad (2.2)$$

and

$$\begin{cases} x_G = -\sqrt{\frac{(k^2+1)\cos^2 u}{k^2 \cos^2 u - \cos^2 u + 1}} \\ y_G = -\frac{\sin u}{(k \cos u)} \sqrt{\frac{(k^2+1)\cos^2 u}{1+(k^2-1)\cos^2 u}} \end{cases} \quad (2.3)$$

Note that the formulas express the fact that F and G are symmetric about the origin.

Now consider the midpoints M and N of the segment EF and EG respectively; their coordinates are given by:

$$\begin{aligned} (x_M, y_M) &= \left(\frac{1}{2}(x_E + x_F), \frac{1}{2}(y_E + y_F) \right) \\ (x_N, y_N) &= \left(\frac{1}{2}(x_E + x_G), \frac{1}{2}(y_E + y_G) \right) \end{aligned} \quad (2.4)$$

A snapshot of a GeoGebra animation is displayed in Figure 2. It is possible to obtain both components looking continuous by changing the parameters of the animation (we mean choosing a lower speed, and a smaller increment on the values of the parameter). Changes in other parameters, enabled with slider bars, provide other shapes for the desired locus. In order for the display for new values of the parameters to be readable, the user has to erase the previous plots.

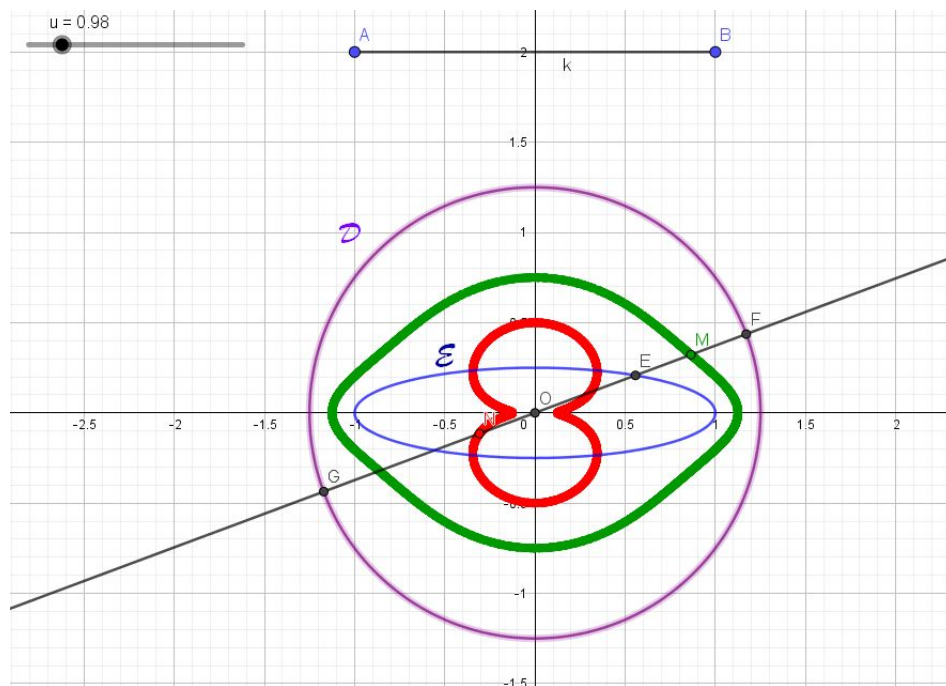


Figure 2. A dynamic construction of the loci of the midpoints.

For this example, the automated **Locus** command of GeoGebra did not work; therefore, we could construct the locus only using the animation based on the slider for the parameter u . Trying other values of the parameter k requires running the animation again and again.

In order to look for an implicit equation (if possible) of the locus of F and the locus of G when u varies, we transfer the work to the CAS. A surprise is waiting with observation of the plots. It has to do with the choice of the parametrization of the ellipse and its director circle. Here is the source code for the beginning of the Maple session:

```

ellps := k^2*y^2 + x^2 = 1;
dircrcl := x^2 + y^2 = 1 + 1/k^2;
x_E := cos(u);
y_E := sin(u)/k;
l_E := y = y_E*x/x_E;
pts := solve({dircrcl, l_E}, {x, y});
parampts := allvalues(%);
x_F := simplify(rhs(parampts[1][1]));
y_F := simplify(rhs(parampts[1][2]));

```

```
x_G := simplify(rhs(parampts[2][1]));
y_G := simplify(rhs(parampts[2][2]));
```

The solve command produces generally an output containing the place holder *RootOf*, which is resolved into different components by **allvalues**. Even then, the expressions may be quite complicated, whence the usage of **simplify**. In the present case, the resulting expressions are not really simpler; they are built on sine and cosine instead of tangent. In order to display the construct, we use the following code:

```
for p from 1/2 by 1/2 to 6 do
  p;
  plot({[subs(k = p, x_E), subs(k = p, y_E), u = 0 .. 2*Pi],
        [subs(k = p, x_F), subs(k = p, y_F), u = 0 .. 2*Pi],
        [subs(k = p, x_G), subs(k = p, y_G), u = 0 .. 2*Pi],
        [subs(k = p, x_M), subs(k = p, y_M), u = 0 .. 2*Pi],
        [subs(k = p, x_N), subs(k = p, y_N), u = 0 .. 2*Pi]});
end do
```

Figure 3 shows the output for some values of the parameter. The case $k = 1$ is very special, it provides 2 concentric circles. In the other cases, the output presents 2 components with different topologies. It is necessary to compute an implicit equation for the obtained curve and to (hopefully) factorize the obtained polynomial in order to distinguish algebraically the two components.

The surprise comes from a trial to master the colors of the different plots, by inserting a color option into the commands of the loop above, as follows:

```
for p from 1/2 by 1/2 to 6 do
  p;
  ellpsplot := plot([subs(k = p, x_E), subs(k = p, y_E), u = 0 .. 2*Pi],
    color = blue, thickness = 3);
  plotF := plot([subs(k = p, x_F), subs(k = p, y_F), u = 0 .. 2*Pi], color = red);
  plotG := plot([subs(k = p, x_G), subs(k = p, y_G), u = 0 .. 2*Pi], color = green);
  plotM := plot([subs(k = p, x_M), subs(k = p, y_M), u = 0 .. 2*Pi], color = orange);
  plotN := plot([subs(k = p, x_N), subs(k = p, y_N), u = 0 .. 2*Pi], color = yellow);
  display(ellpsplot, plotF, plotG, plotM, plotN);
end do;
```

The plots have been named separately and the final output obtained by a display command, in order to follow precisely after what happens, i.e. from which line comes which part of the plot. The original ellipse is in blue, the director circle is divided into two parts, as it is plotted as the geometric locus of F and of G .

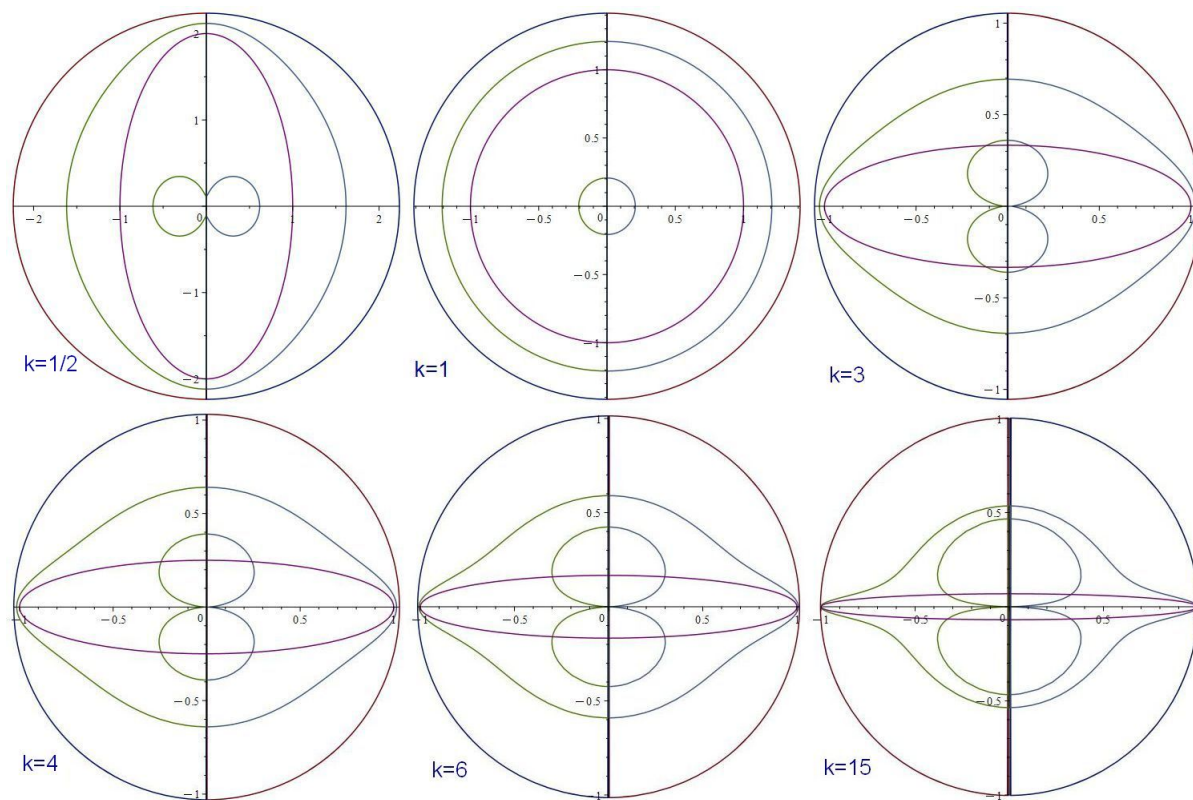


Figure 3. Several plots from the Maple exploration.

As a whole, it was expected that the resulting curves will be symmetric about the coordinate axes, the specific symmetries being a result of the role exchange between F and G when passing from the left half-plane to the right half-plane, and vice versa. Indeed, the x -coordinate of F is always taken as the positive root of some expression, while its y -coordinate is positive if both $\sin(u), \cos(u)$ have the same sign, and negative otherwise. Likewise, the x -coordinate of G is always taken as negative, while its y -coordinate is negative if both $\sin(u), \cos(u)$ have the same sign, and positive otherwise. Thus, when angle u takes point E from the left half-plane to the right half-plane, suddenly F is replaced by G (as the x -coordinate becomes negative) and F jumps to the symmetrical position of G with respect to the origin.

Some of the outputs are displayed in Figure 4.

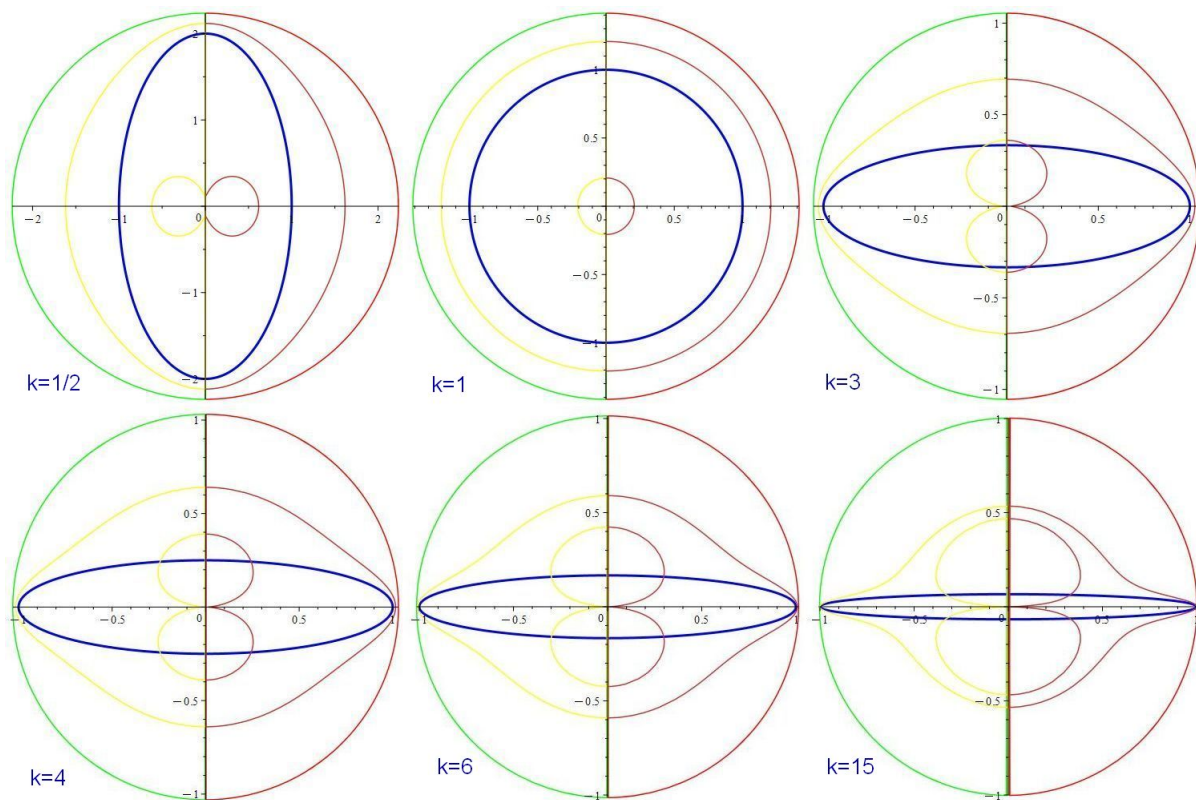


Figure 4. Several plots from the Maple exploration with prescribed colors.

2.2. Rationalization and implicitization

Recall that for every $u \in \mathbb{R}$, there exist $v \in \mathbb{R}$ such that $\cos u = \frac{1-v^2}{1+v^2}$ and $\sin u = \frac{2v}{1+v^2}$. By substitution into Eq (2.4), we obtain:

$$\begin{cases} x_M = \frac{1}{2(v^2+1)} \left((v^2+1) \sqrt{\frac{(v^2-1)^2(k^2+1)}{(v^2-1)^2k^2+4v^2}} - (v^2-1) \right) \\ y_M = -\frac{v}{k(v^4-1)} \left((v^2+1) \sqrt{\frac{(v^2-1)^2(k^2+1)}{(v^2-1)^2k^2+4v^2}} - (v^2-1) \right) \end{cases} \quad (2.5)$$

By symmetry, we obtain similar equations for N . In order to transform these equations into polynomials, we used the following code:

```
x*denom(xx_M) + v^2 - 1 = numer(xx_M) + v^2 - 1:
simplify(%^2):
p1 := numer(lhs(%) - rhs(%)):
y*denom(yy_M) - v*(v^2 - 1) = numer(yy_M) - v*(v^2 - 1):
simplify(%^2):
p2 := numer(lhs(%) - rhs(%)):
```


The computed polynomials are the following:

$$\begin{aligned}
 P_1(x, y, k, v) &= 4k^2v^8x^2 + 4k^2v^8x - 8k^2v^6x - 4k^2v^6 - 8k^2v^4x^2 - v^8 + 16v^6x^2 + 16v^6x + 8k^2v^4 + 4v^6 \\
 &\quad + 32v^4x^2 + 8k^2v^2x - 4k^2v^2 + 4k^2x^2 - 6v^4 + 16v^2x^2 - 4k^2x - 16v^2x + 4v^2 - 1 \\
 P_2(x, y, k, v) &= k^4v^{12}y^2 - 2k^4v^{10}y^2 - 2k^3v^{11}y - k^4v^8y^2 + 4k^2v^{10}y^2 + 6k^3v^9y + 4k^4v^6y^2 - 4k^3v^7y \\
 &\quad - 8kv^9y - k^4v^4y^2 - 4k^2v^8 - 8k^2v^6y^2 - v^{10} - 4k^3v^5y + 8kv^7y - 2k^4v^2y^2 + 8k^2v^6 + 4v^8 \\
 &\quad + 6k^3v^3y + 8kv^5y + k^4y^2 - 4k^2v^4 + 4k^2v^2y^2 - 6v^6 - 2k^3vy - 8kv^3y + 4v^4 - v^2
 \end{aligned}
 \tag{2.6}$$

The polynomials $P_1(x, y, v, k)$ and $P_2(x, y, v, k)$ generate an ideal $J \subset \mathbb{R}[x, y, v, k]$. Elimination of parameter v alone did not work with the CAS, but it is possible to eliminate v for various values of the parameter k , to provide some hints about the elimination ideal (bearing in mind that, in general, the specialization of the elimination, for some values of the parameter, does not necessarily have to coincide with the elimination of the specialized ideal, but it is an ideal contained on it). Here is the Maple code:

```

for p from 1/2 to 10 do
  J[p] := <subs(k = p, p1), subs(k = p, p2)>;
  JE[p] := EliminationIdeal(J[p], {x, y});
  gen[p] := Generators(JE[p])[1];
  cc := implicitplot({subs(k = p, dircrcl) = 0, subs(k = p, ellps) = 0},
    x = -4 .. 4, y = -4 .. 4, color = navy);
  fac[p] := evala(AFactor(gen[p])); #absolute factorization
  facts[p] := factors(gen[p]); #factorization over the rationals
  comp[p][1] := facts[p][2][1][1];
  comp[p][2] := facts[p][2][2][1];
  comp[p][3] := facts[p][2][3][1];
  c[1]:= implicitplot(subs(k = p, comp[p][1]) = 0, x = -4 .. 4, y = -4 .. 4,
    color = magenta);
  c[2]:= implicitplot(subs(k = p, comp[p][2]) = 0, x = -4 .. 4, y = -4 .. 4,
    color = green);
  c[3]:= implicitplot(subs(k = p, comp[p][3]) = 0, x = -4 .. 4, y = -4 .. 4,
    color = brown);
  display(cc, c[1], c[2],c[3]);
end do

```

Of course, the values of the local variable p can be modified. We ran this code not only for the values of p appearing here, but for other ones as it is clear from the illustrations). We should mention that for $p = 1$ the computation fails. The **factors** command provides more information than the regular **factor** command. For every positive value of k , the elimination ideal JE is generated by a unique polynomial, denoted above by gen , of degree 19. The output of the **factors** command says that over the rationals this polynomial is the product of a factor x^3 and 2 factors of degree 8. We call the corresponding curves *octic curves*, or simply *octics*.

This is not enough to check that the obtained varieties (the y -axis, and the two degree 8 curves) are irreducible over the complexes. For this, another command has to be applied: the **AFactor** command,

for absolute factorization, i.e. over the field \mathbb{C} , is a place holder, and its output is resolved by the command **evala**. Thus, we have addressed this (technical) issue, which requires human observation, in order to check validity of the output, by comparing the outputs of the rows

```
fac[p] := evala(AFactor(gen[p])); #absolute factorization
facts[p] := factors(gen[p]); #factorization over the rationals
```

The conclusion is that both outputs yield the same factorization (obviously, a very particular result, just holding in this case, as both factorizations are, in principle, different), namely, the two factors of degree 8 and the factor x^3 . But the output of the **factor** command does not keep the order of the factors for different values of the parameter. The factor x^3 can appear in any position in the list. The 3 components can be plotted simultaneously, using what is performed in the 3 subsequent rows, and the user has to identify each component. Comparing with the outputs of the work in parameterized setting (in previous section), it appears that in each case, only one plot of an octic is relevant. Identifying a pattern may be possible, but not useful, as here we show examples for p being either an integer or half-integer, but other cases exist of course. In any case, one component only coalesces with what has been obtained, either with the GeoGebra exploration or with Maple in the parametric setting.

Figure 5 shows the output for some values of k . The different colors are a consequence of the order of the factors obtained by the command *factors*. Note also that for $k = 15$, more irreducible components seem to exist. The algebraic computations, in the next section, show that this is not the true situation. What appears on the display is a consequence of the fact that a software has often difficulties to plot close to a singular point. This plotting issue has been discussed in [19]. In conclusion: a detailed comparison between this output with the output of the previous subsection, and the interpretation of coincidences/differences, seems to deserve further, future work.

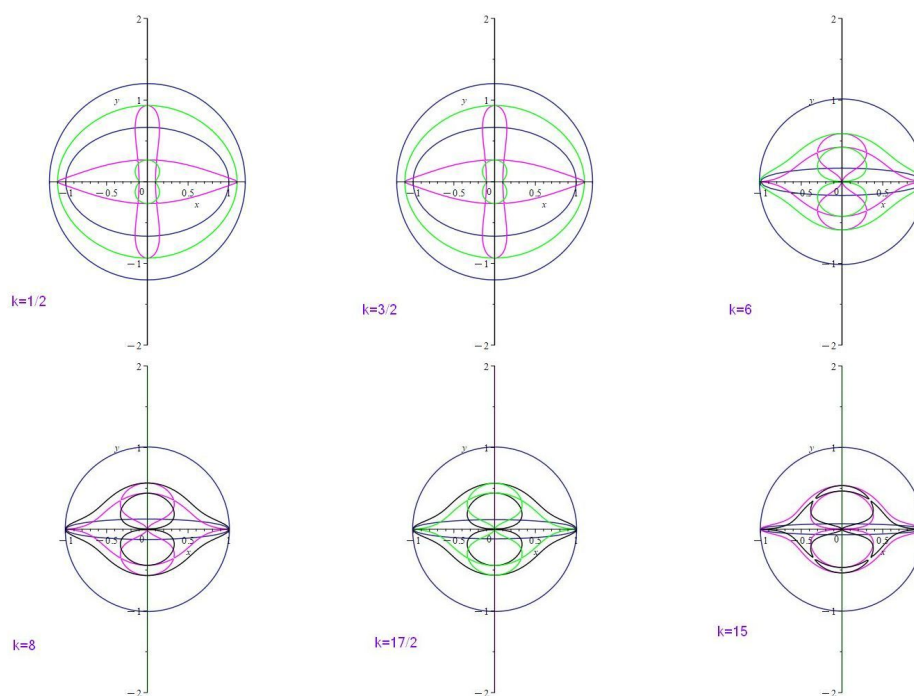


Figure 5. Several plots from the Maple exploration after specialization-elimination.

3. A construction based on implicit equations only

3.1. A particular case

Aiming to obtain the general implicit equation of this family of curves, in this Section we will approach this goal starting with an alternate description of the construction, using implicit equations. In order to make things easier for the non-familiar reader, we begin with the special case $k = 1$, i.e. we consider the circles C and \mathcal{D} whose respective equations are : $x^2 + y^2 - 1 = 0$ and $x^2 + y^2 - 2 = 0$. A point $E(e_1, e_2)$ on the circle C verifies the condition $e_1^2 + e_2^2 - 1 = 0$, the line OE (where O denotes the origin) has equation $e_2 x - e_1 y = 0$. We denote by $P(x, y)$ the point of intersection of the line OE with the circle \mathcal{D} . The midpoint $F(f, g)$ of EP is determined by

$$\begin{cases} f = \frac{1}{2}(x + e_1) \\ g = \frac{1}{2}(y + e_2) \end{cases} \quad (3.1)$$

Actually, Equation (3.1) represent the two midpoints from E to the intersections F, G of the line OE with the circle \mathcal{D} . Define an ideal

$$H = \langle 2f - x - e_1, 2g - y - e_2, -e_1y + e_2x, e_1^2 + e_2^2 - 1, x^2 + y^2 - 2 \rangle.$$

Now we apply commands from Maple's *PolynomialIdeals* package. We obtain $HilbertDimension(H) = 1$ and $EliminationIdeal(H, e_2) = \langle 0 \rangle$. So the construction is of dimension 1, ruled by one coordinate of E (say, e_2) on the circle C . Now the locus of $P(f, g)$ is a quartic (i.e. , a curve of degree 4): Applying once again elimination, we obtain:

$$EliminationIdeal(H, f, g) = \langle 16f^4 + 32f^2g^2 + 16g^4 - 24f^2 - 24g^2 + 1 \rangle.$$

Denote by $Q(x, y)$ the generator of this ideal. Using absolute factorization with the **AFactor** command, we have:

$$\begin{aligned} Q(x, y) &= 16f^4 + 32f^2g^2 + 16g^4 - 24f^2 - 24g^2 + 1 \\ &= (4f^2 + 4g^2 - 2 \operatorname{RootOf}(Z^2 - 2) - 3) (4f^2 + 4g^2 - 2 \operatorname{RootOf}(Z^2 - 2) - 3). \end{aligned} \quad (3.2)$$

The variety $V(Q)$ is the union of two circles, concentric with C and \mathcal{D} , with respective radii approximately 0.2071067816 and 1.207106781. The output is displayed in Figure 6 (we used the *algcurve* package and the **plot_real_curve** command).

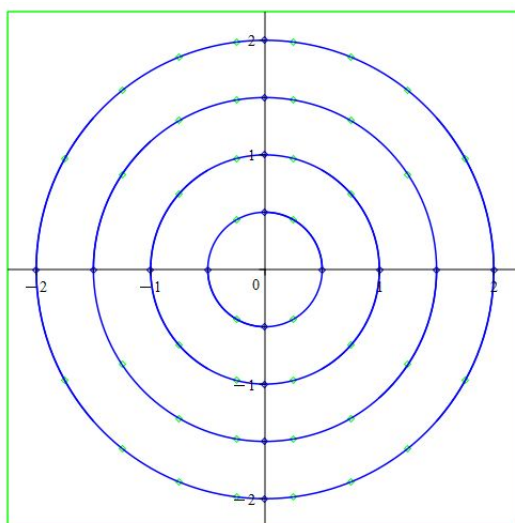


Figure 6. The loci of the midpoints for $k = 1$.

3.2. The general case

Now we deal with a family of loci, depending on parameter k . There are two ways. One, less satisfactory, is to convert all input into polynomials in all the variables, including k (thus, requiring the elimination of denominators). So the ideal now is of dimension 2 (say, taking e_1 and k as free variables) and the elimination of all the variables except the coordinates f_1, f_2 of the midpoint F , and k , provides a collection of f_1, f_2 -curves depending on k . Here is the Maple code of the session:

```
HH:=<e_1^2+k^2*e_2^2-1,k^2*x^2+k^2*y^2-k^2-1,x*e_2-y*e_1,f_1-(x+e_1)/2,f_2-(y+e_2)/2 >;
HilbertDimension(HH);
EliminationIdeal(HH, {e_1,k});
EHH:=EliminationIdeal(HH,{f_1,f_2,k});
```

The Hilbert dimension is equal to 2, and the 1st elimination ideal is equal to $\langle 0 \rangle$. Finally we obtain that the ideal EHH is generated by the polynomial

$$\begin{aligned}
 Loc_k = & 16f_1^4 f_2^4 k^8 + 32f_1^2 f_2^6 k^8 + 16f_2^8 k^8 + 32f_1^6 f_2^2 k^6 + 64f_1^4 f_2^4 k^6 + 32f_1^2 f_2^6 k^6 - 8f_1^2 f_2^4 k^8 - 8f_2^6 k^8 \\
 & + 16f_1^8 k^4 + 32f_1^6 f_2^2 k^4 + 16f_1^4 f_2^4 k^4 - 24f_1^4 f_2^2 k^6 - 40f_1^2 f_2^4 k^6 - 16f_2^6 k^6 + f_2^4 k^8 - 16f_1^6 k^4 \\
 & - 40f_1^4 f_2^2 k^4 - 24f_1^2 f_2^4 k^4 - 8f_1^6 k^2 - 8f_1^4 f_2^2 k^2 + 2f_1^2 f_2^2 k^4 + f_1^4. \quad (3.3)
 \end{aligned}$$

Let us compute and plot the curves for $k = 1/2, 3, 15$. We have:

$$\begin{aligned}
 Loc_{1/2} &= f_1^8 + \frac{5}{2}f_1^6f_2^2 + \frac{33}{16}f_1^4f_2^4 + \frac{5}{8}f_1^2f_2^6 + \frac{1}{16}f_2^8 - 3f_1^6 - \frac{39}{8}f_1^4f_2^2 - \frac{69}{32}f_1^2f_2^4 - \frac{9}{32}f_2^6 + f_1^4 \\
 &\quad + \frac{1}{8}f_1^2f_2^2 + \frac{1}{256}f_2^4 \\
 Loc_3 &= 1296f_1^8 + 25920f_1^6f_2^2 + 152928f_1^4f_2^4 + 233280f_1^2f_2^6 + 104976f_2^8 - 1368f_1^6 - 20808f_1^4f_2^2 \\
 &\quad - 83592f_1^2f_2^4 - 64152f_2^6 + f_1^4 + 162f_1^2f_2^2 + 6561f_2^4 \\
 Loc_{15} &= 810000f_1^8 + 366120000f_1^6f_2^2 + 41736060000f_1^4f_2^4 + 82377000000f_1^2f_2^6 \\
 &\quad + 41006250000f_2^8 - 811800f_1^6 - 275401800f_1^4f_2^2 - 20959965000f_1^2f_2^4 \\
 &\quad - 20685375000f_2^6 + f_1^4 + 101250f_1^2f_2^2 + 2562890625f_2^4
 \end{aligned} \tag{3.4}$$

In all the cases – including the generic one – the obtained polynomial is absolutely irreducible (use the **AFactor** command), saying that the two components of the obtained curve cannot be distinguished by algebraic means. The resulting plots are displayed in Figure 7. It is interesting to compare with the plots in Figure 3, supporting the idea that in both cases these are, indeed, the true “octics”.

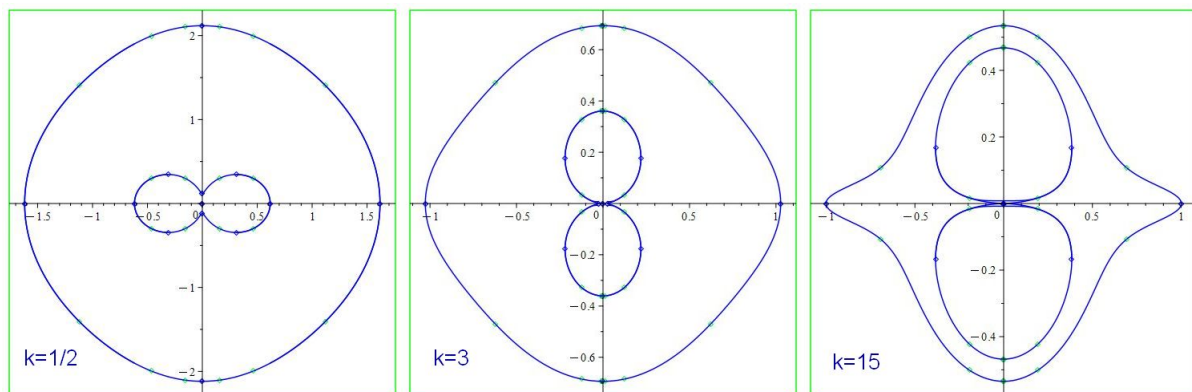


Figure 7. Three examples of the locus of midpoints.

This way is less satisfactory since it means that one could be losing values of k such as $k = 0$ (but in this case this has no sense). The other, more canonical way, is to consider the ideal GH , with polynomials in all variables, but with k in the denominator, and to consider the field of coefficients $\mathbb{Q}(k)$. It is like specializing the polynomial ideal to a – say – “numerical” value k . Thus, we have to bear in mind what we have previously commented about the relation between specialization and elimination. The ideal has now dimension 1, with e_1 as only free variable, for example. The Maple code for the session is as follows:

```

GH:=<e_1^2+k^2*e_2^2-1,x^2+y^2-1-1/k^2,x*e_2-y*e_1,f_1-(x+e_1)/2,
      f_2-(y+e_2)/2, variables={e_1,e_2,x,y,f_1,f_2}>;
HilbertDimension(GH);
EliminationIdeal(GH, {e_1});

```

```

EliminationIdeal(GH, {f_1, f_2});
GL:=Generators(GH)[1];
simplify(GL-Loc_k);

```

Here the Hilbert dimension is 1 and the 1st elimination ideal is $\langle 0 \rangle$. The polynomial GL is given by:

$$\begin{aligned}
GL = & 16f_1^4 f_2^4 k^8 + 32f_1^2 f_2^6 k^8 + 16f_2^8 k^8 + 32f_1^6 f_2^2 k^6 + 64f_1^4 f_2^4 k^6 + 32f_1^2 f_2^6 k^6 - 8f_1^2 f_2^4 k^8 \\
& - 8f_2^6 k^8 + 16f_1^8 k^4 + 32f_1^6 f_2^2 k^4 + 16f_1^4 f_2^4 k^4 - 24f_1^4 f_2^2 k^6 - 40f_1^2 f_2^4 k^6 - 16f_2^6 k^6 \\
& + f_2^4 k^8 - 16f_1^6 k^4 - 40f_1^4 f_2^2 k^4 - 24f_1^2 f_2^4 k^4 - 8f_1^6 k^2 - 8f_1^4 f_2^2 k^2 + 2f_1^2 f_2^2 k^4 + f_1^4.
\end{aligned} \tag{3.5}$$

Finally, we have verified that, in this case, the locus result GL is equal to Loc_k , confirming the previous results!

4. Discussion and conclusions

As we have described along the paper, this apparently simple geometric locus gives rise, through different approaches: Implicit, parametric, specialized, etc., to a series of challenging mathematical issues, that require the use of CAS tools for their analysis and interpretation. Thus, we have shown (and explained) in subsection 2.1 the surprising behavior of the parametric-tracer points for the geometric locus under consideration, splitting the obtained curves in two halves, divided by the y -axis. Then, subsection 2.2, presented the failure to implicitize the generic equation of the locus family, although it succeeded in different specialized cases. It also described some involved issues concerning the presence of three different factors of such equation (while only one of them should be considered as the true “octic”), and the need of further study to identify and clarify this fact.

Finally, Section 3 dealt with a pure algebraic geometry approach, but here also considering two different possibilities (taking the parameter as a variable or as an element of the field of coefficients). We have shown that both approaches do well in this particular case, in obtaining the general implicit equation of the family of curves, showing also its absolute irreducibility and allowing the visual and symbolic verification of the coincidence of the outputs in both the implicit and parametric approaches to the octics.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

The first author has been partially supported by the Roland and Astrid Chair (CEMJ) at JCT. Second author partially supported by a grant PID2020-113192GB-I00 (Mathematical Visualization: Foundations, Algorithms and Applications) from the Spanish MICINN.

The authors wish to thank Zoltan Kovács for help with GeoGebra Discovery.

Conflict of interest

The authors report that for this work, there is no conflict of interest.

References

1. M. Abanades, F. Botana, A. Montes, T. Recio, An algebraic taxonomy for locus computation in dynamic geometry, *Comput. Aided Design*, **56** (2014), 22–33. <https://doi.org/10.1016/j.cad.2014.06.008>
2. J. G. Alcazar, On the shape of rational algebraic curves depending on one parameter, *Comput. Aided Geom. D.*, **27** (2009), 162–178. <https://doi.org/10.1016/j.cagd.2009.11.004>
3. J. G. Alcazar, J. Schicho, J. R. Sendra, A delineability-based method for Computing Critical Sets of Algebraic Surfaces, *J. Symb. Comput.*, **42** (2007), 678–691. <https://doi.org/10.1016/j.jsc.2007.02.001>
4. D. Cox, J. Little, D. O’Shea, *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*, Undergraduate Texts in Mathematics, NY: Springer, (1992).
5. Th. Dana-Picard, Automated study of a regular trifolium, *Math. Comput. Sci.*, **13** (2018), 57–67.
6. Th. Dana-Picard, Safety zone in an entertainment park: Envelopes, offsets and a new construction of a Maltese Cross, *Electronic Proceedings of the Asian Conference on Technology in Mathematics ACTM 2020; Mathematics and Technology* (2020), ISSN 1940–4204 (online version).
7. Th. Dana-Picard, Exploration of envelopes of parameterized families of surfaces in a technology-rich environment, *Electron. J. Math. Technol.*, (2023), In press.
8. Th. Dana-Picard, S. Hershkovitz, From Space to Maths And to Arts: Virtual Art in Space with Planetary Orbits, *Electron. J. Math. Technol.*, (2023), In press.
9. Th. Dana-Picard, Z. Kovács (2021), Networking of technologies: a dialog between CAS and DGS, *Electronic J. Math. Technol.*, **15** (2021), 43–59. <https://doi.org/10.1515/econ-2021-0004>
10. Th. Dana-Picard, Z. Kovács, Dynamic and automated constructions of plane curves, *Maple Transactions*, (2023), In press.
11. Th. Dana-Picard, A. Naiman, W. Mozgawa, V. Ciésłak, Exploring the isoptics of Fermat curves in the affine plane using DGS and CAS, *Math. Comput. Sci.*, **14** (2020), 45–67.
12. K. Jin, J. Cheng, On the Complexity of Computing the Topology of Real Algebraic Space Curve, (2023) (preprint). Available from: https://www.researchgate.net/publication/330726079_On_the_Complexity_of_Computing_the_Topology_of_Real_Algebraic_Space_Curves
13. Z. Kovács, Th. Dana-Picard, Inner isoptics of a parabola, Conference: 5th Croatian Conference on Geometry and Graphics, Dubrovnik, September 4-8, 2022. Available from: https://www.researchgate.net/publication/363263393_Inner_isoptics_of_a_parabola
14. Z. Kovács, B. Parisse, *Giac and GeoGebra-improved Gröbner basis computations*, In Gutierrez, J., Schicho, J., Weimann, M. (eds.), *Computer Algebra and Polynomials*, Lecture Notes in Computer Science 8942, (2015), 126–138. Springer. https://doi.org/10.1007/978-3-319-15081-9_7
15. E. Roanes-Lozano, E. Roanes-Macías, M. Villar-Mena, A bridge between dynamic geometry and computer algebra, *Math. Comput. Model.*, **37** (2003), 1005–1028. [https://doi.org/10.1016/S0895-7177\(03\)00115-8](https://doi.org/10.1016/S0895-7177(03)00115-8)

16. E. Roanes-Lozano, N. van Labeke, E. Roanes-Macías, Connecting the 3D DGS Calques3D with the CAS Maple, *Math. Comput. Simulat.*, **80** (2010), 1153–1176. <https://doi.org/10.1016/j.matcom.2009.09.008>
17. M. Selaković, V. Marinković, P. Janičić, New dynamics in dynamic geometry: Dragging constructed point, *J. Symb. Comput.*, **97** (2020), 3–15.
18. J. R. Sendra, F. Winkler, S. Pérez-Díaz, *Rational Algebraic Curves: A Computer Algebra Approach*, NY: Springer, (2008).
19. D. Zeitoun, Th. Dana-Picard, Accurate visualization of graphs of functions of two real variables, *Int. J. Comput. Math. Sci.*, **4** (2010), 1–11.



AIMS Press

©2023 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)