

Automatic Determination of Algebraic Surfaces as Loci of Points

Francisco Botana

Departamento de Matemática Aplicada

Universidad de Vigo

Campus A Xunqueira, 36005 Pontevedra, Spain

fbotana@uvigo.es

Abstract. 3D–Loci Discovery (3D–LD), a web–based system for automatic discovery of geometric loci in the space, is described. Its basic engine is based on the Groebner basis method. 3D–LD can be used to easily (re)discover many loci–related geometric surfaces, and it has allowed us to generalize a recent extension of Simson–Steiner’s theorems due to Guzmán.

Being accessible through the web, 3D–LD is platform independent, and can be linked with other graphical tools. It uses the symbolic capabilities of Mathematica and CoCoA, by means of the recently distributed application webMathematica, a Java servlet technology allowing the generation of dynamic web content. A simple description of 3D–LD and some examples of use are available at

<http://rosalia.uvigo.es/sdge/web/3D/>.

1 Introduction

Two algebraic methods developed during the 80s have been successfully used to prove geometric theorems. Wu [13,7] and Groebner basis [5,1] methods have led to the development of powerful geometric theorem provers over the last 15 years. Recio and Vélez [10] have extended the applications of the Groebner basis method to deal with geometric discovery in the plane. They suggested to link their techniques with a dynamic geometry system in order to perform discovery in a fully automated way, a proposal also given by Roanes and Roanes [11]. Such a system has been reported in [2]. Using the same approach, a program specialized in the obtaining of plane loci, Lugares, is described in [4], and a comparison with other approaches is reported in [3]. Since its inception, Lugares has claimed for an extension to 3D. Nevertheless, the complexity of designing and implementing a dynamic 3D geometry environment signified a drawback. The recent distribution of webMathematica, a Java servlet technology allowing the remote access to the symbolic capabilities of Mathematica, has made possible the implementation of 3D–LD, a web–based resource for automatic discovery of 3D geometric loci and other derived surfaces such as pedals. 3D–LD uses the Groebner basis method in order to discover the equations of surfaces defined

through Euclidean operations. Currently, 3D-LD allows us to describe the constructions by means of an extensive set of predicates. These are the inputs of a Mathematica Server Page, which returns the implicit equation of the surface and plots it using the LiveGraphics3D or the Javaview applets.

2 The User Interface and the Architecture of 3D-LD

Using 3D-LD is straightforward. On the client's side, the user must specify a set of geometric constraints defining a surface by means of a generic point. Once uploaded this construction to the server, the implicit equation and an interactive plot of the surface are returned.

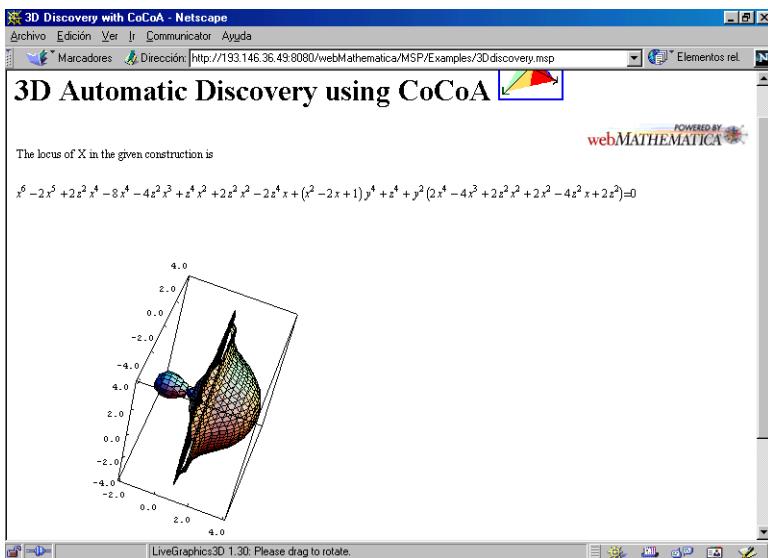


Fig. 1. A generalized conchoid of Sluze

Let us illustrate it with a single case. Given a point P on the plane defined by $A(1, 0, 0)$, $B(0, 1, 0)$ and $C(0, 0, 1)$, which is the locus of the points X such that the origin O , P and X are aligned, and $\overline{OP} \cdot \overline{XP} = 3$? These constraints can be specified as follows (see the URL above for a complete description of the grammar):

```
FreePointD[0,0,0,0];
FreePointD[A,1,0,0];
FreePointD[B,0,1,0];
FreePointD[C,0,0,1];
PlaneD[ABC,A,B,C];
PointOnPlane[P,ABC];
LineD[OP,O,P];
```

```

LocusPointD[X];
PointOnLineC[X,OP];
Distance[0,P]*Distance[X,P]==3;
Range[-7,4,-4,4,-4,4];

```

In a few seconds the server returns the implicit equation and an interactive plot of the algebraic surface found (Fig. 1).

The 3D-LD architecture is sketched in Fig. 2 and roughly described as follows. The textual specification of the geometric construction is uploaded to a web server via a CGI-POST request. Once uploaded the file, the server forwards a request to a Java servlet container (Tomcat) where webMathematica is installed. A Mathematica Server Page carries out an algorithm for discovering the equation of the locus point. This algorithm is adapted from a 2D algorithm fully described in [4]. Its main tasks consist of translating the geometric constraints into polynomials, and eliminating, via the Groebner basis method, the dependent variables (in a geometric sense) from the polynomial system.

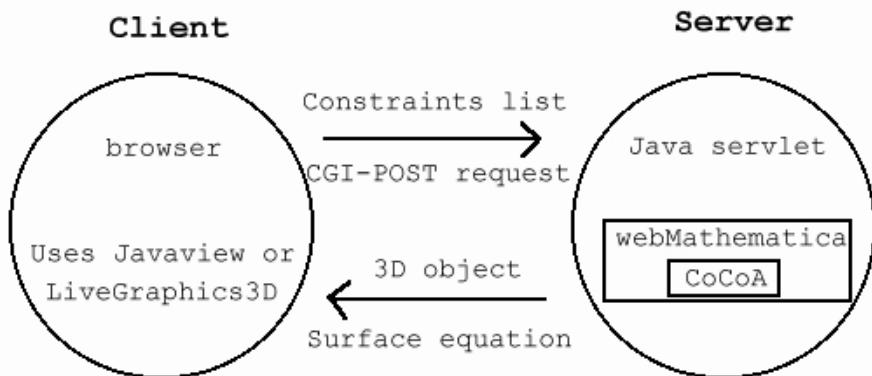


Fig. 2. The architecture of 3D-LD

In the example above, four points and the plane $x + 1 = 0$ are firstly defined. Point $P(x_1, y_1, z_1)$ is bounded to this plane through the polynomial $x_1 + 1$. P is also used to set a line where the locus point $X(x, y, z)$ lies in, giving the polynomials $xy_1 - x_1y$, $xz_1 - x_1z$ and $yz_1 - y_1z$. The squared condition involving distances introduces the last polynomial $((x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2)(x_1^2 + y_1^2 + z_1^2) - 9$. So, the construction is described by the system

$$\begin{aligned}
p_1 &= x_1 + 1, \\
p_2 &= xy_1 - x_1y, \\
p_3 &= xz_1 - x_1z, \\
p_4 &= yz_1 - y_1z,
\end{aligned}$$

$$p_5 = ((x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2)(x_1^2 + y_1^2 + z_1^2) - 9.$$

where $p_1, \dots, p_5 \in Q[x_1, y_1, z_1, x, y, z]$. Thus, the affine variety defined by $V = \{p_1 = 0, \dots, p_5 = 0\} \subset C^6$ contains all points $(x_1, y_1, z_1, x, y, z) \in C^6$ which satisfy the construction requirements, that is, the set of all common zeros of p_1, \dots, p_5 in the 6-dimensional affine space of C describe all the possible positions of the construction points. In particular, the positions of the locus point define the locus we are searching for. Thus, the projection

$$\pi : (x_1, y_1, z_1, x, y, z) \in V \subset C^6 \rightarrow (x, y, z) \in C^3$$

gives an extensional definition of the locus in the affine space C^3 . This projection can be computed via the third elimination ideal of $\langle p_1, \dots, p_5 \rangle$, I_3 . The Closure theorem states that $V(I_3)$ is the smallest affine variety containing $\pi(V)$, and the Elimination theorem states that, being G a Groebner basis of $\langle p_1, \dots, p_5 \rangle$, $G_3 = G \cap Q[x, y, z]$ is a Groebner basis of I_3 . In the case we are dealing with, the polynomial in Fig. 1 is the generator of the ideal. Finally, a 3D graphic object is sent to the web browser, viewable via the LiveGraphics3D or the Javaview applets.

It must be noted that 3D–LD does not square any condition involving distances, thus yielding sometimes a non polynomial system. The elimination process is performed by the Mathematica kernel, in this case. Nevertheless, the inefficiency of the Mathematica `Eliminate` command can be avoided by the user manually replacing the distances constraint by its squared version. Whenever the constraint list can be translated into a polynomial system, CoCoA [6] is launched from inside webMathematica for performing the elimination process.

3 Linking 3D–LD with a 3D Dynamic Geometry Environment

An important application of 3D–LD is the integration as a viewer and as a symbolic processing engine to other packages. 3D–LD has been linked with Calques3D [12], a dynamic geometry software for 3D explorations. This environment provides the user with tools for defining and visualizing 3D constructions. Following the dynamic geometry paradigm [8], these constructions generally behave according the user-defined constraints when a basic element is dragged.

Calques3D can export its constructions in a form suitable for 3D–LD input. Although it is planned using the hybrid system for performing general geometric discovery in the space, currently the link just works for discovery of algebraic surfaces as loci of points. We consider the Bohemian dome in order to illustrate the drawbacks of purely interactive approaches for loci discovery (see [3] for a discussion of this issue in the plane) and the integration of 3D–LD and Calques3D.

A Bohemian dome can be constructed as follows. Given a circle C and plane E perpendicular to the plane of C , move a second circle K through space so

that its center always lies on C and it remains parallel to E . Then K sweeps out the Bohemian dome. Fig. 3 shows the wrong answer of Calques3D and the result returned by 3D-LD.

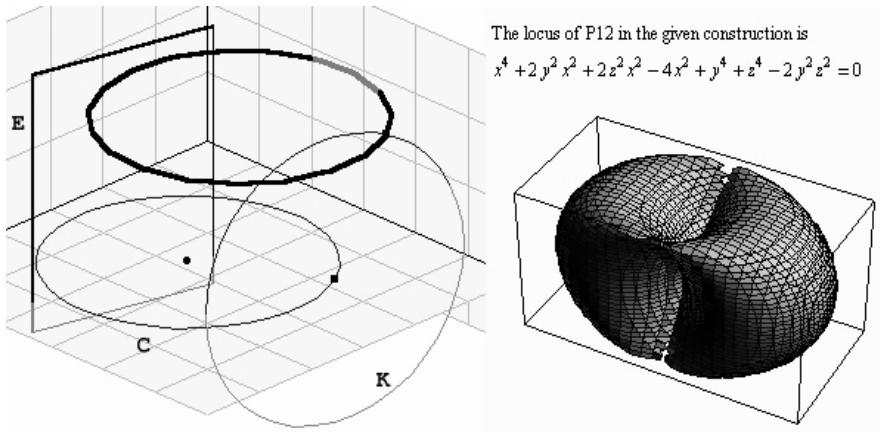


Fig. 3. A Bohemian dome

4 A Cubic Surface Generalizing Simson–Steiner–Guzmán Theorems

If one draws from a point X in the Euclidean plane the perpendiculars to the sides AB , AC , BC of a triangle ABC and if the feet of the perpendiculars $P \in AB$, $Q \in AC$, $R \in BC$ lie on a line (the Wallace line of X for the triangle ABC) then X lies on the circumcircle of the triangle ABC . This result, known as the Simson–Wallace theorem, was generalized by Steiner replacing the alignment of P, Q, R by the invariance of the oriented area of the triangle PQR and by Guzmán [9] projecting X in arbitrary directions.

In turn, Roanes–Macías and Roanes–Lozano [11] extended to 3D Simson–Steiner theorems using Wu’s method. The result of Guzmán is also generalizable to space as follows.

Given a tetrahedron $OABC$, an arbitrary point X , four projection directions RS, RT, RU, RV , not all four equal, nor parallel to the planes spanned by the faces of the tetrahedron, and the projections of X along these directions on the faces, M, N, P, Q , the locus of X such that the volume of the tetrahedron $MNPQ$ is a constant is a cubic.

The theorem remains open since it can not be solved for an arbitrary tetrahedron due to the doubly exponential cost of the Groebner basis method. Nevertheless, assigning simple numbers to the vertices of the tetrahedron, 3D-LD easily finds the cubic surface. Fig. 4 shows the surface when the vertices are the origin and unit points of a reference system.

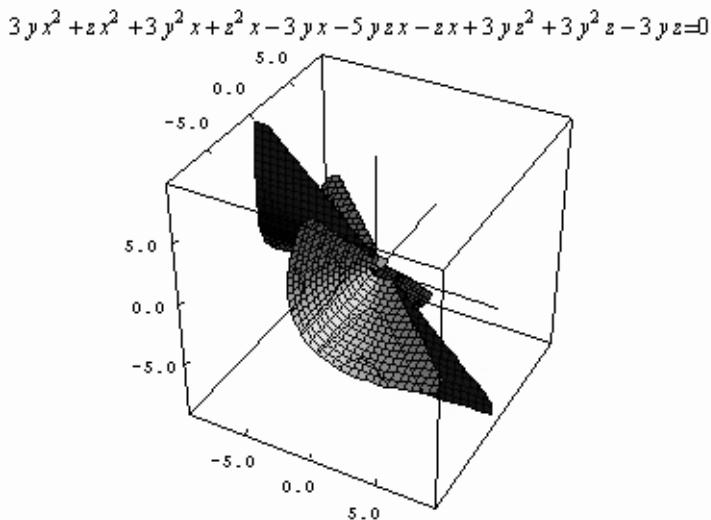
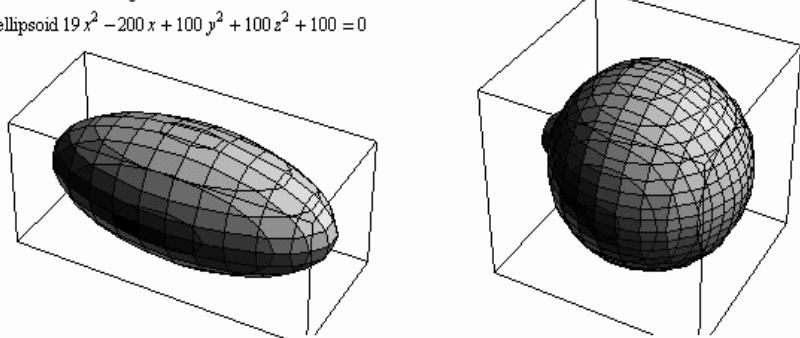


Fig. 4. An instance of a generalization of Guzmán's theorem

The locus of X in the given construction is

$$\text{the ellipsoid } 19x^2 - 200x + 100y^2 + 100z^2 + 100 = 0$$



The pedal surface of the obtained locus with respect to the point R(2,0,0) is

$$19x^4 - 276x^3 + 38y^2x^2 + 38z^2x^2 + 976x^2 - 276y^2x - 276z^2x - 1200x + 19y^4 + 19z^4 + 319y^2 + 38y^2z^2 + 319z^2 + 400 = 0$$

Fig. 5. An ellipsoid and its pedal surface wrt (2,0,0)

5 Other Derived Surfaces: Pedals and Envelopes

In addition to algebraic surfaces as loci of points, 3D-LD computes pedal surfaces and some single envelopes. Since the pedal surface of a given surface with respect to a point is the locus of the projection of a surface's point on the tangent plane, the system easily gets such new surface. Fig. 5 shows an ellipsoid as locus of a point and its pedal surface with respect to another point. Note that the grammar allows us to enter the implicit equation of a surface in order to obtain its pedal.

Nevertheless, the actual version of the software does not compute pedals of surfaces with singular points.

A canal surface is the envelope of a family of spheres with centers moving along a space curve. 3D-LD can compute these surfaces if the curve is a straight line. Fig. 6 shows a cone as the envelope of family of spheres centered at line joining the origin and $(0,1,0)$, and with radii half the distance between their centers and the origin.

The envelope of the given family of spheres is

$$\text{the cone } -3x^2 + y^2 - 3z^2 = 0$$

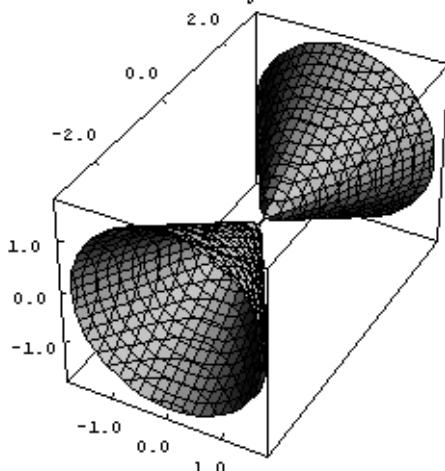


Fig. 6. A cone as the envelope of a family of spheres

6 Conclusion

A web-based system for automatic discovery of geometric loci in the space has been described. It uses the Groebner basis method as a bridge between the geometric formulation of a construction and its algebraic description. It has been developed using webMathematica, a Java servlet technology allowing dynamic generation via the web. Since some specialized computation are beyond the current capabilities of Mathematica, CoCoA is called from inside Mathematica to accomplish such tasks. 3D-LD can be used as a standalone system or linked with other tools, overcoming current limitations of 3D dynamic geometry environments.

References

1. Becker, T., Weispfenning, V.: *Gröbner Bases. A Computational Approach to Commutative Algebra*. Springer, Berlin (1998)
2. Botana, F., Valcarce, J.L.: A dynamic-symbolic interface for geometric theorem discovery. *Computers and Education*, 38(1–3), 21–35 (2002)
3. Botana, F.: Interactive versus symbolic approaches to plane loci generation in dynamic geometry environments. *Proc. I Int. Workshop on Computer Graphics and Geometric Modelling CGGM'2002, Lecture Notes in Computer Science*, 2330, 211–218 (2002)
4. Botana, F., Valcarce, J.L.: A software tool for the investigation of plane loci. *Mathematics and Computers in Simulation*, 61(2), 141–154 (2003)
5. Buchberger, B.: Gröbner Bases: an Algorithmic Method in Polynomial Ideal Theory. In N.K. Bose, *Multidimensional systems theory*, Reidel, Dordrecht, 184–232 (1985)
6. Capanni, A., Niesi, G., Robbiano: CoCoA, a system for doing Computations in Commutative Algebra. Available via anonymous ftp from: cocoa.dima.unige.it
7. Chou, S. C.: *Mechanical Geometry Theorem Proving*. Reidel, Dordrecht (1988)
8. Gao, X.S.: Automated geometry diagram construction and engineering geometry. *Proc. Automated Deduction in Geometry ADG'98, Lecture Notes in Artificial Intelligence*, 1669, 232–257 (1998)
9. Guzmán, M.: An extension of the Wallace–Simson theorem: projecting in arbitrary directions. *American Mathematical Monthly*, 106(6), 574–580 (1999)
10. Recio, T., Vélez, M. P.: Automatic discovery of theorems in elementary geometry. *Journal of Automated Reasoning*, 23, 63–82 (1999)
11. Roanes–Macías, E., Roanes–Lozano, E.: Automatic determination of geometric loci. 3D-extension of Simson–Steiner theorem. *Proc. Artificial Intelligence and Symbolic Computing AISC 2000, Lecture Notes in Artificial Intelligence*, 1930, 157–173 (2000)
12. Van Labeke, N.: Calques 3D: a microworld for spatial geometry learning. *Proc. Intelligent Tutoring Systems ITS'98. System Demonstrations* (1998)
13. Wu, W. T.: *Mechanical Theorem Proving in Geometries*. Springer, Wien (1994)