

Human-computer interfaces applied to numerical solution of the Plateau problem

Antonio Elias Fabris¹, Ivana Soares Bandeira¹, Valério Ramos Batista²

¹IME-USP, r. do Matão 1010, 05508-090 São Paulo-SP, Brazil

²CMCC-UFABC, av. dos Estados 5001, 09210-580 St André-SP, Brazil

E-mail: aefabris@gmail.com

Abstract. In this work we present a code in Matlab to solve the Problem of Plateau numerically, and the code will include human-computer interface. The Problem of Plateau has applications in areas of knowledge like, for instance, Computer Graphics. The solution method will be the same one of the Surface Evolver, but the difference will be a complete graphical interface with the user. This will enable us to implement other kinds of interface like ocular mouse, voice, touch, etc. To date, Evolver does not include any graphical interface, which restricts its use by the scientific community. Specially, its use is practically impossible for most of the Physically Challenged People.

1. Introduction

In 1989 K. Brakke introduced the first version of a general-purpose program to perform physical experiments in a virtual way. His program is devoted to studying the behaviour of surfaces when submitted to surface tension and other energies. For this reason it is named *Surface Evolver*, and its latest version is 2.70 [1]. Even some earlier versions were already complete and broad enough for several applications in many Areas of Knowledge like Physics, Chemistry, Mathematics and Engineering [2–5]. Most recently we have [6, 7] and [8] in Medicine.

The source-code of *Evolver* is open and freely available. It is all written in standard C-language, and consists of circa 100 files with a total of nearly 180,000 lines of source-code. A curious fact is that such a complex and complete program results from the dedication of a work team that consists of a sole member, namely Prof Kenneth Brakke. Studying the Evolver code is a challenging task, not to mention the maintenance of this program in a distant future.

One of the reasons for which Evolver's development was so rapid is that K. Brakke gave priority to the *technical* part of his program. Consequently, in Evolver data input is only possible by means of datafiles and command lines. Namely, there is no Graphical User Interface (GUI) like clicking a mouse arrow on a virtual button. This seems to be just a minor detail but nowadays academia has been using human-computer interfaces increasingly. Voice recognition, touchscreen and head mouses are resources that greatly simplify the daily work of researchers, professors and students.

In this sense our present work is also devoted to *Physically Challenged People* (PCP) that cannot use the hands when working on a computer. Their participation in academia has been growing steadily and their access to programs like Evolver is highly desirable.



Already in the version 1.0 Evolver included the *Numerical Solution of the Plateau Problem* (NSPP). It consists of finding the least area surface that spans a given 3D curve in space. The solution is called *minimal surface*, which also has the least overall surface tension among all spanners of the given curve. Physically they are soap films obtained by dipping a closed wire curve into soap solution.

Our approach is to implement Evolver's technique to NSPP in one of the easiest programming languages, namely Matlab [9]. This language can be used through voice recognition and touch pen. Moreover, it is highly compact and we estimate that it can reduce the C-code of Evolver's NSPP by almost 70%.

This is because our Matlab version will *not* be a mere translation between programming languages. Rather, our NSPP has been implemented directly from the Technical References of Evolver Manual (Chapter 16), and already in accordance with Matlab's short-programming philosophy: toolboxes, syntactical and semantical conciseness, GUI, etc. This task is not trivial because it requires a perfect understanding of Evolver's NSPP technique, which includes deep knowledge of Discrete Differential Geometry and also Optimization Algorithms.

We are still at the first stage of a year long research. Our Matlab code will be available in future and it will enable researchers to include other capabilities of Evolver. Additionally, our GUI will be implemented to work with the *Eye Tribe Tracker*, which is one of the most comfortable eye tracking devices to date. See Section 2 for details. In this way we hope to contribute to PCP like many other researchers' works in the literature [10–13].

2. Methodology

The *Eye Tribe* is a Danish enterprise devoted to eye tracking technology. One of its products is a case box with photosensitive cells. The case box can be attached to the bottom of a screen, as depicted in Figure 1. The photosensitive cells detect eye movements, which can be used as a mouse.

Evolver requires the description of initial surfaces within a datafile. These are very rough polyhedral surfaces that are later refined and smoothened by Evolver commands. In order to describe an initial surface one has to write a list of its vertices and edges within the datafile. Our Matlab version does it graphically with a mouse (see Figure 2).



Figure 1: The Eye Tribe Tracker.¹

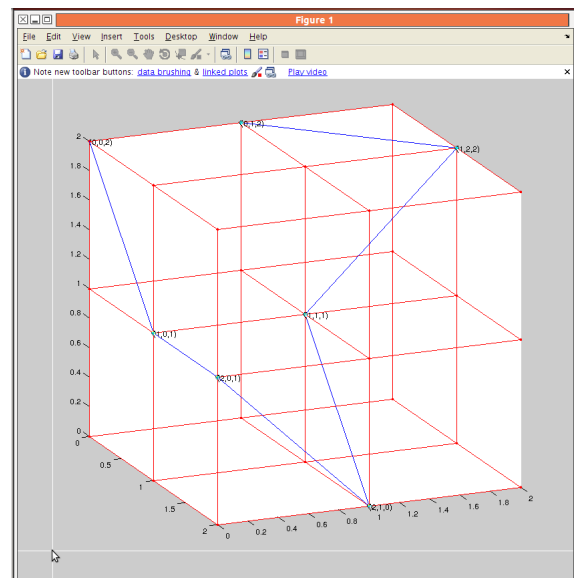


Figure 2: Plotting a 3D polygonal curve.

¹ Taken from <https://theeyetribe.com/products>

Regarding the NSPP, we invoke Evolver built-in commands that refine the initial triangulation, iterate the surface towards minimum energy, make triangles uniform, add or remove them either locally or globally, save current surface, etc. These commands can be repeated and invoked in any order. For the time being we have implemented just two of them: surface refinement and iteration. See Figures 3-5.

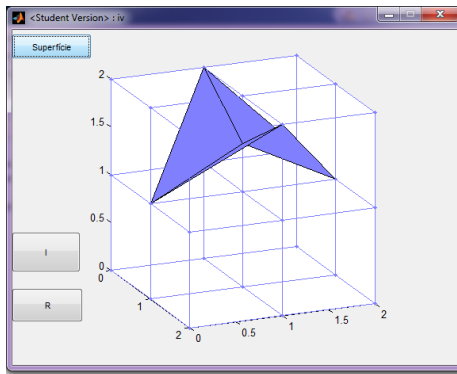


Figure 3: Buttons I and R.

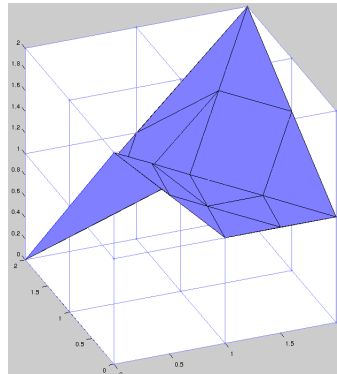


Figure 4: Refining.

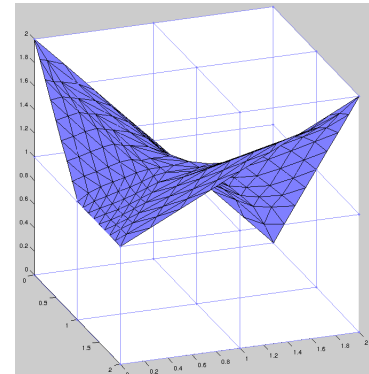


Figure 5: Minimal surface.

Our commands are activated by mouse click on virtual buttons, as shown in Figure 3. In future we shall implement the Eye Tribe Tracker in Matlab to work as a mouse in our program.

As mentioned in the Introduction, our code is being programmed according to the Technical References of the Evolver Manual. Therefore, details about how to manipulate the triangulated surface are omitted here. However, the 2nd author is preparing a PhD-Thesis that will present the theory behind NSPP, together with pseudo-codes, block diagrams and illustrations that explain how the whole program works. In this way we strive for accessibility in a much broader sense, for the code will be comprehensible even to undergraduate students.

3. Results

The main result is our program, devoted both to users and developers. Its source code will be freely available in future, together with a complete technical documentation.

4. Conclusions

We have just presented the achievements of a year long research. Admittedly, both the involved theory and programming strategies are highly non-trivial. However, we are focusing only on the NSPP. Because of that, a short code that is also easy to handle and to understand is a totally feasible task, even considering that it will include the programming of human-computer interfaces. In our case it is the Eye Tribe Tracker as a mouse to control some of Evolver's capabilities.

5. Acknowledgements

This research is supported by FAPESP proc.No.14/20219-0.

References

- [1] Brakke K 2013 *Susquehanna University* URL <http://www.susqu.edu/brakke/evolver/evolver.html>
- [2] Tegtart J 1991 *AIAA, SAE, ASME, and ASEE, 27th Joint Propulsion Conference* vol 1
- [3] Michalet X and Bensimon D 1995 *Science* **269** 666–668
- [4] Francis G, Sullivan J M, Kusner R B, Brakke K A, Hartman C and Chappell G 1997 *Visualization and Mathematics* (Springer) pp 3–20

- [5] Racz L, Szekely J and Brakke K 1993 *ISIJ international* **33** 328–335
- [6] Nascimento M Z and Ramos Batista V 2014 *Journal of Physics: Conference Series* vol 490 (IOP Publishing) p 012146
- [7] Nascimento M Z and Ramos Batista V 2015 *Journal of Physics: Conference Series* vol 574 (IOP Publishing) p 012072
- [8] Nascimento M Z and Ramos Batista V 2015 *Recent Patents on Medical Imaging* **to appear**
- [9] MatLab M 2012 *The MathWorks, Inc.* URL <http://www.mathworks.com>
- [10] Hornof A, Cavender A and Hoselton R 2004 *ASSETS '04 ACM SIGACCESS Conference on Computers and Accessibility* (ACM Press) pp 86–93
- [11] Perini E, Soria S, Prati A and Cucchiara R 2006 *Proceedings of the 2006 International Conference on Computer Vision in Human-Computer Interaction ECCV'06* (Berlin, Heidelberg: Springer-Verlag) pp 99–108
- [12] Chau T, Memarian N, Leung B, Treherne D, Hobbs D, Worthington-Eyre B, Lamont A and Pla-Mobarak M 2012 *Handbook of Ambient Assisted Living* pp 581–597
- [13] Jian Y and Jin J 2012 *Proceedings of the 2012 International Conference on Systems and Informatics (ICSAI)* (IEEE) pp 505–509