Vera Sacristán Adinolfi

Computational Geometry-GEI

Bachelor Degree in Informatics Engineering (GEI)

Departament de Matemàrica Aplicada II
Facultat d'Informàtica de Barcelona
Universitat Politècnica de Catalunya

Contents

Goal of the course

The Computational Geometry course has an algorithmic and applied flavor. Its contents are oriented to dealing with massive geometric data, and the lab exercises are intended to make students familiar with real problems coming from computer graphics, geographic information systems, robotics, land planning, etc., and they are solved with C++ and OpenGL. The goal of the course is to present the most frequent geometric algorithms that one can find behind all sorts of software having a geometric component. We want the student to learn how to optimally construct optimal solutions to this kind of problems by combining a) the geometric properties of the problem, b) an efficient algorithmic paradigm, and c) the most convenient data structure.

What is Computational Geometry?

Just for starting, here are some links that can help you getting an idea of what Computational Geometry is.

For the lazy, a short and clear article explaining what Computational Geometry is, and what it applies to.

For those who want to see it in action, [many applets](#) made by the Computational Geometry students of the Facultad de Informática de Madrid.

For those who want to know it all, here are some web pages with more information: [Computational Geometry pages](#) (Jeff Erickson), [Geometry in Action](#) (David Eppstein), [Computational Geometry links](#) (Godfried Toussaint).

Material for the course

The general information related to a specific semester of the course (calendar, schedule, class rooms, evaluation, etc.) can be found at the [FIB web page](#). Announcements along the semester will be done at the [FIB corner (Racó)](#). In this web page (which is always under construction) I will offer you some material for the course.

In this space, as the course advances, we will post presentations and links related to the several subjects that we will work through in class.

1. [Presentation of the course](#)
2. Background
   - [Analyzing algorithms](#)
   - [Using the appropriate data structures](#)
   - [Describing and implementing basic geometric objects](#)
   - [Why can the median value of a set of real numbers be computed in (worse and/or expected) linear time?](#)
   - [How to compute the running time of a recursive algorithm](#)
3. Basic tool: [orientation tests](#)
4. A classical sweep-line algorithm: [Intersecting of line-segments: Bentley-Ottmann's algorithm](#)
5. Basic problems on polygons
   - [Stefan Schirra's reliability experiments for point-in-polygon localization strategies](#)
6. Convex hull of a set of points
   - [Convex hulls in 2D](#)
   - [Survey paper comparing the incremental algorithm to other convex hull algorithms in moderately high dimensions](#)
7. Duality. Intersection of half-planes
   - [Intersecting halfplanes and related problems](#)
   - Applet: [the parabola and the circle dualities](#)
8. [Triangulating polygons](#)
9. Proximity
   - [Proximity problems. Voronoi diagrams. Properties](#)
   - [Storing a Voronoi diagram](#)
   - [Computing Voronoi diagrams: algorithms](#)
   - [Some applications for Voronoi diagrams](#)
   - Applet: [the farthest point Voronoi diagram](#)
10. Triangulating point sets
    - [Triangulating point sets: Delaunay triangulation](#)
    - [Applying Delaunay tiangulations: Terrains](#)

- Applets showing the algorithms presented in class (and more):
    - http://www.cs.cornell.edu/Info/People/chew/Delaunay.html
    - http://www.pi6.fernuni-hagen.de/GeomLab/VoroGlide
    - http://www.dma.fi.upm.es/recursos/aplicaciones/geometria_computacional_y_grafos/
    - http://www.dma.fi.upm.es/mabellanas/tfcs/flips/Intercambios/Intercambios.htm
    - http://www.dma.fi.upm.es/mabellanas/tfcs/terrenos/proyecto.html
    - http://www.cs.unc.edu/~snoeyink/terrain/Demo.html
11. Arrangements of lines
12. Point location in planar decompositions
13. Extra topics
    - Range searching and Kd-trees: Slides by Marc van Kreveld
    - Robustness and CGAL: Slides by Rodrigo Silveira from the course Discrete and Algorithmic Geometry (DAG-MAMME)
14. Students' presentations

Problems

Each student must solve and present a number (to be discussed and agreed in class) of exercises from the problems list.

To know which problems are available and choose your problems, follow the link provided through the FIB corner (Racó).

Lab exercises

- Here is a pdf document containg the statements of the exercises, together with a presentation of the tool that can be used to solve them.
- Everything else needed for the lab work can be found at the FIB lab workspace.

Theory presentations

Please read the list of the proposed presentation subjects. To know which subjects have been already taken and choose your subject, follow the link provided through the FIB corner (Racó). You can also choose a subject different from the ones proposed. In this case, you must send me a proposal, so that I can evaluate its difficulties and its adequacy to the course, before accepting it.

Some recommendations about the presentations

From our experience, Spanish students have some expertise in carrying out lab exercises and preparing their corresponding documentation, but they have almost none in preparing and doing presentations. Let me give you some ideas that may be useful:

1. The first thing to do is to seriously study the subject, without thinking of its presentation, but just of understanding the subject.

2. Once this done, and before designing the presentation, it is convenient to answer the following questions:
   - Which are the main ideas behind the subject that I studied?
   - What can the audience understand, given their knowledge of the subject and the time that I have to make my presentation?
   - Which is the message that I want to transmit in my presentation?
   - How can I do it?

3. There is no unique presentation model. The same subject can be presented by giving a lot of technical detail of a given aspect, if this is what we want to underline because of its importance, as well as giving an overall vision by skimmimg the subject, if what matters is that the audience gets a global vision and the general flavor of the subject. One can make a very formal presentation (giving rigorous details of each step of an algorithm, for example), and a very intuitive one (by showing a demo of the same algorithm). It all depends on the goal of the presentation and the previous knowledge of the audience on the subject. In any case, it is convenient that you take into account some things:
   - Whatever your option is, it is important, for a good comprehension from your public, that in your presentation you make as explicit and clear as possible what the goal is of the subject that you will present.
   - Your slides, if any, must be clear and easy to read. Don't fill them with text!
   - Do not forget to also indicate which references you read.

4. When your presentation will be ready, I recommend that you rehearse it at least once. Rehearsing should help you to evaluate at least:
   - Whether you manage to make your talk to accurately express what you want to say.
   - Whether your presentation fits in the time that you have been assigned (for this purpose, I recommend that you rehearse out loud in front of a chronometer).

Following this link you will find more precise good advice on how to give a good talk: http://www.cs.dartmouth.edu/farid/downloads/tutorials/howtogiveagoodtalk.pdf

Resources

Mathematical text editor

Mathematical edition is almost always and everywhere done using LaTeX. Not only it is used in universities (LaTeX has been used to write all the documents of this course, as well as all the problems lists, exams, and other mathematical texts that you had in your hands during your studies at the FIB) but it is the most extended editor of scientific texts (LaTeX is used to write all the textbooks of scientific publishers as important as Springer, and also most of the mathematics and computer science conferences around the world).

It's a free source software that has versions for al operating systems (Linux, Mac, Windoxs, etc.), it helps writing all sort of scientific texts, such as articles, books and presentations, while it allows to incorporate figures previously produced in PDF by any drawing program.

You can download LaTeX from http://www.tug.org/.

References:

- Introductory text: L. Lamport, *LaTeX, a Document Preparation System*, Addison-Wesley, 2nd ed., 1994.
- The LaTeX project site.
- The LaTeX wikibook.

For those of you who work on windows, WinEdt is a very convenient editor for writing LaTeX code. You can download it from http://www.winedt.com/.

Drawings

Each person likes preparing his/her figures with his/her favorite drawing program. I use IPE (an evolution of xfig designed by a computational geometer), because it allows me to draw the geometric figures that I need, because it integrates text in LaTeX, and because it allows me to also prepare my presentations in a very easy what-you-see-is-what-you-get way.

It's a free source software that has versions for all the operating systems (Linux, Windoxs, Unix, etc.). You can download it from http://ipe.otfried.org/.

Geometric constructions

In addition to producing your own drawings, you may whish to create and experiment with geometric constructions. If so, we recommend GeoGebra.

Thesis, grants and projects

- Are you interested in doing a Masther Thesis or a Bachelor Thesis in Computational Geometry? Have a look!
- Are you interested in working in a Computational Geometry project? Have a look!
- Do you want to see what our research group does? Look here!

Vera Sacristán Adinolfi
Proudly powered by WordPress.