Computing and the Spatial Sciences

Christopher Gold
Department of Land Surveying and Geo-Informatics
Hong Kong Polytechnic University

Abstract

One of the puzzling issues in Geomatics, particularly in the North American context, is the continuing shortage of graduating GIS experts, despite a continuing strong market. This started twenty years ago, and has not yet been resolved.

This shortage occurs especially for jobs where both computing skills and knowledge of spatial concepts are required. Most spatially-oriented disciplines do not give an adequate grounding in Computer Science and Mathematics, and most Computing Science programs do not provide a background in spatial problems and algorithms. This leaves the graduates of both these programmes with an inadequate skill set for a significant portion of the job market. This is unfortunate - not only for the industry that needs these employees, but for the significant number of students in both disciplines who find this an interesting direction for study and employment. This is particularly noticeable given the current interest in game programming, which involves a similar emphasis on 'virtual reality'.

My experience suggests that some kind of joint program is required - the structure depending on the particular institution. Basic Computer Science courses in programming and data structures would be combined with introductory courses in one or more of the Spatial Sciences, leading to more advanced course work on spatial data structures, database design and the development of various applications of interest to the student. Examples of these could include map projections, terrain modelling, various simulations, and web-based projects. We believe that such a programme is both fascinating to the students and of significant value to potential employers.

Introduction

I recently had the opportunity to give a talk to a joint meeting of the Mathematics, Computing Science and Geography departments of Queens University, Canada. To me this symbolized something of a breakthrough, and perhaps a hope for the future – but also a warning.

The talk concerned what we might call “Spatial Science” – the conjunction of those who use space, or spatial concepts (not necessarily just geographic space), those who think about space in relatively abstract terms, and those who take the abstract concepts and develop computer algorithms and tools. It is salutary to see where the future of Spatial Science may be going, in the context of these three groups.

Some years ago (Gold, 1989) I wrote my opinions on the same subject, from the point of view of a geographer. The concern was “Breadth versus Depth”, and that concern remains today. In summary, I stated that there was a strong market for GIS trainees, and universities and technical colleges were not satisfying that demand. The same remains true today. I stated that there were not enough academics involved in the training of students, and in the development of new techniques. The same is true today. I stated that, as in the case of statistics, the misuse of GIS techniques would continue if there was not a sufficient emphasis on the understanding of the techniques used. This is even more obvious now – but there is some progress in the field of “metadata”, the documentation of the quality and source of a dataset. This is definitely progress – but even now there is little understanding, or documentation, of the effect of particular transformation algorithms on the quality of the secondary dataset. Unfortunately this problem becomes even more serious with the widespread distribution of government data, without any discussion of the original raw data and the algorithms used. We are left with the widespread acceptance that it is “truth”. As a journal reviewer I frequently see this, even from those who are producing academic publications. Of course, to some extent this is unavoidable, due to the prevalence of commercial “black boxes” at all levels of use.

What kind of programme is needed?

In 1989 I divided education issues into those from a non-technical background such as Geography, those with more of an “engineering” emphasis, who have a knowledge of an applications domain and who wish to learn the GIS tools, and those who, as academics, should be understanding and developing the techniques in use. I believe that this is still broadly true.

One approach is the “technical training” programme, with an emphasis on the use of existing tools for specific applications. This is most useful where the student has some previous experience in a spatial applications field (the “engineer”) – in this case the understanding of the spatial relationships involved in geology, forestry, urban planning, etc. The addition of GIS technical skills is here of great value, with readily-understood short-term gains. The technical college approach has been very successful here. In my opinion it has been less successful when taking students without any experience in a spatial discipline, as it becomes a learning of technique with less of an underlying understanding. However, there is no denying that the market is there, and these technicians find ready employment.

Those from the “soft” disciplines, such as geography, tend not to have much background in the mathematics and programming aspects of GIS. In some cases they pick these skills up outside their traditional background. In others they remain at the overview level, without much hands-on experience. In other cases there is a real emphasis on the Computer Science aspects, but this is relatively rare.

Thus the teaching of GIS is often limited to the training component, which introduces students to the tools of a particular black box, but without preceding this with any deep understanding of spatial problems. If they already have it from another source, then these programs can be very effective. Otherwise it becomes unthinking use of a proprietary product.

Where to put the Computer Science?

However, even if there is some sense of spatial understanding that produces a basic level of critical thinking about the problems to be solved, there is a major handicap when there is no knowledge of the computing tools being used. This is particularly embarrassing in an academic environment, where we are supposed to teach concepts rather than just methodology, and where we are suppose to encourage critical thinking. We must go beyond the black box – but that leads us into the world of algorithms, data structures and computer science.

In some ways it appears that GIS should be a part of Computer Science rather than Geography, Surveying, etc. However, in my experience with teaching mixed Geography and Computer Science classes, it became clear that programming skills alone did not produce an understanding of the complex interrelation of spatial relationships that must be taken into account by an experienced GIS practitioner.

Especially when we need to develop critical evaluation of spatial problems and methods, we must escape the black box syndrome. It helps somewhat if we are able to use, and compare, several software products, and use tools that have a wide range of functionality. (A good possibility here is the latest Manifold GIS, which integrates many vector and image processes. It is Windows-based, moderately priced and may be examined at www.manifold.net.) It helps a great deal more if we can examine and compare algorithms and data structures, with an emphasis on the pre-suppositions and simplifications used for each algorithm and class of problems – topology models and construction, terrain modelling and slope estimation, etc. In the absence of detailed information from the developers, we must turn to Computing Science.

The Computer Science emphasis is important for several reasons. Firstly we frequently do not realize how much difference there is between the results produced using different algorithms. Therefore we forget that all derived data should include algorithm information in the metadata description, and that this should be included with each data set as a quality control. Secondly, we must understand the basic problems of time and space efficiency of algorithms, and how decisions on these practical issues affect the possible output type and quality. Thirdly, Computer Science is developing an ever-increasing body of knowledge about spatial algorithms (especially in image processing and computational geometry), and these research results greatly affect the trade-offs just mentioned as well as the quality of the results. In addition, they are capable of providing formal proofs of efficiency and accuracy. Fourthly, if we in the Spatial Sciences do not take some control of our destiny here, rest assured that Computer Science will! The number of Computer Science contributions to GIS journals is increasing rapidly, and GIS problems are showing up in Computer Science journals.

However, as mentioned before, there is a considerable body of knowledge possessed by the practitioners of the various Spatial Sciences that is not readily available to Computer Science. In particular, we can produce a large variety of interesting problems, based on real-world issues, for those whose pleasure is algorithm development. Some time soon there must be a real effort to combine the skills of the two fields, to the benefit of both. We need to provide an environment to combine the skills of spatial understanding and computer science – but where do we find it?

Geography, despite its fundamental early contribution, often abrogates its responsibility, and retreats to generalizations, because there are too many pre-requisite studies before the link with spatial studies becomes apparent (mostly Mathematics and

Computer Science). There are, and have been, outstanding exceptions to this, but the average Geography department does not fulfill this role. This is a great misfortune, because they should be a natural focus for “Spatial Science”.

Other disciplines with a spatial component, and a more technical flavour, do fill the role to some extent. Forestry, Geology, Engineering and others develop skill sets within their fields, and are often involved in the development of special-purpose tools. The problem here is their usually narrow focus.

Surveying is another “natural” home for “Spatial Science”, and it often does a very good job, combining as it does a strong mathematics (and, hopefully, computing) skill set with a fundamental emphasis on “measuring the globe”. Its problems, other than insufficient emphasis on Computing Science in some cases, are due to being a relatively small discipline. While originally the emphasis may have been rather narrow – on geodesy perhaps, or the cadastre – this has now improved greatly, and now has the potential to do the job, if only it was big enough.

The “sleeping giant” is Computer Science itself. It is now a large, well-funded and popular field of study. While “algorithms” and “computer graphics” are only a small part of the total, they are of considerable popular interest. They attract many of the brightest technical students, often bringing dreams of writing new operating systems, or computer games. While Computer Science was not originally much of a Spatial Science, this is no longer true. Computer graphics, artificial vision, robotics, games and various other developments have greatly increased their interest in spatial problems. Computational geometry, while often abstruse, has provided a solid theoretical grounding for the solution of spatial algorithms. Indeed, it is clear that many of the older GIS questions concerning polygon overlay, etc., have new and better solutions readily available in the literature. It has been said, perhaps somewhat cynically, that GIS will soon be “just a part of IT (Information Technology)”. At the low end this is clearly already true – maps are as available for business software as are spreadsheets or pie charts. Less noticed, perhaps, is the increasing interest of Computer Science specialists in GIS or Spatial Science problems. I could cite Snoeyink, van Kreveld, Sack, de Berg and others who are directly publishing in GIS journals, as well as in those of Computer Science. In addition, there are many peripheral topics that are being well covered: network analysis, image processing, graph structures and many others. While there used to be a lack of understanding of the “real world” applications, this is less and less true. As I see it, a tidal wave is coming, and we must develop a strategy to meet it.

The need for joint programmes

If we do not develop a response, then we will become what we have often chosen to be: tool users, not tool developers. There will remain a role in technical training: how to use product X for application Y. There will remain a role in data collection, precision and quality. There will remain a variety of consultancy projects that require an understanding of the client’s problem. But we will be pushed more and more toward the margins.

If we are to have any long-term influence on the direction of what is still, for the moment, our discipline, we must find ways to join forces while there is still time. At the theoretical level this is already being done, in a small way, but the bigger problem is in the field of education. We can continue to perform technical training. We can continue to hold non-technical discussions of regional issues, and educate students from a management perspective. But we will be unable to influence the direction of technology, or to claim that we are training the next generation of experts. That will have been taken from us (indeed, some of it already has).

I have been in the situation of teaching spatial algorithms and data structures to very varied audiences for the last 15 years. It is my opinion that a good set of appropriate computing skills can be provided to non- Computer Scientists in a modest number of courses, depending on the level (undergraduate or postgraduate) and the background of the students. This material does not cover proofs and theorems in depth, but will explain the results in straightforward language. Nor does it attempt to cover all possible algorithms for any particular problem – we will leave this for the Computer Scientists. However, it has been my experience that the tools made available in this way are not only sufficient to let the students develop everything from simple to fairly elaborate computer programs for their application needs, but that these tools also serve to give them a background for critical evaluation of tools and products available in the marketplace. In addition, my experience has been that these skills are sufficient to let the student continue onwards to develop new and significant research contributions that would not have been feasible otherwise.

Nonetheless, this kind of work should be seen as an introduction to more advanced Computer Science skills for those who have both the spatial skills and the interest to pursue further research and software development. Students starting with one or other variant of this programme have ended up working in game development and virtual reality in Silicon Valley, and GIS software development in Canada and the US, as well as within the more traditional GIS employment in government and industry. Thus I believe the results to be very fruitful, so long as one remembers that the overall inspiration comes from both Geomatics and Computer Science, and will only continue to be useful as long as it is the result of collaboration and communication between the two.

A suggested programme on spatial algorithms for non-programmers

The underlying motivation for the whole programme, whether taught as a B.Sc minor or as a suite of post-graduate courses, is based on the view that spatial data is different from non-spatial data because of the existence of spatial relationships between objects near to each other. Where there is no spatial linkage, either explicit or implicit, then traditional tools, such as DBMS, may suffice. However, when we need spatial relationships – some form of topology – then things become more complex. In brief, there are many effective, and often straightforward, algorithms that may be performed on graphs, since the connectivity is a binary condition – a pair of vertices are connected or not. This fits very well with the discrete nature of computer operations.

By contrast, the original input data usually arrives in the form of coordinate pairs (or occasionally triples, for 3D problems). As is well known, these coordinates have only finite precision, and this leads to difficulties in connecting boundaries together, generating good surface models or simulating change over time. This is compounded by our habit of discretizing space (finite precision coordinates, or generating raster grids), attributes (such as classification of polygon data) and time (where simulation is usually performed at incremental time steps). See Gold (1996) for more details.

Thus in many cases the particular problem for spatial data is the transformation from coordinate data to a graph structure. In order to understand this problem, and to resolve it for at least simple cases, we need to develop tools to develop elementary geometrical primitive operations (such as for sidedness, orientation and intersection), as well as to understand the nature of basic coordinate transformations (affine transformations, map projections, etc.). In addition, an introduction to standard Computer Science data structures is needed. These are usually for one-dimensional problems, such as searching and sorting, but these concepts are necessary preliminaries for the later two dimensional structures used for graphs.

The first part of the existing programme is therefore intended to provide an introduction to spatial algorithms and data structures. An introductory course in linear algebra is required, as is practical experience using a structured programming (and preferably an object-oriented) language. The student acquires a working knowledge of the basic algorithms needed for simple applications in Geomatics, and the ability to develop the appropriate software.

Topics covered include: an introduction to various definitions of space; the distinction between fields and objects; discretization of space in the digital computer; coordinates and their limitations; vector algebra in two and three dimensions; intersection and area calculations; perspective transformations; rubber-sheeting and digitizing corrections; perspective map projections; simple abstract data structures: stacks, queues, linked lists, trees; tree traversal; one and two dimensional searching; raster structures and spatial indexing.

In the second part of the programme the student acquires a working knowledge of spatial data structures and algorithms needed for map input and construction . Emphasis is placed on the map as a graph structure, and on the development of algorithms for its input and maintenance. In many cases the simple Voronoi diagram serves as an elegant transformation between the coordinate and the graph views of spatial data.

Topics include: the definition of a graph; planar graphs; graph duals; the map as a graph; classification of possible data structures, and examples; weighted graphs; graph traversals (shortest path, minimum spanning tree, etc.); flow algorithms; polygon map construction; map input (from coordinates to graphs);traditional structures and their problems; simple Voronoi diagrams and Delaunay triangulation algorithms and data structures; map input; the Quad-Edge and Quad-Arc data structures; flow modelling; interpolation; moving point algorithms, navigation and robot movement; dynamic line segment algorithms; map history maintenance; overlay, terrain and network models and simulation.

Conclusions

At the end of this, a student with a modest prior background in mathematics and programming is able to complete projects for polygon construction, triangulation-based terrain modelling, network flow, map projections, and various others, as has been my experience over the last 15 years at the technical college level, in undergraduate geography, and in graduate programmes in Geomatics. This has typically taken three undergraduate courses or one (preferably two) post-graduate courses. However, these are extremely labour-intensive, as is usually the case with courses having a strong emphasis on hands-on programming.

It is surprising how far these techniques may be taken. Offshoots of this basic programme have led to effective research projects on dynamic mapping, on navigation problems, on the processing of scanned maps, of map update maintenance, and a variety of others. See Gold (1999) for an overview from the point of view of the development of a marine GIS. Other aspects have led to fruitful research questions resolved in collaboration with Computer Scientists – for my most recent case see Gold and Snoeyink (2000). For earlier ones, and a variety of interesting applications, see www.voronoi.com.

It does not seem appropriate to delve further into the details of the research directions here. However, they are mentioned briefly to show that a modest emphasis on computer science basics, together with some of the simpler spatial algorithms, opens up new vistas to Geomatics students whose previous computer background had been limited. A modest investment (at the level of a minor at the B.Sc. level) greatly increases the skills, and the marketability, of the students. It must be emphasized that this must be an option, rather than a compulsory portion of a Geomatics programme, as not everyone likes computers that much! Those who do have derived great pleasure and satisfaction from the programme – and have often gone on to achieve surprising things, as already mentioned. It has, in addition, brought in students from other disciplines, such as engineering and geology, because the same need exists for the management of spatial data, along with the same paucity of resources.

Thus the introduction of a minor portion of Computer Science into a Geomatics programme has, I believe, been shown to be both valuable and successful. In addition, the link with Computer Science can be seen to be a necessary part of our own future, as the search for effective and elegant models and algorithms for spatial problems continues to develop.

References

1989, Gold, C.M. Breadth versus depth: the dilemma of G.I.S. education. Proceedings, G.I.S. - challenge for the 1990s, Ottawa, 1989, pp. 886-900.

1996, Gold, C.M. An event-driven approach to spatio-temporal mapping, Geomatica , v. 50, no. 4, pp. 415-424.

1999, Gold, C.M, Chapter 4: An algorithmic approach to marine GIS, in: Wright, D. and Bartlett, D. (eds.) Marine and Coastal Geographical Information Systems, Taylor and Francis, London.

2000, Gold, C.M. and Snoeyink, J. A one-step crust and skeleton extraction algorithm. Algorithmica (in press).