Computer Algebra Systems and Computational physics

Chandra Sheth [a,1]

[a] *Department of Physics, University of Swaziland, P.Bag 4, Kwaluseni, M201, Swaziland*

Abstract

Computer algebra systems (CAS) is attractive due to its capability of symbolic manipulations, graphics and ease with which normal mathematical requirements are met. Science and engineering students who lack the basic computer literacy, CAS is seen to train these students quickly for computational techniques, including basics of programming. CAS based programs meet majority of the requirements of "computational science" and only partially of computationally intensive work. A hybrid program with CAS and a high level programming language is suggested as the best alternative.

Key words: Computational science education; computational physics education; scientific computing

1. Introduction

In recent years, there has been substantial progress in the use of computational techniques in science, engineering, economics and arts. The overall activity can be classified as "Computational Science" and therefore we have a new breed of scientists and researchers called "Computational Scientists". The real problems in majority of cases do not have analytical solutions. The only approach available is numerical solutions. The expertise required in this case is multi-disciplinary, that is, a knowledge of mathematical tools, computer science and a command on the subject of individual interest. We have a top-down situation as illustrated in Fig. 1.

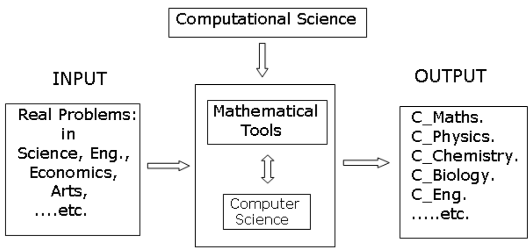

Fig. 1.

We clearly see that mathematical tools and computer science play a primary role. We also notice that, both have "histories", in other words, techniques we wish to use depend much on the initial beginnings. **e.g.** Evolution of programming languages and other software.

On the other hand, data input (result of a measurement or modeling parameters) is the present reality and does not depend on histories. However, data has no meaning without an interface with information (existing individual knowledge) and together we have **"input"** This is the input we want to use which has associated histories and could be an averaged situation. Now given an input and the primary resources, what options do we have to get "proper" output? This seems to be the most difficult part for all computational scientists.

We may like to answer this question, by relying on our experience. Can the experience, which is already biased, "search" the unknown in its absolute

[1] e-mail: cvsheth@yahoo.com, csheth@uniswacc.uniswa.sz

form? Truly, we are all involved in "research", that is, searching the unknown with the known. With this in mind, let us examine our primary resources:

2. Structure of Computational Science

2.1. *Computer Science*

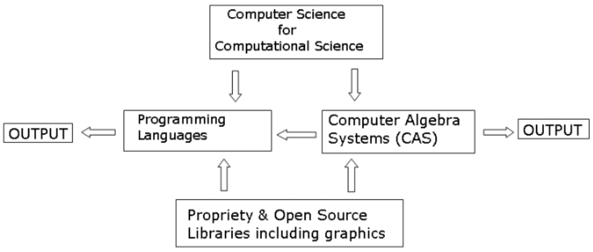

Fig. 2.

CAS: We have Maple, MathCad, Mathematica, MatLab etc. CAS has higher-level mathematical libraries and graphics as part of the software package unlike programming languages. Syntax used in various CAS systems differs marginally in spite of having same numerical or analytical approaches. Each has certain advantages. Each of these have their own style of programming some of the features coming from C, C++, fortran etc.

We clearly see that Computer Algebra Systems (CAS) allow us to practice programming languages through interface for code generation and not vice-versa unless one is a software developer. Another important advantage of CAS is symbolic manipulations along with inbuilt graphics. Due to its limited capacity of large-scale data handling and limitation of array manipulations, CAS is not suitable for large- scale calculations or modeling. **However CAS gives the opportunity to initially explore the problem quickly with minimal demand on the expertise on computational or mathematical tools and hence has an advantage in its use at undergraduate and graduate training**.

2.2. *Mathematical Tools*

The mathematical tools required in majority of applications can be classified into following:
– Differential equations,
– Linear& non-linear equations,
– Integrals,
– Linear Algebra,
– Statistical packages,
– Special Topics: FFT, Random Number Generators, Cellular Automata etc.

We note that, the mathematical tools available for programming are only numerical and are available through propriety or open source resources.

One of the logical implication of primary resource grouping, is to note that a computational physicist need to concentrate on the solution of a given physics problem than becoming a programmer. **One of the basic problem, computational community is facing, the lack of common programming or CAS styles. This has resulted breakdown in efficient communication among computational scientists**. The only solution is to have efficient translator software package. The recent versions do indicate some progress in this direction.

3. A undergraduate program in Computational Physics (CP)

Based on the ideas described above, it is possible to have an undergraduate computational physics program as follows:

(i) **Introductory level with CAS:** After few practice sessions on CAS software basics that include programming aspects within CAS environment, applications in chosen topics of physics are practiced. We do assume that the students have all the necessary mathematical background including numerical methods. This level can be run without any familiarity of the students on any computer based computation techniques other than a foundation level, which includes word-processing and "excel" like packages. Therefore, this type of program can run at the beginning of undergraduate program.

(ii) **Advanced level with a selected programming language:** With the initial level course as a foundation, a student can now enter into programming exercises, if necessary, with some help of code generation from CAS. In some cases, comparing one's computational skills with available CAS procedures. The speed of learning and confidence building is very high with the students. Examples, related to physics applications can be selected based on the local interests and the level selected.

One of the bottlenecks in having such a program is time-factor. One of the solutions is to integrate the essential topics with CAS. e.g. Quantum mechanics taught in a normal style (a necessary approach for all physics courses) and solving intensive analytical, numerical and graphical exercises with the use of CAS.

4. Conclusion

For the developing countries, since CAS systems are not open source software, there are financial implications. Our experience shows trainees, with no previous background of programming or scientific computing, learn to solve the problems and get introduced to programming techniques quickly. Implementation of such a program depends on the availability of manpower and will by the existing academics for the suggested change and integration. In the developing countries this is a serious problem. I have elaborated on these problems and recommendations with reference to **African context** in my previous presentations/discussions at CCP(1). We have tried to implement some of the recommendations by trying to provide essential software resources in the form of programs through a website and the response was very poor or non-existent. However, with the progress in CAS systems, this has been addressed but at a price. **I think, in the global scenario, there is a need for a change the way physics is taught at undergraduate and graduate level. I strongly recommend some amount of integration of computational methods with physics courses. For this we need a serious effort by all the stakeholders, be it a developed or developing country. One of the important driving principal is the potential of employment and inter-disciplinary foundation gain by such a program**.

We in the Physics Department, University of Swaziland, are trying to practice these ideas. Our initial results show a promising success. We now demonstrate few examples of such a program (2; 3) [2].

Finally I wish to note, there is the necessity to unify all the fields using computational techniques under one single group: **"Computational Science"** or **"Computational Science and Engineering"** with its appropriate affiliations, such as physics, chemistry, biology etc. Such a grouping will be beneficial to all of us in terms of inter-disciplinary academic co-operation, communication, standardization of curriculum in terms of certification and accelerating the deserving recognition of the activity such as Computational Physics by industries. **We need "Society of Computational Science" or "Scientific Computing Society"**.

[2] Some physics examples will be demonstrated using Maple.

Acknowledgements

I wish to thank the organizing committee of CCP2006 for giving me this opportunity to exchange few of my ideas.

References

[1] C.V. Sheth, Proceedings of CCP2001 and 2002.

[2] C.V. Sheth , Unpublished work and author's teaching material, 2005-06.

[3] Maple help page in Maple 9.5, MapleSoft.