

Supporting Teaching and Learning of Optimisation Algorithms with Visualisation Techniques

D. K. Liu, S.D. Huang, T.A. Brown
Faculty of Engineering, Level 6, Building 2,
University of Technology, Sydney
{[dkliu](mailto:dkliu@eng.uts.edu.au), [sdhuang](mailto:sdhuang@eng.uts.edu.au), [terry.brown](mailto:terry.brown@eng.uts.edu.au)}@eng.uts.edu.au

***Abstract:** Many engineering students find the advanced mathematics behind optimisation algorithms are difficult to understand. In order to assist teaching and student learning of design optimisation algorithms in the subject “Optimisation Methods and Applications in Engineering”, visualisation and animation techniques have been applied. A MATLAB based visualisation package, which consists of a number of visualisation programs associated with various optimisation algorithms, has been deployed. It provides a mean of interactive learning for students doing this subject. This paper describes the structure of the package and uses two example modules to show how these programs help students to understand abstract mathematical concepts and optimisation algorithms and their applications in solving optimal design problems. The student feedback and the subject survey results presented in this paper partially support the effect of the usage of the visualisation package on student learning.*

Introduction

Students in engineering need to understand both the theoretical and practical aspects of engineering design. Many subjects that engineering students have to do require knowledge of advanced mathematics. Optimisation Methods and Applications in Engineering is one such subject taught at the University of Technology, Sydney.

Teaching and learning (T&L) of this subject is challenging. The material covered in the Optimisation subject consists of both theoretical algorithms and practical applications. Students need to understand the algorithms that involve advanced mathematical concepts. Many students do not like (or are not good at) mathematics but they are usually more interested in, and motivated by, practical applications, i.e. solving practical problems using optimisation algorithms. It is particularly difficult to present algorithms to students by following traditional mathematics based teaching methods. Therefore, we have attempted to teach this subject using innovative assistive tools. Visualisation techniques have been used to present the course material and concepts visually.

Visualisation and animation techniques can facilitate students gaining important insights when learning abstract mathematical concepts and attempting to solve practical problems. Visualisation is a means of connecting mathematics and engineering. For example, visualisation based on MATHEMATICA, computational software, has been used to assist teaching and learning of mathematics courses [Kim, et al, 2003]. Besides the T&L of mathematics, visualisation techniques have also been used to assist T&L of computer programming [Baldwin, et al, 2000], computer networks [Kethireddy, et al, 2004] and operating systems [Sparks, et al, 2000], to support tele-teaching [Brand, et al, 2000] and teaching and learning of simplex algorithms [Lazaridis, et al, 2003].

In order to support the teaching of the subject Optimisation Methods and Applications in Engineering, and to assist student learning, we developed a visualisation and animation package. This package has been developed in the MATLAB[®] environment. The features of the programs include actual solutions to various optimisation problems, on screen animation and interactivity. Lecturers make use of the

programs to enhance their teaching and presentation and students use them for their study, for solving assignment problems and especially for solving optimal design problems in their workplace.

This paper explains how these programs may be effectively used in the teaching of optimisation algorithms and illustrates how students can use this package to visualize optimisation concepts and applications of optimisation algorithms in solving practical optimal design problems. The paper also describes the components of the visualisation and animation package and presents two visualisation modules: a one-dimensional nonlinear programming module, and a global optimisation module. Feedback results show that the students feel using the package is valuable. Finally, conclusions are drawn and possible improvements and extensions of the work are discussed.

The visualisation package

This package consists of a number of modules associated with various optimisation algorithms. Figure 1 shows the components of the package. It contains 6 modules and each module has a few visualisation programs. A number of practical optimal design problems in areas such as mechanical, structural, civil, transport and manufacturing engineering are included in this package and used for demonstration of the various optimisation algorithms.

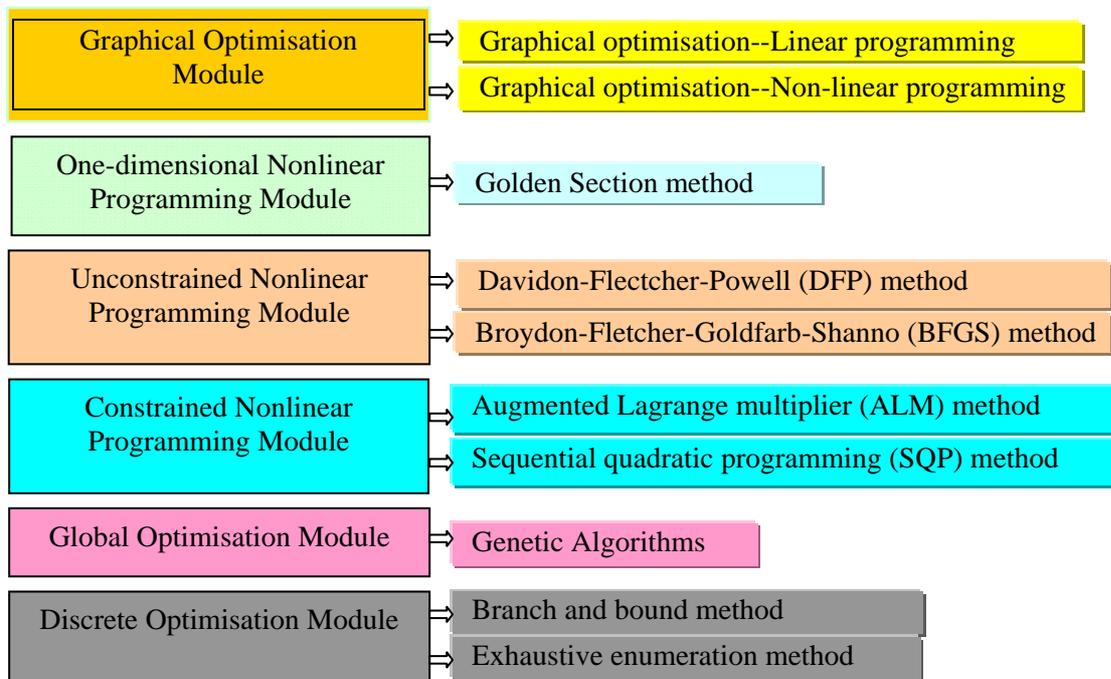


Figure 1: Components of the package

Most of the modules were developed in the MATLAB[®] environment. Some of them were developed based on the MATLAB code provided in [Venkataraman, 2002] by adding visualisation and animation programs. Students are encouraged to run the programs in MATLAB environment that lets them interact with the programs and apply the programs to solve real (often work-based) optimisation problems.

Example modules

One-dimensional nonlinear programming module

One-dimensional nonlinear optimisation algorithms play a key role in most of the multi-dimensional nonlinear optimisation algorithms (e.g. determining the step size in a given search direction). The

Golden Section method is one of the popular one-dimensional nonlinear optimisation algorithms. Important concepts in this method include bracketing, step size variation and golden section ratio. A visualisation program is developed for the Golden Section method to clearly explain:

- (1) the process of bracketing the optimal solution,
- (2) the iterative reduction of the interval size, and
- (3) the effectiveness of using the golden section ratio.

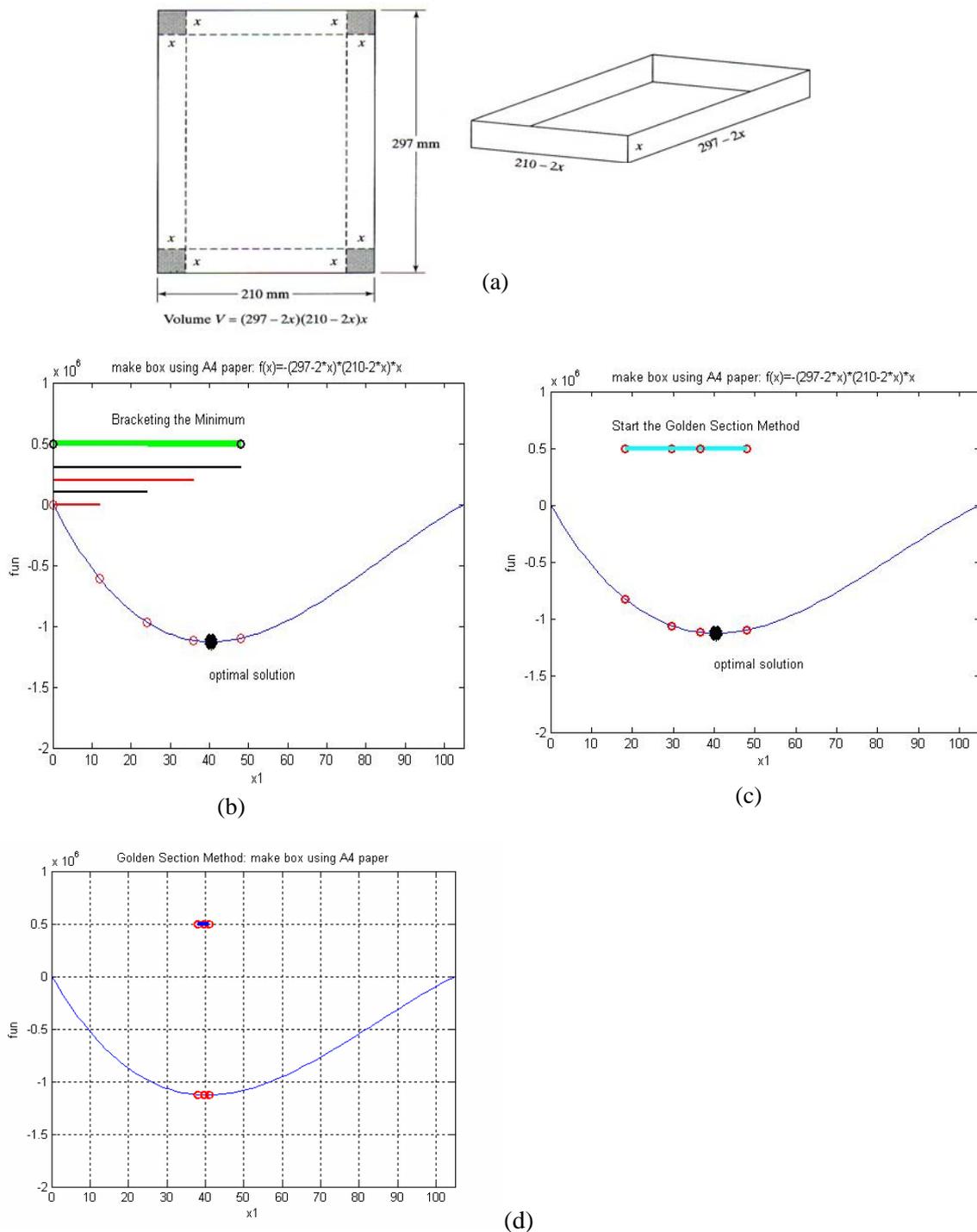


Figure 2: One-dimensional optimal design problem and visualized solution process by Golden Section method

One of the one-dimensional optimal design examples in this module is making an open box using an A4 paper by cutting four squares of side x from the corners and folding and gluing the edges (Figure 2(a)),

The volume of the box is to be maximized [Belegundu, et al, 1999]. Figure 2(b) and Figure 2(c) are two intermediate figures obtained in running the program for this example. To solve this optimization problem with the visualisation module, students will first see the bracketing stage step by step as shown in Figure 2(b), after they set the objective and constraint functions to the module. In this stage, the algorithm finds the interval that contains the minimum by incrementally changing the size of the interval.

With the final interval, students will see how the Golden Section method chose the two points inside the interval based on the golden section ratio and how the search region can be reduced (step size changing). Figure 2(c) shows a search region at the beginning of optimal design. The region is gradually reduced step by step, Figure 2(d) shows the reduced region after several steps. If students change the step size and tolerance/accuracy of solution, they can observe visually how step size and tolerance affect convergence and the results, and gain insight and greater understanding of why the step size and the tolerance value need to be adjusted to make the algorithm converge more quickly.

Genetic algorithm module

Genetic algorithms (GA) are one of the modern powerful global optimisation methods. The concepts upon which GAs are based are totally different from traditional optimisation algorithms. The important concepts of a GA are interactively demonstrated by the visualisation program module, including:

- (1) the population evolution process,
- (2) global optimal solution search performance, and
- (3) genetic operators (crossover, mutation, etc.) and their effect on the performance of GA and on the solution of the problem.

One of the optimal design examples in this module is to find the global maximum of a multi-peak function (Equation 1). Once students set up the objective function, the variable x and genetic operators, they can see the evolutionary process in solving this problem with a GA. Figure 3 shows a demonstration of the GA visualisation module by calling GeneHunter functions [GeneHunter]. The dots in the figure represent the individuals (solutions) of a population in a generation. At the beginning of evolution, the dots are scattered in the whole search range because they are generated randomly (Figure 3(a)).

$$f(x) = -5(x-1)^2 + 8\cos(4\pi x) + 3\cos(6\pi x) + 20, \quad 0 \leq x \leq 2 \quad (1)$$

After a number of generations, e.g. 56 generations, most of the individuals (dots) converge to the global maximum or the near global maximum as shown in Figure 3(b), which demonstrates clearly the population evolution process and the performance of GA in finding global optimum. The change of the fitness function values can also be visualised in the user interface.

Students are allowed to choose genetic operators, e.g. crossover, mutation, elitism, etc., and their probabilities through the interface. By doing this, students can gain deeper understanding of how genetic operators and/or GA parameters such as probabilities of crossover and mutation affect the GA's performance, i.e. convergence, efficiency and global optimal solution. This gives students greater insight into, and conceptual understanding of, GAs. Students can also use the module to solve their practical and work related optimisation problems.

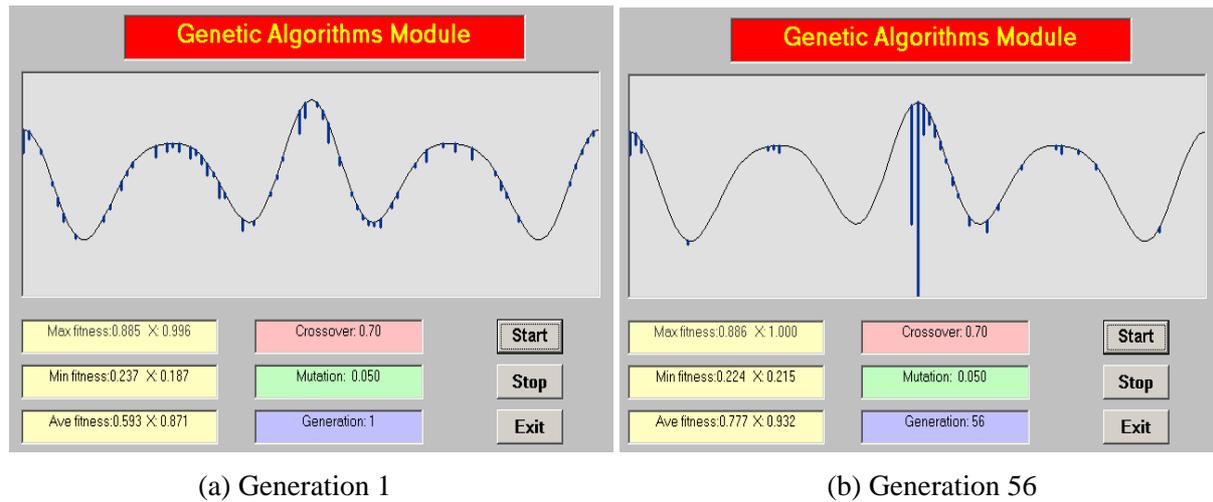


Figure 3: Visualisation of GA search process

Student feedback and discussions

The visualisation and animation package was first used in the teaching of the subject *Optimisation Methods and Applications in Engineering* in Spring semester 2004. Modifications and improvement have been made based on student feedback since then. It has been observed that both the students and lecturers benefit significantly from using the package. As the lecturers of this subject, we found that the optimisation concepts and algorithms can be clearly explained by using the package. In order to know what students say about the package, we informally interviewed many students doing this subject in 2005 and 2006. Student feedback and comments are overall positive. Majority of the students appreciated the development of the package and said that it is easy to follow when the algorithms are taught in this way.

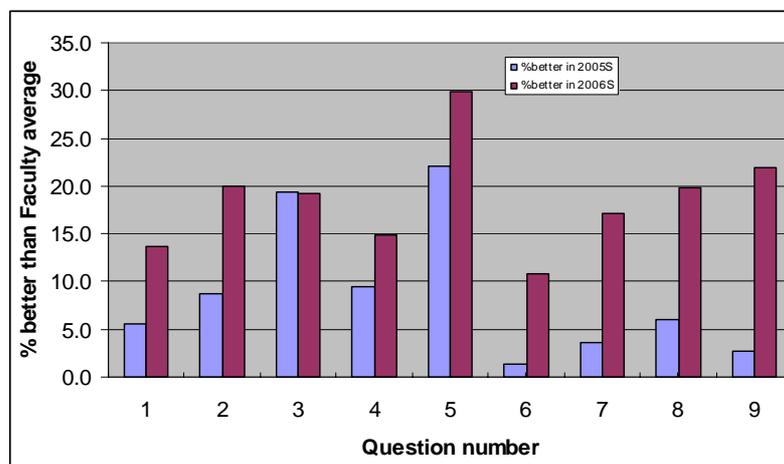


Figure 4: Student feedback results comparing the Faculty average

Apart from the use of the developed visualisation and animation package in this subject, another initiative is relating the subject to industry. Students are encouraged to bring optimisation problems from their workplace as their assignment/project problems in this subject. We help them to solve their problems. Students showed great interests in learning and applying design optimisation techniques to solve practical design problems in their work. For example, one student applied a nonlinear programming algorithm to solve a chemical storage tank design problem during the Spring semester 2004, i.e. minimizing the cost of building a concrete bund for chemical storage tanks layout.

With the use of visualization techniques and the visualisation and animation package, along with other efforts in the teaching of this subject, we received quite positive student feedback results in the subject

survey. Among the nine questions asked in the subject survey, four of them are relevant to the use of the package: (Q1) The subject was delivered in a way which was consistent with its stated objectives; (Q4) There were appropriate resources available to support the subject; (Q7) The teacher is able to explain concepts clearly; (Q9) Overall I am satisfied with the quality of this subject. Student responses to all the nine questions asked in the survey are above the average rating for all the subjects within the Engineering Faculty in 2005-2006. Figure 4 shows the percentage above the Faculty average in Spring semester 2005 (2005S) and Spring semester 2006 (2006S). It can also be seen from Figure 4 that student responses in 2006 are much better than those in 2005, which indicates in part that the improvements of the visualisation package provided better support to student learning.

Conclusion and Remarks

We consent that the use of visualisation techniques and the visualisation package in the teaching of the subject “Optimisation Algorithms and Applications in Engineering” can help students learning. In particular, it helps students to understand abstract mathematical concepts, provides students with useful tools for solving optimal design problems. The visualisation package also helps us, as lecturers of this subject, to explain mathematical algorithms and optimal design procedures.

Although the use of the visualization techniques and package is promising, some issues are realized. For example, some students did not have enough MATLAB programming skills. Another issue is that a friendly user interface is needed, which is part of our future improvement on the package. More surveys, e.g., pre and post surveys, are also needed for the purpose of evaluating the effect of the package on student learning outcomes.

References

- Hyang Sook Kim, Young Mi Kim (2003), *Visualization for Math-Education using Mathematica*, 2003 International Mathematica Symposium
- Baldwin, L.P., Kuljis, J. (2000), Visualisation techniques for learning and teaching programming, *Proceedings of the 22nd International Conference on Information Technology Interfaces (ITI)*, 13-16 June 2000, pp:83–90.
- Kethireddy, J., Suthaharan, S. (2004), Visualization Teaching Tool for Simulation of OSI Seven Layer Architecture, *Proceedings of IEEE SoutheastCon*, 2004, 26-29 Mar 2004, pp:335 – 342.
- Sparks, V., Suthaharan, S. (2000), *Operating systems: visualization technique for teaching and learning*, Proceedings of the IEEE Southeastcon, 7-9 April 2000, pp:387 – 390.
- Brand, O., Petrak, L., Sturzebecher, D., Zitterbart, M. (2000), *Supporting tele-teaching: visualization aspects*, Proceedings of the 26th Euromicro conference, 5-7 Sept 2000, Volume: 2, pp:64 – 71.
- Lazaridis, V., Samaras, N.; Zissopoulos, D.(2003), *Visualization and teaching simplex algorithm*, Proceedings the 3rd IEEE International Conference on Advanced Learning Technologies 9-11 July 2003, pp:270 – 271.
- Venkataraman, P. (2002), *Applied Optimisation with Matlab Programming*, John Wiley & Sons.
- Belegundu, A.D. and Chandrupatla, T.R. (1999), *Optimisation Concepts and Applications in Engineering*, Prentice-Hall, Upper Saddle River, New Jersey
- GeneHunter, Ward Systems Group, Inc. <http://www.wardsystems.com/products.asp?p=genehunter>

Copyright statement

Copyright © 2007 D.K. Liu, S.D. Huang, T.A. Brown: The authors assign to AaeE and educational non-profit institutions a non-exclusive licence to use this document for personal use and in courses of instruction provided that the article is used in full and this copyright statement is reproduced. The authors also grant a non-exclusive licence to AaeE to publish this document in full on the World Wide Web (prime sites and mirrors) on CD-ROM and in printed form within the AaeE 2007 conference proceedings. Any other usage is prohibited without the express permission of the authors.