Designing Computer System Engineering Programs: Single-track or Multi-track ?

Harvey A. Cohen
Computer Science and Computer Engineering Department
La Trobe University
Bundoora Victoria 3083

Abstract

Four year undergraduate degree programs in Computer Systems Engineering, that integrate hardware and software engineering aspects of computer technology and science have recently been established. The published program outlines is used as a basis for describing the scope and in determining the implicit philosophy and rationale implicit. There is much commonality in the published program details, but there are some striking differences. The major difference lies in the frameworks of the programs, whether a single course, the "single-track" model, or one admitting multiple streams of specialty, the "multi-track model". Two recently instituted programs, the La Trobe "multi-track" CSE, and the SAIT "single-track" CSE are contrasted. There is a very high degree of commonality in the two programs. In the "multi-track" course examined, scope for an impressive level in specialisation in software engineering, digital computer engineering, and network engineering has been traded for somewhat less in science fundamentals, and less training of the specialists in the other two specialties. These differences in course model are highlighted when considering the scope for using these CSE degree programs as vehicles for the emerging specialty of Imaging Engineering.

Introduction

Combined Electrical Engineering and Science Degree Courses have a long history. In the post-World War II period the science content was devoted to Physics and/or Mathematics; in that era the overlap between Electrical Engineering and Science was such that there were no curriculum issues involved in such joint degrees.

The first Computer Science Department in Australia, the Basser Department of the University of Sydney, was initially established as part of the School of Physics; while similarly, early computing was linked to the Physics Department at the University of Melbourne. Other Computer Science Departments as they became established were located inside schools or faculties that were essentially physical science, or placed in other settings, but rarely within Electrical Engineering. A marked exception to this situation was the University of Newcastle which established the first Computer Engineering degree course in 1974, within Engineering. The Australian situation was in contrast with that in the US, where many, but not all, Computer Science Departments were located within Electrical Engineering Schools. As evidence of current US trends it is pointed out that at the University of Washington, the Computer Science Department, for long located within the College of Arts and Sciences, had its name changed to the Department of Computer Science and Engineering, and was in 1989 relocated within the College of Engineering. In parallel changes in Australia: Computer Science and Electrical Engineering were placed in a single faculty at the University of Melbourne; and elsewhere a number of established departments and faculties have been renamed to indicate a scope that includes both engineering and computer systems.

Since, until recently in Australia, Electrical and Electronic Engineering departments have been in separate schools or faculties from Computer Science. This separation by itself has limited professional engineering training in Australia. To meet the demand and need to equip students with qualifications in both Computer Science and Electrical Engineering, various combined Electrical Engineering and Computer Science DOUBLE degree programs have been established over the past few years. These programs have involved some measure of syllabus modification and re-arrangement so that students could complete the units required for two degrees over a five full-time year period. However, within these double degrees are EE and CS courses no different than those separately available.

Early in the 80's, the Institute of Engineers established a Task Force, whose initial objective of examining the role of software engineering in engineering became focused on Computer Systems Engineering as an emerging engineering discipline. Subsequently, a six year sandwich course was initiated at the N.S.W. University of Technology (UTS), whose first students will shortly graduate. Other courses have been initiated, and more recently, Computer Systems Engineering courses have been initiated at La Trobe University, and the South Australian Institute of Technology (SAIT).

It must be mentioned that initially the pioneering program in Computer Engineering at the University of Newcastle could be characterised as a course in Electronics Engineering with a number of computer and microprocessor options, but since the merger of the Departments of Electrical Engineering and Computer Science five years ago this course has moved some way towards computer systems.

It is impossible within the confines of a conference paper to give a full comprehensive analysis of these new programs in C.S.E., and the trends within evolving courses. The objective here is to indicate just what characterises computer systems engineering as an engineering course yet in many ways non-traditional, and to suggest further evolutionary trends.

For completeness the analysis is prefaced with a definition of Computer Systems Engineering (C.S.E.) This definition suggests that CSE is a coherent area of Engineering, but should a program in CSE be unitary or generalist, with only modest number of options, or should it properly be a program offering specialties WITHIN CSE? To provide substance to a discussion, a comparison is made between the published course outlines of the 'single track' CSE program at the South Australian Institute of Technology (S.A.I.T.), and the 'multi-track' CSE program at La Trobe University. As evidence of the further specialties arising within CSE, the topic of Imaging Engineering is described, and discussions are presented how, and to what extent, this specialty may be accommodated within unitary or multi-track programs in CSE. Finally we present some general conclusions.

Definition of Computer Systems Engineering

The National Panel on Software Engineering of the I.E.A. prepared a report, "Perspectives on Engineering and Computing Systems", in May 1985 which defined Computer Systems Engineering (CSE) *as:*

Computer Systems Engineering is the professional engineering discipline which covers the activities required to create a computing system to achieve an end-purpose in its own right and includes the design, construction, and effective integration of hardware and/or software components.

Although the concept of an 'end-purpose in its own right' is, to say the least, obscure, the IEA specification that CSE 'includes the design, construction, and effective integration of hardware and/or software components' does encapsulate well the substance of C.S.E. expertise, but omits a vital ingredient well stated by Belcher et al (1987) as "formally enhanced skills of abstraction" . It is vital that our engineers gain an overview of design, development, and life-cycle issues.

Two New CSE Courses

In order to provide a concrete basis for discussion, a table has been prepared showing in tabular form the short form syllabus of the CSE course due to start in 1991 at La Trobe University, along the equally newly inaugurated course in CSE of the SAIT (which is to become the University of SA, Levels Campus in 1991). Both courses are equally 'modern' and have very much in common; yet there are some significant differences.

The reader is referred to the Table presented as an appendix to this paper. Note that for the SAIT course, the number of semester hours - (an hour per week for 13 weeks) - is given to give an accessible idea of the 'size' of each component. In the case of La Trobe, only "unit weights" are given; unfortunately the size of a "unit" is different at each year. A similar succinct version of the syllabus of the N.S.W. University of Technology course in CSE is contained in Belcher et al (1987). However, the UTS course is a sandwich/part-time course over six years.

The La Trobe program in CSE, offered by the Department of Computer Science and Computer Engineering with some participation with the Department of Electronic Engineering, permits students to major in one of three specialties:

Digital Computer Engineering
Network Engineering
Software Engineering

For all three streams, the first two years are common. There is some choice as to components chosen within the subject Computer Science III.
The SAIT program was developed at SAIT by Computer Studies and Electronics. The course includes a number of electives (amounting in Year 4 to 8 semester hours).

The most striking feature of the La Trobe program is the definitive treatment of software engineering as an engineering discipline per se. Note that a software engineering (SE) specialist of this course will have completed two years of Electronics, which includes both analogue and digital electronics, and an introduction to communication systems. He/She will also have encountered in year 3 further computer architecture, and alternate architectures, including multi-processor systems. However, fully 40% of year 4 will be devoted to Software Engineering IV, a subject directed to formal methods of software specification and verification, software quality control and management, system test beds, CASE tools, software reliability, software engineering economics, and various advanced issues. In sum the CSE student majoring in software engineering will certainly be an extremely well-trained software engineer, but will have a really solid knowledge of computer hardware and networks.

There is no Chemistry at all in the La Trobe program, while the SAIT program includes engineering chemistry and materials. (Year 1, 4 semester hours). In dropping Chemistry, La Trobe is following the pattern of other Victorian universities in related Electrical Engineering programs.

The Physics content of both programs seem at first sight to be comparable, but only at SAIT is there a definite course in electromagnetic waves, appropriate to a course in (low-level) signal transmission, and full appreciation of optical fibres. Note however that the La Trobe program includes Applied Mathematics II, which has the mathematical elements of field theory, Laplace and other solutions to PDE's,, and an introduction to numerical methods (with Gaussian elimination)

Both programs give significant attention to digital communication and networks; to catch the spirit of the differences in style one needs to examine the detailed proposals. Examining program outlines in modest detail, it appears fair to regard the La Trobe Computer Systems IIIC course where higher level protocols are introduced to be more or less equivalent to the SAIT Computer Networks course. Note however, that La Trobe DCE and SE majors will encounter networks only in a component of Computer Science III. However, NE specialists at La Trobe do Computer Network Engineering IV, which covers formal description, conformance testing and interoperability testing of higher layer protocols, particularly the transport, session, presentation and application layer protocols, design of specialized networks such as EFTPOS, interworking of OSI networks and non-OSI networks (such as SNA). In short - the advantage of a modest degree of specialisation has been extended study (for specialists only) for Protocol Engineering.

Imaging Engineering

Imaging Engineering is defined by the writer as the engineering specialty that is deals with the Capture, Transmission, Coding, Analysis, Reconstruction, Synthesis, and Data-base Issues of multi-dimensional data using a combination of software and hardware techniques. Although that part of imaging engineering concerned with the capture or sensing of images in analogue form arose with the birth of television, the science of imaging originated primarily in the early space exploration projects. In the early 60's at JPL techniques were developed to refine images from deep space for human viewing. The extension of the scope of imaging to three dimensional images arose with the development of computer tomography for medical images. Note that the definition I have given embraces the whole gamut of imaging, from high level to low level. What can be described as a low level task is the refinement of mouse-drawn curves for graphics purposes and the shaping and refinement of 'blown-up' digitised fonts. Feature detection and character recognition, as exemplified by targetting and machine reading of the number-plates of car 'captured' by police speed and stop cameras. (Cohen, 1990). The higher level task of determining the three dimensional surfaces revealed by two dimensional images, by image pairs in stereo vision, or by using shape from shading, is clearly a sub-set of image analysis. This higher level task, the major part of Computer Vision, is also classified as a part of Artificial Intelligence. However, it is notable that over the past ten years, international computer vision conferences have been primarily conducted by IEEE sub-groups, sometimes in conjunction with the IAPR (Image and Pattern Recognition International Society), and not held as part of AI conferences. Machine vision can be described as covering the application of imaging engineering to problems of production and assembly, including quality control. Thus machine vision is the less theoretical, and more applied specialty linked to Computer Vision. International conferences on machine vision have been primarily sponsored by IAPR, with some participation by IEEE, although the SME also conducts the major machine vision conference, simply called 'Vision', whilst machine vision is also a major part of robotics and computer aided manufacture conferences. The link between Computer Vision and Image Synthesis has been highlighted over the past ten years as certain major researchers such as Terzopoulos, (1987 a,b) have contributed significantly both to the major annual graphics conference SIGGRAPH, and to such CV conferences as the 1st International Conference on Computer Vision, held at London in 1987.

Until very recently, the science content of imaging engineering has been seen as procedural or 'how to do it' knowledge of image analysis procedures and synthesis. About ten years ago, the first examples of 'imaging theory' arose with the development of the concept of 'scale-space' in explaining the scaling of images. (cf Witkin, 1983). More recently, with research and other work devoted to classifying images for archival and other data-base related purposes, data base issues have emerged as a vital theoretical component. (Sammet 1986). My inclusion of Image Synthesis within Image Engineering is by no means novel, as is attested by the fact that the oldest journal in imaging, the Journal of Computer Vision, Graphics, and Image Processing was established by Rosenfeld in 1971 as the journal of Graphics and Image Processing. However, there is a not well appreciated interplay between texture recognition and surface rendering, between edge and surface recognition and highlighting (as Phong shading), that makes a distinction between the image synthesis and image analysis largely a matter of emphasis. (Suter and Cohen, 1986). This interplay has become enriched as fractals, used over the past ten years for rendering realistic synthetic images, have been demonstrated as of utility for highly impressive image compression (Barnsley 88, Cohen 89).

The problems of processing images which are often data structures of huge size and complexity has lead to great attention to distributed and parallel processing, to a concern for algorithmic complexity, hardware/software trade-offs, and to overall system design and integration, all recognisably aspects of CSE. Neural net computational models have shown potential for significant application in the image analysis area.

Integration of Imaging Engineering into CSE

In this section the possible way, and the extent to which, Imaging Engineering could be integrated into the two CSE programs discussed above (that at La Trobe and that at SALT). This examination has interest both because of the significance of imaging, and as an indicator for other emerging specialties within CSE

In the La Trobe CSE, an introduction to basic image processing, including the capture of images, and elementary convolution, is included with the first year half unit Computer Systems 1. In the year 3 subject, Computer Science III (of size six components) and also in Computer Science IIIA, (of size 5 components including CS3BM and CS3FGL) the optional components include CS3VIS Computer Vision and image/speech processing, and CS3DSP Digital Signal Processing, and CS3GRA Graphics Programming. Thus in the existing CSE program for network engineering, a significant introduction to imaging is available. However, it would be vital for a specialist imaging engineer to have also taken, in a four year period, CS3DIS distributed computing, CS3PAR parallel computing, and at least one of CS#AIP Artificial intelligence programming, CS3AIS Artificial intelligence systems, while CS3DBS Database systems, or a new DB component geared more to image data bases and/or geographic information systems would be important. Hence, to make available a specialty stream in imaging comparable to the existing three specialties, one could require of the imaging specialist program that it be identical to either the networks or the DCE majors s, except for the replacement of the NE or DCE based Computer Systems IV by an imaging based Computer Sys Eng IV in year 4. Greater specialism would be obtained by developing a specific imaging Comp. Sys. III, which could be structured so that either in Computer Sci III or in Computer Sys III all the relevant Computer Science II components were taken, together with an introduction to imaging engineering.

In the SAIT program, there is significant digital signal processing content, including aspects of digital filtering, but no speech fundamentals or speech coding. There is a second year subject Graphics and User Interfaces. An introduction to image analysis is available as one elective in Year 4. To augment the imaging engineering content of this program minimally, would require the availability of further such imaging electives. Of course, the Computer Systems Project in year 4 could be directed to imaging engineering. Machine vision would naturally form part of the content of the third year unit on Computer-Aided Manufacturing. In sum, this more generalist program at SAIT does provide an introduction can support modest degree of specialization in imaging engineering.

Other issues and questions

Belcher et al (1987) have detailed the difficulties in properly implementing C.S.E. in a tertiary institution unfamiliar with dealing such a rapidly evolving high technology field. Some of these issues, such as the need for multiple users licenses for software used at networked work stations, can be solved by admitting that CSE has a high recurrent funding needs, in addition to the high capital and establishment costs.

The issues of maintaining a teaching staff that is both abreast of current technology and possibilities as well as fully aware of the state and needs of Australian industry is also canvassed by Belcher, who favours part-time involvement in teaching by engineers from industry.

In looking and comparing various CSE programs, many questions arise. In listing these questions I have tried to highlight context.

How much physics? Some is essential, but must all CSE students study field theory?
Students need some course in material science, but do they need chemistry?
At what level of description should material science operate?
Students need Boolean algebra, and finite state machines, but do they need to know about related applications in software engineering as decision table approaches to design (especially of real-time systems)?
Students need Pascal, C, exposure to assembler, but what about AI languages, LISP, SCHEME, and Prolog? And expert system shells? Neural nets?
Students must experience CAD or CAM tools - but should all CSE students have a formal course in graphics programming?
Should students have formal exposure to symbolic mathematics, and/or to *Mathematica?*
Many new products depend on human factors, on the man-machine interface; how should this anticipated need be met?
Should there be a subject leading to philosophic insights into CSE? Should all read Moravec?

Conclusions

The Institute of Engineers Australia in 'Engineering Education to the Year 2000' argued that
The technological core of the engineering course should be of primary importance and as general as possible.
However, on the preceding page, in Recommendation 33, it was asserted that
Identification of discrete engineering disciplines should not be allowed to inhibit new fields ...

Computer Systems Engineering is centred about a vital area of engineering, covering much of what the general public identifies as 'high tech' and is growing at a prodigious rate. (cf Osborne, 1979) No one degree program can hope to cater for the twin needs of general expertise, and well as specialised expertise within CSE. In this paper two new programs in CSE were described, one a single-track program, the other a multi-track offering scope for specialization.

An outline of Imaging Engineering, an engineering specialty based on imaging science combined with CSE, was described as a further instance of an area of growing importance within CSE Integration of Imaging Engineering within both single-track and multi-track CSE has been discussed, to show how this new specialty can be accommodated.

Disclaimer

The opinions and proposals expressed in this paper are the personal opinions of the author.

References

Barnsley, M.F. and A.D. Sloan, (1988) *A Better Way to Compress Images*, Byte January 1988 pp 215-223. (1988)

Belcher, W.R., C.E. Peterson and J.R. Leaney, *(1987)The NSWIT Computer Systems Engineering Course - and the Educational Challenge* Proc. 1987 I.E.A. Computing Systems Engineering Conference, Brisbane

Cohen, Harvey A. (1989) *The Application of IFS [Iterated Function Systems] to Image Analysis*, Proceedings of the IEEE International Conference on Image Processing ICIP'89, Singapore, Sept 5-8, 1989, Vol 2, pp 583-587.

Cohen, Harvey A. and Alan L. Harvey, (1990) *Targetting Number Plates Effectively using Sparse/Full Templates and Coarse/Fine Template Marching* Paper accepted for Publication in the Proceedings of the IAPR International Workshop on Machine Vision Applications, MVA'90, Tokyo, Japan, Nov 24-28, 1990

Goaling, W., (1984)*Changing Role of the Engineer in the Electronics Industry* Proc. IEE Vol 131 Part A No 9 pp 665.

Institute of Engineers, Australia, (1985) *Perspectives on Engineering and Computing Systems* Report of Working Party on Software Engineering.

Institute of Engineers, Australia, (1987) *Engineering Education to the Year 2000* Report of the Task Force on Engineering Education.

Moravec, H., (1988) *Mind Children: The Future of Robot and Human Intelligence* Harvard University Press, Cambridge Mass.

Osborne,A., (1979) *Running Wild: the next industrial revolution* Osborne-McGraw-Hill., Berkeley, California.

Samet, H. and M. Tamminen, (1986) *An Improved Approach to Connected Component Labelling of Images* Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR'86, Miami Beach, Florida, June 1986, pp 312-318.

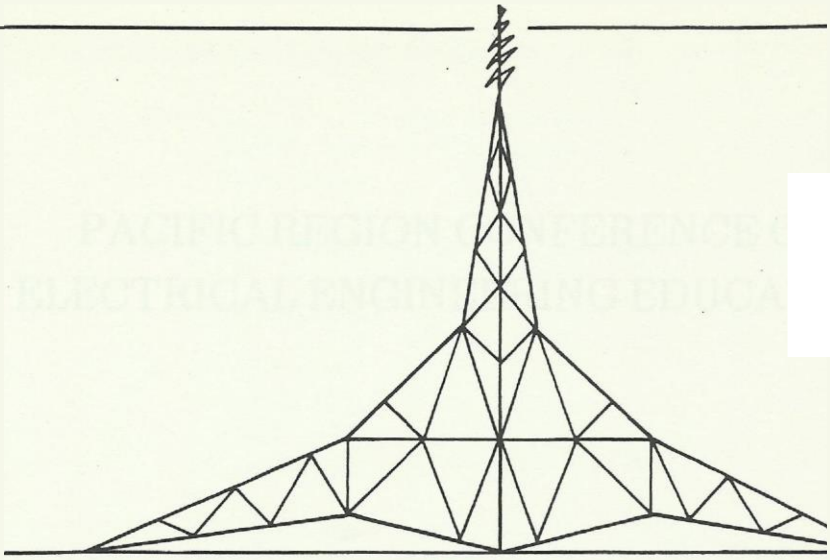
Suter, D and H.A. Cohen, (1987) *Fractals: representations for visual recognition, and for graphics*, Proceedings of AUSGRAPH 87 , Perth May 1987.

Terzopoulos, D. (1987 a) *Elastically Deformable Models,in* SIGGRAPH'87 Conference Proceedings, Computer Graphics, Vol 21,m No 4, pp205-214

Terzopoulos, D.A. Witkin and M. Kass, (1987 b) *SymmetrySeeking Models for 3D Object Reconstruction* Proc 1st International Conf. Computer Vision, ICCV'87, London, pp 269-283.

Witkin, A.P., (1983) *Scalespace Filtering* Proc IJCAI'83, Karlsrule, West Germany, Vol 2, pp 1019-22.

SAIT Program in CSE	La Trobe Program in CSE
[semester hours given]	**[unit values given]**
First Year [47]	**First Year (Common)** [4.5]
Engineering Mathematics I + **II** [8]	Electronics I 1
Engineering Physics I [4]	Physics 1 CS 0.5
Engineering Chemistry & Materials 1 [4]	Computer Science I 1
Engineering Mechanics 1[3]	Computer Systems I 0.5
Engineering Drawing 1[2]	Mathematics IA 1
Introduction to Computing [2]	Mathematics 1DM 0.5
Electronic Fundamentals 1+2 [8]	
Language in Use [2]	**Second Year (Common)** [3.5]
Engineering Physics 2 [4]	Electronics II 1
Discrete Mathematics [2]	Applied Maths II 0.5
Programming (Pascal) [4]	Computer Systems Engineering II 1
Electronic Processes [4]	Computer Science II 1
Second Year [45]	**YEAR 3 [2.4]**
Mathematics for Electronics 1 [3]	
Physics for Computer Engineering 1 [4]	**Y3:Software Engineering Major**
Data Structures & Algorithms 1+2 [6]	Computer Sci III A 0.8
Network Theory [3]	Computer Sys Eng IIIB 0.5
Electronic Devices & Circuits [4]	Software Eng III 0.7
Digital Circuits 1[4]	Statistics III CS 0.2
Communication Studies [2]	Soc. Impl. of Eng. 0.2
Mathematics for Electronics 2 [3]	
Physics for Computer Engineering 2 [2]	**Y3:Digital Computer Engineering Major**
Signals and Systems 1 [4]	Computer Sci III 1
Linear Circuits 1 [3]	Comp Sys Eng ILIA 0.5
Computer Hardware 1 [3]	Comp Sys Eng HIB 0.5
Queuing & Scheduling [2]	Statistics DICS 0.2
Materials 2 [2]	Soc. Imp. of Eng. 0.2
Third Year [47]	**Y3:Network Engineering Major**
Mathematics for Electronics 3 + 4 [6]	Computer Sci III 1.0
Graphics & User Interfaces [2]	Comp. Syst. Eng. HIA 0.5
Software Engineering 1+2 [8]	Comp. Syst. Eng. 11ICS 0.2
Digital Circuits 2 [4]	Soc Imp of Eng 0.2
Computer Hardware 2 + 3 [4]	
Signals and Systems 2 [4]	**YEAR 4 [1.25]**
Software Development Environments [4]	**Y4:Software Engineering Major**
Operating Systems [2]	Comp. Sci. IVA 0.5
Computer-Aided Manufacture [2]	Software Eng IV 0.5
Control Systems Engineering [3]	Comp. Sys. Proj IV 0.25
Electronic and Communication Engineering[4]	
RealTime Systems [4]	**Y4:Digital Computer Engineering Major**
Contrasting Studies [2]	Comp. Sci. IVA 0.5
	Comp. Sys. Eng. WA 0.25
Fourth Year [44]	Comp. Sys. Eng IVB 0.25
Computer Hardware Design [5]	Comp. Sys. Proj IV 0.25
Computer Communications [2]	
Digital Signal Processing [2]	**Y4:Network Engineering Major**
Computer Systems Design [4]	Comp. Sci. WA 0.5
Computer Systems Project [8 formal hours]	Comp. Sys. Eng. IVA 0.25
Product Management 1+2 [6]	Comp. Net. Eng. IV 0.25
Elective Subjects I+II [8]	Comp. Sys. Proj IV. 0.25
Computer Networks [2]	
Data Management [3]	
The Professional in Society [2]	
Contrasting Studies [2]	



PRCEEE - 90

PACIFIC REGION CONFERENCE ON ELECTRICAL ENGINEERING EDUCATION

CONFERENCE

PROCEEDINGS

PHILLIP ISLAND

17 - 20 DECEMBER 1990

VENUE

THE CONTINENTAL

PHILLIP ISLAND