Engineering Thinking and Rhetoric

JOHN A. ROBINSON
Faculty of Engineering and Applied Science
Memorial University of Newfoundland
St. John's, NF, Canada

ABSTRACT

Engineers seek optimal solutions to problems. The solution criteria are often of several different types, and there is no formal way to find the best trade-offs. Nevertheless, engineers make judgments and provide explanations to justify their choices. The thinking which identifies a particular solution as optimal relies on deduction and analogy. The rhetoric which explains the choice of solution uses precedents in a similar way to legal argument. Therefore the formal study of analogical thinking has a role in engineering education, and descriptive case-based examples are important to the student, not just as illustrations, but as source analogs for problem solving.

I. INTRODUCTION

William Wordsworth, in "The Prelude,"[1] writes:

> Science appears as what in truth she is,
> Not as our glory and our absolute boast,
> But as a succedaneum*, and a prop
> To our infirmity.

Scientists are certain that Wordsworth got it wrong. Science is our glory, they say, because its formal, systematic, unbiased method is our most reliable route to discovering the truth about the universe. Providing "a prop for our infirmity" sounds much more like engineering—solving problems to meet human needs. That cause is a noble one, and engineers are happy to embrace it. (For a similarly poetic evocation of this, see reference 2.) But does engineering also provide a way of thinking, distinct from that of science, that gives it its own intellectual glory? On surface evidence, it appears not. Thousands of books have been published on the philosophy of science, very few on the philosophy of engineering. Engineers find intellectual depth within their discipline, but they rarely claim that there is anything special, fundamental, or glorious about their way of thinking.

But if there is anything distinctive about engineering thinking compared with, for example, the thinking of mathematicians, physicists, anthropologists, or historians, it is certainly worthwhile trying to state what it is. A complete characterization of engineering thinking would mark out the intellectual scope of engineering, showing how it draws on and contributes to other areas of scholarship. It would identify the academic role of engineering education. It would reveal what kind of argument or rhetoric is appropriate for explaining engineering design decisions. Towards these worthwhile ends, this paper attempts a preliminary exploration of engineering thinking, identifying the central role of analogy in finding and justifying engineering solutions.

II. WHAT ENGINEERS DO

The Oxford English Dictionary[3] defines an engineer as one who "contrives, designs or invents; an author, designer; also an inventor, plotter, a layer of snares." The Encyclopaedia Britannica says "engineering [is] the application of scientific principles to the *optimal* conversion of natural resources into structures, machines, products, systems and processes for the benefit of humankind."[4] Campbell Martin succinctly identifies the "essence of the engineering approach" as "using models to make proper decisions."[5] Many alternative definitions are given in the Engineering FAQ for the sci.eng.mech Usenet newsgroup,[6] and University calendars have still others. The following five-point description of engineering gives a synthesis:

- Engineering is applying scientific knowledge and mathematical analysis to the solution of practical problems.
- It usually involves designing and building artifacts.
- It seeks good, and if possible, optimum solutions according to well-defined criteria.
- It uses abstract and physical models to represent, understand, and interpret the world and its artifacts.
- It applies well-established principles and methods, and uses proven components and tools.

None of these definitions says how engineers think. What can be added to express the intellectual root of engineering? I suggest the following:

- Engineering is explaining why a particular solution to a problem is the best.

This supplementary definition builds on the idea of optimal problem solving already suggested in the earlier definitions, but it emphasizes explanation. The idea is that engineering has a rhetoric, or a mode of argument to justify what it does. Indeed, there are at least two modes of argument, and these depend on what the word "best" means for a particular problem. For some problems, which here will be termed "simple problems," best means the solution

* A succedaneum is a substitute.

which can be proved optimal through mathematical analysis or other deductive reasoning. For other problems, here called "compound problems," it is not possible to find such an analytic optimum, and best means the solution which is judged the most suitable tradeoff.

III. How Engineers Think

A. Simple Problems

In simple problems, the constraints and criteria for evaluating the solution are all qualitatively similar. Even difficult problems in computational terms can be simple according to this definition. The traveling salesman problem, which involves working out the shortest path to visit a number of cities, is computationally hard, but because it has a single evaluation criterion (distance) it is a simple problem. Many other engineering optimization problems are simple in this sense. Designing a circuit that has to meet its specification with the minimum number of devices is a simple problem, because two solutions can be compared and the better one selected.

The explanatory framework of simple problem solving is deductive. Engineers solving such problems are thinking like mathematicians, moving by logical stages through analysis to design. But even in simple problem solving, engineers look for evidence that the space of possible solutions was properly searched, and the chosen solution correctly proved to be optimal. Finding *a* solution is not enough; it must be shown to be the best.

B. Compound problems

In compound problems, the evaluation criteria are not qualitatively similar and cannot be jointly optimized. Engineering jobs which require the balancing of cost, safety and aesthetics are compound. Most systems engineering jobs are compound. Wherever there are choices of materials, subsystems, or methods that emphasize one or another property, the problem is compound. The engineer can now apply several strategies:

1. Disqualify (ignore) criteria that cannot be measured.
2. Express relative values of criteria in a common currency, based on some evidence, and thus reduce the problem to a simple one.
3. Divide the problem into parts which can be independently solved as simple problems.

Strategy 1 sometimes has to do. For example, it may be impossible to say how the aesthetics of a bridge are to be measured. However, if a criterion like aesthetics is rejected, there may still be some implicit lower limit on ugliness. It is part of the job of engineering, as an intellectual discipline, to understand how immeasurable but implicit criteria are to be dealt with.

Strategy 2 is widely applied. Cost-benefit analysis uses money as the common currency of diverse constraints and criteria. When engineers do this, they are acting like economists, and must answer the same economic and philosophical questions about attributed value. But engineers have a wider range of mappings between qualitatively different constraints. Speed/accuracy and speed/size are common tradeoffs. When the engineer chooses a tradeoff, a judgment is being made about relative value, and that must be explained.

Strategy 3 is pervasive. Most real engineering projects are decomposed into subproblems which are then solved almost independently. Explaining why the problem has been decomposed is usually easy—it would be insoluble otherwise. But engineers should also be able to explain why a particular decomposition has been chosen, to justify that the aggregate of optimal subproblem solutions will be the best overall solution, or close to it.

Compound problems include simple problems and their solution is therefore partly deductive. But trading off between qualitatively different domains requires a different kind of thinking that has similarities with legal reasoning. In law, some decisions are made by the interpretation of legislation; some by developing earlier case decisions. These two routes to a decision are different: the first applies an abstract rule to a particular instance, the second deals with an instance according to similar previous instances. The first is a top-down theory-to-application route; the second is a sideways precedents-to-application route. Compound problem solving uses the same two routes. Abstract rules are applied when the values of different courses of action can be measured and compared. But when quantitative comparison is impossible, exemplars (previous designs) have to be considered. By analogy with these precedents, compound problem solving decides on a best solution.

Practicing engineers probably make use of analogy as often as practicing lawyers. Reference to previous jobs, identifying similarities and differences, making linkages between contexts, are all regular habits. Often the analogies will be simple and direct, but, especially in systems engineering, the linkage can be between two very different domains. The ability to see analogical situations, particularly in balancing the values of different criteria, is central to engineering judgment. The ability to explain these analogies, and argue their relevance, is engineering rhetoric.

IV. Links with Other Disciplines

Engineering solves problems using physical science and mathematics. Its links to those disciplines are clear. Yet, in terms of engineering thinking and rhetoric, its dependence on them is accidental rather than essential. Engineering's goal (problem solving) and its method (deduction and analogy) are much closer to medicine and ethics than to science. Its rhetoric (justifying its analogies) is close to law, and perhaps to economics. Table 1 summarizes three approaches to thinking, which groups engineering with these disciplines.

Engineering does differ from other disciplines that rely on analogical reasoning. For medicine and law it is usually easy to define the terms of success. Not so for engineering, which must begin its

	Aim	Method	Argument	Application
Science	To explain	Observe Hypothesize Test	Falsifiable hypothesis has been corroborated and not refuted	Physics, Chemistry, Psychology...
Humanities	To interpret	Collect Critique Synthesize	Interpretation is coherent and revealing	Literature, Metaphysics, History...
Engineering	To solve	Specify Design Verify	Solution is optimal analytically or by analogy	Engineering, Medicine, Law...

Table 1. Categorization of modes of thinking used in introducing students to engineering thinking and rhetoric.

search for solutions by demanding clarity on what sort of solutions will do, and how they will be measured. The criterion question, "How will I know I have succeeded?," is the first step in design, uncovering user requirements, presuppositions, physical limitations, and values.

V. Teaching Engineering

Engineering students are taught both simple and compound problem solving. Because the modes of thinking are different, the teaching methods should also be different.

Simple problem solving is deductive. It may be taught abstractly, rigorously, using mathematical analysis. In simple problem solving, illustrative examples do not validate concepts; the concepts stand in their own right, derived from theory. Examples simply help students to master the application of those concepts.

In compound problem solving, abstractions and general theories are still important, but equally the role of analogy should be taught and exploited. Examples are more than just learning aids now. They are the raw materials of analogical problem solving. First, then, students should understand how to think analogically. Courses on critical thinking are useful for alerting students to the dangers of reasoning based on patterns (including analogies) alone. However, they rarely incorporate insights about how to use analogy safely and effectively. This subject is now beginning to be taught in philosophy departments alongside traditional reasoning courses (see, for example references 7-9), and engineering students could be a prime audience.

Analogy-based teaching in the engineering discipline itself means that students learn new subjects by encountering casebooks of practical examples. These can be drawn from the teacher's own design activity, from classic examples of engineering successes and failures, and from hands-on work. Laboratory experiments are fundamental to learning compound problem solving. It is important to stress this to students who often see labs as merely illustrating the theories taught in class; instead labs are providing them with a gamut of analogies from which to reason.

Because design methodology is common to all kinds of engineering, it has often been understood, researched, and taught as an abstraction. This reflects the stance of simple problem solving — develop a common theory, then apply it to examples. But if the understanding of compound problem solving as analogical judgment is correct, design methodology should be taught from cases (examples). Rather than learning to solve problems by following a generic recipe, students learn by experiencing many analogous previous problems, their solutions, and their solution processes, and then applying this information. Students themselves may abstract or generalize a design methodology from the examples which they see, but this will not be normative (that is, it will not be a recipe for all situations). It will simply describe the commonplace links between different instances of design.

In teaching courses in digital systems and software engineering, I have, over several years, introduced students to the subject with a discussion of analogy in engineering design. In particular, I explain how to recognize and apply relational and system mappings, as discussed in reference 9. The introductory lecture invariably excites a portion of the class, for whom seeing the relationship of engineering thinking to other disciplines is highly motivational. Within the courses I outline a framework for design, and appropriate analytic tools, but spend the bulk of the time on examples. These are deliberately chosen as a gamut of representative problems, with interconnecting analogical links that are repeatedly noted. When possible, exercises and assignments require students to give explanations of designs by identifying specific linkages with other designs. The success of this mode of teaching is hard to measure. The bulk of the course material is conventional, it is only the high role given to reasoning from examples that is unusual. Student feedback is overwhelmingly positive, but I detect no effect (positive or negative) on exam scores, and there is no testable evidence about the long-term benefit.

VI. Conclusion

In making decisions concerning qualitatively different constraints and criteria, engineers draw on similar previous problems and solutions. Analogical reasoning is thus at the heart of engineering thinking. Engineering students should be trained in the use of analogy, and given a rich set of source analogs from which to reason.

Engineers are not alone in facing the problems of technology, society, and values, but they have a special responsibility. With training in finding solutions subject to qualitatively different criteria, engineers also have special expertise for meeting situations where costs must be balanced, but there are ambiguities about relative value. Understanding engineering thinking therefore leads to better training of engineers as society's servants.

References

1. Wordsworth, W., "The Prelude," in M L Reed (ed.) *The Thirteen-Book Prelude* (The quote occurs in book II), Cornell University Press, Ithaca, NY, 1991, and many other editions.

2. Kipling, R., "The Sons of Martha," in his *Complete Verse*, Anchor Press, New York, 1989, and many anthologies.

3. *Oxford English Dictionary*, Second Edition, Oxford University Press, 1989.

4. "Engineering," entry in *New Encyclopaedia Britannica Micropaedia*, 15th edition, 1997 revision.

5. Campbell, Martin, J., *The Successful Engineer*, McGraw-Hill, New York, 1993.

6. http://www.lance.colostate.edu/depts/me/FAQ/engineering.html.

7. Keane, M.T., *Analogical Problem Solving*, Ellis Horwood, Chichester, UK,1988.

8. Vosniadou, S., and A. Ortony (eds.), *Similarity and Analogical Reasoning*, Cambridge University Press, 1989.

9. Holyoak, K.J., and P. Thagard, *Mental Leaps: Analogy in Creative Thought*, MIT Press, Cambridge, MA, 1995.