Telling Tales: Models, Stories and Meanings

CHRIS BISSELL, CHRIS DILLON

Mathematical modelling forms an important part of engineering education and practice Yet precisely what is meant by the term 'modelling' is often extremely unclear - and, moreover, much of what students are told about the subject is considerably problematic from both a philosophical and a pedagogical point of view. In this article, we shall explore some of the issues behind mathematical modelling for technology, with reference to undergraduate teaching and professional practice.

Theory and practice in mathematical modelling

Traditional engineering education accords pre-eminence to a very particular set of mathematical techniques centred, in the main, on applied eighteenth- or nineteenth-century mathematics: algebra, calculus, vectors, matrices, etc. Students are required to carry out tasks such as algebraic manipulation, the setting up and solving of a (rather limited) range of differential equations, matrix inversion, manipulation of vector calculus expressions, and so on.

Yet although these techniques are presented to students as vital constituents of a modelling 'toolbox' fundamental to engineering practice, all too often what seems to be required in engineering education is simply the computation of 'right' answers to tried and trusted academic questions: the derivation of standard formulae, for example (and the reproduction in examination of such derivations), or the solution of linear differential equations with constant coefficients under given initial conditions

The majority of practising engineers, on the other hand, are unlikely to use many of these skills in the exercise of their profession Certainly, they will need to be able to select and use appropriate *standard* models, to scale and transform *known* results and to *manipulate* a great variety of notations and patterns - usually involving only fairly elementary mathematics. And increasingly, of course, they must be able to use mathematically-based computer tools with insight and understanding.

The modelling process

If the mathematical skill set required by engineers and technologists can be called into question, then the *process* of applying such skills is also problematic. A recent textbook on system dynamics and control listed the stages in the modelling and design process as follows:

- describe the system physically (*physical modelling*);
- describe the system mathematically (*model construction*);
- analyse the mathematical description (*model solution*);
- synthesise a preferred manifestation of the system (*system design*)

(Umez-Eronini, 1999, p 1)

Such a view of the mathematical modelling process is typically expanded, to include iteration, as in the so-called 'modelling cycle' shown in Figure 1. [1] It looks deceptively simple. All we need to do, it implies, is to apply our mathematical skills and iterate sufficiently to derive 'the' solution to any 'real-world' problem. Yet the figure barely hints at many of the practical difficulties encountered by anyone attempting to use this method. For example:

- how, precisely, do we 'formulate the problem'?
- what simplification is involved - how to decide?
- what counts as 'solving the mathematical problem'? (The manipulation of notations - as in conventional algebraic manipulation? .. the generation of texts - as in writing a computer program? ... simply plugging data into a commercial package?)
- how do we validate the result?

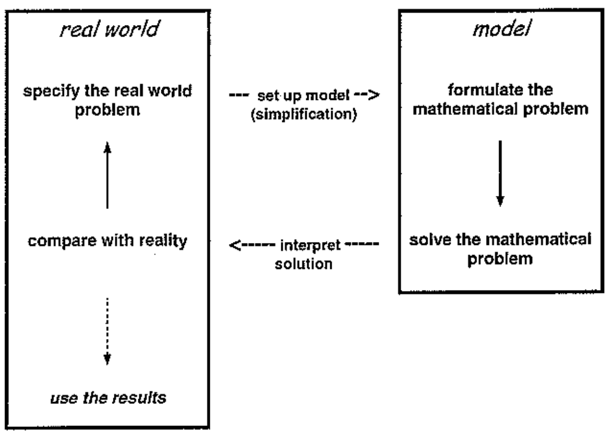

Figure 1 The 'hard' version of the modelling cycle

Most serious, perhaps, from both a philosophical and practical point of view, is the figure presupposes some Platonic correspondence between the 'real world' and the 'model'. How we deal with 'real-world' problems that do not map conveniently onto the given modelling domain is not considered at all: the implication is either that *every* problem can be so dealt with - or that those that cannot are literally not our problem.

A somewhat 'softer' version of the modelling cycle is shown as Figure 2 (OU, 1997). Here, there is (wisely) no attempt to mirror the 'real' world in a world of models and the particular processes involved in the generation, manipulation and evaluation of the model have not been identified

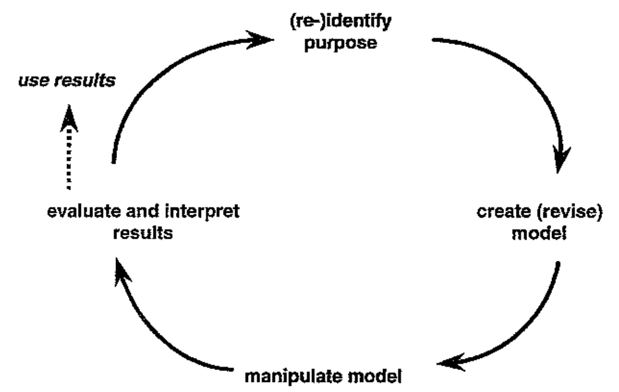

Figure 2 A less prescriptive, more flexible version of the modelling cycle

The implication is still, however, that a method exists which, if employed wisely, will lead to the required result. What neither version makes explicit are such considerations as:

- we do not, as a rule, start with a (metaphorical) blank sheet of paper; it is fairly unusual to create a model from 'first principles';
- engineers often select from well-known standard models, whose solutions are also well-known, and which merely have to be scaled, adapted or slightly modified;
- the modelling procedure is often incremental, based on the fine-tuning of existing models, and on practice and experience - including experience of models that have failed;
- modelling is not an algorithmic process, but is subjective, often relying on tacit knowledge or 'craft' skills unique to a given discipline;
- intuition is important - successful modellers often have a surprisingly well-developed 'feel' for the type of model likely to be successful in given circumstances;
- a mathematical model is useful only if it can be employed successfully - so an easily-solved, but less accurate, model may be far preferable to a more sophisticated but less tractable one

Using a mathematical model

When presenting the modelling process, teachers of mathematics, engineering and technology tend to emphasise the *creation* of the model. Yet, in technological practice, it is far more likely that a practitioner will be called upon to apply an existing model. It is useful to think of a hierarchy of skills involved in *using* a given mathematical model - skills quite closely linked to the stages of the cycle of Figure 2.

At the lowest level are manipulative mathematical skills: the ability to rearrange formulae, to substitute numbers correctly into an expression or to change the number or formula in a spreadsheet cell, for example. But being able to *manipulate* the formulae is of no use unless you are also able to *interpret* what you have done. (The two stages are iterative - the initial interpretation of your manipulated model may mean you have to start again and manipulate it in a new way.) Finally, you need to be able to *apply* the conclusions reached from your interpretation of the model - perhaps in the light of information other than that contained in the original model. As before, you may need to iterate - that is, review your interpretation and perhaps reinterpret the model before applying it anew. We can look at all this in a layered, hierarchical way, as illustrated in Figure 3.

application	ability to apply the interpretation and make appropriate recommendations; essentially 'proactive'
interpretation	ability to interpret the modified form of the model in a way relevant to the situation; essentially 'reactive'
manipulation	ability to modify the form of the basic model, using algebraic and other skills; essentially 'mechanical'

Figure 3 A hierachy of skills when using a given mathematical model

What is most striking, perhaps, about Figure 3, is that the majority of skills involved are not 'mathematical' in the conventional sense, although they assume a familiarity and ease with mathematical representations. (This will be discussed in much greater detail below, in the context of one particular type of engineering model.)

From hardware to mathematics

It is useful to distinguish between three major approaches to the problem of obtaining a mathematical representation of a technological process, system or object.

Empirical modelling
This is, in many ways, the least problematic – and the least interesting. By empirical modelling, we mean any technique for deriving a mathematical model with little or no underlying theory or assumptions about the object being modelled: for example, fitting a curve to a set of data points, the classic Reynolds expression for pipe flow in fluid dynamics or the corrections to radio propagation equations used to model urban mobile phone reception.

Physical modelling
This is the type of modelling most commonly taught as part of applicable mathematics courses. It involves using the assumptions and implications of some scientific or other law (such as those of Newton, Maxwell, Kirchhoff, Ohm, etc. [2]) to derive a mathematical model expressed in terms of physical variables such as mass, friction, voltage, current, etc. It is the type of modelling most susceptible to asking students to rehearse the derivation of accepted formulae.

System identification
The third approach is much less widely known outside engineering circles, and is known as *system identification*. In this approach, the system under consideration is treated as 'black box'. A model is derived by means of input–output testing (impulse or step response, frequency response, I/O correlation with random input). Typically, system identification operates with a restricted range of useful model types (often linear differential equations) and, most importantly, it delivers no model of individual *component* behaviour.

Simplification

The simplification process is clearly at the heart of mathematical modelling. It is itself a deceptively simple concept. In an attempt to shed light on it, we begin by distinguishing between 'simplification by abstraction' and 'simplification by idealisation' (see, for example, Morgan and Morrison, 1999, especially p. 38) – recognising, however, that these categories are by no means always clearly distinguished.

Simplification by abstraction
By 'abstraction', here we mean choosing to neglect a particular phenomenon or phenomena. A classic case would be to neglect friction in a model of a mechanical system. This type of simplification is associated primarily with what we have termed 'physical modelling', when it enables matters of minor interest (or ones too complicated to model) to be ignored. Note that we can envisage a 'hierarchy of abstraction', by which a model can become more sophisticated as fewer phenomena are neglected. We might start by neglecting friction entirely, for example, then we might include viscous friction (since it is easy to model by a differential equation) and finally we might allow static friction, with its non-linear complications.

Simplification by idealisation
By 'idealisation', we mean the process of deliberately treating one thing as an idealised 'other thing' for the purpose of modelling. A slightly varying waveform might be idealised as a constant value, for example, or a sudden change from one steady value to another as a mathematical 'step change' of infinite gradient. The idealisation process is, perhaps, inherent in system identification, where the best match to a limited range of standard, ideal models is sought. Note yet again, we can usefully think of a hierarchy of idealisation.

In modelling a hydraulic actuator, for example (as in a robot arm), we might first model the device as a perfect 'integrator' – that is, one for which a constant input (valve position) gives an output (actuator position) that increases steadily with a constant slope – as in the ideal mathematical integration of a constant value, shown in Figure 4.

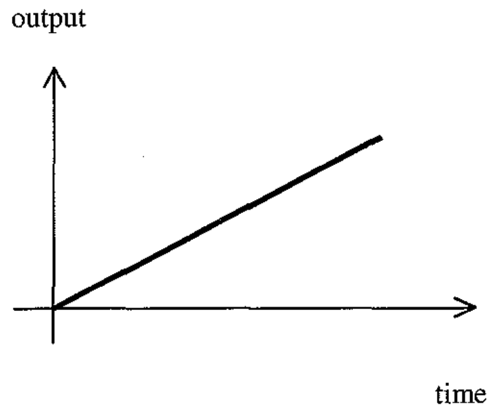

Figure 4 The step response of a pure integrator model

The next stage in such a hierarchy of idealisation might be to accept a 'lag' before the output responds fully to the input, as shown in Figure 5.

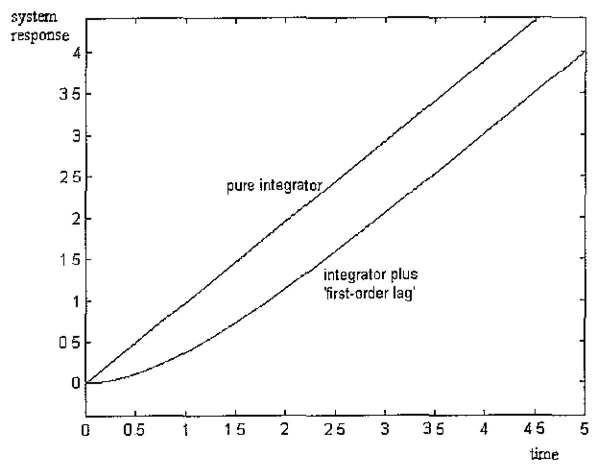

Figure 5 The step response of an integrator plus 'lag' [3]

Further levels in the hierarchy might allow for oscillatory behaviour of the actuator position, [4] a pure time delay or other more complex behaviour.

Generalisation
It is worth noting that both abstraction and idealisation can often be interpreted as *generalisation* - in other words, phenomena or systems which under other circumstances might be viewed as separate are lumped together as part of the modelling process. By choosing to neglect friction, for example, we deliberately abandon the distinction between static and viscous friction. Or we might idealise random noise in an electronic system as additive, white and Gaussian - without distinguishing between various non-ideal probability density functions encountered in practice.

Abstraction versus idealisation
Applying both the abstraction (physical modelling) and idealisation (system identification) processes to the same situation can often be very revealing. For example, ignoring friction, inertia and compressibility and using physical reasoning, leads to an integrator model of an actuator. Including inertia and viscous friction, and applying Newton's Laws, leads to a model of the form of Figure 5; while allowing for fluid compressibility as well admits the possibility of oscillation.

Yet the numerical values of the model parameters obtained by the two routes for a given system can vary considerably. System identification tends to give the 'best fit' to a given order of model, while physical analysis provides useful information about model sensitivity and the behaviour of individual components. *The two approaches are not equivalent.* In one practical example of modelling a robot arm, for example, estimates of important system parameters such as natural frequency differed by more than 50% using the two approaches [5].

Modelling versus design

The significance of the design process for technological modelling will be considered in more detail below. But to set the scene, consider the following statements:

> When modelling a system we cannot change, we often aim for a model of a type we can handle easily. When building a new device or system, we often try to design it to behave like a suitable ideal model.

For example, inflation can be *modelled* (idealised) as a geometric progression, leading to easy mathematics. A savings account, on the other hand, is *designed* to behave as a geometric progression: the mathematics *defines* the system. Or, to give a more technical example, Ohm's Relationship is the design brief for the device we call a resistor. V/I for a resistor is *designed* to be constant: it is not a natural feature of conductors in general. Similarly, inductors, capacitors, servomotors and a whole range of other technological artifacts are carefully engineered to behave in a way defined in advance by models with particular properties.

Telling tales – stories and models

Let us now step back from the problems of the modelling process(es) and pose the more general question: *What is the position of mathematics in engineering?* Our starting point is the observation that educators and professional engineering bodies often argue that formal mathematics is a central and essential part of an engineer's education (e.g. IMA, 1995; Croft *et al.*, 2000). However, the special status of mathematics does not seem to rub off on students who, after graduation, often claim that a lot of their mathematical expertise is largely irrelevant to their jobs.

What seems to be missing in educational practice is any link between the formal mathematical procedures that students are taught and the type of mathematics that practising engineers actually use. It is not that mathematics is not used at all, but that there is clearly a significant difference between what a mathematician calls 'doing mathematics' and what an engineer calls 'doing mathematics'.

We want to suggest that an overly formal mathematical approach can obscure those very areas which mathematics is supposed to illuminate. Attention is diverted away from the physical behaviour of the system and concentrated on the details of the mathematics. Mathematics then becomes a goal in itself and it is easy to forget or ignore the often quite fragile links that were originally set up between the system and its mathematical model. As a result, there can easily be a confusion between those results and procedures which relate primarily to the mathematical structure of the model and those which may be interpreted (with care) in terms of the behaviour of the system.

Engineered systems are complex structures in their own right, designed and built by people to fulfil particular functions. From an engineer's point of view, the goal is to produce a system whose behaviour achieves a particular function. Focusing too closely on mathematical details may not be helpful where the goal is to understand the behaviour of a system better: the engineer needs to be able to think and talk in ways that are appropriate to this goal. What counts as a valid engineering approach, therefore, is likely to differ from what counts as a valid mathematical approach.

Engineers use a variety of models to understand the systems they design and build. They can be mathematical, computer-based or physical; they may exist as equations, graphs, diagrams, description programs or scale models. Models are used for various purposes: to give insight, to provide a basis for explanations of behaviour and performance, to design new artefacts to perform particular functions or to predict behaviour of existing or non-existing systems under particular conditions. But a model of a system is neither the system itself, nor a self-evident representation of it.

Models are not autonomous, stand-alone entities encapsulating all that is worth knowing about a component or system in a convenient mathematical package. Rather, they are starting points for conversations among practitioners about the systems they are claimed to represent. Models have to be mediated and negotiated within a community of practice to make any sense. As part of their development, engineers learn how to talk about their models: they learn what stories to tell about them and to recognise what sorts of conversations are legitimate.

Engineering explanations

Models are used as part of explanations about why this or that happens, to predict future behaviour, to talk about as-yet-unbuilt devices. Explanations - answers to 'how', 'why' and 'what if' questions - can take different forms One formal way is by explanation-proof; a particular pattern of behaviour is shown to be the logical consequence of the structure of the model and a set of rules. Explanation is generated in this approach by referring to the rules and constraints of the mathematical model. This or that happens because, deductively and logically, nothing else could happen This is the mathematical approach and one that will be familiar to many engineering students

But this is not enough on its own: an engineering explanation does not begin and end with a mathematical procedure The problem with this approach is that it gives little or no insight into how a particular behaviour comes about. The solution of a set of equations is essentially a static process: the equations themselves are no more and no less than a set of constraints between variables and the solution is a realisation of those constraints 'all at once'.

Gaining insight into what is going on in a system requires more than just mathematics, however: it requires a story about the system that suggests reasons for a particular type of behaviour, proposals about what can be done to produce different behaviour and predictions about what may or may not occur in the future Learning how to tell these stories is part of the enculturation of the engineer.

Causal accounts

Stories told by engineers about system behaviour are often in the form of causal accounts A causal account provides a way of understanding how the system behaviour unfolds with time, often in the form of 'if this happens, then that will happen' statements. Causal accounts can be generated from the simplest of relationships.

For example, in an electrical circuit the voltage V and current I in a particular component may be related by the mathematical model V = IR: that is, the voltage you would measure across the component is directly proportional to the current flowing through it, where the constant of proportionality R is called the resistance Now you do not need to know about electrical circuits to see that this model is just an identity between mathematical variables The binding implicit in the model simply describes a constraint between voltage and current that is valid for every instant of time and for all time.

But if the relationship is defined for every instant, then how does change take place? What can be said to explain how changes in voltage and current propagate through a circuit? From an engineering point of view, the equation on its own is not enough to say much about electrical circuit behaviour and how changes in voltage and currents take place; it needs to be built on and a good story (one that is plausible and acceptable within the relevant community) told about what is going on.

An engineer might say something like "the model says that the voltage across the component is equal to the resistance multiplied by the current. This means that a change in the current causes the voltage to change" This interpretation indicates a small but highly significant shift. The story has changed subtly and now contains an implied time-ordering that is not contained in the mathematics of the model.

The equation model describes a situation in which changes in voltage and current happen simultaneously, because the left-hand side and the right-hand side are bound always to be equal in value. In contrast, the engineering account makes the change in the current a *cause* and the change in the voltage the resulting *effect*: that is, the change in current is the antecedent of the change in voltage.

This is a *causal account* of the relationship between current and voltage in the component. If this were put together with other causal accounts about other components in the electrical circuit, then a story could be told about how changes in voltage and current propagate through the circuit from component to component - thus explaining how and why a particular circuit behaves as it does Knowing how to turn models into plausible and acceptable stories about what is going on is part of the understanding of the engineer.

Engineering models

To illustrate quite how different the 'mathematical' and 'engineering' worlds can be, it is worth looking in a little detail at a specific example One of the resources in the standard library mentioned earlier is a particularly useful class of models with wide applications in modelling growth, decay and oscillation Such models are used in the design of many engineered systems, including industrial control, telecommunications and signal processing applications and domestic television and hi-fi equipment.

Classically, such models are expressed as linear differential equations with constant coefficients For example, a simple model that is often used where the system responds in a non-oscillatory way to input changes is the standard, first-order linear differential equation

$$\tau \frac{dy(t)}{dt} + y(t) = kx(t)$$

Here, $x(t)$ and $y(t)$ represent the variations with time of the input and output of the system. The parameter τ is called the time constant, and describes how fast the output responds to changes in the input The parameter k is a scaling factor that describes the magnitude of the output relative to the input. This model may used to describe, for example, the behaviour of an electric motor (where $x(t)$ is the input voltage variation and $y(t)$ is the resulting speed of the motor), an electrical amplifier (where $x(t)$ and $y(t)$ represent input and output voltages), or a thermal process (where $x(t)$ and $y(t)$ represent temperatures) [6]

Now a 'mathematical' approach to handling such a model is to set about solving the differential equation for various forcing functions $x(t)$ to find the response of the system to a particular input. However, in a practical system, the input will be a physical quantity whose variation with time, even if it were known, is unlikely to have a well-defined mathematical form.

Furthermore, the emphasis of the engineer is not on solving the equation but on interpreting the behaviour of the system The framework for doing this is to look at the system's predicted response not to every possible input but to very particular 'test' inputs. Interpreting the response to such inputs provides information that can be applied to a wide range of situations - and which draws on a common 'non-mathematical' engineering language

Standard input–output models

The two most common test inputs are the step change and the steady sinusoid. For the step change, the forcing function is just:

$$x(t) = u(t), \text{ where } u(t) = 0 \text{ for } t < 0$$
$$= 1 \text{ for } t \geq 0$$

For the steady sinusoid, it is:

$$x(t) = \cos(\omega t)$$

where $\omega = 2\pi/T$, where ω is the natural angular frequency of the sinusoid of period T.

The solution of the differential equation for each case is well-known. These are standard models, so engineers do not have to solve such equations from first principles each time. What is important is not the mathematical forms of the solutions but an understanding of how the results can be interpreted – and here graphical methods are widely used.

Figure 6 shows the general shape of the unit step response. Notice that it is drawn on normalised axes: the horizontal axis is time normalised to the time constant τ of the system and the vertical axis is the magnitude of the output, normalised to the scaling factor k.

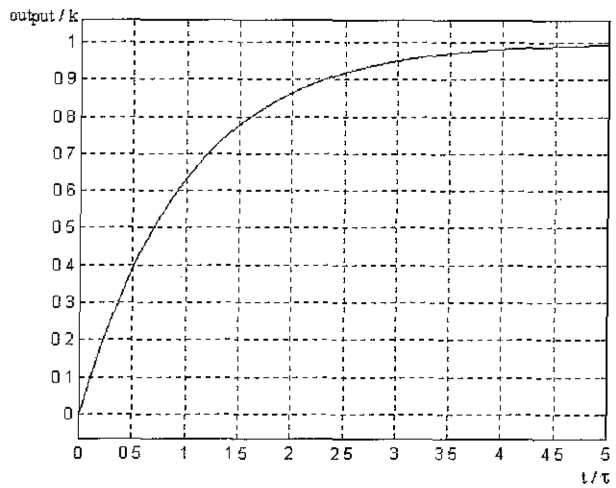

Figure 6 Normalised first-order step response

In engineering terms, what matters is an appreciation of the time constant as a measure of the speed at which the system responds to input changes, the gain or attenuation of the input, indicated by the value of k, and the time taken for the response to reach and settle at its new value after an input change. An engineering rule of thumb is that the output will have reached its final value (actually, 99 3% of its final value) after a period of time equal to five time constants: that is, where $t/\tau = 5$ on the normalised step-response curve. For any specific time constant, the actual time involved can be easily found.

For example, if the response of the system were characterised by a time constant of 0 1 second (very slow by electronics standards where time constants of the order of microseconds or less are common), then the system output would have completed 63% of its response to the input in 0 1 second, or reached its final value in about 0 5 seconds.

In the case of a steady sinusoidal test input, the output will also be a steady sinusoid at the same frequency as the input but with a different amplitude and phase shift. *Frequency response* curves can be plotted which show how the relative amplitude and phase of the output varies as the angular frequency ω of the input varies.

Figure 7 shows the frequency response curves associated with the first-order model, normalised as before with respect to the time constant and the scaling factor. Notice also that the engineering presentation of the curves uses a logarithmic scale for the horizontal normalised frequency axis. The relative amplitude is also logarithmic, since the vertical axis is plotted in terms of decibels, where the value in decibels of the output is equal to $20 \log_{10} k$.

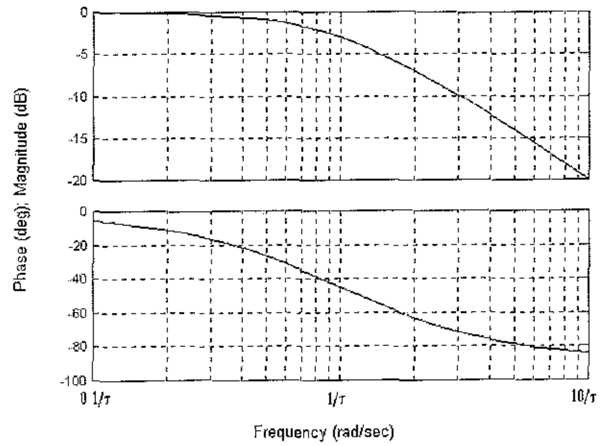

Figure 7 Normalised first-order frequency response curves

Again the mathematical expressions derived from formally solving the differential equation for a sinusoidal forcing function are far less important than being able to interpret the response curves themselves. As with the step response curve, an experienced engineer can read out information from the frequency response graphs about behaviour at low, mid-range and high frequencies. Depending on the system, such information is interpreted in terms of speed of response, signal distortion, interference rejection, and so on.

The language of the engineer in dealing with these representation is no longer the language of the mathematician, but a different way of talking using expressions such as 'rise time', 'settling time', 'cut-off frequency', 'low-frequency gain' and 'high-frequency roll-off' to interpret the curves from an engineering standpoint. What the differential equation models lead to, therefore, are not simply solutions to particular forcing functions but to interpretations which give insight into the behaviour of the system for a wide range of situations. The step response and frequency response curves are not just limited solutions but rich patterns of signification carrying meaning for the engineer.

Visualising behaviour – the s-plane

While the time domain (step response) and the frequency domain (sinusoidal response) can be treated separately, we can take a further step and bring all this information together in one particularly rich representation. In this approach, the differential equations are transformed by using simple look-up tables based on Laplace transform theory into equivalent representations relating the input and outputs of the system.

Traditionally, a reason often given for teaching engineering students Laplace transform theory is that it is used to solve higher-order differential equations (see, for example, Dorf, 1989; Kuo, 1987) Yet, as we have seen, the engineering emphasis is not on solving the equations at all, but in gaining insight about the system What the Laplace approach offers is a way of representing pictorially the dynamic characteristics of complex linear systems which, with a little practice, can be interpreted in terms of behaviour in both the time domain and the frequency domain

Assuming quiescent initial conditions and applying Laplace transform ideas to the first-order model results in:

$$s\tau Y(s) + Y(s) = kX(s)$$

from which the system output is related to the input as:

$$Y(s) = \frac{k}{s\tau + 1} X(s)$$

giving a transfer function model:

$$\frac{Y(s)}{X(s)} = \frac{k}{s\tau + 1}$$

In this representation, s is a complex variable with real and imaginary parts. Now, the point of expressing the models like this is that details about the inputs and outputs of the systems (which will be changing from moment to moment) are separated from the fixed characteristics of the systems themselves And it is the system information on the right-hand side of these expressions that engineers learn to interpret using simple pictorial representations.

For example, suppose again that the value of the time constant τ is 0 1 second If we take the value of k to be 1, the system can be characterised by the expression:

$$\frac{1}{s0.1 + 1} = \frac{10}{s + 10}$$

Since s is a complex variable, we can represent this on an Argand diagram, or *s-plane*, as in Figure 8. In this diagram, the horizontal axis represents the real part of s and the vertical axis represents the imaginary part of s The cross on the diagram is at the value of s, -10 in this case, at which the value of the denominator of the expression is zero, and hence the value of expression as a whole is infinitely large This feature is called a 'pole'.

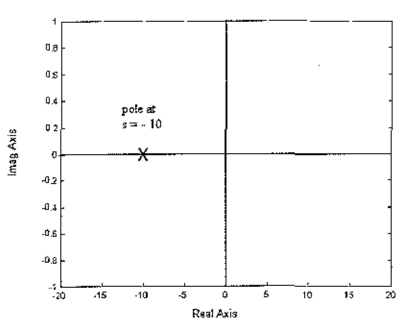

Figure 8 s-plane plot of the function 10/(s + 10)

The position of the pole, or poles, of a system model can be interpreted to provide information about system behaviour or possibilities for design. If the pole lies further to the left along the real axis, for example, this corresponds to a system with a smaller time constant and hence a faster system response. A pole lying closer to the origin of the s-plane, on the other hand, corresponds to a larger time constant and a more sluggish response.

If the model represented a control system, for example, then system changes leading to a shift of the pole to the right would imply that the system may not be able to respond quickly enough to its inputs to perform its function correctly If changes led to the pole moving beyond the origin into the right-hand side of the s-plane, then this would indicate to the designer that the system will be unstable and unusable

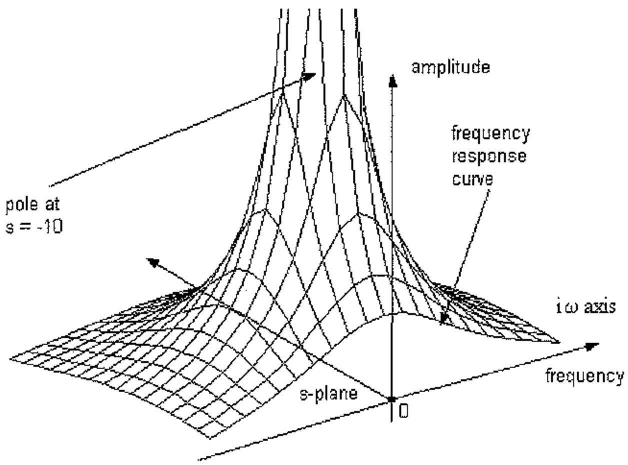

Figure 9 3-D surface for the function 10/(s + 10)

Frequency response information can be also obtained from this model by visualising the magnitude of the transfer function as the height of a surface above the s-plane Figure 9 shows such a three-dimensional plot for the first-order model Notice that the surface surges upwards towards infinity above the pole at $s = -10$: indeed, the pole can be thought of as holding up the surface at this point.

The 3-D surface is cut along the axis of the s-plane, where $s = i\omega$. If s is replaced by $i\omega$ in the transfer function and then the modulus of the resulting function evaluated, this corresponds to the magnitude of the frequency response function found from the original differential equation

In Figure 9, this modulus function is represented by the height of the 'cut' in the 3-D surface along the $i\omega$-axis from $\omega = 0$ to $\omega = \infty$. (The amplitude frequency response curves were drawn on logarithmic axes, so the shapes visualised from such 3-D plots will not correspond exactly.)

What is important to engineers is not the properties of the complex plane from the point of view of mathematical complex analysis but interpreting system behaviour by understanding how the positions of the poles and zeros (zeros indicate the values of s where the numerator goes to zero in more complex models) represent the time- and frequency-domain behaviour of a system With experience, systems can often be modelled directly using transforms without first constructing a differential equation.

From a design point of view, poles at different positions give different overall systems characteristics. Pole-zero patterns can be related closely to important aspects of system behaviour, such as the speed at which a control system can move the arm of an industrial robot, the magnitude and frequency of oscillations likely to occur in a ship stabilisation system or the range of frequencies that can be handled without distortion by a hi-fi amplifier.

Now, the mathematical treatment of poles and zeros within the specialist area of complex analysis is highly abstract and complex indeed. In contrast, though, many of the manipulations carried out using the engineering models are remarkably simple and 'non-mathematical'. Such models have led to engineers talking a language very different from that of conventional mathematics: a remarkably concrete language. For example, system designers describe certain systems as 'possessing' so many poles; they talk about the need to 'place' poles in certain regions or 'move' them to a more favourable position; and they become highly adept at visualising the pattern of poles and zeros in the complex plane in terms of how they want the system to behave.

A different way of talking

This is an example of electronics engineers 'doing mathematics'. The important point is that the language they use is *not* the traditional language of mathematics, even if the manipulation of their models may be completely analogous to other manipulations of other, more conventionally 'mathematical' models. Moreover, this linguistic shift is more than just jargon, and more than just a handy way of coping with the mathematics; the shift indicates a way of thinking about systems behaviour in which the features of the models are deeply linked to the systems they are describing. So, in the case of our electronic engineer, the poles effectively cease to be just convenient visualisations of mathematical complex variable theory and become system features which are just as real as the electronic components from which the system has been built.

Explanations of physical behaviour also shift in the same direction: at a systems level, accounts of how and why a system behaves as it does are often couched in terms not of physical variables such as voltage and current, but in terms of where the poles of the model lie. In this different ontology, the system poles are not simply part of a convenient mathematical model, they are what *cause* the system to behave as it does, as if they had the tangible existence of the physical components and measurable signals present in a system. From an engineering viewpoint, then, this language is a powerful way not only of representing important aspects of system's behaviour, but also of explaining how that behaviour comes about.

So although a great deal of importance has been placed on formal mathematics in an educational context to ensure that engineers 'understand the fundamentals' of what they are doing, formal mathematics is often not what engineers use. Over the years, practising engineers have developed a range of ingenious tools, techniques and insights to move them away from having to deal with the mathematics directly. While the language of mathematics focuses on 'solving equations', engineering techniques often focus on interpretative understanding of graphical representations of the same information.

When engineers carry out this type of modelling, what they actually do is a far cry from the 'applied mathematics' they tend to have been taught - with or without the 'modelling cycle'. They begin to talk a very different language from that of mathematics, and the notations and patterns that they manipulate bear very little resemblance to the symbols and equations of courses in engineering mathematics.

Although seldom taught explicitly, this linguistic skill is part of the tradecraft of engineering. It is an integral part of what counts as doing engineering and requires a quite subtle blend of knowledge about how built systems actually behave and how that behaviour can most appropriately be represented and analysed. To achieve that blend, practising engineers have, over the years, put a great deal of ingenuity into inventing graphical and pictorial techniques which avoid much of the 'essential and fundamental' material deemed so important by many academic 'engineers'.

Although originally evolved to provide design tools for the engineer equipped with little more than paper, pencil, a slide rule and a set of mathematical tables, these techniques take on a new lease of life as computer-generated graphics. Well-designed computer-based tools using some of the traditional engineering models and representations can contribute to a clearer understanding of systems behaviour. They can enhance expertise by allowing the engineer to use the processing power of the computer to explore easily and quickly the predicted effect on system behaviour of different design strategies. And this exploration is carried out directly in the language of the engineer, rather than in that of the mathematician.

Mathematics in engineering education: a crisis of confidence?

We can draw some of this together as a tentative conclusion. The way mathematics is used in engineering is not simply a straightforward application of mathematical techniques. There are a number of factors here. First, the aims and purposes of engineers are not those of mathematicians. There is a focus on explanation and design, in contrast to mathematical structure and rigour.

Second, traditions of explanation are different in different domains: what counts as an acceptable mathematical argument is unlikely to count as an acceptable engineering argument. Third, different communities of practice lead to different ways of talking and doing, even when they are apparently dealing with the 'same thing'. Tacit skills learnt by experience in engineering may not integrate well with the formal skills laid down in mathematics courses.

But if this is the case, then why persist in an educational approach that is apparently disconnected from practice? One reason arises from seeing the operation of communities of professional practice in terms of trust (Porter, 1996). Engineers, scientists, doctors - professional groupings - need to establish a tradition of trust within wider society - if only because technical matters cannot be resolved elsewhere. Traditions of trust can usefully refer to privileged areas for support.

Appeals to mathematical reasoning, scientific method, statistical analysis, for example, can add weight to a community's claim that they are the best people to make the relevant decisions in their area. Thus, from the outside looking in, engineering develops its trustworthy *persona* by appealing to certain formalities that are recognised and respected in wider society.

On the inside, however, the story is quite different. Here, the meanings relevant to the internal working of the community are forged and negotiated. Non-formal methods and techniques contribute to a meta-narrative - an insider language - that is used and accepted between peers but which never appears in the journals, proceedings or textbooks that form the written - and therefore public - record of the community.

Finally, what of the relationship between the community of engineering educators and the community of practising engineers? If the importance of formal mathematical knowledge is a feature of the former but is redefined and renegotiated by the latter, what does this say about the security and self-confidence of each community? Porter suggests that, in relatively secure fields, the trappings of formality between practitioners are often missing. Within a framework of negotiated and approved understanding, there is no need for the rigour necessary for dealing with outsiders.

However, in less secure areas, in communities whose boundaries are changing or whose members feel vulnerable to external criticism, there is an emphasis on formality that is a direct result of perceived external pressures. What lessons lurk here? In the informality of its meta-narratives, do we see the community of engineering practice as secure and well-established while the community of engineering education, with its mathematical focus, appears by contrast to be wary of criticism, unsure of its intellectual status and unclear of its role?

Notes

[1] This version is taken from OU (1985), but very similar versions can be found throughout the literature on mathematical modelling

[2] Or there are the less scientific 'laws' of economics or other social sciences, in which case the variables might be money supply, inflation rate, GDP, etc

[3] The 'lag' here is equivalent to a first-order differential equation, discussed in more detail below.

[4] This might replace the 'first-order lag' by a second-order linear differential equation model.

[5] This is a case study used in the development of Open University teaching material on control engineering

[6] This is the 'first-order lag' model described earlier.

References

Croft, A. C., Hibberd, S., Lawson, D. A., Mustoe, L. R. and Shaw, I. (2000) 'Engineering mathematics in the UK: SARTOR - a timely opportunity for reform', *IEE Engineering Science and Education Journal* 9(2), 77-87.

Dorf, R. C. (1989, 5th edn) *Modern Control Systems*, Reading, MA, Addison-Wesley.

IMA (1995) *Mathematics Matters in Education*, Southend-on-Sea, Essex, Institute of Mathematics and its Applications.

Kuo, B. C. (1987, 5th edn) *Automatic Control Systems*, London, Prentice-Hall

Morgan, M. S. and Morrison, M. (1999) *Models as Mediators*, Cambridge, Cambridge University Press

OU (1985) TM282 *Modelling with Mathematics*, Milton Keynes, Bucks, The Open University

OU (1997) T293 *Communicating Technology*, Milton Keynes, Bucks, The Open University

Porter, T. M. (1996) *Trust in Numbers*, Princeton, NJ, Princeton University Press

Umez-Eronini, E. (1999) *System Dynamics and Control*, London, PWS Publishing (ITP)