Empirical Modeling Methodologies for Construction

Ian Flood, M.ASCE[1]; and Raja R. A. Issa, M.ASCE[2]

Abstract: The paper provides a review of empirical modeling and its application within construction engineering and management. The scope of application and trends in use of this approach are first assessed, and the potential for its further development is identified. This is followed by an examination of the key components of empirical modeling, namely: the structure and operation of the model and the scheme used in its development. The paper then provides a rigorous methodology that must be followed to ensure the validity and value of the end model, covering the steps: strategizing; data collation and assessment; model development; model evaluation and final selection; final validation; and implementation. The methodology is designed to cater for all forms of empirical modeling including the procedurally more demanding development algorithms that have become available in recent years, such as simulated evolution. Overall, the paper is designed to provide researchers embarking on an empirical modeling study with an overview of when it is appropriate to use this approach, what type of system to adopt, and how to ensure development of a successful end product.

DOI: 10.1061/(ASCE)CO.1943-7862.0000138

CE Database subject headings: Empirical equations; Dynamic models; Algorithms; Data collection; Data analysis; Validation; Construction management.

Author keywords: Empirical modeling; Structured model; Static model; Dynamic model; Development algorithm; Data collection; Data analysis; Model validation.

Introduction

Mathematical models are abstractions of systems described using the language of any branch of mathematics including, for example, algebra, statistics, logic, and algorithms. They are developed to serve either as experimental tools that can be used to extend our understanding of a system or as predictive tools to assist in tasks such as decision-making and automated systems control. Certainly they have become significantly more powerful and accessible with the advent of desktop computing, finding important applications in almost all branches of science and technology. Mathematical models can be classified in many ways, though a common dichotomy relevant to this paper is that of empirically versus theoretically derived models. An empirical model is one developed from observations of the response of the system under investigation (or of an analog of that system) for a range of situations. In contrast, a theoretical model is one developed from what are considered to be the fundamental laws or principles that govern the response of the system. From these definitions, it is often reasoned that empirical models provide compromised solutions to problems; they are relatively easy to develop but are *black box* devices providing no explanation of their output, and have no ability to extrapolate beyond the set of observations used in their development.

The above characterization of empirical and theoretical modeling, while widely accepted, is an oversimplification of the truth. Ultimately, all mathematical models (theoretical or otherwise) are derived empirically and, contrary to popular belief, empirical models can play a key role in developing a deeper and more generalized understanding of a system. To illustrate these points, consider the problem of modeling the dissipation of heat in a large concrete structure during curing. At the heart of this model would most likely be the following heat equation describing how the temperature of a solid material changes with time

$$\partial T/\partial t = k \cdot (\partial^2 T/\partial x^2 + \partial^2 T/\partial y^2 + \partial^2 T/\partial z^2) \tag{1}$$

where T=temperature; t=time; k=thermal diffusivity of the concrete; and x, y, and z=spatial dimensions.

This would be considered to be the underlying theory from which the heat dissipation model would be built; yet it can be shown that ultimately every aspect of this model will have been derived using empirical techniques. First, the broader form of the equation (which states that the rate of change of temperature over time is directly proportional to the second derivative of the temperature distribution across space) must have been inferred from experience of how heated objects behave. The use of experience as such to develop a model is an empirical procedure, requiring reference to observations of the response of the system both for inspiration in the design of the model, and for evolving that design into a more accurate or complete definition of reality. Indeed, developing a set of general principles about a system (a theory) based on experience or observation is a branch of artificial intelligence (AI) that has significant potential for application in construction engineering and management (CEM), and is a concept that will be returned to later.

Next, the parameter k in the equation (the thermal diffusivity of the concrete) is a constant that will have been determined

[1]Associate Professor, Rinker School, College of Design, Construction and Planning, Univ. of Florida, Gainesville, FL 32611-5703 (corresponding author). E-mail: flood@ufl.edu

[2]Professor, Rinker School, College of Design, Construction and Planning, Univ. of Florida, Gainesville, FL 32611-5703. E-mail: raymond-iss@ufl.edu

Note. This manuscript was submitted on August 28, 2008; approved on March 27, 2009; published online on August 20, 2009. Discussion period open until June 1, 2010; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Construction Engineering and Management***, Vol. 136, No. 1, January 1, 2010. ©ASCE, ISSN 0733-9364/2010/1-36–48/$25.00.

directly from laboratory experiments or derived from other material constants (the thermal conductivity, specific heat capacity, and density of the concrete) that will themselves have been determined from laboratory experiments. A constant derived from laboratory experimentation is by definition an empirical model, albeit the simplest form possible.

Finally, the complete model representing heat dissipation within the concrete structure would require Eq. (1) to be discretized in time and space, most likely using the finite-difference method. The discretized form of this equation would be set within a spatial grid defining the shape and size of the concrete structure, which would then be used to simulate the heat dissipation process. Importantly, the accuracy of the model's predictions will depend on, among other things, the distance between the discrete elements within the grid. Appropriate distances would usually be determined by trial-and-error (again an empirical modeling technique) in which the proximity of the discrete elements would be varied until the behavior of the model is sufficiently close to that observed in the real system. Alternatively, the distances could be based on rules-of-thumb; nevertheless, these rules-of-thumb would have been derived at some earlier point in time by experimenting with similar models.

Clearly empirical modeling has a broad potential, being at the heart of just about all forms of mathematical modeling. The objective of this paper is to identify the extent to which this potential has been realized within the field of CEM and to identify the most promising areas for future development and application. This includes an evaluation of the application of the technique as reported within the ASCE Journal of Construction Engineering and Management (JCEM) and related journals, classifying the methods that have been considered, and identifying trends in their adoption and usage. Despite their diversity, the development of any empirical model must follow the same basic procedures to ensure the production of a valid and useful end product. Moreover, recent advances in model development techniques have made this procedure more tortuous. This paper, therefore, provides a comprehensive review of the empirical modeling methodology that recognizes the extended demands of these new tools.

Application in CEM

Empirical modeling is clearly one of the most widely used analytical tools in CEM. For example, in the 17-year-period from 1991 to 2007, a total of 97 articles have been published in JCEM that consider the two most widely used forms of empirical model: regression analysis, and artificial neural networks (this was based on a search for articles with the terms "neural" and/or "regression" within their titles, keywords, and abstracts). This is comparable in popularity to simulation modeling, for example, which is considered in 99 JCEM articles according to a similar search using the terms "simulation" AND "model." Moreover, the last couple of decades have seen an increasing trend in the use of these most commonly used forms of empirical modeling, as is evident from the distribution in Fig. 1. Indeed, the number of articles that consider regression analysis and/or artificial neural networks has increased from an average of 1.8/year for the first five years covered by the chart to 9.8/year for the past 5 years, a 5.4 factor increase. This can be explained in part by an increase in the number of articles published each year in JCEM, although this has only doubled indicating that the proportion of published articles dedicated to empirical modeling has also increased.

The applications of empirical modeling reported in JCEM

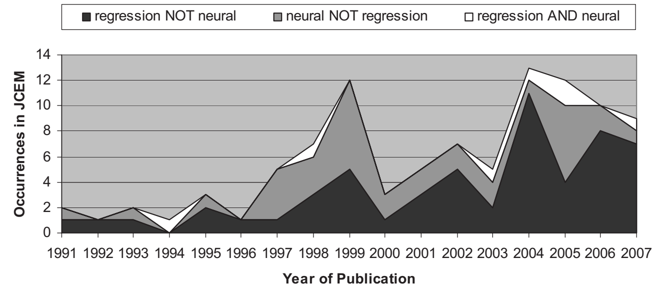


Fig. 1. Number of articles published in JCEM that consider the most popular types of empirical model

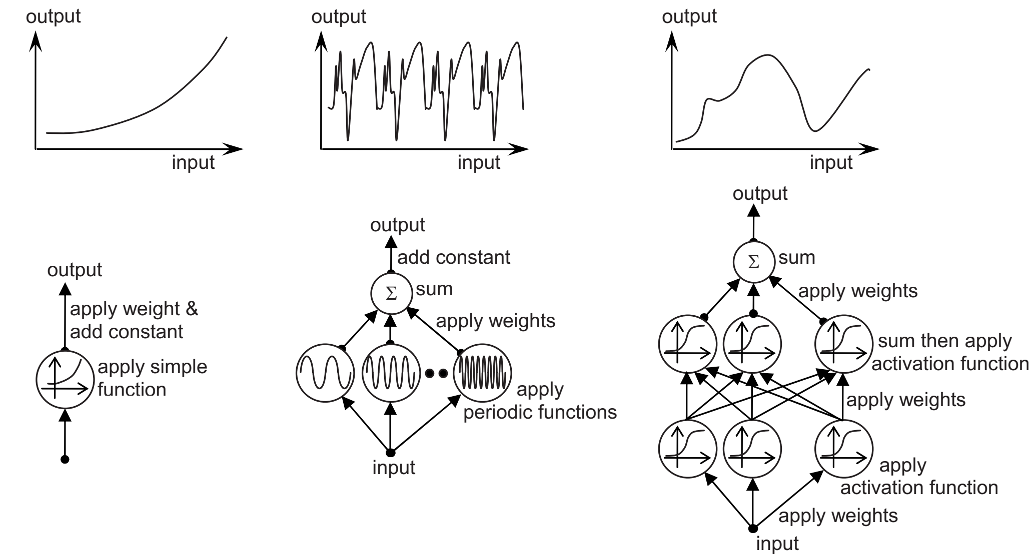
cover almost every aspect of CEM. A small yet diverse sample of some more recent applications includes: forecasting labor demand (Wong et al. 2008); predicting the performance of project managers (Ahadzie et al. 2008); predicting profit performance of projects (Han et al. 2007); studying dispute negotiation styles (Cheung et al. 2006); predicting construction costs (Lowe et al. 2006); estimating productivity (Ezeldin and Sharara 2006); assessing project safety (Fang et al. 2006); estimating operation durations (Marzouk and Moselhi 2004); and assessing risk in bidding for projects (Fang et al. 2004).

Although there are many examples of the application of empirical modeling to CEM, the majority of these consider very simple modeling systems that develop continuous functions that map directly from a vector of inputs to a vector of outputs. The most common and simplest of these is regression analysis, which provides either a linear or simple nonlinear mapping between the inputs and outputs, as illustrated in Fig. 2(a). More sophisticated varieties of direct-mapping models include: Fourier analysis which produces a compound of periodic functions that map from the inputs to the outputs, as illustrated in Fig. 2(b); and Feed forward Artificial Neural Networks which develop multiple layers of compound functions, of any form, to map from the inputs to the outputs, as illustrated in Fig. 2(c). Example applications of these various forms of direct-mapping model in CEM include: early-design cost estimation using multivariate regression analysis (Stoy et al. 2008); determining truck loads from bridge strain readings using Fast Fourier Transforms (Gagarine 1991); and prediction of project success using artificial neural networks (Ko and Cheng 2007). Beyond this, there has been a growing interest in simulated evolutionary methods, such as genetic algorithms, for model development [see for example Salem et al. (2007)], although the potential of this method has not been fully tapped having only been used to develop relatively simple model structures. Another area that has attracted some interest is the development of empirical models that operate dynamically, providing insight into the time dependent behavior of a system. Examples include simulating construction excavation processes (Flood and Christophilos 1996), and modeling dynamic heat-flow through buildings (Flood et al. 2004).

Modeling Systems

Empirical methods can be used to develop models that are far more sophisticated than the direct-mapping devices discussed above, significantly extending the scope of application of the technique, although to date this potential remains largely unexplored. In fact, just about any type of mathematical model can be developed empirically. While a complete discussion of all model

J. Constr. Eng. Manage., 2010, 136(1): 36-48



(a) Regression Model (b) Fourier Series (c) Artificial Neural Network

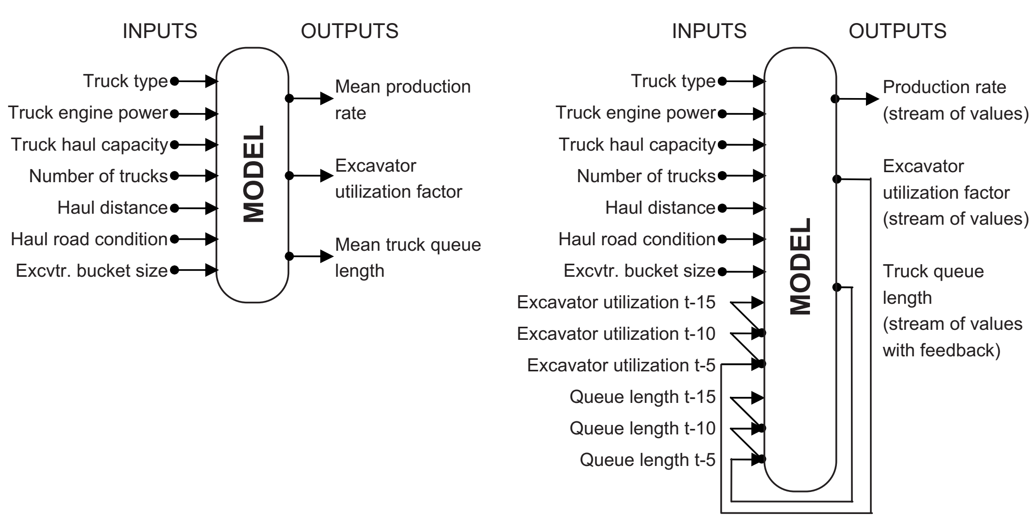
Fig. 2. Examples of direct mapping models (a) regression model; (b) Fourier series; and (c) artificial neural network

types is beyond the scope of this paper, there are two broad issues that must always be considered in an empirical study and thus warrant discussion. These are: (1) the structure of the model (which can be further broken down into the interface and internal structure); and (2) the scheme used to develop the model. These broad issues, and their relevance to the success of an empirical modeling study, are discussed in the following subsections.

Interface Structure

A model's interface comprises input variables which are used to define the specific instance of the problem being investigated and output variables which register the models response to the values presented at the inputs. Depending on the modeling system being used, these variables may be determined by default (such as is the case for many simulation modeling systems such as CYCLONE (Halpin and Woodhead 1976) or they may have to be designed by the model developer. Careful consideration should be given to the design of these variables since they will have a profound impact on both the value and accuracy of the model.

A primary decision to be made when designing the structure of the model's interface is whether it will run statically (generating a single set of outputs in response to a set of inputs) or dynamically (producing a stream of values at the outputs). Almost all empirical modeling studies adopt the static approach, due to its simplicity. However, dynamic models provide additional information on the behavior of a system, and greater modeling flexibility. The issues relating to this choice will be illustrated by reference to Fig. 3 which shows the interfaces for a static and dynamic version of a model of an excavation system comprising one excavator loading a fleet of dump-trucks. The static version of this model [Fig. 3(a)] provides a single estimate of the production rate of the system, averaged over an indefinite period of time, in response to a set of



(a) Static Model (b) Dynamic Model

Fig. 3. Interface structure for alternative models of an excavator and truck based excavation systems (a) static model; (b) dynamic model

J. Constr. Eng. Manage., 2010, 136(1): 36-48

inputs ranging from "*truck type*" to "*excavator bucket size*." Two other outputs are provided in this model, "*excavator utilization factor*" and "*mean truck queue length*" that would help the user determine whether the excavator performance and number of trucks in the system are balanced. The dynamic version of the model [Fig. 3(b)] runs iteratively, providing a sequence of estimates of production rate at constant increments in simulated time (perhaps every minute). Most of the input variables are the same as the static version of this model, except that recent truck queue lengths and excavator utilization factors are fed back as input to the model at each cycle, given that these will correlate with the future production rate. The advantages of the dynamic model are that: (1) it is possible to see changes in the performance of the excavation system over time; and (2) changes to the specification of the excavation system (such as the number of operational trucks) can be made during the simulation to see how this impacts performance. On the downside, development of the dynamic model will require more observations of the performance of the system, and these observations will have to cover more variables. An example of dynamic modeling of excavation system using empirical modeling techniques is provided by Flood and Christophilos (1996).

The next point to consider in the structuring of the model's interface is the selection of the input variables. Obviously, the only input variables that should be included are those that are significant in terms of either affecting the values to be generated at the output or being correlated with them in some way. However, the significant variables are often not known at the outset of an empirical modeling study. Determining an appropriate set of input variables requires: (1) expert judgment by the model developer, including a review of other work; and (2) experimentation in which the performance of the model is compared for alternative sets of inputs. Consider, for example, the variables *truck type*, "*truck engine power*," and "*truck haul capacity*" in the excavation models of Fig. 3. It is likely that *truck type* implicitly defines both the "*engine power*" and the "*haul capacity*" of the trucks, resulting in some redundancy in this set of input variables. In this case, a choice will have to be made between these alternative variables. Using *truck type* has the advantage that it may be associated with other relevant information about a truck beyond its *engine power* and *haul capacity* (such as the weight of the truck and thus its traction) potentially increasing the accuracy of the model. This would have to be confirmed in a comparative study, developing the model with the alternative sets of inputs and measuring which version produces the most accurate output. A second advantage of using *truck type* instead of "*truck engine power*," and "*truck haul capacity*" is that there will be one less input variable in the model which will, in turn, reduce the number of observations required to develop the model. On the other hand, the use of *truck engine power* and *truck haul capacity* as the inputs will provide greater modeling versatility, allowing for any combination of values for these two variables.

Another point to consider in the development of a model's interface is the type of value each input and output variable will assume. Referring to Fig. 3(b), variables such as "*mean production rate*" and "*haul distance*" assume real-numbered values, "*truck queue length*" and "*number of trucks*" assume positive integers, while "*haul road condition*" is an enumerated type which may assume values such as "*poor*," "*satisfactory*," and "*good*." Other systems of modeling may use radically different ways of encoding information. Pulse frequency coding, for example, represents values by the number of pulses generated in a given period of time [see for example, Dayhoff (1991)], while

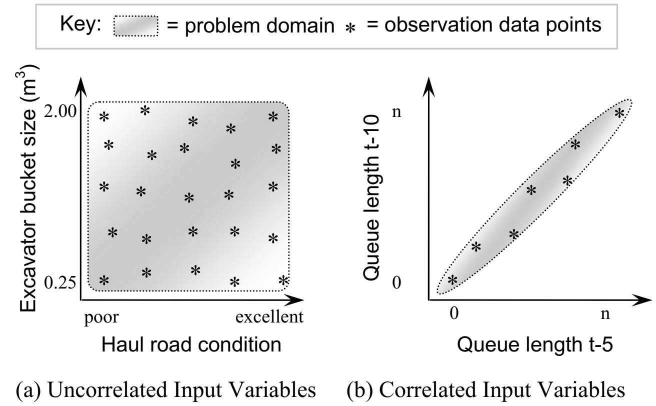


Fig. 4. Affect of correlation between input variables on the observations required to develop a model (a) uncorrelated input variables; (b) correlated input variables

Markov models may generate values stochastically as random events in time (Zhang 2006). Whatever variable types are used, care must be taken to ensure that they are compatible with the type of model being used. Consider, for example, the input variable *truck type* in Fig. 3(a) which may assume values such as "*type 1*," "*type 2*," and so on. It is an enumerated type, but unlike the other variables in the model, its values cannot necessarily be placed in a progressive order that has some functional relationship with the output variables. If a meaningful order can be identified (perhaps based on the haul capacity of the trucks in this case) then this should be adopted; certainly an arbitrary order of values should always be avoided. The inclusion of unordered variables in a model is usually problematic, particularly if the mapping from inputs to outputs is a continuous function (such as is the case for regression models and many artificial neural networks), since it will introduce discontinuities to the problem that are difficult to model. Other examples of unordered variables commonly misused in this way include "*type of construction project*" (assuming values such as "*residential*," "*commercial*," and "*industrial*"), "*country*," and "*project manager*." A good way to handle unordered variables is to develop a separate model for each value of the variable—for the excavation model, this would require developing a separate model for each type of truck.

There are a couple of practical issues that should also be considered when developing the interface for an empirically derived model. First, the number of observations required to develop a model tends to increase geometrically with the number of input variables, and so there is a strong incentive to include as few inputs as possible. Fortunately, this is not true for input variables that are correlated such as "*queue length t*-5′," "*queue length t*-10′," and "*queue length t*-15′" in the dynamic version of the excavation system shown in Fig. 3(b). The reason for this can be understood from Fig. 4 which shows the distribution of problems that can occur for two uncorrelated input variables [Fig. 4(a)] and two correlated input variables [Fig. 4(b)]. Clearly, the correlated variables have a problem range that is highly constrained and thus can be represented by far fewer observations.

The second practical issue affecting the development of a model's interface stems from the quality and availability of the observed data used to develop the model. It could be, for example, that the observed data set does not provide a representative distribution of values for a variable (in terms of numbers and distribution of data points across the range of a problem), or it may not be possible to measure a variable accurately. Conse-

J. Constr. Eng. Manage., 2010, 136(1): 36-48

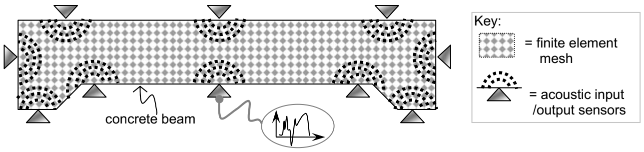


Fig. 5. Finite-difference model of the acoustic response of a concrete member used to identify the location and size of reinforcing steel

quently, including that variable in the study may provide little extra benefit or, in some circumstances, it could impede the model's ability to develop an accurate representation of the problem.

Internal Structure

Some models have a rich internal structure (such as discrete-event simulation models) while others may be very simple in form (such as regression models). This internal structure may be defined implicitly by the type of model being used, it may be derived automatically by a model development algorithm, or it may be crafted by the model developer. Where a developer has some choice or control over the internal structure, this should be done with care to ensure the accuracy and value of the resultant model. These points will be discussed in detail in the following paragraphs.

The objective of most empirical modeling studies is to develop a device that can make accurate estimates or predictions about some aspect of a system—the focus is on the output generated by the model. However, a potentially powerful yet under exploited application of empirical modeling focuses on the internal structures that a model forms when developed to solve a given problem. If the model is configured carefully, these structures can tell us something important about the physical structure of the system being modeled or provide a set of rules or principles that can be used to solve related problems. This type of application of empirical modeling can be classified as a branch of AI, providing an automated approach to inductive reasoning.

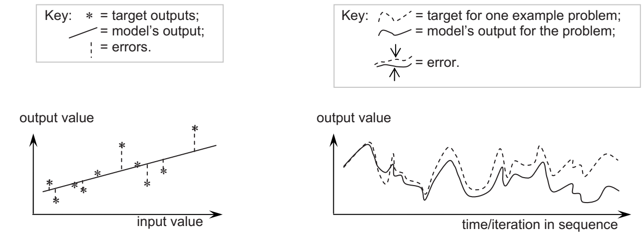
To illustrate this concept, consider the problem of detecting the location of steel reinforcement in a concrete member from its response to acoustic signals generated at discrete points on its surface. A finite-difference based mesh could be set-up that simulates the acoustic behavior of the structure as if it is composed entirely of concrete, as shown in Fig. 5. This model could then be adapted by substituting elements representing concrete (indicated by the gray diamonds) with those representing steel or boundaries between steel and concrete, or by adjusting the values of the coefficients, until it is able to replicate the acoustic response observed in the actual concrete structure. This adaptation process may be achieved using, for example, genetic algorithm techniques [such as adopted by Salem et al. (2007)]. A critical decision for the modeler in this example would be to select a mesh density for the model that would be able to identify the location and size of the reinforcing steel to the required precision.

In some cases, the basic internal structure of a model may be crafted by the model developer. The end product may take the form of a set of empirically developed modules that can be pieced together as required to represent each new version of the problem. Zhu et al. (1999) adopted this approach for static modeling of the flow of commercial traffic through inland waterways. The system comprises a menu of neural network modules, each trained to estimate traffic-flow through a type of component in a waterway system, such as a lock or stretch of canal. The inputs to each module are used to define the specific characteristics of the section of waterway represented by that component, and the modules are assembled to provide a model capable of estimating traffic-flow through a complete waterway system. The approach has also been applied to dynamic modeling by Flood et al. (2004), in this case to model transient heat-flow in buildings. The primary advantage of the modular approach (compared to using a monolithic model to solve the problem) is an ability to model a wider range of problems given that there is indefinite number of ways in which the modules can be assembled. It is also extensible, allowing an increase in the functionality of the modeling system through the addition of new modules.

Development Schemes

Determining a complete definition for an empirical model will usually involve the expert judgment of a model developer complemented by the application of a model development algorithm. An artificial neural network, for example, may require the model developer to decide on the number and connectivity of the nodes in the network and the type of activation function used at each node, then an error-gradient descent algorithm (such as backpropagation) may be employed to determine the weights on the connections. In the extreme case, development of the model will be done entirely by the model developer without the use of a development algorithm. An example of this would be the development of a simulation model in which the observed data are used by the model developer to direct the design of the model and to validate its performance. At the other extreme, the development algorithm may perform all the model development work. For example, in a regression study, the development algorithm may determine both the type of function to use (based on some statistical test such as the goodness-of fit) as well as the coefficients of the function. Typically, the model developer has been responsible for performing tasks such as selecting the basic type of model and determining its structure, while the development algorithm has been responsible for deriving the model's coefficients. However, recent advances in computer-based search techniques have made automated development of models with complicated structures a viable option. Simulated evolutionary methods, for example, are sufficiently flexible to allow development of all aspects of just about any type of mathematical model. The problems for the model developer then become: determining the right type of algorithm to adopt for a given problem; and coding the problem in a way that facilitates application of these techniques.

Most model development algorithms operate iteratively, advancing the model in discrete steps toward a design that is able to replicate, to within an acceptable tolerance, the response of the system under investigation. Progress in model development is measured and directed by an objective function, which measures error at the outputs from the model for a comprehensive set of example problems. Often, this measure is the sum of the square of the differences between the models actual output and the required output for the examples. The objective is to develop the model to minimize this error. For static models, the objective function is based on the errors the model generates for a set of independent observations of the problem, each mapping from an input vector to a corresponding output vector, as illustrated in Fig. 6(a). For dynamic models, the issue is more complicated than this, since the model will generate a stream of vectors at the outputs in response to each input vector. The concern is that when the model is running dynamically, it will be feeding information back (either at its interface or internally) thus possibly leading to a compound-



(a) errors for multiple problems in a static model (b) error for 1 example problem in a dynamic model

Fig. 6. Comparison of errors for static and dynamic models (a) errors for multiple problems in a static model; (b) error for 1 example problem in a dynamic model

ing of errors. This is illustrated in Fig. 6(b) which shows that the first output response for a dynamic model can have a small error, but with subsequent iterations, the error can start to increase. For dynamic models, the error for each example of a problem should be integrated across the full sequence of outputs from the model, not just be based on the first output in the sequence. However, operating an objective function in this way will require a lot of computation, requiring a complete run of the model for each example problem to be tested; there may be many such examples to be tested in this way and, moreover, this process will be nested inside the model development algorithm which itself will require many iterations to fully develop a model. Consequently, it is often not practicable to use full sequences of outputs from dynamic models to measure error during the development stage (the first outputs generated in the sequence may have to be used instead); however, the final validation of the model should always be based on the errors measured over the complete sequence of outputs for each example problem.

Empirical modeling offers many advantages over other modeling techniques. Based on the experiences of the writers, there are three main classes of problem to which the empirical approach is best suited, namely: those that are poorly understood; those that are overstated; and those that require results to be produced quickly. An overview of these classes is provided in Table 1, along with an explanation of the reasons why empirical modeling is successful in each of these cases, an illustration of potential applications within construction, and an identification of the corresponding limitations of the approach. As noted in the table, the main constraint on empirical modeling is the fact that the number of examples of a problem needed to develop a model tends to increase geometrically relative to the number of input variables in the problem. This often places a practical limit on the complexity of the problem that can be considered. Fortunately, this limitation does not hold for problems where the values of the input variables are strongly correlated with each other (in this case, the number of examples required may only increase linearly). Moreover, as discussed later, it may be possible to circumvent this limitation by developing models with more complex internal structure.

Empirical Modeling Methodology

Development and implementation of an empirical model must follow a rigorous set of procedures to ensure the validity and value of the end product. These procedures can be divided into the following six steps.

Step 1: Strategizing

The first step in model development is to establish as much familiarity as possible with the problem at hand, including:

1. Establishing the application objectives of the model—addressing issues such as what output variables are required, whether the output should be static or dynamic in form, whether the focus of the study is on the internal structures developed by the model or on the output values it generates, and the level of accuracy required.
2. Determining the input variables likely to be significant.
3. Gaining a feel for how the system being modeled responds to different conditions, such as, whether it is linear versus nonlinear, or stochastic versus deterministic.
4. Determining the availability of data for developing the model and the sources for additional data.

The questions that should be answered at this stage are:

1. What type and structure of model is most appropriate to solve the problem?
2. What development algorithm will help derive the most effective model?
3. What is the objective function?
4. What new studies will likely be required to acquire the necessary data for development and validation of the model?

In addition, the feasibility of the study in terms of its time, cost, and availability of resources should be assessed. It is often the case that several of the above questions cannot be answered fully and with confidence until after some development and analysis work has been completed. If this is critical then it may be necessary to conduct a pilot study developing a less advanced version of the model to help answer these strategic questions more thoroughly.

One of the most insightful tasks that can be performed at the start of a modeling study is to gain a graphical understanding of the problem using, if available, existing observations of the response of the system or of a similar system. There are two basic types of graph that should be considered here, the first plotting each output variable against each of the input variables, and the second plotting each of the input variables against each other. Fig. 7 shows examples of the first type of plot, demonstrating:

1. The relevance of the input variables—for example, the highly random distribution of the points for input Variable 1 in the figure indicates that it has little relevance to the problem and thus might be excluded from the study.
2. The complexity of the response of the system in terms of whether it is linear or nonlinear and, in the latter case, the form of the nonlinearity—for example, input Variable 2 in the figure appears to have a nonlinear relationship with the output variable.
3. The existence of unexplained variance in the response of the system, possibly due to noise or missing input variables that are significant—for example, input Variable 2 does not appear to fully explain the response of the system.

An example of the second type of graph (plotting input variables against each other) is provided earlier in Fig. 4. This type of plot is useful for determining correlation between the input variables, which would suggest possible redundancy between those inputs or at east a reduction in the distribution of the observations required to represent those variables.

The importance of developing an understanding of the response of the system to selecting an appropriate model type and structure can be demonstrated by the simple example problem shown in Fig. 8. The figure shows three different models of the

J. Constr. Eng. Manage., 2010, 136(1): 36-48

Table 1. Problem Areas where Application of Empirical Modeling is Most Appropriate

Application area	Empirical modeling advantage	Specific examples in CEM	Issues
Problems that are poorly understood, that is: •where there is limited or no theory quantifying the relationship between the input and output variables; •where a comprehensive list of relevant input variables has not been established; and/or •where the intent is to base the model on correlations between the input and output variables rather than on cause-effect relationships.	A problem, even if it lacks a strong theoretic framework, may be characterized by a comprehensive set of examples of its performance or behavior. Empirical modeling methods have the ability to develop a representation of a problem from such a set of examples.	•Estimating concrete slump and cured strength from the properties and proportions of its constituents, including the profile of aggregate sizes, and the inclusion of admixtures. •Estimating project manager suitability for assignment to a given project, based on key performance parameters measured from previous similar projects, and key descriptors of the current project.	The complexity of the problem that can be considered will be limited by the fact that the number of fitting data examples required to develop the model will increase geometrically with the increase in the number of input variables.
Problems that are overstated, that is: •where there is a lot of overlap/correlation between the values of the input variables; and •where the problem can be solved with alternative subsets of the input variables.	Empirical modeling methods can determine a mapping from an overstated set of input variables to the output variables, which compensates for the excess of information. Moreover, in these cases, an empirical model can work effectively even if some of the input values are missing or contain significant errors.	•Predicting the duration for earth hauling equipment to travel along a haul road, based on parameters such as a series of gradients measured at increments along the haul road, as well as attributes of the haul equipment. •Estimating the load on each axle of a truck, based on the series of strain readings it induces on a bridge girder during the truck crossing event.	The number of fitting data examples required may increase geometrically with an increase in the number of input variables, but only in relation to those input variables that are not correlated with each other. If there is the possibility of missing input values, then the empirical model will require sophisticated internal structuring in order to solve the problem.
Problems that require rapid execution and existing methods are unacceptably slow. In particular, acting as an alternative to numeric simulation techniques (which are notorious for taking several hours or days to run, especially if many simulations must be executed to help determine an optimal design solution or to gain a statistical summary of stochastic processes).	Most empirical models provide a simple function which maps directly from the inputs to the outputs. These functions are not very demanding computationally and can be executed in a fraction of a second. Aspects of a problem that require assembly from many components in a traditional simulation model can be treated as a single variable in an empirical model.	•Modeling dynamic heat-flow through composite materials, such as occurs for endothermic heat generation and dissipation in curing concrete structures. •Modeling acoustic wave propagation in enclosed spaces, such as from noisy construction equipment operated indoors causing health concerns.	Modeling more complicated and/or a broader scope of systems requires a modularized approach. This can place a large burden on the programmers/developers of the original modeling system, that can be time consuming and expensive.

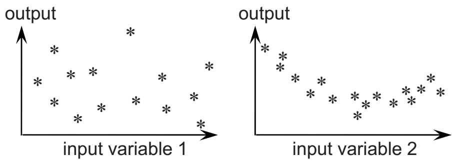


Fig. 7. Plotting output against input for a set of existing observations of the response of a system

same problem, that of estimating the duration to sink a pile given the length of the pile. Each model attempts to fit a line to a set of observations collected from various construction sites. These observations are represented by the *fitting data set* (indicated by the black diamonds) and the performance of each model is evaluated relative to the observations in the *testing data set* (the white squares). The first model attempts to fit a straight line to the data, which appears to be quite successful except for some relatively small errors indicated by the differences between the white diamonds and the straight line. The second model [Fig. 8(b)] recognizes a slight acceleration in the data (probably resulting from increased friction in longer piles) and thus attempts to fit a curved line. This provides a better model in that the errors for the testing data are reduced. There is still some unexplained variance between the data points and the curve, which could possibly be removed by using a more complicated shaped curve or by including additional input variables (such as soil type or rig operator experience). Alternatively, the variance could be due to error (noise) in the data in which case no improvement in the model could be achieved. In fact, using a more complicated curve than is warranted by a problem can lead to a loss in performance. This is seen in the third version of the model [Fig. 8(c)] which is able to

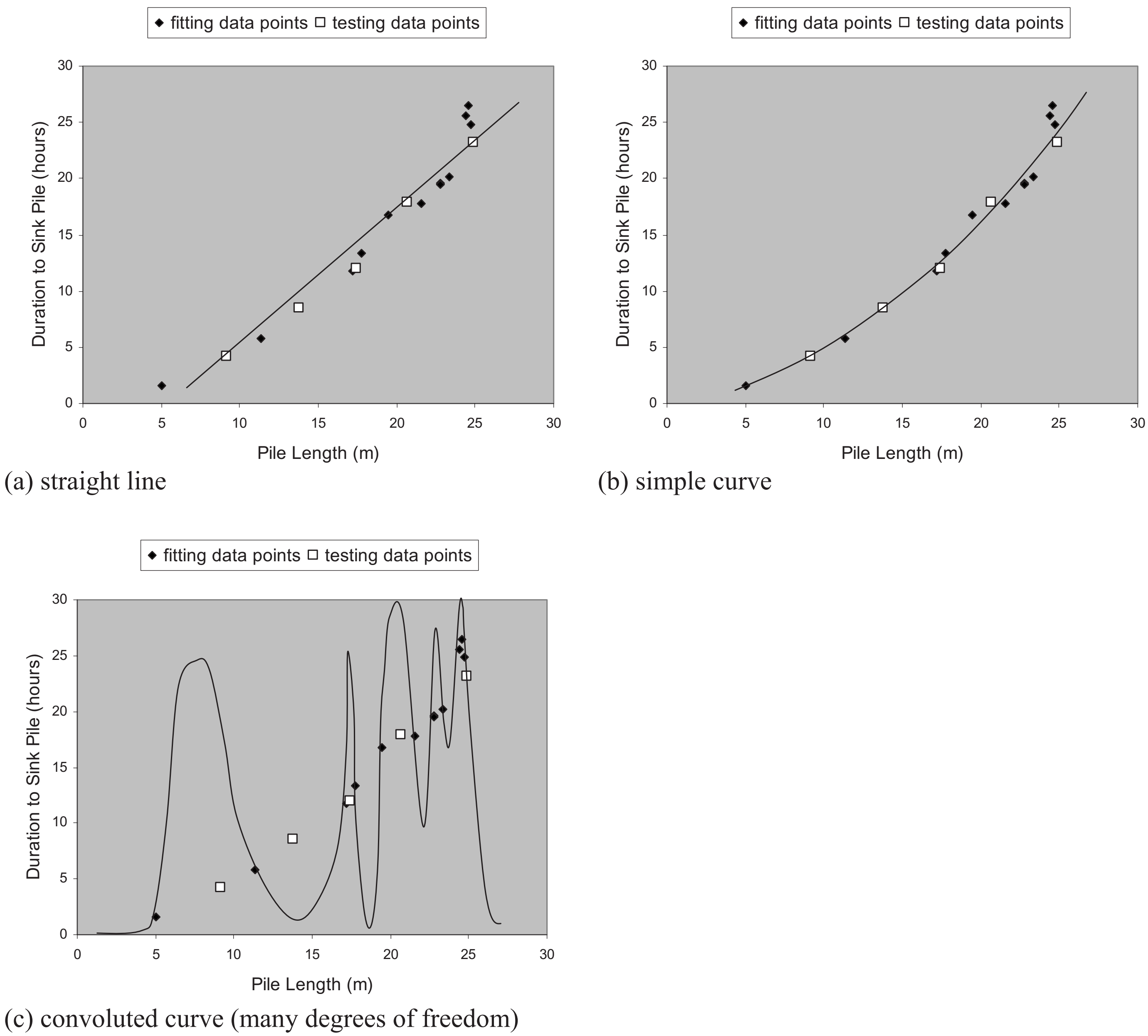


Fig. 8. Fitting functions of different complexity to a set of observations (a) straight line; (b) simple curve; and (c) convoluted curve (many degrees of freedom)

J. Constr. Eng. Manage., 2010, 136(1): 36-48

fit a curve (with many degrees of freedom) very closely to the fitting data points. However, in this case the curve is too flexible (having more degrees of freedom than data points in the fitting set) and thus behaves erratically between the fitting data points. As a consequence, the model performs very poorly when it is evaluated with the testing data (artificial neural networks, for example, can suffer from this problem since they allow the model developer to add an unlimited number of degrees of freedom to the solution). It can be seen that, even for this very simple example, the type and structure of a model must be matched carefully to the problem.

Generally, decisions about the type of model to adopt for a study will precede selection of the model development algorithm. Nevertheless, for any given type of model, there will likely be many choices of development algorithm available. Important factors to consider in selecting an appropriate algorithm are:

1. Whether the algorithm is able to develop the structure of the model or just its coefficients—if it cannot develop the structure of the model, then this may have to be done by the model developer.
2. The degree of expertise required in its use—for example, artificial neural network training algorithms often require a lot of experience in their use in order to get them to converge on a solution, while simulated evolution requires skill in determining a functional coding mechanism for the problem.
3. The likely rate of model development—model development algorithms can be computationally very expensive requiring much iteration to converge on an acceptable solution, and may not be completed within a reasonable period of time.

Finally, it should be remembered that modeling studies often incur failures but, at the same time, provide insights that can stimulate new ideas for solving a problem. The study should, therefore, allow for flexibility in its approach.

Step 2: Data Collation and Evaluation

Most empirical modeling studies require three independent sets of observed data:

1. *Fitting data set*—the observations that the model, through development, attempts to replicate or approximate.
2. *Testing data set*—the observations used to evaluate the performance of the model during its development and to help select between competing models.
3. *Validation data set*—the observations used to make a final assessment of the performance of the model.

The content of all three types of data set needs to be selected carefully to ensure the value and performance of the end model. Ideally, the testing of a model under development should be done in the context of its end application. For example, if a model is dynamic in operation then each observation used for testing the model's performance should be a series of values representing a complete simulation run. Likewise, if the purpose of a study is to identify the internal model structures that form during the development process, such as for the acoustic response model discussed earlier (Fig. 5), then testing of the model should be made relative to the accuracy or utility of those internal structures, not relative to the output from the model. This may not be feasible for testing during the model development stage due to excessive computational demands, but it should at least be adopted when it comes to the final validation of the model.

If existing observations are used for model development, they must be assessed to see if they are sufficient both in number and distribution across the problem domain. The number of observations required will tend to increase with each of the following factors:

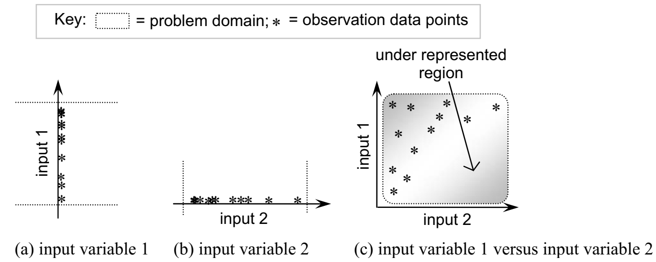


Fig. 9. Distribution of twelve observations across the problem domain (a) input variable 1; (b) input variable 2; and (c) input variable 1 versus input variable 2

1. The number of input variables (generally, this will be a geometric relationship except where the input variables are correlated with each other);
2. The complexity of the shape of the function being modeled; and
3. The amount of error (noise) in the observed data.

While statistical tests exist for determining an appropriate number of sample observations [see for example, Fellows and Liu (1997)] these are limited to assuring the statistical properties of the data such as its mean value. In general, a sensitivity analysis is the most effective way of determining an appropriate number of observations for model development—this will require the model to be developed for a range in the number of observations, starting with a relatively small number then increasing until the required level of model performance is achieved.

The distribution of observations in an existing set should be checked by plotting them as scatter diagrams for each input variable. Figs. 9(a and b) show such plots for a data set comprising 12 observations in a two input variable problem. Both figures indicate a good distribution of points between the limits of the problem domain. However, plotting the distribution for individual input variables can be misleading. Fig. 9(c) shows the same 12 observations plotted for both input variables simultaneously, revealing a lack of points in the lower right corner of the problem domain. For similar reasons, plotting the observations for two input variables for problems comprising three or more variables can also be misleading. Nevertheless, this type of visualization will provide useful insight into the quality of the data set.

If it is necessary to generate additional observations, or if the study requires the generation of a completely new set of observations, then there are several alternative sources that may be tapped:

1. Systems that cannot be controlled for the purposes of data collection such as ongoing construction projects.
2. Systems that can be controlled including laboratory based physical models.
3. Computer-based models such as simulation—in this case, the purpose of the study may be to develop an empirical model that can generate solutions significantly faster than the existing model).

The first of these sources provides the least control over the distribution of the data within the problem domain and thus may require careful assessment of the collected data to make sure that it is well distributed—this may necessitate collecting observations from several examples of the system to achieve a comprehensive set of observations. Systems that can be controlled provide more

J. Constr. Eng. Manage., 2010, 136(1): 36-48

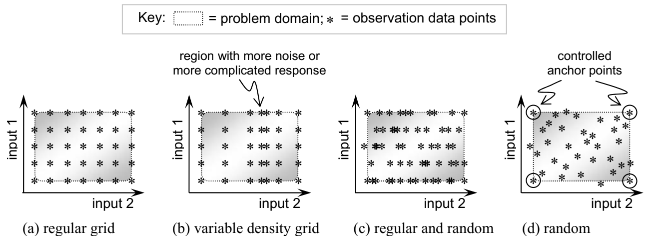


Fig. 10. Distribution of observations collected from controllable systems (a) regular grid; (b) variable density grid; (c) regular and random; and (d) random

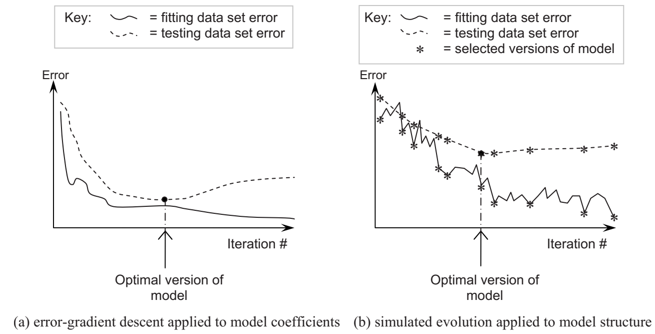


Fig. 11. Progress in model development for studies that use search algorithms (a) error-gradient descent applied to model coefficients; (b) simulated evolution applied to model structure

flexibility in selecting the distribution of observations, but time constraints and other limited resources can be restrictive. Computer-based experiments provide the greatest control, although simulation models can run extremely slowly thereby limiting the number of experiments that can be performed within the project's time constraints.

When no control over a system is available, the model developer should still be able to choose the aspects of the system observed and the timing of the observations. For controllable systems (including computer-based models that are used to generate observations) the model developer must design a set of system input conditions that will ensure the entire problem domain is covered. Within this domain, the model developer may choose between an orderly grid of observations and a random set of values, such as shown in Fig. 10 for a two input variable problem. Fig. 10(a) shows a regular grid, which is usually the most effective distribution strategy to adopt for the *fitting data set*. However, sometimes more information is required about some regions of the problem domain, either because the response of the system is more complicated in that area or because it contains more noise, and so the grid for the *fitting data set* should provide a greater density of points at that location [as illustrated in Fig. 10(b)]. An alternative strategy is to select observation points at random, as shown in Fig. 10(d). This strategy is useful for the *fitting data set* when the collection of observations and development of the model is to be performed in stages. In this case, the number of observations collected at any stage may not be sufficient to form a regular grid. When a random distribution is used for a *fitting data set*, it is still helpful to at least include anchor-point observations at each corner of the problem domain as indicated in Fig. 10(d), to ensure the full scope of the problem is represented. It is also possible to use a regular grid for some input variables and a random distribution for other input variables such as shown in Fig. 10(c).

For the *testing data set* and the *validation data set*, the observations should always be randomly distributed across the problem domain. A mistake sometimes made by researchers is to generate a set of observations using a grid strategy (or mixed grid and random strategy) and then to reserve a random sample of these for testing and validation. The problem with this approach is that these observations will all fall along a finite number of lines, along which the observations used to develop the model are also located, thus biasing the assessment of the performance of the model.

Sometimes, it is desired to measure the performance of a model in a way that is weighted toward variations of a problem that are more likely to be encountered in the end application. In this case, observations in the *testing* and *validation data sets* should be distributed across the problem domain with a varying density that corresponds to the likelihood of that situation being encountered. This can be achieved using Monte Carlo sampling techniques provided that the distribution is known.

As a final point, it should be remembered that empirical models are often only valid when operating within the bounds of the *fitting data set*, so it is imperative that this scope be set to embrace all problems that may be encountered in the end application.

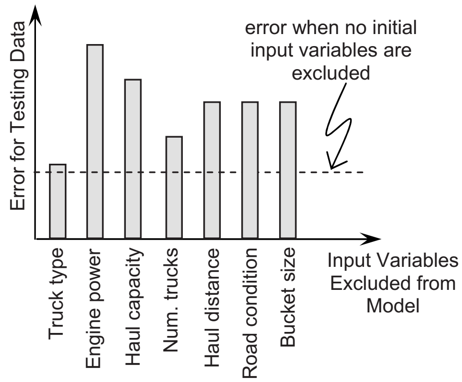
Step 3: Model Development

A conceptual design for the model will have been determined in Step 1. This is developed into a finalized design in Step 3, using the observations in the *fitting data set* and the *testing data set*. The fitting data are the target observations used for model development, while the testing data are the observations used to evaluate the performance of the model at each stage in its development and to select between competing models. A common mistake is to use the same set of data for both fitting and testing. The problem is that the model will be inherently biased toward the data used in its development (the fitting data) and so if these were also used to assess performance it would result in an overestimation of the accuracy of the model. The use of a second independent data set for testing of the model avoids this bias. This can be illustrated by inspecting the output from the three piling models considered earlier in Fig. 8. If the performance of these models was measured for the observations used to develop the model (the black diamonds), then we would conclude that the version shown in Fig. 8(c) is the best (having the least error between the model output and these observations). However, measuring performance for the independent observations in the testing set (the white squares) concludes, correctly, that this is actually the worst of the three models.

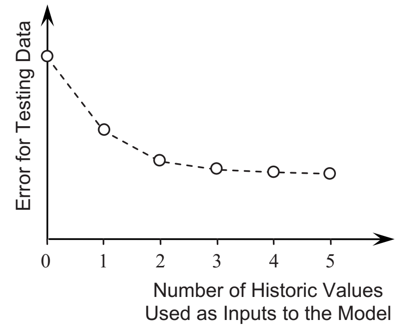
Model development can be a straightforward process executed on a computer in a fraction of a second (such as calculating the coefficients in a linear regression), or it may involve a search algorithm that uses both computer-based methods combined with human judgment and can take several days to complete. Search algorithms can be applied at any level in the development of a model from determining its coefficients through to forming its complete structure.

For studies that use search algorithms, the *testing data set* plays an important role in helping determine when to terminate the model development process. Fig. 11, for example, shows typical progress in the development of a model where: (a) the model parameters are adjusted using an error gradient descent technique

J. Constr. Eng. Manage., 2010, 136(1): 36-48



(a) alternative sets of input variables (see Fig 3(a))



(b) alternative numbers of historic values (see Fig 3(b))

Fig. 12. Searching for an input configuration for the excavation model that minimizes the testing error (a) alternative sets of input variables [see Fig. 3(a)]; (b) alternative numbers of historic values [see Fig. 3(b)]

(such as error backpropagation applied to artificial neural networks (Rumelhart et al. 1986)]; and (b) the structure of the model is adjusted using a heuristic search procedure (such as simulated evolution). In the latter case, milestone versions of the model that provide an improvement in performance (and are thus adopted as the basis for the next iteration in the search) are indicated by the asterisks. In both cases, when the performance of the model is measured for the *fitting data set*, the error curve (shown as a solid line) appears to continue improving. However, when performance is measured for the *testing data set*, the error curve (the dashed line) reaches a low-point and then rises implying that the performance of the model is starting to degrade. In other situations, the error curve for the testing data may just level-out. Given that the *testing data set* should provide a less biased assessment of the performance of the model, it is clear that the version of the model that is adopted should be the one where the testing error curve (not the fitting error curve) ceases to improve.

Finding the minimum point on the error curve for the testing data (as shown in Fig. 11) is only the first step in developing the model. Model parameters that are not adjusted by the search algorithm need to be adjusted manually in a series of comparative performance studies. This should be done methodically, altering the parameters one-at-a-time and seeing how this affects the minimum error that can be achieved by the development algorithm. Examples of model parameters that are commonly adjusted in this way are: the set of input variables included in the model, the type and number of elements included in the model's internal structure, and the number of observations used for developing the model.

To illustrate this point, Fig. 12(a) shows the results for a series of performance experiments involving the static version of the excavation model [shown in Fig. 3(a)] in which each input variable is excluded in turn from the model development process. For comparison, the curve also shows the performance of the model when none of the inputs in the initial set are excluded. From this, a decision can be made as to whether any of the inputs can be permanently excluded. The study could proceed by testing the exclusion of combinations of input variables. Sometimes this type of study will take the form of a sensitivity analysis. For example, Fig. 12(b) shows a search for an acceptable number of historic values to be included as inputs in the dynamic version of the excavation model [shown in Fig. 3(b)]. The model includes three historic inputs as drawn, measured at 5, 10, and 15 times steps in the past, although increasing this number will likely improve the models performance as indicated in Fig. 12(b). It is likely that increasing the number of historic values will increase the accuracy of the model asymptotically as shown in the figure. In this case, the decision about the number to include must be based on practical issues such as at what stage does the model reach an acceptable level of performance, when does the improvement in performance become insignificant, or does the addition of new inputs make the model operate too slowly? Abi-Shdid (2005) provides an example of this type of experimentation to determine the number of historic values to use in an empirically derived dynamic model of heat-flow in buildings.

As a final point, it should be noted that changing the value of one model parameter can affect the optimal value for another parameter, and so the manual search process can require a considerable effort from the model developer in order to home-in on an optimal or at least acceptable solution.

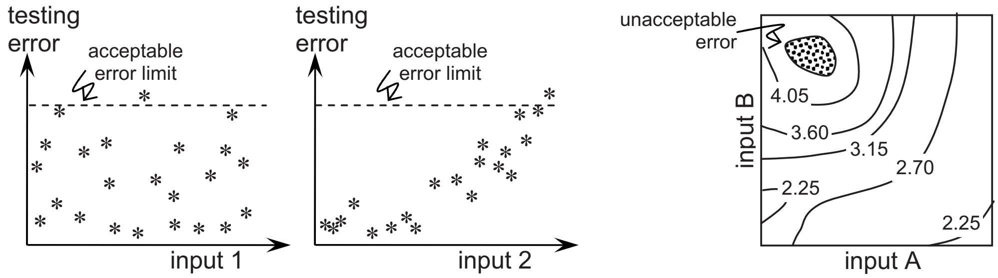
Step 4: Model Evaluation and Final Selection

The development step will have produced one or more candidate models. The next task is to evaluate those models to see whether they perform satisfactorily and, where required, to select the best. This should be executed using the observations in the *testing data set*.

The first set of tests should focus on the generalized performance of the models (integrated across the entire problem domain) using measures such as: the mean absolute error; the mean actual error (for determining positive or negative bias in the model); the worst error; and variance in the error (to assess the distribution of the error). If the model is dynamic with each observation representing a sequence of outputs, then the generalized tests would require the errors to be integrated across each sequence before integrating across all observations. This may include, for example, calculating the mean absolute error for each sequence and then finding the mean or worst of these for all observations. Alternatively, a statistical measure such as Theil's test (Theil 1961) could be applied to each sequence of outputs to assess its closeness to the target sequence, and then the mean and worst of these results could be used as a generalized assessment of performance.

An evaluation should also be made of the distribution of the errors in the model across the problem domain. This is a step that is neglected in most modeling studies, but is essential to assure that the model is performing sufficiently well in all regions of the

J. Constr. Eng. Manage., 2010, 136(1): 36-48



(a) error plotted against each input variable (b) error plotted as a contour map

Fig. 13. Evaluating error across the problem domain (a) error plotted against each input variable; (b) error plotted as a contour map

problem domain. This can be done visually by plotting error against each input [as shown in Fig. 13(a)] or by plotting an error contour for two inputs at a time [as shown in Fig. 13(b)]. Fig. 13(a) shows a model that appears to be performing relatively well across all values of its input Variable 1, but tends to perform worse for higher values of input Variable 2—this may indicate that more observations are required for development of the model in this region of the problem. The approach of plotting an error contour map, as shown in Fig. 13(b), provides additional information about any correlation between pairs of input variables in terms of the occurrence of error. In this case, the error appears to be worst where input A is low and input B is high.

This type of analysis should be performed for all input variables. It may be found that overall the model performs well, but there are certain regions of the problem domain where its performance is unacceptable. Consequently, the model with the best overall performance may not be the one that is selected for final implementation. Alternatively, it may be worthwhile returning to Steps 2 and 3, to collect additional data focused on the poorly performing regions of the problem domain, with the intent of developing an improved version of the model.

Step 5: Final Validation

Once the final version of the model has been selected, it is necessary to reevaluate its performance using a third independent set of observations, the *validation data set*. This provides a final assessment of the performance of the model and confirmation of its validity. Note that the testing data set cannot be used for this purpose since it was used to select versions of the model during the development and final selection stages, and thus may have some bias toward the final model. The same tests that were used in Step 4 can be adopted for the final validation. However, it is essential that the validation relate directly to the way in which the model will be used in the end application: if the model is dynamic, then it must be validated for complete sequences of output values; alternatively, if the study is focused on the internal structures that form during model development, then it must be validated in terms of the accuracy or utility of those structures rather than the accuracy of the output from the model.

Step 6: Implementation and Review

Implementation of the model requires education of the end users (through documentation or tutorials) and should cover the following issues:

1. The collection and organization of the data to be presented as input to the model, to ensure its validity and accuracy;
2. Interpretation of the output from the model; and
3. The usage of the model for problem solving.

When possible, feedback on the performance of the model in practice should be obtained to reconfirm its validity and if necessary to assist in its further development.

Conclusions

Empirical modeling is one of the most commonly used analytical methods in CEM, and is certainly the most diverse in terms of its scope of application. The advent of fast, inexpensive desktop computing, combined with recent advances in empirical modeling development algorithms, has helped make this possible.

A major criticism of empirical modeling is that it is a *black box* method, providing little explanation of its output results and no ability to extrapolate, and is therefore best used as a compromise in situations where a theoretical modeling framework is unavailable. However, this characterization is somewhat unfounded in that: (1) empirical techniques are critical to the development of any theoretical model, making the approach more prevalent than may be realized; (2) if used carefully, empirical modeling can provide insights into the internal structure and principles that drive a system; and (3) modularization of a model can extend its scope of application. That said, most applications have been limited to simple input to output mapping devices that are not designed to identify the internal operation of a system, and often make use of correlated relationships as much as cause-effect relationships between the system's variables. This may be a reasonable approach for many studies but it leaves much of the potential of empirical modeling unexploited. Future efforts should focus on realizing this potential. Classes of problem that cannot be modeled using simple input to output mapping devices, but could be solved by empirical models that have a more complex internal structure, include those where:

1. The set of input values may be incomplete (eg., estimating the construction cost of a water treatment plant from a set of its design parameters where, for some versions of the problem, not all of these parameters will have been determined);
2. The set of input variables may differ from problem to problem (eg., estimating the production rate of earthmoving systems where the number of features on the haul road that have to be defined, such as slopes, will differ from problem to problem);
3. The objective of the study is to develop insights about the system being investigated from the internal structure of the resultant model (e.g., identifying the location and diameter of steel reinforcing bars within concrete structures from the

J. Constr. Eng. Manage., 2010, 136(1): 36-48

acoustic response of the structure—see the example discussed earlier in Fig. 5); and

4. The important information defining a problem is part of a time or spatial series of values, but it is not clear where the key information starts or finishes within that series (e.g., automated recognition of quantities of material in stockpiles from video images where the location of the material within the frame of reference may vary).

References

Abi-Shdid, C. (2005). "ANN coarse-grain based-model for simulating two-dimensional heat transfer in buildings." Ph.D. dissertation, Univ. of Florida, Rinker School.

Ahadzie, D. K., Proverbs, D. G., and Olomolaiye, P. O. (2008). "Model for predicting the performance of project managers at the construction phase of mass house building projects." *J. Constr. Eng. Manage.*, 134, 618–629.

Cheung, S. O., Yiu, T. W. Y., and Yeung, S. F. (2006). "A study of styles and outcomes in construction dispute negotiation." *J. Constr. Eng. Manage.*, 132, 805–814.

Dayhoff, J. (1991). "Pattern recognition with a pulsed neural network." *Proc., Conf. on Analysis of Neural Network Applications*, ASM International, Materials Park, Ohio, 146–159.

Ezeldin, A. S., and Sharara, L. M. (2006). "Neural networks for estimating the productivity of concreting activities." *J. Constr. Eng. Manage.*, 132, 650–656.

Fang, D., Chen, Y., and Wong, L. (2006). "Safety climate in construction industry: A case study in Hong Kong." *J. Constr. Eng. Manage.*, 132, 573–584.

Fang, D., Fong, P. S., and Li, M. (2004). "Risk assessment model of tendering for Chinese building projects." *J. Constr. Eng. Manage.*, 130, 862–868.

Fellows, R., and Liu, A. (1997). *Research methods for construction*, Blackwell Science, Oxford, U.K.

Flood, I., and Christophilos, P. (1996). "Modeling construction processes using artificial neural networks." *Automation in construction*, Vol. 4, Elsevier Science, New York, 307–320.

Flood, I., Issa, R. R. A., and Abi-Shdid, C. (2004). "Simulating the thermal behavior of buildings using artificial neural networks-based coarse-grain modeling." *J. Comput. Civ. Eng.*, 18, 207–214.

Gagarine, N. (1991). "Advance in weigh-in-motion method with pattern recognition and prediction of fatigue life of highway bridges." Ph.D. dissertation, Univ. of Maryland, Dept. of Civil Engineering.

Halpin, D. W., and Woodhead, R. W. (1976). *Design of construction and process operations*, Wiley, New York.

Han, S. H., Kim, D. Y., and Kim, H. (2007). "Predicting profit performance for selecting candidate international construction projects." *J. Constr. Eng. Manage.*, 133, 425–436.

Ko, C. H., and Cheng, M. Y. (2007). "Dynamic prediction of project success using artificial intelligence." *J. Constr. Eng. Manage.*, 133, 316–324.

Lowe, D. J., Emsley, M. W., and Harding, A. (2006). "Predicting construction cost using multiple regression techniques." *J. Constr. Eng. Manage.*, 132, 750–758.

Marzouk, M., and Moselhi, O. (2004). "Fuzzy clustering model for estimating Haulers' travel time." *J. Constr. Eng. Manage.*, 130, 878–886.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). "Learning internal representations by error propagation." *Parallel distributed processing*, Vol. 1, D. E. Rumelhart and J. McClelland, eds., MIT, Cambridge, Mass.

Salem, O., Shahin, A., and Khalifa, Y. (2007). "Minimizing cutting wastes of reinforcement steel bars using genetic algorithms and integer programming models." *J. Constr. Eng. Manage.*, 133, 982–992.

Stoy, C., Pollalis, S., and Schalcher, H. R. (2008). "Drivers for cost estimating in early design: Case study of residential construction." *J. Constr. Eng. Manage.*, 134, 32–39.

Theil, H. (1961). *Economic forecasts and policy*, North-Holland, Amsterdam.

Wong, J. M. W., Chan, A. P. C., and Chiang, Y. H. (2008). "Modeling and forecasting construction labor demand: Multivariate analysis." *J. Constr. Eng. Manage.*, 134, 664–672.

Zhang, X. (2006). "Markov-Based optimization model for building facilities management." *J. Constr. Eng. Manage.*, 132, 1203–1211.

Zhu, L., Schonfeld, P., Kim, Y. M., Flood, I., and Ting, C. J. (1999). "Waterway network analysis with artificial neural networks." *Artif. Intell. Eng. Des. Anal. Manuf.*, 13(5), 12.

J. Constr. Eng. Manage., 2010, 136(1): 36-48