META-MODELLING IN CHEMICAL PROCESS SYSTEM ENGINEERING

Olumayowa T. Kajero¹, Tao Chen¹, Yuan Yao², Yao Cheng Chuang², and David Shan Hill Wong²*

¹Department of Chemical and Process Engineering, University of Surrey, Guildford, GU2 7XH, United Kingdom
²Department of Chemical Engineering, National Tsing Hua University, Hsinchu, Taiwan, R.O.C.

*corresponding author: dshwong@che.nthu.edu.tw

ABSTRACT

Use of computational fluid dynamics to model chemical process system has received much attention in recent years. However, even with state-of-the-art computing, it is still difficult to perform simulations with many physical factors taken into accounts. Hence, translation of such models into computationally easy surrogate models is necessary for successful applications of such high fidelity models to process design optimization, scale-up and model predictive control. In this work, the methodology, statistical background and past applications to chemical processes of meta-model development were reviewed. The objective is to help interested researchers be familiarized with the work that has been carried out and problems that remain to be investigated.

Keywords: Meta-modelling, chemical process system engineering
1. **INTRODUCTION**

Systematic accumulation of knowledge and drive towards optimal design is the key to intelligent and rapid development of chemical processes and products. Traditionally this is done in two distinct approaches: the first-principle approach and data-driven or black-box approach.

First principles, or physical approach requires scientific understandings of the workings of process and integrates them into deterministic input-output simulation models. Simulation models can be developed at different physical scales, e.g. steady state and dynamic process simulations model the operation performance of a plant, computer fluid dynamic (CFD) simulations model the momentum, material and heat transfer in an equipment, and molecular simulations model the relation between molecular structure as input and material property as output.

The black-box or data-driven approach rooted on the statistical theory of design of experiment (DOE) to direct experiments. DOE can be divided into two categories, the exploration of design space, e.g. screening designs and finding the optimum, e.g. response surface method. Traditional DOE theory were based on the assumptions that the input-output relations are relatively simple, consisting of linear, interaction, quadratic effects etc.

With the development of powerful computers, we can include more and more details into first-principle simulation models so as to improve the fidelity of the model. For example, we can model a continuous stirred tank reactor (CSTR) by assuming that it is a well-mix reactor. The mixing and heat transfer can be modelled using a CFD simulator and their effects can be integrated to the well-mix reactor through residence time distribution and heat transfer rate. Alternatively, one can take into account
reactions and change in physical properties with change in composition and temperature in a CFD simulation. Even we a given simulation model, the fidelity can increase by including more mesh into the solver. As the fidelity of the physical model increases, the number of parameters needed to be estimated, i.e. costs of calibrating the first-principle model increase. However, the computer time required for simulation also increase and the high cost the first-principle model becomes difficult to use.

This dilemma leads to a continuous effort to develop meta-models(models of model, or surrogate models) so that knowledge accumulated in such high fidelity models can be used efficiently in design, optimization and control. There have already been many useful reviews and books in the development and application of meta model and design of computer experiments, a brief list is provided here\cite{1,2,3,4,5,6,7,8}. However, to the best of our knowledge, there is no such review attempt specifically from the perspective of a chemical process system engineer. Specifically, we shall attempt to address the following important issues:

(1) model representation
(2) model construction and evolution, and
(3) applications in chemical process system engineering.

2. Meta-Models Representation

2.1 Input-Output Relation

Consider an actual process with input $x = [x_1 \cdots x_K]_x$ and output $y = [y_1 \cdots y_K]_y$.

$$y = \Phi(x)$$ (1)

Let $\Psi$ be a high-fidelity physical model with a set of parameters $\beta$ that predict an output $y^\Psi$ at a given set of input with a set of physical meaning parameters $\theta$
\[ y^w = \Psi(x, \vartheta) \]  

(2)

A meta-model is a model that approximates the high-fidelity model

\[ \Psi(x, \vartheta) = \Omega(x, \beta) + \Gamma(x) \]  

(3)

\( \Omega \) are functional approximators with good flexibility, ability to achieve sufficient accuracy, computational efficiency and simple implementation. \( \Gamma(x) \) is the error of the meta model at a specific input configuration. Common types of meta-models include polynomial, kriging, radial basis function and artificial neural network, etc.

We should bear in mind that the definition of input \( x \) and output \( y \) may be different in different applications. For example, let us consider the CFD model of a heat exchanger with fixed geometry. We can try to construct a meta-model that only applies to hot and cold streams with specific physical properties. The input parameters \( x \) are the inlet flowrates and temperatures of the inlet hot and cold streams. However, if we want to construct a more general meta-model that can be applied to different fluids, then physical properties such as viscosities and thermo-conductivities will also be classified as inputs. Similarly, we can define various sets of \( y \). A simple version of \( y \) will be the average temperature of outlet streams. A detailed version of \( y \) can be the temperatures and the velocities at different points inside the heat exchanger.

2.2 Types of Meta-Models

2.2.1 Polynomial

Polynomial meta-model is perhaps the simplest form of model presentation used in meta-modelling research. Some studies in the literature on this method are: Simpson et al\(^9\), Palmer and Realf\(^10\), Dutournie et al\(^11\), Chen et al\(^12\). Simplicity implies ease in construction and application but also inability to describe complex input-output
relationships.

2.2.2 Kriging, Gaussian Process Model, and Radial Basis Function:

The work of Kriging\(^{13}\) was widely used in geostatistics\(^{14}\) and spatial statistics\(^{15}\). Kriging assumes some form of correlation between points in the multi-dimensional input space, with the correlation being used to predict response values between observed points. A brief introduction to the formulation and construction of Kriging model\(^{16,17}\) are described as follows.

Let \(\hat{X} = [\hat{x}_1, \cdots, \hat{x}_N]^T\) be a set of training data points (sites) and \(\hat{Y} = [\hat{y}_1, \cdots, \hat{y}_N]^T\) be the corresponding response variables for development of a Kriging model. The prediction for a new data point, \(x\) is given by

\[
y(x) = f^T(x) \beta + r^T(x) \Sigma(\hat{X})^{-1} (\hat{Y} - F(\hat{X}) \beta),
\]

where \(f(x)\) contains a set of regression functions of the input variables, and \(\beta\) is the corresponding regression coefficients to be estimated. \(F(\hat{X}) = [f(\hat{x}_1), \cdots, f(\hat{x}_N)]^T\) is a matrix containing the regression functions calculated for all the training data points. \(R(\hat{X})\) is the correlation matrix which is obtained from correlation functions evaluated at each pair of the training points:

\[
\Sigma(\hat{X}) = \begin{bmatrix}
\rho(\hat{x}_1, \hat{x}_1) & \cdots & \rho(\hat{x}_1, \hat{x}_N) \\
\vdots & \ddots & \vdots \\
\rho(\hat{x}_N, \hat{x}_1) & \cdots & \rho(\hat{x}_N, \hat{x}_N)
\end{bmatrix}.
\]

A widely used correlation function is the Gaussian function

\[
\rho(x, x') = \exp[-(x - x')^T \text{diag}[\theta_1 \cdots \theta_{K_x}](x - x')],
\]

while

\[
r(x) = [\rho(x, \hat{x}_1), \cdots, \rho(x, \hat{x}_N)]
\]

is vector of the correlation between a general point in the input space and the training sites. The parameters of the Kriging model are the parameters in the correlation function \(\theta = [\theta_1 \cdots \theta_{K_x}]\) and the regression coefficient \(\beta\). They can be estimated by
the following iterative procedure. First assume a value for \( \theta \), estimate the regression coefficient \( \beta \) by

\[
\hat{\beta} = \left( F(\bar{X})^T \Sigma(\bar{X})^{-1} F(\bar{X}) \right)^{-1} F(\bar{X})^T \Sigma(\bar{X})^{-1} Y
\]

The process variance can be calculated as

\[
\sigma_p^2 = \frac{1}{N} (Y - F(\bar{X}))^T \Sigma(\bar{X})^{-1} \left( Y - F(\bar{X}) \right)
\]

A new set of correlation parameters \( \theta \) can be estimated by

\[
\hat{\theta} = \min_{\theta} \left( |\Sigma(\bar{X})|^{1/N} \sigma_p^2 \right)
\]

The above procedure is repeated until values of \( \theta \) and \( \hat{\beta} \) converge.

Kriging is also termed Gaussian process in the literature with slightly different formulation\(^{18,19,20}\).

Applications of Kriging, Gaussian process models in meta-model development have been extensively investigated by many authors. A chronological, but far from exhaustive, list is provided here\(^{18,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47}\). A review was provided by Kleijin\(^4\) in 2009.

2.2.3 Support Vector Machine

Support vector machine (SVM)\(^{48}\) was originally developed as a supervised learning classifier so that data in a (high dimensional) input space can be classified into groups according to their locations. The SVM can also be formulated into an input-output known as support vector regression (SVR)\(^{49}\). For a specific dimension in the output space, given a training data set \( \bar{X} = [\bar{x}_1, \cdots, \bar{x}_N]^T \) and \( [\bar{y}_1, \cdots, \bar{y}_N]^T \); a nonlinear SVR can be expressed as

\[
y(x) = \alpha_0 + \sum_{i=1}^N (\alpha_i - \alpha_i^*) K(x, \bar{x}_i)
\]

where \( \alpha_i, \alpha_i^* \) can be obtained by solving optimization problem
\[
\max \left\{ -\frac{1}{2} \sum_{n=1}^{N} \sum_{n'=n+1}^{N} \left[ (\alpha_n - \alpha_n') (\alpha_{n'} - \alpha_{n'}) K(\hat{x}_n, \hat{x}_{n'}) \right] + \varepsilon \sum_{n=1}^{N} (\alpha_n + \alpha^*_n) + \sum_{n=1}^{m} \hat{y}_n (\alpha_n - \alpha^*_n) \right\} \\
\text{subject to:}\left\{ \sum_{n=1}^{N} (\alpha_n - \alpha^*_n) = 0 \right. \\
\left. 0 < \alpha_n, \alpha^*_n < C \right. 
\]

The off-set parameter \( \alpha_o \), tolerance parameter \( \varepsilon \) and constraint parameter \( C \) are parameters to be chosen in training\(^50\). The solution of the above problem can be determined using a least square approach that uniquely determined by the input-output training data; the resulting model is known as known as least square support vector machine (LS-SVM)\(^51\). Several reports of using SVR in meta-modelling are given here\(^52,53,54,55,56\).

### 2.2.4 Multivariate Adaptive Regression Spline

The multivariate adaptive regression spline (MARS) was introduced by Friedmann\(^57\) attempts to fit a set of training data \( \hat{X} = [\hat{x}_1, \cdots, \hat{x}_N]^T \) and \( \hat{Y} = [\hat{y}_1, \cdots, \hat{y}_N]^T \) by selections from a set of basis functions

\[
y(x) = \sum_{m=1}^{M} \beta_m B_m(x) 
\]

The set of basis functions is given by:

\[
B_m(x) = \prod_{l=1}^{L_m}[s_{l,m}(x_v(l,m) - t_{l,m})]_+^q 
\]

\( s_{l,m} \) can take values of \( \pm 1 \). \( L_m \) is an interaction order of the \( m^{th} \) basis function.

\( \lfloor \cdot \rfloor_+^q \) is hinge function with the following form

\[
[s_{k,m}(x_v(k,m) - t_{k,m})]_+^q = \begin{cases} 
[s_{k,m}(x_v(k,m) - t_{k,m})]^q & s_{k,m}(x_v(k,m) - t_{k,m}) > 0 \\
0 & \text{otherwise}
\end{cases}
\]

\( x_v(k,m) \) is value of one of the input variables, and \( t_{k,m} \) is hinge point so that the basis function \( B_m(x) \) is cutoff either above or below the hinge point. The basis functions are selected using a forward-backward stepwise selection. MARS offers a flexible spline fit of an input and output relation that can be piecewise continuous. Use of MARS in meta-modelling have been discussed by many authors\(^58,59,10,60,61,62,63,64\).
2.2.5  **Radial Basis Function Network**

A similar form of the above method is the radial basis function network (RBFN)\textsuperscript{65}

\[
y(x) = w_0 + \sum_{i=1}^{n} \omega_i f(||x - c_i||).
\]

The function \( f \) can take many forms such as linear, cubic, thin plate spline, Gaussian, mult-quadratic, and inverse multi-quadratic etc. The parameters of the function include the weighting coefficients \( \omega \) and the basis function centers\([c_1 \cdots c_n]\). They can be obtained by three step algorithm as follows. First a set of basis function centers were chosen by some clustering of training data in the input space (unsupervised learning). Secondly the weighting coefficients were determined by linear regressions. Then both the weighting coefficients and the basis function centers are updated by gradient search. Applications of radial basis function in meta-modelling have also been extensively investigated in the literature\textsuperscript{66, 67, 58, 60, 68, 69, 70, 71, 72, 73}.

2.2.6  **Artificial Neural Network (ANN)**

Artificial neural network (ANN) has been built with the aim of modelling how the human brain functions. They were used in many fields of machine learning and artificial intelligence research such as speech recognition, image processing etc. It should be pointed out that RBFN and SVM are also forms of ANN interpreted in a general sense. However, in this manuscript, we used ANN to denote a common form known as the feed-forward multilayer perceptron (MLP). MLP can be expressed mathematically in the following form:

\[
y^k_i = f \left( \sum_{j=1}^{K^{k-1}} w_{ij}^k f \left( y_{j}^{k-1} \right) + b_i^k \right)
\]

\( y^k_i \) is the output of the \( i \)th neuron in the \( k \)th layer. \( w_{ij} \) is a synaptic weight connecting the output of the \( j \)th neuron in the \((k - 1)\)th layer to the input of the \( i \)th neuron in the \( k \)th layer. \( b_i^k \) is the bias of the \( i \)th neuron in the \( k \)th layer. \( K^k \) is the number of neurons in the \( k \)th layer. \( f \) is known as the activation function which can take many forms.
Sigmoids such as hyperbolic tangent and logistic functions are commonly used. It was proved that a single layer of such network is able to approximate and continuous function\textsuperscript{74}. Applications of ANN in meta-modelling can also be found in wide variety of areas\textsuperscript{75,58,76,77,78,79,80,81,82,83,84,85,86}.

2.3 Feature Selection

In many high fidelity simulations, especially CFD, the output are not limited to a few variables but maps of spatial and temporal variations. While such maps can be discretized to generate a large number of variables, it is not clear how they should be incorporated into the meta-model. Using a dynamic system that has an input vector $x$ and produces output in a period which can be discretized into $y = [y_1, \ldots, y_T]$, Conti et al\textsuperscript{87} explained different possible ways of creating an emulator (meta-model). A multiple output (MO) emulator expressed the output at different time steps as a function of input

$$y = \Omega(x)$$

A multiple single output (MS) emulator expressed the output at each time instant as

$$y_i = \Omega_i(x)$$

A time input (TI) emulator expressed

$$y = \Omega(x, t)$$

Let us use Kriging model as an example, and $n$ be the number of regression functions and $m$ be the number of training data. The training of the MO emulator will required an inversion of $m \times m$ matrix $\Sigma(\tilde{X})$ and $Tn \times Tn$ matrix $F(\tilde{X})^T \Sigma(\tilde{X})^{-1} F(\tilde{X})$. Hence the size of matrix being inverted is dependent of $T$. The training of the MS emulator will required an inversion of $m \times m$ matrix in $\Sigma(\tilde{X})$ and $n \times n$ matrix in $F(\tilde{X})^T \Sigma(\tilde{X})^{-1} F(\tilde{X})$. The size of the matrix inversion is always limited but the training time increases with $T$. In the TI emulator, the number of regression function
$n$ has to be increased to account for variation due to time dependence. Similar conclusions can also be extended to spatially distributed output and spatial-temporal output.

One way of simplifying the meta-model representation is to reduce the dimension of the output in a MO emulator using feature extraction method such as principle component analysis (PCA)\textsuperscript{88}. For example, Chen et al\textsuperscript{20} used PCA to reduce the dimension of a two dimensional CFD simulation of dispersion dynamics of particulate aerosols and showed that output variations of 182250 spatial-temporal grid points can be reduced to scores of 8 principal components that can be efficiently modelled by a GPR. Similar works were reported by Jia et al\textsuperscript{89} in real-time storm /hurricane risk assessment and Wang and Chen\textsuperscript{90} in vapor cloud dispersion.

2.4 **SUMMARY**

In this section, we have examined several input and output relations commonly used for meta-modelling. However, other forms of input and output relations may be used depending on the nature of the problem. Meta-models used are often referred to being parametric or nonparametric. Parametric models (e.g. MARS/RBFN/ANN) usually assumed some forms of basis functions. The model coefficients are obtained from the training data using regression analysis. In principle, we can forget the training data set once the model is determined. The model complexity will increase as more basis functions were added to the systems. Since the number of parameters are not prefixed, therefore is always the danger of overfitting and the generalizability of the model is compromised. Cross validation is usually used to ensure that there is no overfitting\textsuperscript{91}. Non-parametric models (e.g. Kriging/GPR/SVM) are basically interpolation models, the training data constitute the essential part of model parameters and must be memorized to make predictions. It should be noted that the need of data storage is not
really a disadvantage because data storage become less and less expensive with cloud technology and recall of old training data to train new model is also necessary even when a parametric model is used. As far as we know, there is no definitive conclusion that a specific form of input-output relation is superior in meta-modelling. It is generally assumed that input-output relation of a high fidelity first-principle model is inherently complex. Therefore the meta-model must be flexible enough to capture the complexity. Hence there must be an element in the meta model that ensure that it can serve as a universal function approximator.

3. Meta-Model Construction

3.1 Design of Computer Experiment

The intuitive approach to meta-model construction may view each simulation run as an experiment and use traditional design of experiment (DOE) procedure to determine the locations of each simulation runs to be executed. This area is known as design of computer experiments.\textsuperscript{18, 21}.

Traditional DOE methods can usually be classified as either one of the following two objectives, (i) screening experiments that try to identify factors (input variables or combination of input) that have statistically significant effects on the response (output); and response surface experiments that try to build a input-output relation which can predict the optimal input conditions\textsuperscript{92, 93}.

Classical screening experiments include Factorial Design, Fractional Factorial Design, , etc. Another general class of screening are “optimal designs” that optimize some form of metric of the information matrix \[\left| \left( \hat{X}^T \hat{X} \right)^{-1} \right|\] where \(\hat{X}\) is matrix of the training data in a generalized parametric input space. For example the D-optimal design minimizes the determinant of Fischer information matrix.

Common experimental design for response surface experiments Central Composite
Design and Box-Behnken Design. Such designs usually possess the desirable property of having low prediction error.

In design of experiments, two kinds of errors are considered: (i) random errors that is measured by the variance of the predictions, (ii) systematic errors caused by mismatch of the model assumed and the actual response, i.e. bias. In design of computer experiments, the random error is expected to be small. Therefore designs that are based upon a preselected model and the assumption that bias is small compared to variance, such as the D-optimal design may not be appropriate.

Furthermore, even simulations are relative inexpensive compared to experiments, their execution takes time and effort. Therefore the number of runs a major concern. Relatively large number of runs are still required for some of these designs, especially when the dimensionality design space increases. Small response surface design can be used to reduce the number of runs Plackett-Burman Design\(^{94}\), small composite design\(^{95}\), saturated design\(^{96}\) (see review by Draper and Lin\(^{97}\) and Meyers and Montgomery\(^{93}\)). However, as pointed out by Allen and Yu\(^{98}\), the desirable properties of having prediction error may be compromised. Therefore design methods that sufficiently accounted for the effects of bias while without sacrificing variance, e.g. based on expected integral mean square error should be considered (Allen et al\(^{99}\)).

Alternatively, another approach is to explore the design space so as to minimize unsampled region. Examples of space filling design include: random or Monte Carlo sampling\(^{100}\), orthogonal arrays\(^{101}\), uniform design\(^{102}\), Latin Hypercubes Sampling (LHS)\(^{103,104}\), Hammersley Sampling (HSS)\(^{105}\). To measure the space filling nature of the design, a measure of the space filling nature should be defined, e.g. the modified \(L_2\) discrepancy is often used for input variables scaled between \([-1,1]\]

\[
ML_2 = \left(\frac{4}{3}\right)^{K_x} - \frac{2^{1-K_x}}{N} \sum_{n=1}^{N} \prod_{k=1}^{K_x} (3 - \hat{x}_{nk}^2) + \]

\[ \frac{1}{N^2} \sum_{n=1}^{N} \sum_{n'=1}^{N} \prod_{k=1}^{K} (2 - \max(\hat{x}_{nk}, \hat{x}_{n'k})) \]

The smaller the value, the better is the space filling properties. Another measure is the minimum Euclidean distance between the design points \(^{106}\)

\[ M_m = \min_{n,n'} \| \hat{x}_n - \hat{x}_{n'} \| \]

The larger the value is the better. To balance between space-filling properties and orthogonality, Cioppa and Lucas \(^{107}\) proposed a design that created the best space filling properties, subject to constraints of orthogonality measures such as the maximum correlation between various dimensions \(\rho_{\max} = \max_{k,k'} \text{cor}(\hat{x}_k, \hat{x}_{k'})\) and conditional number of the information matrix of sampling points \(\text{cond}(\hat{X}^T \hat{X})\) in the design space.

3.2 **Iterative Approach**

Despite the development of numerous sampling schemes, it is generally not realistic to expect a reliable meta-model can be build or the optimum of the actual model can be located using a single set of sampling data. It is necessary to balance between optimization and the fidelity of the approximate-model using iterative approach \(^{108}\). Early work in this area, reviewed by Wang and Shan \(^3\) include reducing the dimension of design space by eliminating unimportant variables \(^{109, 110}\). An alternate approach is to survey a small design space and move limits towards the optimal region \(^{111, 112, 113, 114}\). Wang and Simpson \(^{115}\) used a preliminary meta-model to generate a large number of points in the large design space and then cluster promising points. Meta-models for local region around cluster centers are then generated to refine the optimum. The above are optimization iterations designed for finding the optimum. Khu et al \(^{60}\) also developed an evolutionary strategy using genetic algorithm for determination of the sample points. However, the objective function to be optimized is the sum square error between the actual model and predicted values of a validation data set. Thus the
iterative approach is designed to increase the fidelity of the model. Chen et al argued that, even in optimization, with limited data, the initial meta-model cannot be completely trusted. Thus in early stages of the search, we need to focus on exploring under-sampled region. At later stages focus should be on finding the optimum since enough data has been accumulated for building an accurate enough model. An index, the predicted fitness of objective function at a sample point is defined as the information energy. An index of information content of sampling point was defined as the information entropy. New samples were generated based on Monte Carlo importance sampling of the free energy, with “temperature” which is related to the number of previously sampled points being a balancing parameter. Tang et al and Chi et al also used an evolutionary meta-model optimization in which the prediction uncertainty of the GPR was used in addition to the prediction mean to determine the future sampling points. An alternative method of preventing premature convergence is to use an index that took care of the uncertainty of the prediction, known as expected improvement (EI). EI recognizes that the predicted improvement is a random variable, because the meta-model gives prediction mean and variance. Therefore, the predicted improvement should be integrated with its probability density function to attain its expected value, i.e. EI, which should be the objective function to be optimized.

3.3 MODEL MIGRATION

In many engineering applications, simulations were often called upon to solve problems with some degree of similarity. For example, CFD simulations were carried out and a “old” meta-model has been constructed for a given equipment with a set of geometrical design parameters with a fluid of given rheological properties. It would most desirable if we can construct “new” meta-model for another piece of
equipment that is similar in structure but with slightly different geometries and/or another fluid with different rheological properties, using the given “old” meta-model plus only a few simulation data. Another typical problem of in chemical engineering is scale-up of reactor. It is well known that optimal operating condition found for a reaction out in a lab size reactor may not work when a reactor is scaled to larger dimensions. Although kinetics and thermodynamics will not change as the equipment is scaled up. Mixing and heat transfer changes with dimension. CFD simulation were often used to solve scale-up problems, but CFD coupled with reaction is notoriously difficult to execute. Hence development of meta-models for design and optimization is necessary.

Gao and coworkers\textsuperscript{120,121,122,123} recognized the problem of how to efficiently modify an existing meta-process model to fit a different yet similar process. The existing model is denoted as the “base model”, while the model to be developed for the new process is called as the “new model”. At least one of the following two objectives are expected to be achieved.

(1) To attain similar prediction accuracy, fewer data are required for migrating from the base model to the new model than for developing an entirely new model without the migration step.

(2) The migrated new model is more accurate than the model developed without the aid of migration if nearly equal number of experimental data are used in model training.

Motivated by the standardization step in calibration model transfer \textsuperscript{124}, the migration can be conducted by a parametric scale and bias correction (SBC) of the base model\textsuperscript{120,123}:

\[
y(x) = \text{diag} \left[ s_{y,1} \cdots s_{y,K_y} \right] f_0 \left( \text{diag} \left[ s_{x,1} \cdots s_{x,K_x} \right] x + b_x \right) + b_y
\]
where $f_o$ denotes the base model; $s_{y,1} \cdots s_{y,K_y}$ and $s_{x,1} \cdots s_{x,K_x}$ are scaling parameters of different dimensions of the output and input space; $b_y$ and $b_x$ are bias vectors in the output and input space respectively. Data of the new process are used to determine the scale and bias transformation parameters. The SBC, being linear transformations in output and input space, is an arbitrary similarity condition that may not have sufficient flexibility to model the new process of interest. To overcome this limitation, Yan et al.\textsuperscript{125} proposed a Bayesian migration method to update the scale-bias functions given experimental data from the new process. Their method is named as functional SBC, which is based on a GPR model framework. The input-output relation of the functional SBC is given by

$$y(x) = s(x)f_o(x) + \delta(x)$$

with $s(x)$ being a linear scaling function

$$s(x) = s_{yo} + diag[s_1 \cdots s_{K_y}]x$$

and bias correction $\delta(x)$ is chosen to be a Gaussian process with zero-mean. The underlying assumption that if the new process is similar to the old process, $\delta(x)$ would be quite close to zero and its determination would require much less data points. Applicability of this method was recently demonstrated using sequential sampling and Bayesian techniques\textsuperscript{126,127,128}.

No physical reasoning has been given in developing SBC migration. However, it is well known that equations in transport phenomena can be reduced into universal form with dimensionless groups such as Reynolds number, Prandtl numbers and Nusselt numbers. Therefore, input-output relations of CFD may be scale-transformable through these dimensionless group. Recently Shen et al.\textsuperscript{129} demonstrated the advantages of performing experimental design using a dimensionless input space. Such advantages should be much more evident if applied to the design of computer
CFD experiments.

The functional SBC transformation can find its root from the work of Qian et al\textsuperscript{38} where the same method was used to migrate between a model of low accuracy (LE) experiment and a high accuracy (HE) experiment. The concept is especially important for design of computer experiments, because complex computer codes can generate results with different levels of mesh densities. It is always desirable that a meta-model can be generated using more low fidelity runs and a small number of high fidelity runs. Kennedy and O’Hagan\textsuperscript{130} first proposed using an auto regressive relation with a Gaussian process model bias term to connect meta-model of different levels of mesh fidelities. The problem of migrating between different levels of fidelity due to mesh densities has been well researched by many authors\textsuperscript{131, 132, 133, 134, 135, 136, 137, 138}.

It should be pointed out the complexity of a computer code can be increased not only increasing the mesh but increasing the physical processes being considered. Chuang et al\textsuperscript{139} demonstrated by using a simple well-mixed CTSR model that contains only the kinetics of the reaction as the base model, new meta-models can be readily developed for full CFD simulations that take into account of mixing and heat transfer.

In theory, using the Gaussian process model for a bias term allows us to include any differences between the “new” and “old” processes. It should be noted that the migration is feasible even if new variables were added to the “new process”. The success of migration depends on the functional similarity between the input-out relations of “new” and “old” processes in the old input-space. In other words, as the computational model become more complex, less and less significant variations will be left out. The bias term approaches a zero function.

3.4 Summary

In this section, various sampling method for constructing meta-models were
reviewed. The general consensus up to now is that space filling design should be the most appropriate for building meta-models. However, when a meta-model was used for optimization, a balance of importance sampling and space-filling needs should be considered, especially in evolutionary procedures. Migration of meta-models is also an important research area. Migration between different meshing densities has been extensively studied. Dimensional analysis is the underlying assumption of scaling transformation. The bias term in the functional scale and bias correction incorporate enough flexibility to allow for migration between models of different complexities due to inclusion of additional physical considerations.

4. APPLICATIONS

4.1 PROCESS DESIGN AND OPTIMIZATION

The most straightforward application of meta-modelling is process design and optimization. Meta-model optimization has already been extensively used in design and optimization of many different processes. A wide variety of applications include flowsheeting\textsuperscript{10,140,141,142,143,144}; boiler and combustion processes\textsuperscript{145,146,147}; separation processes such as simulated moving bed chromatography\textsuperscript{148}, pressure swing adsorption\textsuperscript{149} heated integrated column\textsuperscript{144}, divided wall column\textsuperscript{150}, CO2 capture process\textsuperscript{151}; reactor operation such as iron oxide reduction\textsuperscript{152}, nano-particle synthesis\textsuperscript{153}, bacteria cultivation\textsuperscript{154}; polymer processing\textsuperscript{155}; chemical processes in semiconductor industry\textsuperscript{156,157,158}; etc. In should be pointed out that in some of these work, actual experiments instead of higher fidelity simulations were used. Using data-driven models as response surface model and iterative evolutionary strategy to direct additional experiments and finding the optimum process condition can be classified in general as “meta-model assisted optimization”. 
A notable development is the use of surrogate model in superstructure optimization, in which a surrogate model with categorical variables is required\textsuperscript{141}. However, the construction of a meta-model with categorical variables has not been thoroughly investigated. The intuitive approach is to create samples at every possible combination of categorical variables. However, such an approach would not work if the number and levels of categorical variables are large. Qian and Wu\textsuperscript{159} have discussed the use of GPR with quantitative and qualitative input as a surrogate model for computer experiments. The covariance structure between category variables were estimated using data sampled so that one does not have to perform experiments at every possible combination of categorical variables. However, the parameter estimation, or training, involved a complex semidefinite optimization problem that is difficult to solve. Experimental design methods have been proposed\textsuperscript{160}. The potential of such models, sampling and optimization strategy in flowsheet and equipment design for chemical engineers is enormous. Unfortunately, there seems to be no research on this subject in the chemical engineering literature.

4.2 Process Control

It should be pointed out that there are numerous studies data-driven models such as ANN\textsuperscript{161}, RBFN\textsuperscript{162}, SVM\textsuperscript{163}, GPR\textsuperscript{164} can be used to represent nonlinear time series. Such models have been used in soft-sensors development to predictive important quality variables online\textsuperscript{165,166,167,168,169,170}. They can of course be used in nonlinear model predictive control (NMPC)\textsuperscript{171,172,173}. However, such data-driven models cannot be considered as meta-model discussed in this work, because they are not surrogates of a more complex model. Moreover, they are usually obtained from online data, thus experimental design to construct these models received much less interest to adaptive
or just-in-time strategies. Tsen et al\textsuperscript{174} proposed a hybrid approach in which first-principle simulation data were trained together with experimental data to obtain an ANN model for use in control. Such hybrid models\textsuperscript{175,176} were developed because of the need of using prior first-principle knowledge to avoid unreasonable extrapolations and the necessity to accommodate with experimental information, i.e. migration to a more accurate and realistic model for control purposes.

Conceivably, a surrogate dynamic model for control can be generated using complex high fidelity dynamic simulation. However, many researchers prefer to use another technique known as the “reduced-order-model” (ROM) in which the surrogate was developed by reducing the original differential algebraic equations (DAE) system into lower order models by collocation or other mathematically rigorous simplification techniques. Such methods have been developed for distillation columns\textsuperscript{177}, bio-reactors\textsuperscript{178}, air separation\textsuperscript{179,180} etc. Since the reduced models is still an equation based models, ROMs are not meta-model as we have discussed.

Perhaps the closest form of meta-model in process control is under the banner of “approximate dynamic programming” for optimal control\textsuperscript{181,182}. The objective of optimal control (and dynamic programming) is to minimize a certain “cost-to-go” function, which is usually a time-discounted sum of costs of individual time points. The major challenge under this framework is that the computation needed to evaluate the cost-to-go is often very high, and thus on-line control is almost infeasible. Here the idea of meta-modelling can help: a large number of simulations can be conducted off-line to generate simulation data, and a meta-model (termed “approximate cost-to-go function”) can be developed to approximate the relationship between state/manipulated variables and the cost-to-go function. This approximate function is then used for on-line control, and also subsequently improved.
4.3 Model Calibration

The advantage of using meta-model as a surrogate of high fidelity model so that design space can be efficiently explored for process improvement is intuitive and straightforward. However, meta-model can also be used to improve predictions of high fidelity model. Typically, a high fidelity model requires a set of physical meaningful parameters \( \theta \) to make predictions (see equation 1). For example, in CFD simulations of reactors, \( \theta \) may consists of transport properties such as viscosity, thermoconductivity, diffusion coefficient, surface tension, thermodynamic properties such as heat capacities, model parameters for vapor-liquid equilibrium calculations, as well as kinetic parameters such as rate constants and activation energies. Theoretically, these parameters can be measured by independent experiments. In practice, they have to be calibrated by fitting simulation results with experimental data. To do so the high fidelity simulations has to be carried out at different parameter settings for each of the experiment conditions. This is of course computationally laborious and often impossible when the number of parameters to be determined is large.

Alternatively, we can construct a meta-model that includes the parameter \( \theta \) as our input and characteristic experimental observations as our output. Meta-model can be constructed with selected simulations at certain experiment conditions and parameter settings. Iterative sampling can be carried to minimize a fitness-function of sum-square-errors (SSE) of meta-model predictions of all experimental data.

The calibration of computer models dates back as far as 1978 (O’Hagan, 1978\textsuperscript{183}), where the Bayesian calibration approach was used. This involves fitting the posterior distribution of best input parameter. The application of Bayesian calibration approach to univariate computer models was demonstrated by Kennedy and O’Hagan\textsuperscript{184}. Similar
application in multivariate, temporal and spatial computer models have been demonstrated in several papers\textsuperscript{185,186,187,87,188,189}. Other calibration methods have been mentioned in the literature different from the Bayesian approach. Some of them can be found in the book by Kleijnen\textsuperscript{46} where calibration was referred to as simulation optimization.

Examples of model calibration applications include: (1) energy simulation model of buildings, using a Gaussian process regression technique with a radial basis function kernel\textsuperscript{190}, (2) water distribution systems, involving an evolutionary-based meta-model with an innovative hybridization of Genetic Algorithm and Radial Basis Function\textsuperscript{60} (Khu et al, 2004); an Artificial Neural Network linked to Genetic Algorithm\textsuperscript{191,192,193} (3) mechanical and aerospace systems which utilizes Radial Basis Function with non-dominated Sorting Genetic Algorithm (NSGA-II) (Khalfallah and Ghenaiet, 2015).

4.4 **SENSITIVITY ANALYSIS**

A computer model can help us to locate the optimal operation of a process, it can also help us to evaluate the sensitivity of the response to a certain input. Sensitivity analysis can also be applied uncertain parameters of a model. Sensitivity can be characterized locally by carrying out one-at-time changes to each input and examine the effect on output. Chang et al provided an example of such approach biochemical network\textsuperscript{194} was analyzed and simplified. Alternatively global variance based index such as the Sobol indices\textsuperscript{195}, fast amplitude sensitivity test (FAST)\textsuperscript{196,197}, high dimensional model representation (HDMR)\textsuperscript{198}, polynomial chaos expansion (PCE)\textsuperscript{199,200} etc. can be calculated. Calculation of these global sensitivity indices is of course time-consuming using the high fidelity model. However these indices can be relatively easy using meta-models.\textsuperscript{30,40,201,202}. Applications of sensitivity analysis
(and the closely related uncertainty analysis), with or without a meta-model to chemical engineering related problems include reaction kinetics\textsuperscript{203,204}, biological system modelling\textsuperscript{205}, process design\textsuperscript{206}, enhanced oil recovery simulation\textsuperscript{207}, vapour cloud dispersion\textsuperscript{90}, etc.

4.5 SUMMARY

In this section, we have examined some applications of meta-modelling in the field of chemical process system engineering. As we have seen, the most common application is process optimization and design. However, meta-model development with categorical variables is only in its early stage. Thus applications of meta-model optimization to design problems involving flowsheet structure and non-linear integer programming have been limited. While there are many applications in using data-driven soft-sensor and reduced order models in nonlinear model predictive control, these models are not meta-model as defined in this study. The closest form of meta-model application is approximate dynamic programming. We have also pointed out that meta-model can be used for model calibration and sensitivity analysis. Application of sensitivity analysis to chemical engineering related problems, especially analysis of kinetics reaction network has been well researched, partly because the need of a meta model is much less due to relative ease of ODE simulations.

5. CONCLUSIONS

The objective of this review is to provide chemical process system engineers with the statistical background that has been developed for design of computer experiment and use of meta-models for optimization. We have introduced various forms of meta-model representation. Interpolation model such as Kriging/GPR seems a useful model in many cases. Methods of sampling in meta-model construction and optimization
has also been surveyed. In general we found that space filling design is the suitable choice of initial experimental design and some form of balance between exploring uncertain region and confirmation of optimal prediction must be struck in evolutionary optimization. Examination of applications of chemical engineering related problems found that most applications of meta-modelling emphasize on optimization. However, optimization with categorical variables is still under-researched because meta-model theory and sampling techniques for this kind of problems have not been fully understood. With such knowledge, our ability to use complex, high fidelity, time consuming simulation tools more efficiently in understanding process behavior, optimize process design, and improve process control, can be enhanced.

ACKNOWLEDGEMENT

O. Kajero’s PhD is funded by the Tertiary Education Trust Fund (TETFUND), Nigeria in collaboration with University of Lagos, Nigeria. This work was also partially supported by an International Exchange grant co-funded by the Taiwan Ministry of Science & Technology (Grant number: MOST 104-2911-I-007-516) and the UK Royal Society (Grant number: IE140859).

REFERENCES


45. Roustant, O.; Ginsbourger, D.; Deville, Y., DiceKriging, DiceOptim: Two R Packages for the Analysis of Computer Experiments by Kriging-Based Metamodeling


64. Costas, M.; Díaz, J.; Romera, L.; Hernández, S., A multi-objective surrogate-based


121. Lu, J.; Gao, F., Model migration with inclusive similarity for development of a


157. Liau, L.-K.; Huang, C.-J.; Chen, C.-C.; Huang, C.-S.; Chen, C.-T.; Lin, S.-C.; Kuo,

159. Qian, P. Z.; Wu, H.; Wu, C. J., Gaussian process models for computer experiments with qualitative and quantitative factors. *Technometrics* **2012**.


176. Fullana, M.; Trabelsi, F.; Recasens, F., Use of neural net computing for statistical


