Automated IC engine model development with uncertainty propagation

A. J. Smallbone  
cmcl innovations, Cambridge, U.K.

G. Brownbridge, W. Phadungsukanan, M. Kraft  
Department of Chemical Engineering and Biotechnology, University of Cambridge, Cambridge, U.K.

B. Johansson  
Division of Combustion Engines, Department of Energy Sciences, Lund University, Sweden

ABSTRACT

This paper describes the development of a novel data model for storing and sharing data obtained from engine experiments, it then outlines a methodology for automatic model development and applies it to a state-of-the-art engine combustion model (including chemical kinetics) to reduce corresponding model parameter uncertainties with respect engine experiments. These challenges are met by adopting the latest developments in the semantic web to create a shared data model resource for the IC engine development community. The relevant data can be extracted and then used to set-up simulations for parameter estimation by passing it to the relevant application models. A methodology for incorporating experimental and model uncertainties into the model optimization procedure is presented.

Data from seven operating points have been extracted from the proposed data model and have been incorporated into a state-of-the-art in-cylinder IC engine model through the optimization of model parameters whilst accounting for the model parameter and experimental uncertainties.

INTRODUCTION

Exploiting the useful information or knowledge held within the vast quantities of experimental data produced during the on-going engine development process is a major challenge for the academic and industrial community. Those that master these data and apply the knowledge effectively can develop more efficient and low cost engines over shorter timescales. The methodology outlined in this research paper seeks to deliver increased predictive capability and model robustness through automated model development.

“Process Informatics seeks to solve these problems through the integration of hardware, middleware, software, databases, and human resources, all integrated through a network” [1]. An example of a practical working system is the Process Informatics Model (PriMe) [2] which is used for the determination of chemical kinetic rates for combustion based on an open-source database written using eXensible Mark-up Language (XML). The objective was to reduce the uncertainty on model parameters through systematic comparison with fundamental experimental data and a set of computer-based tools to process data consistently from all available data sources. Researchers aimed to systematically identify regions in which the models are unsuccessful, then suggest to the community the most useful future experiments thus resulting in more rapid development timescales [3, 4].

This methodology has already been adopted to carry out systematic model developments for simple empirical expressions [5]. This research paper describes the continuing implementation of a Process Informatics based model for advancing IC engine development. Details of a web based engine data storage tool and model integration are described, this work builds on previous research activities [5] by firstly extending the data model and the complexity of the application model, this time a state-of-the-art engine combustion model including chemical kinetics with a total of 42 model parameters with their corresponding uncertainties.

THE PROPOSED METHODOLOGY

To allow the more rapid development of quantitatively robust and accurate models several changes have been made to the current approaches to model development. These are:

- The definition of a model has to be extended to include all of the physical knowledge that goes into its development.
- The storage of experimental data needs to be standardised to make it easier to use it for model development.
• The optimisation of model parameters that are not known accurately enough to give good predictions needs to be performed automatically, using mathematical algorithms.

By making these changes it will be possible to maximise the amount of physical information that goes into the model. Each of these points is described in more detail in the following sections followed by descriptions of the specific data model and optimisation procedure that were used to optimise the present engine model, the results of which are presented and discussed at the end.

THE DEFINITION OF A MODEL

Conventionally, a model is usually considered to be the set of mathematical equations which describe a physical system or process. In general, these descriptions are incomplete at some level and require the inclusion of unknown parameters. An optimal set of values for the parameters are then usually found using experimental data. This means that not only are these values coupled to the data that is used but the uncertainty in the values is dependent on the uncertainties in the data. Using different experimental data will often produce a different set of values with a different level of uncertainty in these values. This can lead to problems where multiple reference values can disagree [6]. The first step in overcoming this problem is to redefine the concept of a model to explicitly include the experimental dataset used for the parametric optimization as well as the mathematical equations.

Experimental data is usually measured using electrical devices that give outputs, for example voltages. This raw data then requires processing to give useful measures such as temperature and pressure. The resulting processed data is therefore dependent on the choice of instrumental model adopted at this the processing stage. Taking this into consideration, a model as described here can be separated into three sub-components: instrumental models, data models and application models. These are shown schematically in Fig. 1. Descriptions of these sub-components and how they are included in the presented methodology are outlined in the following sections.

Data models

A data model is a description of how data is stored. The data model vocabulary employed for the storage of results obtained from an experiment often depends on the apparatus and/or personal preference. Furthermore, the data can be poorly labelled with incomplete descriptions of the process conditions being recorded making it redundant without its corresponding experimentalist. This limits the data’s useful lifetime and its accessibility whilst increasing its perceived uncertainty. Ultimately, the potential to exploit the knowledge held within the data to facilitate other relevant development activities is reduced. Finally, due to the on-going advancement of data storage technologies, these data are sensitive to loss: via computer hard drive failure; use of storage formats which are no longer supported e.g. zip drive; accidental neglect (and deletion); or even the thought that older experimental measurements are irrelevant. Given the financial costs associated with the acquisition of each data point and the knowledge of the "real world" held within them, any losses of these kinds are inexcusable. The methodology presented here aims to encourage the use of standardised data models such as the one for engine data presented in this paper.

Instrumental models

In order to extract useful data for use in analysis or modelling, some observations of the "real world" must be made. Observations are usually carried out using electronic devices which produce raw data such as voltages. This raw data are then processed to produce point data such as brake power, fuel consumption, exhaust emission etc., as well as time dependent data such as in-cylinder pressure measurements etc. This processing is done by generating practical converting the raw data into something useful via an instrumental model, this conversion carries some uncertainty as it is often based on a set of parameters determined from correlations, hence associated uncertainty must be stored and its influence considered whenever adopted.

Figure 1: Overview of the sub-components of a model
Currently instrumental models are generally incorporated into the data and application models. For example data are usually processed using instrumental models prior to storage and then further processing often happens within the application model to get the parameters in the form required by the model.

**Application models**

An application model is the set of mathematical equations which describe the physical processes which occur in the "real world". This aspect of the model is often the main focus of the development process for example by extending the underlying physics from a 1D to a 3D computation.

**MODEL VALIDATION AND OPTIMIZATION**

The automation of these procedures allows the use of larger data sets than can be processed manually. Also the use of statistical algorithms enables information, such as uncertainties in the experimental data, to be made use of in the development process more easily.

The tuning/optimisation of models and in particular chemical mechanisms is already common both for the generation of comprehensive mechanisms [7] and for producing reduced mechanisms [8] that are more practical for industrial I.C. engine development. Historically this has been carried out by hand but with the continuing increase in computing power mathematical algorithms are being used more frequently [8]. Whilst these algorithms cannot replace human experience they can greatly speed up the model development process; allow much larger data sets to be used and enable engineers to measure model robustness in far shorter timescales.

Some parameters found within engine models are very hard to measure from experiments and are too complex to calculate using fundamental theory; this means that they need to be tuned to fit experimental data. Parameters can be calculated or inferred from experiments but also often require further optimising to be used for engine development because the initial estimates are usually determined using combinations of experience, experimental data and chemical theory. Different combinations of these can yield different sets of parameters [6] leading to inconsistencies in the chosen parameters for different research groups. This is largely due to uncertainties in the experimental measurements and theoretical models used. The methodology described here takes these uncertainties into account so as to generate models that are more generally applicable whilst also providing estimates of the uncertainty in the adjusted values.

To senior engineers, the most significant aspect of model development is the validation and optimization phase as success or failure dictates the distribution of future model development resources. Hence, measures of model success must be considered more systematically by determining both the robustness and accuracy of the model when compared to a wide range of experimental data.

It is important to recognize that models have sets of parameters that require optimization against experimental data. It is therefore necessary to embrace this fact and try to control the influence that the experimental data has on the model predictions itself. Here we distinguish five potential sources of inaccuracy of a model:

- Errors associated with the numerical solution method of a mathematical model;
- Errors that are based on the missing chemical or physical insight into the process that is to be modelled;
- Errors based on the inaccuracies of the data used in a model;
- Errors that arise because there are conflicting data for a number of parameters.
- Errors that are associated with an inadequate experimental dataset.

If one wants to make models more robust one needs to address all of the above issues. In the past scientists and engineers have concentrated on the first two aspects. The third aspect is mainly looked at by statisticians but has not had a large impact on the way automotive engineers use their models. The fourth aspect is used by the machine learning community to (a) help identify erroneous experimental/model data, and (b) carry out further analysis to identify the source and thus improve the second aspect.

The final aspect is relevant to every model during the optimization and validation, the scope of the experimental dataset. Examinations of model accuracy are a common research activity in the engine community, however these are often limited to a handful of experimental data points acquired during testing of an engine designed to meet the latest set of emission targets. The result is a limited dataset coupled with an application model which contains a large number of parameters such as CFD. Due to “overfitting” the final validation demonstrates an adequate model result compared to experiments. However, when these same model parameters are applied to other experimental data, large model errors are often reported. Indeed, any model should be optimized against whole datasets of engine data, engine types, etc. and model robustness properly assessed.

Conventionally, a metric of the accuracy of a model is often comparing the model result with the corresponding experimental data. Model accuracy is often loosely defined by assessing the deviation between these data and considering any experimental uncertainty. A metric to describe the robustness of the model should also be defined by addressing the uncertainties associated with the optimization of model parameters against experimental data, as such an uncertainty is associated with each parameter depending upon how well known it is or the confidence and thus ultimately on the final
model result. This will yield both a measure of the model accuracy as well as a measure of the robustness.

To apply these concepts to an automated model development process three major developments are required, (1) a consistent data model for storing and sharing experimental and model data, (2) a graphical interface to visualize the data/model inputs, (3) for application models to be extended to include parametric uncertainties, and (4) for the adopted optimization routine to include experimental and parametric uncertainty.

The sections which follow detail the methods employed in applying these concepts to the engine development process.

**A DATA MODEL: ENGINERDF**

The eXtensible Markup Language (XML) is used as the fundamental code of the engine Markup Language (*engineML*) [5]. The structure is fully extensible, an open standard and is platform independent making it potentially timeless and thus ideal for Process Informatics approaches. In addition, an XML schema can be adopted to ensure consistency between files created from multiple users and multiple programs which is important in creating standardization for large collaborative research activities such as engine development. The *engineML* data model is separated into General data (independent of operating point) and Case data (dependant on the operating point), each are then divided into Basic, Intake, Injection, Fuel, Cylinder and Exhaust for simple referencing. The extensibility of the structure means that if the database does not contain a component that the user requires, it can be added without compromising the structure of the infrastructure (the additional data would also be compatible with any legacy software). Table 1 shows data and metadata which are defined in an example *engineML* property entry. The design of a property entry is flexible enough to contain a wide range of engine data. A sample *engineML* property is presented in Fig. 2 according to the data structure shown in Table 1.

```xml
<property
  variable_name="Intake valve diameter"
  short_name="Int.Dial"
  detailed_description="Intake valve diameter"
  measurement_device="unknown"
  measurement_location="NA"
  data_type="constant"
  data_structure="point"
  unit="mm"
  unit_reference="none"
  unit_type="length">
  <value>50.0</value>
  <uncertainty>0.00</uncertainty>
</property>
```

**Figure 2: Example of engineML data model: cylinder property group**

<table>
<thead>
<tr>
<th>XML tag or name</th>
<th>XML syntax type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>engineml:variable name</td>
<td>xs:attribute</td>
<td>Name of data</td>
</tr>
<tr>
<td>engineml:short name</td>
<td>xs:attribute</td>
<td>Short name of data</td>
</tr>
<tr>
<td>engineml:detailed</td>
<td>xs:attribute</td>
<td>Description of the data</td>
</tr>
<tr>
<td>description</td>
<td></td>
<td></td>
</tr>
<tr>
<td>engineml:measurement</td>
<td>xs:attribute</td>
<td>Equipment used to take data</td>
</tr>
<tr>
<td>device</td>
<td></td>
<td></td>
</tr>
<tr>
<td>engineml:measurement</td>
<td>xs:attribute</td>
<td>Measurement location</td>
</tr>
<tr>
<td>location</td>
<td></td>
<td></td>
</tr>
<tr>
<td>engineml:unit</td>
<td>xs:attribute</td>
<td>Units (e.g. degrees)</td>
</tr>
<tr>
<td>engineml:unit reference</td>
<td>xs:attribute</td>
<td>Relative unit (e.g. bTDC or aTDC)</td>
</tr>
<tr>
<td>engineml:unit type</td>
<td>xs:attribute</td>
<td>Data type (e.g. a crank angle)</td>
</tr>
<tr>
<td>engineml:data structure</td>
<td>xs:attribute</td>
<td>Data structure</td>
</tr>
<tr>
<td>engineml:value</td>
<td>xs:element</td>
<td>Value of data</td>
</tr>
<tr>
<td>engineml:profile</td>
<td>xs:element</td>
<td>Profile of data (e.g. x,y)</td>
</tr>
<tr>
<td>engineml:uncertainty</td>
<td>xs:element</td>
<td>Uncertainty of data</td>
</tr>
</tbody>
</table>

For data to be useful in the long term, it is critical to properly define the apparatus and measurement devices as this ensures a comprehensive record of the experiment is held with the experimental measurements. Furthermore this information is of great importance for model developments which include error and uncertainty propagation as described in the example in this paper.

Often a large number of automatically measured data are categorized or grouped into sets of the same type and each data set is normally stored into separate XML files, i.e. *engineML*. However, in order to use the formatted XML data effectively, one must understand its representation or the corresponding schema. Naturally it is preferable to access these data without learning the full knowledge of an XML based representation, which can be done via the adoption of Semantic Web technologies, such as RDF (Resource Description Framework).

In this work, *engineML* data are transformed in such ways that each useful data resource is uniquely identified by URI (Uniform Resource Identification) and these are related to each other using a proper RDF statement as shown in Fig. 3. Firstly, in order to relate these data to one another, the semantic relationship of the data resource must be defined. The datatypes and relationships (predicate) for *engineRDF* are defined in Table 2 and Fig. 3 respectively. The URI pattern shown in Fig. 4 enables users to uniquely identify each resource whilst containing the full information about its higher level resources.
### Table 2: engineRDF datatype

<table>
<thead>
<tr>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>enml:EngineML</td>
<td>An RDF type which represents an engineML document.</td>
</tr>
<tr>
<td>enml:Case</td>
<td>An RDF type which represents a case of the engineML. The case can be general case data or operating-point data.</td>
</tr>
<tr>
<td>enml:GroupProperty</td>
<td>An RDF type which represents property group within the case, i.e. Basic, Intake, Injection, Fuel, Cylinder and Exhaust.</td>
</tr>
<tr>
<td>enml:Property</td>
<td>An RDF type which represents property in EngineML.</td>
</tr>
</tbody>
</table>

**Figure 3: URI of engineRDF datatype**

**Figure 4: The engineRDF predicate which relate the subject to the object**

The transformation of engineML into engineRDF turns legacy data into a web network of engine data which allows everyone, including computer resources, to understand these data through their relationships rather than their representation. Since each engineRDF contains unique information, this was considered more robust than equivalent XML-based approaches, and thus an improvement upon our original engineML structure [5]. In this work we have adopted a triplestore database engine, an example is openRDF-sesame which is a Java-based with support for RDF Schema inferencing and querying. It supports both local and remote through http protocol allowing data to be exchanged globally over the internet.

A data repository has been populated with a comprehensive set of state-of-the-art experimental data supplied by the Lund Institute of Technology [9–12], at present this data can be queried and visualized via the Cambridge University web pages [13] enabling engineers from all over the world to access these data for their own model development purposes.

### OPTIMIZATION

The optimization consists of two main stages. The first stage, parameter optimization, was done by evaluating the full model and was performed by running two algorithms sequentially. The first algorithm, which used Sobol sequences, was used to find a point close to a possible global minimum. The second algorithm, the Levenberg-Marquardt algorithm, was used to try to find the candidate for the global minimum close to the previously found point. The second stage consisted of parameter and model response uncertainty estimation done by evaluating linear response surfaces.

The parameter vector $x$ is defined in Eq. (1), where $E$ and $A$ are the vectors of engine parameter values and forward reaction rates that were varied. The individual components are as described in Table 5.

$$x = (E, A) \quad (1)$$

As the first step of the parameter optimization, data points in the K-dimensional parameter space were generated using Sobol [14] low discrepancy sequences. The model was evaluated at these points and the objective function $\Phi_1$, Eq. (2) determined.

$$\Phi_1 = \sum_{i=1}^{N} \sum_{j=1}^{M_i} \left( \frac{\eta_{ij} - \eta_{ij}(x)}{\sigma_{ij}^{exp}} \right)^2 \quad (2)$$

$N$ is the number of operating points and $M_i$ is the number of experimental data points from the $i^{th}$ operating point. The $\eta_{ij}^{exp}$ are the experimentally measured values; the $\sigma_{ij}^{exp}$ are the uncertainties associated with the experimental values and the $\eta_{ij}(x)$ are the model responses. The set of parameters $x_i^*$, defined in Eq. (3), that minimize the objective function $\Phi_1$ has been determined.

$$x_i^* = \arg \min_{x} \{ \Phi_1(x) \} \quad (3)$$
As the second step of parameter optimization the model has been optimized using the Levenberg-Marquardt algorithm [15] taking $\hat{x}_{0}$ as the starting point. A new set of parameters $\hat{x}_{2}^{*}$ have been determined that minimized $\Phi_1$ further.

As the final stage a linear response surface optimization has been performed around the point $\hat{x}_{2}^{*}$ and the uncertainties in the parameters and in the model response have been estimated. This methodology has been proposed by Sheen et al. [16] and used by Braumann et al. [17, 18] to optimize a granulation model and a system level soot model [5].

It is assumed that the free parameters $x$ are Gaussian distributed with a mean $x_0$ and a standard deviation $c$, see Eq. (4), where $\xi$ is a normally distributed random variable.

$$x = x_0 + c\xi$$ (4)

It is convenient to normalize the free parameters $x$ to $\tilde{x}$ and $c$ to $\tilde{c}$, where $\tilde{x}_k \in [-1,1]$ and $\tilde{c}_k \in [0,2]$, $k = 1,2,\ldots,K$. The model responses $\eta_k(x)$ have been approximated by linear response surfaces $\mu_k(\tilde{x})$, defined in Eq. (5) where the $i$ and $j$ indices have been suppressed, around the point $\hat{x}_{2}^{*}$.

$$\eta(x) \approx \mu(\tilde{x}) = \beta_0 + \sum_{k=1}^{K} \beta_k \tilde{x}_k$$ (5)

The coefficients $\beta_0$ and $\beta_k$ have been calculated using finite differences, Eqs. (6) and (7) where $e_k$ are unit vectors in the $k$th dimension.

$$\beta_0 = \eta(\hat{x}_{2}^{*})$$ (6)

$$\beta_k = \frac{\eta(\hat{x}_{2}^{*} + e_k) - \eta(\hat{x}_{2}^{*} - e_k)}{2}$$ (7)

The variance $\sigma^2(\tilde{c})$ of the model response can be calculated using Eq. (8).

$$\sigma^2(\tilde{c}) = \sum_{k=1}^{K} (\beta_k \tilde{c}_k)^2$$ (8)

A different objective function, Eq. (9), based on the principle of moment matching was used to include the optimization of parametric uncertainties to the experimental error.

$$\Phi_2(\hat{x}_{0}, \tilde{c}) = \sum_{i=1}^{N} \sum_{j=1}^{M} \left( \frac{\eta_{ij}^{\text{exp}} - \mu_{ij}(\bar{x}_{0})}{\sigma_{ij}^{\text{exp}}} \right)^2 + \left( \frac{\sigma_{ij}^{\text{exp}} - \sigma_{ij}(\tilde{c})}{\sigma_{ij}^{\text{exp}}} \right)^2$$ (9)

Minimizing $\Phi_2(\hat{x}_{0}, \tilde{c})$ using the Levenberg-Marquardt routine leads to the optimal set of model parameters, $\hat{x}_{0}$, and their associated errors, $\tilde{c}$, see Eq. (10).

$$(\hat{x}_{0}, \tilde{c}) = \arg \min_{\hat{x}_{0}, \tilde{c}} \{ \Phi_2(\hat{x}_{0}, \tilde{c}) \}$$ (10)

**APPLICATION TO AN IN-CYLINDER IC ENGINE MODEL**

The methodology outlined above was applied to optimize an engine combustion model against the engine experimental data stored in the repository. The underlying engine model was the Stochastic Reactor Model (SRM) [19], in the past the SRM has been successfully employed in a number of studies of port fuel injected HCCI combustion [20], surrogate fuel blends [21], single early direct injection HCCI [22], dual injection HCCI [23], multi-cycle transient simulation and control [24], soot formation [25], and has been coupled to the Computational Fluid Dynamics (CFD) code KIVA [26].

Further details of the model can be obtained from these studies however in summary, the SRM is able to simulate the internal mixing and chemical processes in the combustion chamber, along with heat transfer to the cylinder walls. It is the preferred tool for modelling HCCI combustion than conventional fully homogeneous models as it allows for temperature and mixture inhomogeneities which are important in smoothing out the observed heat release rates.

The fuel oxidation chemical kinetic models have a large number of parameters, often in the thousands [7] requiring serious computational cost, however given that this is an example of the methodology and to keep it industrially relevant, a skeletal iso-octane/n-heptane oxidation mechanism employed by the automotive industry containing 33 species and 38 reactions was adopted [27]. Importantly, the infrastructure of carrying out the optimization with the employed model is identical to that of a semi-detail, detailed or comprehensive chemical kinetic mechanism and engine model.

The SRM code was manipulated to read directly from the engineML files, enabling rapid and consistent model initialization directly from experimental data. However, not all data was immediately accessible for example, the SRM is an in-cylinder model and thus only solved between intake valve closure and exhaust valve opening, however in practice, measurements for initial mixture pressure and temperature at intake valve closure often come via estimation based on the
measurements obtained in the intake manifold. Here we imposed a simple engine breathing model, via a single multiplier upon the manifold pressure and temperature measurements to generate a relevant initial mixture state.

Details of the data adopted for the optimization to set up the initial parameters are outlined in Tables 3 and 4. The employed data were those relevant to pure iso-octane fuel blends, with the engine operated in HCCI operating mode. In this example, seven operating points have been included however, the infrastructure enables as many as required to be adopted.

Listed in Table 5 are a list of the initial model parameters and their corresponding one standard deviation uncertainty for the model parameters. Initial engine parameters were obtained directly from the experimental data stored in the engineRDF webportal, with typical expected uncertainty bands employed. The SRM engine model was adopted with 100 stochastic particles and with an initial stochastic heat transfer parameter of 2000. The value of 2000 has been found to give good results in previous studies [19-26]. The initial reaction rates adopted in the fuel model were obtained from [27], however an uncertainty was only employed for the pre-exponential term, $A$ with the other fuel model parameters assumed to have no uncertainty. An optimization was carried out by varying these parameters minimizing the objective function Eq. (2), within their uncertainty bounds with respect to the ignition delay time, peak pressure and the in-cylinder pressure profile (at 5 bar increments), experimental errors of 2 CAD, 5% and 5% were assumed respectively. The first objective function, Eq. (2), was calculated from the model response at over 300 points during the Sobol sequencing step taking 6 days, the Levenberg-Marquardt optimization evaluating the model response took 2 days and the response surface was determined within 2 days on a standard desktop PC.

RESULTS

The final results of the optimization are presented in Table 5. A new set of model parameters were found and a set of model uncertainties were obtained including a reduction of the uncertainties on the reaction rates, here shown as a mean value.

The result of the minimization of first objective function yielded a new set of model parameters which best fitted these experimental data. A summary of the values of these parameters are presented in Table 5. In all cases compression and expansion were computed well, with the main heat release generally computed a little early, however there is a significant improvement compared to the results obtained with the original set of model parameters. Neither Cases 6 nor 7, which were at a higher engine speed of 995 RPM, achieved the peak pressure though.

The corresponding initial model uncertainty is also presented as the dark grey shaded regions, whilst these are often based on the initial values, these can often be arbitrary as shown in Table 5, however they have been constrained to what might be expected as a reasonable deviation. During compression, the errors associated with the initial pressure, temperature and compression ratio are smaller than the thickness of the line, however once the heat release begins, typically at around 10° aTDC, the influence of the chemistry becomes more influential and thus the uncertainty upon those parameters contained in the fuel oxidation model become more significant. In all cases, the largest uncertainties are noted in those regimes where the chemical kinetics are most active. The reduction of the uncertainty in the compression ratio multiplier to zero is obviously not physical possible causes will be discussed in the next section.

The result of the minimization of the second objective function yields set of uncertainties for the optimised model, which are presented in Fig. 5. Firstly the uncertainty of the model during the compression phase reduced due to the reduction of the uncertainty on the engine parameters E1 to E4 shown in Table 5. However the main reduction in the uncertainty was in the regime dominated by the chemical kinetic parameters, $R_1$ to $R_3$. Here the uncertainty was reduced from up to 20 bar to around 3 bar, typically this uncertainty was largest at around TDC and then reduced during the expansion stroke as the influence of the kinetics reduced and those associated with thermodynamic expansion began to dominate.

DISCUSSION

The general improvement in the models ability to fit the data demonstrates the potential of this methodology. The fact that the simulations of neither case 6 nor 7 achieved the peak pressure though demonstrates the potential of this methodology.
pressure is likely to be because the objective function was dominated by the other 5 cases, all at the lower speed. This suggests that the application model may need to be improved to better account for engine speed. One way might be to increase the turbulent mixing time which is well known to have a linear relationship with mean piston speed $[28]$ proportionally with engine speed. The inclusion of more cases with a wider range of engine speeds will help to examine this problem further.

It has been recognised that the inclusion of all of the reaction rate pre-exponents in the optimisation procedure is not necessarily the most productive strategy. A more careful selection of the parameters being optimised will be carried out in future work along with better estimations of the initial uncertainties associated with them. Also if they are to be varied they should be normalised using a logarithmic transform not a linear one as they can in principle carry an uncertainty of orders of magnitude.
The use of linear response surfaces to propagate uncertainties through the model may be responsible for the uncertainty in the compression ratio multiplier tending to zero. The use of higher order response surfaces or even more complex methods of uncertainty propagation, such as Bayesian methods [29], may correct this and will be investigated in the future.

Despite any short comings of the specific optimisation procedure used in this paper the example shown here demonstrates the power of automating data storage, retrieval and model optimization to reduce the man-hours required to tune models to large experimental datasets. Therefore making better use of the knowledge contained within them than could be made by manual tuning. The calculation of uncertainties in the optimised values opens up the possibility to suggest the next set of experiments that should be performed as well as which aspects of the model may need improvement. This constant feedback between experiment and model should speed up the development of more robust models, and therefore speed up engine development, by enabling senior engineers to distribute resources more efficiently.

The web portal enables all engineers to have access to a comprehensive engine data repository. Researchers are encouraged to upload their experimental data to the repository with tools being developed to make it possible to store experimental data in the EngineML format either directly form the experiment or during the post processing stage. This ensures that these data are stored securely in a consistent format for future research activities and facilitating knowledge transfer between research groups. The most significant advantage will be for the modelling community as a whole who now have, for the first time a data repository of well marked-up experimental data for automated model operation and optimization.

The implementation of the methodology in this paper is currently being improved so that setting up the optimisation procedure is much easier whilst giving much more flexibility to choose the experimental data, optimisation algorithms and application models that are used.

CONCLUSIONS

The engineML data model used for storing engine based experimental data has been extended to an engineRDF format for improved searching and querying. A web portal for storage, searching and visualization of engine data has been developed for engineers to extend a repository of shared engine data for improved knowledge transfer and automated model development.

Using the described methodology, an engine model containing more than forty input parameters has been systematically optimized against engine experimental data held in the data repository. Adoption of the methodology has demonstrated that engine model parametric uncertainties can be reduced with respect to experiments and highlighted those aspects of the model which may require further investigation.

REFERENCES