Programme Area: Bioenergy

Project: Biomass to Power with CCS

Title: Model Data Requirements Specification and Model Template Release

Abstract:

These two documents were the first deliverables under WP3 (Parameterised Model Development). They outline the Model Data Requirements Specification and the Model Template Release for the eight technologies modelled as part of the project. They were used in the development of the Bioenergy Value Chain Model (BVCM).

Context:

The Biomass to Power with CCS Phase 1 project consisted of four work packages: WP1: Landscape review of current developments; WP2: High Level Engineering Study (down-selecting from 24 to 8 Biomass to Power with CCS technologies); WP3: Parameterised Sub-System Models development; and WP4: Technology benchmarking and recommendation report. Reports generally follow this coding. We would suggest that you do not read any of the earlier deliverables in isolation as some assumptions in the reports were shown to be invalid. We would recommend that you read the project executive summaries as they provide a good summary of the overall conclusions. This work demonstrated the potential value of Biomass to Power with CCS technologies as a family, but it was clear at the time of the project, that the individual technologies were insufficiently mature to be able to ‘pick a winner’, due to the uncertainties around cost and performance associated with lower Technology Readiness Levels (TRLs).

Biomass to Power with CCS Project

TESBiC: Techno-Economic Study of Biomass to power with CCS

PM 03 – D3.1 and D3.2

Internal Deliverable Report

on

D3.1: Model data requirement specification and standardisation

D3.2: Beta model template

02/11/11

V0.1

Title	Internal Deliverable on Model Data Specification and Beta Model Template
Client	Energy Technologies Institute LLP (ETI)
Reference	BwCCS PM03 D2.1 & D3.2
Date	02 November 2011
Version	0.1
Authors	The TESBIC consortium:

cmcl innovations

Doosan Babcock

Drax

EDF

E4tech

Imperial College London

University of Cambridge

University of Leeds

Distribution

This report has been prepared by the TESBIC consortium under contract to the ETI.

CONTENTS

1. INTRODUCTION

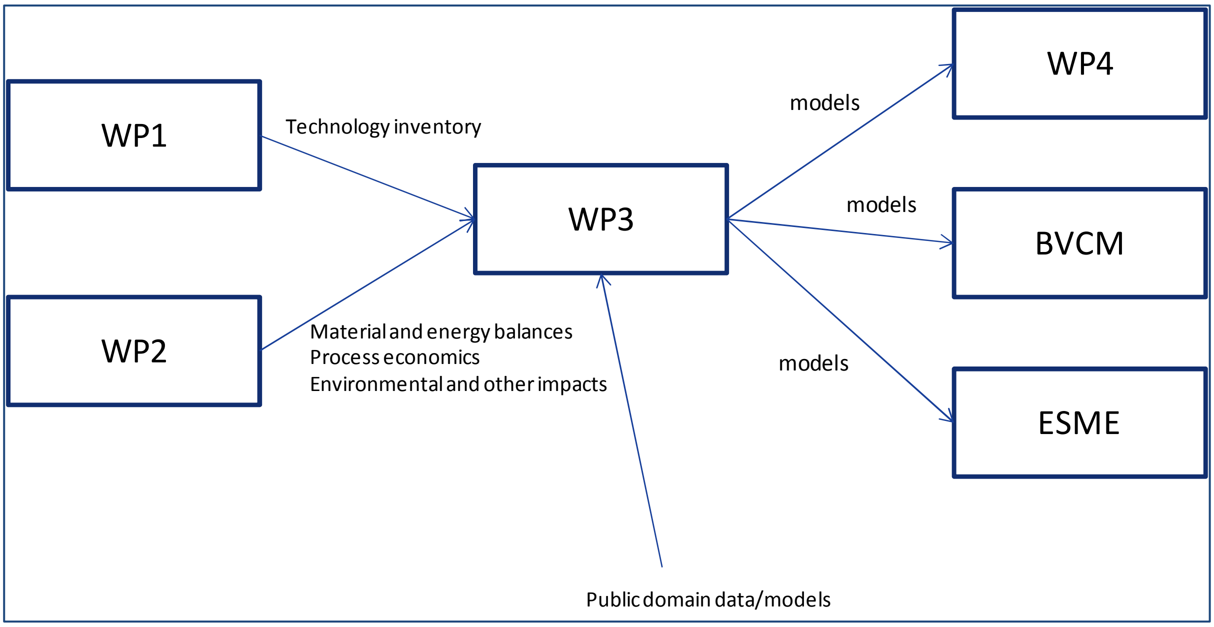
The **T**echno-**e**conomic **S**tudy of **Bi**omass to Power with **C**CS (TESBIC) project, which has been commissioned by ETI, is concerned with the performance of an overview techno-economic assessment of the current and potential future approaches to the combination of technologies which involve the generation of electricity from biomass materials, and those which involve carbon dioxide capture. The present document forms the internal deliverable within work package, WP3; and it covers the work on data standardisation and model beta release, carried out under D3.1 and D3.2.

2. WP3 OVERVIEW

The main **objectives** of WP3 are:

- Systematising and encapsulating knowledge from WP1 and WP2
- Generating models for:
    - ETI BVCM project
    - ETI ESME model
- Supporting integrated assessment of Biomass CCS
    - Feeding into WP4

WP3 is intimately connected with the rest of the programme, as illustrated in **Error!**



Reference source not found..

Figure 1: Work packages, WP and work-flow.

The key **internal deliverables** in WP3 are the following:

- D3.1 – model data and standards definition [04.11.2011]
- D3.2 – model beta release [04.11.2011]

The **critical success factors** for WP3 are:

- Good integration with WP2:
    - We need:
        - Material and energy balances
            - Including input and fate of fuel/biomass contaminants
            - Including parasitic energy requirements
        - Economics (CAPEX, OPEX) at different scales
        - Where possible, the dependence of the above on degree of capture
        - Design details of major equipment items
- Good communications with BVCM and ESME teams

3. OVERALL MODELLING APPROACH

Having reviewed the objectives, data in the literature and seen the BVCM and ESME modelling environments, we have agreed upon a meta-modelling approach to building the simple systems models for TESBIC. This approach is illustrated in

Figure 2: Overview of metamodelling approach.

and outlined below.

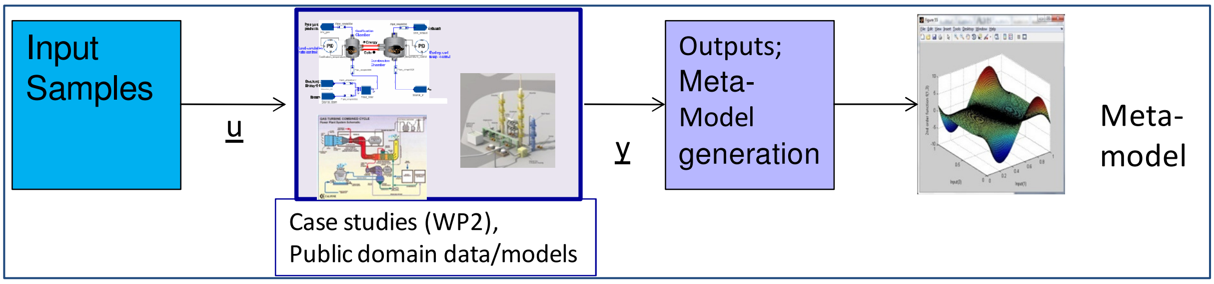


Figure 2: Overview of metamodelling approach.

Overview of modelling approach:

- Once requirements, inputs and outputs are agreed:
- Define standard units and reporting structures for these, including confidence measures for data
- Define metamodel structure to be used
- Identify sensible ranges for input variables
- Identify technology inventory (5+3, but with potential sub-processes)
- Identify data sources for outputs associated with inputs and fix data standards
- Develop and publish meta-model standard (piecewise linear recommended)
- Use WP2 models and data and other data to generate outputs from a sampled range of inputs
    - Prototype meta-model fitting (e.g. through least-squares optimisation)
    - Review meta-model approach and finalise model by model development
    - Produce model library and documentation
- Agree with ETI model storage and transfer protocol

4. DATA STANDARDISATION

Within the WP3 of the TESBIC project, data from various sources, i.e. published/literature, other R&D projects, high-level engineering case studies studies (such as WP2) will be utilised to parameterise the sub-system models developed. It is important that a systematic standardisation of data is employed, as this ensures that data collected from multiple sources are comparable in terms of actual content, as well as units. It can also impose certain values to be present (i.e. no missing data), as well as ensuring that the values given are sensible (i.e. lie in preset ranges). Within TESBIC, this process is achieved via the usage of XML (eXtensible Markup Language).

Example data: For the sake of simplicity of exposition, we shall consider possible data from various biomass plants of a certain type. This data shall be composed of:

- Inputs (4-dimensional vector x)
    - Nameplate capacity (MWe)
    - Operating capacity (MWe)
    - Co-firing (%)
    - Carbon capture extent (%)
- Outputs (7-dimensional vector $y = (y_1, y_2, y_3, y_4, y_5, y_6, y_7)^T$)
    - Capital cost (k £/MWe)
    - Non-fuel operating cost (k £/MWhe)
    - Generation efficiency (%)
    - CO_2 emissions (kg CO_2/MWhe)
    - SO_2 emissions (kg SO_2/MWhe)
    - NO_x emissions (kg NO_x/MWhe)
    - Solvent losses (kg amine/MWhe)

We suppose that the observed inputs y_j have noise associated with it (known or unknown).

XML: XML is a widely used markup language which provides a way of encoding data into machine-readable format. It has strong similarities to HTML in that it is tag-based, and thus some limited "meanings" and semantics can be attached to values. An example of this is given in Figure 3 (a snippet from our test XML file):

```
<CoFiring unit="k GBP/MWe">

        <NormalDist>

                <value>960</value>   <!--obligatory -->

                <sd>30</sd>            <!--optional -->

        </NormalDist>

</ CoFiring >
```

Figure 3: A sample XML representation.

Here, there is a tag named “CoFiring” which has an associated attribute of “unit”. Furthermore, this tag has a ‘value’ of a distribution – in this case the normal distribution. The normal distribution tag “NormalDist” itself must be specified by its mean value and standard deviation – the corresponding tag names being “value” and “sd” respectively. Such an XML structure, however, is open to misuse – it is desirable to have some constraints on the values given. Some examples are:

- The “unit” attribute must be exactly equal to the string “%” for the “CoFiring” tag.
- The value of “CoFiring” must be one of the available distribution tags specified in the XML schema – at the moment, these are “NormalDist” and “UniformDist”.
- For “NormalDist”, the “value” must be present, but the “sd” tag can be omitted if the information is not available.
- The “sd” tag within the “NormalDist” must be positive.
- For the “CoFiring”, the “value” tag within the “NormalDist” tag must lie in 0 – 100.

To implement limited constraints such as those just mentioned, we use an XML schema. This outlines the tags that may/may not exist, their values that they may/may not take, as well as child tags that they may/may not have. A snippet of our sample XML schema which implements some of the constraints above is given in Figure 4.

```
<xs:complexType name="CoFiringType">

        <xs:choice>

                <xs:element name="NormalDist"   type="NormalDist_Percentage_Type" />

                <xs:element name="UniformDist"   type="UniformDist_Percentage_Type" />

        </xs:choice>

        <xs:attribute name="unit"   type="xs:string"   use="required"   fixed="%" />

</xs:complexType>
```

Figure 4: A sample XML schema representation.

Here, the XML schema has defined a *datatype* – all the information held within the “CoFiring” tag earlier is described here – more explicitly:

- The “CoFiring” tag must have the attribute “unit” which must be given the value “%” which is of string type.
- The value held within the “CoFiring” tag must be exactly one of “NormalDist” or “UniformDist” to specify the distribution of the datum.
- If “NormalDist” is chosen, the specification of any values within this tag must be compatible with the notion that the CoFiring datum is a percentage value (its mean must be between 0 and 100, and the standard deviation is positive if specified). This

specification is held in the definition of another complex type called "NormalDist_Percentage_Type", which is defined elsewhere in the XML schema.

This is but a small sample of the types of restrictions that can be imposed using XML schema. Without giving too much detail, we flesh out the main structure of input XML files as imposed by our XML schema for our example (a more pictorial version of this is given in Figure 5):

- The outermost tag is "root". There must be exactly one of these.
- The next outermost tag is "dataset". This holds all data. There must be exactly one of these.
- The "dataset" holds multiple "datapoint" tags.
- Each "datapoint" tag holds a single (x, y), as well as some attribute data. This is ensured by requiring each "datapoint" tag to contain exactly one "covariate" tag (for the inputs x) and one "response" tag (for the outputs y). The following attributes must be given:
    - Process_idnum – number identifying the process type.
    - Process_name – name of process under which the datapoint was obtained.
    - Datapoint_idnum – number identifying the datapoint.
    - Datapoint_type – whether it's a base or a delta datapoint.
    - Datapoint_source – source of data.
- Each "covariate" tag must contain exactly one of each of the input tags ("NameplateCapacity", "OperatingCapacity", "CoFiring", "CarbonCaptureExtent"). Each of these tags are specified basically as in Figure 1 – the co-firing input variable is specified within the "CoFiring" tag and the Nameplate capacity input variable is specified within the "NameplateCapacity" tag etc. The only difference between these tags are the associated units attributes (e.g. "NameplateCapacity" must have an attribute of "units" which must be exactly equal to the string "MWe").
- Each "response" tag is specified in exactly the same fashion as the "covariate" tags.

Just to elaborate a little on the specification of uncertainties:

- The "NormalDist" must have a "value" tag, but any "sd" tag is optional (but if it is given, the value given within the "sd" tag must be positive). Here the "value" tag gives the mean of the distribution and the "sd" tag gives the standard deviation.
- The "UniformDist" tag is specified in exactly the same way, except that it has the "uncertainty" tag rather than the "sd" tag (has the same conditions as "sd"). Here the "value" is the mean of the distribution, and "uncertainty" gives the half-width of the distribution.

```
<root>
	<dataset>
		<datapoint>
			<covariate>
				<CoFiring>
					…
				</CoFiring>
				…
			</covariate>
			<response>
				<Capital_cost>
					…
				</Capital_cost>
				…
			</response>
		</datapoint>
	</dataset>
</root>
```

Figure 5: Main structure of input XML file.

5. METAMODEL STRUCTURE

Based on the requirements of the system, two types of meta-models were shortlisted and reviewed:

- Simple piecewise linear models of the form: $\underline{y} = \mathbf{A}\,\underline{x} + \underline{b}$, where $\underline{y}$ is a vector of output variables, $\underline{x}$ is a vector of input variables and $\mathbf{A}$ and $\underline{b}$ are a parameter matrix and vector respectively

- "Base+delta" linear models of the form: $\underline{y} = \underline{y}(\underline{x}_b) + \mathbf{A}(\underline{x} - \underline{x}_b)$ where $\underline{y}$ is the vector of output variables, $\underline{x}$ is a vector of input variables, $\underline{y}(\underline{x}_b)$ is the value of the outputs for a specific ("base") set of values of $\underline{x}$ ($\underline{x}_b$) and **A** is a parameter matrix.

We have considered the merits of both and believe that the base+delta approach will be more suited to the TESBIC application. It proves to be more stable and provides more certainty being based on a set of well grounded base values. It has now become what is considered best practice in refinery and petrochemical plant modelling, where base+delta models of unit processes are believed to be more accurate than simple discrete yield models. This model is suitable for processes where the input factors do not exhibit strong interactions (e.g. degree of co-firing and degree of carbon capture do not have strong multiplicative or other non-linear interactions on outputs), which we believe to be the case for our systems.

Metamodel inputs and outputs

The inputs and outputs are closely related to the system KPIs being investigated in WP2. The inputs ($\underline{x}$) are:

1) Process nameplate capacity (to capture scale-dependent costs/efficiencies) range – with a base case and then +/- a capacity increment on the base case

2) Actual operating capacity (e.g. % of nameplate, fuel rate or other...) – a study at nominal capacity and a study at turndown to capture turndown-dependent performance

3) Co-firing % (a base value, e.g. 20% and a "high" value e.g. 50%)

4) Extent of carbon capture (a base value of e.g. 90% and a high value of e.g. 95%)

The outputs ($\underline{y}$) are:

1) Capital cost per unit (within the nameplate capacity range) – with a commentary on potential improvements in the future

2) Non-fuel operating cost per unit (i.e. O&M)

3) Efficiency of generation (power per fuel rate) – with a commentary on potential improvements in the future

4) Other output flows [from the material balance]

   a) CO2 – overall abatement and net emissions

   b) Fates and flows of other major outputs/pollutants (g, l, s), including spent solvent, amine salts, spent sorbents etc

   c) Where available, excess heat potentially available per major temperature interval

5) Other key, non-main fuel input requirements

Note that these will be to some extent technology dependent. These data all require standards to ensure consistency across technologies, as described in Section 4 above.

Sample model view

A sample model has been developed in Microsoft ExcelTM. The models will be delivered to the ETI in this format. A screenshot of a sample model is shown in

Figure with some explanations.

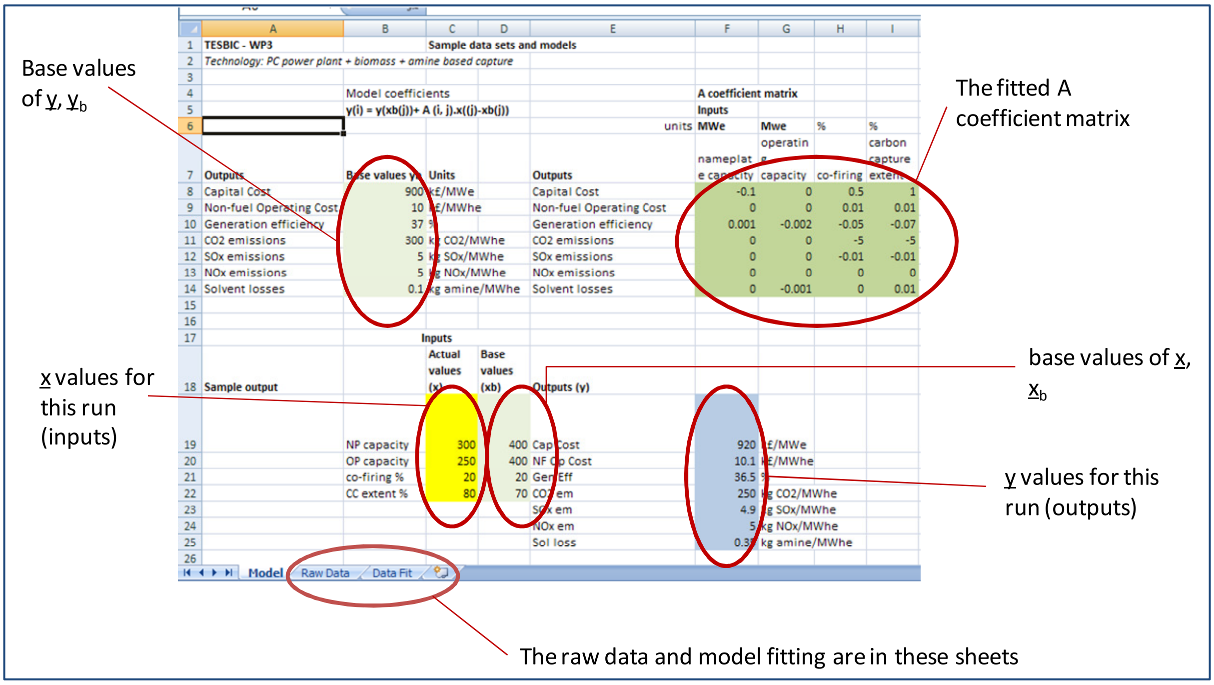


Figure 6: Snapshot of model sheet.

6. NEXT STEPS

The next steps are to analyse the data coming through from WP2 (having specified the data requirements for WP3 during the WP2 specification these should be approaching adequate). These will be organised using the standards of WP3.1. The requirements for additional data will then be assessed and additional data generated or sought from the public domain as appropriate.

Once there is enough data for statistical adequacy, the base+delta model parameters will be estimated and the metamodels developed and tested against data and detailed models. The metamodels will then be fully documented and transferred in appropriate formats to the ESME and BVCM teams, as well as to the ETI in the format in Figure 6.