IO Tutorial

Table of Contents

First, initialize the Cobra Toolbox and switch to the tutorial folder

```
initCobraToolbox
cd(fileparts(which('tutorial_ReadModel.mlx')));
```

1. Reading a model

1.1 Available input formats

The COBRA Toolbox supports the use of models in multiple formats. Internally, it uses a simple MATLAB struct with fields defined in the Documentation. These models are commonly provided in a .mat file.

There are additional model formats for which io functions exist in the Toolbox. These incluse

- Models in SBML Format
- Models in SimPheny format
- Models in an Excel Format

1.1.1 Format Specifications

Matlab save files:

The format of a model struct provided in a .mat file has to stick to the rules defined in the Documentation.

SBML Format

The COBRA Toolbox currently supports SBML models provided as Level 3 version 1 (as defined here) and has legacy support for older versions of SBML. It also supports the Level 3 FBC package (both in version 1 and version 2).

The Toolbox will use the provided SBML IDs as IDs for the respective elements of the model structure, and use the name fields as names. It is assumed (but not necessary), that metabolite IDs start with a "M_", reaction ids start with a "R_" , gene ids start with a "G_" and compartment ids start with a "C_".

This is due to the limitation on Identifiers in SBML and those starting sequences will be removed if they are consistently present in the model. Since metabolites in the COBRA Toolbox model struct are commonly provided with a metabolite ID followed by a compartment identifier e.g. ('ala_L[c]'), and brackets are illegal characters for an SBML id, it is assumed, that if all non boundary species have a trailing compartment identifier preceded by a underscore (e.g. SBML ID: M_ala_L_c) those identifiers should be converted to model compartment identifiers.

The Toolbox has a legacy support for the NOTE Fields defined in Schellenberger et al, Nature Protocols, 2011, but it is suggested to instead use annotations whenever possible. In general, if a fbc-package field and a NOTES field is present, the fbc-package value will be used (e.g. CHARGE for metabolites, or GENE_ASSOCIATION for reactions). The same applies to annotations, i.e. if there is an annotation for an EC number, the Notes field EC Number will be ignored. However, the charge field in SBML Level 2 will be overwritten by the Notes field definitions.

SimPheny Format

SimPheny models provide their models in 3 or 4 files (4 if GPR rules are provided). The model identifiers will be used as presented in the SimPheny files.

Excel Format

Excel files adhereing to the COBRA xls specifications listed in the Documentation. Parseable excel models contain two sheets for reactions and metabolites respectively.

1.2. Example

The most straightforward way to import a model into the Toolbox is to use the readCbModel function. To e.g. load a model from a mat file, you can simply use the filename (with or without file extension).

```
fileName = 'Acidaminococcus_intestini_RyC_MR95'
```

```
fileName = Acidaminococcus_intestini_RyC_MR95
```

```
model = readCbModel(fileName);
```

```
Error using readCbModel (line 217)
There were no valid models in the mat file.\n Please load the model manually via ' load
Acidaminococcus_intestini_RyC_MR95.mat' and check it with verifyModel() to validate it
```

readCbModel assumes, that .mat files are a matlab save file, .xml files are models in SBML format, .sto are SimPheny models, and .xls or .xlsx are models in Excel format.

You can also call the function without a fileName to get a file selection dialog

```
model = readCbModel();
```

The model loaded in this way can directly be used with COBRA ToolBox functions. You can view the data stored in the model by e.g. using

```
open model
```

2. Writing a model

The COBRA Toolbox offers several output formats described below.

2.1 Output formats

Matlab .mat files

This format is simply the model being saved as a matlab save file, and is the default save method. It has the advantage of lossless data storage even for model specific fields not supported by the Toolbox.

SBML format (.xml)

The systems biology markup language (SBML) is a very common format to store biological models. The COBRA Toolbox allows generation of models using Level 3 Version 1 and uses the fbc-package extension to encode constraint based properties. This is the format that is recommended for publication, as it can be used by many different tools and allows the best use of the model.

Excel format

Historically, models were often exchanged using excel files, and this is still in use today. Some users prefer to get an overview over a model using Excel, since it provides a general overview at a first glance. The toolbox offers an Excel export using the format described in the Documentation (see above).

Text Format

Finally the toolbox offers a simple textual export, which is essentially a tab separated file containing the reactions with their reaction formulas along with the associated GPRs, but no further information. This format only uses the required fields and will ignore any optional fields.

2.2 Example

To write files, use the writeCbModel function. The function can be called directly with a model.

```
load('Acidaminococcus_intestini_RyC_MR95.mat')
writeCbModel(model)
```

```
Error: File: writeCbModel.m Line: 346 Column: 1
The function "writeCbModel" was closed with an 'end', but at least one other function
definition was not. To avoid confusion when using nested functions, it is illegal to use
both conventions in the same file.
```

This will call a file selection dialog, which allows the selection of a filename and, depending on the selected format from the dropdown, the output will be generated.

It is also possible to specify the format and filename explicitly:

```
writeCbModel(model,'SBML','Acidaminococcus.xml')
```

which will write the model to the file Acidaminococcus.xml.

If no fileName is provided, a popup will ask you to provide a fileName with the specified format.

```
writeCbModel(model,'text')
```

The available format options are:

- 'mat' - for a matlab save file
- 'sbml' - for a SBML model
- 'xls' - for a model in Excel format
- 'text' - for a textual export.