

Installing and Running MCMAS

Natasha Alechina, Kristina Gogoladze and Brian Logan

1 Introduction

The coursework for INFOMLSAI uses the model checker MCMAS developed by the Verification of Autonomous Systems group at Imperial College. There are several versions of MCMAS, and the aim of this document is to help you choose which is most appropriate for you.

2 Downloading

As of April 2022, the current version is 1.3.0, which can be downloaded from:

<https://vas.doc.ic.ac.uk/software/mcmas/>

Precompiled binaries are available for Linux (32 and 64 bit). However, if you are using Windows or macOS, you will have to build it from source (see below). When you click on a download link, you will be prompted to register. This is simply to track the use of the software—your name and email address will not be used for anything else.

There are three issues with version 1.3.0:

1. it is “command line only”—the GUI provided with previous versions has not been updated and is does not form part of the download;
2. it does not compile on some older OSs—more precisely, you will get an error if your version of `bison` is earlier than 3.5, as the support for LTL, LDL and CTL* in version 1.3.0 assumes `bison` 3.5; and
3. there are some bugs which can result in segmentation faults on some platforms.

If you would like to use the GUI, you can download version 1.1.0 from Blackboard. Version 1.1.0 supports all the logics used in the model-checking courseworks. It also has the advantage that it does not assume `bison` 3.5, and so builds without problems on a wider range of OSs. If you would prefer to use 1.3.0, we can provide a (partially) bugfixed version which compiles with `bison` 2.3 — contact us for details. The rest of this document assumes version 1.1.0; the procedure to install 1.3.0 is very similar, but in this case Sections 5 and 7 are not relevant.

3 Installing on macOS¹

- download file `mcmas-download.tgz` in the folder in which you want to install MCMAS, e.g., `myfolder`
- in a Terminal window, `cd` to the folder containing the MCMAS download, e.g., `myfolder`, and unpack the downloaded file `mcmas-download.tgz` using the command “`tar xvfz mcmas-download.tgz`”
- in the Terminal window, `cd` to `myfolder/mcmas-1.1.0`, and type “`make`”
- this should build the MCMAS executable `mcmas` (note: this may take several minutes, and it’s safe to ignore any compiler warnings)
- note also: if you do *not* have the Apple Developer Command Line Tools installed, you will be prompted to download them on typing “`make`” — when the download completes, you should be able to proceed as above

4 Installing on Windows

- download `cygwin`, and the packages `g++` (version 11.2.0-1), `flex` (version 2.6.4-2), `bison` (version 3.8.2-1) and `make` (version 4.3-1) from:

<http://www.cygwin.com/>

Other versions of the packages may work, but have not been tested. You may find it helpful to use the Search field to search for the package name in the list of packages (see Figure 1).

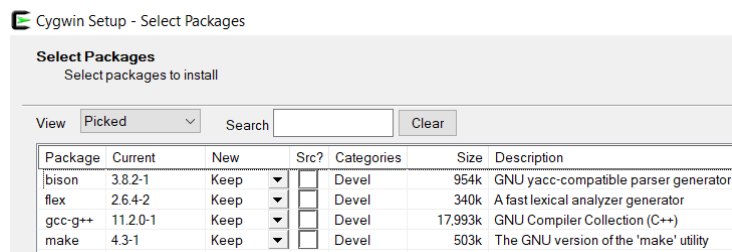


Figure 1: Required Cygwin packages

- download file `mcmas-download.tgz` in the folder in which you want to install MCMAS, e.g., `myfolder`

¹Building from source on Linux is similar to macOS.

- in a cygwin shell window, `cd` to the folder containing the MCMAS download, e.g., `myfolder`, and unpack the downloaded file `mcmas-download.tgz` using the command “`tar xvfz mcmas-download.tgz`”
- in the cygwin shell window, `cd` to `myfolder/mcmas-1.1.0`, and type “`make`”
- this should build the MCMAS executable `mcmas` (note: this may take several minutes, and it’s safe to ignore any compiler warnings)

5 Installing the GUI

There is a GUI for version 1.1.0 that provides syntax checking and highlighting of `ispl` files, support for interactive simulation of models, and graphical visualisation of counterexamples/witnesses.² The GUI is a plugin for the Eclipse IDE, which runs on Linux, macOS and Windows. (Note: other IDEs are *not* supported.) Eclipse can be downloaded from:

<https://www.eclipse.org/downloads/>

Eclipse is a Java application, but the download includes a JRE for macOS, Windows and Linux as appropriate (see the Eclipse documentation for details).

The plugin uses the `dot` program from the Graphviz package for visualisation.³ Graphviz is available from

<https://graphviz.org>

The GUI has a couple of limitations. Only the results for the first 10 properties are displayed in the Verification Results pane: if your ISPL file defines more than 10 properties, you will need to run `mcmas` from the command line. Also, visualisation is only possible for fairly small counterexamples and witnesses: if a counterexample or witness has tens of states, MCMAS will correctly display the textual form, but no visualisation is displayed, as `dot` is incapable of rendering the graph. The use of `dot` is optional. If you can’t or don’t want to install Graphviz, the other features of the plugin will still work; however there will be no visualisations.

To install the GUI, copy the file `org.mcmas.ui.1.1.0.jar` from the `mcmas-1.1.0` folder to the Eclipse dropins folder. See the Eclipse documentation for the location of the dropins folder for your OS. For example, on macOS, the path to the dropins folder is:

`/Applications/Eclipse.app/Contents/Eclipse/dropins/`

Exit eclipse if it running. In a shell window, `cd` to the folder in which eclipse is installed and type the command “`eclipse.exe -initialize`” (on Windows) or “`eclipse -initialize`” (on Linux/macOS). When eclipse is restarted, it should load the jar file.

²Counterexamples and witnesses will be explained in the lectures.

³Note for Eclipse users: there are better ways of visualising graphs in Eclipse, e.g., GEF-DOT; however, these are not supported by the plugin.

The GUI needs to know the locations of the `mcmas` executable, `dot` executable, and (on Windows) of `cygwin` executable. These are specified in the MCMAS group in the Eclipse preferences. Note that you need to specify the paths to the *folders* (directories) containing `mcmas`, `dot`, and `cygwin` executables not the paths to the executables themselves. Figure 2 shows how the eclipse preferences should look on Linux, and Figure 3 shows how they should look on Windows. (The paths to the MCMAS Directory, Dot Directory and Cygwin Directory should be those for your system; the ones in the figures are just examples.)

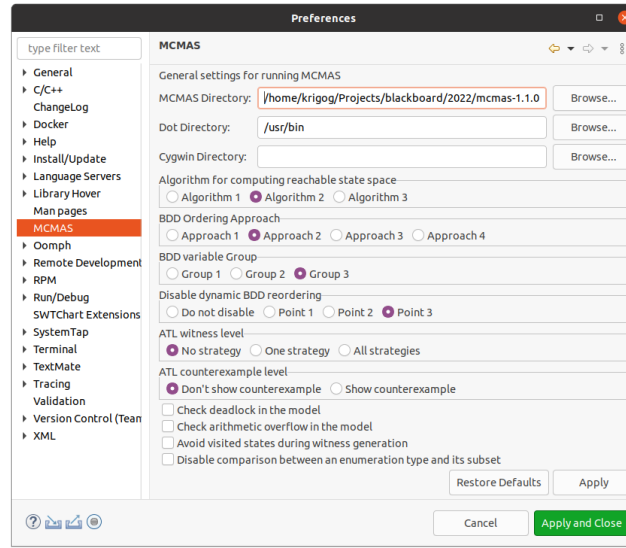


Figure 2: MCMAS preferences on Linux

6 Running MCMAS from the command line

To run MCMAS, in a shell window, `cd` to the folder `mcmas-1.1.0`⁴ and type the command:

```
./mcmas <filename>.ispl
```

`<filename>.ispl` is the `ispl` file containing the specification of the model and the properties to be verified. For example, the command

```
./mcmas examples/bit_transmission_protocol.ispl
```

should produce something like:

⁴You may find it convenient to add the path to the `mcmas-1.1.0` folder to your `PATH`, so that you can run `mcmas` from any folder.

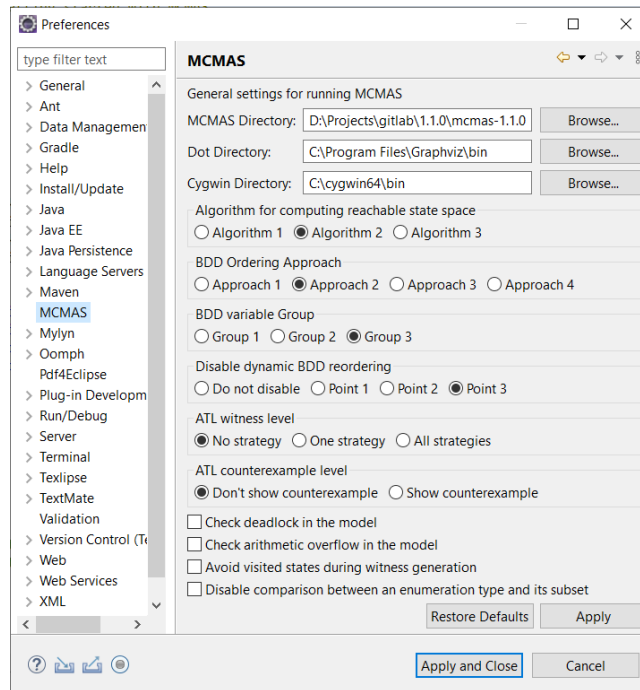


Figure 3: MCMAS preferences on Windows

```
meier:(554$) ./mcmas examples/bit_transmission_protocol.ispl
*****
MCMAS v1.1.0

This software comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

Please check http://vas.doc.ic.ac.uk/tools/mcmas/ for the latest release.
Please send any feedback to <mcmas@imperial.ac.uk>
*****

Command line: ./mcmas examples/bit_transmission_protocol.ispl

examples/bit_transmission_protocol.ispl has been parsed successfully.
Global syntax checking...
1
1
1
Done
Encoding BDD parameters...
Building partial transition relation...
Building BDD for initial states...
Building reachable state space...
Checking formulae...
Building set of fair states...
Verifying properties...
  Formula number 1: (AF K(Sender, (K(Receiver, bit0) || K(Receiver, bit1))))), is TRUE in the model
  Formula number 2: (AG (recack -> K(Sender, (K(Receiver, bit0) || K(Receiver, bit1))))), is TRUE in the model
done, 2 formulae successfully read and checked
execution time = 0.008
number of reachable states = 18
BDD memory in use = 9018048
```

`.ispl` files can be produced using the GUI (see below), or any text editor.

There are several command line options (the command `./mcmas` without a filename produces a full list of options). Many of these are used to control the way MCMAS

generates its internal data structures, or to enable experimental features, and are not relevant for INFOMLSAI. The most useful options are:

- s interactive execution (i.e., simulation)
- c [1--3] specify the way counterexamples/witnesses are displayed
- p specify the path to store counterexample files

Other options, e.g., relating to ATL, will be covered in the relevant practicals.

7 Running MCMAS from the GUI

You can create a new MCMAS project in Eclipse by selecting File -> New -> Project and then choosing MCMAS project. The new project contains a template `ispl` file. The file can be renamed and it should be completed with the specification of the model and the properties to be checked. Syntax errors are flagged and some contextual help is provided to fix them. An example `ispl` file is shown in Figure 4

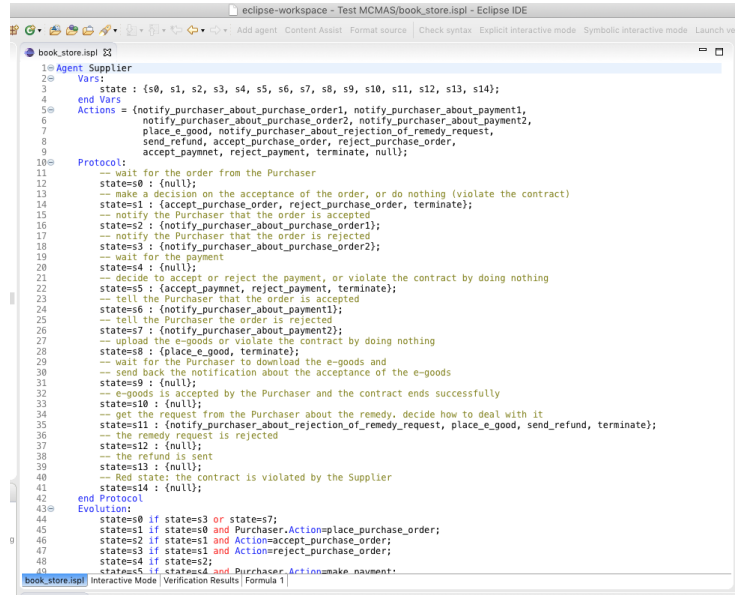


Figure 4: GUI `ispl` editing pane

Verification, simulation, and counterexample/witness analysis can be performed from the graphical interface, using both the commands in the MCMAS menu, or the buttons in the Eclipse tool bar. For example, Figure 5 shows the verification pane, and Figure 6 shows a graphical visualisation of a counterexample to the first property.

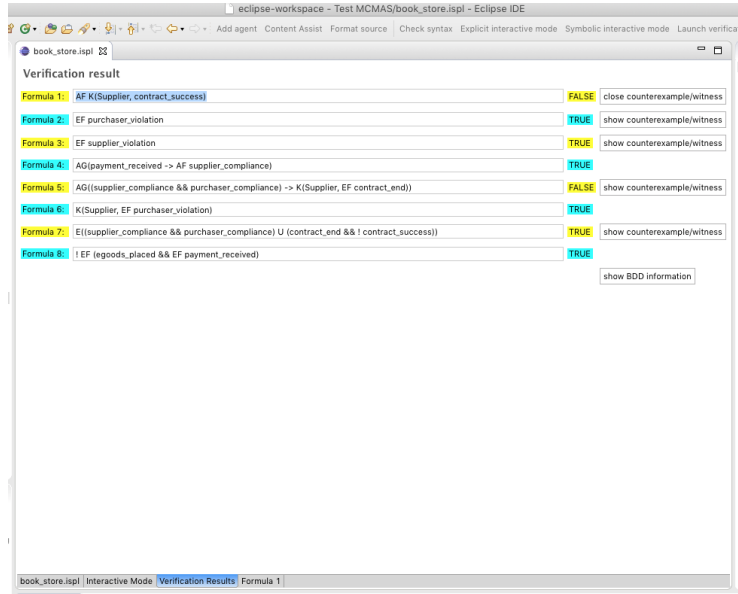


Figure 5: GUI verification pane

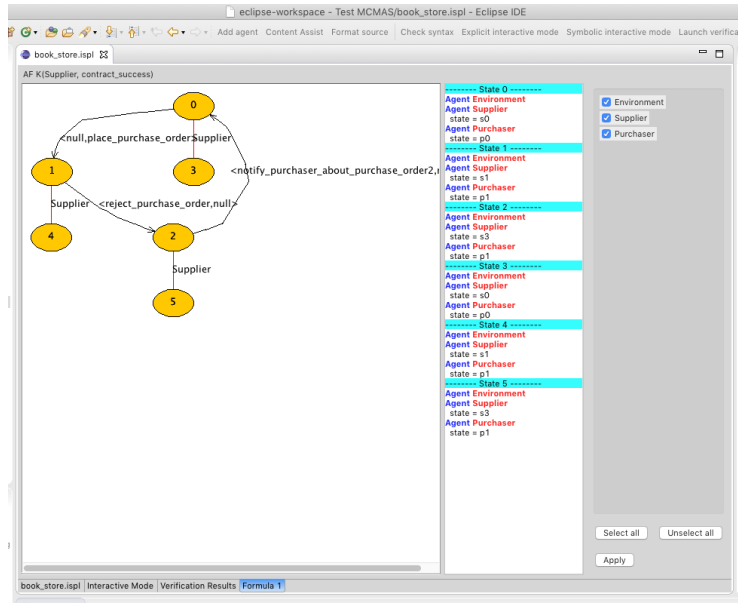


Figure 6: GUI counterexample/witness pane