Chapter 6

How to use Melodizer

This chapter will explain exactly how to load and use Melodizer inside OpenMusic. It assumes you already have installed OpenMusic and Gecode and have downloaded the libraries. Appendix A gives a step by step guide on how to install everything you need. This chapter will first explain how to create a patch and load the libraries. It will then explain how to give input to the tool and how to specify aditional constraints to use. It will also cover how to search for solutions, how to modify them and how to save them. Finally, it will explain how to combine generated solutions to create musical pieces.

6.1 Setting up the environment

6.1.1 Create a patch

The first step to be able to use Melodizer is to create an OpenMusic project. To do so, the user should open OpenMusic and select "create a new workspace", give it a name and press enter. This creates an empty workspace. Then click on file/new patch, and your workspace should look like figure 6.1. Double clicking on the patch will open it.

6.1.2 Load Melodizer and GiL

Now that you have created a patch, the next step is to load the Melodizer[1] library. Since it requires GiL[2] to work, it needs to be loaded too. To do so, click on windows/library. This will open the library window (figure 6.2a). Then click on file/add remote user library and navigate to the Melodizer and GiL folder on your desktop. The library window should now look like figure 6.2b. The libraries can be loaded by right-clicking

[1] https://github.com/sprockeelsd/Melodizer
[2] https://github.com/sprockeelsd/GiLv2.0

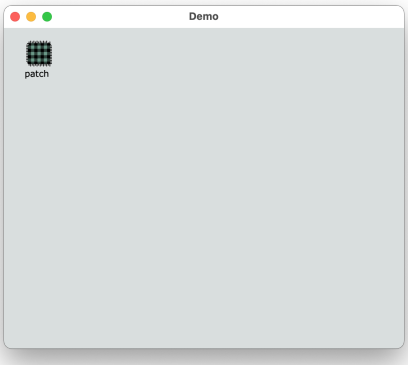
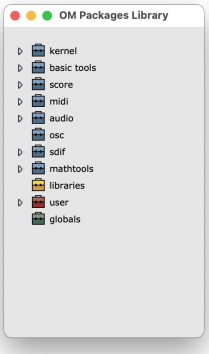
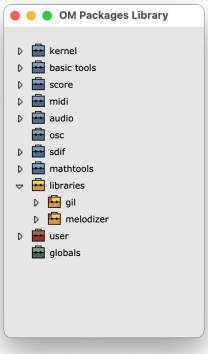


Figure 6.1: Workspace after adding a patch

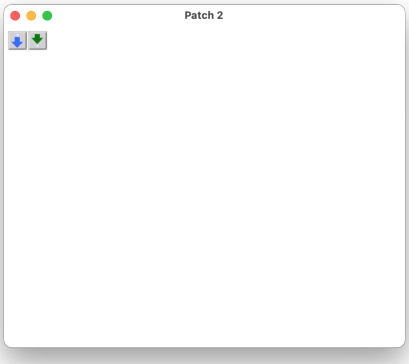


(a) Library window in OpenMusic

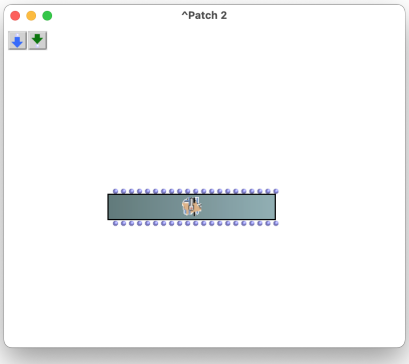


(b) Library window in OpenMusic when the libraries have been fetch

Figure 6.2: Library window before and after fetching the libraries



(a) Patch before creating a melodizer object



(b) Patch after creating a melodizer object

Figure 6.3: Patch before and after creating a melodizer object

the melodizer library and selecting “load library”. The boxes should turn blue, meaning that the libraries are loaded and ready to use.

6.1.3 Creating a melodizer object

Now that the libraries are loaded, they can be used in the patch that was just created. In the patch, select Classes/Libraries/Melodizer/ALL/MELODIZER, then click anywhere in the patch and a melodizer object will appear. Alternatively, you can double click in the patch and write “mldz::melodizer” to achieve the same result. Figure 6.3 shows a patch before and after creating a melodizer object. Once the object is created, the editor can be opened by double clicking on it. The melodizer object has 21 inlets and outlets, but only a few should be used for input and output. The inlets that can be used to give input to Melodizer are stated in section 5.3.1, and the ones that can be used to get output from Melodizer are stated in section 5.3.3.

6.2 Motif, phrase, period

In the following sections, motifs, phrases and periods will be mentioned multiple times so we feel it is important to give a proper definition of those terms [10].

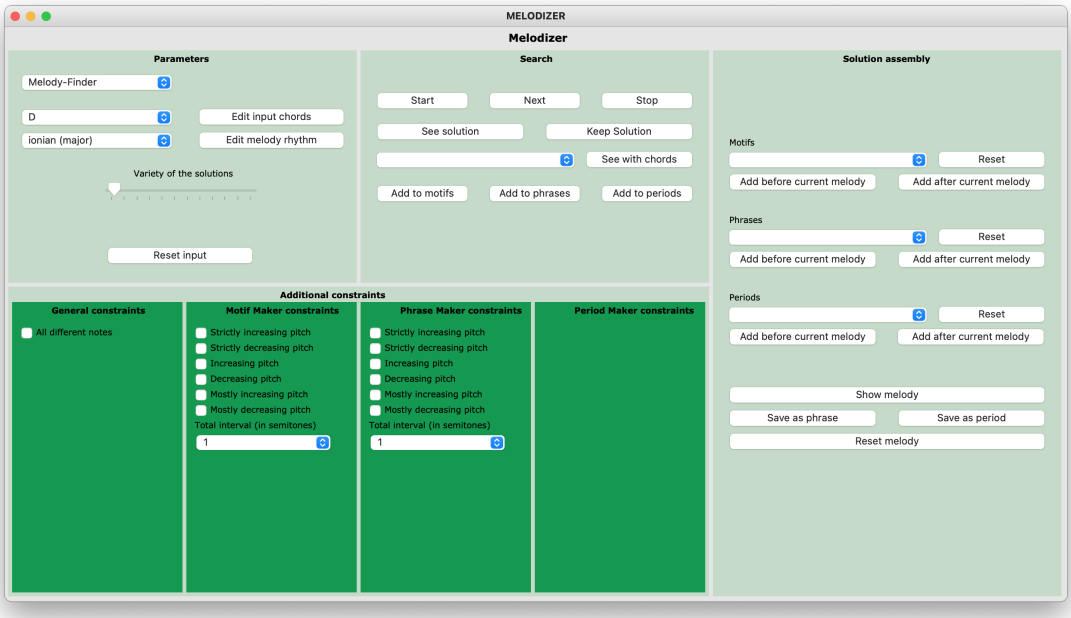


Figure 6.4: Editor window of the melodizer object

Motif A motif is a short melodic idea. It can be as simple as two notes, and can be more complex but is generally really short (up to one measure long).

Phrase A phrase is a little bit longer than a motif. It is generally a few bars long, and expresses a more complex melodic idea.

Period A period is a complete melodic idea. It can be composed of several phrases.

6.3 Giving input to the object

Now that you have created a melodizer object, the next step is to give it parameters to customize the problem to fit your needs. The input parameters are described in section 5.3.1. There are two ways to specify those parameters : through the editor window of the melodizer object or by connecting boxes to the corresponding inlets (the circles on top of the object) of the melodizer object.

6.3.1 Giving input from the editor window

This is the recommended way to set the parameters of your problem. To open the editor window, double click on the melodizer object. The window in figure 6.4 should appear. The editor window has four panels : the input panel on the top left, the search panel in the middle, the optional constraints panel at the bottom and the solution assembly panel on the right. For this section, we are interested in the input panel. The input panel is represented in figure 6.5. All the input parameters can be specified there. Here is how to enter parameters to the interface :

- To select the **mode of the tool**, the pop-up menu on the top left can be used. It currently has two modes : melody-finder, that allows you to find pitch for a melody that can be played on top of the input chords and that has the rhythm specified as input, and variation-maker, that allows you to generate variations of a melody that is given as an argument and that can be played on top of the chords given as an argument.
- To specify the **tonality** desired for the melody, there are two pop-up menus that allow you to select a key and a mode.

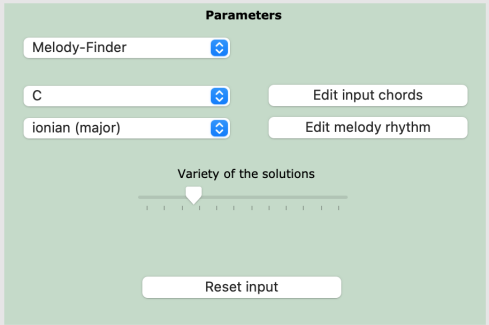


Figure 6.5: Input panel of the melodizer object editor window

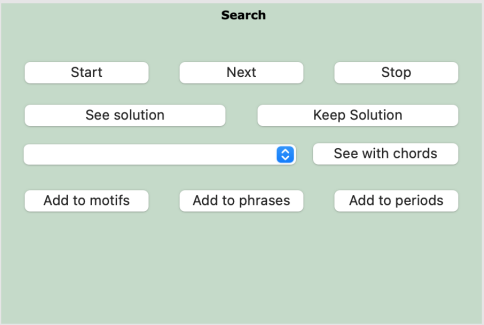


Figure 6.6: Search panel of the melodizer object editor window

- To enter a **chord progression** that serves as a base for the melody, you can press the *Edit input chords* button. This will open the editor window of a voice object and allow you to edit it like it is explained in section 5.2.
- To enter a **melody** or a **rhythm for a melody**, you can press the *Edit melody rhythm* button. That will open the editor window of a voice object and allow you to edit it.
- To **reset the input chords and melody** that you entered, you can press the *Reset input* button.

6.3.2 From the patch

Alternatively, you can give arguments to the object by creating data boxes and connecting them to the corresponding inlet of the melodizer object (the input-chords inlet for the input chords, the key inlet for the key, and so on). The name of an inlet will appear when trying to connect something to it, or can be found in the documentation of the melodizer object. Pay attention that giving inappropriate values to parameters can lead to errors (e.g. trying to set the input chords to a string). Additionally, connecting data boxes to the inlets of the melodizer object means that it must be evaluated to take these changes into account, and will therefore reset the object and everything that was saved in it. If you are more comfortable with creating voice objects from a rhythm-tree and a list of pitches, it is advised to create a voice object separately and then copy the staff and paste it inside the editor window after clicking on the *edit input chords* button.

6.4 Specifying the optional constraints

Now that the parameters to the problem are set, it is time to specify additional constraints if you want to add some to your problem. The optional constraints panel is displayed in figure 6.7. If the mode of the tool is melody-finder, depending on what you are looking for, there is a set of constraints that can be added to the problem. To do so, just check the corresponding box. Some constraints may require additional information (e.g. the mostly increasing/decreasing constraints). Be aware that currently, there is nothing preventing you from selecting incompatible constraints. Doing so will result in hazardous behaviour. Similarly, selecting a constraint that is not compatible with the mode of the tool will have hazardous behaviour.

6.5 Searching for a solution

Now that the parameters and the additional constraints have been set, you can start searching for solutions. Figure 6.6 shows the search panel of the editor window. The first thing to do is to press the *Start* button. At this point, the parameters and the optional constraints are locked for the search for a solution, and any changes will not be taken into account until the *Start* button is pressed again. The problem is now created, and you can **start searching for solutions** by clicking the *next button*. Depending on the problem and the constraints selected, the solver might take some time to find a solution. If you wish to **stop the search**, you can press the *Stop* button. A window will appear to inform you that the search has been stopped. When the

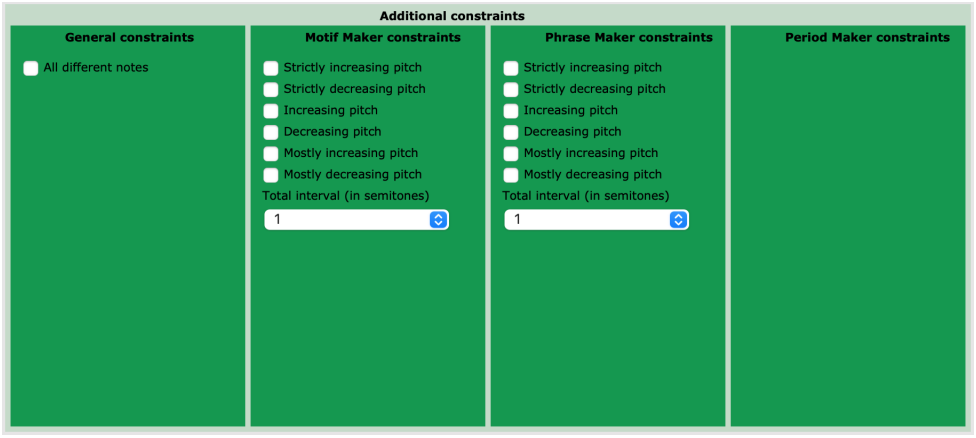


Figure 6.7: Optional constraints panel of the melodizer object editor window

solver finds a solution, a voice editor window containing the found melody will automatically open. You can modify it if you want, and then close the window.

If you want to **see the current solution** again, you can click the *See solution* button. If you want to **keep the solution for later** and keep searching for other solutions, you can press the *Keep solution* button. Doing so will save the solution in the pop-up menu below. Pressing the *Next* button again will erase the previous solution if it wasn't saved. If you want to **see a saved solution with the input chords**, you can select it in the pop-up menu and then press the *See with chords* button. This will open a window displaying the solution with the chords. Be careful that any modification in this window are not saved and will be lost.

If you want to **keep a saved solution for later**, you can add it to the list of motifs, phrases or periods by pressing the *Add to motifs*, *Add to phrases* or *Add to periods* button. You can modify the parameters and optional constraints for another problem, and click the *Start* button to start searching for solutions. Note that doing so will erase all solutions saved in the search panel, so make sure you save what you want by adding solutions to the motifs, phrases or periods accordingly.

6.6 Assembling a melody using generated solutions

Once you have kept solutions you liked, you might want to combine them together to create a musical piece. This can be done through the solution assembly panel, depicted in figure 6.8. The list of saved motifs, phrases and periods can be found in the pop-up menus. You can create a melody by adding different elements before or after the current melody using the corresponding buttons. The current melody can be opened and edited by pressing the *Show melody* button. You can add elements, then modify the melody or add notes/measures to the melody, then add more elements, etc. The melody can be saved as a phrase or as a period using the corresponding button, and can be erased to start a new melody by pressing the *Reset melody* button. The lists of motifs, phrases and periods can also be erased by pressing the corresponding *Reset* button.

6.7 Making Melodizer interact with other elements in OpenMusic

Other elements can interact with Melodizer by connecting to its inlets/outlets. Solutions can be exported from the melodizer object to be used in the patch using the "solutions-list", "motifs-list", "phrases-list", "periods-list", "output-solution", "output-motif", "output-phrase", "output-period" and "melody" outlets.

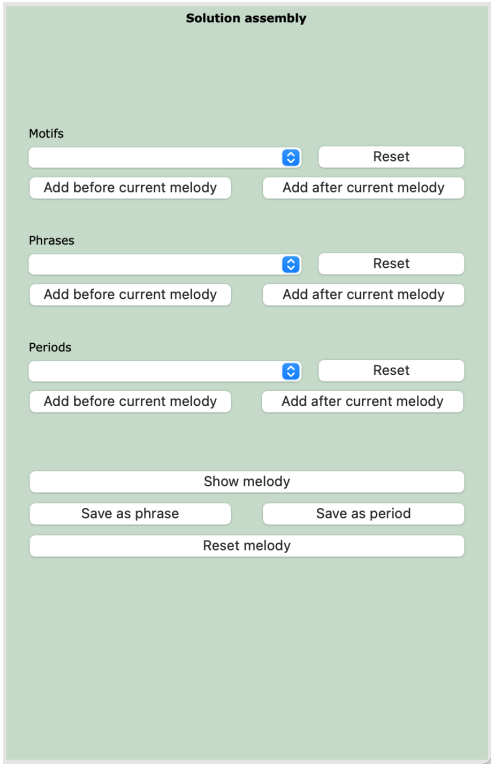


Figure 6.8: Solution assembly panel of the melodizer object editor window