

Using MEdit with the Sony AIBOs

Kelly Cannon and Maitreyi Nanjanath

Using MEdit with the Sony AIBOs requires no programming skills and, with limited leader assistance, has been a fun activity for middle school students at the University of Minnesota. In an effort to promote the use of the Sony AIBOs, I have compiled the following help document.

Software and Hardware Set Up

Hardware

To use MEdit for the Sony AIBOs, you will need access to a Sony AIBO, a PC, a Memory Stick reader, and an AIBO Memory Stick.

Software for the PC

Install MEdit which is freely available for non-commercial use (http://openr.aibo.com/openr/file/MEdit_ver1.zip). Also install the MEdit Sample (http://openr.aibo.com/openr/file/sample_contents_ver1.zip). Be careful to place the MEdit Sample in a directory structure without spaces. Placing the MEdit Sample in a directory structure with spaces such as under “My Documents” will result in a failure by MEdit because of its inability to handle spaces in directories.

Software for the Memory Stick

Install both the Memory Stick image for the Open-R SDK (http://openr.aibo.com/openr/file/OPEN_R_SDK-1.1.5-r5.tar.gz) as well as the RTool (http://openr.aibo.com/openr/file/RTool_ver1.zip) on the Memory Stick. Be careful to select the proper variant of the Open-R SDK before copying to the memory stick. There are many options available in the directory structure. Assuming that the file was unzipped to C, the correct directory to copy to the Memory Stick would be “C:\R_CODE_plugin_r1\OPEN_R_SDK\OPEN_R\MS_ERS7\RCODE\Eng\RCODE.”

Creating Movements with Optional Music for the Sony AIBO

Creating movements with MEdit is simply a matter of creating individual poses and stringing those poses together using time intervals. Complex movements can be created simply by moving between individual poses at varying speeds. In order to create movements, you will need to create motion sets. A motion set is a series of poses that can be performed by the AIBO without a major position shift—that is, without a shift between standing, sitting, or sleeping. Motion sets may be tied together to create more advanced movements, and music may also be added.

Getting Started

First, become familiar with the MEdit window layout. There are four windows with which a user will interact: the motion control window, the pose control window, the contents control window, and the main window which has a 3d representation of the Sony AIBO. Buttons within the individual windows are specific to the window type. For example, to save a pose, always hit save within the pose control window. A sample MEdit screen shot may be seen in Figure 1.

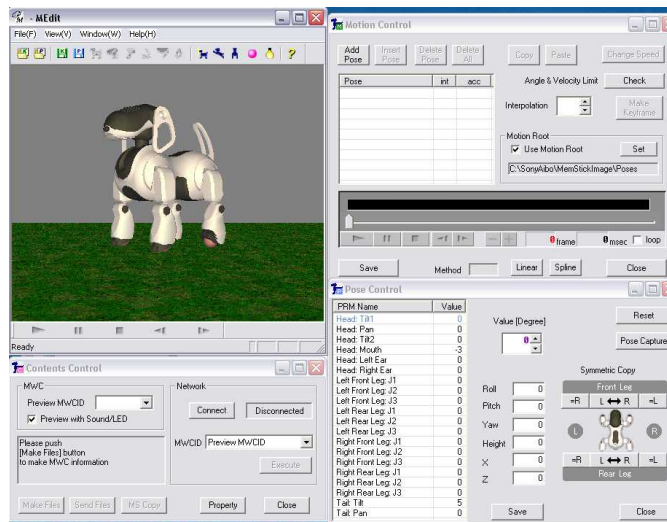


Figure 1: Screenshot of MEdit

To start creating your desired movement, select an initial pose. Always start and stop any motion set with one of the three files available in the OrgPose directory of the MEdit Sample: sit.pse, stand.pse, or sleep.pse. Transitions between these positions should not be handled manually. If you would like to transition between the positions, make multiple motion sets and string them together as described below. To add this motion to the motion set, click “Add Pose” from the motion control window, find the desired file, and click “Ok.” You are now ready to design your own poses.

Creating a New Pose

To design a pose, you have two choices; you may either click and drag the 3d representation of the AIBO, or you may change the attributes within the pose control window. Although clicking and dragging is easier, more precise poses may be designed by using the pose control attributes.

To use the 3d representation, left click and drag to rotate the camera view. Right clicking and dragging on individual dog parts will move the individual parts. The 3d representation will not allow you to overextend the dogs joints, but keep in mind that gravity is not considered, so what may appear to be a safe and stable position may in fact be unstable.

To use the pose control attributes, simply change the values to your desired values. Note that if your value is outside of the safe joint positions, you will receive a warning and must reduce the value to that which is safe. Also, when you click on a value that you wish to change, the part of the dog which is controlled by that value will be highlighted, which helps greatly in distinguishing between the different parts.

Once you have created a pose, you should save that pose by clicking “Save” within the pose control window. Be sure to save the pose under easily accessible directories which do not have spaces. Also, avoid using spaces when naming any files used by MEdit. Although no error message will be given immediately, an export of the motion file will fail.

Adding a New Pose

After your pose has been saved, add that pose by clicking “Add Pose” in the motion control window. Once you locate the file and click “Ok,” you will be asked for a time interval. Feel free to explore with varying time intervals. You may click the play button, which is represented by the traditional right pointing triangle, to see how your motion will appear given the time interval provided. Time intervals may be changed in this window as necessary. At

this time, I strongly recommend that you click “Check” to check the safety of your motion. Checking ensures that the given motions will not cause damage to the AIBO. Although you may postpone checking until a motion set is complete, it will be more difficult to find and correct those movements which are not safe if you wait.

Be careful to consider the speed of the computer you are using when creating simulations. The AIBOs will not suffer the lag seen on many slower machines, so it may be possible that your actual motions will be much faster than those displayed on the CPU. Similarly, if you have a very fast CPU, the motions may be identical. Ideally, you would want to test this before finishing your movements.

Finishing Your Motion Set

To finish your motion set, simply repeat the process of creating poses, adding poses, and checking those poses for safety. If you make a mistake, use the “Delete Pose” or “Delete All” buttons on the motion control window to correct them. Be sure that your motion ends in the same position that it started.

Transferring Previously Created Movements from MEdit to the AIBO

Before you begin the process of transferring the movements from MEdit to AIBO, always run a final check to ensure that the movements are safe for the AIBO by clicking “Check” in the motion control window. If there are any error messages, make the recommended changes. Only after a successful check has been run should you proceed to the following steps.

Saving Files

First, you will need to save two types of files. The first is a kfm file. This may be created by click “Save” in the motion control window. This file may be saved anywhere that is easily accessible. The second is a mtn file. This may be created by clicking “File\Export Mtn All” in the main window. This file should be saved on the Memory Stick in the directory “RTool/ODA/Motion.” This may be repeated for as many motion sets as are created. Similarly, if you wish to add sound to your movements, drop any midi files into “RTool/ODA/Sound.”

Creating the Necessary Code

Next, you need to create code which will tell the AIBO which motions to perform and in what order. There are two files which hold the code. The `mwc.cfg` file is located in “RTool/MCW. This file contains the instructions for the individual motion sets and is partially generated by MEdit. The second file is the “R-CODE.R” file which is located in “OPEN-R/APP/PC/AMS.” This file gives the order in which motions should be played.

Creating the MWC.CFG File

To create the `mwc.cfg` file, drag and drop the `kfm` file and, if you choose, a midi file onto the 3d representation of the Sony AIBO in the main window. Then, hit Cntl-C. If you paste what was copied, you will find something similar to the code shown below:

```
#MWC 1.0
1 1
9000 1
cmagentMOTIONPERFORMER a_mymotionname 1 1 0x0 0x0 0
```

The first line should never be changed. The second line has two values. The first is the number of MWCommands that are included in the file, and the second is the number of CMAgent commands that are included in the file. An MWCommand is defined as a motion and optionally a sound and/or led configuration which are intended to be run simultaneously. In the example above, there is only one MWCommand—command 9000 which has 1 CMAgent command. The third line lists a number which is used to identify the MWCommand, 9000, and a number indicating the total number of CMAgent commands in the given MWCommand, 1. A second and more complex sample is provided below:

```
#MWC 1.0
6 11
26000 3
cmagentMOTIONPERFORMER a_stand#stand_so0r_greet 1 1 0x0 0x0 0
cmagentSOUNDPERFORMER soc_d00greetso0r_x1x 1 1 0x0 0x0 0
cmagentFACELIGHT so12_d00greetso0r_l1f 1 1 0x0 0x0 0
26001 1
cmagentMOTIONPERFORMER a_stand#stand_so0r_greet 1 0 0x0 0x0 0
26002 1
```

```

cmagentSOUNDPERFORMER soc_d00greetso0r_x1x 1 0 0x0 0x0 0
26003 1
cmagentFACELIGHT so12_d00greetso0r_l1f 1 0 0x0 0x0 0
26004 2
cmagentSOUNDPERFORMER soc_d00greetso0r_x1x 1 1 0x0 0x0 0
cmagentFACELIGHT so12_d00greetso0r_l1f 1 1 0x0 0x0 0
26005 3
cmagentMOTIONPERFORMER a_sit#sit_sox_bye 1 1 0x0 0x0 0
cmagentSOUNDPERFORMER sox_t00byesox_x1x 1 1 0x0 0x0 0
cmagentFACELIGHT sox_t00byesox_l1f 1 1 0x0 0x0 0

```

If you only intend to use one MWCommand, there will be no need to do any editing. You may simply use the drag/drop/copy method. If you intend to use multiple MWCommands, you may still use the drag/drop/copy method, but you will need to edit the numbers so that a) they accurately reflect the number of CMAgent commands and MWCommands and also so that b) each MWCommand has a unique number. MWCommand numbers should be greater than 9000 and less than 20000. Also, remove the first and second lines from any consecutive pastes, as the first two lines should only be provided once. After editing, save this file under the same name.

Creating the R-CODE.R File

Once you have created the “mwc.cfg” file, you must create the “R-CODE.R” file. This file simply lists the MWCommands to be performed in the order desired. For example, if you had three MWCommands that you wished to run which were given the numbers 9000, 11003, and 12078, your “R-CODE.R” file would simply be as follows:

```

PLAY:MWCID:9000
WAIT
PLAY:MWCID:11003
WAIT
PLAY:MWCID:12078
WAIT

```

This is also the time where a robot may be moved between the three basic poses provided. These may be performed by adding the following lines of code:

PLAY:ACTION:SIT	Move safely to sitting pose
PLAY:ACTION:STAND	Move safely to standing pose
PLAY:ACTION:LIE	Move safely to sleeping pose

If I had an MWCCommand (15000) which started and finished in the sitting position that I wished to be followed by an MWCCommand (150001) which started and finished in the sleeping position, the following code would be necessary:

```
PLAY:ACTION:SIT
WAIT
PLAY:MWCID:15000
WAIT
PLAY:ACTION:LIE
WAIT
PLAY:MWCID:15001
WAIT
```

The first movement moves the AIBO from the traditional standing position, which is always the starting position unless otherwise indicated, to the sitting position which is necessary for 15000. The third movement moves the AIBO from the sitting position to the sleeping position which is necessary for 15001.

After editing, save this file under the same name.

Creating and Transferring the AIBO Files

After placing the mtn and midi files in their respective directories and creating the “mwc.cfg” and “R-CODE.R” files, the next step is to use RTool to compile the code. Open up the application RTool which is found in the “RTool” directory. Click “Make MWC.” Once this has completed successfully, click on the “ODA” tab and then click “Make ODA.” There is no need to change any of the variables if you have followed the instructions above.

Once both the MWC and the ODA are created, simply copy the “OPEN-R” directory found under “RTool/MS” to the root of the Memory Stick. Click “Yes” if there is a request to overwrite files. Appropriately remove the Memory Stick from the Memory Stick Reader and insert into an AIBO. After initial reboot music, your movements and or actions should play.

For more help with using MEdit, please reference the MEdit Manual (http://openr.aibo.com/openr/file/MEdit_Manual_E_Ver1.zip) or this R-Code SDK Tutorial (http://www.ouroboros.org/rcode_tutorial_1v2.pdf).