



2.1 PRELIMINARY DOCUMENTATION

mlrv is a sampling instrument for grid-based music controllers. it is an refactoring of brian crabtree's original mlr, extended by trent gill (galapagoose) and michael felix (%) of /parallelogram/.

connecting your controller

mlrv includes support for a wide variety of grid-based music controllers. click the setup button to open the setup pane and take note of the 'control' section. within, you'll find a drop down box which will allow you to choose one of many available preset configurations for various controllers. mlrv will attempt to auto-select whatever device was connected when Max/MSP started up.

importing samples

mlrv supports aif, wav, and mp3 sample formats. samples may be imported into mlrv by dragging and dropping them from your filesystem into the mlrv window. dropping a sample into the left-hand portion of a row will auto-load the new sample into the respective row and replace whatever was previously there. dropping a sample into the right-hand side of the row will simply load the sample into mlrv; you'll have to manually assign it to a row or cell using the file select dropdowns.

mapping and playback

a sample may be mapped across each row of the grid controller and assigned both a triggering behaviour and playback group. each of the four (or six) groups will allow only one sample to play at a time, and triggering a different sample in that same group whilst another is playing will cancel the playback of the previously playing sample (like 'mute groups' on a traditional sampler). each group has a 'group stop' button that when pressed will cancel all audio playback in its respective group.

samples assigned to rows may be subdivided with a number of steps, which mlrv will attempt to interpolate to the cells of the corresponding row on the grid controller. the user may dynamically reselect the start and end points of the sample and mlrv will reinterpolate the sample with the desired step subdivision. when a button is pressed on the grid controller, mlrv will play the sample back from the corresponding position. the current playback position of the sample is visualized by a lit pad on the grid controller. sample playback behaviour is controlled via one of the following triggering modes:

- loop — the sample will playback and repeat when it reaches the end.
- shot — the sample will play from the beginning of the chosen slice to the end of the sample and then stop.
- slice — the sample will play when held and continue through the loop until released.

recording button presses

button presses may be recorded and played back with a pattern recorder. each one of the four available pattern recorders can be set to a length of 1 to 32 bars. when armed, the pattern recorder will wait until a button is pressed, then immediately begin to record button presses until it reaches the end of its recording buffer. then and there, the pattern will begin to playback automatically. the user may overdub a pattern by pressing and holding down its respective trigger button and providing more input. patterns may be silenced by simply pressing the trigger button or deleted with a press-and-release gesture.

recording live audio

live audio input (or resampled audio output) may be recorded into one of several audio buffers. just like samples, these buffers can be assigned to a row or grid cell and triggered in a similar fashion. each buffer can be assigned a length of 1 to 32 bars and can optionally be placed in a looping mode which will continually record and overwrite itself. pre-roll allows you to delay recording for a number of bars.

dsp configuration

the audio driver can be selected within the setup pane. mlrv will attempt to display any compatible audio driv-

ers within the driver drop-down box. within the mixer section, group outputs may be routed to different outputs by clicking the button directly underneath the group fader. the available options in this list will be dependent upon the selected audio driver. by default, all group outputs go to the master channel.

if you'd like to use rewire, please first make sure that Max/MSP is configured as a rewire slave (by opening it after you open your rewire host, e.g. ableton live). then, click the rewire button in the setup pane and mlrv will automatically assign group outputs to rewire channel pairs and disable any custom audio routing you may have set up.

mapping

every parameter within mlrv can be remotely controlled via MIDI, OSC, or via your computer's keyboard. check the setup pane to configure the MIDI input and OSC configurations for your particular setup. all mappings are configured through the mapping tab in the right-hand side of the mlrv interface. top-row controls are also controlled via the mapping interface.

MIDI MAPPING

to map an interface parameter to a MIDI control, first enable mapping mode within the mapping tab (or hit option/alt + m on your keyboard). this will highlight all of the available parameters that can be controlled. to map a parameter, simply click on the interface element and then turn a knob or press a key on your MIDI device. the mapping table will update to show the newly mapped parameter. you can adjust the minimum and maximums by double clicking and editing the data within the table cells. to delete a mapping, simply click within a row of the mapping table and press delete.

OSC MAPPING

to map an interface parameter to an OSC control, enable mapping mode and click on an interface parameter. the mapping tab will reveal the associated parameter's OSC address. your OSC device can send control signals to this address, and they will be mapped from 0 to 1, with 32 bit floating point precision. please note that minimum and maximums cannot be set within mlrv, it's up to your OSC device to handle these. please refer to the mlrv OSC specification at the end of this document for a complete list of available addresses.

KEY MAPPING

to map an interface parameter to a key on your computer's keyboard, enable mapping mode and click on an interface parameter then press a key.

TOP ROW CONTROLS

in contrast to previous versions of mlr and mlrv, all top-row controls (like group stops, pattern recorders, etc) are handled via mapping. by default, the top-row control arrangement is the same as before:

- row 1, buttons 1-4 — stops for groups 1-4, respectively
- row 1, buttons 5-6 — pattern recorders 1 and 2
- row 1, button 7 — aux send
- row 1, button 8 — delay feedback

to re-assign top-row controls, enter mapping mode. group stops are shown in the bottom of the group faders and pattern recorder triggers are positioned next to their respective recorder interface elements. pressing a button on the top row of your grid controller whilst in mapping mode will assign these controls.

saving presets

mlrv 2.0 incorporates a redesigned preset management system that saves most every interface parameter within individual presets (with the exception of quantization, and a few other things). mlrv also saves mapping configurations, albeit these are saved globally within the preset. when mlrv restarts, it will attempt to recall all previously saved setup preferences, along with the last loaded preset file as well.

to save a preset, simply click the 'store' button underneath the preset dropdown. you'll be prompted to give the preset a name via a popup window. clicking 'save' will save the current preset, and 'save as' will save the entire preset file to your filesystem. preset files are saved in a human-readable json format, in the case that your samples have moved or if you want more finer-grained control over the particulars.

OSC specification

all parameters mapped from 0 to 1 with 32bit floating point precision

TIMING

/time/tempo
/time/quantize
/time/tempo[next]

PRESETS

/presets/current

ROWS

where X is channel/row number:

/grid/Xch/sel-start - selection start for loaded sample
/grid/Xch/sel-end - selection end for loaded sample
/grid/Xch/mode - row mode (loop/slice/shot)
/grid/Xch/filename - loaded sample
/grid/Xch/steps - step subdivision
/grid/Xch/octave
/grid/Xch/speed
/grid/Xch/transpose
/grid/Xch/gain
/grid/Xch/group

PATTERN RECORDERS

where X is pattern recorder number (1-4):

/Xpattern/length

MIXER

/mixer/aux/vfader - volume fader for aux group
/mixer/aux/sendtog - send toggle
/mixer/dac/vfader - master fader

where X is group number (1-6):

/mixer/Xgrp/vfader - volume fader
/mixer/Xgrp/sfader - send fader
/mixer/Xgrp/route - audio routing
/mixer/Xgrp/mute - group mute