MUS 4711 – Interactive Computer Music

Course Lecture Notes

Intro to MaxMSP, Math, and Computer Logic

PART 1a: Basics

Objects – [bang], [toggle], [slider], [dial], [print], [gain~], [float], [integer], [number~], [kslider], [nslider]

MaxMSP is a dataflow programming language developed by Miller S Puckette in 1989 to control multiple MIDI devices in real-time, faster than humans could possibly change their parameters. It has since expanded to include objects for real-time audio analysis and manipulation, video manipulation, the use of OpenGL for hardware accelerated video rendering, advanced coding through the inclusion of [gen] and [codebox], and integration with Ableton Live as Max4Live. Due to its ease of use, MaxMSP is considered to be the lingua franca for audio programming.

Dataflow and Order of Operations

- -As a dataflow language, MaxMSP patches are constructed from a number of objects, messages, numbers, sub patches, and UI elements that are arranged on a canvas and connected together with cables, similar to the modules in a modular synth.
- -The order in which things happen is very important in MaxMSP – it's a frequent cause of headaches for even the best programmers.
    - -The order of operations is Right to Left, Top to Bottom.
    - -So, a patch with a bunch of things happening in it will always run the top right instructions first, then work its way down in rows, until it hits the bottom left most point.
        - -This is because most objects have a "hot" inlet on the left that triggers the execution of the object, and any number of "cold" inlets moving to the right that change parameters of the object.
            - -The "hot" inlet is typically expecting some type of input, and runs when it's received – it doesn't typically require a special trigger.

Objects, Messages, and Bears, Oh My!

- -Objects are always rectangular. Their function is to do a certain task.
    - -There are two types of objects. **Audio** objects are marked with a tilde (~) and operate at the sample rate you've set MaxMSP to process at – usually 44100 kHz
    - -**Control** objects typically handle things that don't require as much real time processing, like data from lists, MIDI, or calculating a metronome BPM. They do **not** have a tilde (~).
    - -Either type of object can be created with the shortcut key **n**.

-Messages tell an object how to do something. They can get pretty complex, but are always recognizable as rounded rectangle without the borderlines.
-Which messages are accepted by an object is always the question. When in doubt, open the Reference for the object.

    -Since version 8, they also have a cold inlet – this will be REALLY useful for testing the output of objects as it replaces the content of the message with whatever it receives. Just trust me on this.
    -Messages are created with the shortcut **m**.

-Comments are transparent boxes that you fill with text. They have no other function in MaxMSP, do not affect the way in which things run, or take up any resources to use/display.

    -They're also the single most useful thing in ANY programming language as they let you leave notes to yourself or others,

-There are two types of numbers you'll interact with in MaxMSP, integers and floating points.

    -Integers display and output whole numbers – they will shave off any decimals received. They are created with the shortcut **i**.
    -Floats display and output decimals numbers (anything with a decimal place) up to seven decimal places. They are created with the shortcut **f**.
    -There is also an object called [number~] that you will want to avoid – it looks like an int or float, but it's a signal display and offset generator.

-UI Elements

    -User interface elements in MaxMSP are objects that can be directly interacted with by the user to change a value, trigger a function, or perform any number of tasks. Some of the more common ones are:
    -Bang – outputs the special **bang** message – MaxMSP speak for "DO IT NOW!" Shortcut is **b**.
    -Toggle – outputs a 1 or 0. Shortcut is **t**.
    -Slider – outputs a range of ints from 0-127 by default (can be configured). Shortcut is **s**.

        -The dial is essentially the same as a slider, but circular.

    -[gain~] is a special slider. It looks like a regular slider with the barbershop poll stripes, but its function is to scale audio volume. It takes audio in and adjusts the volume for output. Don't confuse the two!
    -[nslider~] and [kslider~] provide graphical ways to generate MIDI note/velocity pairs. They output MIDI note integers on the left and velocity on the right.

The Max Console

-The Console is a separate window that you can bring up with Command-Shift-M. It's where error message, feedback from the program, or other important messages occur. You'll grow to love/hate it.

Edit Mode and Performance Mode

-There are two modes to MaxMSP. Edit and Performance.

-Edit mode is where you can create objects/messages/etc., patch things, and move elements around on the screen.
-Performance mode is where dragging on an object like a slider changes the values, but not the position. This is where you'll want to be when using your patch.
-Change between them by selecting/deselecting Edit under the View menu, Command-E, or holding Command and clicking on the patch.
-You'll also notice the little padlock is unlocked in Edit mode, and locked for presentation mode. It can also be clicked to toggle between the two.

Inspector

-If you want to see what messages an object takes, change its parameters, tweak how it looks, etc. you'll want to use the inspector. In edit mode, select the object, then hit Command-I, or click on the circled *i* icon on the right.
-This is really useful for customizing how objects look and behave in MaxMSP. You'll use it a lot.

Help and Documentation

-Getting help is pretty easy in MaxMSP. There are two major types of help the individual help files for anything in MaxMSP, and the respective references.
-To open a Help file, select anything in a patch and hit Command-Shift-H. This opens the individual help, with an interactive patch, an open reference page showing what messages and attributes can be used with your selected object, and "See Also" suggestions. There can be multiple tabs of examples for you to play with. Feel free to unlock these and copy things into your patch – it's totally valid.
-References are the short, "tech-y" blurbs that tell you just the messages and attributes that something can take in MaxMSP. These are great as a quick "what do I need to send to this" type reminder. Select anything in MaxMSP and hit Command-Shift-R to bring that up.
-These can also be opened from the Help Menu
-The Full Documentation is available. Make sure nothing is selected, and hit Command-Shift-R to open the separate Reference window, then hit Home. This takes you to the Master Documentation list. It has tutorials, as well. The Max Tutorials are mostly MIDI and logic based, MSP is audio, and Jitter is video.
-On the sidebar, hit the File Browser icon (looks like a bullseye) to open it. This lets you search and filter. The Cycling '74 content is particularly useful.
-Online. References are in the syllabus and on the course page (go over a few)
    -PD as a sister language – logic is the same.

Extra time – go over documentation, show example lessons, brainstorm what people want to learn how to do. Show chains of bangs, sliders changing sliders, etc.

Max Puzzle 1 – Create a new patch that takes user data from a [kslider] of any *logical* size (e.g. no more than 88 keys) and displays the MIDI data as a note in an [nslider], and the velocity in a slider or dial connected to an integer. Using the Inspector, set each object to have the same color scheme (e.g. [kslider] black keys, [nslider] staff, slider or dial value, and integer number are the same color. All background colors and white keys are the same color). Be aware that the names of these color parameters will change from object to object. Align your patch cords.

Advanced Mode: Instead of using the Inspector, use two [swatch] objects to send RGB color values via variable messages – e.g. (bgcolor $1 $2 $3) – to control the color of all objects at once. Check the Reference for each object and “Arguments and Special Characters” in the *Helpful Links* patch.