## Composition: Electronic Media II

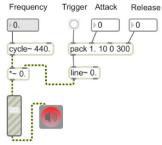
## [line] objects

## Mar. 7, 2012

- 1. [line]
  - a. Creates a 'ramp' from (current value) to (target value) in (time in ms).
  - b. Right inlet can be used to set (time in ms)
  - c. Left inlet will accept either:
    - i. single number [line] will jump immediately to this value
    - ii. pair of numbers [line] will generate a ramp as per 1a.
  - d. Does not 'schedule' ramps. This means that giving [line] a new value before the current ramp is complete will force it to start moving to the new target value from whatever its current value is.
  - e. Can be created with two arguments
    - i. default starting value
    - ii. 'Grain rate.' The higher the grain rate (20ms is default), the more continuous the ramp will be. Higher grain rates are also more CPU intensive. The lowest grain that MAX allows is 1ms, and any value given lower than 1 will be defaulted to 20ms.
  - f. Important to note: [line] outputs float numbers. This means that there will be minute gaps between 'steps' in the ramp.

## 2. [line~]

- a. The significant difference is that [line~] outputs audio signal and not float numbers. This means that the 'steps' present in [line]'s output are done away with.
- b. Accepts one argument, which establishes [line~]'s initial value
- c. Inlets both function as per [line]
- 3. Scheduling ramps
  - a. Using [line~] ramps can be 'scheduled'
  - b. Sending {0 100 1. 100 0. 100} will result in a ramp from [line~]'s current value to 0 in 100ms, then to 1 in 100ms, then back to 0 in 100ms.
  - c. A 'sustain' can be inserted by giving the same value as the preceding arrival point. So, {0 100 1. 100 1. 500 0. 100} will make [line~] stay on 1 for 500ms before moving to 0.
  - d. Sending a new message will replace ramps scheduled by the earlier message.
  - e. So, we can create an envelope generator like this:



- f. Here, the 'Attack' and 'Release' float boxes are replacing the values 10 and 300 in the [pack] object. Pressing the 'Trigger' button sends this list to [line~], {1. a 0 b}.
- g. This schedules these two points within [line $\sim$ ], so that [line $\sim$ ]'s output will increase to 1 within a ms, and then decrease to 0 within b ms.