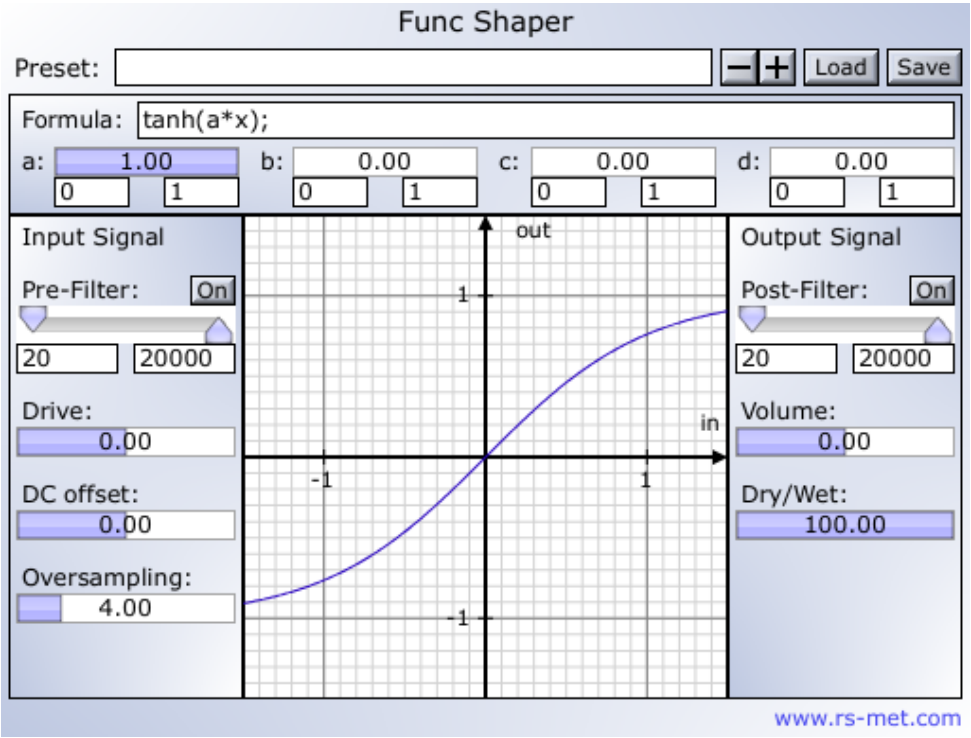
Func Shaper - User Manual



What is Func Shaper?

Func Shaper is a waveshaping distortion plugin based on a mathematical expression evaluator. You can enter an expression to create the function which will be used as waveshaping transfer function. The expression may contain the usual arithmetic operators (+,-,*,/,^) where the last one denotes the raise-to-power operation (works only for integer powers), well known standard functions such as sin, cos, tanh, etc. as well as some proprietary special functions, which are specifically geared towards the application domain. Refer to the appendix of this document for a reference on the supported functions.

The Preset Section

Here you can load and save presets with the respective buttons (where the save-button actually works as a 'save as' button). The +/- buttons allow for fast switching to the next and previous preset in the currently chosen directory (presets are traversed in alphabetical order with wraparound).

The Formula Section

In this section you enter the transfer function as a mathematical expression. Use the character x to represent the input signal. You can also use the parameters a, b, c, d in your expression, the numerical value of which can later be tweaked via the corresponding sliders. Below each parameter slider, you can enter the minimum and maximum values between which the slider can be adjusted. Each expression must

end with a semicolon - this is because you can also enter several expressions, separated by the semicolon. Thereby you create intermediate variables - the final result will be the result of the last expression, for example: $z = 3*x; tanh(a*z);$ will be the same as $tanh(a*(3*x));$. When you use this feature, avoid using names for your intermediate variables which collide with the internal names, that is: don't use x, a, b, c, d for intermediate results.

The Input Section

Pre-Filter: This is a pair of first order highpass and lowpass filters which can be applied to the input signal. The left handle controls the highpass cutoff, the right handle controls the lowpass cutoff and the 'On' button switches the filter on (or off).

Drive: This controls the gain of the signal at the input of the waveshaping transfer function. The scaling unit is decibels.

DC offset: Adds a DC offset to the signal. This makes the input signal drive different regions of the transfer function. The DC is applied after the amplitude has been scaled via 'Drive'.

Oversampling: Nonlinear distortion creates harmonics, and when a harmonic so created is above half the sample-rate, we will face the problem of aliasing (see http://www.rs-met.com/documents/tutorials/DigitalSignals.pdf for some details on this subject). A means of reducing the problems of aliasing is to run the signal processing algorithm on a higher sample-rate than the target sample-rate. This is oversampling - the slider chooses, by which factor the plugin oversamples the input before applying the actual waveshaper. Experiments indicate that a factor of 4 is a reasonable value in most cases.

The Output Section

Post-Filter: Similar to the Pre-Filter on the input side, the Post-Filter is a pair of first order highpass and lowpass filters which can be applied to the distorted output signal.

Volume: Adjusts the output volume of the distorted signal.

Dry/Wet: Adjusts the mix between the distorted (wet) and original input (dry) signal. Distortion is most often used as an insert effect with 100 percent wet, but in some situations, it may be desirable to be able to mix with the original signal.

have much fun in making music, Robin

Appendix: Function Reference

Arithmetic Operators

- +, -, *, /, ^: addition, subtraction, multiplication, division and raise-to-power operation (works only for integer powers)

Elementary Transcendental Functions

- exp(x): exponential function (raises Euler's number $e = 2.71828...$ to the power of x)
- pow(x,y): power function: x raised to the power of y
- sqrt(x): square root
- ln(x), log(x), logn(x,n): natural, base 10, and base n logarithm
- sin(x), cos(x), tan(x): trigonometric functions of x (expressed in radians)
- asin(x), acos(x), atan(x): inverse trigonometric functions, output is in radians
- sinh(x), cosh(x), tanh(x): hyperbolic functions

Special Mathematical Functions:

- cheby(x,n): Chebyshev polynomial of order n of the input argument x
- gauss(x,m,v): Gaussian distribution function with mean m and variance v (both parameters optional with default values of m=0 and v=1 respectively).

Number Manipulation Functions:

- ipart(x), fpart(x): integer and fractional part
- floor(x), ceil(x): floor and ceiling (closest integer below or above)
- abs(x): absolute value
- sign(x): sign function: $+1$ for $x > 0$, -1 for $x < 0$, 0 for $x = 0$
- mod(x,y): modulo operation: remainder of x divided by y
- quant(x,q): quantize x with quantization interval q
- quantToBits(x,b): quantizes the range between -1...+1 with a quantization interval which corresponds to a resolution of b bits
- clip(x, min, max): clips the input argument x to the range min...max
- softClip(x, min, max, s1, s2): clips the input argument x to the range min...max with tanh-shaped transition regions determined the 'softness' parameters s1 and s2 (assumed to be in the range 0...1). The softness parameters are optional with default values of 0.1.
- clamp(x, min, max): clamps x to the range between min and max, wrapping as needed

Comparison Functions:

- min(x,y,z,...), max(x,y,z,...): returns the value of the smallest or largest of the input arguments
- if(c,t,f): returns the value/result of 't' if 'c' is nonzero, the value/result of 'f' if 'c' is zero
- select(c,n,z,p): returns n for $c < 0$, z for $c = 0$ and p for $c > 0$
- equal(x,y): returns 1 for $x = y$, 0 otherwise
- below(x,y), above(x,y): below returns 1 for $x < y$, 0 otherwise, vice versa for above

Logical Functions:

- and(x,y): returns 1 if both x and y are nonzero, else returns 0
- or(x,y): returns 1 if either x or y is nonzero, else returns 0
- not(x): returns 1 if $x = 0$, else returns 0

Miscellaneous Functions:

- deg(x): converts x from radians to degrees
- rad(x): converts x from degrees to radians
- rescale(x, o1, o2, n1, n2): rescales a number x from the range o1...o2 to the range n1...n2
- poly(x, ...): evaluates the polynomial of x with polynomial coefficients given by the subsequent parameters which are assumed to be in decreasing order (the leftmost coefficient is the multiplier for the highest power of x, the rightmost coefficient is the constant term).

References

For deeper information on waveshaping, you might want to have a look at the article:
http://www.rs-met.com/documents/tutorials/Waveshaping.pdf

Credits

Thanks to Brian A. Vanderburg II for the ExprEval C++ library which empowers the expression evaluator engine used in this plugin. The project can be found on: http://sourceforge.net/projects/expreval