I. Introduction

The Convergence plugin integrates several modulated effects such as Chorus, Flange, Vibrato, Modulated Filter, Tremolo, Panning, and Ring Modulation, all of which are controlled by two LFO's.  The effects are not merely switched on or off, but mixed together as desired by the user using a vector joystick.  The LFO's are also not just summed together, but can be subtracted, multiplied, or divided for an enormous range of possible modulation patterns.  Convergence is intended for advanced producers, especially those who wish to experiment and create unique sounds.

II. LFOs

Convergence's effects are modulated by two LFO units, each made up of sine, saw, triangle, square, noise, and Smoothed Random Sample and Hold (SRSH) waveforms.   The sine, saw, triangle, and square waves are initialized in the plugin's constructor and stored in wave tables.  The noise and random sample and hold waveforms are executed in the plugins "doOscillator" function, which chooses the correct waveform through a case statement and returns the current LFO value for the sample.  The "doOscillator" function also interpolates  from the wave tables.

III. LFO Operations

A slider connected to the gain of the two LFO's controls the overall mix of the two signals.  Once the gain has been applied,the two LFO signals can be added, subtracted, multiplied, or divided.  While the results of several of these functions (ie adding a sine and a saw)  are fairly intuitive, many of the functions can create highly unpredictable results, creating much potential for experimentation.  However, some precautions have to be taken in coding due to the division function.  The LFO mix is limited to 1% to 99% so that there will never be a division by 0.  Also, the SRSH and noise functions also must be prevented from ever returning 0.

IV. Modulated Filter

The filter that is being modulated is a Moog Ladder Filter code block (written by Will Pirkle).  This filter consists of four low pass filters in series with a feedback path.  The Moog filter is reknowned for its lush tone and analog sound, adding much more warmth than a normal digital low pass filter.  Instead of the normal implementation of the filter where the coefficients are only set when the frequency slider is adjusted, the coefficients are set each sample, modulating the frequency based on the LFO depth and current oscillator value.



Fig. 1. Block Diagram of the Moog Ladder Filter

V. Modulated Delay

The modulated delay block is based on a Digital Delay Line Plugin from Will Pirkle's Designing Audio Effect Plug-Ins in C++.  The modulation occurs by updating the read location of the delay line based on the LFO each sample.  The modulated delay can operate in flanger, vibrato, or chorus mode.  The flanger and vibrato both modulate the read address to create a delay from a range of 0-10 mSec.  The main difference is the flanger's wet/dry mix is 50%, creating phasing between the wet and dry signals.  The flanger also has feedback built in for extra resonance.  The chorus has a delay of 7-40 mSec and a 50% wet/dry mix.

VI. Tremolo/Ring Modulator

The last two modulated effects are the ring modulator and tremolo effects.  While both of these effects are based on multiplying an input signal times an LFO signal, there are some slight differences.  The amplitude modulation (tremolo) works with a unipolar modulator while the ring modulator uses a bipolar modulator.  When ring modulation has a modulator of over 20 Hz, audible frequency sidebands occur which are the sum and difference of the carrier and modulator.  In amplitude modulation,  the sidebands are half the amplitude of the carrier.  Therefore, the two effects create slightly different frequency content due to use of unipolar vs. bipolar modulation.

VII. Final Mix

The final output is a sum of the outputs from the modulated filter, modulated delay, tremolo, and ring modulator, multiplied by a scalar value from 0-1, which sum to 1.  This is controlled by a vector joystick with four corners, each corresponding to a different effect.  This feature allows the user to create their own unique blend of the four available effects and to maximize their artistic potential.  The last two processes are a standard wet/dry mix and master gain.  The master gain is important, as some LFO operations, especially those that are subtractive, may greatly attenuate the signal.

VIII. GUI

The user interface was created with Will Pirkle's RackAFX software and is set up to make the signal flow easy to comprehend.  There is a sequence of arrows flowing from the LFO type to LFO operation, vector joystick, and final wet/dry mix.  Separate sections around the main signal flow are for each of the individual effects.  The Modulated Filter section contains  a mini filter response display made out of 6 different meters.  This is created by first finding the difference between the filtered and unfiltered signals, and then using 6 different band pass filters and sending each of the resulting signals to a different display meter.  With a wider range of graphical options, this display would be much higher resolution and the modulation of the filter would be much easier to see.

IX. Future Ideas

Future plans for the plugin include a wider display of LFO options including exponential, logarithmic, and Quasi-Random Sample and Hold algorithms.  Additional operations may be added to the operation section, as well as wave shaping options for the LFO's and output signals.  The option of modulating the LFO rate with an additional LFO will also be added in the future, although this will require a significant rearranging of the code.  Because the plugin often obscures the original signal with interesting results, implementation of the effects as a synth will also be looked into.

References

Pirkle, Will “Designing Audio Effect Plug-Ins in C++” Burlington, MA: Focal Press, 2013.



Fig. 2. Block Diagram for the Entire Plugin