

Tape-Echo with Physically Informed Tape Modelling

Danilo Pešević
MUMT 618 Final Project
Fall 2024

I. INTRODUCTION

The logic of efficiency underlying technological determinism has a hard time taking hold within the field of music recording. Despite the demonstrably higher fidelity of modern digital recording methods (not to mention their greater user-friendliness), many musicians and engineers eschew 24-bit digital-to-analog converters in favor of long obsolete magnetic tape machines. The previously lamented disadvantages of tape recordings have now become desirable sonic qualities: saturation, the deadzone effect, wow and flutter, and tape hiss.

In [1], Chowdhury draws on magnetic recording theory to create a real-time physical tape machine model. This model simulates tape magnetization (i.e., saturation and the deadzone effect) and wow/flutter, as well as loss effects resulting from the magnetization and playback processes – all informed by the physical principles behind magnetic recording. This model was adapted into a VST plugin.

An application of tape modelling is in tape-based effects units, whether they be novel or emulations of existing tape effects units such as the Echoplex or the Roland Space Echo. For example, the Catalinbread Belle Epoch is a digital guitar pedal which emulates the EP-3 Echoplex model [2]. These devices allow users to faithfully reproduce the sounds of magnetic tape, without the unwieldiness and maintenance associated with real tape units.

For my MUMT 618 final project, I aimed to implement the physical tape machine model from [1] in MATLAB and to use it in a tape-echo effect, loosely inspired by the Roland Space Echo SE-201 model in [3]. My implementation will not include

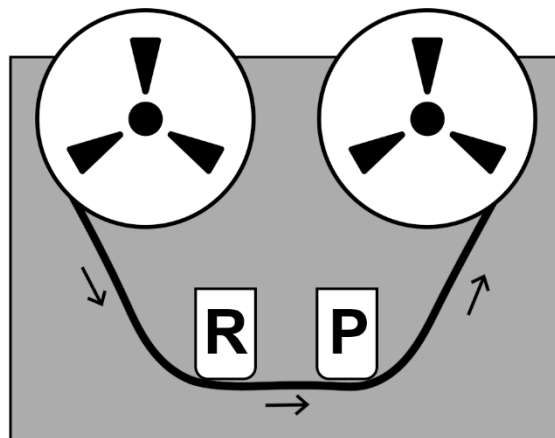


Figure 1. Simplified tape machine diagram.

real-time processing but will instead focus on building a transparent and easy to understand script, conducive to parameter testing and development.

The first section of this paper presents an overview of Chowdhury’s tape model and how it is implemented in my project. To avoid restating the results of that paper, this summary is fairly high-level. Next, the tape-echo effect implementation is briefly explained. The paper concludes with a discussion of the project’s results, challenges, and potential future work.

II. TAPE MACHINE MODELLING

Tape machines, at their simplest, consist of a plastic tape embedded with magnetic particles (domains) passing across two “heads”: the record head, which aligns the magnetic particles by exposing the tape to a magnetic flux corresponding to an audio signal; and the play head, which converts the flux of the tape to a voltage reading, thus reproducing the recorded audio [4]. Figure 1 provides an illustration of this setup.

The nature of magnetic recording results in certain sound characteristics. When all the domains are magnetized, the tape is saturated – increasing the record signal strength past this point does not linearly increase the tape magnetization, resulting in compression. Further, because the domains are particles with a mass, very low signals do not produce a sufficient “coercive force” for domain alignment and thus are not captured on the tape (Chowdhury calls this the “deadzone” effect) [1][5]. This is addressed by adding a high-frequency, high-amplitude bias sinusoid onto the input signal, ensuring that even quiet signals are captured by the magnetic domains, without any audible effect on the recorded signal. Imperfections in steady tape speed cause small changes in pitch, called “wow” when in the range of 0.1 to 10 Hz and “flutter” when in the range of 0.1 to 200 Hz [5].

Chowdhury broadly separates his model into three sections: the record head, tape magnetization, and the play head. These are modelled separately and applied to an input signal in that order; this approach has been reproduced in this project.

2.1. The Record Head

The magnetic field output of the record head, for a given audio input, is given using the Karlqvist medium field approximation from [4]. By assuming that the magnetic field is position-independent, and thus only dependent on time, the expression for magnetic field as a function of input current simplifies to:

$$H(t) = \frac{NEI(t)}{g} \quad (1)$$

where H is the magnetic field in amperes per meter, N is the number of turns of wire around the record head, E is the head efficiency, g is the head gap (space in the tape head used to make the field fringe into the tape) in meters, and I is the instantaneous input current in amperes [1]. This can be discretized to give:

$$\hat{H}(n) = \frac{NE\hat{I}(n)}{g} \quad (2)$$

which is the equation used to model the record head in MATLAB.

2.2. Tape Magnetization

The characteristics arising from the magnetic nature of the tape can be implemented as a hysteresis loop [1]. This is the loop traced by the magnetic flux of a material as the magnetization field changes, capturing the history-dependent nature of magnetization and thus both saturation and the deadzone effect [4].

To find the tape magnetization as a function of the magnetic field, Chowdhury applies the Jiles-Atherton model of magnetization from [6] to construct a non-linear ordinary differential equation in the form:

$$\frac{dM}{dt} = f\left(t, M, \begin{bmatrix} H \\ \dot{H} \end{bmatrix}\right) \quad (3)$$

where M is the magnetization in amperes per meter, t is time in seconds, and \dot{H} is the time-derivative of H , obtained with the trapezoidal rule for derivative approximation [1]. This is solved using the Runge-Kutta fourth-order method to produce an explicit solution:

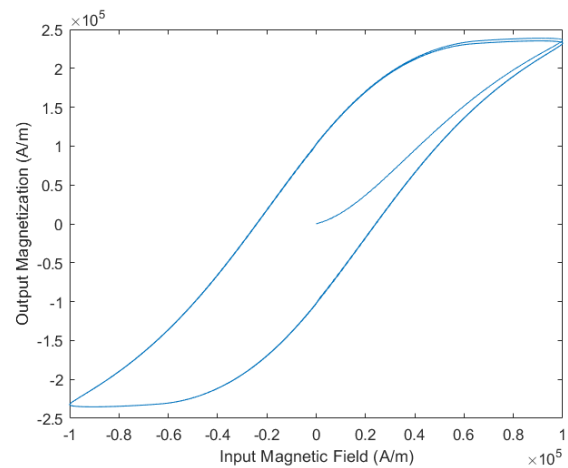


Figure 2. Hysteresis loop simulation with sinusoidal input.

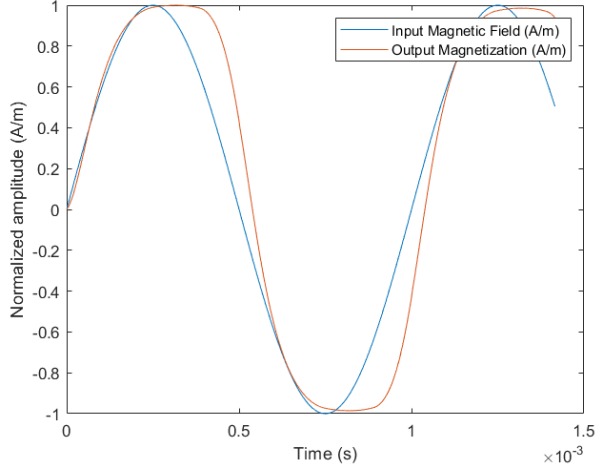


Figure 3. Hysteresis saturation for sinusoidal input.

$$\hat{M}(n) = \hat{M}(n-1) + \frac{k_1}{6} + \frac{k_2}{3} + \frac{k_3}{3} + \frac{k_4}{3} \quad (4)$$

where k_n represents the estimate of the slope of the solution curve at the n th point within the time step [7].

This magnetization model was implemented in MATLAB. Figure 2 shows the hysteresis loop produced for a sinusoidal input signal with frequency 1 kHz and an amplitude of 100 kA/m. Figure 3 shows the saturation of a sinusoidal input as a result of the hysteresis simulation.

2.3. The Play Head

Using the equation for ideal playback voltage as a function of tape magnetization from [4], and again employing the simplifying assumption that the magnetization is position-independent, Chowdhury derives the following simplified time-dependent equation for ideal playback voltage:

$$V(t) = NWEv\mu_0gM(t) \quad (5)$$

where V represents the playback voltage in volts, N the number of turns of wire around the playhead, W the playhead width in meters, E the playhead efficiency, v the tape speed in meters per second, μ_0 the magnetic permeability of free space ($4\pi \times 10^{-7}$ henries per meter), and g the playhead gap [1]. In discrete terms, this becomes

$$\hat{V}(n) = NWEv\mu_0g\hat{M}(n) \quad (6)$$

and this was the equation used to model the playhead in MATLAB.

Additionally, to account for loss effects by the playhead, Chowdhury uses a frequency-dependent formula for loss effects during playback, sourced from [8]:

$$V(t) = V_0(t)[e^{-kd}] \left[\frac{1 - e^{-k\delta}}{k\delta} \right] \left[\frac{\sin(kg/2)}{kg/2} \right] \quad (7)$$

where V_0 is a sinusoidal input in volts, d is the distance between the tape and the play head in meters, δ is the tape thickness in meters, and k is the wavenumber in inverse meters, given as

$$k = \frac{2\pi f}{v} \quad (8)$$

where f is the frequency in hertz. By taking the inverse discrete Fourier transform of the loss equation, a low-pass finite impulse response filter is generated, which can then be used to filter the output of equation 7. A plot of a loss filter generated in MATLAB is given in figure 3.

2.4. Other Considerations

To model tape biasing, based on the implementation in [1], a high frequency sinusoidal signal with an amplitude five times greater than that of the audio

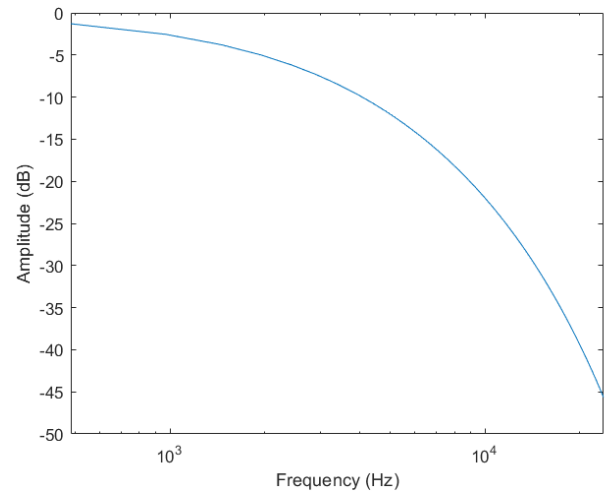


Figure 3. Frequency response of filter modelling playhead loss effect, with $d = 20 \mu\text{m}$, $v = 0.381 \text{ m/s}$, $\delta = 35 \mu\text{m}$, and $g = 3 \mu\text{m}$.

signal was added onto the input. The bias frequency was chosen to be 55 kHz, same as the frequency used in [1] based on that of the Sony TC-260 tape machine Chowdhury chose to model.

Modelling of wow and flutter was accomplished using a vibrato effect, through a sinusoidally modulating delay line function adapted from [9]. This function was used twice in the MATLAB script, once for wow and again for flutter. Compared to the method used in [1], where a train of impulses was sent into the TC-260 and a modulation function was constructed from the measured response, this was a much simpler implementation, but did not capture the time-varying modulation frequency observed in the tape machine.

An aspect of the tape sound that I found missing from Chowdhury’s model was the characteristic white noise heard in the background of low-fidelity tape recordings, otherwise known as tape hiss. It was modelled in this project as a random noise signal added onto the audio signal, since tape noise can be considered a type of Johnson noise, and therefore flat-spectrum white noise is a fair approximation [5].

Finally, based on a recommendation from [1], the input audio was upsampled by a factor of 16 before entering the tape modelling code to avoid instability and aliasing caused by the non-linear magnetization process and the high-frequency bias signal.

III. TAPE-ECHO EFFECT

To achieve the tape-echo effect, a MATLAB function was written which performed a simple echo effect on an input signal, loosely based on the architecture in [3]. The function uses a single delay line and takes as arguments an input signal, a delay time in seconds, and a feedback gain and mix

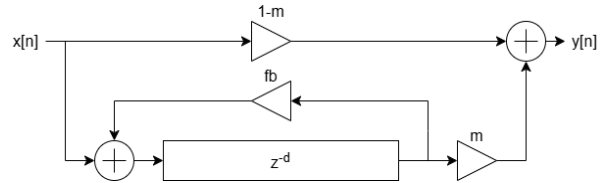


Figure 4. Signal-flow diagram depiction of the echo effect MATLAB function, where fb represents the feedback gain and m the mix gain.

coefficient. Figure 4 provides a block-diagram depiction of the tape-echo effect.

IV. DISCUSSION

This section of the report will discuss the overall MATLAB implementation of the tape-echo effect and its successes and challenges. It will conclude with a discussion of potential future work.

4.1. Results

A block diagram of the overall tape-echo effect with physically informed tape modelling is given in figure 5. The MATLAB script was developed to be modular and transparent, making for easier testing and prototyping of the different model parts and parameters. The code for the hysteresis and playhead filter was adapted from code available on Jatin Chowdhury’s GitHub page [10]. Similarly, the default tape machine parameters were those of the Sony TC-260 and γ -Fe₂O₃ tape, sourced from [1].

The final MATLAB script was successful in emulating a tape-echo effect. Based on subjective testing, the processed audio was comparable to audio recorded on a tape machine. By modifying the tape hiss level, or the echo feedback, for example, users can apply the effects to their liking – results ranging from a barely perceptible touch of tape sound to more experimental, ambient sounds¹. The range of sounds possible with the tape-echo effect makes for an engaging and entertaining experience while playing with the script.

¹ See the audio files included with this report for a demo.

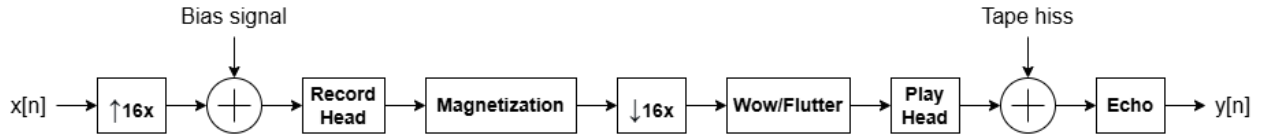


Figure 5. Tape-echo effect block diagram.

While the model does not allow for real-time control, the modular nature of the code makes it easy to change parameters in a systematic way during prototyping, as well as to change the order of signal-flow (for example, applying the noise before the echo and vice versa).

4.2. Challenges and Limitations

While the tape-echo effect was ultimately functional, compared to the results in [1], the implementation of the tape machine model is considerably less realistic. Adapting parts of the MATLAB script from Chowdhury’s code proved challenging. His code is written in a manner that was difficult to parse due to my limited knowledge of C/C++ and VST plugin architecture. For that reason, I chose to implement wow and flutter in a more simplified manner. Additionally, the hysteresis loop in particular proved difficult to implement and my code required a very arbitrary addition to work (dividing the signal by 100 before it entered the magnetization section of the script). In general, while the magnetization code did successfully emulate tape saturation, the results were not identical to those of [1]. In my testing, I was not able to produce the deadzone effect and the hysteresis loops appeared different from those in [1] at lower magnetic field amplitudes.

Additionally, the MATLAB script only works in mono and performs a simple normalization before exporting the processed audio to avoid clipping. This limits the versatility of the effect and may impact the signal dynamics in an undesirable way.

A major limitation of this project is that it is a MATLAB script. Even without real-time controls, to be able to use the tape-echo effect as a plugin in a digital audio workstation, or even in a more

accessible format such as a Python script, would make the project much more usable in a music production context.

4.3. Future Work

The next step for this tape-echo effect would be to develop real-time processing code, either for a VST plugin like in [1] or for inclusion in a hardware effects unit. While it is nice to be able to control every parameter of a tape machine, I imagine the average user would not be interested in changing tape width or head gap. If the script were adapted for an effects unit, the controls should be simplified to make for a more intuitive and less overwhelming experience. The MATLAB script would be very useful for the prototyping stage, though the current script would also benefit from more intuitive parameter values. For example, tape noise level could be specified as the number of decibels below the signal strength, or wow/flutter could be specified as percentages (as wow/flutter measurements typically are).

Before a real-time version is developed, the current script should be evaluated more systematically, and issues such as the lack of deadzone effect must be resolved. Further, it should be determined whether it is necessary to improve the subsystems within the model. This could involve finding participants to subjectively evaluate the sounds produced by the model and provide feedback on where they believe it is lacking.

V. CONCLUSION

This project developed a tape-echo effect in MATLAB which emulated the sound characteristics of tape recordings, including saturation, wow/flutter, and tape hiss. The implementation adapted the real-time, physically

informed tape machine model from [1], simplifying the wow/flutter effect and adding tape hiss. The processed audio was then fed into an echo effect algorithm, producing a tape-echo effect akin to an Echoplex or Roland Space Echo.

The project successfully produced a convincing tape-echo effect, with adjustable model parameters and echo controls for customization and model prototyping. Before the effect can be used in a real music production context, the model should be improved by addressing the lack of a deadzone effect in the magnetization model and by adding real-time processing.

VI. REFERENCES

- [1] J. Chowdhury, "Real-time physical modelling for analog tape machines," in *Proc. 22nd International Conference on Digital Audio Effects*, 2019.
- [2] "Belle Epoch (Black and Silver)," *Catalinbread Effects*. Accessed Nov. 30, 2024. [Online]. Available: <https://catalinbread.com/products/belle-epoch>
- [3] J. Downing and C. Terjesen, "Real-time digital modeling of the Roland Space-Echo," class project, Dept. Elect. Eng., Univ. of Rochester, 2016.
- [4] H. N. Bertram, *Theory of Magnetic Recording*. Cambridge, MA, USA: Cambridge Univ. Press, 1994.
- [5] M. Camras, *Magnetic Recording Handbook*. New York, NY, USA: Van Nostrand Reinhold Company, 1988.
- [6] M. Holters and U. Zölzer, "Circuit simulation with inductors and transformers based on the Jiles-Atherton model of magnetization," In *Proc. 19th International Conference on Digital Audio Effects*, 2016.
- [7] E. Cheever, "Fourth Order Runge-Kutta," *Swarthmore.edu*. Accessed Oct. 17, 2024. [Online]. Available: <https://lpsa.swarthmore.edu/NumInt/NumIntFourth.html>
- [8] J. Kadis, *The Science of Sound Recording*. Waltham, MA, USA: Focal Press, 2012.
- [9] P. Dutilleux et al., *Digital Audio Effects*, 2nd edition. Hoboken, NJ, USA: John Wiley & Sons, 2011, ch. 2.
- [10] J. Chowdhury, "AnalogTapeModel." GitHub. Accessed Nov. 30, 2024. [Online]. Available: <https://github.com/jatinchowdhury18/AnalogTapeModel>