

Building versatile audio crossfades using recurrence relations

Lucian Lupşa-Tătaru

Faculty of Electrical Engineering and Computer Science,
“Transilvania” University of Braşov, Bd. Eroilor No. 29, Braşov, Romania,
lupsa@programmer.net, lucian.lupsa@unitbv.ro

Abstract. The audio crossfade is a well-known technique used to perform smooth transitions between two audio contents (segments) by fading out one content and, simultaneously, fading in the other. This effect is commonly used in music production, film and TV sound design, podcasting, creation of simulation software, media development and entertainment software, respectively. In order to help with blending different styles, tempos, and even genres smoothly, and to create a more natural and seamless flow in audio, we propose a procedure of building highly versatile audio crossfade effects, adaptable across different scenarios. To facilitate the implementation of real-time crossfades, each fade envelope is generated by employing a recurrence relation that yields the current value of the audio volume as rational function of its previous value.

Keywords. Real-time audio, audio crossfade, recurrence relation, timing events, adjustable fade out, adjustable fade in.

1. Introduction

Crossfading is especially useful when transitioning between different sources of audio, e.g. when one needs to blend background music with speech. Within multimedia projects, crossfades are frequently used to make the sound experience immersive. Audio crossfades are often used to preserve the emotional flow, allowing the listener to ease into a new mood or tempo more naturally. From an artistic perspective, crossfading enables the experimentation with various textures and sonic qualities. In essence, crossfading enhances the overall audio experience by smoothing out the boundaries between audio contents and reducing disruptions in the sound [1]-[6].

The audio crossfade effect involves two audio contents (segments) that are being transitioned between. During the crossfade process, the volume of one content is gradually decreased, i.e. faded out, whilst the volume of the other audio content is gradually increased, i.e. faded in. The two audio contents overlap in the course of the crossfade, meaning that both contents are audible at the same playback position within the crossfade interval. The overlap that occurs will smooth the transition between the sounds since one content fades out and the other fades in [6]-[10].

The crossfade duration (length) can vary depending on the desired effect. Anyhow, a long crossfade always leads to a more subtle, unobtrusive blending of sounds. When the audio gradually shifts in a straight line from one audio content to the other, one receives the widely used linear crossfade. However, various DAWs (digital audio workstations) offer crossfade curves that are non-linear, such as exponential or logarithmic curves. These curve shapes, obtained by valuating intricate transcendental functions of playback position, can make the crossfades sound more natural or tailored to the mood or pacing of the audio.

To preserve the suitability with real-time processing, the crossfade implementation in [11] makes use of rational functions to shape the crossfade curves, where the position within the crossfade interval is selected as the input of the employed rational functions. Since most programming languages and software frameworks currently permit the calling of timing events, it seems quite natural to create highly versatile audio crossfades by adopting recurrence relations to customize the crossfade curves [12]-[15]. This will ease the implementation and will give users flexibility by allowing for employing adjustable fade curves and by enabling the controlling of the length of overlap between the two audio contents, having in view that one may want a longer overlap for smoother transitions or a shorter one for quick switches.

To shape the crossfade envelopes, we will use the recurrence relation advanced in [15]. Thus, the current value of the audio volume is here the return value of a rational function with the input being just the previous value of the audio volume:

$$v_{n+1} = \frac{av_n - h}{bv_n - c}, \quad n \geq 0 \quad (1)$$

wherein the coefficients are as follows [15]:

$$a = \beta h - \alpha\beta + \gamma, \quad b = \beta^2 h, \quad c = \beta h + \alpha\beta - \gamma, \quad (2)$$

$$\alpha = \frac{\tau_f v_0}{v_0 + \left(1 - \frac{1}{\varepsilon}\right) v_f}, \quad \beta = \frac{2 - \frac{1}{\varepsilon}}{v_0 + \left(1 - \frac{1}{\varepsilon}\right) v_f}, \quad \gamma = \frac{\tau_f}{v_0 + \left(1 - \frac{1}{\varepsilon}\right) v_f}. \quad (3)$$

Within recurrence relation (1), the output v_{n+1} designates the audio volume current value, the input v_n stands for the audio volume previous value while h , also interfering in (2), represents the time step adopted for discretization. Furthermore, relations (3) highlight the fade length τ_f , the initial audio volume v_0 , the final audio volume v_f and the shape parameter ε , which is directly proportional to the playback position at which the audio volume arrives at its mean value [15].

2. Recurrence relation for performing the audio fading out

As previously mentioned, a crossfade audio effect involves fading out the volume of one audio content while simultaneously fading in the volume of another.

In order to receive a fade out effect via recurrence relation (1), within the set of expressions (3), one has to replace:

$$v_0 = v_m, \quad v_f = 0, \quad (4)$$

with v_m designating the initial audio volume, wherefrom the fade out effect is triggered. With (3) and (4), one obtains:

$$\alpha_O = \tau_f, \quad \beta_O = \frac{2 - \frac{1}{\varepsilon_O}}{v_m}, \quad \gamma_O = \frac{\tau_f}{v_m}, \quad (5)$$

$$\alpha_O \beta_O - \gamma_O = \tau_f \frac{2 - \frac{1}{\varepsilon_O}}{v_m} - \frac{\tau_f}{v_m} = \left(1 - \frac{1}{\varepsilon_O}\right) \frac{\tau_f}{v_m}, \quad (6)$$

wherein subscript “O” denotes the fading out process.

Thus, for the specific case of audio fading out, the coefficients (2), interfering in the recurrence relation (1), obtain the appropriate expressions:

$$a_o = \beta_o h - (\alpha_o \beta_o - \gamma_o) = \left(2 - \frac{1}{\varepsilon_o}\right) \frac{h}{v_m} - \left(1 - \frac{1}{\varepsilon_o}\right) \frac{\tau_f}{v_m}, \quad (7)$$

$$b_o = \beta_o^2 h = \left(2 - \frac{1}{\varepsilon_o}\right)^2 \frac{h}{v_m^2}, \quad (8)$$

$$c_o = \beta_o h + (\alpha_o \beta_o - \gamma_o) = \left(2 - \frac{1}{\varepsilon_o}\right) \frac{h}{v_m} + \left(1 - \frac{1}{\varepsilon_o}\right) \frac{\tau_f}{v_m}. \quad (9)$$

One observes that coefficients (7)-(9) are explicitly given just in terms of the parameters of the effect i.e. the duration (length) τ_f and the shape parameter

$$\varepsilon_o = \frac{\tau_{\mu,0}}{\tau_f}, \quad \varepsilon_o \in (0, 1), \quad (10)$$

which is here directly proportional to the position $\tau_{\mu,0}$ at which the audio volume has the value of $v_m/2$ that is the audio volume mean value in the course of fading out.

Hence, the fading out process has to be carried out by means of recurrence relation:

$$v_{o,n+1} = \frac{a_o v_{o,n} - h}{b_o v_{o,n} - c_o}, \quad n \geq 0, \quad (11)$$

wherein h remains the discretization step size. If one sets

$$h = \tau_f/2$$

then a single invocation of (11) yields precisely the audio volume at the halfway point:

$$v_{o,1} = \frac{a_o v_m - \frac{\tau_f}{2}}{b_o v_m - c_o} = \frac{\left(2 - \frac{1}{\varepsilon_o}\right) \frac{\tau_f}{2} - \left(1 - \frac{1}{\varepsilon_o}\right) \tau_f - \frac{\tau_f}{2}}{\left(2 - \frac{1}{\varepsilon_o}\right)^2 \frac{\tau_f}{2} - \left(2 - \frac{1}{\varepsilon_o}\right) \frac{\tau_f}{2} - \left(1 - \frac{1}{\varepsilon_o}\right) \tau_f} v_m = \varepsilon_o v_m. \quad (12)$$

3. Recurrence relation for performing the audio fading in

A programmatic approach to creating a highly versatile fade in audio effect can be achieved using relations (1)-(3), wherein one has to take into account that the audio volume gradually increases from 0 (silence) up to the final value v_m , where the audio becomes fully audible. Hence, with conditions:

$$v_0 = 0, \quad v_f = v_m, \quad (13)$$

the coefficients (3) obtain the appropriate expressions:

$$\alpha_I = 0, \quad \beta_I = \frac{2 - \frac{1}{\varepsilon_I}}{\left(1 - \frac{1}{\varepsilon_I}\right) v_m}, \quad \gamma_I = \frac{\tau_f}{\left(1 - \frac{1}{\varepsilon_I}\right) v_m}, \quad (14)$$

with subscript “I” denoting the fading in process.

Besides, having in view (14), the three coefficients of recurrence relation (1) can now be provided as expressions in terms of the fade in effect parameters. Thus, with τ_f as the effect duration and ε_I as the shape parameter, one gets both the set of coefficients in (1), namely

$$a_I = \beta_I h + \gamma_I = \frac{2\varepsilon_I - 1}{\varepsilon_I - 1} \frac{h}{v_m} + \frac{\varepsilon_I}{\varepsilon_I - 1} \frac{\tau_f}{v_m}, \quad (15)$$

$$b_I = \beta_I^2 h = \left(\frac{2\varepsilon_I - 1}{\varepsilon_I - 1} \right)^2 \frac{h}{v_m^2}, \quad (16)$$

$$c_I = \beta_I h - \gamma_I = \frac{2\varepsilon_I - 1}{\varepsilon_I - 1} \frac{h}{v_m} - \frac{\varepsilon_I}{\varepsilon_I - 1} \frac{\tau_f}{v_m} \quad (17)$$

and the playback position, within the fading in process, at which the audio volume reaches the mean value of $v_m/2$, i.e.

$$\tau_{\mu, I} = \varepsilon_I \tau_f, \quad \varepsilon_I \in (0, 1). \quad (18)$$

Thus, the recurrence relation used here to build the fade in envelope is the following:

$$v_{I, n+1} = \frac{a_I v_{I, n} - h}{b_I v_{I, n} - c_I}, \quad n \geq 0. \quad (19)$$

Based on (19), for

$$h = \tau_f/2, \quad v_{I, 0} = 0, \quad (20)$$

one straightforwardly receives the audio volume at the effect midpoint in terms of shape parameter and the final volume:

$$v_{I, 1} = \frac{a_I v_{I, 0} - \frac{\tau_f}{2}}{b_I v_{I, 0} - c_I} = \frac{\tau_f}{2c_I} = \frac{\tau_f}{\frac{2\varepsilon_I - 1}{\varepsilon_I - 1} \frac{\tau_f}{v_m} - \frac{2\varepsilon_I}{\varepsilon_I - 1} \frac{\tau_f}{v_m}} = (1 - \varepsilon_I)v_m. \quad (21)$$

4. Implementation and effect versatility

The audio crossfade envelopes are used in order to smoothly blend the two audio segments, preventing abrupt changes or glitches when switching between them. The common types of crossfade envelopes include the linear envelope, the logarithmic envelope, and the exponential crossfade envelope. The linear crossfade envelope creates a straight-line transition between the two audio sections, typically going from one audio source to the other in a uniform manner over time. This is the most common type of crossfade envelope. The logarithmic and the exponential crossfade envelopes are non-linear and can make the transition sound more natural or more dynamic and dramatic [8]-[10].

From the technical point of view, the linear crossfade reflects a simple approach, where the audio volume fades in and out linearly over a fixed time interval whilst the logarithmic or the exponential crossfades use logarithmic or exponential curves for carrying out both the fading in and fading out processes. Each of these types of crossfade envelopes can be used depending on the desired outcome of the transition, whether that is a smooth fade, a dramatic shift, or a more organic sound.

Recurrence relations (11), (19) allow the implementation of custom shaped crossfade envelopes, with the crossfade curves being adjusted according to specific needs. The first audio section is faded out, from volume v_m , via relation (11) while the second audio section is faded in, up to volume v_m , by

calling (19). Thus, with recurrence relations (11) and (19), the volume envelopes of the two audio sections being transitioned can be manipulated individually by setting appropriate values for the shape parameters ε_0 and ε_I , which occur in expressions of coefficients (7)-(9) and (15)-(17), respectively.

The more general recurrence relation (1) has previously been validated for the quality of being highly effective for carrying out real-time adjustable fades by means of an implementation in plain JavaScript, put forward in [15] with the purpose of immediate experimentation. Having this in view, it should be recalled that an implementation in pure JavaScript of the procedure advanced here has to include the following statements:

```

var a0, b0, c0, aI, bI, cI;
function recRelCoeff( tauF, vM, eps0, epsI, hSize ) {
    var h_to_vM = hSize / vM;
    var tauF_to_vM = tauF / vM;
    var m1 = 2.0 - 1.0 / eps0; var m2 = m1 - 1.0;
    var m4 = epsI - 1.0;
    var m3 = ( epsI + m4 ) / m4; m4 = epsI / m4;
    a0 = m1 * h_to_vM - m2 * tauF_to_vM;
    b0 = m1 * m1 * h_to_vM / vM;
    c0 = m1 * h_to_vM + m2 * tauF_to_vM;
    aI = m3 * h_to_vM + m4 * tauF_to_vM;
    bI = m3 * m3 * h_to_vM / vM;
    cI = m3 * h_to_vM - m4 * tauF_to_vM;
}
function updateVolO( vol0, hSize ) {
    var newVol0 = ( a0 * vol0 - hSize ) / ( b0 * vol0 - c0 );
    if ( newVol0 < 0.0 ) { newVol0 = 0.0; }
    return newVol0;
}
function updateVolI( volI, vM, hSize ) {
    var newVolI = ( aI * volI - hSize ) / ( bI * volI - cI );
    if ( newVolI > vM ) { newVolI = vM; }
    return newVolI;
}

```

wherein functions “updateVolO()” and “updateVolI()” represent just the implementation of recurrence relations (11), (19) whilst function “recRelCoeff()” enables the computation of the set of coefficients provided by expressions (7)-(9) and (15)-(17), respectively, and designated by global variables “aO”, “bO”, “cO”, “aI”, “bI”, “cI”.

The 4 s transitions portrayed in Fig. 1 till Fig. 7 highlight the versatility of the adaptable approach to crossfade building advanced in the present investigation. Since the crossfade length τ_f stands out as parameter in (7), (9), (15), (17), one can lightly experiment with transitions of different durations.

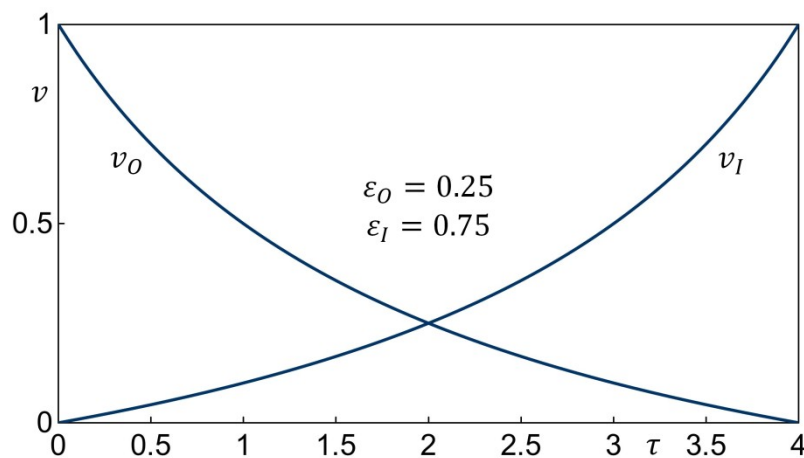


Fig. 1. Depiction of the audio crossfade received for parameters:
 $\tau_f = 4$ s, $v_m = 1$, $\varepsilon_0 = 0.25$, $\varepsilon_I = 0.75$.

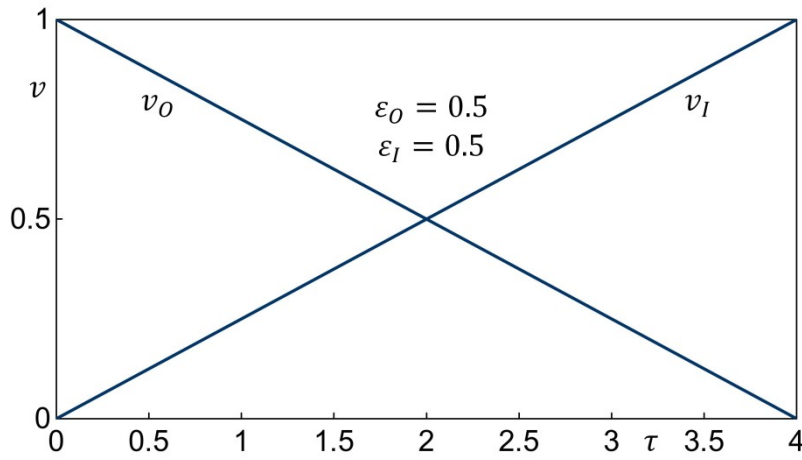


Fig. 2. Depiction of the audio crossfade received for parameters:
 $\tau_f = 4 \text{ s}$, $v_m = 1$, $\varepsilon_O = 0.5$, $\varepsilon_I = 0.5$.

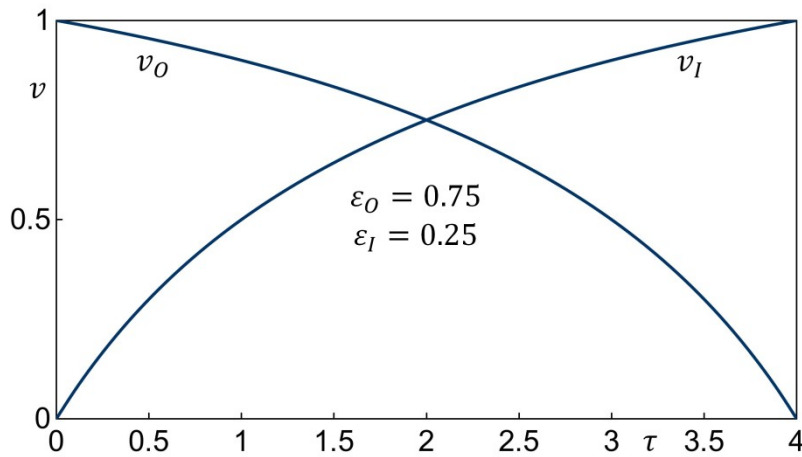


Fig. 3. Depiction of the audio crossfade received for parameters:
 $\tau_f = 4 \text{ s}$, $v_m = 1$, $\varepsilon_O = 0.75$, $\varepsilon_I = 0.25$.

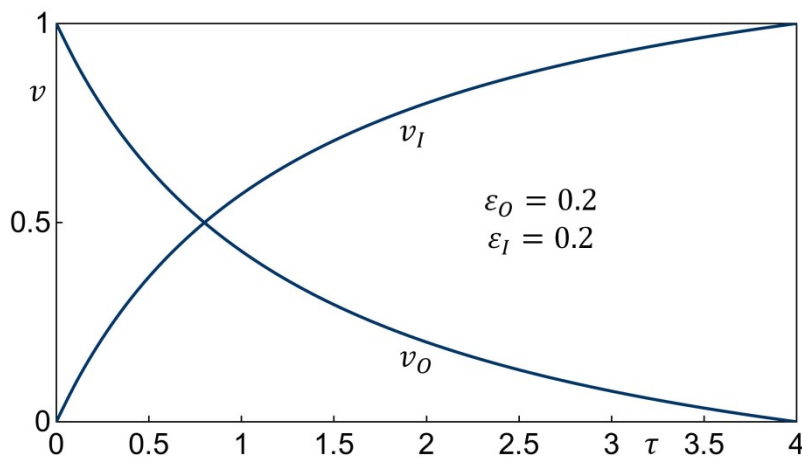


Fig. 4. Depiction of the audio crossfade received for parameters:
 $\tau_f = 4 \text{ s}$, $v_m = 1$, $\varepsilon_O = 0.2$, $\varepsilon_I = 0.2$.

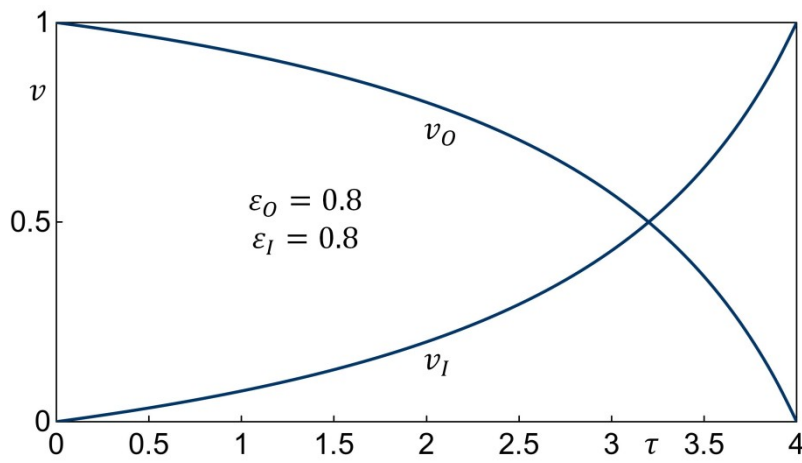


Fig. 5. Depiction of the audio crossfade received for parameters:
 $\tau_f = 4$ s, $v_m = 1$, $\varepsilon_O = 0.8$, $\varepsilon_I = 0.8$.

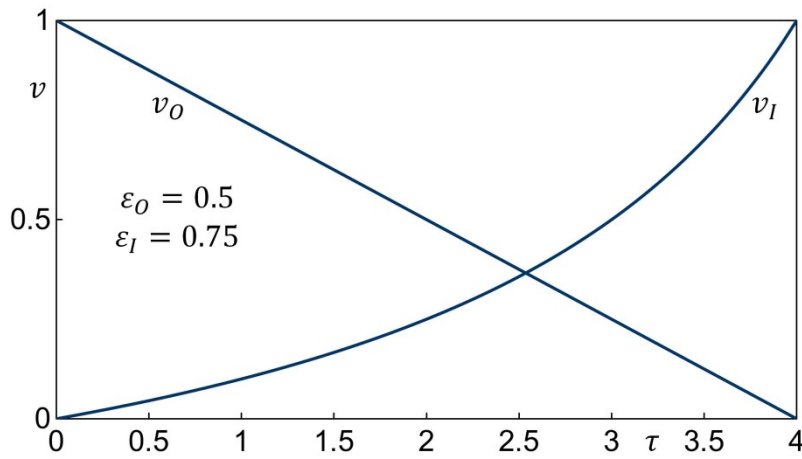


Fig. 6. Depiction of the audio crossfade received for parameters:
 $\tau_f = 4$ s, $v_m = 1$, $\varepsilon_O = 0.5$, $\varepsilon_I = 0.75$.

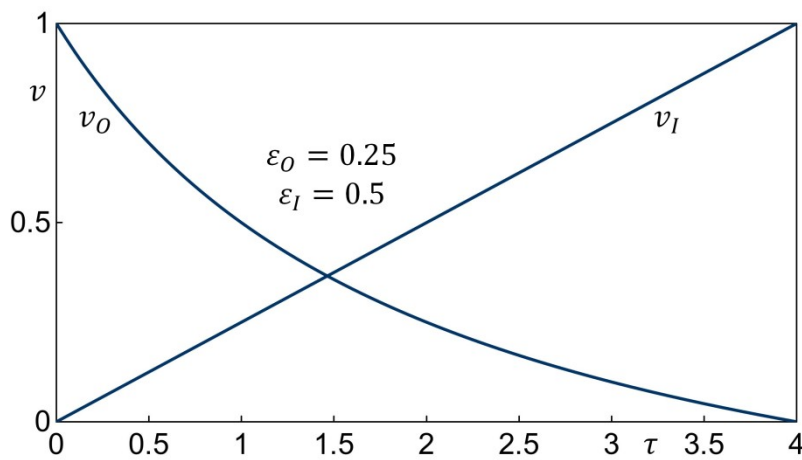


Fig. 7. Depiction of the audio crossfade received for parameters:
 $\tau_f = 4$ s, $v_m = 1$, $\varepsilon_O = 0.25$, $\varepsilon_I = 0.5$.

It can be observed that the expressions of coefficients (7)-(9) incorporate the shape parameter ε_O while the expressions of coefficients (15)-(17) encompass the shape parameter ε_I . This leaves room for experimentation with different fade out and fade in shapes to construct the crossfade envelopes. Hence, the envelopes of the crossfades depicted in Fig. 1 up to Fig. 7 have been built independently.

One perceives that the crossfade illustrated in Fig. 1 will act similar to the exponential crossfade. According to (10), in Fig. 1, the playback position at which the audio volume of the faded out section reaches the mean value of 0.5 is as follows:

$$\tau_{\mu,O} = \varepsilon_O \tau_f = 0.25 \times 4 \text{ s} = 1 \text{ s} .$$

On the other hand, according to (18), within Fig. 1, the playback position at which the audio volume of the faded in section has the value of 0.5 is

$$\tau_{\mu,I} = \varepsilon_I \tau_f = 0.75 \times 4 \text{ s} = 3 \text{ s} .$$

For shape parameters $\varepsilon_O = \varepsilon_I = 0.5$, one receives the linear crossfade of Fig. 2 while, for the shape parameters $\varepsilon_O = 0.75$ and $\varepsilon_I = 0.25$, the resulted effect, depicted in Fig. 3, appears to be similar to the logarithmic crossfade.

It has to be pointed out that, for each of the crossfades illustrated in Fig. 1 till Fig. 3, the volume at the halfway point of the faded out section, yielded by relation (12), is equal to the volume (21) of the faded in section, also computed at the transition midpoint. Thus, for the effects portrayed in Fig. 1, Fig. 2 and Fig. 3, we have $\varepsilon_O = 1 - \varepsilon_I$.

The crossfade curves portrayed in Fig. 4 and Fig. 5, respectively, intersect precisely at the playback position at which the audio volume of each of the two sections reaches the mean value of 0.5. For the case of Fig. 4, the intersection position is $\tau_{\mu,O} = \tau_{\mu,I} = 0.8 \text{ s}$, while for the case of Fig. 5, the playback position of intersection is $\tau_{\mu,O} = \tau_{\mu,I} = 3.2 \text{ s}$. On the other side, the crossfades of Fig. 6 and Fig. 7 have been received by applying linear fades accompanied by non-linear ones in an attempt to uncover the audio content of specific sections.

5. Conclusion

The purpose of performing an audio crossfade is to create a seamless blend between sounds, avoiding abrupt transitions or cuts that could disrupt the flow of audio contents. An audio crossfade involves a gradual decreasing of the volume of one audio segment accompanied by an increasing of the volume of another. This results in a smooth transition, where the outgoing track fades out and the incoming track fades in. Thus, to apply the crossfade, one has to manipulate the volume envelope of both tracks. The volume of the first track is gradually reduced i.e. the track is faded out whilst the audio volume of the second track is gradually increased i.e. the track is faded in. The duration (length) of the crossfade should be adjustable in order to control how long the transition lasts. A short crossfade (e.g., of 1 s) will create a quick fade, being suitable for energetic mixes, while a longer crossfade (e.g., of 5 s) will produce a more gradual, subtle transition.

Traditionally, the crossfade techniques are based on employing linear, logarithmic or exponential envelopes, where the logarithmic and the exponential crossfade envelopes are applied by means of valuating intricate transcendental functions of playback position. Not only to add versatility but also to lessen the labour of real-time implementation, we advance a crossfade technique based on custom shaped envelopes, where the crossfade curves can be adjusted according to the specific needs of the audio being transitioned. In the paper, it is shown that the design of custom crossfade envelopes can be plainly achieved by using recurrence relations, with the current value of the audio volume as rational function of its previous value only. By experimenting with various shape parameters values in order to construct the crossfade fade in and fade out envelopes, we have illustrated different transitions, which emphasize the high degree of versatility that can be achieved by adopting the suggested programmatic approach.

References

- [1] W. JACKSON: Digital Audio Editing Fundamentals. Get Started with Digital Audio Development and Distribution. Berkeley, CA, USA, Apress Media, 2015.
<https://doi.org/10.1007/978-1-4842-1648-4>
- [2] V. COSTELLO: Multimedia Foundations. Core Concepts for Digital Design. New York, NY, USA, Routledge, 2016.
- [3] K.C.C. YANG, et al.: Cases on Immersive Virtual Reality Techniques (*Advances in Multimedia and Interactive Technologies*). Hershey, PA, USA, IGI Global, 2019.
- [4] R. L. BLEIDT et al.: Building the world's most complex TV network: a test bed for broadcasting immersive and interactive audio. *SMPTE Motion Imaging Journal*, **126** (5), 26-34 (2017).
<https://doi.org/10.5594/JMI.2017.2698618>
- [5] M. SWEET: Writing Interactive Music for Video Games. A Composer's Guide. Upper Saddle River, NJ, USA, Addison-Wesley Professional, 2014.
- [6] J. D. REISS, A. McPHERSON: Audio Effects. Theory, Implementation and Application. Boca Raton, FL, USA, CRC Press, 2015.
- [7] A. U. CASE: Sound FX. Unlocking the Creative Potential of Recording Studio Effects. Burlington, MA, USA, Focal Press, 2007.
- [8] S. LANGFORD: Digital Audio Editing. Correcting and Enhancing Audio in Pro Tools, Logic Pro, Cubase, and Studio One. Burlington, MA, USA, Focal Press, 2014.
- [9] C. SCHRODER: The Book of Audacity. Record, Edit, Mix, and Master with the Free Audio Editor. San Francisco, CA, USA, No Starch Press, 2011.
- [10] The Audacity Team: Audacity(R) Free, Open Source, Cross-platform Audio Software. Audacity Manual, 2025. Creating a Crossfade.
https://manual.audacityteam.org/man/creating_a_crossfade.html
- [11] L. LUPSA-TATARU: Audio cross-fade implementation via rational functions. *Technium: Romanian Journal of Applied Sciences and Technology*, **3** (5), 39-48 (2021).
<https://doi.org/10.47577/technium.v3i5.3881>
- [12] A. BURNS, A. WELLINGS: Timing events and execution-time control. In: *Concurrent and Real-Time Programming in Ada*. Cambridge, Cambridge University Press, 2007, 361-390.
<https://doi.org/10.1017/CBO9780511611230.016>
- [13] T. ABDELLATIF, J. COMBAZ, J. SIFAKIS: Rigorous implementation of real-time systems - from theory to application. *Mathematical Structures in Computer Science*, **23** (4), 882-914 (2013).
<https://doi.org/10.1017/S096012951200028X>
- [14] D. FLANAGAN: JavaScript: The Definitive Guide, 7th Edition: Master the World's Most-Used Programming Language. Sebastopol, CA, USA, O'Reilly Media, 2020.
- [15] L. LUPSA-TATARU: Recurrence relation for real-time audio fading. *Technium: Romanian Journal of Applied Sciences and Technology*, **12**, 15-22 (2023).
<https://doi.org/10.47577/technium.v12i.9328>