IMPLEMENTATION OF LOUDNESS IN A DIGITAL INSTRUMENT

Hans G. Kaper[1] and Sever Tipei[2]

[1] Mathematics and Computer Science Division
Argonne National Laboratory, Argonne, Illinois, USA

[2] Computer Music Project, School of Music
University of Illinois at Urbana–Champaign, Urbana, Illinois, USA

Draft — May 18, 2001

1 Introduction

A digital instrument requires the formalization of the concept of sound and its properties as well as the implementation of this formal framework in mathematical algorithms. The algorithms are the foundation of the computer programs that make up the instrument.

One of the most challenging concepts to formalize is that of *loudness*. Sound is transmitted through sound waves—pressure variations that cause the eardrums to vibrate. This is the easy part of the problem. But loudness *perception* has as much to do with the amount of energy that the sound wave carries (its intensity) as with the processing of this energy that takes place in the ear and the brain of the listener once the sound wave has hit the eardrums. Although a number of psychoacoustic experiments focusing on loudness have been performed, little research has been conducted into ways of precisely controlling the perceived loudness of sounds regardless of their frequency or the complexity of the waveform involved.

When a number of sounds are integrated in a piece, each sound being described by a complex wave which in turn is the result of the summation of a large number of partials, hundreds of waves interact with one another. Specifying the amplitude of such an aggregate will not result in the desired perceived loudness, because the human ear does not respond to intensity in a uniform way across the frequency domain and the response depends upon the specific distribution of the frequencies of the constituent partials over the range of audible frequencies.

Another practical problem occurs when sounds with a wide dynamic range are combined in a piece. Often, the computed amplitude exceeds the maximum allowable amplitude of the instrument (overflow, resulting in "clipping"). Scaling the entire audiowave may allow maximum perceived loudness for the high-amplitude fragments without clipping, but most likely will also result in a scaling down of the soft sounds to the point that they disappear in the system noise.

In this article we discuss these issues and their solution as implemented in the loudness and anticlip routines of DISCO (Digital Instrument for Sonification and Composition). DISCO is a collection of software written in C++, which is being developed jointly by the authors at Argonne National Laboratory and the University of Illinois at Urbana–Champaign.

DISCO and its predecessor DIASS (Digital Instrument for Additive Sound Synthesis) have been characterized as the Rolls Royces of digital instruments because of their sophistication and the amount of control the user has over their functionalities. They enable the user to generate sounds of arbitrary complexity by the method of additive synthesis. In principle, there is no limit to the number of partials in a sound or their complexity, and the parameters defining a partial can be changed dynamically, either individually or collectively. Of course, the more complex the sound, the more detailed the specifications that define the sound.

DISCO takes the complete set of specifications and produces a score file, which can be interpreted by an acoustic instrument or converted into a sound file. *Perceived loudness* is one of the (dynamic) degrees of freedom of a sound that are under the control of the user. The loudness routines in DISCO ensure that each sound has the desired perceived loudness. The anticlip routines addresses the problem of scaling the score file so the resulting sound file is free from overflow ("clipping") without affecting the relative perceived loudness of the various sounds in the piece.

In Section 2, we address the formal definition of loudness and its implementation in the loudness routines of DISCO. In Section 3, we discuss the anticlip routines.

2 Loudness

Since DISCO uses the method of additive sound synthesis, we think of a sound as a superposition of *partials*—sinusoidal waves, each with a well defined frequency and amplitude (which may vary with time). Hence, the definition of the perceived loudness of a sound builds upon the definition of the loudness of a single partial (pure tone). We discuss the case of a pure tone in Section 2.1, the more complicated case of a sound all of whose constituent partials lie within a critical band in Section 2.2, and the general case of a complex sound in Section 2.3.

2.1 Pure Tones

2.1.1 Definitions

The definition of the loudness of a pure tone is based on the energy flow or *intensity* of the sound wave. The intensity, denoted by I, is a function of the *sound pressure level* (SPL) of the wave. The SPL is a dimensionless quantity, which is obtained by dividing the average pressure variation Δp by some reference value Δp_0. (The *average* variation of a quantity that varies sinusoidally in time is equal to the amplitude of the variation divided by $\sqrt{2} = 1.41$.) The expression for I is

$$I = 20 \log_{10}(\Delta p / \Delta p_0). \qquad \boxed{\text{p2i}} \tag{2.1}$$

The unit of I is the *decibel* (dB). The reference value Δp_0 is usually identified with the average pressure variation of a traveling wave of 1,000 Hz at the treshold of hearing, $\Delta p_0 = 2.0 \cdot 10^{-5}$ newton/m^2.

The sensation of loudness is strongly frequency dependent. For example, although an intensity of 50 dB at 1,000 Hz is considered *piano*, the same intensity is barely audible at 60 Hz. To produce a given loudness sensation, we need a much higher intensity at low frequencies than at high frequencies.

In 1933, Fletcher and Munson [1] published their now famous diagram representing the results of a number of loudness-matching experiments under free-field conditions of listening; see Fig. 1. (A free field is an environment in which there are no reflections.) The *equal-loudness curves*—curves showing the intensity required to make (single continuously sounding) pure tones sound equally loud—show clearly that, to be perceived as equally loud, very low and very high frequencies require much higher intensities than frequencies in the middle range of the spectrum of audible sounds.

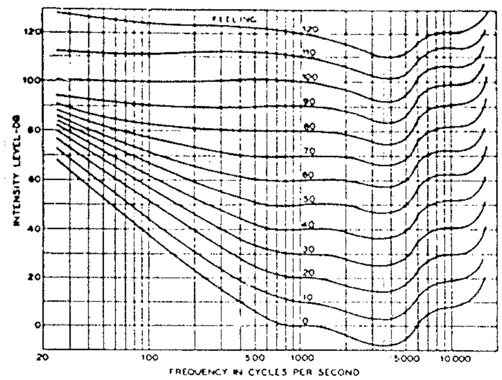

Figure 1: Equal-loudness curves [1, Fig. 4].

The *(physical) loudness* L_p of a Fletcher–Munson curve is identified with the value of the intensity I at the reference frequency of 1,000 Hz. The unit of L_p is the *phon* (plural *phons*). The Fletcher–Munson curves range from a loudness level of 0 phons (threshold of hearing) to 120 phons (limit of pain) over a frequency range from 25 to 16,000 Hz. They are reproduced in the monographs of Jeans [3, Fig. 59] and Roederer [4, Fig. 3.13].

Another set of equal-loudness curves was given in 1937 by Churcher and King [2]. They show some significant discrepancies over parts of the auditory diagram, probably because of the limited number of observations (Fletcher and Munson reported 297 observations using eleven observers) and because the observations did not discriminate for age. The importance of the age of the listener was brought out in a series of experiments by Robinson and Dadson [7] in the 1950s. Their results for two age groups are given in Fig. 2. They cover a frequency range from 25 to 15,000 Hz and a sound pressure level up to 130 dB (relative to $2.0 \cdot 10^{-5}$ newton/m^2).

The curves for the younger age group, which show two local minima around 4,000 and 12,500 Hz, are reproduced in the monograph of Rossing [5, Fig. 6.4]. The same curves are recommended for "otologically normal persons within the age limits from 18 to 30 years inclusive" over the frequency range from 20 to 12,500 Hz (up to the second minimum) by the International Organization for Standardization [6].

The (physical) loudness L_p still does not measure loudness in an absolute manner: doubling L_p does not lead to the perception that the tone is twice as loud. An absolute measure is given by the *(subjective) loudness* L_s, which is defined in terms of L_p by the formula [5, Section 6.5],

$$L_s = 2^{(L_p-40)/10}. \qquad \boxed{\text{p2s}} \tag{2.2}$$

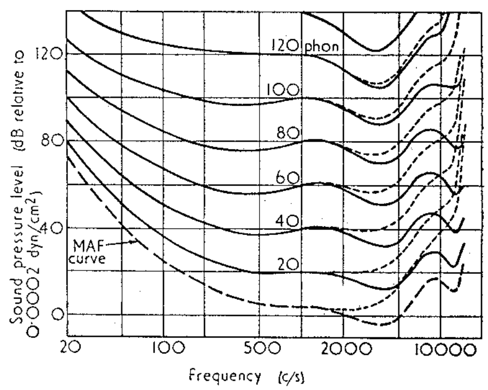

Figure 2: Equal-loudness curves; solid curves: age 20 years; dashed curves: age 60 years [7, Fig. 8].

This definition assumes that the perceived loudness doubles whenever the physical loudness increases by 10 dB, as found experimentally by Stevens [8]. The unit of L_s is the *sone* (plural *sones*). Loudness scaling in DISCO is done on the basis of sones.

There have been several attempts to capture the functional dependence of the loudness on frequency and intensity in an analytical expression. An approximate expression for L_s in terms of the pressure variation, which bypasses the intensity and the physical loudness altogether, was given by Stevens [9],

$$L_s = C(f)(\Delta p/\Delta p_0)^{2/3}. \qquad \boxed{\text{p2l}} \tag{2.3}$$

Here, C is a parameter which depends on the frequency. A similar formula with the power 0.60 instead of 2/3 is suggested by Rossing [5, Section 6.5].

In the ISO publication [6], the following expression is given for L_p as a function of the frequency f and the intensity I (our notation),

$$L_p = 4.2 + \frac{a(f)(I - T(f))}{1 + b(f)(I - T(f))}. \qquad \boxed{\text{i2l-ISO}} \tag{2.4}$$

Here, T is the threshold value of the intensity at the given frequency, and a and b are parameters which depend on the frequency f; values for selected values of f are given in tabular form.

We propose a slightly different rational approximation,

$$L_p = L_0 \frac{1 + L_1(f)(I - I_0(f))}{1 + L_2(f)(I - I_0(f))}, \qquad \boxed{\text{i2l}} \tag{2.5}$$

where L_0 is a constant scaling factor, $L_0 = 21.8033$, and L_1, L_2, and I_0 are functions of the frequency f. The graphs of the functions L_1, L_2, and I_0 are given in Fig. 3; selected values are given in Table 1.

Remark. The constant L_0, as well as the values of the functions L_1, L_2, and I_0 marked by the symbol $\times$ in Fig. 3 were obtained from a least-squares fit of the isofrequency data (L_p as a function of I at fixed values of the frequency f) in [6, Table 3]. The graphs of L_1, L_2, and I_0, as well as their values given in Table 1 were obtained from cubic-spline interpolations.

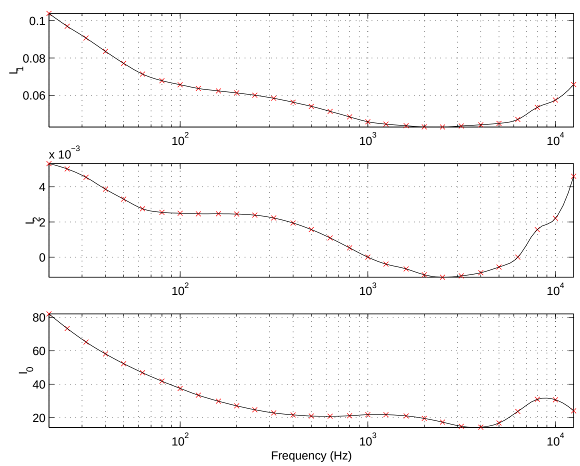

Figure 3: Graphs of L_1, L_2, and I_0 (spline interpolation).

The expression (2.5), combined with linear interpolation in Table 1, gives a very good approximation (within the experimental error) of the equal-loudness curves for all frequencies in the range from 20 to 12,500 Hz; see Fig. 4.

2.1.2 Implementation

We now turn to the actual implementation of the algorithms in the loudness routines in DISCO. As stated in the Introduction (Section 1), the loudness of a sound is one of its attributes to be specified by the user. Loudness is always understood in the sense of *perceived loudness* and is specified in sones.

In the present section, we are concerned only with the simplest sounds, namely sounds that consist of a single partial (pure tones).

While the user may think that the loudness of a pure tone is directly related to its size (the amplitude of its envelope), we have seen in the preceding

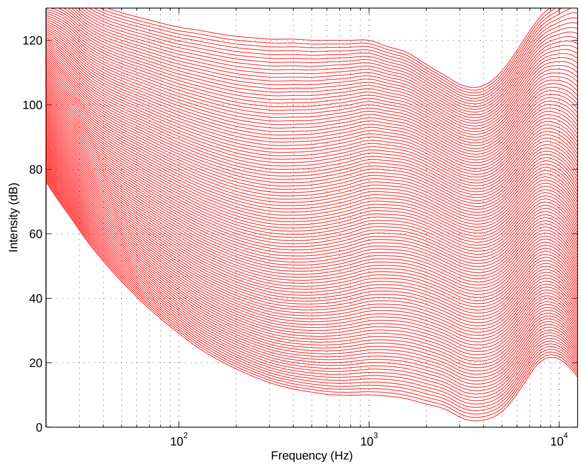

Figure 4: Equal-loudness curves, from Eq. (2.5).

section that the is actually a complicated nonlinear function of the amplitude. The functional dependence is expressed graphically through the equal-loudness curves or formally through, for example, Eq. (2.5). Hence, to achieve the target loudness specified by the user, DISCO must solve the *inverse* problem of finding the intensity that yields the target loudness at the given frequency.

The first step consists of the conversion of the specified loudness from sones to phons. The inverse of the formula given in Eq. (2.2) is

$$L_p = 40 + 10 \log_2 L_s. \qquad \boxed{\text{s2p}} \tag{2.6}$$

To find the intensity that yields the target loudness (in phons), we have developed a formula that is inverse to the one given in Eq. (2.5),

$$I = I_0 \frac{1 + I_1(f)(L - L_0(f))}{1 + I_2(f)(L - L_0(f))}, \qquad \boxed{\text{l2i}} \tag{2.7}$$

where I_0 is a constant scaling factor, $I_0 = 55.3215$, and I_1, I_2, and L_0 are functions of the frequency f. With the frequency f specified in Hz and the loudness L_p in phons, the formula gives the intensity I in dB. The graphs of the functions I_1, I_2, and L_0 are given in Fig. 5; selected values are given in Table 2.

Remark. The constant I_0, as well as the values of I_1, I_2, and L_0 marked by the symbol $\times$ in Fig. 5 were obtained from a least-squares fit of the isofrequency data (I as a function of L_p at fixed values of the frequency f) in [6, Table 2]. The graphs of I_1, I_2, and L_0, as well as their values given in Table 2 were obtained from cubic-spline interpolations.

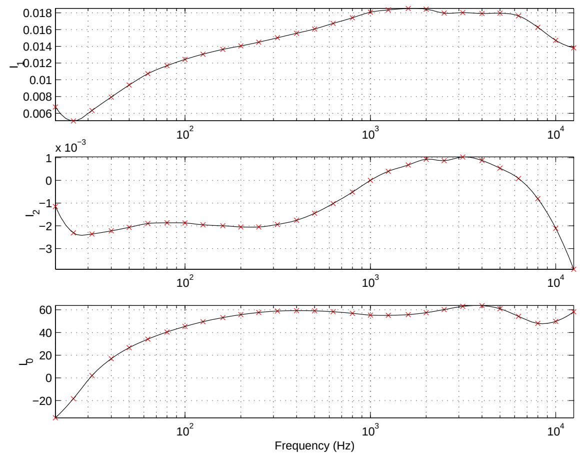

Figure 5: Graphs of I_1, I_2, and L_0 (spline interpolation).

The expression (2.7), combined with linear interpolation in Table 2, gives a very good approximation of the equal-intensity curves for all frequencies in the range from 20 to 12,500 Hz; see Fig. 6.

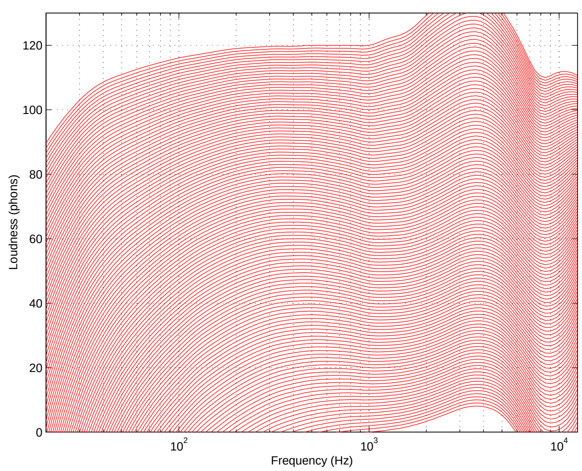

Figure 6: Equal-intensity curves, from Eq. (2.7).

The SPL follows from the intensity I,

$$\Delta p / \Delta p_0 = 10^{I/20}. \qquad \boxed{\text{i2p}} \qquad (2.8)$$

The amplitude a of the pure tone, which is a measure of the SPL, is stored as a real number in the interval $(0, 1)$. The scaling is set by fixing the maximum allowable intensity, I_m, so

$$a = (\Delta p/\Delta p_0) \cdot 10^{-I_m/20} = 10^{-(I_m-I)/20}. \qquad \boxed{\text{p2a}} \qquad (2.9)$$

2.2 Critical Bands

When two or more tones are superimposed, the way in which their individual loudnesses combine depends on how close they are to each other in frequency. In this section we discuss the case where the frequencies are the same or fall within a *critical band.* The width of a critical band varies with frequency but is independent of intensity [11]. We use the expression

$$\Delta_{\text{CB}} = 25 + 75(1 + 1.4(f/1,000)^2)^{0.69} \qquad \boxed{\text{CB}} \qquad (2.10)$$

for the bandwidth of a critical band centered at f.

The loudness of a sound composed of partials within a single critical band is computed from the total intensity I,

$$I = \sum_i I_i. \qquad \boxed{\text{Iband}} \qquad (2.11)$$

Here, I_i is the intensity of the ith partial within the critical band. The total intensity I is associated with the center frequency of the critical band. The loudness of the critical band in phons is found from the equal-loudness curves or from Eq. (2.5), and its loudness in sones from Eq. (2.2).

In DISCO, the inverse problem must be solved. Given the (perceived) loudness of a sound whose constituent partials belong to a single critical band, together with the relative size of each partial's envelope, find the amplitudes of the constituent partials.

Given the sound's target loudness L_s and the center frequency of the critical band, f, we can find the total intensity I of the sound by the method described for pure tones in Section 2.1, using the equal-intensity curves of Fig. 5 or the formula given in Eq. (2.7). The problem is how to distribute this total intensity among the partials in the sound.

Suppose the sound consists of n partials. Let f_i be the frequency of the ith partial, and let its size a_i (that is, the amplitude of its envelope) be specified as a fraction γ_i of an adjustable parameter a,

$$a_i = \gamma_i a, \quad i = 1, \ldots, n. \qquad \boxed{\text{ai}} \qquad (2.12)$$

An expression for the intensity I_i of the ith partial in terms of γ_i can be derived from Eqs. (2.1), (2.9), and (2.12),

$$I_i = I_m + 20\ \log_{10}\gamma_i + 20\ \log_{10} a, \quad i = 1, \ldots, n. \qquad \boxed{\text{Ii}} \quad (2.13)$$

According to Eq. (2.11), the sum of these intensities must equal the total intensity I of the sound,

$$I = nI_m + 20\ \log_{10}(\gamma_1 \ldots \gamma_n) + 20n\ \log_{10} a. \quad (2.14)$$

Solving this equation for a, we find

$$a = (\gamma_1 \ldots \gamma_n)^{1/n} \cdot 10^{-(I_m - I/n)/20}. \quad (2.15)$$

Having found a, we compute the unknowns a_i for $i = 1, \ldots, n$ from Eq. (2.12). The sound is thus completely determined.

In a sense, it is more natural to specify the relative loudness of each partial rather than its relative size. If the loudness of the first partial, $L_{s,1}$, is taken as an adjustable parameter, and the loudness of each partial is specified as a fraction of it,

$$L_{s,i} = \gamma_i L_{s,1}, \quad i = 1, \ldots, n, \qquad \boxed{\text{Lsi}} \quad (2.16)$$

the procedure to find the amplitudes a_i is not as straightforward, because of the nonlinear dependence of the intensity on the loudness. But Eq. (2.3) provides a useful shortcut if the factor C is constant across the critical band.

First, we compute the total intensity I of the sound band from the target loudness L_s in the same way as before. Using Eq. (2.3), we obtain the value of the constant C,

$$C = L_s \cdot 10^{-I/30}. \qquad \boxed{\text{C}} \quad (2.17)$$

An expression for the intensity I_i of the ith partial in terms of γ_i can be derived from Eqs. (2.1), (2.3) and (2.17),

$$I_i = 30\ \log_{10}\gamma_i + 30\ \log_{10}(L_{s,1}/C) = I + 30\ \log_{10}\gamma_i + 30\ \log_{10}(L_{s,1}/L_s). \quad (2.18)$$

The sum of these quantities must equal the total intensity I, so

$$(n-1)I + 30\ \log_{10}(\gamma_1 \ldots \gamma_n) + 30n\ \log_{10}(L_{s,1}/L_s) = 0. \quad (2.19)$$

Solving this equation for $L_{s,1}$, we find

$$L_{s,1} = (\gamma_1 \ldots \gamma_n)^{1/n} \cdot 10^{-(1-1/n)I/30}\, L_s. \quad (2.20)$$

Having found $L_{s,1}$, we compute the unknowns $L_{s,i}$ for $i = 1, \ldots, n$ from Eq. (2.16). From the loudness $L_{s,i}$ and the frequency f_i we compute the size a_i in the usual manner. The sound is thus completely determined.

2.3 Complex Sounds

If the range of frequencies in a sound exceeds one critical bandwidth, the resulting loudness is less than the sum of the loudnesses of the individual critical bands. Particularly, if the frequency differences are very large, a listener tends to focus primarily on one component, for example the loudest one or the one with the highest frequency, and assign a total loudness nearly equal to the loudness of that component [4, Section 3.4].

An approximate formula for the loudness of a sound was given by Rossing [5, Section 6.6]. Let $L_{s,i}$ denote the loudness (in sones) of the ith critical band, and let m be the index of the loudest critical band,

$$L_{s,m} = \max_i L_{s,i}. \tag{2.21}$$

Then the loudness of the composite sound is approximately given by the expression

$$L_s = L_{s,m} + 0.3 \sum_{i \neq m} L_{s,i}, \qquad \boxed{\text{l-complex}} \tag{2.22}$$

where the sum extends over all critical bands except the loudest.

The implementation of this algorithm in DISCO requires again the solution of the inverse problem. Given the composition of a sound with partials whose frequencies are spread over more than one critical band, find the amplitudes of the individual partials that collectively give the sound its specified (perceived) loudness.

The first problem is how to divide the frequency range into critical bands. The solution to this problem is to some degree arbitrary. In the current version of DISCO, the lowest frequency f_1 in a sound is taken as the center frequency of the first critical band, so the first critical band covers the range $(f_1 - \frac{1}{2}\Delta_{\text{CB}}(f_1), f_1 + \frac{1}{2}\Delta_{\text{CB}}(f_1))$, where Δ_{CB} is defined by Eq. (2.10) with $f = f_1$. The first frequency beyond the first critical band is associated with the center frequency of the second critical band, and so on. Notice that, with this scheme, the lower half of each critical band is empty. Also, this scheme allows for overlapping critical bands, and it is possible that a partial falls within more than one critical band. The implementation accounts for each frequency only once, incorporating it in the critical band that is encountered first as one goes up the frequency scale.

Alternatively, one could subdivide the entire frequency range once and for all into nonoverlapping critical bands, using Eq. (2.10). Better methods would take the actual distribution of frequencies into account. One possible method would be to rank the partials in the order of the distance to their nearest neighbor and center the first critical band mid-way between the first and second partial, and repeat this process with the remaining partials, until the entire frequency range has been covered.

Once the frequency range has been divided into critical bands, the next problem is how to distribute the loudness, which is specified for the composite sound, among the various critical bands.

To find the proper scaling, DISCO uses a *predictor-corrector* strategy. In a first pass, the program estimates the relative contribution to the loudness from each critical band; in a second pass, it corrects these estimates to obtain the target value of the loudness.

To estimate the relative contribution to the loudness from each critical band, the additive synthesis process is executed with the amplitude of one of the partials in the sound as the unit of pressure variation amplitude. The amplitudes of all the partials in the sound are specified as fractions of this adjustable parameter. The intensity of each critical band is computed in accordance with Eq. (2.11), and the equal-loudness curves or Eq. (2.5) are used to find the corresponding values of the physical loudness of each critical band. These values are then converted to sones by means of Eq. (2.2). The result is a vector of loudness values, one for each critical band,

$$(L^{\mathrm{c}}_{s,1}, \dots, L^{\mathrm{c}}_{s,n}). \qquad \boxed{\text{Lc-vec}} \tag{2.23}$$

From this vector one finds an estimate L^{c}_s of the loudness of the sound (in sones) by applying Eq. (2.22).

The computed loudness L^{c}_s will, in general, be different from the target loudness L^{t}_s. The ratio defines a *scaling factor*,

$$p = L^{\mathrm{t}}_s / L^{\mathrm{c}}_s. \qquad \boxed{\text{p}} \tag{2.24}$$

In the second pass, the process is reversed. First, the scaling factor is applied to each component of the vector (2.25),

$$(L^{\mathrm{t}}_{s,1}, \dots, L^{\mathrm{t}}_{s,n}) = (pL^{\mathrm{c}}_{s,1}, \dots, pL^{\mathrm{c}}_{s,n}). \qquad \boxed{\text{Lt-vec}} \tag{2.25}$$

The vector of target loudness values (in sones) is converted to a vector of target values in phons by means of Eq. (2.6), and the intensity of each critical band is found from the equal-loudness curves (in reverse mode) or Eq. (2.7). Within each critical band, the intensity is then equally distributed among the constituent partials, and the corrected value of the pressure variation amplitude computed in the usual manner.

3 Anticlip

When several sounds coexist simultaneously, their waveforms are simply added to obtain the complete audiowave of the composition. As a result, the magnitude of the audio signal in the sound file may increase. For a digital instrument that accepts sound files of a predetermined format the increase may result in overflow. Overflow gives rise to *clipping* (a popping noise) when the sound file is played—clearly, an undesirable result. The anticlip routine in DISCO checks the score file for potential overflow and rescales the sounds as necessary while preserving the ratio of perceived loudness levels. Thus, it is possible to produce an entire sound file in a single run from the score file, even when the sounds cover a wide dynamic range.

To appreciate the difficulty inherent in scaling, consider a sound cluster consisting of numerous complex sounds, all very loud and resulting in clipping, followed by a barely audible sound with only two or three partials. If the cluster's amplitude is decreased to fit the instrument's format, and that of the tiny soft sound following it is scaled in the same proportion, the latter disappears under system noise. On the other hand, if only the loud cluster is scaled, the relationship between the two sound events is completely distorted. The anticlip routine in DISCO deals with this problem by adjusting both loud and soft sounds so that their perceived loudness matches the desired relationship and no clipping occurs.

The score is partitioned into segments, with breakpoints marking either the entry or the disappearance of a sound. Thus, the composition is constant on each segment, in the sense that no sounds appear or disappear during the time interval covered by the segment. After converting the score file into a sound file, DISCO checks whether and, if so, where overflow occurs relative to the specified format. Segments where overflow occurs are marked, as are neighboring segments that share sounds with the marked segments, neighbors of these neighbors, and so on. Thus, stretches of the score are identified wihin which all sounds are scaled. The scaling is done on the basis of perceived loudness.

Several sounds superimposed within a segment do not differ fundamentally from a single sound composed of the constituent partials of all the sounds in the segment, so the loudness techniques described in Section 2 can be applied to the segment marked by overflow. After computing the perceived loudness of each critical band, the program finds the loudness of the segment by means of Eq. (2.22). The latter is then reduced by reducing the loudness value for each critical band by the same factor, which is estimated from the observed amount of overflow.

Given the (reduced) loudness value of each critical band, the program finds the pressure variation amplitude in the usual manner, as described in the last paragraph of Section 2.3. A linearly decaying reduction scale is developed, which vanishes at the endpoints of the stretch of the score identified for rescaling, and applied to the neighboring segments. If the resulting score file still results in overflow, the reduction procedure is repeated with a larger reduction, until overflow no longer occurs.

ACKNOWLEDGMENTS

The work of H.G.K. was supported by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, U.S. Department of Energy, under Contract W-31-109-Eng-38.

References

[1] H. Fletcher and W. A. Munson, "Loudness, Its Definition, Measurement and Calculation," *J. Acoustical Soc. America* **5** (1933) 82–108

[2] B. G. Churcher and A. J. King, *J. Instn Elect. Engrs* **81** (1937) 57

[3] Sir James Jeans, *Science & Music*, Cambridge Univ. Press (1937); Dover Publ., New York (1968)

[4] J. G. Roederer, *The Physics and Psychophysics of Music*, Third Ed., Springer-Verlag, New York (1995)

[5] T. D. Rossing, *The Science of Sound*, Addison-Wesley Publ. Co. (1982)

[6] International Standards Organization, "Acoustics — Normal equal-loudness level contours," Standard ISO 226: 1987(E)

[7] D. W. Robinson and R. S. Dadson, "A re-determination of the equal-loudness relations for pure tones," British J. of Appl. Phys. **7** (1956) 166–181

[8] S. S. Stevens, "Measurement of loudness," *J. Acoustical Soc. America* **27** (1955) 815–

[9] S. S. Stevens, "Calculation of the loudness of complex noise," *J. Acoustical Soc. America* **28** (1956) 5–

[10] H. G. Kaper, D. Ralley, and S. Tipei, "Perceived Equal Loudness of Complex Tones: A Software Implementation," Proc. 4th Int'l Confg. on Music Perception and Cognition, Montreal, Canada, August 1996, pp. 127–131

[11] E. Zwicker, G. Flottorp, and S. S. Stevens, "Critical bandwidth in loudness summation," *J. Acoustical Soc. America* **29** (1957) 548–557

Authors' e-mail addresses:

`kaper@mcs.anl.gov`
`tipei@mcs.anl.gov`

Table 1: Function values for Eq. (2.5).

f	L_1	L_2	I_0
20	0.1039	0.0053	82.1131
21	0.1018	0.0052	79.4898
23	0.0998	0.0052	76.9334
24	0.0979	0.0051	74.4436
26	0.0960	0.0049	72.0207
28	0.0942	0.0048	69.6645
30	0.0925	0.0047	67.3903
32	0.0907	0.0045	65.2242
34	0.0888	0.0044	63.1668
36	0.0868	0.0042	61.2169
38	0.0849	0.0040	59.3425
41	0.0829	0.0038	57.5283
44	0.0810	0.0036	55.7745
47	0.0791	0.0035	54.0797
50	0.0773	0.0033	52.4391
53	0.0755	0.0031	50.8523
57	0.0738	0.0030	49.3192
60	0.0723	0.0028	47.8290
64	0.0710	0.0027	46.3745
69	0.0698	0.0026	44.9555
73	0.0689	0.0026	43.5727
78	0.0681	0.0026	42.2291
84	0.0674	0.0025	40.9252
89	0.0667	0.0025	39.6610
95	0.0662	0.0025	38.4206
102	0.0656	0.0025	37.1885
108	0.0649	0.0025	35.9645
116	0.0643	0.0025	34.7575
124	0.0638	0.0025	33.6248
132	0.0634	0.0025	32.5767
141	0.0630	0.0025	31.6133
150	0.0627	0.0025	30.7099
160	0.0624	0.0025	29.8329
171	0.0621	0.0025	28.9824
182	0.0618	0.0025	28.1592
195	0.0615	0.0025	27.3744
208	0.0611	0.0024	26.6310
222	0.0608	0.0024	25.9290
237	0.0604	0.0024	25.2669
253	0.0600	0.0024	24.6422
270	0.0596	0.0023	24.0549
288	0.0591	0.0023	23.5065
307	0.0586	0.0023	23.0171
328	0.0581	0.0022	22.5927
350	0.0576	0.0021	22.2333
373	0.0570	0.0020	21.9284
398	0.0564	0.0019	21.6551
425	0.0557	0.0018	21.4125
454	0.0551	0.0017	21.2012
484	0.0545	0.0016	21.0346

f	L_1	L_2	I_0
517	0.0538	0.0015	20.9184
551	0.0530	0.0014	20.8528
588	0.0523	0.0012	20.8321
628	0.0515	0.0011	20.8398
670	0.0507	0.0010	20.8745
715	0.0499	0.0008	20.9366
763	0.0491	0.0006	21.0397
814	0.0482	0.0005	21.1923
869	0.0474	0.0003	21.3944
927	0.0466	0.0002	21.6235
990	0.0460	0.0000	21.7849
1056	0.0454	-0.0001	21.8677
1127	0.0451	-0.0002	21.8721
1203	0.0448	-0.0003	21.8188
1284	0.0445	-0.0004	21.7237
1370	0.0442	-0.0005	21.5868
1462	0.0440	-0.0006	21.4045
1560	0.0438	-0.0006	21.1555
1665	0.0436	-0.0007	20.8366
1777	0.0434	-0.0008	20.4477
1896	0.0432	-0.0009	19.9895
2024	0.0431	-0.0010	19.4629
2160	0.0430	-0.0011	18.8678
2305	0.0430	-0.0011	18.2074
2460	0.0431	-0.0011	17.5062
2625	0.0431	-0.0011	16.7691
2801	0.0433	-0.0011	15.9960
2990	0.0435	-0.0011	15.2781
3190	0.0437	-0.0011	14.7330
3405	0.0438	-0.0010	14.3613
3634	0.0440	-0.0010	14.1670
3878	0.0442	-0.0009	14.1918
4138	0.0443	-0.0008	14.4470
4416	0.0445	-0.0008	14.9324
4713	0.0447	-0.0007	15.7117
5030	0.0450	-0.0006	16.9059
5368	0.0453	-0.0004	18.5181
5728	0.0458	-0.0003	20.5342
6113	0.0466	-0.0001	22.7001
6524	0.0479	0.0002	24.9207
6962	0.0496	0.0006	27.1963
7430	0.0517	0.0011	29.3403
7929	0.0534	0.0015	30.8238
8462	0.0547	0.0018	31.6143
9030	0.0556	0.0019	31.7126
9637	0.0567	0.0020	31.2204
10,285	0.0582	0.0024	30.2092
10,976	0.0603	0.0029	28.6789
11,713	0.0628	0.0037	26.6297
12,500	0.0658	0.0046	24.0614

Table 2: Function values for Eq. (2.7).

f	I_1	I_2	L_0
20	0.0067	-0.0011	-35.1670
21	0.0059	-0.0016	-30.7158
23	0.0054	-0.0020	-25.8910
24	0.0051	-0.0022	-20.6924
26	0.0051	-0.0024	-15.1202
28	0.0054	-0.0024	-9.1743
30	0.0059	-0.0024	-3.2165
32	0.0063	-0.0024	2.1256
34	0.0068	-0.0023	6.8409
36	0.0072	-0.0023	10.9365
38	0.0077	-0.0022	14.6233
41	0.0081	-0.0022	17.9998
44	0.0085	-0.0022	21.0660
47	0.0089	-0.0021	23.8432
50	0.0093	-0.0021	26.4057
53	0.0098	-0.0020	28.7602
57	0.0101	-0.0020	30.9070
60	0.0105	-0.0019	32.9099
64	0.0108	-0.0019	34.8128
69	0.0111	-0.0019	36.6157
73	0.0114	-0.0019	38.3208
78	0.0116	-0.0019	39.9395
84	0.0118	-0.0019	41.4734
89	0.0121	-0.0019	42.9226
95	0.0123	-0.0019	44.3080
102	0.0125	-0.0019	45.6506
108	0.0127	-0.0019	46.9503
116	0.0129	-0.0019	48.2008
124	0.0130	-0.0020	49.3596
132	0.0132	-0.0020	50.4193
141	0.0134	-0.0020	51.3798
150	0.0135	-0.0020	52.2686
160	0.0136	-0.0020	53.1228
171	0.0138	-0.0020	53.9427
182	0.0139	-0.0020	54.7262
195	0.0140	-0.0020	55.4511
208	0.0141	-0.0021	56.1112
222	0.0143	-0.0021	56.7066
237	0.0144	-0.0021	57.2414
253	0.0145	-0.0020	57.7228
270	0.0147	-0.0020	58.1509
288	0.0148	-0.0020	58.5242
307	0.0150	-0.0020	58.8240
328	0.0151	-0.0019	59.0448
350	0.0153	-0.0019	59.1866
373	0.0154	-0.0018	59.2628
398	0.0156	-0.0018	59.3025
425	0.0157	-0.0017	59.3067
454	0.0158	-0.0016	59.2749
484	0.0160	-0.0015	59.1916

f	I_1	I_2	L_0
517	0.0162	-0.0014	59.0503
551	0.0163	-0.0013	58.8509
588	0.0165	-0.0012	58.5991
628	0.0167	-0.0010	58.3117
670	0.0169	-0.0009	57.9899
715	0.0171	-0.0008	57.6336
763	0.0173	-0.0006	57.2314
814	0.0175	-0.0005	56.7764
869	0.0177	-0.0003	56.2684
927	0.0179	-0.0002	55.7450
990	0.0181	-0.0000	55.3640
1056	0.0182	0.0001	55.1435
1127	0.0183	0.0002	55.0829
1203	0.0183	0.0003	55.1090
1284	0.0184	0.0004	55.1657
1370	0.0184	0.0005	55.2529
1462	0.0185	0.0006	55.3815
1560	0.0185	0.0006	55.6135
1665	0.0185	0.0007	55.9587
1777	0.0185	0.0008	56.4172
1896	0.0185	0.0009	56.9755
2024	0.0184	0.0009	57.6180
2160	0.0183	0.0009	58.3445
2305	0.0181	0.0009	59.1487
2460	0.0180	0.0009	59.9834
2625	0.0179	0.0009	60.8392
2801	0.0180	0.0009	61.7160
2990	0.0180	0.0010	62.5216
3190	0.0180	0.0010	63.1370
3405	0.0180	0.0010	63.5616
3634	0.0180	0.0010	63.7900
3878	0.0179	0.0009	63.7658
4138	0.0179	0.0008	63.4742
4416	0.0179	0.0007	62.9151
4713	0.0180	0.0006	62.0522
5030	0.0180	0.0005	60.8165
5368	0.0179	0.0004	59.2063
5728	0.0179	0.0003	57.2365
6113	0.0177	0.0002	55.1800
6524	0.0175	-0.0000	53.1389
6962	0.0172	-0.0002	51.1129
7430	0.0168	-0.0005	49.2810
7929	0.0163	-0.0008	48.1513
8462	0.0159	-0.0011	47.7547
9030	0.0154	-0.0015	48.0905
9637	0.0149	-0.0019	49.0449
10,285	0.0146	-0.0023	50.5380
10,976	0.0142	-0.0028	52.5699
11,713	0.0140	-0.0033	55.1404
12,500	0.0138	-0.0039	58.2496