Equal tempered scales

W. Luis Mochán

October 24, 2021

1 Introduction

If you create a musical scale by fourths or fifths starting from a given initial note, and their subharmonics you never reach the original note exactly, though you come closer than all previous notes for some magical numbers: 2, 5, 7, 12... Thus, the usual pentatonic, diatonic and chromatic scales. The chromatic scale has to be tempered, its notes slightly shifted from that corresponding to fifths or fourths, in order to avoid some disagreable disonant combinations. The most common nowadays is the *equal temperament*, in which the n-th note gets the frequency $\nu_n = 2^{n/12}\,\nu_0$. Thus, for example, to the diatonic scale there correspond the frequencies

Note	n	frequency (Hz)
A_4	0	440.00
B_4	2	493.88
C_5	3	523.25
D_5	5	587.33
E_5	7	659.26
F_5	8	698.46
G_5	10	783.99

Notice that this is part of an equal tempered chromatic scale, but it is not an equal tempered 7 note scale. Thus, Elías Mochan suggested to make a synthesizer to find out how an authentic equal tempered 7 note scale, with notes given by the sequence $\nu_n = 2^{n/7}\nu_0$. The table of notes would then be

Note	n	frequency (Hz)
A_4	0	440.00
B_4	1	485.80
C_5	2	536.37
D_5	3	592.20
E_5	4	653.84
F_5	5	721.90
G_5	6	797.04

I use the synth effect of the play command to play a note of a given duration and frequency.

2 Code

I will read a melody from the command line, parse it using a primitive syntax and play it with synth.

I start with the required packages.

```
use v5.12;
use List::Util qw(any);
use Getopt::Lazy
  'Chromatic=s'=>'File for equal tempered chromatic',
  'Diatonic=s'=>'File for equal tempered diatonic',
  'Pentatonic=s'=>'File for equal tempered pentatonic',
  'notes=s'=>'Melody to play',
  'help|h|?'=>'Get help',
  -summary=>"Play a melody in white equal tempered notes, "
            ."chromatic or diatonic",
  -usage=>'%c %o'
  ;
```

Figure 1: Packages

Then we initialize variables.

```
#read options from command line
GetOptions;
show_help and exit if $help;
show_help("Require at least one of -Chromatic, -Diatonic or Pentatonic")
     and exit unless any {defined $_} ($Chromatic, $Diatonic, $Pentatonic);
#
```

```
my @chromatic=qw(a z b c z d z e f z g);
my %chromatic_frequencies;
$chromatic_frequencies{$chromatic[$_]}=2**($_/12)*440
      for 0..@chromatic-1;
$chromatic_frequencies{$chromatic[$_]}/=2 for 3..@chromatic-1;
$chromatic_frequencies{z}=undef;
my @diatonic=qw(a b c d e f g);
my %diatonic_frequencies;
$diatonic_frequencies{$diatonic[$_]}=2**($_/7)*440 for 0..@diatonic-1;
$diatonic_frequencies{$diatonic[$_]}/=2 for 2..@diatonic-1;
my @pentatonic=qw(a c d f g);
my %pentatonic_frequencies;
$pentatonic_frequencies{$pentatonic[$_]}=2**($_/5)*440 for 0..@pentatonic-1;
$pentatonic_frequencies{$pentatonic[$_]}/=2 for 1..@pentatonic-1;

my %durations;
map {$durations{$_}=1/$_} (1,2,4,8,16,32); # valid times;
my @dots;
map {$dots[$_]=2-1/2**$_} (0..3);

my (@chromatic_commands, @diatonic_commands, @pentatonic_commands);
```

Figure 2: Initialize

Now, the main routine assembles a command to play the melody by calling sox when finished processing notes.

```
while(my $note=next_note()){
    assemble_chromatic($note);
    assemble_diatonic($note);
    assemble_pentatonic($note);
}
system join " ", "sox ", @chromatic_commands, $Chromatic if $Chromatic;
system join " ", "sox ", @diatonic_commands, $Diatonic if $Diatonic;
system join " ", "sox ", @pentatonic_commands, $Pentatonic if $Pentatonic;
```

Figure 3: Notes

We define a routine to parse the notes. The syntax is a string with a sequence of notes. Each note may be followed by optional commas or apostrophes to increase or decrease the octave, an optional number to set the duration and optional points to increase the time by a half.

```
sub next_note {
    state $duration=4; # defaults
    state $octave=4;
    return if $notes=~m/^\s*$/; # no more notes.
    die "Wrong format: $notes"
        unless $notes=~s/^\s*([a-g])(,+|'*)([0-9]*)([.]*)//;
    my ($note, $c, $d, $dots)=($1, $2, $3, $4);
    my $commas=$c=~/,/?length $c:0;
    my $apostrophes=$c=~/'/?length $c:0;
    $octave-=$commas;
    $octave+=$apostrophes;
    $duration=$d unless $d eq "";
    die "Wrong duration: $duration" unless defined $durations{$duration};
    my $time=$durations{$duration}*$dots[length $dots];
    return [$note, $octave, $time];
}
```

Figure 4: Next

Finally, we assemble sox commands to play the meloy in the desired scale.

```
sub assemble_chromatic {
    my ($note, $octave, $time)=@{shift @_};
    my $time2=.2*$time;
    my $frequency=$chromatic_frequencies{$note}*2**($octave-4);
    push @chromatic_commands,
         "\"|sox -n -p synth triangle $frequency fade 0 $time $time2\"";
}

sub assemble_diatonic {
    my ($note, $octave, $time)=@{shift @_};
    my $time2=.2*$time;
    my $frequency=$diatonic_frequencies{$note}*2**($octave-4);
    push @diatonic_commands,
         "\"|sox -n -p synth triangle $frequency fade 0 $time $time2\"";
}

sub assemble_pentatonic {
    my ($note, $octave, $time)=@{shift @_};
    my $time2=.2*$time;
    return unless defined $pentatonic_frequencies{$note};
```

```
    my $frequency=$pentatonic_frequencies{$note}*2**($octave-4);
    push @pentatonic_commands,
         "\"|sox -n -p synth triangle $frequency fade 0 $time $time2\"";
}
```

Figure 5: Assemble

The complete program is the following:

```
<<Packages>>
<<Initialize>>
<<Notes>>
<<Next>>
<<Assemble>>
```

Figure 6: Program

We can run the program as shown below.

```
./temperamento.pl -Chromatic  chromatic-scale.wav -Diatonic diatonic-scale.wav \
                  -notes "cdefgabc'"
./temperamento.pl -Chromatic chromatic-doremi.wav -Diatonic diatonic-doremi.wav \
                  -notes "c4.d8 e4.c8 e4c e2 d4.e8 ffed f1 e4.f8 g4.e8 g4e g2
                          f4.g8 aagf a1 g4.c8 defg a1 a4.d8 efga b1 b4.e8
                          fgab c'2.c8b, a4f bg c'g,ed c1"
./temperamento.pl -Chromatic  chromatic-5scale.wav -Diatonic diatonic-5scale.wav \
                  -Pentatonic pentatonic-scale.wav \
                  -notes "cdfgac'"
./temperamento.pl -Chromatic  chromatic-amanecer.wav -Diatonic diatonic-amanecer.wav \
                  -Pentatonic pentatonic-amanecer.wav \
                  -notes "c'a,g fga c'a,g fg8aga c'4a,c' da,d' ca,g f1."
```

The resulting sound files can then be played with any player, such as the terminal based `play` command from the `sox` package.