

Português (Portugal) → Inglês ▾



Informatics for Musicology (IPM) 2024/25

Jupyter Notebooks

Teacher: [JN Oliveira](#)

Department of IT at U. Minho, in collaboration with



Class from 3-Dec : TP class on IPM libraries in Haskell

Summary : Musical analysis by *horizontalization* of chords: the function `horiz`. Case study: Prelude nr. 4 of opus 28 by [Fryderyk Chopin](#) (1810-1849). Completion of exploration of IPM (Haskell) libraries for 'Computer-Aided Musicology'.

Exercises to review the given material.

⚠️ **Important** : run without moving the next cells.

In []:

```
: opt no - lint
: m Data . Char
: m Date . List
: m Date . List . Split
: m Data . Ratio
```

Modules developed for the discipline:

In []:

```
: l .../src / Cp . hs
: l .../src / Reducer . hs
: l .../src / Ipm . hs
: l .../src / Abc . hs
abcPlay k m b = abcPlay_ k m . dividet b
```

Data ("case studies"):

In []:

```
: l .../src / CS . hs
```

'Horizontalization'

Prelude no. 4 (in E minor) from opus 28 by [Fryderyk Chopin](#) (1810-1849) is one of his most famous works, for its simplicity and harmonic richness. The tradition of so-called *laments* in music comes from, as Prof. explains so well. [Matthew King](#) (Guild School of Music and Drama):

" This video is an introduction to Chopin's Prelude in E minor: the quintessential romantic mourning, popular among pioneering virtuosos and amateur players alike. Composed in the late 1830s, Chopin discovered new, unexplored harmonic possibilities in this deceptively simple music, creating a wonderful awareness and poetic depiction of melancholy in just two 'sentences': each one consisting of a sighing melody of fixed notes in the right hand over subtly shifting chords in the left hand from the tonic to the dominant, the music gives rise to a rich and labyrinthine path of magical harmonizations .

In this case study, we will detail the vision expressed in the quote above, exploring yet another (the last) of the functions of the IPM (Haskell) libraries for 'Computer-Aided Musicology' - `horiz` , cf:

Designation	Meaning	Detailed description
<code>horiz</code>	horizontal expansion	<code>horiz m</code> decomposes a sequence of chords into the (horizontal) sequence of their respective implicit melodic parts, in the order in which the notes appear. It is somewhat <i>inverse</i> of <code>dvert</code> .
<code>dvert</code>	verticalization	<code>dvert d [a,b,...]</code> vertically groups the notes of several melodic lines <code>[a,b,...]</code> previously sampled by <code>d</code>

12.1 - Start listening to the work by (a) copying the ABC code it generated onto the *clipboard abcShow* , given below; (b) enter the [online ABC editor](#) , paste and tap.

```
%%scale 0.7
%%pagewidth 20cm
%%barnumbers 10
X:1
%-- Abc file generated by Haskell library Abc.hs (IPM 2024/25)
M:C|
L:1/1
K:G
%%staves {1 2}
V:1
V:2
%-- the parts now
[V:1] B,,/16B1/16|B3/4c1/4|B3/4c1/4|B3/4c1/4|B3/4_B1/4|A3/4=B1/4|A3/4B1/4|A3/4B3/16A1/16|
A3/4^G1/4|-|^G1/4A1/8B1/8 d1/8c1/8E1/8A1/8| F3/4A1/4|F3/4A1/4| =G1/8F1/8C1/8B,1/8
^D1/8F1/8(3=d1/8c1/8B1/8| B3/4c1/4|B3/4c1/4|B3/4c1/4|B3/16^A1/16A1/4f3/16e1/16|
e1/8^d1/8c'1/8d1/8 d1/8e1/8g1/8B1/8| =d1/8c1/8(3e1/8E1/8A1/8F3/8A1/8| F3/4A1/4|
F3/4F3/16E1/16| E3/4F1/4| E3/4F1/4| E1/2z1/2| E1/2^D1/2| E1/1|
[V:2] z1/4| [EB,G,]1/8[EB,G,]1/8[EB,G,]1/8[EB,G,]1/8
[EB,G,]1/8[EB,G,]1/8[EB,G,]1/8[EB,G,]1/8| [EA,F,]1/8[EA,F,]1/8[EA,F,]1/8[EA,F,]1/8
[_EA,F,]1/8[EA,F,]1/8[EA,F,]1/8[EA,F,]1/8| [_EA,=F,]1/8[EA,F,]1/8[EA,F,]1/8[EA,F,]1/8
[DA,=F,]1/8[DA,F,]1/8[D^G,F,]1/8[DG,F,]1/8| [D^G,E,]1/8[DG,E,]1/8[DG,E,]1/8[DG,E,]1/8
[D=G,E,]1/8[DG,E,]1/8[^CG,E,]1/8[CG,E,]1/8| [C=G,E,]1/8[CG,E,]1/8[CG,E,]1/8[CG,E,]1/8
[CF,E,]1/8[CF,E,]1/8[CF,E,]1/8[CF,E,]1/8| [CF,E,]1/8[CF,E,]1/8[CF,E,]1/8[CF,E,]1/8
[CF,^D,]1/8[CF,D,]1/8[CF,D,]1/8[CF,D,]1/8| [CF,=D,]1/8[CF,D,]1/8[CF,D,]1/8[CF,D,]1/8
[CF,=D,]1/8[CF,D,]1/8[CF,D,]1/8| [C=F,=D,]1/8[CF,D,]1/8[CF,D,]1/8[CF,D,]1/8
[B,=F,=D,]1/8[B,F,D,]1/8[B,F,D,]1/8| [B,E,C,]1/8[B,E,C,]1/8[A,E,C,]1/8[A,E,C,]1/8
[A,E,C,]1/8[A,E,C,]1/8[A,E,C,]1/8[A,E,C,]1/8|
[A,E,B,,]1/8[A,E,B,,]1/8[A,^D,B,,]1/8[A,D,B,,]1/8
[A,E,C,]1/8[A,E,C,]1/8[A,E,C,]1/8[A,E,C,]1/8|
[A,^D,B,,]1/8[A,D,B,,]1/8[A,D,B,,]1/8[A,D,B,,]1/8
[A,E,C,]1/8[A,E,C,]1/8[A,E,C,]1/8[A,E,C,]1/8| [A,^D,B,,]1/4z1/4 z1/2|
[EB,G,]1/8[EB,G,]1/8[EB,G,]1/8[EB,G,]1/8 [EB,G,]1/8[EB,G,]1/8[EB,G,]1/8[EB,G,]1/8|
[EA,F,]1/8[EA,F,]1/8[EA,F,]1/8[EA,F,]1/8 [_EA,=F,]1/8[EA,F,]1/8[EA,F,]1/8[EA,F,]1/8|
[_E_A,=F,]1/8[EA,F,]1/8[DA,F,]1/8[DA,F,]1/8 [D^G,=E,]1/8[DG,E,]1/8[DG,E,]1/8[DG,E,]1/8|
[D=G,=E,]1/8[DG,E,]1/8[^CG,E,]1/8[CG,E,]1/8
```

```
[^A,E,^C,]1/8[A,E,C,]1/8[=A,E,=C,]1/8[A,E,C,]1/8| [B,,B,,,]1/8[AFCA,]1/8[AFCA,]1/8
[F^DB,G,]1/8[EB,G,]1/8[EB,G,]1/8[EB,G,]1/8| [ECA,]1/8[ECA,]1/8 [A,,]1/8[CF,E,]1/8
[B,E,B,,]1/8[B,E,B,,]1/8[A,E,C,]1/8[A,E,C,]1/8|
[B,E,B,,]1/8[B,E,B,,]1/8[B,E,B,,]1/8[B,E,B,,]1/8
[A,E,C,]1/8[A,E,C,]1/8[A,E,C,]1/8[A,E,C,]1/8|
[B,E,B,,]1/8[B,E,B,,]1/8[B,E,B,,]1/8[B,E,B,,]1/8
[B,^D,B,,]1/8[B,D,B,,]1/8[A,D,B,,]1/8[A,D,B,,]1/8| [G,C,]1/8[G,C,]1/8[G,C,]1/8[G,C,]1/8
[_B,C,]1/8[B,C,]1/8[A,E,C,]1/8[A,E,C,]1/8| [A,E,B,,]1/8[A,E,B,,]1/8[^G,E,B,,]1/8[G,E,B,,]1/8
[=G,E,B,,]1/8[G,E,B,,]1/8[G,E,B,,]1/8[G,E,B,,]1/8| [=G,C,_B,,]1/2 z1/2| [B,F,E,=B,,=B,,]1/2
[B,F,^D,F,,=B,,]1/2| [B,G,E,E,,E,,]1/1|
%-- end of generated Abc, key is 1
%ExitSuccess
```

12.2 - Using auxiliary settings,

```
In [ ]:
```

```
b = ( 1 % 4 : bin )
q = ( 1 % 4 : quaternion )
chopin = abcPlay "G" "C|" q
```

view the melody line chopin1 (right hand of the piano):

```
In [ ]:
```

```
chopin chopin1
```

12.3 - Switching to the left hand (chopin2), select from x your first ten notes:

```
In [ ]:
```

```
x = undefined
---
x
```

12.4 - Compare now x with abchs x the cell below. What did you do abchs ?

```
In [ ]: abchs x  
---  
chopin it
```

12.5 - We now want to know how many notes await us in the chords of `chopin2`. Complete the following cell to obtain this result:

```
In [ ]: ( t , r ) = unzip chopin2  
---  
.....
```

12.6 - We now want to decompose the chords `chopin2` horizontally, into as many melodic lines as necessary. This is what you get by running the next cell:

```
In [ ]: lines = horiz chopin2
```

How many melodic lines were generated? Ask this question in the following cell:

```
In [ ]:
```

12.7 - We will now want to eliminate repeated successive notes, which are frequent here due to having been extracted from eighth note chords. For example (run):

```
In [ ]: ( chopin . nrep . head ) lines
```

12.8 - Compare what was obtained above with something we heard previously:

```
In [ ]: ba = [( "G,,," , 1 % 1 ),( "G," , 1 % 2 ),( "^F," , 1 % 1 ),( "=F," , 1 % 2 ),( "=E," , 1 % 1 ),  
          ( "_E," , 1 % 2 ),( "D," , 1 % 1 ),( "B,,,," , 1 % 2 ),( "C, " , 1 % 2 ),( "D," , 1 % 2 ),( " D,,,"  
          ( "G,,," , 1 ),( "G," , 1 % 2 ),( "^F," , 1 % 1 )]  
--  
abcPlayM "Bb" "6/4" ba
```

(Where?)

12.9 - In the next cells repeat what was done above, but now for the other voices:

```
In [ ]:
```

12.10 - What do you expect to happen in the next cell?

```
In [ ]:
```

```
( chopin.P ) ( chopin1 : map nrep lines )
```

12.11 - Finally, in the result of the following cell, make the following changes:

1. Replace `%%staves [1 2 3 4 5 6]` with `%%staves [1 {2 (3 4 5)} 6]` - what effect do you expect from this change?
2. Add

```
%%MIDI program 1 68  
%%MIDI program 2 20  
%%MIDI program 3 20  
%%MIDI program 4 20  
%%MIDI program 5 42
```

3. And what happens if you add the line

```
%%staffnonote 0
```

at the beginning of the generated ABC?

Listen to the effect of these changes - what kind of music do you think was achieved?

In []:

```
abcShow
```