Finally, the only decision left for creating a cell of type A is pitch selection. The algorithm used isn't optimized for speed, but it has the advantage of being relatively simple; also, given that the current generation of computers has a lot of processing power, an execution of this whole implementation takes just a few seconds. The algorithm always generates four random pitches, regardless if the chord has 3 or 4 notes. If the chord has 3 notes, it will simply ignore the highest pitch. The generation of these pitches follow a trial and error method, by which the computer first generates a random set of pitches and then test if they conform to the specified constrains. The rules are:

 - all four pitches must have unique pitch classes
 - the two bottom pitches and the two top pitches must be within a major sixth from each other
 - no minor seconds are allowed

Given that this composition must be played using two mallets per hand, the second rule actually defines a conservative maximal interval for each hand, ensuring that the chords can always be played using this type of technique. The third rule avoids difficult minor second intervals to be played with a single hand, as well as between the inner intervals which can cause problems notation-wise (for the sake of simplicity, this implementation uses only sharp accidentals and this rule helps to avoid notation problems such as simultaneous F4 and F#4).

Cell B is the "tremolo cell". It may consist either of a single note or of a grace note

```
! selecting a random duration
call RANDOM_DURATION(duration)

! selecting a random dynamic
call RANDOM_DYNAMIC(dynamic,dynamic_string,previous_dynamic)

! selecting between 3 or 4 note chords
call RANDOM_NUMBER(x)
if (x < 0.5) then ! 50% of the time the top note is removed, creating a 3-note chord
  four_note_chord = .FALSE.
  pitch_vector(4) = 0
endif

if (four_note_chord) then ! only if this is a 4-note chord
  call RANDOM_NUMBER(x)
  if (x < 0.2) then ! 20% of the time
    grace_note = .TRUE. ! bottom note is a t
    else if (x < 0.35) then ! 15% of the time
      arpeggioUp = .TRUE.
    else if (x < 0.5) then ! 15% of the time
      arpeggioDown = .TRUE.
  endif
endif

call TIME(duration,16) ! time signature

if (grace_note) then ! if grace note
  call GRACE("\slashedGrace")
  call NOTE(pitch_vector(1),"8",TRIM(dynamic_string))
  call TIE()
  call END_GRACE()
  dynamic_string = "" ! erasing dynamic since it appears below the grace anyway
```
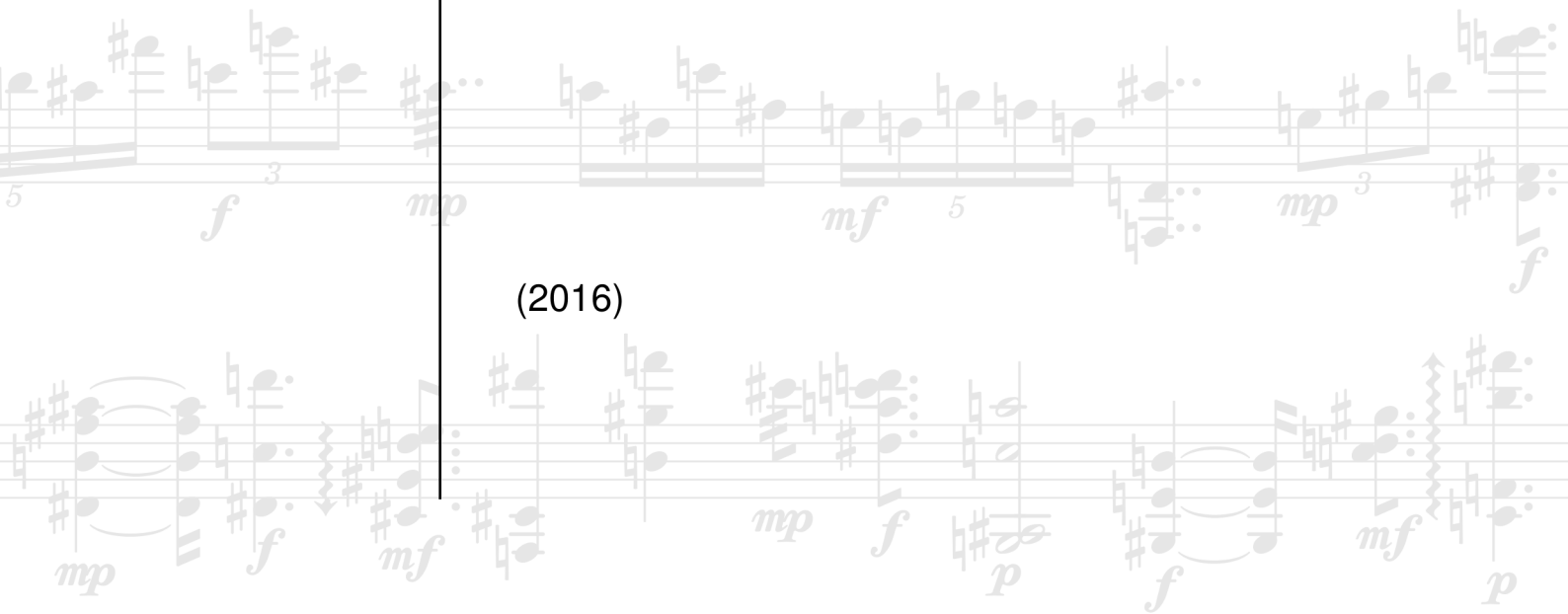
Rosetta

for solo marimba

GILBERTO AGOSTINHO

(2016)

General Information

Rosetta is an algorithmic work for solo marimba generated by a computer program. Because this program uses random procedures, a different version of this work is output at each execution. It is recommended that the performer generate a new version for each concert (more on how to generate a version below), though this is not a requirement. Also, the performer may play more than one single version on a same concert.

Durata: averagely 1'30"

Notation And General Performance Notes

- grace notes should be played as fast as possible and always on the beat

- tremolos are unmeasured and should also be played as fast as possible

- two types of arpeggios are used: upwards arpeggio and downwards arpeggio

- the range used in this piece is F3–C6

Generating a unique version

In order to generate a unique version of this work, it is necessary to have the following programs installed on a Linux computer:

- The program `rosetta.out`, which can be freely obtained by sending a request message to gilberto.agostinho.f@gmail.com

- LilyPond, which can be downloaded from www.lilypond.org

- LaTeX, which can be downloaded from www.latex-project.org

To generate a score, simply execute the file `rosetta.out` and wait until a score is automatically generated as a pdf file named `rosetta.pdf` (this will take just a few seconds). To generate a new score, simply run the executable file once again. A midi sample of this work is also generated, together with the score in LilyPond format files (`.ly`).

IMPORTANT NOTE: after generating a score, another execution of `rosetta.out` will result in the loss of all previously generated files (they will all be overwritten)! If you wish to keep a particular version of this composition, please copy all its files to a safe location.