Globbian Syntax

2021

Tamir Globus

Abstract

It may be easier to read letters of a language with unique symbols than a language such as musical notes which all look more or less the same. Languages such as English, Hebrew and French use different sets of symbols, which we refer to as 'letters' to instruct the reader how to pronounce the word or in which group the word exists. In other words, the reader does not need to read the whole word to understand it. Researchers have found that some of us can even complete the word just by seeing the first and last letters of it.

Since they all look similar, musical notes require a great amount of effort to read, and there is a good reason for that. If you've ever looked at a piano closely, you saw it has 88 keys. If we had 88 letters in the English alphabet, we would have had much harder time to recognize each letter in our mind, and much more to recognize a whole word - since the number of permutations of a vocabulary with that size is enormous. For the same reason, the language in which music players read notes couldn't be with 88 unique symbols. As a result, musicians have developed a cycle based technique called 'octave', which allows us to refer to each note more easily.

The following paper isn't a mathematical/statistical optimization paper where we attempt to optimize several parameters, and get close to the best language for the human mind. Rather than that, in this paper, we'll design an explicit new language that is based on mathematics, and is probably easier to understand than the current one.

“In order to define it, you must understand how it works.”

- **v** = octa**v**e
  The coefficient of v is the number of octaves of the note relative to the middle-C octave. Relative to the middle-C octave, higher pitches will have positive **v** and negative ones will have negative **v**.

- **t** = **t**ime
  If positive, the coefficient of **t** describes the number of counts one needs to play note. If negative, it tells you that no music should be played.

- **p** = ty**p**e
  C, D, E, F, G, A, B are all of the existing types the note can have. Any sharps and flats should be written as C#, E# or Cb, Ab or Bb. There is no sign for cancellation of the Sharpened or flattened note, and therefore, the sign does not change the pitch of all the notes of the same type for the same musical box.

- **Note Variable** - Any note or a group of notes (a note-matrix) can be assigned to a variable.

- **Musical Box Variable** - Any Musical Box is assigned to a number, such that the first is 1, the second is 2 and so on. If “3 - 10” was written, then the musical player should play all the musical boxes from the 3rd note, up to the 10th one, including the 3rd and the 10th.

- **Calculation First** - All musical boxes are played only after all calculations were made.

- **Operations** - All additions may be substituted. For operations such as sharpening a note, there may be counter operations such as flattening the note. For operations such as transforming a note to another note, may be counter operation too.

- **Representation** - A note is usually written as a three dimensional vector that its top component is the **type** of that note, the middle is the **octave** of the note and the lower is the **time** of the note, as follows:

$$\begin{bmatrix} F \\ 2 \\ 1/2 \end{bmatrix}$$

Time Matrix - If several notes share the same value such as type, time or octave, then can be written in a matrix. The **first** (most left - first column) note is played **first**, the **second** (second column) is played **secondly** and so on. The following matrix column-items share a common number of time counts, which is +2. therefore, it could be written as follows (pay attention that the last column note has +1 more time counts than the others, which means it will be played for +1 count longer):

$$\begin{bmatrix} F & G & G \\ 2 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix} = \begin{bmatrix} F+0 & G+0 & G+0 \\ 2+0 & 1+0 & 1+0 \\ 0+2 & 0+2 & 1+2 \end{bmatrix} = \begin{bmatrix} F & G & G \\ 2 & 1 & 1 \\ 2 & 2 & 3 \end{bmatrix}$$

Chord Matrix - If the **c** sign is on the bottom right of the matrix, then the user will have to play the notes simultaneously. However, each note in the chord can have a different time and octave, which is one of the reasons that this syntax is so powerful. Pay attention that some notes are played longer, but all are starting at the same time. The right matrix calculation is the explicit form of the left side calculation:

$$\begin{bmatrix} F & G & G \\ 2 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}_c + 2t = \begin{bmatrix} F & G & G \\ 2 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}_c + \begin{bmatrix} 0 \\ 0 \\ 2 \end{bmatrix}$$

From the last example we understand that **t**, **v** and **p** are all 3 dimensional vectors:

$$v = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \; t = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

In order to define the third vector **p**, we must define the group of all the note types and then define the addition/substitution methods when acting on it. As you can see in the example below, we first define the known notes from basic music theory, and then define their group in the mathematical form that provides us with the freedom to add/sub from certain elements of the group and get others:

$$\begin{aligned} types &= \{C, D, E, F, G, A, B\} \\ &= \{C,\ C+1,\ C+2, C+3, C+4, C+5, C+6\} \end{aligned}$$

Such that...

$$p = \begin{bmatrix} C \\ 0 \\ 0 \end{bmatrix}$$

The first inverse of the following chord is written using the **-1** at the top right of the matrix, as in mathematics. As you can see, all of it's notes are played for the same period of time but possibly located on different octaves. Additionally, the first note of the chord was raised by one octave, as would have happened in regular notes syntax:

$$\begin{bmatrix} C & E & G \\ v_1 & v_2 & v_3 \\ t & t & t \end{bmatrix}_c^{-1} = \begin{bmatrix} C & E & G \\ v_1 + 1 & v_2 & v_3 \\ t & t & t \end{bmatrix}_c$$

The second inverse of the chord is defined as follows:

$$\begin{bmatrix} C & E & G \\ v_1 & v_2 & v_3 \\ t & t & t \end{bmatrix}_c^{-2} = \begin{bmatrix} C & E & G \\ v_1+1 & v_2+1 & v_3 \\ t & t & t \end{bmatrix}_c$$

The **k**th inverse of **any** chord is defined as follows:

$$\begin{bmatrix} p_1 & p_2 & p_3 \\ v_1 & v_2 & v_3 \\ t_1 & t_2 & t_3 \end{bmatrix}_c^{-k} = \begin{bmatrix} p_1 & p_2 & p_3 \\ v_1 + floor\left(1+\frac{k}{4}\right) & v_2 + floor\left(\frac{x+1}{3}\right) & v_3 + floor\left(\frac{k}{3}\right) \\ t_1 & t_2 & t_3 \end{bmatrix}_c$$

In the last definition of the group of **types** of notes, we didn't include the sharps and flats, which are known as the black keys on the piano. In order to do so, we'll define the **types** group one last time:

$$\begin{aligned} types &= \{C,\ C\#,\ D,\ D\#,\ E,\ F,\ F\#,\ G,\ G\#,\ A,\ A\#,\ B\} \\ &= \{C,\ C+0.5,\ C+1,\ C+1.5,\ C+2,\ C+2.5,\ C+3,\ C+3.5,\ C+4,\ C+4.5,\ C+5,\ C+5.5,\ C+6\} \end{aligned}$$

In the last definition of **types** we defined the sharps as an additional half tone to the last element, except where we add **E** half tone and get a white key. Since, all additions may be substituted, we can define the two vectors for the sharpening and flattening operations:

$$sharp = \begin{bmatrix} 0.5 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \# \\ 0 \\ 0 \end{bmatrix}$$

$$flat = -\begin{bmatrix} 0.5 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -0.5 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \flat \\ 0 \\ 0 \end{bmatrix}$$

In order to apply a transformation on certain notes or values of the chord specifically, we must add to it a matrix with the same dimensions. Since we don’t know what the notes of the chord would be transformed to, me must calculate the expression, and then play the chord:

$$\begin{bmatrix} A & C\# & E \\ 0 & 1 & 1 \\ 2 & 1 & 0 \end{bmatrix}_c + \begin{bmatrix} 0 & 0 & 0 \\ 2 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} A & C\# & E \\ 2 & 1 & 1 \\ 3 & 2 & 1 \end{bmatrix}_c$$