Tablatures for Stringed Instruments and Generating Functions

Davide Baccherini, Donatella Merlini, and Renzo Sprugnoli

Dipartimento di Sistemi e Informatica
viale Morgagni 65, 50134, Firenze, Italia,
[baccherini,merlini,sprugnoli]@dsi.unifi.it
baccherini@gmail.com

Abstract. We study some combinatorial properties related to the problem of tablature for stringed instruments. First, we describe the problem in a formal way and prove that it is equivalent to a finite state automaton. We define the concepts of *distance between two chords* and *tablature complexity* in order to study the problem of tablature in terms of music performance. By using the Schützenberger methodology we are then able to find the generating function counting the number of tablatures having a certain complexity and we can study the average complexity for the tablatures of a music score.

1 Introduction

The music perfomance process involves several representation levels [13], such as physical, perceptual, operational, symbolic, structural. Consequently, a performance environment should be concerned at least in:

1. getting a score in input (symbolic level);
2. analysing it, like a human performer would do (structural);
3. modelling the constraints posed by body-instrument interaction (operational);
4. manipulating sound parameters (physical).

In particular, in the present paper we focus our attention on a problem present in both structural and operational levels, and very relevant for string instrument, namely the problem of *tablature*. Some instruments, such as the piano, have only one way to produce a given pitch. To play a score of music on a piano, one needs only to read sequentially the notes from the page and depress the corresponding keys in order. Stringed instruments, however, require a great deal of experience and decision making on the part of the performer. A given note on the guitar may have as many as six different positions on the fretboard on which it can be produced. A fretboard position is described by two variables, the string and the fret. To play a piece of music, the performer must decide upon a sequence of fretboard positions that minimize the mechanical difficulty of the piece to at least the point where it is physically possible to be executed. This process is time-consuming and especially difficult for novice and intermediate players and, as a result, the task of reading music from a page, as a pianist would, is limited only to very advanced guitar players. To address this problem, a musical notation known as *tablature* was devised. A tablature describes to the performer exactly how a piece of music is to be played by graphically representing the six guitar strings and labeling them with the corresponding frets for each note, in order. *Fingering* is the process that, given a sequence of notes or chords (set of notes to be played simultaneously), yields to assigning to each note one position on the fretboard and one finger of the left hand. Fingering and tablature problem has been studied from many points of view (see, e.g., [12,14,18]).

In this paper, we study some combinatorial properties of tablature problem. This problem can be described by a finite state automaton (or, equivalently, by a regular grammar) to which we can apply the Schützenberger methodology (see [8,9,15,16] for the theory and [6,7,11] for some recent applications). We define the concepts of *distance between two chords* and *tablature complexity* to study the problem of tablature in terms of the music performance. It is then possible to compute the average complexity of a tablature. In particular, we prove the following basic results:

1. every tablature problem is equivalent to a finite state automaton (or to a regular grammar);
2. an algorithm exists that finds the finite state automaton corresponding to a tablature problem;
3. by using the Schützenberger methodology we find the generating function $\Xi(t) = \sum_n \Xi_n t^n$ counting the number Ξ_n of tablatures with complexity n.

The concept of complexity introduced in this paper takes into consideration the total movement of the hand on the fretboard during the execution. This quantity is certainly related to the difficulty of playing a score of music but, of course, many other measures could be considered. Moreover, the difficulty depends on the artist who plays the song.

We will show an example taken from *Knocking on Heaven's Door* by Bob Dylan.

2 Stringed Instruments, Tablature and Symbolic Method

A *note* is a sign used in music to represent the relative duration and pitch of sound. A note with doubled frequency has another but very similar sound, and is commonly given the same name, called *pitch class*. The span of notes within this doubling is called an *octave*. The complete name of a note consists of its pitch class and the octave it lies in. The pitch class uses the first seven letters of the latin alphabet: A, B, C, D, E, F, and G (in order of rising pitch). The letter names repeat, so that the note above G is A (an octave higher than the first A) and the sequence continues indefinitely. Notes are used together as a musical scale or tone row. In Italian notation, the notes of scales are given in terms of Do - Re - Mi - Fa - Sol - La - Si rather than C - D - E - F - G - A - B. These names follow the original names reputedly given by Guido d'Arezzo, who had taken them from the first syllables of the first six musical phrases of a Gregorian Chant melody *Ut queant laxis*, which began on the appropriate scale degrees.

In this section we define in a formal way the problem of the tablature of a stringed instrument score. In order to do this, we take into consideration the set of the notes defined in the MIDI standard (see [4,5]), which uses the note-octave notation. In fact, this set contains a range of integer numbers $\daleth = \{0, ..., 127\}$, where every element represents a note of an octave.

Table 1. A representation of notes in the MIDI standard

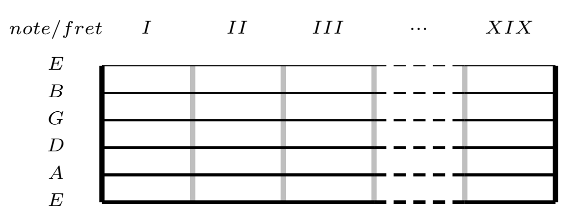
Octave	**Note numbers**											
	Do	**Do#**	**Re**	**Re#**	**Mi**	**Fa**	**Fa#**	**Sol**	**Sol#**	**La**	**La#**	**Si**
	C	**C#**	**D**	**D#**	**E**	**F**	**F#**	**G**	**G#**	**A**	**A#**	**B**
0	0	1	2	3	4	5	6	7	8	9	10	11
1	12	13	14	15	16	17	18	19	20	21	22	23
2	24	25	26	27	28	29	30	31	32	33	34	35
3	36	37	38	39	40	41	42	43	44	45	46	47
4	48	49	50	51	52	53	54	55	56	57	58	59
5	60	61	62	63	64	65	66	67	68	69	70	71
6	72	73	74	75	76	77	78	79	80	81	82	83
7	84	85	86	87	88	89	90	91	92	93	94	95
8	96	97	98	99	100	101	102	103	104	105	106	107
9	108	109	110	111	112	113	114	115	116	117	118	119
10	120	121	122	123	124	125	126	127				

A string instrument (or stringed instrument) is a musical instrument that produces sounds by means of vibrating strings.

Definition 1 (String Instrument). *We define a* string instrument *with m strings $\mathcal{SI}_m$ as a pair (S_m, n_f) where:*

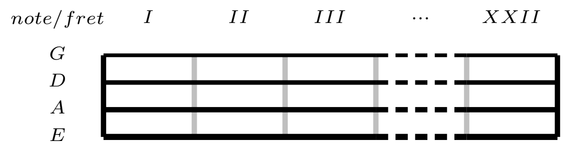
- $S_m = (note_1, \cdots, note_m) \in \daleth^m$ *represents the notes of the corresponding strings;*
- $n_f \in \mathbb{N}$ *represents the number of frets in this instrument.*

Example 1. For convention, the enumeration of the strings begins from the highest note to the most bass note. Therefore, we can define the *classical guitar [3]* as follows:



$$S_6 = (64, 59, 55, 50, 45, 40)$$
$$n_f = 19$$

instead for the *bass guitar [1]* we have:



$$S_4 = (31, 26, 21, 16)$$
$$n_f = 22$$

In music and music theory a *chord* is any collection of notes that appear simultaneously, or near-simultaneously over a period of time. A chord consists of three or more notes. Most often, in European influenced music, chords are tertian sonorities that can be constructed as stacks of thirds relative to some underlying scale. Two-note combinations are typically referred to as dyads or intervals. For the sake of simplicity, we use the following:

Definition 2 (Chord). *We say that ξ is a* chord*, if it belongs to $2^{\urcorner}$.*

Example 2. A famous chord is the *G Major* (or *Sol Major*). It is characterized by the following notes:

D, G, B (or *Re, Sol, Si* in Italian notation).

Using the previous definitions, $\xi = \{50, 55, 59\}$ corresponds to a *G Major* on the fourth octave. In the same way, *C Major* (formed by *C, E, G*) on the fourth octave is representable with $\xi = \{48, 52, 55\}$, whereas $\xi = \{60\}$ is a simple note *C* (or *Do*) on the fifth octave.

Definition 3 (Chord for a String Instrument). *Let $\mathcal{SI}_m = (S_m, n_f)$ be a string instrument with m strings and $\xi = \{a_1, \ldots, a_j\}$ a chord with $j \leq m$. ξ is a* chord *for $\mathcal{SI}_m$ iff there exists an injective function $f : \xi \to \{1, \ldots, m\}$ such that $\forall i = 1, \ldots, j$ we have $0 \leq a_i - note_{f(a_i)} \leq n_f$. We indicate with $\Gamma_{\mathcal{SI}}$ the set of these chords.*

A chord progression (also chord sequence, harmonic progression or sequence) is a series of chords played in order. In this paper, we give the following:

Definition 4 (Chord Progression). *Given a string instrument $\mathcal{SI}_m$, we call* chord progression *for the instrument $\mathcal{SI}_m$, every finite sequence $\xi_1, ..., \xi_n$ such that $\forall i = 1, \ldots, n$ we have $\xi_i \in \Gamma_{\mathcal{SI}}$.*

Example 3. If we use a classical guitar, a chord progression can be defined as follows:

C Major, A Minor, D Minor, G 7th
(or Do Major, La Minor, Re Minor, Sol 7th)

where:

Chord name	Notes of chord	Numeric representation
C Major	*C, G, C, E* (or *Do, Sol, Do, Mi*)	$\{48, 55, 60, 64\}$
A Minor	*A, A, C, E* (or *La, La, Do, Mi*)	$\{45, 57, 60, 64\}$
D Minor	*D, A, D, F* (or *Re, La, Re, Fa*)	$\{50, 57, 62, 65\}$
G 7th	*G, G, B, F* (or *Sol, Sol, Si, Fa*)	$\{43, 55, 59, 65\}$

While standard musical notation represents the rhythm and duration of each note and its pitch relative to the scale based on a twelve tone division of the octave, tablature (or tabulatura) is instead operationally based, indicating where and when a finger should be depressed to generate a note, so pitch is denoted implicitly rather than explicitly. Tablature for plucked strings is based upon a diagrammatic representation of the strings and frets of the instrument. In a formal way, we give the following:

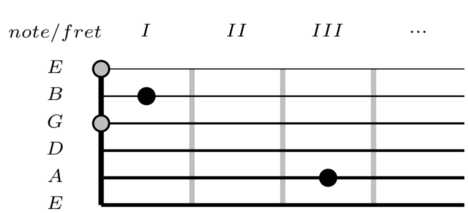
Definition 5 (Position). *Given an intrument* $\mathcal{SI}_m$ *with* m *strings, we call* position *the following function* TAB*:*

$$TAB : N_s \rightarrow N_f \cup \{\square\}$$

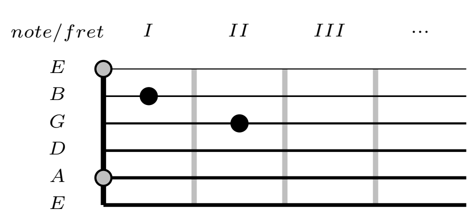
where

- $N_s = \{1, \ldots, m\}$ *is the set of strings in the instrument* $\mathcal{SI}_m$*;*
- $N_f = \{1, \ldots, n_f\}$ *is the set of frets in the instrument* $\mathcal{SI}_m$*;*
- $\square$ *is the null position.*

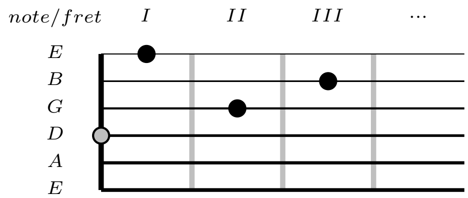
Example 4. We can also describe a position using a graphic representation. In this way, we represents only the notes that the musician must pluck. For example, given a classical guitar we can represent the chord progression described in the Example 3 as follows:



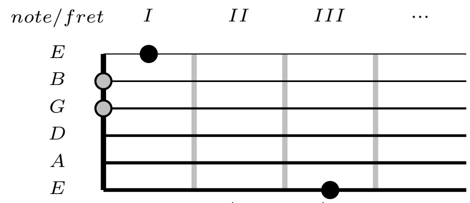
$C\ Major\ (Do\ Major)$



$A\ Minor\ (La\ Minor)$



D $Minor$ (Re $Minor$)



G $7th$ (Sol $7th$)

We use the gray dots to indicate that the string must be used without fret pressure.

Definition 6 (Tablature). *Given a string instrument $\mathcal{SI}_m$, we call* tablature *a finite sequence of position $TAB_1, ..., TAB_n$.*

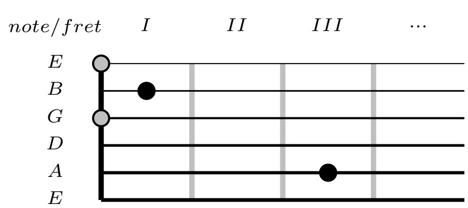
A position will be *realizable* on a string instrument $\mathcal{SI}$, iff a common hand can realize such position on $\mathcal{SI}$. Otherwise, the position will be bad for the instrument. Given an instrument $\mathcal{SI}$, let $T_{\mathcal{SI}}$ be the set of the positions on $\mathcal{SI}$. $T_{\mathcal{SI}} = \widehat{T}_{\mathcal{SI}} \bigcup \widetilde{T}_{\mathcal{SI}}$ where $\widehat{T}_{\mathcal{SI}}$ is the set of the realizable positions and $\widetilde{T}_{\mathcal{SI}}$ is the set of the bad positions.

Definition 7 (Expansion function). *Given an instrument $\mathcal{SI}_m$ we define the* expansion function *as follows:*

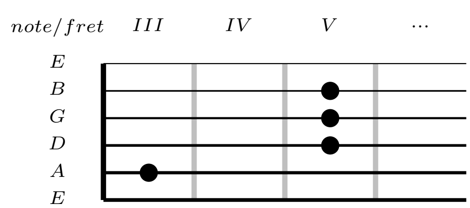
$$\delta : \Gamma_{\mathcal{SI}} \rightarrow 2^{\widehat{T}_{\mathcal{SI}}}$$
$$\textit{where } \forall \xi \in \Gamma_{\mathcal{SI}} \textit{ we have}$$
$$\delta(\xi) = \{TAB^{[\xi]} \mid TAB^{[\xi]} \in \widehat{T}_{\mathcal{SI}} \textit{ corresponds to the chord } \xi\}$$

With the previous definition we understand that, given a string instrument, we can associate a set of positions to the same chord. Extending the concept, we can associate to a specific chord progression a set of tablatures.

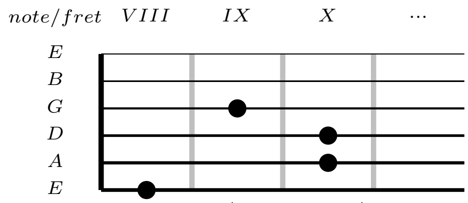
Example 5. Given a classical guitar we can play the same *C Major* defined in the Example 3 using the following positions:



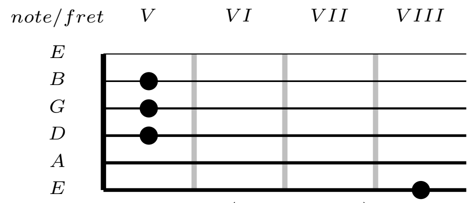
C $Major$ (Do $Major$)



C $Major$ (Do $Major$)



C Major (Do Major)



C Major (Do Major)

Proposition 1 (Automaton of chord progression). *Given a string instrument $\mathcal{SI}_m$ and a chord progression $\xi_1, ..., \xi_n$, we can define a deterministic finite state automaton $A = (q_{0,0}, \Omega, \Omega_f, \digamma)$ which represents the set of all possible tablatures for the progression in $\mathcal{SI}_m$, where:*

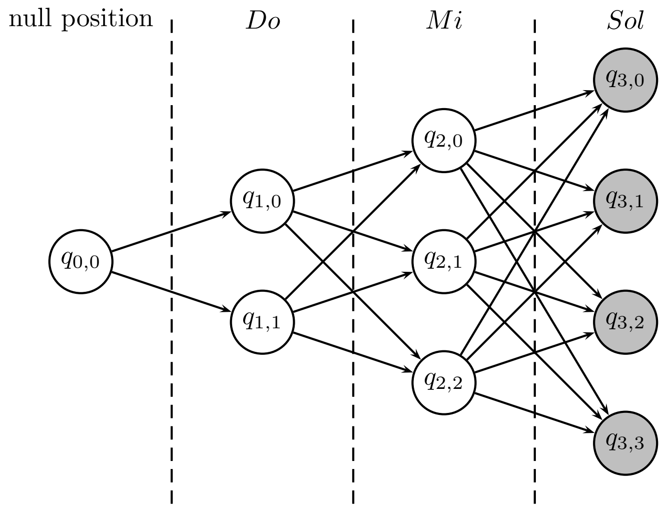
- $q_{0,0}$ *is the initial state;*
- Ω *is the set of the states;*
- $\Omega_f \subseteq \Omega$ *is the set of the final states;*
- $\digamma$ *is the transition function with* $\digamma : \Omega \times \widehat{T}_{\mathcal{SI}} \rightarrow \Omega$.

Proof. *We associate the null position to the initial state. Moreover,* $|\Omega - \{q_{0,0}\}| = \sum_{i=1}^{n} |\delta(\xi_i)|$ *where* $\forall \xi_i$ *we introduce the subset of states* $\{q_{i,0}, \ldots, q_{i,|\delta(\xi_i)|-1}\}$. *The transition function will be* $\digamma = \{(q_{i-1,j}, TAB_k^{[\xi_i]}, q_{i,k})\}_{i,j,k \in N}$ *with* $TAB_k^{[\xi_i]} \in \delta(\xi_i)$.

Example 6. We take into consideration the sequence C, E and G in the fourth octave, in other words $Do = \{48\}$, $Mi = \{52\}$, $Sol = \{55\}$. Using the classical guitar and the expansion function on these notes, we have the following positions:

$$
\begin{aligned}
\delta(\{48\}) = \{&\langle(1,\Box),(2,\Box),(3,\Box),(4,\Box),(\mathbf{5},\mathbf{3}),(6,\Box)\rangle,\\
&\langle(1,\Box),(2,\Box),(3,\Box),(4,\Box),(5,\Box),(\mathbf{6},\mathbf{8})\rangle\}\\
\delta(\{52\}) = \{&\langle(1,\Box),(2,\Box),(3,\Box),(\mathbf{4},\mathbf{2}),(5,\Box),(6,\Box)\rangle,\\
&\langle(1,\Box),(2,\Box),(3,\Box),(4,\Box),(\mathbf{5},\mathbf{7}),(6,\Box)\rangle\\
&\langle(1,\Box),(2,\Box),(3,\Box),(4,\Box),(5,\Box),(\mathbf{6},\mathbf{12})\rangle\}\\
\delta(\{55\}) = \{&\langle(1,\Box),(2,\Box),(\mathbf{3},\Box),(4,\Box),(5,\Box),(6,\Box)\rangle,\\
&\langle(1,\Box),(2,\Box),(3,\Box),(\mathbf{4},\mathbf{5}),(5,\Box),(6,\Box)\rangle\\
&\langle(1,\Box),(2,\Box),(3,\Box),(4,\Box),(\mathbf{5},\mathbf{10}),(6,\Box)\rangle\\
&\langle(1,\Box),(2,\Box),(3,\Box),(4,\Box),(5,\Box),(\mathbf{6},\mathbf{15})\rangle\}
\end{aligned}
$$

where every pair (c, t) indicate the string and the fret number respectively. In this way we obtain the following automaton:



Definition 8 (Distance between two positions). *Given a string instrument $\mathcal{SI}_m = (S_m, n_f)$, we define the* distance function *as follows:*

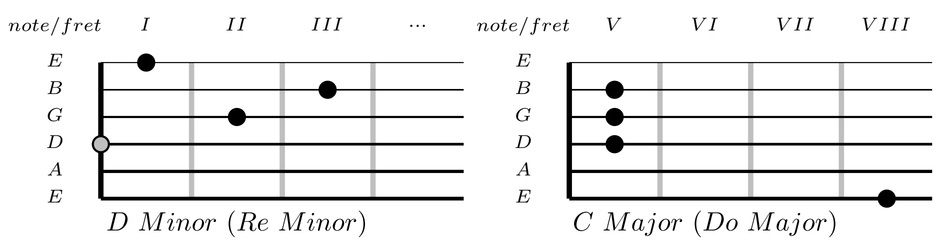
$$d : \widehat{T}_{\mathcal{SI}} \times \widehat{T}_{\mathcal{SI}} \rightarrow \{1, ..., n_f\}$$

where $\forall\ TAB_1, TAB_2 \in \widehat{T}_{\mathcal{SI}}$ *we set*

$$\lambda_1 = \min\{fret \neq \square \mid \exists j = 1, \ldots, m \text{ such that } TAB_1(j) = fret\},$$
$$\lambda_2 = \min\{fret \neq \square \mid \exists j = 1, \ldots, m \text{ such that } TAB_2(j) = fret\},$$

then $d(TAB_1, TAB_2) = |\lambda_1 - \lambda_2|$. *If* $\forall j\ TAB_1(j) = \square$ *or* $TAB_2(j) = \square$ *then* $d(TAB_1, TAB_2) = 0$.

Example 7. We take into consideration the following positions:



In this case the distance between the two positions is equal to four.

The concept of distance is important to determinate the complexity of a tablature. In fact, a greater distance requires a great deal of experience on the part of the performer. Therefore, we give the following:

Definition 9 (Tablature complexity). *Given a string instrument $\mathcal{SI}_m$, let $TAB_1, \ldots, TAB_n$ be a tablature for this instrument. We call* complexity *of the tablature the following quantity:*

$$\sum_{j=1}^{n-1} d(TAB_j, TAB_{j+1}).$$

Proposition 2. *Given a string instrument $\mathcal{SI}_m$ and a chord progression $\xi_1, \ldots, \xi_j$ for the instrument, let A be its associated automaton. We can obtain the following generating function:*

$$\Xi(t) = \sum_n \Xi_n t^n$$

which counts the number Ξ_n of tablatures having complexity equal to n.

Proof. We use the Schützenberger's methodology (or the symbolic method) to associate the indeterminate t to the distance between two sequential positions. Therefore, when we change the position from $TAB_k^{[\xi_i]}$ to $TAB_h^{[\xi_{i+1}]}$, the transition $q_{i,k} \to q_{i+1,h}$ becomes a term of the generating function $\Xi_{i,k}(t)$ in the form $t^{d(TAB_k^{[\xi_i]}, TAB_h^{[\xi_{i+1}]})} \Xi_{i+1,h}(t)$ where $d(TAB_{i,k}, TAB_{i+1,h})$ is the distance between the two positions. By solving the obtained equations system in the unknown $\Xi_{0,0}(t) = \Xi(t)$ we have the desired generating function.

Example 8. A very famous song is *Knocking on Heaven's Door* by Bob Dylan (see [2]). This song is a good example, because every strophe is characterized by the following chord progression:

G Major, D Major, A Minor, G Major, D Major, C Major

or, in Italian notation:

Sol Major, Re Major, La Minor, Sol Major, Re Major, Do Major.

Table 2. ***Knocking on Heaven's Door*** **by** ***Bob Dylan***

Introduction
G Major, D Major, A Minor, G Major, D Major, C Major

G Major D Major A Minor
Mama, take this badge off of me

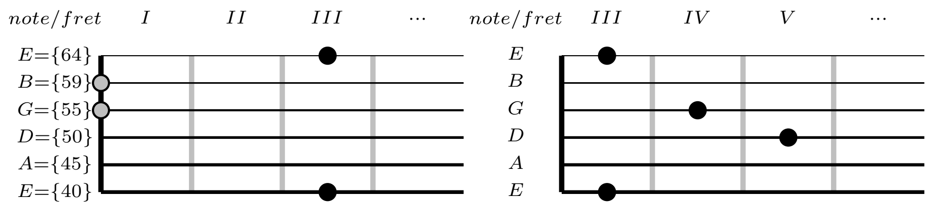
G Major D Major C Major
I can't use it anymore.

G Major D Major A Minor
It's gettin' dark, too dark for me to see

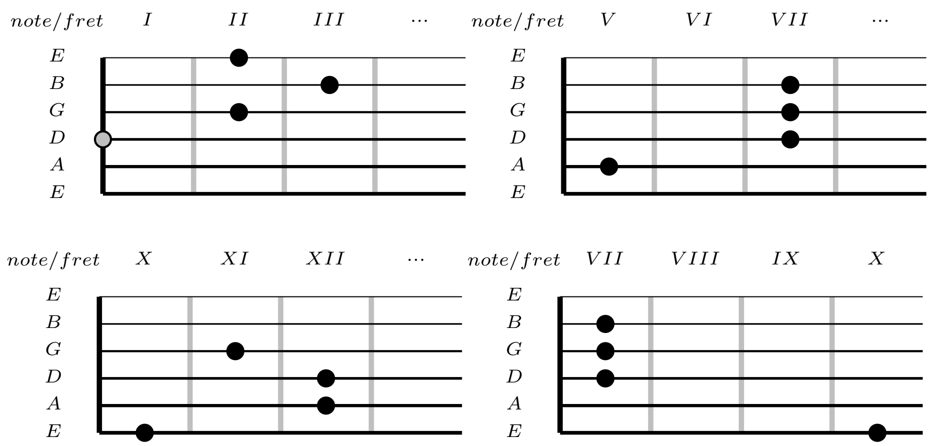
G Major D Major C Major
I feel like I'm knockin' on heaven's door.
. . .

We want to study the complexity of a single strophe of this song. Using the Definition 5 we can give the following realizable positions (you can find C Major positions in Example 4):

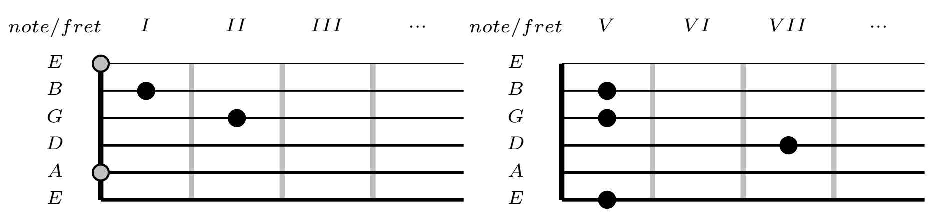
G Major (or Sol Major) = $\{67, 59, 55, 43\}$



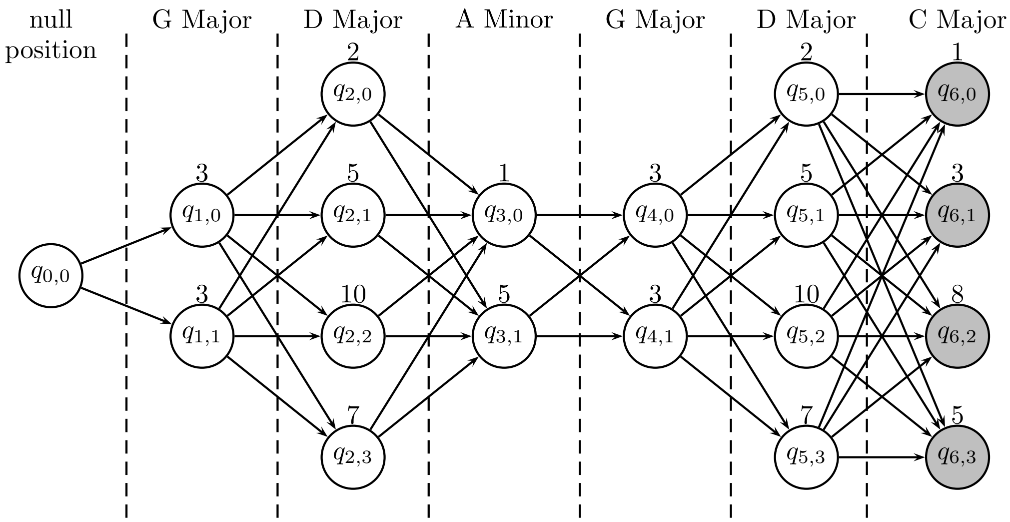
D Major (or Re Major) = $\{66, 62, 57, 50\}$



A Minor (or La Minor) = $\{64, 60, 57, 45\}$



Using Proposition 1 we can generate the following automaton:



In the previous figure, for all $q_{i,k}$ we have written also the

$$min\{fret \neq \square \mid \exists j = 1, \ldots, m \ such\ that\ TAB_k^{[\xi_i]}(j) = fret\}.$$

These values are necessary to compute the distance between two positions. Moreover, we can observe as, in this simple case, there are 512 different tablatures for the same chord progression. Using Proposition 2 we obtain the following system of equations:

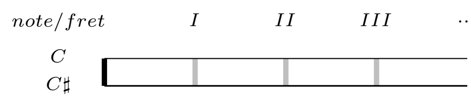
$$\begin{aligned}
\Xi_{0,0}(t) &= \Xi_{1,0}(t) + \Xi_{1,1}(t)\\
\Xi_{1,0}(t) &= t\Xi_{2,0}(t) + t^2\Xi_{2,1}(t) + t^7\Xi_{2,2}(t) + t^4\Xi_{2,3}(t)\\
\Xi_{1,1}(t) &= t\Xi_{2,0}(t) + t^2\Xi_{2,1}(t) + t^7\Xi_{2,2}(t) + t^4\Xi_{2,3}(t)\\
\Xi_{2,0}(t) &= t\Xi_{3,0}(t) + t^3\Xi_{3,1}(t)\\
\Xi_{2,1}(t) &= t^4\Xi_{3,0}(t) + \Xi_{3,1}(t)\\
\Xi_{2,2}(t) &= t^9\Xi_{3,0}(t) + t^5\Xi_{3,1}(t)\\
\Xi_{2,3}(t) &= t^6\Xi_{3,0}(t) + t^2\Xi_{3,1}(t)\\
\Xi_{3,0}(t) &= t^2\Xi_{4,0}(t) + t^2\Xi_{4,1}(t)\\
\Xi_{3,1}(t) &= t^2\Xi_{4,0}(t) + t^2\Xi_{4,1}(t)\\
\Xi_{4,0}(t) &= t\Xi_{5,0}(t) + t^2\Xi_{5,1}(t) + t^7\Xi_{5,2}(t) + t^4\Xi_{5,3}(t)\\
\Xi_{4,1}(t) &= t\Xi_{5,0}(t) + t^2\Xi_{5,1}(t) + t^7\Xi_{5,2}(t) + t^4\Xi_{5,3}(t)\\
\Xi_{5,0}(t) &= t + t + t^6 + t^3\\
\Xi_{5,1}(t) &= t^4 + t^2 + t^3 + t^0\\
\Xi_{5,2}(t) &= t^9 + t^7 + t^2 + t^5\\
\Xi_{5,3}(t) &= t^6 + t^4 + t + t^2
\end{aligned}$$

By solving in $\Xi_{0,0}(t)$ we obtain:

$$\Xi_{0,0}(t) = 24t^6 + 28t^8 + 16t^9 + 48t^{10} + 16t^{11} + 32t^{12} + 28t^{13} + 40t^{14} + 12t^{15} + 40t^{16} + 16t^{17} + 36t^{18} + 12t^{19} + 44t^{20} + 8t^{21} + 28t^{22} + 12t^{23} + 24t^{24} + 4t^{25} + 12t^{26} + 4t^{27} + 12t^{28} + 8t^{30} + 4t^{32} + 4t^{34}$$

This generating function counts the number of tablatures for each complexity. For example, the term $24t^6$ indicates the existence of 24 different tablatures with complexity equal to 6 to execute the chord progression. These 24 solutions correspond to tablatures requiring the shortest total movement of the hand on the fretboard. From the generating function we also find that the average complexity and the variance are: $\overline{\Xi} = 16.25$ and $\sigma = 141,195$. This means that if we play the song with a *random* tablature we make a total hand jump corresponding to 16 frets, on the average.

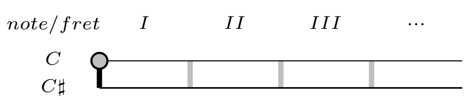
Example 9. We can use the previous proposition also with *infinite sequence of chords*. In fact, in this example we define a very simple string instrument $\mathcal{SI} = ((48, 49), n_f)$ as follows:



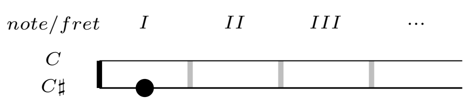
We study the tablatures of the following infinite notes progression:

$$C = \{48\}, D = \{50\}, C, D, C, D, ...$$

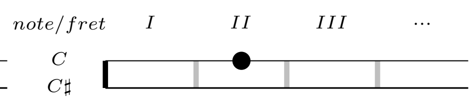
In this case note C has only one position, but note D has two positions.



C (or *Do*)



D (or *Re*)



D (or *Re*)

We have the following system of equations:

$$\begin{aligned}
\Xi_{0,0}(t) &= 1 + \Xi_{1,1} \\
\Xi_{1,1}(t) &= t\Xi_{2,0}(t) + t^2\Xi_{2,1} \\
\Xi_{2,0}(t) &= \Xi_{0,0}(t) \\
\Xi_{2,1}(t) &= \Xi_{0,0}(t)
\end{aligned}$$

We set the term 1 in $\Xi_{0,0}(t)$ because we can consider the initial state as a final state. By solving the system, we obtain:

$$\Xi_{0,0}(t) = 1 + t\Xi_{0,0}(t) + t^2\Xi_{0,0}(t)$$

and this equation corresponds to the generating function of Fibonacci's numbers F_n, in fact:

$$\Xi_{0,0}(t) = \frac{1}{1-t-t^2} = 1 + t + 2t^2 + 3t^3 + 5t^4 + O(t^5).$$

Therefore, in this example, there are F_n different tablatures with complexity n.

3 Conclusion

In this paper we introduce the problem of tablatures for stringed instruments and explain some combinatorial properties. We don't present an exhaustive study, in fact there are some questions which require a further study. In the Example 8 we have 24 tablatures with lowest complexity. Which one is the better? A tablature can be considered better also in terms of the mechanical difficulty of any single position. This concept is linked with the problem of fingering. In a next paper we will study this kind of problems.

References

1. Bass guitar. `http://en.wikipedia.org/wiki/Bass_guitar`.
2. Bob Dylan: Knockin' on heaven's door. `http://www.bobdylan.com/songs/knockin.html`.
3. Classical guitar. `http://en.wikipedia.org/wiki/Classical_guitar`.
4. Midi committee of the association of musical electronic industry. `http://www.amei.or.jp`.
5. Midi manufacturers association. `http://www.midi.org`.
6. D. Baccherini. Behavioural equivalences and generating functions. *preprint*, 2006.
7. D. Baccherini and D. Merlini. Combinatorial analysis of tetris-like games. *preprint*, 2005.
8. Ph. Flajolet and R. Sedgewick. The average case analysis of algorithms: complex asymptotics and generating functions. Technical Report 2026, INRIA, 1993.
9. Ph. Flajolet and R. Sedgewick. The average case analysis of algorithms: counting and generating functions. Technical Report 1888, INRIA, 1993.
10. J. R. Goldman. Formal languages and enumeration. *Journal of Combinatorial Theory, Series A*, 24:318–338, 1978.
11. D. Merlini, R. Sprugnoli, and M. C. Verri. Strip tiling and regular grammar. *Theoretical Computer Science*, 242,1-2:109–124, 2000.
12. M. Miura, I. Hirota, N. Hama, and M. Yanigida. Constructiong a System for Finger-Position Determination and Tablature Generation for Playing Melodies on Guitars. *System and Computer in Japan*, 35(6):755–763, 2004.
13. R. F. Moore. *Elements of computer music*, volume XIV, 560 p. Prentice-Hall, 1990.
14. S. Sayegh. Fingering for String Instruments with the Optimum Path Paradigm. *Computer Music Journal*, 13(6):76–84, 1989.
15. M. P. Schützenberger. Context-free language and pushdown automata. *Information and Control*, 6:246–264, 1963.

16. R. Sedgewick and P. Flajolet. *An introduction to the analysis of algorithms.* Addison-Wesley, 1996.
17. T. A. Sudkamp. *Languages and machines.* Addison-Wesley, 1997.
18. D. R. Tuohy and W. D. Potter. A genetic algorithm for the automatic generation of playable guitar tablature. *Proceedings of the International Computer Music Conference*, 2004.
19. H. S. Wilf. *Generatingfunctionology.* Academic Press, 1990.