FROM MIDI TO RICH TABLATURES: AN AUTOMATIC GENERATIVE SYSTEM INCORPORATING LEAD GUITARISTS' FINGERING AND STYLISTIC CHOICES

Pierluigi BONTEMPI (0000-0002-4360-8708) [1], **Daniele MANERBA** (0000-0002-3502-5289) [2],
Alexandre D'HOOGE (0000-0003-1634-3406) [3], and **Sergio CANAZZA** (0000-0001-7083-4615) [4]

[1,4]*Centro di Sonologia Computazionale, Department of Information Engineering*, **University of Padua**, Italy
[2]*Department of Information Engineering*, **Università degli Studi di Brescia**, Italy
[3]**Univ. Lille, CNRS, Centrale Lille, UMR 9189 CRIStAL**, F-59000 Lille, France

ABSTRACT

Although the automatic identification of the optimal fingering for the performance of melodies on fretted string instruments has already been addressed (at least partially) in the literature, the specific case regarding lead electric guitar requires a dedicated approach. We propose a system that can generate, from simple MIDI melodies, tablatures enriched by fingerings, articulations, and expressive techniques. The basic fingering is derived by solving a constrained and multi-attribute optimization problem, which derives the best position of the fretting hand, not just the finger used at each moment. Then, by analyzing statistical data from the *mySongBook* corpus, the most common clichés and biomechanical feasibility, articulations, and expressive techniques are introduced. Finally, the obtained output is converted into *MusicXML* format, which allows for easy visualization and use. The quality of the tablatures derived and the high configurability of the proposed approach can have several impacts, in particular in the fields of instrumental teaching, assisted composition and arranging, and computational expressive music performance models.

1. INTRODUCTION

Fretted string instruments (which include guitar, banjo, ukulele, and bass guitar, among others) allow the musician to play most of the notes on at least two different strings, at different frets. In addition, distinct fingers (usually index, middle, ring, or little finger) of the fretting hand can be used to produce the same note, on the same string and at the same fret. In some cases the notes can also be played on the open strings. In music, the term *fingering* is used to indicate which fingers and, consequently, hand positions should be used to perform a part [1]. Opting for well-reasoned fingerings is crucial, either for expressive reasons or to enable the musician to minimize performance effort.

The performer can then make choices, in addition to fingering, regarding articulations and expressive

techniques from a range of options generally associated with the specific instrument being played. The term *articulation* refers to how a single note is sounded. Each articulation corresponds to a specific performance technique and generates sounds endowed with distinctive sonic characteristics. The parameters involved may be attack, decay, timbre, dynamics, and in some cases even pitch. The term *expressive technique* instead refers to all those techniques used to imprint variations in sound parameters for specifically expressive purposes. Clearly, there may be at least partial overlap between the two categorizations. Since we specifically address lead electric guitar parts, in this context the most common articulations are those of the plucked string (usually with the use of a plectrum) and the legato obtained through *hammer-on* (a finger of the fretting hand strikes the string from above, causing it to resonate) or *pull-off* (a finger of the fretting hand "plucks/pulls" the string, when it is already vibrating, reaching a note lower than the preceding one). In terms of expressive techniques, we consider *vibrato* (the pitch of the note is made to oscillate with the use of the finger(s) of the fretting hand or the whammy bar), *slide legato* (or glissando), and *bending* (pushing the fingered string upward or pulling it downward, perpendicular to the neck, the pitch of the note is raised) [2].

In this paper, we propose an automatic system able to generate, from monophonic MIDI melodies (including note-on and note-off messages only) [3], electric guitar tablatures enriched by fingering, articulations, and expressive techniques. The basic fingering is derived by solving an optimization problem that considers the possibility of playing the same notes on different string-fret combinations or on open strings as well as biomechanical constraints, and that tries to minimize the time needed for the hand movements and the performer's discomfort, while also accounting for timbre characteristics associated with the generated tablature. The system also keeps track of the position of the fretting hand, not just of the finger used at each moment. Then, taking into account the most common clichés and biomechanical feasibility, articulations, and expressive techniques are introduced. The usage frequencies of such features are derived by analyzing statistical data from the *mySongBook* corpus, an impressive proprietary collection of guitar-oriented transcriptions, already used in the past in other publications [4, 5]. Finally, for ease of visualization

and use, the result is converted into *MusicXML* format.

Two important observations are in order. First, we address by choice guitar solos and melodies composed of single notes only and leave the possibility of playing multiple notes simultaneously for future system evolutions. However, it should be noted that lead guitar sections include a very high percentage (>90%) of single notes (see Cournut et al. 2021 [4]). Second, our system is based on symbolic encodings of music, i.e., MIDI, a specially developed dataframe, and MusicXML. Clearly, the automatic generation of transcriptions from audio recordings of guitar performances, a subject of significant attention in the literature (see, among other works, [6, 7]), shares the same object of study, but observes it backward, i.e., from performance to tablature or score. In our case, the goal is to move from a minimalistic score (represented in MIDI format) to a guitar tablature enriched with fingering, articulations, and expressive techniques.

The contribution of this work is multifaceted. On the one hand, earlier approaches to automatic fingering generation based on optimization techniques are somehow revived and greatly expanded (see Section 2). To the best of our knowledge, none of the approaches for automatic fingering generation presented in the past considered simultaneously all of the parameters and attributes described later, nor allowed for a level of user-side customization comparable to that proposed here. On the other hand, the automatic insertion of articulations and expressive techniques is deeply explored, while being a largely marginal subject of research in the past (a notable exception is the work by Sarmento et al. 2023 [8], although focused on automatic music generation). We also believe that this work can constitute an important piece toward the construction of models for computational musical expressiveness, in teaching, composition, assisted arranging of guitar parts, and, indirectly, in the stylistic analysis of guitarists. Lastly, the *mySongBook* corpus confirms itself to be a precious resource for understanding contemporary guitar practice and for the generation of automatic systems.

The rest of the paper is organized as follows. Section 2 presents the optimization approach followed to derive the basic fingering, while Section 3 presents first the *mySongBook* corpus and the statistical data derived from it, and then describes the rule system developed for the automatic addition of articulations and expressive techniques. In Section 4, the generation of the MusicXML code is introduced and an example lead guitar part processed by the system is presented, in the form of *rich* tablature and score in standard notation. Finally, we conclude in Section 5, also discussing some possible applications and future developments.

2. FROM A MIDI MELODY TO THE BASIC FINGERING

The automatic search for valid or optimal fingerings in relation to specific musical instruments has a scientific literature of significant size that cannot be entirely reviewed here. Hence, we first report the main existing approaches related to plucked string instruments,

highlighting the gaps we are trying to fill. Then, we describe our novel and comprehensive methodology.

2.1 Existing approaches for fingering optimization

Sayegh (1989) [9] tackles the fingering procedure by solving a Shortest Path Problem (SPP) on a weighted and layered directed graph (a so-called *Viterbi* network) in which each layer corresponds to each note of the sequence and contains nodes corresponding to admissible fingerings of such a note. Since only position and string changes between a note and the consecutive one affect the fingering attributes, the weights of such attributes can be directly associated with arcs between two consecutive layers. Concerning the considered attributes, two broad classes of rules may be identified. The first one relates to ease of execution and the second to the homogeneity of the sound generated.

A similar approach, albeit slightly more refined, is proposed in Radicioni et al. (2004) [10]. Here, not only movements along and across the fretboard are taken into consideration, but also finger span, and minimum/maximum spans between each pair of fingers are used to evaluate the level of difficulty of the transitions between notes and identify when hand repositioning is needed. The sequence is segmented into musical phrases at the preprocessing phase and the optimization takes place first of all within the individual phrases, then a valid connection between phrases is created. The approach of Miura et al. (2004) [11] is also based on the segmentation of the melody while the objective is to minimize the number of movements of the fretting hand wrist (along the neck) within each section. The trade-off emerges between the increase of finger span and the hand position changes, where the latter is however considered less uncomfortable for a beginner. Interestingly enough, techniques such as hammer-on, pull-off, and slide are considered.

Other published works focus more on algorithmic aspects of the fingering. In particular, Tuohy and Potter (2005) [12] propose a Genetic Algorithm for creating polyphonic guitar tablatures. Their fitness function is based on hand movements along and across the neck, and fretting hand configurations. However, the system does not include information about which finger should be used to play each note. A Hidden Markov Model (HMM) is used in Hori et al. (2013) [13] to derive the basic fingering of monophonic and polyphonic guitar parts. The fretboard hand arrangements are considered as the hidden states, and the note sequence as the observed one. In a subsequent work by Hori and Sagayama (2016) [14], the authors try to simplify the most difficult movements required to play a part, instead of aiming at minimizing the difficulty of the overall execution. The paper introduces a variant of the basic algorithm for multi-layered networks, called the *minimax Viterbi algorithm*, to solve the problem. When applied to an HMM-based fingering decision network, the algorithm looks for the sequence of the hidden states that maximizes the minimum transition probability on the sequence (transition probabilities are large for easy moves and small for difficult ones). The reader is referred

to Roohnavazfar et al. (2019) [15] for more recent and sophisticated approaches for finding optimal paths in general multi-stage decision networks.

Finally, a Dynamic Programming approach is used in Raboanary et al. (2017) [16] to minimize performance effort on monophonic bass guitar lines, with special emphasis on limiting movements along the neck.

2.2 A new comprehensive optimization framework

Our approach for generating the basic fingering consists in the formalization, development, and resolution of an optimization problem, which generalizes all those already presented in the literature that appeared under the simplified form of an optimal path search through a multi-layered network (see, e.g., Sayegh 1989 [9]). In particular, we enrich the quality of the fingering by introducing i) additional requirements, that affect the available feasible fingerings for each note of the melody, and ii) simultaneous attributes to optimize, that affect the cost of the transition from the fingering of a note to that of the following one.

Concerning the attributes of the fingering to be optimized, we identified three macro-categories:

- *time-related* attributes: relate to movements of the fretting hand along (*position change*, PC) and across (*string change*, SC) the neck. Their weights depend on the time needed to physically reach the fingering by the hand and the fingers.

- *discomfort-related* attributes: relate to finger span (*hand spread*, HS) and to playing notes on specific parts of the fretboard (e.g., very near or very far from the headstock). Their weights depend on the physical discomfort generated by the fingering in the player.

- *timbre-related* attributes: relate to attributes that affect the type of sound output produced. Their weights depend on the sound quality generated by the fingering, according to the player's intent and taste. In this regard, we consider here the frequency of use of open strings.

We remark that, differently from most of the existing approaches, the system keeps track of the fretting hand position (i.e., the fret at which the index finger is located at each moment) to estimate the spreads between fingers and the longitudinal hand movements.

Moreover, our optimization setting provides the user with great flexibility, allowing the customization of several aspects and preferences. In particular, the system can be finely configured about:

- the intrinsic characteristics of the instrument (number of strings and frets, and notes generated by each open string or string/fret combination);

- the fretboard area on which it is more comfortable to play. Typically the preferred area is the middle of the neck, between the fifth and twelfth positions, as

noted by Cournut et al. (2021) [4] and by Heijink et al. (2002) [17] concerning classical guitar;

- the maximum and the minimum allowed spread between each finger and the index finger. Constantly keeping track of the position of the index finger allows for a potentially more accurate assessment of the hand spread than the approach traditionally followed, which only monitors the spread between the finger used to play one note and the one used for the immediately following one. It also makes it possible to correctly assess position changes, i.e., shifts of the hand along the neck (see below);

- the time required for the hand to make longitudinal movements (proportional to the number of frets to be covered) and vertical movements (proportional to the number of strings covered). Transitions for which the *Inter Onset Interval* (IOI) is not sufficient are discarded from the possible alternatives;

- how fingers are used. For example, it could be denied to use the same finger to play two contiguous notes unless they are played at the same fret (as by a *barrè* o *semi-barrè* technique) or on the same string (so to allow sliding techniques);

- the proportional level of penalty to be associated with each attribute to be optimized, e.g., for position changes, string changes, use of open strings, finger spread, and distance from the optimal fretboard area.

2.3 Implementation details

In the following, we briefly describe the system implementation from a purely technical point of view.

Given a monophonic melody in MIDI format, a dataframe is generated in Python language using the *Mido* library (https://github.com/mido/mido, last accessed: 2024/03/07). For each note, the dataframe contains pitch, start and end positions in time, duration, IOI, and all the data needed later to generate the final MusicXML file. Data on the structure of the instrument (number of strings and frets, and notes generated by each open string and string/key combination) and the configuration parameters of our optimization problem are also generated.

The obtained dataframe is then processed through the *IBM ILOG CPLEX Optimization Studio* software, v20.1.0 (https://www.ibm.com/products/ilog-cplex-optimization-studio, last accessed: 2024/03/07), in which we implemented our optimal fingering problem as a static and compact Mathematical Programming formulation. The problem is described via the CPLEX's dedicated *Optimization Programming Language*. The software returns the string/fret combination (and the associated finger to be used to press the string) or the open string on which to play each note implying the best achievement in terms of time-, discomfort-, and timbre-related attributes.

Figures 1 and 2 show the same melody processed by the optimizer with different attributes' importance.
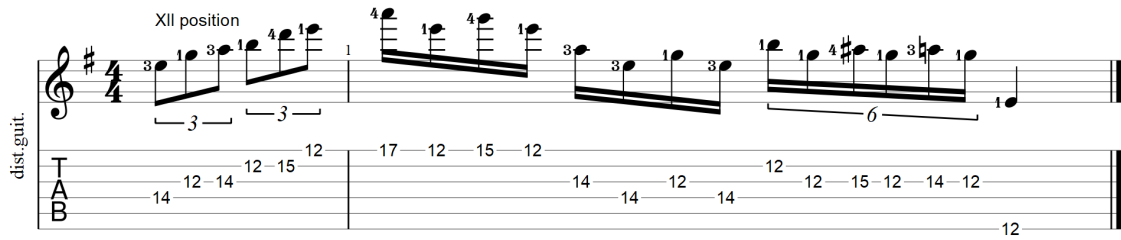
Figure 1. To obtain this solution, the least penalized feature was the string change (SC). The hand spread (HS) was penalized twice as much as SC, while the position change (PC) was penalized twice as much as HS.
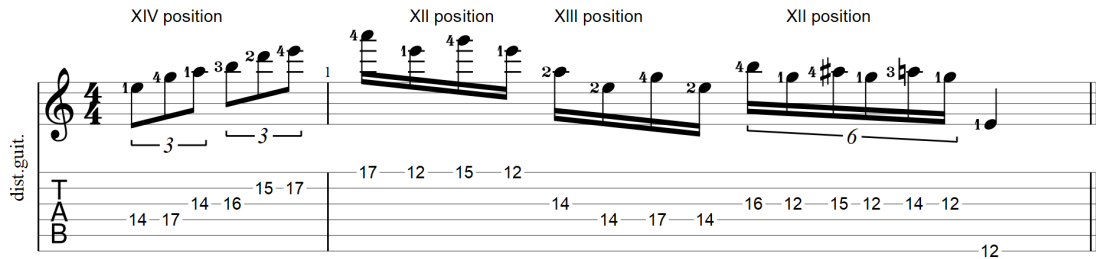

Figure 2. To obtain this solution, the least penalized feature was HS. PC was penalized twice as much as HS. SC was penalized twice as much as PC.

The numbers placed next to the heads of the notes in standard notation refer to the finger of the fretting hand to be used, i.e., 1-index, 2-middle, 3-ring, and 4-little finger. Both results are biomechanically valid, but significantly different from each other, reflecting the high configurability of the approach.

3. FROM THE BASIC FINGERING TOWARD A *RICH* TABLATURE

In this Section, we describe how the dataframe is enriched with articulations and expressive techniques. First, we extracted the distributions of the main techniques from the *mySongBook* dataset to use them as statistical references. Then, we use a rule-based system for the actual insertion of the techniques, allowing wide freedom of configuration to the user.

3.1 Statistical data extraction

The *mySongBook* corpus comprises more than 2000 scores/tablatures transcribed by professional musicians (https://www.guitar-pro.com/c/18-music-scores-tabs, last accessed: 2024/03/07), of different genres. The transcriptions are made with *Guitar Pro* software, produced by *Arobas Music*, and are geared first and foremost to an audience of guitarists. However, they usually also contain vocal parts or parts of other instruments, such as bass guitar or drums. Given the object of investigation of this paper, the statistical data described below were extracted from the sections labeled as *lead electric guitar parts*. These parts are extracted after eliminating rhythmic guitar sections, identified using the procedure described in Régnier et al.

(2021) [18]. Then, the quantity of playing techniques on each string is determined and divided by the total number of notes per string to obtain the proportion of notes with each technique. The obtained ratios are illustrated in Figure 3.
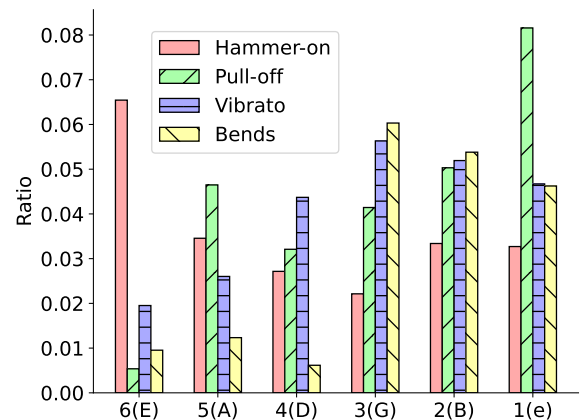

Figure 3. Ratio of the use of the considered techniques to the total number of notes for each string. Strings are labeled according to guitar standard tuning, the 6-th string being the low "E" and the 1-st string the high "e".

It can be observed that vibrato and bends are used similarly on higher strings. However, bends are rare on lower strings because thicker strings are harder to bend, while a vibrato is a smaller movement that is manageable even on stiffer strings. It should also be noted that, even if hammer-on and pull-off are techniques theoretically similar, their distribution across strings is quite different.

Our system uses the statistical data presented for deciding on the insertion of articulations and expressive techniques, as described in Section 3.2.

The data were also used as a comparison benchmark for the basic fingerings generated by the optimization procedure. On a test dataset made up of lead parts comprising a total of around 1500 notes and by setting the configuration weights in such a way as to balance the different discomfort factors, it was observed that the distribution of notes for each string was reasonably similar to that found in the entire *mySongBook* corpus (Figure 4).
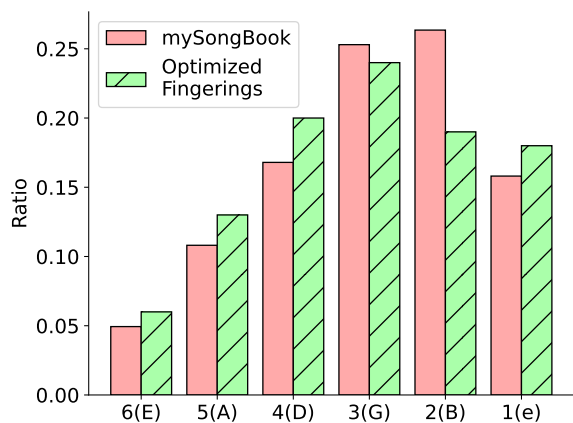

Figure 4. Ratio of the notes played on the different strings to the total number of notes in the test dataset (applying our optimized fingering) and the entire *mySongBook* corpus.

Finally, specifically regarding bending techniques, a manual analysis of numerous transcriptions taken from the *mySongBook* corpus and from teaching manuals [19–21] revealed some typical situations where bends are used, specifically in the presence of two or more consecutive notes in unison, or of groups of three consecutive notes with first and third notes equal to each other and the second one with higher or lower pitch within the limits consonant with the technique in question (e.g., G4-A4-G4, where A4 is performed in bending, keeping the fret/chord/finger combination steady), or when two contiguous notes can be played at the same fret of the same string, using a bend on one of them. Such typical situations are again used as a rule-of-thumb for bending decisions, as described in the next Section.

3.2 Inclusion of articulations and expressive techniques

The dataframe generated from the MIDI melody, combined with the fingering solution obtained from the optimization procedure, is further processed in Python to insert articulations and expressive techniques. Those considered are listed below, along with a summary of the rules for evaluating the possibility or opportunity of their execution:

- hammer-on: only the parameter of biomechanical feasibility is evaluated. If the note is preceded by

another lower-pitched note on an open string or played with a lower finger on the same string, the articulation is possible;

- pull-off: only the parameter of biomechanical feasibility is evaluated. If the note is preceded by another higher-pitched note played with a higher finger on the same string, the articulation is possible. The pull-off is always possible between fretted and open strings;

- vibrato: two user-customizable duration values are set; if the note duration reaches the first, the expressive technique is considered possible; if it reaches the second it is estimated highly likely. Vibrato is not allowed on open strings;

- slide legato: only the parameter of biomechanical feasibility is evaluated. If a note is followed by another note on the same string and at a different fret, played with the same fretting hand finger, the articulation is possible;

- bending: the field of bending is highly articulated and complex to model [5]. For this first implementation of the software, only basic upward bends (and possibly the releases associated with them, if the following note is played with the same finger and at the same string/fret combination) were considered. The user is first asked to set the maximum amplitude in semitones of the bends that can be performed by each finger of the fretting hand. Setting this parameter to 0 excludes fingers (typically the little finger) from the possibility of doing bends. Bending is considered possible if the hypothetical application note has a duration equal to or greater than a user-settable value. Biomechanical compatibility between the execution of the bending and the fingering of the previous and next notes is also taken into consideration. The procedure tries to include bends by recreating where possible the cliché situations described in Section 3.1.

The actual distribution of articulations or expressive techniques depends on the user's choices and the reference statistics described in Section 3.1. The user can let such statistics be taken as targets to be achieved, consistent with the biomechanical possibility of the execution. In other words, the procedure tries to include, where possible according to the conditions described above, the techniques to respect the proportions between technique use and the total number of notes for each string. If the points of possible insertion of the techniques are insufficient in number to reach the targets, the procedure will simply insert the techniques wherever possible, without forcing their insertion into unnatural contexts. If, on the other hand, the potential insertion points are more numerous than the target, the use of the different techniques is chosen on a random basis, to get as close as possible in the overall part to the target values. In the case of vibrato, priority is given to the longest notes.

The user also has the possibility to customize the reference targets in relation to each of the above-mentioned techniques, so as to be able to better adapt the system's response according to musical genre, personal preferences, or educational needs. The user can also ask the procedure to insert vibratos, hammer-ons, and pull-offs wherever possible. Only in the case of the slide legato, given the small number of potential occurrences, it was deemed appropriate to limit ourselves to asking the user if s/he would like different notes played with the same finger on the same string to be connected with this technique.

We remark that, clearly, the system also takes into consideration the compatibility between the different techniques, so as not to overlap conflicting ones.

4. MUSICXML CONVERSION AND GRAPHICAL RENDERING

Once the dataframe is completed, including all useful information concerning pitch, timing, use of open or fingered strings, articulations, and expressive techniques, a file in MusicXML format is generated via a dedicated Python script. The MusicXML format is considered a standard for the interchange and distribution of musical scores, provides for tablature encoding, and is supported by many music notation programs [22, 23]. We opted for *Guitar Pro 8*, to obtain a graphical rendering of the processed parts.

The final result concerning a sample lead part, processed by the entire system, is shown in Figure 5. The melody is an original work composed by the first author, but any MIDI melody could be used. The part was specifically written by imagining a standard 12-bar blues harmonic progression in A. Here, the reader may notice that the fingering is biomechanically valid and comfortable. The rich presence of bends, vibratos, hammer-ons, and pull-offs is because higher target values were given when inserting articulations and expressive techniques than those found in the *mySongBook* corpus, to better embody the blues or rock/blues feel of the phrasing [21]. A slide legato also appears in the last bar.

5. CONCLUSIONS

We presented an innovative automatic system for generating (starting from simple MIDI melodies) lead guitar tablatures, enriched with fingering as well as articulation and expressive techniques. For the basic fingering, we used an optimization approach significantly richer than the ones already existing in the literature, that allowed us to embed more complex requirements. Moreover, we also developed a procedure that provides a very high level of customization for the user, both in terms of biomechanical preferences/requirements and stylistic intents. Starting from the basic fingering, a user-customizable rule-based system, which evaluates note by note the biomechanical feasibility and probability of use of specific articulations and expressive techniques, introduces such techniques whenever appropriate. Finally, the generated dataframe is converted into MusicXML

format, compatible with numerous music notation software. The code is available upon request to the authors.

The computational generation of optimal or alternative *rich* fingerings for a given melodic part can be of interest, beyond the support of the performer in the study phase, in multiple areas such as teaching [24], automatic music generation [25, 26], computational performative expressiveness [27, 28], stylistic analysis, evaluation of existing tablatures [29], and testing of audio-to-tablature models. Both the Centro di Sonologia Computazionale of the University of Padua [30] and the TABASCO project team (http://algomus.fr/tabasco/, last accessed: 2024/05/16), to which part of the authors belong, share previous relevant work and scientific interest in relation to these issues. We believe that in each of the fields mentioned above the verification of performative plausibility and the possible generation of alternative *rich* fingerings can prove valuable (if not indispensable) aids.

The future improvements of the system's functionalities may comprise the management of lead parts including multiple simultaneous notes, the possibility of inserting decisions concerning articulations and expressive techniques within the optimization procedure, as well as the application of the system in different fields such as teaching (even self-taught) and computational musical expressiveness. Moreover, the much-needed integration of a system such as the one proposed here into the workflow of contemporary music production, often based on the use of virtual instruments, would allow even those who are not expert guitarists to obtain more realistic and convincing parts. Finally, it should be noted how a system such as the one proposed cannot be easily validated, as a given starting melody may correspond to many divergent solutions, equally valid but responding to different expressive intentions and personal preferences. In other words, it would not be correct to compare the output of the system to specific actual performances or existing transcriptions to sanction or not its validity. Although the results obtained so far have been evaluated by the authors as fully satisfactory and realistic, it would therefore be desirable in the future to subject them to the judgment of a wider audience of experienced guitarists.

6. REFERENCES

[1] M. Lindley, G. Jenkins, R. Slatford, P. Walls, S. Monosoff, A. Crum, S. Wijsman, A. Myers, M. Ecochard, and B. Haynes, "Fingering," in *Grove Music Online*. Oxford University Press, 2001.

[2] D. R. Grimes, "String theory - the physics of string-bending and other electric guitar techniques," *PLOS ONE*, vol. 9, no. 7, pp. 1–9, 07 2014.

Figure 5. Rich tablature and standard notation of a sample lead part automatically generated by our system.

[3] B. Moog, "Midi," *Journal of Audio Engineering Society*, vol. 34, no. 5, 1986.

[4] J. Cournut, L. Bigo, M. Giraud, N. Martin, and D. Régnier, "What are the most used guitar positions?"

in *Proceedings of the International Conference on Digital Libraries for Musicology (DLfM)*, July 2021, pp. 84–92.

[5] A. D'Hooge, L. Bigo, and K. Déguernel, "Modeling

bends in popular music guitar tablatures," in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, November 2023.

[6] J. Michelson, R. Stern, and T. Sullivan, "Automatic guitar tablature transcription from audio using inharmonicity regression and bayesian classification," in *Audio Engineering Society Convention 145*. Audio Engineering Society, 2018.

[7] A. Wiggins and Y. E. Kim, "Guitar tablature estimation with a convolutional neural network." in *ISMIR*, 2019, pp. 284–291.

[8] P. Sarmento, A. Kumar, D. Xie, C. Carr, Z. Zukowski, and M. Barthet, "Shredgp: Guitarist style-conditioned tablature generation with transformers," in *Proceedings of the International Symposium on Computer Music Multidisciplinary Research*. Zenodo, 2023.

[9] S. I. Sayegh, "Fingering for string instruments with the optimum path paradigm," *Computer Music Journal*, vol. 13, no. 3, pp. 76–84, 1989.

[10] D. Radicioni, L. Anselma, V. Lombardo *et al.*, "A segmentation-based prototype to compute string instruments fingering," in *Proceedings of the Conference on Interdisciplinary Musicology*, vol. 17, 2004, p. 97.

[11] M. Miura, I. Hirota, N. Hama, and M. Yanagida, "Constructing a system for finger-position determination and tablature generation for playing melodies on guitars," *Systems and Computers in Japan*, vol. 35, no. 6, pp. 10–19, 2004.

[12] D. Tuohy and W. Potter, "A genetic algorithm for the automatic generation of playable guitar tablature," in *Proceedings of the International Computer Music Conference (ICMC)*, 01 2005, pp. 499–502.

[13] G. Hori, H. Kameoka, and S. Sagayama, "Input-output hmm applied to automatic arrangement for guitars," *Journal of Information Processing*, vol. 21, no. 2, pp. 264–271, 2013.

[14] G. Hori and S. Sagayama, "Minimax viterbi algorithm for hmm-based guitar fingering decision." in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, 2016, pp. 448–453.

[15] M. Roohnavazfar, D. Manerba, J. C. De Martin, and R. Tadei, "Optimal paths in multi-stage stochastic decision networks," *Operations Research Perspectives*, vol. 6, p. 100124, 2019.

[16] T. H. Raboanary, F. H. Randriamahenintsoa, H. A. Raboanary, T. M. Raboanary, and J. A. Raboanary, "Finding optimal bass guitar fingerings," in *Proceedings of the IEEE Africon Conference*. IEEE, September 2017, pp. 65–71.

[17] H. Heijink and R. G. Meulenbroek, "On the complexity of classical guitar playing: functional adaptations to task constraints," *Journal of Motor Behavior*, vol. 34, no. 4, pp. 339–351, 2002.

[18] D. Régnier, N. Martin, and L. Bigo, "Identification of rhythm guitar sections in symbolic tablatures," in *Proceedings of the International Society for Music Information Retrieval (ISMIR) Conference*, November 2021.

[19] P. Fischer, *Masters of Rock Guitar*. Ama Verlag, 2000.

[20] T. Stetina, *Metal lead guitar*. Hal Leonard, 1992.

[21] J. Wheatcroft, *Improvising Blues Guitar*. Schott & Co. Ltd, 2008.

[22] M. Good, "Musicxml: An internet-friendly format for sheet music," in *Proceedings of XML conference and exposition*, December 2001, pp. 03–04.

[23] M. Good and G. Actor, "Using musicxml for file interchange," in *Proceedings of the International Conference on WEB Delivering of Music (WEDELMUSIC)*. IEEE, 2003, p. 153.

[24] T. Izumi, "Cognitively oriented design of a multimedia system to learn guitar fingering," in *Proceedings of the International Conference on Computers in Education (ICCE)*, vol. 1, 2002, pp. 789–791.

[25] F. Carnovalini and A. Rodà, "Computational creativity and music generation systems: An introduction to the state of the art," *Frontiers in Artificial Intelligence*, vol. 3, 2020.

[26] M. Civit, J. Civit-Masot, F. Cuadrado, and M. J. Escalona, "A systematic review of artificial intelligence-based music generation: Scope, applications, and future trends," *Expert Systems with Applications*, vol. 209, p. 118190, 2022.

[27] C. E. Cancino-Chacón, M. Grachten, W. Goebl, and G. Widmer, "Computational models of expressive music performance: A comprehensive and critical review," *Frontiers in Digital Humanities*, vol. 5, 2018.

[28] P. Bontempi, S. Canazza, F. Carnovalini, and A. Rodà, "Research in computational expressive music performance and popular music production: A potential field of application?" *Multimodal Technologies and Interaction*, vol. 7, no. 2, 2023.

[29] A. Radisavljevic and P. F. Driessen, "Path difference learning for guitar fingering problem," in *Proceedings of the International Computer Music Conference (ICMC)*, November 2004.

[30] S. Canazza and G. De Poli, "Four decades of music research, creation, and education at Padua's Centro di Sonologia Computazionale," *Computer Music Journal*, vol. 43, no. 4, pp. 58–80, 2020.