

2013

Chords not required: Incorporating horizontal and vertical aspects independently in a computer improvisation algorithm

Martin Norgaard

Georgia State University, mnorgaard@gsu.edu

Mariana Montiel

Georgia State University, mmontiel@gsu.edu

Jonathan Spencer

Georgia State University

Follow this and additional works at: https://scholarworks.gsu.edu/music_facpub



Part of the [Music Commons](#)

Recommended Citation

Norgaard, M., Montiel, M., & Spencer, J. (2013). Chords not required : Incorporating horizontal and vertical aspects independently in a computer improvisation algorithm. In A. Wiliamon & W. Goebel (Eds.), *Proceedings of the International Symposium on Performance Science 2013* (pp. 725–730).

This Article is brought to you for free and open access by the School of Music at ScholarWorks @ Georgia State University. It has been accepted for inclusion in Music Faculty Publications by an authorized administrator of ScholarWorks @ Georgia State University. For more information, please contact scholarworks@gsu.edu.

Chords not required: Incorporating horizontal and vertical aspects independently in a computer improvisation algorithm

Martin Norgaard¹, Mariana Montiel², and Jonathan Spencer³

¹ School of Music, Georgia State University, USA

² Department of Mathematics and Statistics, Georgia State University, USA

³ Department of Computer Science, Georgia State University, USA

Most previous computer improvisation algorithms for tonal jazz create an output based primarily on the underlying chord progression. This approach may partly ignore melodic continuity often seen in transcriptions of traditional jazz artists. Here we suggest a different approach implemented as a computer algorithm that creates material solely based on probabilities related to past note choices. This approach aligns with theoretical work suggesting that stored motor patterns are the basis of improvised music. Our computer algorithm analyzes pitch and rhythm patterns from a given corpus and then creates improvisations using this information. We describe an example in which a corpus of 48 solos by jazz saxophonist Charlie Parker was used by our algorithm to create an improvisation of the same length. The artificial corpus contained pattern structures similar to that of the original corpus. In contrast, previous research by one of the authors showed that a chord-based computer algorithm generated an output with a pattern structure very different from that of the human improviser even though the same chord structure was used as input. Future work will add a vertical aspect to our model in which a given chord pattern influences note choices in addition to the current horizontal focus.

Keywords: improvisation; patterns; rules; computer modeling; jazz

Two prominent theories have been proposed to explain how improvised musical material is created. One theory posits that stored fragments of musical material is retrieved from memory and linked during improvisation (Pressing 1988). Pressing divided improvisations into collections of concatenated note

groupings. Each grouping is triggered by a creative intention in the form of a mental schema that contains a cognitive image of sound and corresponding motor realization. His theory implies that these mental schemas are retrieved from a stored library. Therefore, if his theory is accurate, improvisations by artist-level improvisers should contain repeated melodic and rhythmic figures as the improviser repeatedly accesses the same mental schema from this library.

A competing theory emphasizes the role of tonal rules and how these rules may guide the improvising performer without storing and reusing material (Johnson-Laird 2002). According to this view, improvisations may still contain repeated melodic figures but they appear by chance due to the guiding tonal rules or are temporarily stored to be used again only within the same improvisation. Johnson-Laird wrote a computer program that can create jazz bass lines from a given chord progression using rules in support of his theory. Similarly, the computer program Impro-Visor uses rules to create monophonic jazz improvisations based on a given chord progression (Gillick *et al.* 2010). One problem with this approach is its dependence on a given chordal framework.

A previous study by one of the authors explored the use of patterns in tonal jazz by analyzing a large corpus of improvisations by the jazz saxophonist Charlie Parker (Norgaard in press). The study showed extensive use of patterns lending support to Pressing's theory that improvisers develop a stored library of patterns serving as the basis for new improvisations. In that study, interval and rhythm patterns were investigated starting on each note position. Results showed that 82.6% of all notes in the corpus began a four-interval pattern and 57.6% began interval and rhythm patterns. Furthermore, patterns up to 49-intervals were identified. Importantly, many of the longer interval patterns were distributed over several improvisations recorded at different times. Specifically, of the 98 identified unique patterns of 15-intervals or longer, 61% occurred in different solos suggesting that these patterns were not temporarily stored during the current improvisation as suggested by Johnson-Laird (2002).

Improvisations containing a large number of repeated patterns may appear more structurally sound as listeners focus on the melodic line. Artist-level jazz improvisers weigh both melodic (horizontal) and chordal (vertical) considerations during improvisation (Berliner 1994, Norgaard 2011). Improvisers may follow the logic of the horizontal line to create material that may or may not fit the actual chords. In a previous qualitative study, one participant described a phrase as having "no map behind that part" referring to the lack of attention to the underlying harmonic "map" (Norgaard 2011, p. 120).

The current project further explores the possibility that improvisations are based on a library of patterns through the implementation of a computer algorithm for improvisation based on this principle. We compare the output from our algorithm both with the results from the Parker study but also with results of a competing computer algorithm. This algorithm uses a rule-based approach where the output is dictated by the underlying chord progression.

The majority of previous computer models of improvisation are based on strict relationships between the improvised line and the underlying chords (Gillick *et al.* 2010, Johnson-Laird 2002, Rolland and Ganascia 2000). In two of these models, grammars based directly on the underlying chord progression are used to create improvised material, thereby overemphasizing vertical elements in improvisational thinking (Johnson-Laird 2002, Keller and Morrison 2007). To counter this bias, the current computer model emphasizes the horizontal aspect exclusively. In future work, we plan to further develop our model to take a given chordal structure into consideration. Our final goal is to create software for improvisation in which both horizontal and vertical aspects interact in a manner that more accurately reflects the thinking of artist-level jazz improvisers.

MAIN CONTRIBUTION

In the previous study, transcriptions of 48 improvisations by Charlie Parker were included in the corpus for analysis (Norgaard in press). The master MIDI file was imported into the Matlab computer environment using a modified version of the Midi Toolbox for Matlab (“Matlab” 2011, Smit n.d.). Then, for example, the use of five-note patterns was investigated by searching for four-interval patterns with an algorithm within Matlab that, starting with the first four intervals (e.g. +2, +2, +1, -1), looked for additional occurrences of this interval sequence in the corpus. The result represented the number of times the interval pattern +2, +2, +1, -1 occurred in the corpus and the number of times the pattern starting on this note position reappeared in the corpus. The program then went on to the interval pattern starting on the following note and looked for the number of occurrences of this pattern. Using this procedure, the number of patterns occurring on each note position was reported. A similar procedure using beat onset times was used to investigate rhythm patterns.

Building on this previous work we decided to find a way to concatenate pitch patterns to generate improvisations based on transitional probabilities within a given corpus. The first step was to create a model for the melody (intervals). In the algorithm, a change in pattern was determined by deleting

the first interval of the previous pattern and adding a new interval. For example, a 4-interval pattern, [2212], might be followed by [2121], with the result that the concatenated interval sequence would be [22121]. Indeed, in this case, the following pattern was contingent upon the last 3 intervals of the preceding pattern.

Results

The initial version of the algorithm only incorporated pitches (see Figure 1). The examples provided are all improvisations based on the probabilities extracted from the Parker corpus mentioned above.

After evaluating the result of the melody algorithm for the concatenation of the patterns, we decided to continue in the same vein for the treatment of rhythm. To keep these techniques as similar as possible, rhythmic patterns were also decided to span 5 notes. A rhythmic pattern is a combination of four contiguous note durations and the time from each of these note onsets to the next. We tested this approach by superimposing separately-generated rhythm and pitch improvisations into the same improvisation. This melody/rhythm algorithm created improvisations in which both pitch and rhythm patterns were present but where no relationship existed between the two parameters (see Figure 2).

In the music created by a human improviser, typically there is a relationship between pitch and rhythm patterns. For example, Charlie Parker often plays arpeggiated chords using a triplet rhythm. Therefore the final implementation of our algorithm takes this relationship into account. In the latest working version of our algorithm, rhythm and interval patterns are played concurrently only when they coincide at some point in the imported corpus (see Figure 3).

IMPLICATIONS

The strength of this approach is evident in that a given chord pattern is not necessary for the algorithm to create new material. It is well known that jazz musicians can improvise without a given chord structure by solely focusing on horizontal considerations. In other words, the underlying chord progression used in tonal jazz is only partly responsible for the creation of melodic material. We believe this is the first time computational modeling of musical tonal improvisation has independently applied vertical and horizontal aspects in the model. In future developments of our algorithm we aim to incorporate underlying chord structures in a way that will independently influence note choices.



Figure 1. Example of an improvisation created with the pitch-only algorithm.



Figure 2. Example of an improvisation created with the independent pitch and rhythm algorithm.



Figure 3. Example of an improvisation created with the most current algorithm in which pitches and rhythms are related.

Our model appears to support the viability of Pressing's (1988) theory in which stored fragments are reused during improvisation. It also aligns with existing motor learning research outlining how general motor programs are acquired and later reused (e.g. Shea and Wulf 2005). Language acquisition theories that emphasize statistical processes for pattern learning also may share features with the described computer model for musical improvisation (e.g. Saffran 2003). The current model may therefore illuminate domain general mechanisms related to pattern-based generative and learning processes.

Address for correspondence

Martin Norgaard, School of Music, Georgia State University, PO Box 4097, Atlanta, Georgia 30302, USA; *Email:* mnorgaard@gsu.edu

References

- Berliner P. F. (1994). *Thinking in Jazz*. Chicago: University Of Chicago Press.
- Gillick J., Keller R. M., and Tang K. (2010). Machine learning of jazz grammars. *Computer Music Journal*, 34, pp. 56-66.
- Johnson-Laird P. N. (2002). How jazz musicians improvise. *Music Perception*, 19, pp. 415-442.
- Keller R. and Morrison D. (2007). A grammatical approach to automatic improvisation. *Proceedings of the 4th Sound and Music Computing Conference* (pp. 330-337).
- Matlab (2011). Mathworks, Inc.
- Norgaard M. (2011). Descriptions of improvisational thinking by artist-level jazz musicians. *Journal of Research in Music Education*, 59, pp. 109-127.
- Norgaard M. (in press). How jazz musicians improvise: The central role of auditory and motor patterns. *Music Perception*.
- Pressing J. (1988). Improvisation: Methods and model. In J. A. Sloboda (ed.), *Generative Processes in Music* (paperback, pp. 129-178). Oxford: Oxford University Press.
- Rolland P. -Y. and Ganascia J. -G. (2000). Musical pattern extraction and similarity assessment. In E. R. Miranda (ed.), *Readings in Music and Artificial Intelligence* (pp. 115-144). Amsterdam: Harwood Academic Publishers.
- Saffran J. R. (2003). Statistical language learning: Mechanisms and constraints. *Current Directions in Psychological Science*, 12, pp. 110-115.
- Shea C. H. and Wulf G. (2005). Schema theory: A critical appraisal and reevaluation. *Journal of Motor Behavior*, 37, pp. 85-101.
- Smit C. (n.d.). *Midi Tools: Extending the Midi Toolbox in Matlab*, accessed at: http://www.ee.columbia.edu/~csmit/matlab_midi.html.