A computer-mediated Interface for Jazz Piano Comping

Rui Dias
University of Porto / INESC-Porto
School of Applied Arts, Polytechnic Institute of Castelo Branco
ruidias@ipcb.pt

Carlos Guedes
NYU Abu Dhabi
cag204@nyu.edu

Telmo Marques
Research Center for Science and Technology of the Arts (CITAR)
Portuguese Catholic University
telmomars@gmail.com

ABSTRACT

This paper presents a dynamic interface and voicing algorithm for real-time performance of jazz piano comping. Starting with a given song with a predefined harmonic progression, the algorithm calculates and maps an array of chord voicings to a virtual piano keyboard that can be played in real-time with any physical multi-touch input device like an iPad or computer keyboard. By taking care of the note selection for the voicings, the interface provides a simplified and intuitive way to play sophisticated voicings, while leaving the control over the performance aspects like timing, register, wideness and density to the user.

1. INTRODUCTION

Traditional jazz music is particularly interesting for computer algorithm design. The crossing between informal oral tradition of improvisatory nature with the catholic European western music culture since the gospels of the slave communities in North America, gave birth to a new music language from which several different styles emerged during the twentieth century. Modern jazz, namely bebop, developed by composers like Charlie Parker and Dizzy Gillespie, is musically very rich yet the formal musical structures are relatively simple [1]. Composers like George Gershwin and Cole Porter, both with classical music training, contributed for the creation of a rich repertoire of solid tonal music harmonic and melodic content, which served as the base for a common music practice in the jazz community.

Several projects in computer music research have targeted different aspects of jazz music. Computer models for jazz improvisation have been described [2], and the generation of jazz solos was particularly focused [1, 3, 4]. Models for harmony development and transformation were devised [5, 6, 7, 8] automatic accompaniment systems like the well-known Band-in-a-Box [9] and research studies by [10]. Very few strategies for devising chord voicings, however, have been approached. [11, 12, 13]. These studies focus on the harmonization and voicing calculation of a given melodic line, using a rule-based implementation of jazz chord theory. Although some of the problems and solutions have common aspects, the algorithm here described departs from a very distinct base concept and aim, which impose important differences, namely:

- it was created for a real-time interface where the user can freely play in any pitch register at any time. This means that the user has an important control on *sequenciality* [12], or horizontal voice-leading and inversion calculation between consecutive chords;
- it is planned for live jazz comping, where the melody and solo come from an external soloist, and thus there is no prediction of the notes that will be played melodically;

The present work derives from previous research on automatic music generation algorithms and music interfaces for live performance in the *Kinetic* project and being continued in the MAT project (see acknowledgments). The algorithm presented in this paper is the development over the keyboard voicing algorithm and interface developed for *Gimme Da Blues* [14], an application for iOS devices that allows the user to play trumpet and piano, while an automatic virtual bassist and drummer are generated in real-time. Other developments on this research have focused on the sequencer and harmonic content, as well as the walking bass algorithm [15].
This paper describes the new voicing algorithm for live piano comping in traditional jazz music.

2. JAZZ PIANO VOICINGS

This section will focus on the main aspects taken under consideration for the development of the present voicing algorithm.

> *"The pianist improvises a statement of the chord sequence, varying the choice of chords and their voicings"*
>
> P. N. Johnson-Laird

2.1 Chords and chord positions

Regular chords in tonal music have at least three notes, starting from the fundamental (base) note, and adding up consecutive major or minor third intervals. The basic triad (three note chord) has a 1st (fundamental), 3rd and 5th degrees, always considering the fundamental as reference. A chord can be in its root or fundamental position, when the order of the notes has the 1st degree as the lowest note. When the lowest note is a different degree, the chord is said to be in a different position, or in this case, an inversion of the initial chord.

2.2 Voicings

In Jazz, the term *voicing* refers to the way a given chord is played. The same chord can be played in innumerous different ways. For this study, the possibilities for voicing variety were traduced in four parameters or characteristics: 1) position, 2) register, 3) wideness and 4) note selection. The decision on what voicing to use for a certain chord at a given moment can depend on several aspects like musical style, personal style, voice leading, density, and it is usually also directly influenced by what is being played by the other musicians, mainly the soloist.

2.2.1 Position

The position corresponds to the order of the notes of the chord, as mentioned above, and comprehends the initial fundamental order, as well as the inversions.

The position is also commonly referred as being open or closed, depending on the interval order. For the sake of clarity, and because at the complexity level of the concept of voicing here described this designation wouldn't be so relevant as only very few voicing are closed, this aspect is included in this paper as wideness (see below).

2.2.2 Register

The register corresponds to the octaves the chord is being played on. The same voicing can sound very differently in different octaves due to the way we perceive the frequency scale. One single note, interval or chord played in a low octave sounds much denser than in a higher register. Hence, a well-balanced voicing tends to avoid small intervals in the lower octaves and use the notes distributed as uniformly as possible.

2.2.3 Wideness

By wideness we refer to how wide (open) or close the voicing is. The basic configuration of a chord, ordered by major and minor third intervals, or their inversions, as long as the chord degrees are consecutive, are closed positions. However, this can lead to some ambiguity when considering chord extensions above the octave.

For example, a possible voicing for a C major chord can have an added note D (a tension), which is the 2^{th} degree (Fig. 2a).

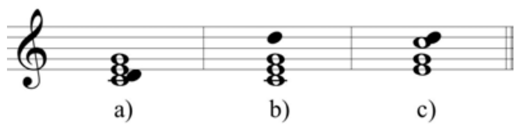

Figure 1 Chord with Added Note.

The procedure for chord construction tells us that this D is indeed an extension of the chord, in this case, the 9^{th} degree of the chord. This is because the way chords are built is by adding an extra third interval, and as such, the D is the chord extension right after the 7^{th} degree, the note B. So, if we consider the root position of the chord including the 9^{th}, and taking into account the above statement that a chord is in its closed position if the chord notes appear in its consecutive order, the chord with the D above the octave is the closed position (Fig. 1b). From a register and range perspective, however, the chord with the D inside the octave (Fig. 1a is comparatively clearly closer.

In addition, if we take the chord in the first inversion (Fig. 1c), the same high D (9^{th} degree) is now the lowest one possible, and as such, there's no other closer form than this one.

In order to avoid any ambiguity, we consider only the "wideness" of the voicing, meaning the distance from the lowest note to the highest, independently of the degree. In the example above, the voicing in Fig. 1a is a perfect 5^{th}, while the voicing in Fig. 1b is a major 9^{th}, and thus has a greater wideness factor.

Considering two-hand piano comping, voicing chords can spread for two or three octaves, and even more as hands move apart.

2.2.4 Note selection

The selection of notes to include in a voicing is by far the more complex parameter and inherently subjective. In jazz music, more than in any other musical style, common practice suggests a complete freedom where the note selection criteria is dependent basically on musical intuition. Starting with a structural chord, this matter concerns the addition, subtraction, and/or alteration of notes of the base chord. In fact, this intuitive performance approach brought a plethora of analytical procedures and normative rules that are worldwide studied in Jazz Universities, Schools, and Academies.

According to several authors, the II-V-I harmonic progression is by far the most used in jazz performance (especially jazz standards) and the most studied in jazz theory and research [16, 17, 18, 19, 20, 21]. Because there are thousands of possible re-harmonizations of that harmonic pattern, the scope of this paper will focus on those whose characteristics were crystalized by influent performers in the Jazz mean. Every theory is fruit of an intuitive convolution before it reaches his postulate. This is the case of the following examples: what is now usually called the *Drop 2* voicing [18] (Figure 2), started with a necessity for certain jazz piano players – like Bill Evans – in order to get more clarity in the soprano voice. Starting with a closed chord (Dm7 with a 9^{th}) the second note counting from the soprano fall an octave down – in this case, the C.

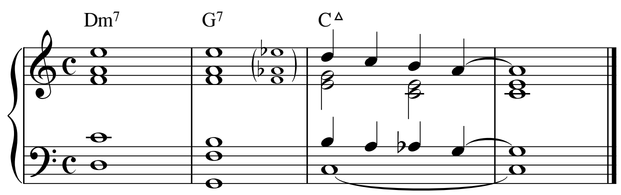

Figure 2 Bill Evans *Drop 2* voicing style.

Figure 3 presents the traditional *Four-way Close* [19] with additional extensions. This example is also constructed with an *Upper-Structure Triads* voicing, getting a powerful sound containing a high level of resonance [16, 17, 19].

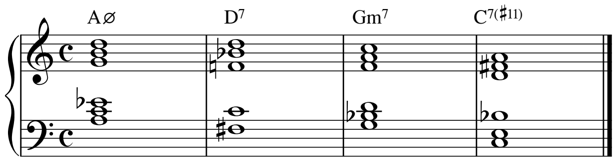

Figure 3 Bill Evans *Four-way Close* voicing.

McCoy Tyner is usually pointed out as the creator of another kind of comping technique and harmonization known as *Voicings in Fourths* (Figure 4). In this case the note chords tend to be distributed vertically in intervals of fourths. The fourths provide an ambiguous sound close to the sound of suspended chords, but without loosing its harmonic functionality.

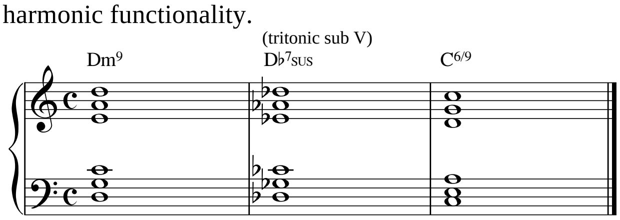

Figure 4 McCoy Tyner *voicings in Fourths*.

Some modern jazz pianists employ the so-called *Altered Chords* [16, 17, 20]. Classical theory defines it as a chord with one or more diatonic notes replaced by a chromatic neighbor; in jazz, it is a chord that borrows the diatonic notes from its parallel minor mode. Figure 5 reflects this attribute – the sound texture is rich and dense.

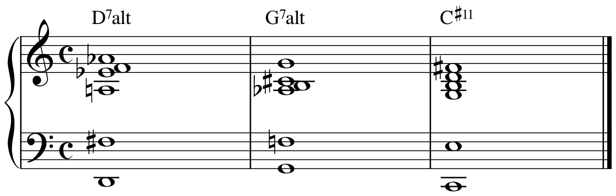

Figure 5 Keith Jarrett *Altered Chords* Harmonization.

The search for uniqueness in musical discourse is a constant struggle even in the jazz culture. Musicians as Herbie Hancock became famous by the use of suspended chords rather than dominant chords. As can be seen in Figure 6, having no leading tone, dominant chords escape the attraction to resolve, bringing a modal flavor to the overall sound.

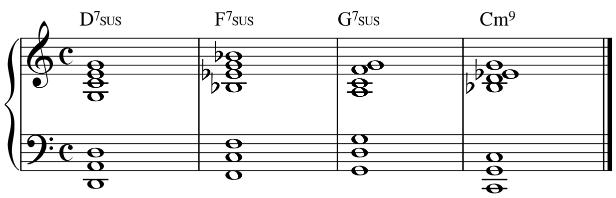

Figure 6 Herbie Hancock *Maiden Voyage* Harmony.

Finally, Figure 7 presents another way to create suspended chords. On the one hand, suspended chords are made-up moving the third over the fourth (usually named *sus4*), on the other hand, the original third is placed in higher register and named as 10th. In fact this is a kind of special triadic chord with a fourth added.

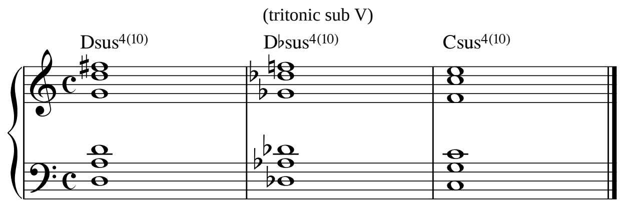

Figure 7 Brad Mehldau *Tenths* Harmony.

3. A COMPUTER-MEDIATED INTERFACE

As mentioned before, the present work derives directly from previous research, namely the creation of the *Gimme Da Blues* app [14]. The base concepts and directives were kept unaltered, namely: 1) the algorithm has to work in real time; 2) the player has an indirect, meta-control over the musical events; 3) although indirect, the control has to provide the essential feeling of performance in a jazz/blues improvisation context.

The interface is planned for iPad/iPhone/iPod, but any other multi-touch device can be used, as long as it allows sending control messages to the computer. A version to use the computer keyboard was also created. Other devices with pads or Ableton Live clip launcher buttons like the Ableton Push can be used as well.

The software was developed in the Max programming environment [22], using both regular Max objects as well as javascript code (also inside Max). The iOS devices use *Fantastick* [23], a great application that sends multi-touch data by wi-fi. Using the UDP network protocol, the data is received inside Max.

The piano interface and algorithm were developed having in mind the simulation of the experience of playing jazz piano with a fair degree of sophistication in the resulting sound. Given that on a conventional keyboard, with direct control over each individual key, and the four parameters described earlier, the number of possible voicings for a given chord is far too great to attain with a simple-to-use approach and with the limitations of the physical device, the voicing algorithm developed was a compromise between complexity and usability.

Very early in the research, an assumption was made by analyzing several aspects of the jazz piano practice. The assumption that we can to some measure separate the decision over the notes being played from the decision of the actual action of playing, in a rhythmic, percussive sense. We believe that an important part of the performative aspect of the live improvisation experience in jazz - and most likely in every other improvisational music - can be thought of as rhythmical, in the sense that the action of deciding WHEN to play a note or chord in any harmonic or melodic instrument is not so different from the same action by the drummer or percussionist. Of course there are several other decisions implied, but they are mostly idiomatic and related to the role assigned to each of them. The decision over WHAT notes to play is the result of the learning process about harmony and tonal music grammar, including chord progressions, scales, and melodic patterns.

Simultaneously, some decisions about the notes to play can be thought of in a sort of meta-level control, independently of the actual note selection. The register and wideness parameters mentioned above, that determine WHERE the notes are being played, can be decided in order to complement the soloist, by playing in opposite registers, or conversely to match the soloist by playing in coincident registers.

This "meta-control" can be observed also by analyzing the overall melodic contours that can happen in groups of a few beats or bars. These contours delineate phrases and behaviors that can greatly contribute to the overall comping quality, and also characterize some of the personal style of the player, and can be considered, analyzed and controlled independently of which exact notes are being played. In other words, the present interface and algorithm provides the WHAT, leaving the WHEN, WHERE and the WHY to the user.

3.1 Interface

The user interface and interaction modes are very similar to the one developed for the *Gimme Da Blues* app, since it proved quite successful in providing a natural and intuitive way to experience most of the afore mentioned rhythmical feeling and meta-control over the note and chord events.

It comprises a virtual keyboard with a variable number of virtual keys (according to the device being used) that allow to play with one or two fingers, each finger corresponding to one hand of the player. With every chord change, the voicings are calculated and dynamically mapped to the appropriate keys, so that every key plays a useful and correct note or voicing. With the iPad multi-touch screen, a total of 16 keys were used (see Figure 8). Like in a conventional piano, the lower pitches are on the left side and the higher pitches on the right side.

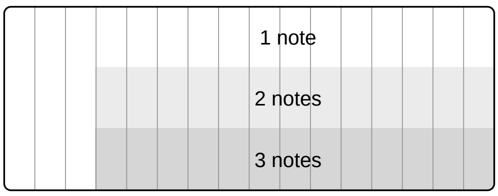

Figure 8 Virtual keyboard on the iPad screen.

While the keyboard in *Gimme Da Blues* had two rows across the entire width, the new version presents three, starting on the fourth key. The different rows correspond to the number of notes to include in a chord. A finger in the lower (darker) row will trigger a three-note voicing, the middle row will trigger two note voicings, and the top row (represented in white) plays only one note. The first three keys on the left side are reserved respectively to the root, fifth and octave of the current chord being played, and triggers one single note.

This configuration allows for a fast, intuitive and versatile playing technique, by combining two fingers in different horizontal and vertical positions.

The top, single note row, allows the user to play simple melodic phrases that can complement the chords, dialogue with the soloist or other accompaniment instruments, or even be used to play the main or secondary melody, giving a whole new range of possibilities.
The selection and mapping process of the assigned notes to the keys will be explained below.

3.2 Sequencer

Before any processing of the chord voicing can be done, a specially developed sequencer module reads a song and style template, also specially created for this system, and parses all the contents to their appropriate destinations. The song tempo, time signature and number of bars are used to define the global transport mechanism that, together with the harmonic structure, also described in the song template, will create the timeline that will be followed during runtime.

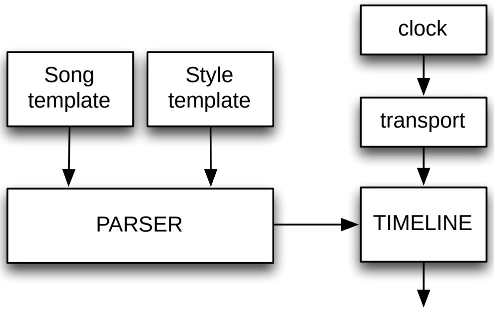

Figure 9 Sequencer structure

During runtime, the chord information will be sent to the piano algorithm, that will instantly calculate an array of chord voicings and map them to the virtual keys.

3.3 Voicing algorithm

The voicing algorithm comprehends three main sections: the Voicing Map Calculator, the User Input Mapping, and the Player (see Figure 10). Every chord change in the song will be read, formatted and sent in real-time by the sequencer. The Voicing Map Calculator will create a list of voicings, with one voicing for each virtual key that will be available for the Chord Player section. When the user touches a key, the key number and touch coordinates will be sent to the Player, that will read the corresponding voicing from the voicing map. The final sounding voicing will be a subset of one, two or three notes that will be different depending on the key number and vertical touch position that was used.

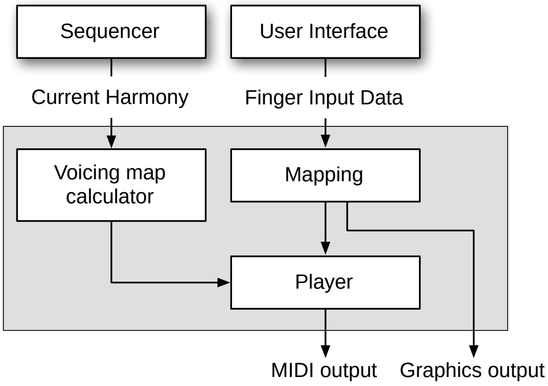

Figure 10 Voicing algorithm global structure

3.3.1 Chord information input

The algorithm takes as real time input the chord data of the current harmony arriving from the sequencer timeline. The real time data includes the chord name and type, as well as an associated scale. For example, “C 7 mixo” specifies the C chord, of type Dominant 7th, and a scale, in this case a mixolidian mode. The chord fundamental (C) will be used as Pitch Class Sets, with a transposition factor, which for C is zero. The chord type correspondence is described in the style template, where the “7” will correspond to a predefined Dominant 7th voicing.

3.3.2 Scales

The scales can be defined in the song template, and correspond to the notes that will be assigned to the virtual keys. The term scale is used here to refer to any combination of notes that will be available for the user to play, and doesn’t necessarily correspond to a conventional scale. It can be a conventional major or minor scale but it can also be a mode or a subset of scale or mode degrees.
In order to have a better keyboard range and to avoid possible incompatibilities between the scale notes and chord notes, better results are obtained using non-complete subsets. The “mixo” (mixolidian mode) in the example above can for example be specified as the subset [0 2 4 7 9 10], which corresponds to the mixolidian mode without the 4th degree.

3.3.3 Voicing format

The voicings are described as a list of Pitch Class sets, according to the chord type specification in the harmonic structure. This list defines the notes that can be included in the voicing, but also each one’s index number, by ordering them from left to right according to their priority. A 7th chord for example, can be defined as “10 4 2 9 7 0”. Pitch Class “10” (Bb) corresponds to the minor 7th of the Dominant 7th chord, “4” (E) to the major 3rd, “2” (D) to the major 2nd and so forth.

3.3.4 Voicing calculation

The Voicing Map Calculator receives the voicing for the current harmony and calculates all the voicings for the entire keyboard. This received voicing is not, however, the final voicing that will be triggered. Instead, this list is a sort of map or key to the construction of the final voicings that will be played. The starting point for the calculation of the voicings is the scale. The scale is mapped to the keyboard. Then, for each key a voicing is calculated by finding the defined degrees below the scale note.

Having the example mentioned before: “C 7 mixo”, with the scale “mixo” “0, 2, 4, 7, 9, 10” and the voicing “key” “10, 4, 2, 9, 7, 0”, the calculator algorithm will map the scale to the keys:

Figure 11 Scale Mapped to the Virtual Keys.

and then calculate the rest of the notes for each voicing, using the order defined in the key. For this example, the resulting voicings would be:

Figure 12 Scale and Voicings for Each Key.

The black notes are the notes of the scale. The white notes are the voicing notes automatically calculated.
As said before, the first three keys have only the fundamental, 5th and the octave. From the fourth key on, the algorithm will calculate the voicing by searching for the degrees defined in the “voicing key”, by their priority.
In the fourth key, where the scale note is C3 (middle C), the voicing notes are a Bb2 and a E2. These correspond respectively to the first two degrees in the voicing key “10, 4”. The same happens in the next voicing, D3. In the sixth however, because the scale note is already an E, the algorithm will advance one index and look for a “2”, which is the next degree in the list.
The algorithm allows also to change the starting index of the lookup list, in any scale degree. In the example voicings above, a change is visible from the scale note C4 on, where the voicing notes are not the same (E, Bb) but instead are “D, A”, which correspond to the “2, 9” defined in the list. This is intended in order to avoid repetition and to create more variety.
Using two hands, the user can play rather sophisticated voicings (Figure 13).

Figure 13 Four Two-Hand Voicing Examples.

3.3.5 Player

The "Player" receives the voicing map from the voicing map calculator and manages the input data arriving from the user interface, to play the corresponding events.

Using the vertical position of the fingers, it uses one, two or all three notes of the voicing (zones in Figure 8), following the same order of the "voicing key". When playing only one note, it will always be the scale note, followed by the second degree and the third.

4. CONCLUSIONS

The algorithm described was the result of a number of different approaches to create interesting sounding piano voicings and is a part of an ongoing research on automatic generation and new interface design for computer-mediated music. By considering the highest note as the starting point for the voicing calculation provides an interesting and effective way to both mimic a part of the physical hand placement and mental process of the jazz pianist, and creates a fast responsive way to provide simultaneous control over the register, wideness and rhythmical aspect of piano comping. The possibility of playing single notes using the top row of the keyboard boost the usability and quality of the comping by allowing the player to use passing notes or small melodic phrases to respond, incite or provoke the other players. This approach also seems very promising for the current ongoing development of an automatic, autonomous player, as it creates fairly complex results using simple control parameters.

Acknowledgments

The Media Arts and Technologies project (MAT), NORTE-07-0124-FEDER-000061, is financed by the North Portugal Regional Operational Programme (ON.2 – O Novo Norte), under the National Strategic Reference Framework (NSRF), through the European Regional Development Fund (ERDF), and by national funds, through the Portuguese funding agency, Fundação para a Ciência e a Tecnologia (FCT).

This study was partially funded by NYU Abu Dhabi.

GimmeDaBlues was done as part of the project "Kinetic controller, driven, adaptive and dynamic music composition systems" funded by the ERDF through the Program COMPETE, by the Portuguese Foundation for Science and Technology (FCT), Project ref. FCOMP-01-0124-FEDER-011414, UTAustin/CD/0052/2008, and partly supported by the European Commission, FP7 (Seventh Framework Programme), ICT- 2011.1.5 Networked Media and Search Systems, grant agreement No 287711 (MIReS).

5. REFERENCES

[1] Pennycook, B., Stammen, D., & Reynolds, D. (1993). Toward a Computer Model of a Jazz Improvisor. International Computer Music Conference.

[2] Johnson-Laird, P. (1991). Jazz improvisation: A theory at the computational level. *Representing Musical Structure, London*.

[3] Biles, J. Al. (1994). GenJam: A genetic algorithm for generating jazz solos. In Proceedings of the International Computer Music Conference (p. 131).

[4] Papadopoulos, G., & Wiggins, G. (1998). A Genetic Algorithm for the Generation of Jazz Melodies. *Knowledge Creation Diffusion Utilization*, 10.

[5] Steedman, M. (1984). A Generative Grammar for Jazz Chord Sequences. *Music Perception*.

[6] Chemillier, M. (2004). Steedman's grammar for jazz chord sequences.

[7] Pachet, F. (1999). Surprising Harmonies. *International Journal on Computing Anticipatory Systems*, 1–20.

[8] Bäckman, K. (2008). Evolutionary Jazz Harmony: A New Jazz Harmony System. In *BIOMA 2008 conference in Ljubljana*.

[9] Band-in-a-box (1995). Canada: PG Music Inc.

[10] Hidaka, I., Goto, M., & Muraoka, Y. (1995). An Automatic Jazz Accompaniment System Reacting to Solo. *In Proc. of ICMC*.

[11] Emura, N., Miura, M., & Yanagida, M. (2006). A System yielding Jazz-style arrangement for a given melody. In *Western Pacific Acoustics Conference* (pp. 1–8). Seoul.

[12] T. Kitahara, M. Katsura, H. Katayose, and N. Nagata. "Computational Model for Automatic Chord Voicing based on Bayesian Network", in *Proc. Int. Conf. Music Perception and Cognition (ICMPC 2008)*, Sapporo 2008.

[13] J. Watanabe et al. "A system generating jazz-style chord sequences for solo piano", in *Proc. Int. Conf. Music Perception and Cognition (ICMPC 2008)*, Sapporo 2008.

[14] R. Dias, T. Marques, G. Sioros and C. Guedes, "GimmeDaBlues: an intelligent Jazz/Blues player and comping generator for iOS devices". in Proc. Conf. Computer Music and Music Retrieval (CMMR 2012). London 2012.

[15] R. Dias, C. Guedes. "A contour-based walking bass generator", in *Proc. Int. Conf. Sound and Music Computing* (SMC 2013), Stockholm, 2013.

[16] S. Davis, *Jazz Piano Comping: Harmonies, Voicings, and Grooves*. Boston: Berklee Press, 2012.

[17] M. Levine, *The Jazz Piano Book*. Petaluma: Sher Music, 1989.

[18] M. Levine, *The Drop 2 Book*. Petaluma: Sher Music, 2006.

[19] T. Pease and K. Pullig, *Modern Jazz Voicings: Arranging for Small and Medium Ensembles*. Boston: Berklee Press, 2001.

[20] R. Felts, *Reharmonization Techniques*. Boston: Berklee Press, 2002.

[21] http://www.freejazzlessons.com (consulted on March 31st 2014).

[22] http://cycling74.com (consulted on March 31st 2014)

[23] http://pinktwins.com/fantastick/ (consulted on March 31st 2014)