The Spiral Array: An Algorithm For Determining Key Boundaries

Elaine Chew

University of Southern California, Integrated Media Systems Center,
Daniel J. Epstein Department of Industrial and Systems Engineering,
3715 McClintock Avenue GER240 MC:0193, Los Angeles CA 90089, USA,
echew@usc.edu; http://www-rcf.usc.edu/~echew.

Abstract. Computer models for determining key boundaries are important tools for computer analysis of music, computational modeling of music cognition, content-based categorization and retrieval of music information and automatic generating of expressive performance. This paper proposes a Boundary Search Algorithm (BSA) for determining points of modulation in a piece of music using a geometric model for tonality called the Spiral Array. For a given number of key changes, the computational complexity of the algorithm is polynomial in the number of pitch events. We present and discuss computational results for two selections from J.S. Bach's "A Little Notebook for Anna Magdalena". Comparisons between the choices of an expert listener and the algorithm indicates that in human cognition, a dynamic interplay exists between memory and present knowledge, thus maximizing the opportunity for the information to coalesce into meaningful patterns.

1 Introduction

This paper proposes a method for segmenting a piece of music into its respective key areas using the Spiral Array model. The *key* of a musical piece imparts information about the principal pitch set of the composition. More importantly, it identifies the most stable pitch, the *tonic*, and its hierarchical relations to all other pitches in that pitch set. The *tonic* is sometimes referred to as the *tonal center*, or simply as "doh". Once the key is established, ensuing notes are heard in reference to the tonic. This reference point allows the listener to navigate through the soundscape generated by the piece. The computational modeling of the human ability to determine the key of a melody is a well-studied problem and numerous methods have been proposed for mimicking this cognitive ability. See, for example, [5], [13], [16] and [21]; and, [1], [18] and [22] in [10].

Modulation or the change of key is a common compositional device. When the key changes, each pitch assumes a different role in relation to the new tonal center. This shift in paradigm transforms the tonal context in which the notes are heard. One way in which the human mind organizes music information into coherent chunks is by way of its key contexts. Conversely, key changes often

occur at structural boundaries. The skill with which a composer moves persuasively or abruptly from one context to another constitutes an important feature of her style. A performer may choose to highlight unusual modulations through dynamic or rate changes. These are but a few examples to illustrate the importance of developing robust and computationally viable methods for determining key boundaries. Computer modeling of the cognitive task of determining key boundaries is potentially an exponentially hard problem, and very little work has been done in this area.

By using the Spiral Array and by incorporating music knowledge in modeling the problem, this paper proposes a computationally efficient, knowledge-based way to determine points of modulation in a piece of music. The Spiral Array is a spatial arrangement of pitch classes, chords and keys proposed by Chew [6] as a geometric model for *tonality*, the system of musical principles underlying tonal music. According to Bamberger [2], "*Tonality and its internal logic frame the coherence among pitch relations in the music with which [we] are most familiar.*" The perception of key results from this system of hierarchical relations amongst pitches.

The Spiral Array is derived from the Harmonic Network (described in various contexts in [7], [8], [14], [15] and [16]), an arrangement of pitch classes on a lattice. Chord pitches and pitches in a key map to compact sets on this grid. The Spiral Array preserves the clustering of tonally important pitch sets. In addition, its 3D configuration allows higher level musical entities to be defined and embedded in the interior of the structure. A generating principle in the Spiral Array is that higher level musical entities can be represented unambiguously as convex combinations of their lower level components. Spatial coordinates representing each key were generated from weighted averages of its I, V and IV chord representations, which were in turn generated from each chord's root, fifth and third pitch representations.

In the Spiral Array, musical information is condensed and summarized by 3D coordinates that represent the center of effect (c.e.) of the pitches. This idea was extended in the CEG algorithm [5] to generate c.e.'s that mapped the establishing of key over time to spatial trajectories. The distance from the moving c.e. to the key representations serves as a likelihood indicator that the piece is in that key. The distance-minimizing key is elected as the most likely solution. It was shown that the CEG algorithm offers a competitive and viable means to identify the key of a melody.

In this paper, we show that determining the points of modulation can be re-cast as a problem of finding the distance-minimizing boundaries for the different key contexts using the Spiral Array framework. For this reason, we call the algorithm the Boundary Search Algorithm (BSA). A straightforward implementation of the algorithm yields an $O(n^m)$ performance time, where n is the number of event onsets and $m(\geq 2)$ is the number of boundaries. With more information, a listener can assess the key identity with more confidence, and correspondingly, the algorithm works best when n is large and m is small.

Given a piece of music, exhaustively enumerating all segmentation possibilities based on key boundaries is a formidable and exponential task. We show that by integrating musical knowledge, one can drastically reduce the search space and propose a tractable polynomial-time solution to the problem of determining key boundaries for the two Bach examples addressed in this paper. Realistically speaking, the number of key changes, m, is typically a small number in relation to the number of event onsets, n. The two Bach examples each contain one point of departure and one return to a primary key area, that is to say, $m = 2$. The problem is further constrained by the fact that the first and third (last) key areas must be identical.

Apart from computational tractability, the benefits of a knowledge-based approach to determining key boundaries also include more accurate models for computer analysis and tracking of music, a better understanding of the perception of musical structure, more realistic computer-generated expressive performances and more robust algorithms for content-based music information retrieval.

Comparisons between a human expert's perception of key boundaries and the algorithm's choices indicate that human processing of music information, a dynamic interplay exists between past (memory) and present knowledge, thus maximizing the opportunity for the information to coalesce into meaningful patterns. Also, the human's perception of key boundaries is influenced by phrase structure.

2 A Mathematical Model for Tonality

Any computer model for processing music information must begin with a representation that converts content information to numerical data for analysis. The representation we use is the Spiral Array model first introduced in [6]. The Spiral Array grew out of a 3D configuration of the Harmonic Network to eliminate periodicity and redundancy (see Figure 1). The use of this geometric model to determine the key of a musical passage with a likelihood score based on spatial proximity is outlined in [5]. We briefly describe how key representations are generated in this structure.

2.1 Placement of Pitch Classes, Chords and Keys

In the Spiral Array, pitch classes are represented by spatial coordinates along a spiral. Each pitch class is indexed by its number of perfect fifths from an arbitrarily chosen reference pitch, C (set at position [0,1,0]). An increment in the index results in a quarter turn along the spiral. Four quarter turns places pitch classes related by a major third interval in vertical alignment with each other. Strict enharmonic equivalence would require that the spiral be turned into a toroid. The spiral form is assumed so as to preserve the symmetry in the distances among pitch entities.

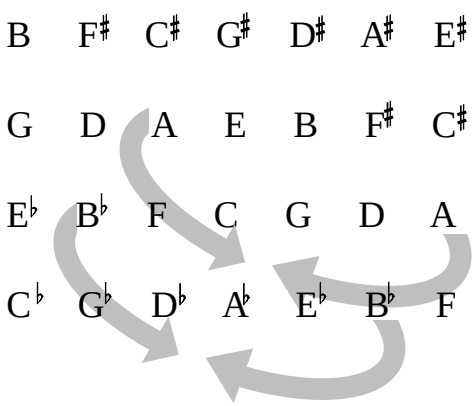

Fig. 1. Rolling up the Harmonic Network to eliminate redundancy.

Definition 1. *The radius of the cylinder, r, and the height gain per quarter turn, h, uniquely define the position of a pitch representation, which can be described as:*

$$\mathbf{P}(\mathrm{k}) \stackrel{\mathrm{def}}{=} \begin{bmatrix} x_{\mathrm{k}} \\ y_{\mathrm{k}} \\ z_{\mathrm{k}} \end{bmatrix} = \begin{bmatrix} \mathrm{r}\sin\frac{\mathrm{k}\pi}{2} \\ \mathrm{r}\cos\frac{\mathrm{k}\pi}{2} \\ \mathrm{kh} \end{bmatrix}.$$

Chord representations are defined uniquely as convex combinations of their component pitches. Triads map to non-overlapping triangles on the Spiral Array. Hence, each triangle can be uniquely represented by the convex combination of its component pitches. The weights are constrained to be monotonically decreasing from the root, to the fifth, to the third to mirror the relative important of each pitch to the chord.

Definition 2. *The representation for a major triad is generated by the convex combination of its root, fifth and third pitch positions:*

$$\mathbf{C}_{\mathrm{M}}(\mathrm{k}) \stackrel{\mathrm{def}}{=} w_1 \cdot \mathbf{P}(\mathrm{k}) + w_2 \cdot \mathbf{P}(\mathrm{k}+1) + w_3 \cdot \mathbf{P}(\mathrm{k}+4),$$
$$\textit{where} \quad w_1 \geq w_2 \geq w_3 > 0 \textit{ and } \sum_{i=1}^{3} w_i = 1.$$

The minor triad is generated by a similar combination,

$$\mathbf{C}_{\mathrm{m}}(\mathrm{k}) \stackrel{\mathrm{def}}{=} u_1 \cdot \mathbf{P}(\mathrm{k}) + u_2 \cdot \mathbf{P}(\mathrm{k}+1) + u_3 \cdot \mathbf{P}(\mathrm{k}-3),$$
$$\textit{where} \quad u_1 \geq u_2 \geq u_3 > 0 \textit{ and } \sum_{i=1}^{3} u_i = 1.$$

Major key representations are generated as convex combinations of their I, V and IV chords. Figure 2 shows an example of a major key representation. The weights are constrained to be monotonically decreasing from I to V to IV.

Definition 3. *A major key is represented by a convex combination of its tonic, dominant and subdominant chords. The weights are restricted to be monotonic and non-increasing.*

$$\mathbf{T}_{\mathrm{M}}(\mathrm{k}) \stackrel{\mathrm{def}}{=} \omega_1 \cdot \mathbf{C}_{\mathrm{M}}(\mathrm{k}) + \omega_2 \cdot \mathbf{C}_{\mathrm{M}}(\mathrm{k}+1) + \omega_3 \cdot \mathbf{C}_{\mathrm{M}}(\mathrm{k}-1),$$
$$\textit{where} \quad \omega_1 \geq \omega_2 \geq \omega_3 > 0 \ \textit{and} \ \sum_{i=1}^{3} \omega_i = 1.$$

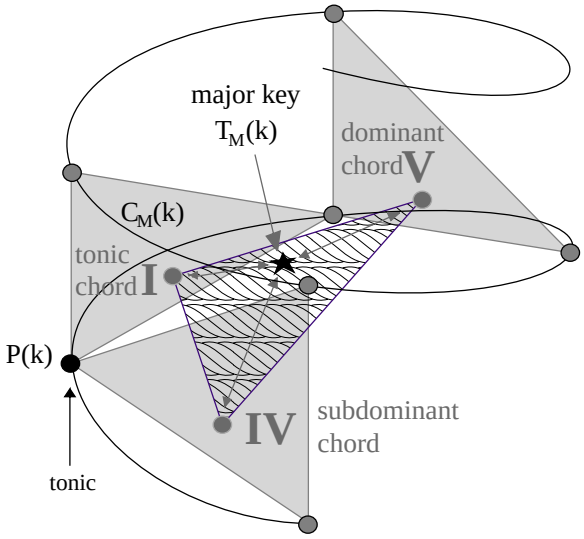

Fig. 2. Geometric representation of a major key, a composite of its tonic (I), dominant (V) and subdominant (IV) chords.

The minor key representation is a combination of the i, (V/v) and (iv/IV) chords. The additional parameters α and β model the relative importance and usage of V versus v and iv versus IV chords respectively in the minor key.

Definition 4. *The minor key representation is generated by a convex combination of its tonic, dominant (major or minor) and subdominant (major or minor) chords as follows:*

$$\mathbf{T}_{\mathrm{m}}(\mathrm{k}) \stackrel{\mathrm{def}}{=} \upsilon_1 \cdot \mathbf{C}_{\mathrm{m}}(\mathrm{k})$$

$$
\begin{aligned}
& +v_2 \cdot [\alpha \cdot \mathbf{C}_{\mathrm{M}}(\mathrm{k}+1) + (1-\alpha) \cdot \mathbf{C}_{\mathrm{m}}(\mathrm{k}+1)] \\
& +v_3 \cdot [\beta \cdot \mathbf{C}_{\mathrm{m}}(\mathrm{k}-1) + (1-\beta) \cdot \mathbf{C}_{\mathrm{M}}(\mathrm{k}-1)], \\
\textit{where} \quad & v_1 \geq v_2 \geq v_3 > 0 \ \textit{and} \ v_1 + v_2 + v_3 = 1, \\
\textit{and} \quad & 0 \geq \alpha \geq 1, 0 \geq \beta \geq 1.
\end{aligned}
$$

See Figure 3 for the spatial representation of a minor key.

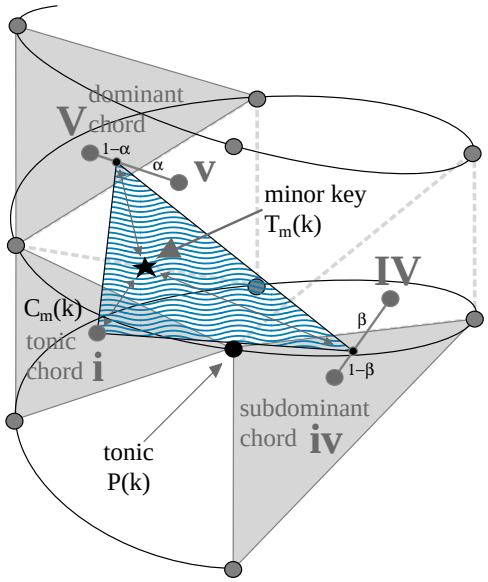

Fig. 3. Geometric representation of a minor key, a composite of its tonic (i), dominants (V/v) and subdominant (iv/IV) chords.

For the two examples in this paper, the key representations in the Spiral Array were generated by setting the weights (w, u, ω, v) to be the same and equal to $[\,0.516, 0.315, 0.168\,]$, h=$\sqrt{2/15}$ (r=1). α is set to 1 and β to 0. These weights satisfy perceived interval relations, such as, pitches related by a distance of a perfect fifths being closer than those a major thirds apart, and so on. In addition, these assignments also satisfy two other conditions: two pitches an interval of a half-step apart generates a center that is closest to the key of the upper pitch; and, the coordinates of a pitch class is closest to the key representation of the same name.

The Spiral Array turns the original 2D lattice of pitch classes into a 3D configuration that eliminates periodicity. By moving to a higher dimensional space, the Spiral Array allows the definition of higher-level musical entities as

coordinates in the same space as the pitch classes, blending both discrete and continuous space.

3 An Algorithm For Finding Key-Change Boundaries

The idea behind the Boundary Search Algorithm (BSA) is a simple one. The BSA collapses each set of pitch and duration information down to a spatial point, a center of effect (c.e.). This point is compared to the key representations in the Spiral Array Model. The better the pitch collection adheres to the pitch classes of the key in the proportions determined by their tonal relations, the closer the c.e. will be to the key representation.

When two or more key areas exist in a piece, the key areas are delineated by modulation boundaries. The problem of finding the point(s) of modulation can then be posed as a problem of determining the placement of boundaries so as to minimize the distances between the respective c.e.'s and their closest keys. This section presents a formal description of the BSA algorithm. Section 3.1 describes how music information is condensed to a spatial coordinate for comparison to a key object in the Spiral Array. Section 3.2 gives the problem statement the solution to which will determine the optimal choice of modulation boundaries.

3.1 The Center Of Effect And Distance To Key

A musical passage consists of a collection of pitch events, each comprising pitch and duration information. If $\mathbf{p}_i$ represents the position of the i-th pitch class in the Spiral Array, and d_i represents its duration, the collection of N notes can be written as $\{(\mathbf{p}_i, d_i) : i = 1 \ldots N\}$. The center of effect of this pitch collection is defined as:

$$\mathbf{c} \stackrel{\text{def}}{=} \sum_{i=1}^{N} \frac{d_i}{D} \cdot \mathbf{p}_i \qquad \text{where } D = \sum_{i=1}^{N} d_i$$

If $\mathbf{T}$ is a finite collection of keys represented in the Spiral Array, both major and minor, that is to say:

$$\mathbf{T} = \{\mathbf{T}_{\mathrm{m}}(\mathrm{k}) \forall \mathrm{k}\} \cup \{\mathbf{T}_{\mathrm{M}}(\mathrm{k}) \forall \mathrm{k}\},$$

then, the most likely key of the passage is given by:

$$\arg \min_{T \in \mathbf{T}} \|\mathbf{c} - T\|,$$

which can be readily implemented using a nearest neighbor search. The likelihood that the pitch collection is in any given key is indicated by its proximity to that key. The minimizing distance, also a likelihood indicator, is given by:

$$d^{\min} = \min_{T \in \mathbf{T}} \|\mathbf{c} - T\|.$$

3.2 Problem Statement and Algorithm

Suppose that m boundaries have been chosen as given by (B_1, ..., B_m). Let time 0 be B_0, and the end of the passage be $\mathrm{B}_{m+1} = \mathrm{L}$. The note material between any two boundaries (B_i, B_{i+1}) generates a c.e. given by $\mathbf{c}_{(\mathrm{B}_i,\mathrm{B}_{i+1})}$. The most likely key has a distance $d^{\min}_{(\mathrm{B}_i,\mathrm{B}_{i+1})}$ from this c.e.

Our hypothesis is that the best boundary candidates will segment the piece in such a way as to minimize the distances $d^{\min}_{(\mathrm{B}_i,\mathrm{B}_{i+1})}, i = 0, \ldots, m,$. Hence, The goal is to find the boundaries (B_1, ..., B_m) that minimize the sum of these distances. Let n_i be the number of pitch events from the beginning up to the boundary B_i, with the appropriate boundary value of $n_0 = 0$. The optimal boundaries, (B^*_1, ..., B^*_m), are then given by the solution to the following system:

$$
\begin{aligned}
\min \quad & \sum_{i=0}^{m} d^{\min}_{(\mathrm{B}_i,\mathrm{B}_{i+1})} \\
\text{s.t.} \quad & d^{\min}_{(\mathrm{B}_i,\mathrm{B}_{i+1})} = \min_{T \in \mathbf{T}} \|\mathbf{c}_{(\mathrm{B}_i,\mathrm{B}_{i+1})} - T\|, \qquad i = 0, \ldots, m \\
& \mathbf{c}_{(\mathrm{B}_i,\mathrm{B}_{i+1})} = \sum_{j=n_i+1}^{n_{i+1}} \frac{d_j}{D_i} \cdot \mathbf{p}_j \text{ where } D_i = \sum_{k=n_i+1}^{n_{i+1}} d_k, \qquad i = 0, \ldots, m \\
\text{and} \quad & \mathrm{B}_i < \mathrm{B}_{i+1}, \qquad i = 0, \ldots, m
\end{aligned}
$$

A naive and viable approach to the problem is to enumerates all possible sets of boundaries and evaluates each objective function value numerically in order to find the objective-minimizing boundaries. Assuming that the set of keys that are of interest $\mathbf{T}$ is finite, and that any piece of music is of finite length with n event onsets, the computational complexity of this approach is $O(n^m)$.

The practical application of this method typically yields far better performance than that implied by the $O(n^m)$ computational time. The number of key changes and associated boundaries is typically small in relation to the number of event onsets ($m << n$). In smaller-scale compositions such as the two Bach examples from "A Little Notebook for Anna Magdalena" discussed in the next section, the pieces can typically be segmented into three key areas.

Using music knowledge and a little common sense, we can add a few constraints to further reduce the search space. Adjacent key areas should be distinct from each other, that is to say, $T_i \neq T_{i+1} (i = 1 \ldots m)$. If the passage to be analyzed is a complete piece, the first and last key areas should be constrained to be the same, with the corresponding constraint being $T_1 = T_{m+1}$. For example, in pieces with three key areas, the first and last segments are in the same key as that of the composition, while the middle portion is in a different key.

4 Two Examples

We now proceed to apply the solution framework to two examples from J. S. Bach's "A Little Notebook for Anna Magdalena". The two examples chosen

each contain one departure and one return to the primary key area. The task at hand is to find the two boundaries, B_1 and B_2 separating the primary and the secondary key areas.

In these two problem instances, the three segments are required to satisfy the constraint that parts 1 and 3 must be in the same key ($T_1 = T_3$), and that this key must be distinct from that of part 2 ($T_1 \neq T_2$). These additional constraints greatly reduce the solution space. The reduced and modified problem instance is as follows:

$$
\begin{aligned}
\min \quad & d^{\min}_{(B_0,B_1)} + d^{\min}_{(B_1,B_2)} + d^{\min}_{(B_2,B_3)} \\
\text{s.t.} \quad & d^{\min}_{(B_i,B_{i+1})} = \|\mathbf{c}_{(B_i,B_{i+1})} - T_i\|, \qquad i = 0,1,2 \\
& T_i = \arg\min_{T \in \mathbf{T}} \|\mathbf{c}_{(B_i,B_{i+1})} - T\|, \qquad i = 0,1,2 \\
& \mathbf{c}_{(B_i,B_{i+1})} = \sum_{j=n_i+1}^{n_{i+1}} \frac{d_j}{D_i} \cdot \mathbf{p}_j \text{ where } D_i = \sum_{k=n_i+1}^{n_{i+1}} d_k, \qquad i = 0,1,2 \\
& T_1 = T_3 \neq T_2 \\
\text{and} \quad & 0 = B_0 < B_1 < B_2 < B_3 = L
\end{aligned}
$$

In both instances, the BSA method identified the correct keys, $T_1 (= T_3)$ and $T_2 (\neq T_1)$, and determined the boundaries B_1 and B_2. These boundaries are compared to a human expert's choices. The BSA's boundaries (A1,A2) and the expert's choices (EC1,EC2) are close but not identical in both cases. Sections 4.1 and 4.2 compare and discuss the rationale behind, and the relative merits of, these solutions.

4.1 Example 1: *Minuet in G* by Bach

The first example is Bach's *Minuet in G* from "A Little Notebook for Anna Magdalena". The Minuet is in the key of G, with a middle section in D major. Figure 4 shows a segment of the piece that contains the transitions away from and back to the intended key G. On the figure, are markings denoting a human expert's choice of boundaries (indicated by EC1 and EC2) between the keys of G and D. Also marked, are the BSA's choices for these same boundaries (indicated by A1 and A2).

The difference between A1 and EC1 is one between a synthetic and analytic choice. The human expert chose EC1, a boundary that occurs before the D major key is established conclusively in bar 20. In light of the D major key context established in bar 20, bar 19 is heard retroactively as being in the key of D major rather than G. In D major, the right hand melody in bar 19 traverses the scale degrees $\{\hat{4}, \hat{2}, \hat{3}, \hat{4}, \hat{1}\}$. It is not an uncommon practice for a human listener to retroactively frame the past in the light of present knowledge. In addition, EC1 also prioritizes phrase symmetry by delineating a 2-bar + 2-bar phrase structure.

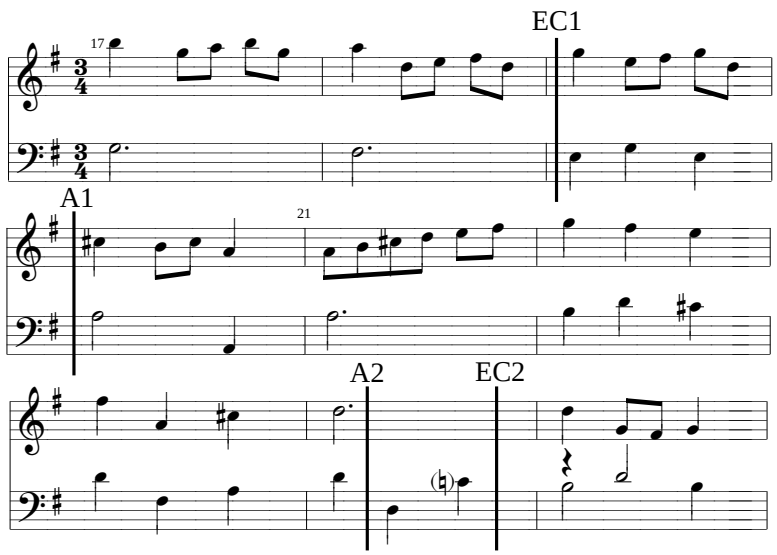

Fig. 4. Modulation boundaries. [EC = human expert; A = BSA algorithm]

The Boundary Search Algorithm described in Section 3 was not designed to have backward and forward analysis capabilities. It selects the boundaries that produce pitch event groupings in such a way that each grouping maps to point clusters on the Spiral Array that aggregate to points closest to the respective key representations. Based on this selection criteria, the BSA chooses the beginning of bar 20 to mark the beginning of the D major section. This choice, in fact, agrees with the textbook definition of modulation.

The textbook boundary for the next key change falls squarely between EC2 and A2. The human expert's decision was influenced by the four-bar phrasing established from the beginning of the piece. Giving preference to boundaries that lock into this four-bar period, the human listener waited for the latest phrase to end (at the end of the bar) before placing EC2 at the barline. With this choice of boundary, the D and C♮ on beats 2 and 3 in bar 24 are heard as an ending of the middle section. A performer, if attempting to highlight this fact, would play it quite different than if these two notes were the beginning of the next section.

Now consider A2. The BSA chose to group the notes in beats two and three of bar 24 with the final G major section, a choice that is closer in spirit to the textbook solution. In this case, the D on beat two of bar 24 is heard as the fifth degree $\hat{5}$ in G major, and not the first degree $\hat{1}$ in D major. Strictly speaking, the C♮ is the first note that belongs to G major and not D. Thus, the textbook boundary would start the G major area on beat three of bar 24.

Both the algorithm's and the expert's choices for modulation boundaries are valid for different reasons. The human expert's choices involved retroactive decisions that framed past information in the light of present knowledge. The human's choices were also influenced by phrase structure, preferring to rein-

force the four-bar phrase boundaries. The BSA considered only the pitch events, taking the pitches literally and choosing the boundaries accordingly.

4.2 Example 2: *Marche in D* by Bach

The second example is Bach's *Marche in D*, also from "A Little Notebook for Anna Magdalena". This piece was chosen because it is a little more complex than the previous Minuet example. In the transition from A major back to D major, the piece hints at a number of different keys before returning unequivocally to D major. The A major middle section shifts to D major in the second beat of bar 12, portending the return to D major. The D acts as a V to the G major tonality at the end of bar 13; a G♯ in the bass at the end of bar 14 acts as a leading note to A major; there is a momentary hint of B major between bars 15 and 16 that acts as the V of E minor; E minor acts as a V to A, which works as a V to D major which returns in bar 18. Clearly, this is a less than perfect example with which to test an algorithm that looks for boundaries between only three parts. The salient portion of the piece is shown in Figure 5, with the same labeling convention as before.

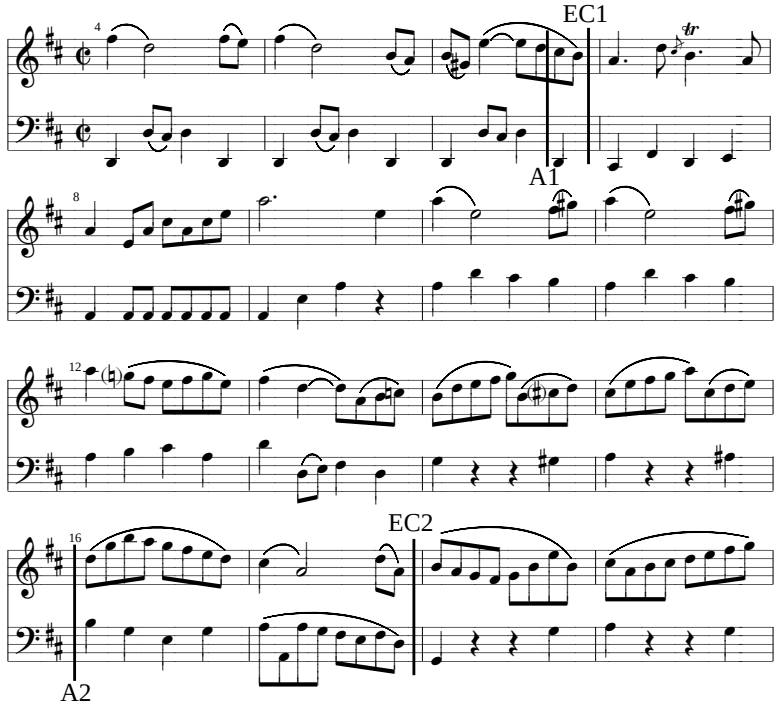

Fig. 5. Modulation boundaries. [EC = human expert; A = BSA algorithm]

The first set of boundaries, EC1 and A1, were almost identical. The BSA breaks up the phrase structure in bar 6 to include the last beat in the new A major section, while the human expert waited for the completion of the phrase to place the boundary EC1. The slight discrepancy resulted because the computer algorithm was not designed to recognize the composer's phrasings. Otherwise, the two choices are not very different. However, it is important to note that G♯ occurs on the first beat of bar 6, an indication of the new key area, A major; so, a textbook boundary should begin the A major section at the G♯.

For the second set of boundaries, marked EC2 and A2 respectively in Figure 5, the choices differed quite a bit. The human expert took into account the sequential pattern starting from the middle of bar 13 and ending at the climax in bar 16, and chose to think of the final D major section as beginning in beat one of bar 18. The BSA, on the other hand, did not account for figural groupings and sequential patterns in the notes. It chose, instead, to begin the last D major section as soon as the note material of the third part agreed with the pitch collection for D major.

5 Conclusions

The Boundary Search Algorithm is designed to segment a stream of music data into contiguous sets by key. The algorithm gave accurate assessments of the goodness-of-fit between each collection of pitch events and the key candidate to suggest optimal placements of key boundaries. For the purposes of this application, the c.e. defined in Section 3.1 summarizes only pitch and duration information. The method does not account for pitch order within each data set, nor does it incorporate beat information. With only pitch and duration information, the BSA was demonstrated to perform well in the problem of determining modulation boundaries. Given only pitch and duration information, a human would probably perform in a comparable manner. Rough temporal information is embodied in the sequence of the key segments.

The BSA does not explicitly use chord functions to determine the key areas. Chord membership and functional relations to each key area are incorporated into the Spiral Array model by design. With some additional work, a functional analysis of a musical passage can be extracted from mappings of the data to the Spiral Array model. The problem of harmonic analysis can perhaps be addressed in a future paper.

In addition, the implementation of the BSA as described in this paper is designed to analyze a piece of music in its entirety. In order for the algorithm to mimick human musical perception as information is revealed over time, the ideas outlined in this paper can be adapted to detect key boundaries in real-time. For example, in a real-time system, analyses can be performed at each time increment and the number of boundaries, m, allowed to increase by at most one.

Finally, the human mind in apprehending music information performs more than sequential data processing. Pitch and duration data is but a fraction of

the total information. The mind also organizes music data into figural groupings and phrase structures, and it can sometimes revise prior assessments based on new information. The analysis of the difference in the human and algorithmic solutions indicates that a dynamic interplay exists between memory and present knowledge so that the chances of finding meaningful patterns in the time series data is maximized.

6 Acknowledgements

I thank Jeanne Bamberger for her guidance and cogent advice that made this research possible; and, Martin Brody for his insightful comments on interpreting the results in an early version of this document.

References

1. Wolfgang Auhagen and Piet G. Vos: Experimental Methods in Tonality Induction Research: A Review. Music Perception **17**:4 (2000) 417–436
2. Jeanne S. Bamberger: Developing Musical Intuition. Oxford University Press, NY (2000)
3. Jeanne S. Bamberger: The Mind Behind the Musical Ear. Harvard University Press, MA (1991)
4. Brown: Tonal implications of the diatonic set. In Theory Only **5** (1981) 3–21
5. Chew, E.: Modeling Tonality: Applications to Music Cognition. Proceedings of the 23rd Annual Meeting of the Cognitive Science Society (2001)
6. Chew, E.: Towards a Mathematical Model of Tonality. PhD Dissertation, MIT (2000)
7. Richard Cohn: Introduction to Neo-Riemannian Theory: A Survey and a Historical Perspective. Journal of Music Theory **42**:2 (1998) 167–180
8. Richard Cohn: Neo-Riemannian Operations, Parsimonious Trichords, and their Tonnetz Representations. Journal of Music Theory **41**:1 (1997) 1–66
9. Peter Desain and Henkjan Honing and Huub Vanthienen and Luke Windsor: Computational Modeling of Music Cognition: Problem or Solution? . Music Perception **16**:1 (1998) 151–166
10. Robert Gjerdingen (ed.): Music Perception **17:4** (2000) Note: Special issue on Tonality Induction
11. Oswald Jonas: Introduction to the theory of Heinrich Shenker: the nature of the musical work of art. Longman, NY. (1982)
12. Krumhansl, C. L.: Perceived Triad Distance: Evidence Supporting the Psychological Reality of Neo-Riemannian Transformations. Journal of Music Theory **42**:2 (1998) 254–281
13. Krumhansl, Carol L.: Cognitive Foundations of Musical Pitch. Oxford University Press, NY (1990)
14. David Lewin: Generalized Musical Intervals and Transformations. Yale University Press, CT (1987)
15. H. C. Longuet-Higgins: Mental Processes. MIT Press, MA (1987)
16. H. C. Longuet-Higgins and M. J. Steedman: On Interpreting Bach. B. Meltzer and D. Michie (eds.) Machine Intelligence **6** Edinburgh U. Press (1971) 221

17. Robert Rowe: Key Induction in the Context of Interactive Performance. Music Perception **17**:4 (2000) 511–530
18. Shmulevich, I., Yli-Harja, O.: Localized Key-Finding: Algorithms and Applications. Music Perception **17**:4 (2000) 531–544
19. Steedman, Mark: The well-tempered computer. Phil. Trans. R. Soc. Lond. **349** (1994) 115–131
20. Temperley, David: What's Key for Key? The Krumhansl-Schmuckler Key-Finding Algorithm Reconsidered. Music Perception **17**:1 (1999) 65–100
21. Temperley, David: The Perception of Harmony and Tonality: An Algorithmic Approach. PhD dissertation, Columbia University (1996)
22. Piet G. Vos and Erwin W. Van Geenen: A Parallel-Processing Key-Finding Model. Music Perception **14**:2 (1996) 185–223