

Automatic Orchestration for Automatic Composition

Eliot Handelman

Centre for Interdisciplinary Research in Music Media and Technology
Montreal, Canada

Andie Sigler

Centre for Interdisciplinary Research in Music Media and Technology
and School of Computer Science
McGill University
Montreal, Canada

David Donna

School of Computer Science
McGill University
Montreal, Canada

Abstract

The automatic orchestration problem is that of assigning instruments or sounds to the notes of an unorchestrated score. This is related to, but distinct from, problems of automatic expressive interpretation. A simple algorithm is described that successfully orchestrates scores based on analysis of one musical structure – the “Z-chain.”

One way for an autonomous system to compose is to initially write parts for different instruments or sounds. A second strategy (one closer to some human practice) is to write a “flat” score first and then orchestrate it. The latter poses a more general problem, since a system capable of orchestrating its own compositions could also be capable of orchestrating other input, from Beethoven sonatas to Tin Pan Alley tunes to Xenakis. The goal of being able to orchestrate “known” music makes the system more susceptible to analysis and evaluation, and therefore to development.

Orchestration is a problem of music generation through analysis, occupying a place on the continuum from expressive interpretation (often framed as timing, articulation and dynamics) through to arrangement and variation-making. This class of problems is different from (and somewhat more constrained than) the goal of generating music from scratch. Since a successful orchestration will be reflective of, or synchronized with, things that are happening in the music, an analytical level is essential.

The flip side of the orchestration problem is that if we treat orchestration as a kind of sonified analysis, then not only can analysis help us make orchestrations, but careful listening and re-analysis of orchestrations generated through computational analysis can help us evaluate and refine our analytic methods.

Copyright © 2012, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

In this paper we define a narrow version of the orchestration problem that deals with assigning notes in the score to instrumental sound samples. A more complete version of the orchestration problem includes articulations and dynamics, while an extended version could include varying and adding to the score (“arranging”). A more constrained version of the orchestration problem would take into consideration the limitations and affordances of physical instrumental performance.

Strategies for Expressive Interpretation and Automatic Orchestration

There has not been much previous research in the area of automatic orchestration in the sense of autonomously assigning sounds to a score.¹ However, there has been research on expressive interpretation. The goal there is generally either to imitate human performance, to impart a mood as described by an adjective, or both. Like automatic orchestration, automatic interpretation starts with a flat score and modifies it through analysis, adding new layers of information. New layers could take the form of instructions for modifying dynamic levels or tempo, addition of accents, or assignment to instrumental sounds.

Approaches to expressive interpretation are generally based on determining sets of rules for assigning dynamic and tempo curves, articulations, and accents to notes. These rules are developed from music theories (Bisesi, Parncutt, and Friberg 2011), inferred through listening consultations with professional musicians (Friberg, Bresin, and Sundberg 2006), or derived through machine-learning analysis of human performances (Widmer 2003). Rules are often based on

¹Research on “automatic orchestration” at IRCAM focusses on analysis of timbre and instrumental capabilities for assisted orchestration, not the automatic assignment of sounds to a score (Carpentier et al. 2012).

analysis of grouping (phrase) structures at different hierarchical levels, as well as local melodic and harmonic events (Widmer and Goebel 2004; Friberg, Bresin, and Sundberg 2006; Canazza et al. 1998; Todd 1992). As a result, the performances are focussed on local effects, with global consistency stemming from consistency in rule parameters and in musical structure.

One general way these approaches could be extended would be to analyze longer range structure, so that it would be possible to control differentiated patterning of performance expression along a series of repetitions or variations. This kind of analysis might bring the addition of expressive layers a little further along a performance-arrangement-variation-composition continuum of music creation.

The orchestration problem is closer to the compositional level of musical action than are the problems of expressive performance. Therefore a different set of challenges are posed, in which the set of interesting solutions is perhaps more open. We need not imitate human-written orchestrations, nor model the expression of musical mood. We would be very happy to produce surprising, innovative orchestrations, as opposed to modelling a given style. In fact, we would like to use computational music analytic methods as a means for discovering *new* kinds of orchestrations.

In contrast with approaches to expressive imitation, therefore, we can proceed *without* a set of rules.² Instead of modelling a desired *outcome*, we can experiment with *processes* that might generate musically interesting yet unpredictable orchestrations.

A null-hypothesis strategy for the orchestration problem is to assign sounds to notes randomly. Another pre-analytic solution would be to assign each pitch to a different sound (so every C5 sounds like a trumpet, etc.). Beyond this, analysis is required.

The proposed strategy is to synchronize aspects of *structure* in the score with structure in the orchestration layer. An analytic system is used to select structures within the score (where a “structure,” generally speaking, is some abstractly specified data type which is instantiated by a set of notes in the score), and each structure is assigned to some sound. Given an analytic system that identifies overlapping structures of several different types, this is a strategy that can generate a large number of different orchestrations. The selection of structures and assignment of sounds can be done randomly, with human assistance, or with a more specific algorithmic approach.

The technical part of this paper describes an experiment in orchestration using just one of several musical structures currently being developed. Unlike the phrasing and grouping structures often used for expressive interpretation, the structures used here are non-hierarchical and non-contiguous, affording interleaving and overlapping of colors in the orchestration. The experimental hypothesis is that innovative and yet musically sensible orchestrations can be produced by synchronization with a (particular) structural level of the score.

²“Rule” in the sense of recognition of a musical situation to which a particular function will be applied.

Experiment

An ultimate goal is to generate full orchestrations with bells and whistles in appropriate places. This will entail a large-scale orchestration system with analytical access to many interacting musical structures as well as means of reasoning about combinations within a large database of articulate samples. The two sides of the coin, then, are reasoning about the score, and reasoning about the sounds.³ In this paper, we report on a preliminary experiment in which the focus is on analysis of the score for orchestration.

Harold Cohen asks: “What is the minimum condition under which a set of marks functions as an image?” (Cohen 1994). Likewise, it is of interest to find a *minimal* set of requirements which will result in orchestrations that function musically – that sound structured in relation to the perceivable musical structure.

A simple algorithm was developed in order to evaluate *one* kind of structure (called a *Z-chain*) for its utility in orchestration.

Z-chains are recursive structures built based on *orientation* (in the present case, orientation of *pitch*). They exist on local and large scales, with large-scale Z-chains built out of local-scale ones. Z-chains are built on the principle of *chaining like to like*, so that a Z-chain contains substructures with identical sets of recursive orientations. Z-chains therefore capture a certain kind of repetition structure which is on the one hand deterministic and rigorous, but on the other hand low-dimensional enough so that musical passages that are quite different can be part of the same Z-chain.

Chains and Z-chains

Suppose we have sequence of pitches (or of durations, loudnesses, or any other “oriented” feature in which for any two elements x and y , either x is greater, y is greater, or they are equal). *Chains* are a way of taking a sequence apart into different “dimensions” so we can look at different aspects of the sequence separately.

A chain is a consecutive subsequence of elements going in just one orientation. Figure 1 shows the three orientations of (pitch) chains in the beginning of the Menuet in G from the Anna Magdalena Bach Notebook. The top view shows chains going up (shaded pitches connected by lines). The middle view shows chains going down, and the bottom view shows “be” chains (repetition chains).

We can proceed by taking each view as a separate dimension, and recursively “chaining” chains together. Recursive chains are called *Z-chains* – named for their zig-zigging shapes. Figure 2 shows chains up with their top pitches respectively going up: a Z-chain up-up. Figure 3 continues the recursion, finding a Z-chain up-up-down-be. The Z-chains include entire chains, not just the top pitches.

We need not only look at the top pitches of chains. Figure 4 shows Z-chains with respect to bottom pitches: a Z-chain up-down-down-be-down, which spans most of the Menuet. We also can look at chain length, interval span, etc.

³Reasoning about the physical possibilities of instruments is a third element, in case human performance is desired.

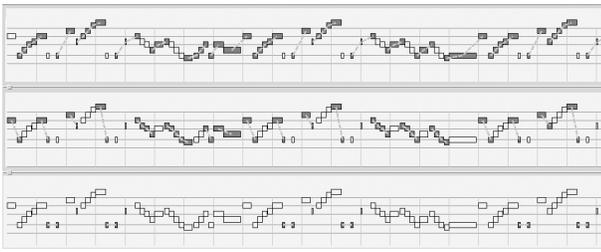


Figure 1: Up, down, and “be” chains in the Menuet.

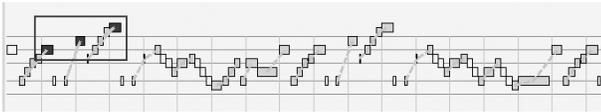


Figure 2: Z-chain up-up in the Menuet.

Z-chains describe aspects of musical structure on large and small scales. They are low-dimensional compared to the initial sequence, so there are many things they cannot describe. The kinds of things they *do* describe turn out to be useful for reasoning about music.

Since Z-chains are built on the principle of *chaining like to like*, with the *same* recursive orientations being chained together at a higher level, they describe low-dimensional repetitions. These, however, are structured in *oriented* relationships, as opposed to saying “here it is, here it is again” – allowing for further recursion and reasoning on larger structural scales.

Z-chains are efficiently computable. Given an initial sequence of n elements, one pass of the (simple and obvious) “chaining” algorithm delivers three lists of chains (up, down, and “be”) in $O(n)$ time. Now for a given comparison feature (e.g. top pitch), we run the chaining algorithm again on each of the lists of chains, resulting in lists of Z-chains with second-order orientations. To find Z-chains of all orders, we recurse again and again on all lists of length greater than one. Since each run of the chaining algorithm results in strictly fewer higher-order elements than the number of input elements, the recursive algorithm to find all existing Z-chains for a given comparison feature runs in $O(n^2)$ time.

Algorithm for Z-chain Orchestration

We want to find out to what extent a Z-chain analysis is a reflection of intuitive musical groupings, and whether simple use of Z-chains can result in musically interesting and differentiated orchestrations. Scores were orchestrated using the algorithm that follows.

Start with an unorchestrated (midi file) score and a set of 15 empty tracks with sounds chosen at random (for now we use general midi). Do the following for each track in the score:

1. Run the Z-chain algorithm in features “top pitch” and “bottom

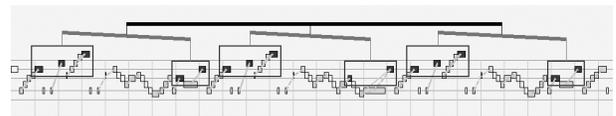


Figure 3: Z-chain up-up-down-be in the Menuet.



Figure 4: Z-chain up-down-down-be-down on bottom pitches in the Menuet.

pitch” in order to obtain a set of Z-chains.

2. Find the Z-chain Z that covers the largest set of unorchestrated notes.
 - if** Z covers less than three unorchestrated notes
 - then** terminate
 - else**
 - assign all notes covered by Z to one of the 15 tracks (chosen at random);
 - mark all notes covered by Z as orchestrated;
 - repeat step 2

The algorithm was run for each track of the midi score, where tracks were determined by the input midi files (which were not produced by the authors). The midi scores were often notated as just one track, or as one track for each hand of the piano, so in fact the Z-chain algorithm was run on a polyphonic or harmonic “voice,” whereas Z-chains have been defined for monophonic lines.

To deal with this, the midi tracks were pre-processed to find “top” and “bottom” lines. For each distinct onset time, the highest note is taken to be part of the top line and the lowest note to be part of the bottom line. Whenever a note from the one of the outer lines is used in a Z-chain, all of the notes in the chord under or above that note (if any) are included in the Z-chain as well.

Results

Results can be heard and subjectively appraised at www.computingmusic.com.⁴ Figure 5 shows an orchestration of the first ten measures of a Bagatelle (Op. 33 No. 1) by Beethoven.

We find that the orchestrations of Beethoven piano works, Debussy’s Children’s Corner, and other “classical” pieces are quirky, engaging, and musically interesting. The style of

⁴Z-chain orchestrations of automatically generated music (Handelman and Sigler 2012) are available as well, though since they were produced in an earlier stage of exploration, Z-chains were selected by hand (not by algorithm) for those pieces.

Figure 5: Automatic Z-chain orchestration of Bagatelle Op. 33 No. 1 (ms. 1-10) by Beethoven.

the orchestrations is evocative of Carl Stalling (arranger for *Looney Tunes*, *Merrie Melodies*, etc.), with fast-paced color changes, but with a large-scale logic that unfolds. There is constant coloristic transition, where a phrase can begin with one group of sounds that metamorphoses into another – a kind of psychedelic effect. Many details and aspects of overall organization sound as if they’re being orchestrated with a sense of what’s happening in the music. This is what we mean by the “synchronization” of orchestration with musical structure.

The orchestrations are as coherent as they are because Z-chains describe things in the music that are related, so that similar things get similar orchestrations: the orchestration is continuous with the music, and should change when the music changes.

Figure 6 shows the tracks of an automatic Z-chain orchestration for the Military March Op. 51 No. 1 by Schubert. In this zoomed-out view it is possible to see that repetitions in the music are orchestrated with similar (but not identical) combinations of sounds. Repetitions can be seen in tracks 7, 10, and 14, while other tracks vary the orchestrations of these passages. The middle section of the piece is readily apparent in the picture, using a different combination of tracks than the outer sections – some that have been largely held in

reserve up until this point (e.g. tracks 2, 11, 15). A nice effect is seen in track 2, where the instrument is introduced in the middle section and then continues through the remainder of the piece.

Orchestrations of Twenties pop tunes are plausible with some nice moments, but in general less interesting, with a more uniform sound throughout the piece (because of reuse of tracks for multiple Z-chains). This gives us a clue that there is something to investigate in the way Z-chains (as an analytic structure) might be distributed differently in different kinds of music.

The Debussy and Twenties pieces were orchestrated from one-voice midi files, with the Twenties tunes being unquantized performances. The algorithm nonetheless performs well on these pieces. Quantization is a non-issue for the analysis, and despite the fact that polyphonic voices were merged together into one track, the orchestration often articulated inner voices within a complex interaction of parts.

Since the algorithm is limited to one kind of structure, many concepts are missing. For example, there is no recognition of beginnings of phrases (– orchestrations in general need not always follow phrasing; e.g. Mahler). Unexpected rhythmic effects sometimes result, bringing out new rhythms systematically (sometimes in a hoquet style) which

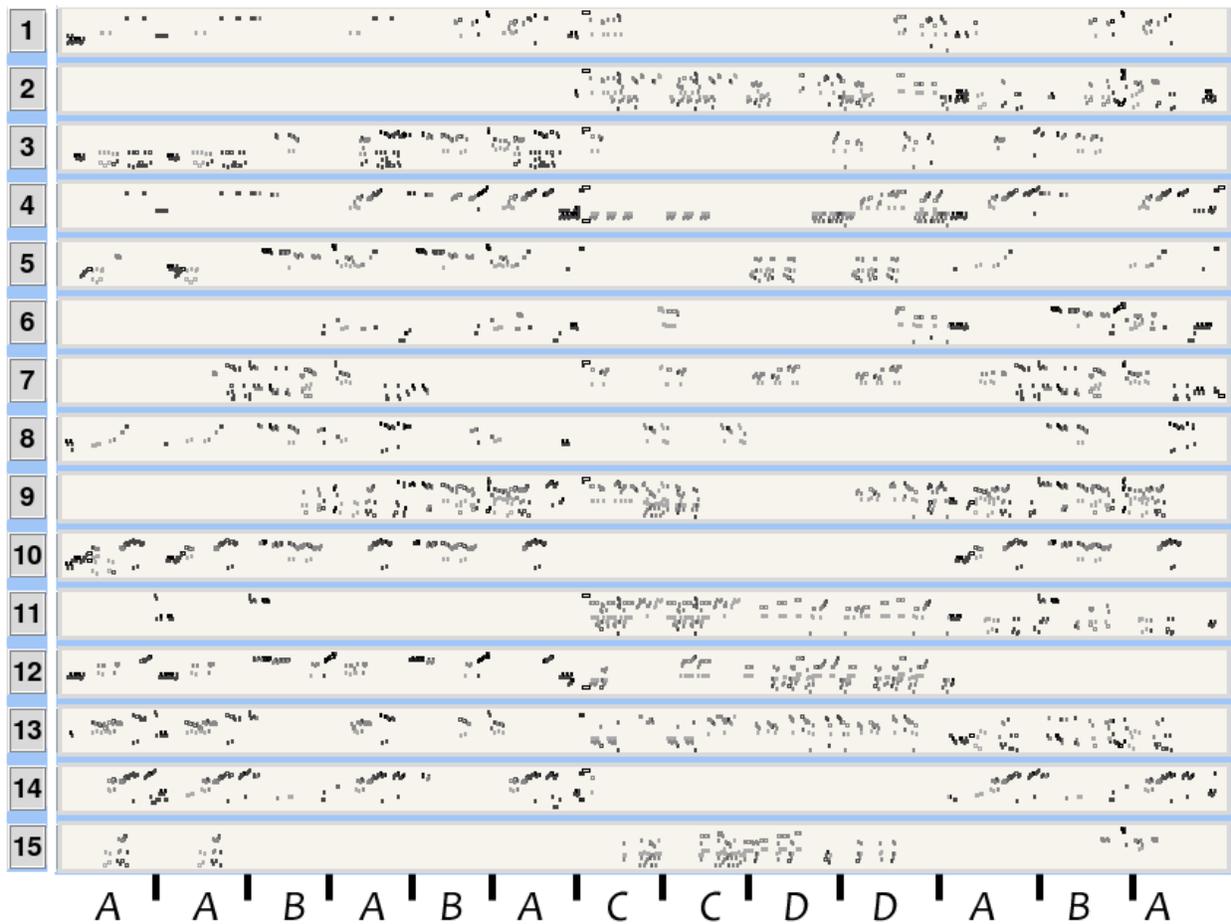


Figure 6: Automatic Z-chain orchestration of Military March Op. 51 No. 1 by Schubert. Letters roughly indicate the formal sections.

nonetheless often add up (that is, as trained musicians and composers, we subjectively find them interesting). Control over many other aspects is missing: for example, there is no concept of tutti; there is no control over the orchestration of chords, and so on. These are the kinds of problems that could likely be addressed as a constraint layer or a separate structural layer in a more complete orchestration system.

Next Steps

A problem with some of the orchestrations (especially of the pop arrangements) is their overall uniformity of sound. This was a result of a large number of Z-chains being selected and assigned to a limited number of tracks. Notable improvement was made by modifying the procedure slightly so that only the first 15 Z-chains were used and no Z-chains shared sounds; the selected structures were therefore more distinct. This algorithm can leave many unorchestrated gaps in the score; a method for completion of missed material is under development using another (related) kind of musical structure selected based on overlaps with orchestrated Z-chains. Another approach would be to reason about which Z-chains

could happily share a track.

A second strategy under development for producing less uniform orchestrations is to (automatically) divide the piece into sections or “zones” based on differentiations of structure, and make different (kinds of) orchestrations for each zone. (The current Z-chain algorithm sometimes creates this effect, but not for all types of pieces).

Experimentation is continuing with simple algorithms for orchestration using other musical structures, including synchronizations of orientation chains and structures dealing with chained intervals and durations. Interactions between different structures are also under investigation.

Another goal for upcoming algorithms is to voice implied polyphony (in midi files notated as one track), and to have stronger bass lines in particular. An approach to voice leading through structural parallels might be an alternative to “short path” methods.

In the stage of preliminary experimentation, general midi sounds were used, but clearly a higher quality and more articulate set of samples is desired, including different performance modes on the instruments. In order to manage and reason about a large database of samples, we are using tech-

nology developed for multi-feature analysis and combination of orchestral samples (Esling and Agon 2010). A longer term goal is to make orchestrations playable by human musicians – which will require specialized knowledge of the instruments. To this end, a program for exploring performance possibilities for string instruments is under development (Handelman and Sigler 2011).

Investigations are also under way to find out whether Z-chains (and other structures) can be used to make an articulation layer. In a preliminary experiment, we slightly modified the algorithm described in this paper to add articulations to solo violin works by Bach. Instead of using a variety of instruments, the tracks were assigned to randomly selected string sounds, each with a random velocity (loudness). Results are promising – certainly an improvement over the flat midi file.⁵ However, the problem of adding accents and dynamics may be harder than the orchestration problem: it may (or may not) be a more constrained problem, with different types of interactions with the music being necessary.

Finally, an important goal is for the orchestration system to be able to make a large number of significantly *different* orchestrations for a given score. In a system that can reason about many available musical structures, difference could be approached by keeping representations of significant structural properties of past orchestrations in memory, and avoiding similar ideas in the future.

Conclusions

The work presented here is not a system with encoded knowledge about music or about orchestration; it's an experiment in the application of a simple analytic structure and algorithm to orchestration – partly in order to test the precept that musical knowledge is necessary for musical creation. The results suggest that a relatively agnostic yet structure-bound system can be used both to comment on existing music and to creatively extend this music through orchestration. This kind of work raises questions about music cognition and the psychology of musical creativity, where we might investigate the role of basic pattern and structure processing as opposed to that of knowledge and learning.

The experiment also has implications for music analysis. Supposing that based on structural analysis of a score, some good orchestrations are made. Then not only was a strategy found to “solve” the orchestration problem, but something was demonstrated about the analytic system – that it makes musical sense and can be musically useful. Since (arguably) the only musical reality is within music, the transformation of an abstract (and probably abstruse) analysis into music is in some sense what makes the analysis real, as the resulting orchestration expresses a sonified structural interpretation of the piece. We might therefore claim not only to make an orchestration, but to make a *musical* analysis of the score.

The musical success of the orchestrations made with Z-chains provides evidence of a type beyond any visualisations, statistics, or argument, that Z-chains are indeed a sensible and useful structure for computational reasoning about music. This does not therefore mean that any composers ever

thought explicitly about Z-chains. It does suggest, however, that the way structural information is encoded in music has not yet been sufficiently described by the conventional categories of musical analysis or by musical information theory: a structural approach such as Z-chains is able to meaningfully cast light on the complexity and hidden riches found in music that likely participate in the cognition of music listening and musical creativity.

Acknowledgements

The authors acknowledge the support of Doina Precup, Prakash Panangaden, and the School of Computer Science at McGill University, the Centre for Interdisciplinary Research in Music Media and Technology (CIRMMT), and the Natural Sciences and Engineering Research Council of Canada (NSERC).

References

- Bisesi, E.; Parncutt, R.; and Friberg, A. 2011. An accent-based approach to performance rendering: Music theory meets music psychology. In *Proceedings of the International Symposium on Performance Science*.
- Canazza, S.; De Poli, G.; Di Sanzo, G.; and Vidolin, A. 1998. A model to add expressiveness to automatic musical performance. In *Proceedings of the International Computer Music Conference*.
- Carpentier, G.; Daubresse, E.; Vitoria, M. G.; Sakai, K.; and Carratero, F. V. 2012. Automatic orchestration in practice. *Computer Music Journal* 36(3):24–42.
- Cohen, H. 1994. The further exploits of Aaron, painter. *Stanford Humanities Review* 4(2).
- Esling, P., and Agon, C. 2010. Composition of sound mixtures with spectral maquettes. In *Proceedings of the International Computer Music Conference*.
- Friberg, A.; Bresin, R.; and Sundberg, J. 2006. Overview of the KTH rule system for musical performance. *Advances in Cognitive Psychology* 2(2-3):145–161.
- Handelman, E., and Sigler, A. 2011. Automated performance analysis of virtuosic music for string instruments. In *Proceedings of the International Symposium on Performance Science*.
- Handelman, E., and Sigler, A. 2012. Simulating the listening of composing. In *Proceedings of the Music, Mind and Invention Workshop*.
- Todd, N. P. M. 1992. The dynamics of dynamics: A model of musical expression. *Journal of the Acoustical Society of America* 91:3540–3550.
- Widmer, G., and Goebel, W. 2004. Computational models of expressive music performance: The state of the art. *Journal of New Music Research* 33(3):203–216.
- Widmer, G. 2003. Discovering simple rules in complex data: A meta-learning algorithm and some surprising musical discoveries. *Artificial Intelligence* 146(2):129–148.

⁵ Available for listening at <http://www.computingmusic.com>.