



HAL
open science

The structural function of musical texture: Towards a computer-assisted analysis of orchestration

Didier Guigue, Charles de Paiva Santana

► **To cite this version:**

Didier Guigue, Charles de Paiva Santana. The structural function of musical texture: Towards a computer-assisted analysis of orchestration. Journées d'Informatique Musicale (JIM 2018), May 2018, Amiens, France. hal-01791398

HAL Id: hal-01791398

<https://hal.science/hal-01791398v1>

Submitted on 14 May 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THE STRUCTURAL FUNCTION OF MUSICAL TEXTURE: TOWARDS A COMPUTER-ASSISTED ANALYSIS OF ORCHESTRATION

Didier Guigue
Mus³–Musicologia, Sonologia
& Computação – University of Campinas
didierguigue@gmail.com

Charles de Paiva Santana
Núcleo Interdisciplinar de Comunicação
Sonora – University of Campinas
charles@nics.unicamp.br

ABSTRACT

Musical form and orchestration are closely related dimensions of the musical discourse. Despite the proposition of several practical tools for assisted-orchestration in the past few years, scarcely any computational methods for analyzing the role of orchestration in shaping musical form and its perception have been put forward. As an answer to that, in this presentation, we discuss a couple of *Open-Music* functions we designed for estimating the textural complexity of a piece's successive formal segments and sections. The calculations behind the functions originate in an appropriation of the mathematical theory of integer partitions and Wallace Berry's ideas on musical texture. In the current stage, our estimation of textural complexity is carried out upon the composer's prescriptions found on the musical score. Nevertheless, the work we present here is part of an overall model that ultimately aims to analyze form and orchestration combining symbolical and signal data, that is, joining the musical score and its performance recordings.

1. INTRODUCTION

From the end of the 19th century onwards, western classical composers began to adopt increasingly sophisticated, systematic strategies of orchestration¹. As part of that trend, they were apt to actively correlate the orchestration technique with the crucial effort of structuring musical segments into a coherent whole, that is, the musical form. Eventually, different approaches for analyzing the orchestration technique of various composers and periods would emerge. For instance, Walter Piston, in his 1955 classic book on orchestration, stresses, although briefly, the relevance of such an undertaking:

The objective in analysis of orchestration is to discover how the orchestra is used as a medium to present musical thought. [...] It is a means of studying how instruments are combined to achieve a balance of sonority, unity and variety of tone color, clarity, brilliance,

¹ By orchestration, we mean the technique used to combine a large number of sonic resources.

expressiveness, and other musical values. Ultimately, the analytical process shows the differences in orchestral style between various composers and periods. [11, p. 397].

Afterward, the literature witnessed several developments concerning the analysis of orchestration; from the empirical, meticulous classifications of Charles Koechlin [10], through the pioneering sound-color theory of the 1978 book *Sound Design* [5], to the liberating and encompassing work of *Instrumentation in der Musik des 20. Jahrhunderts: Akustik-Instrument* [8].

Nowadays, in the era of computer-assisted composition and computational musicology, several practical tools for assisted-orchestration have been proposed. Even so, scarcely any computational methods for analyzing the role of orchestration in shaping and perceiving musical form have been put forward. As an answer to that, we have been developing a methodological model for investigating such structural functions of orchestration using quantitative and qualitative methods supported by symbolical (musical score) and signal (audio files) data.

In this presentation, we are going to introduce the overall model followed by the implementation of one of its main stages, namely *the qualitative evaluation of musical texture using symbolical data*. More specifically, we would like to share the following contributions:

- A formalism allowing the correspondence between the segmentation of a musical score and audio files of its performance (section 2).
- An adaptation of the theory of integer partitions, together with an interpretation of Wallace berry's ideas on texture, for evaluating texture complexity as a function of orchestration (section 3).
- A demonstration of the above concepts, involving symbolical data, implemented and running on the Openmusic[4] environment (section 4).

The Openmusic functions are available as part of the SOAL library. The presentation is concluded with a discussion on future developments and perspectives of the overall model.

2. THE OVERALL MODEL: FROM THE SYMBOLICAL TO THE MATERIALITY OF THE PERFORMANCE

In our model, the orchestration is regarded as a two-dimensional phenomenon. The first dimension is the normative level of the written code, that is, the composer's prescriptions as they appear on the musical score. The second one is the practical, acoustic level of its performance. In other words, it concerns the actual result of those abstract prescriptions which, in turn, can be examined through audio files (via signal processing).

By acknowledging both dimensions, we intend to keep away potential biases and to cover orchestral styles less reliant on the normative level.

The segmentation of the musical piece into instrumentally distinctive, successive segments, in both dimensions, is pivotal to the overall model. That means that musical score and audio file(s) are sectioned into corresponding subsections. The segmentation, however, is not carried out concurrently in both mediums. It is achieved by firstly examining the musical score and cataloging every individual component of the orchestral sonic palette. It means recognizing not only the instruments required to play the composition but also every mode of execution specified on the score, including extended techniques, as *pizzicato*, *flutter-tonguing*, *col legno*, *sul ponticello*, etc. We designate this catalog as being the *Index of Sonic Resources* (SRI), and it takes the form of a textual list.

By tracking every change in the combination of components of the index, we can segment the musical score into successive individual blocks. We call each of those blocks a *Local Sonic Setup* or LSS for short.

It is only after the determination of the musical score's LSS's that the respective audio file(s) of its performance(s) are fittingly sectioned. We identify each of these audio file segments as a *Local Audio Unit* or LAU for short².

LSSs and LAUs can be examined using quantitative or qualitative methods. LAU's analysis can be addressed by several audio feature's extraction algorithms.

An LSS can be examined, quantitatively, by counting the number of sonic resources it employs. For this purpose, we conceived a measurement that we call as the *Weighted Number of sonic Resources* or WNR for short³. Finally, an LSS can be examined, qualitatively, by a function we call as *Relative Voicing Complexity* (RVC) which we shall describe in this presentation. The analysis of LSS's and LAU's can be synthesized to obtain a measure of the orchestration's relative complexity. Figure 1 shows a flowchart outlining the overall model.

² We have been using AudioSculpt's *Positive Spectral Differencing* to achieve very satisfactory segmentations, which are then fine-tuned to match LSSs.

³ WNR workings should be detailed in a future paper.

3. THE QUALITATIVE EVALUATION OF MUSICAL TEXTURE

3.1. Partitional Analysis

We believe that the organization of sound resources into compound autonomous flows, that is, the orchestral texture, directly influences the dynamics of musical form. The texture is a dimension that characterizes a composer's style and reflects her particular way of negotiating with each instrument uniqueness and their role in stratified sound masses.

In *Structural Functions in Music* [3], Wallace Berry proposes a representation of musical texture based on the interdependency of the sound resources employed in a given musical structure. According to him, the musical texture consists of "real-components" – a part, or voice, that comprises one or more coordinated instruments – which are individualized in the polyphonic totality. Additionally, the sum of the sound resources used in a musical structure would also determine the musical texture.

This reasoning, as well as the numerical representation that it proposes, intersects with some fields of discrete mathematics, especially what is called as the theory of integer partitions[1, 2]. Its origins date back to Euler's work in 1748, and it was adapted to music theory by the Brazilian musicologist Gentil-Nunes [7, 6]. For instance, consider the number 5. It has 18 partitions, that is, 18 ways in which it can be represented as the sum of other integers. It means that if a composer wants to employ up to five sonic resources on a given musical passage, he or she can choose to lay them out into one of 18 combinations of groups and individual parts. Those combinations include all the partitions of the sub-groups as well, as the combinations of four, three and two simultaneous sound resources plus one resource alone. The partitions for a group of up to five sonic resources and its numerical representation are shown in table 1.

- (5) (4 1) (3 2) (3 1 1) (2 2 1) (2 1 1 1) (1 1 1 1 1)
- (4) (3 1) (2 2) (2 1 1) (1 1 1 1)
- (3) (2 1) (1 1 1)
- (2) (1 1)
- (1)

Table 1: All possible partitions of a *Local Sonic Setup* containing 5 (the sum) sound resources.

In our model, the partitions of a given group of sonic resources are represented by a list of integers enclosed in parenthesis. In this representation, the most extended list represents the most dispersed, viz. the most complex texture. For instance, consider the list (1 1 1 1 1). It is five elements in length and describes a texture of five sonic resources playing independent parts. This configuration is more complex than, let's say, the configuration represented by the list (2 1 1) whose length is 3 and refer to a texture composed by two resources playing coordinated parts and other two playing independent parts. Each element of such lists represents what is called a real component. The sim-

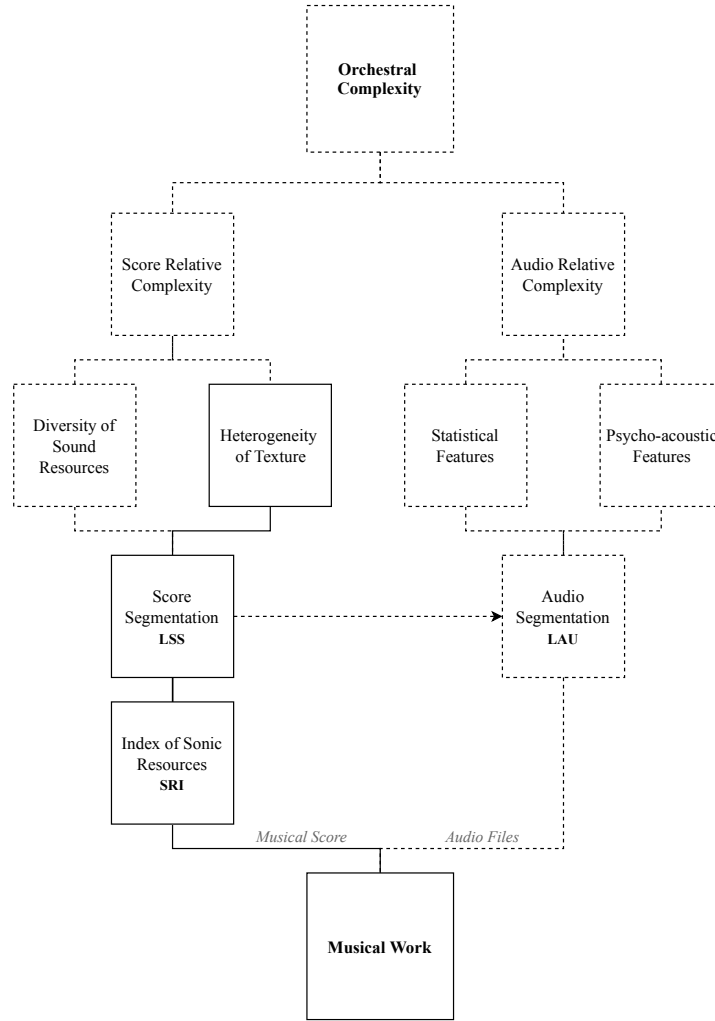


Figure 1: The overall model of our proposal for the computer-assisted analysis of orchestration. In this communication, we introduce the concept of an *Index of Sound Resources* which takes into consideration not only instruments but also their different modes of execution allowing the segmentation of the musical score into *Local Sonic Setups* (LSS). Each of these sonic configurations is analyzed by a qualitative method in terms of heterogeneity of the musical texture.

plest case for a texture of any number of sonic resources is the singleton list (1) that represents a texture comprised of one single solo instrument. Thus, we say that the complexity of texture is a function of its rate of dispersion and the magnitude of its real components.

To calculate the rates of interdependence and independence, denominated bellow as the agglomeration and dispersion rates, respectively, of a given *Local Sonic Setup*, firstly we need to count every combination, or rather every possible relation of any two elements of the LSS. We can do it by referring to the general formula for finding the number of combinations of p objects from a set of n objects, known as n choose p ,

$$C(n, p) = \frac{n!}{p!(n-p)!} = \frac{n(n-1)(n-2) \dots (n-p+1)}{p!} \quad (1)$$

For instance, in a setup composed of 4 sonic resources playing agglomerated parts, let's say a woodwind quartet, represented by the singleton list (4), there is a total of 6 unique pairs as $C(4, 2) = 6$, that is,

$$(Fl Ob) (Fl Cl) (Fl Fg) (Ob Cl) (Ob Fg) (Cl Fg).$$

On the other hand, in an LSS of four soloists playing independent parts, that is (1 1 1 1), when we consider any of its real-components or instruments, we find a total of zero unique pairs as $C(1, 2) = 0$.

We will refer to the total number of unique pairs of any resource or real component of a given setup as T_2 or simply T . By reworking the equation given at 1, we will define it as a function in the following way:

$$T_2: \mathbb{N}^* \rightarrow \mathbb{N} \\ n \mapsto \frac{n(n-1)}{2} \quad (2)$$

The successive total unique pairs $T_2(n)$ when n is mapped to the first eight positive integers, that is $1, 2 \dots 8$ is equal to 0, 1, 3, 6, 10, 15, 21, 28.

It follows that, in order to calculate the rate of interdependence, or agglomeration, of a given LSS, we need

to sum the T_2 value of each of its components. For instance, the rate of agglomeration of the setup represented by the list (2 1 1) is given by $(T_2(2) + T_2(1) + T_2(1))$ which results in 1. It is formally defined by the following summation function

$$\mathcal{A}: \mathbb{N}^r \rightarrow \mathbb{N}$$

$$(a_0 \dots a_{r-1}) \mapsto \sum_{i=0}^{r-1} T_2(a_i), \quad (3)$$

where the list $(a_0 \dots a_{r-1})$ represents an LSS, a_i each of its elements, that is, its real-components, and r the length of the LSS.

We denote the dispersion rate of a given LSS as the difference between T_2 value of its sum by its agglomeration rate:

$$\mathcal{D}: \mathbb{N}^r \rightarrow \mathbb{N}$$

$$(a_0 \dots a_{r-1}) \mapsto T_2(\rho) - \mathcal{A}(a_0 \dots a_{r-1}), \quad (4)$$

where ρ is the sum, the number of sonic resources of the LSS, that is, $\sum_{i=0}^{r-1} a_i$.

Both functions can be used to form a pair of indices from which graphs can be built and used to visualize the textural dynamics through time. In addition, the rates of dispersion and agglomeration can be combined and then visualized in one single dimension, symbolizing the global tendency of textural qualities, denoted by

$$\mathcal{I}(a_0 \dots a_{r-1}) = (\mathcal{D} - \mathcal{A})(a_0 \dots a_{r-1}). \quad (5)$$

In figure 2, the top chart shows the dispersion and agglomeration rates for a section (bars 56–73) of Anton Webern’s *Variations for Orchestra, Op. 30*, while the bottom chart displays the overall trend in texture complexity (eq. 5).

3.2. Relative Voicing Complexity

To correspond the texture of an LSS with the piece’s global sonic design, we propose to calculate “scaled” values for \mathcal{I} , where the lower boundary, 0, stands for the less complex texture and the upper limit, 1, the most complex. For now, the solution we are working with is to divide the \mathcal{I} value by the T_2 value of the greatest LSS sum⁴, i.e. the number of sound resources of the most dense LSS of the work, symbolized below by ρ_{\max} . We denote the ensuing quotient as the *Relative Voicing Complexity* or RVC for short,

$$\text{RVC}(a_0 \dots a_{r-1}) = \frac{\mathcal{I}(a_0 \dots a_{r-1})}{T_2(\rho_{\max})}. \quad (6)$$

The RVC value is scaled through the application of $\frac{(\cdot) - \min \lambda}{\max \lambda}$, where λ stands for $(a_0 \dots a_{r-1})$.

⁴ Alternatively, the number of total resources used in the musical work, that is, the number of elements of the score’s SRI, can be used.

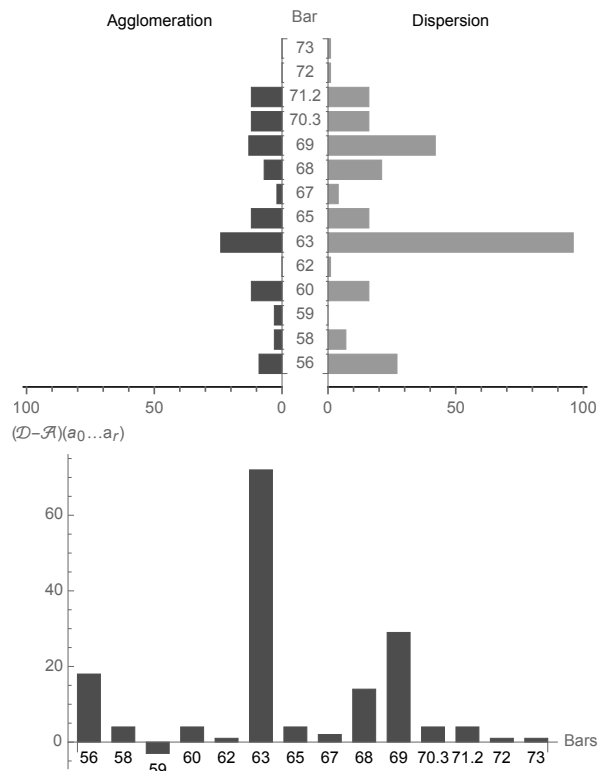


Figure 2: *Top:* agglomeration and dispersion rates for the bars 56 to 73 of Anton Webern’s *Variations for Orchestra, Op. 30*. *Bottom:* Difference between dispersion and agglomeration rates for the same excerpt.

3.3. Criteria for the evaluation of dispersion rates

When we identify the local sonic setups, the most common criterion to designate a sound resource as part of a given real-component is the rhythmic coordination between resources. Such criterion follows Wallace Berry’s correlation between rhythmic heterogeneity and textural complexity. But we preferred to leave to the choice of the musicologist the parameter or parameters he considers most appropriate, (generally according to the context and the work). We also give the possibility of placing them in hierarchical order by assigning them a more or less significant weight which would correspond to the impact that the parameter has on the dispersion.

This way, in our preliminary analysis of *Variations Op. 30*, besides rhythmic heterogeneity, we considered as a second heterogeneity parameter the tone-color heterogeneity, which is the textural organization into different flows of tonally homogeneous groups, like when we say *strings against brass*. In this particular analysis, tone-color heterogeneity was considered higher in the hierarchy and therefore weighted the calculation of an LSS’s dispersion rate towards greater complexity.

There are other parameters that we have experimentally included in our analysis. They include heterophony, heterogeneity of articulation, the heterogeneity of intensity.

Each criterion is mapped to a different positive integer, following the order 1 = *strongest* to n = *weakest*. The

RVC value can then be weighted by $(\cdot) [(c_{\max} - c)s + 1]$, where c stands for the criterion of dispersion and c_{\max} for the number of criteria. The weighting is further controlled by a percentage, s .

In case of non-dispersion (when all the resources of a setup are aggregated) (no criteria apply), an arbitrary integer, greater than the one corresponding to the total number of criteria adopted, is inserted so that the algorithm understands that the setup has no dispersion.

3.4. Structural organization of orchestration's and texture's analytical data

Following the examination of the musical score, the information regarding the orchestration and textural complexity is organized in the form of a multidimensional table. In this format, the temporal succession of LSS's is represented horizontally and labeled, seen on the *header row*, by their starting points in the form *bar.beat*.

The table's *header column* refers to the piece's index of sonic resources. Each cell of the table indicates whether the specified sonic resource is employed in the respective LSS. The integer number which fills a cell designates how many simultaneous sounds were allocated for the specified active sound resource.

Other information about the successive LSS's is conveyed in additional rows at the bottom of the table. These include the total number of resources used in the LSS, the weighted number of sound resources, the rates of agglomeration, dispersion, their sum, and so forth. Further supplementary rows may be used to convey additional evaluations given by other functions of the SOAL library. The multidimensional table may be processed with the assistance of spreadsheet software.

4. OPENMUSIC IMPLEMENTATION

SOAL, the *Sonic Object Analysis Library* [9], is an OpenMusic external library that we continuously develop at the *Musicologia, Sonologia & Computação* group, the Mus3⁵. It is conceived to be useful for a range of analytical purposes and supports a top-down approach.

SOAL's central concept is the *Compound Sonic Unit* or CSU. We define it as the combination and interaction of musical primary and secondary components: the former refers to collections of pitches and the latter to aspects as intensities, ranges, registers, densities and so forth. We also recognize as secondary components statistical measurements such as distributions, deviations, and entropy, among others.

SOAL is modular; new ad-hoc functions and components can be easily incorporated into the library. SOAL allows the inference of musical structures by comparing the relative sonic qualities of a sequence of CSU's and ultimately symbolizing them by a *vector of relative complexity*.

⁵ The library can be downloaded at <http://git.nics.unicamp.br/mus3-OM/soal4/tags>

	56	58	59	60	62	63	65
Fl	1					1	1
Fl <i>flatterzung</i>							
Ob	1		1			1	1
Cl[Bb]	1		1			1	1
Bcl			1			1	1
Hn[F]							
Hn[F] <i>sord.</i>	1	1		1		1	1
Tpt[C]							
Tpt[C] <i>sord.</i>	1	1		1		1	1
Tbn							
Tbn <i>sord.</i>	1	1		1		1	1
Tba							
Tba <i>sord.</i>				1		1	1
Cel					4		
Hp		4			4		
Hp <i>harm.</i>							
Timp							
Timp trill							
Vn 1 <i>solo arco</i>		1					
Vn 1 <i>solo pizz.</i>							
Vn 1 <i>solo pizz. sord.</i>							
Vn 1 <i>div. pizz.</i>						1	
Vn 1 <i>tutti arco</i>							
Vn 1 <i>tutti sord.</i>							
Vn 1 <i>tutti harm.</i>							
Vn 1 <i>tutti pizz.</i>							
Vn 1 <i>tutti pizz. sord.</i>							
[etc.]							
Db <i>pizz.</i>							
Total resources	9	5	3	8	2	16	8
WNR	.67	.49	.34	.64	.21	.85	.64
Agglomeration	9	3	3	12	0	24	12
Dispersion	27	7	0	16	1	96	16
Difference	18	4	-3	4	1	72	4
Setup Complexity	.9	.52	.33	.69	.21	1	.69

Table 2: Structured data from orchestration and textural analysis of Anton Webern's *Variations for Orchestra, Op. 30*. *Header column* refers to the SRI of the work, while the *header row* indicates the bar numbers.

The analysis of the *Relative Voicing Complexity* is, thus, one more option in the collection of functions for top-down music analysis. Indeed, an LSS is a CSU described by its sonic or instrumental configuration.

As with every other OpenMusic library, SOAL's functions are organized by purpose in different folders. SOAL's folder partitional analysis includes the function *soal-texture-complexity* which proposes an assisted-analysis of musical texture according to the premises exposed earlier in this presentation.

Figure 3 shows an OpenMusic patch as typically programmed for use with *soal-texture-complexity*. The list connected to the leftmost input of the function refers to the sequence of LSS's in the form ((*partition*) (*criterion*)). The list of bar numbers is inserted in the 5th input (from left to right). Other arguments include the list of total sound resources per LSS and the total number of dispersion criteria (cf. section 3.3).

The figure also highlights some of the function's outputs

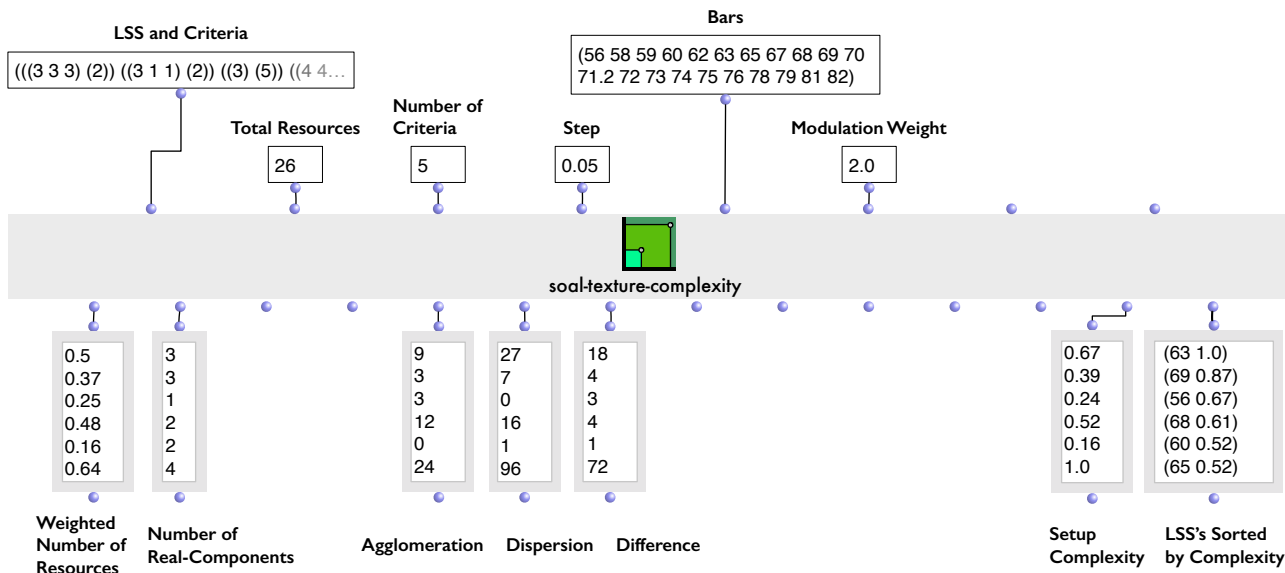


Figure 3: OpenMusic patch with the `soal-texture-complexity` function prototype in use to analyze bars 56–82 of Anton Webern’s *Variations for Orchestra, Op. 30*.

such as:

1. the *weighted number of resources*, $\text{WNR}, \frac{\log \rho}{\log \rho_{\max}}$;
2. the number of real components, r ;
3. the list of agglomeration rates, eq. 3;
4. the list of dispersion rates, eq. 4;
5. the list of differences between dispersion and agglomeration rates, eq. 5;
6. the *relative setup complexity*⁶, $\text{WNR}(\text{RVC}_w + 1)$;
7. The LSSs list, labeled by bar number, sorted from the most complex to the less complex.

Figure 4 shows an example of the kind of results our approach can produce. It depicts the analysis made on an excerpt from Beethoven’s *5th symphony*, more precisely, the last bars of the 3rd movement together with the *attacca* which opens the finale. Even though not surprising, the analysis indicates that the finale’s opening tutti - the place where all the expanded orchestra’s resources are ultimately activated - correspond to the less complex texture of the excerpt, as practically all instruments play in homophony. The homophony explains why the last bar of the chart is so flat (cf. bar 242), even though the music reaches, at that moment, its most thrilling point, one could say.

5. CONCLUSIONS

Our experimental model for computer-assisted analysis proposes a formal strategy for evaluating the role of orchestration in musical structure and perception. It works by

⁶ The RVC weight onto WNR can be adjusted by the user

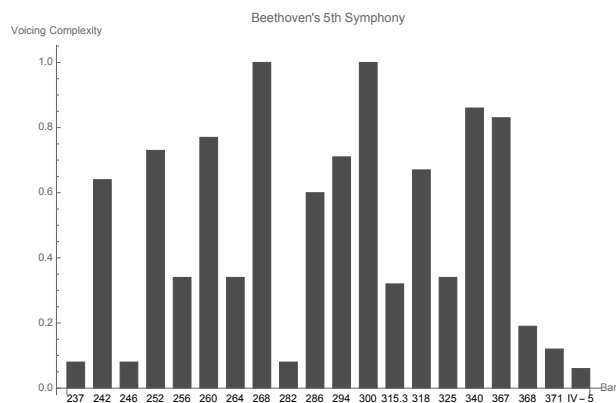


Figure 4: A sample of a scaled and weighted RVC analysis: Beethoven, *5th Symphony*, third movement, bars 237–371 plus first bars of *Finale*.

looking at the symbolic level of the score’s prescriptions and the “perceptual level” of the performed music. At the symbolic level, the musical texture is considered as a qualitative aspect of orchestration, whereas the number of sonic resources involved in each local setup is the qualitative aspect. The analysis we put forward makes use of an application of the theory of partitions. We described how to collect and format (section 3.4) data from the score, its mathematical background (section 3.1), how the implemented function works (section 4), and the kind of results it may return (figures 2, 4 and table 2).

Nevertheless, there is still a great deal of research to do for the completion of the project as exposed in figure 1. For instance, the audio branch of our model (right side of figure 1) is at a very incipient stage: choosing, importing, developing and implementing appropriate descriptors is not a trivial task and demand further extensive research and

experiments. Furthermore, the impact of an LSS' duration upon the overall musical perception should be properly addressed in the next stages of our work.

Some aspects of orchestration are omitted in our model, such as the impact of polyphonic resources (e.g., a piano) playing agglomerated sounds (e.g., chords) in an overall dispersed texture. We are inclined to think such behavior works against the global sensation of dispersion, but we have no data, at the moment, that could help us to quantify to which extent this occurs.

In a more technical note, soal-texture-complexity should be able to incorporate a (semi)-automated tool for analyzing and structuring data as presented in table 2. Pattern recognition of texture or instrumental configurations in a single work, in a composer's whole oeuvre, or for some historical period, could bring a relevant complementary analysis. Such devices would significantly reduce the fastidious manual labor involved in the process. In fine, soal-texture-complexity, and the library as a whole should be implemented as stand-alone software in future works.

In any case, based on our first experimental results, we are quite confident that this approach consists of a helpful contribution in the field of computer-assisted music analysis.

6. REFERENCES

- [1] Andrews, G. *The theory of partitions*. Cambridge University Press, 1984.
- [2] Andrews, G. & Eriksson, K. *Integer Partitions*. Cambridge University Press, 2004.
- [3] Berry, W. *Structural functions in music*. Dover, New York, 1987.
- [4] Bresson, J, Agon, C., Assayag, G. "OpenMusic: visual programming environment for music composition, analysis and research." In *Proceedings of the 19th ACM international conference on Multimedia*, pp. 743-746. Scottsdale, USA, 2011.
- [5] Cogan, R. & Escot, P. *Sonic Design: The Nature Of Sound And Music*. Prentice-Hal, New Jersey, 1976.
- [6] Gentil-Nunes, P. *Análise participacional: uma mediação entre composição musical e a teoria das partições*. PhD thesis. Universidade Federal do Estado do Rio de Janeiro, 2009.
- [7] Gentil-Nunes, P. & Carvalho, A. "Densidade e linearidade na configuração de texturas musicais". *Proceedings of IV Colóquio de Pesquisa do Programa de Pós-Graduação da Escola de Música da UFRJ*. Rio de Janeiro: UFRJ, 2003.
- [8] Gieseler, W., Lombardi, L., Weyer, R. D. *Instrumentation in der Musik des 20. Jahrhunderts*. Moeck Verlag, Celle, 1985.
- [9] Guigue, D. *Sonic Object Analysis Library – OpenMusic Tools For Analyzing Musical Objects Structure*. 2016. <http://www.ccta.ufpb.br/Mus3>.
- [10] Koechlin, C. *Traité de l'Orchestration*. Max Eschig, Paris, 1954.
- [11] Piston, W. *Orchestration*. Norton, New York, 1969 [1955].