A Computer Model for the Analysis of Diversity and Coordination in Orchestration

Charles de Paiva Santana
University of Campinas
Campinas, Brazil
cdepaiva@unicamp.br

Didier Guigue
Universidade Federal da Paraíba
João Pessoa, Brazil
didierguigue@gmail.com

We introduce a method for computer-assisted analysis of musical texture and orchestration. Our aim is to establish a formal understanding of these symbolic dimensions that could be on par with existing computational approaches to rhythm and pitch. We also propose to investigate the role that texture and orchestration control has in structuring musical form. Our research is based on the theoretical considerations made by Wallace Berry in the classic *Structural Functions in Music* and the subsequent numerical representation and combinatorial manipulation, grounded on the mathematical *theory of integer partitions* proposed by Pauxy Gentil-Nunes. To do this, we assumed that each local sonic configuration (an orchestral configuration in the case of symphonic music), or *Local Sonic Setup*, delineate a sound unit. The qualification of these setups according to the number of sonic resources used and the way in which they are distributed to create more or less polyphonic complexity is information that is added to the other dimensions that contribute to the development of a dynamic of the form through sound.

Music analysis. Computational musicology. Orchestration. Musical texture.

1. INTRODUCTION

Instrumental sound-colour and musical texture are two aspects of compositional expression that are of fundamental importance for understanding music, whether in its historical, creative or technological dimension. Many of the major stylistic and aesthetic breakthroughs that have occurred throughout western music have represented a significant change in the type of prevailing musical texture and vocal or instrumental technique. It would be enough to look, by way of example, at what represented the development that started from the monophonic and exclusively vocal style of the medieval plainchant and led to the homophony of the *Organum* (ca. 9th century) and the *Ars Antiqua* (12th century), or the development starting from the latter and leading to the polyphony of *Ars Nova* (14th century) and ultimately to the apogee of the modal polyphony in the vocal and instrumental music of the Franco-Flemish school (15th and 16th centuries).

The historical trajectory as seen through the use of a certain predominant texture, necessarily involved technological or creative possibilities related to the technique and quality of vocal and instrumental sound. Starting from composers like J.-P. Rameau (1683–1764), H. Berlioz (1803–1869), L.

Beethoven (1770–1827) and R. Wagner (1813–1883), instrumental timbre and musical texture increasingly became part of the formal articulation within an individual musical work. In the modern and contemporary periods, where melodic and rhythmic elements, or even harmonic progression, are no longer the preferred and main articulators of musical form, various compositional practices are defined and appreciated specially by the specific pattern of texture and sonority they employ in their creations.

Although the pointillistic texture and the *Klangfarbenmelodie* of the Second Viennese School, the 'clouds' of G. Ligeti's (1923–2006) micropolyphony, and the liminal sound-objects of G. Grisey's (1946–1998) spectral music are common themes in specialised literature, few proposals for a systematic theory of sound-colour and texture as articulators of musical form have come to the fore. Even when we look at musical scores where sound and texture are the essential raw materials of musical construction, we approach them with tools of representation and quantification that do not have the sophistication and precision that the analytical work requires, especially if we

compare such tools with what is normally employed in terms of pitch, harmony and rhythm.

While a few practical computer-assisted orchestration tools have been proposed recently, most (symbolic) assisted analysis systems, including top-of-the-line tools such as C. Ariza's *Music21* or D. Huron's *Hundrum*, do not offer specific models for detailed observation of textural setups, and the question of timbre is generally relegated to what is offered by the restrictive MIDI and MusicXML formats. As long as we continue to focus almost exclusively on the world of the 'note' away from the reality of the 'sound', our analytical technology will remain below the needs of contemporary creation and research.

As a response to what we might call a disparity between the leading role of the notion of sonority in modern composition and the absence of analytical models by musicological studies, we have been developing a strategy to objectively investigate the role of timbre and musical texture as articulators of the musical form. In this communication we would like to introduce the global model of our approach, which is a work in progress, followed by the implementation of some of its aspects, notably the assessment of heterogeneity and diversity between different sonic configurations.

Our research is based on the theoretical considerations made by Wallace Berry in the classic *Structural Functions in Music* (1976) and the subsequent numerical representation and combinatorial manipulation, grounded on the mathematical *theory of integer partitions*, proposed by Brazilian composer Pauxy Gentil-Nunes (2009).

Our strategy puts forward:

- A general numerical representation that allows the abstraction and subsequent computational manipulation of textural configurations.
- A hierarchy of 'criteria of dispersion', or 'textural situations', that allow the stratification of the musical surface into different real components.
- A measure that allows the quantification of heterogeneity relationships in textural configurations.
- A measure that estimates how diversely sound resources are used on the realisation of textural configurations.
- A model for relative texture complexity based on how diverse is the orchestration and how intricate is the allocation of real components in a given musical work.

Functions implementing the model were programmed as part of a package (SOAL) for *OpenMusic*, a computer-assisted composition environment developed at the French *Institut de Recherche et Coordination Acoustique/Musique*, and as a Jupyter Notebook (using the Python programming language). The Library can be downloaded at <https://git.nics.unicamp.br/mus3-OM/soal4>.

2. MAIN CONCEPTS AND PRE-ANALYTICAL WORK

Our strategy considers instrumentation and musical texture as elements of musical expression, which are primarily approached through two complementary perspectives. The first perspective is concerned with what could be called as the normative aspect of the written code, that is, the issues related to composers' instructions and decisions as they appear in the musical score or are revealed in a transcription. It is connected to what we call in the musicological jargon as the symbolic level.

In general, symbolic-representation-based analyses might deal with, roughly speaking, questions about compositional grammar and musical notation. Symbolic-level analysis of orchestration and texture is the norm because the score allows us to access data which otherwise would be difficult to obtain, specially in relation to what precisely constitutes each line or instrumental part in a textural setup. Information which is easily accessible in a musical score or musicological transcription may be very difficult to get by listening to a performance alone or through computational means with sound files.

The second viewpoint goes into the practical and acoustic reality of the musical performance. That is, the second, complementary perspective is concerned with the sonic result or 'sonority' of such symbolic prescriptions and instructions. After all, how a particular note or instrumental instruction actually sounds is also part of what stimulates composers' imagination to select and structure them in a particular formal design. Musical performances may be examined, for instance, by means of its recording's audio files and tools for signal processing.

By acknowledging the weight and complementarity of these two facets, we expect to avoid precipitated conclusions or overestimations when an observation is made by looking either mostly at the score or mostly at a performance's recording.

The discretisation of the musical composition into instrumentally distinctive, successive segments,

both in the symbolic and the audio formats, plays a central role in our approach. This means that the musical score and recording's audio files are divided into pertinent subsections, which, as a matter of fact, may or may not correspond to more traditional formal assumptions. In other words, when a piece of music is segmented according to the instrumental timbre and playing technique employed by the composer, it may coincide with how other parameters are structured, such as the melodic outline and harmonic progression, or it could be revealed to act as an independent dimension.

As an example, we could mention M. Ravel's *Boléro* (1928), a remarkable illustration of placing orchestration at the centre of musical development while other parameters, such as pitch, rhythm and tempo, are fixed. In the French composer's most recognised score, whenever the C-major theme or the phrygian counter-theme is restated, a new orchestral configuration is deployed. As a result, segmenting and isolating the orchestral form of Ravel's *Boléro* would correspond to delineating each melodic succession, which occurs at every 16 or 18 bars.

Anton Webern's (1883–1945) masterful orchestration of *Ricercar a 6* (1935) from J.S. Bach's *Musical Offering* splits its famous royal theme and counter-melodies into several fragments, to each a wide and varied palette of orchestral colours is brought into play. The presentation of the 21-note theme at the very beginning of the score, is rendered by five unique instrumental configurations in the course of six melodic fragments (a solo horn with mute, playing softly, is employed in two different fragments) and from there the orchestration becomes increasingly more detailed. In the *Ricercar* case, segmenting and isolating the orchestral form reveals a much less linear relationship with the thematic and contrapuntal material, although the distinctive way Webern's approaches them is strongly related with his interpretation of the original composition's inner structural features.

Segmentation in the symbolic domain is achieved by first examining the score and cataloguing each individual component of the orchestral sound palette. This means that not only are the instruments required to play the composition identified, but also each mode of execution indicated in the score, including so-called extended techniques, and effects, such as *pizzicato*, *con sordina*, harmonics, flutter-tonguing, *col legno*, *sul ponticello*, etc. Other information is collected as well, such as how many times the same sound resource can be used simultaneously.

We refer to this catalogue as the *Sonic Resource Index* (SRI), and it usually takes the form of a textual list or table (see Table 1). By following every change in the orchestration, we can divide the musical score into a sequence of single blocks or units. Each of these blocks is called a *Local Sonic Setup* or LSS for short.

After the LSS of the score has been established, the relevant audio files from its performances are appropriately divided into sections. Each of these audio file segments is called a *Local Audio Unit* or LAU for short. The LSS and LAU can be studied using a number of numerical descriptors. LAU analysis can be performed using several algorithms to extract audio features, such as spectral centroid, zero-crossing rate, spectral roll-off, MFCC, etc.

An LSS can be approached, for instance, by counting the sound resources it uses. For this purpose, we have developed a measure, which we call the *Weighted Number of Sonic Resources* or WNR for short. An LSS can also be examined with a function called *Relative Voicing Complexity* (RVC), which we will describe later in this document. A number of numerical descriptors for LSS and LAU are combined to achieve a measure of relative complexity of orchestration and musical texture.

Table 1: *Sonic Resource Index of De Natural Sonoris no.1 (1966) by Krzysztof Penderecki (1933). The score specifies 86 musicians, including the exact number of string players needed for each section. The score indicates a total of 60 different sound effects, of which 52 different sounds can be played simultaneously.*

instr.	#players	#sounds	max-sim.
4 fl. (2 picc.)	4	2	2
ob.	3	1	1
eh.	1	1	1
cl.	2	1	1
bcl.	1	1	1
sax.	2	1	1
bn.	3	1	1
cbn.	1	1	1
hn.	6	1	1
tpt.	4	1	1
tbn.	3	1	1
tba.	1	1	1
perc.	6	24	6
pf.	1	1	1
harm.	1	1	1
flexatone	1	1	1
vn.	24	8	8
vl.	8	8	8
vc.	8	8	8
db.	6	6	6
Total	86	60	52

3. NUMERICAL REPRESENTATION AND PARTITIONAL ANALYSIS

In our approach, musical texture is represented by a nested list of integers. For instance, consider the following simple list: [2,1,2]. It may refer to a musical segment, where: 2 voices are coordinated in a first 'textural part' or layer; a middle layer is composed by a single voice and, finally; another 2 voices are also coordinated in a separate third layer. This textural setup could, for instance, in the case of a wind quintet, be materialised by: 2 instruments, a piccolo and an oboe, playing, let's say, same rhythm staccato dyads; A single clarinet playing a different, independent part; the remaining two instruments, a French horn and a bassoon, assure the lower register while reinforced by some kind of doubling or contrapuntal transformation.

That is the case of a small excerpt, the beginning of the coda of the second movement of Arnold Schoenberg's (1874–1951) *Wind Quintet* (1924), Op. 26, bars 360-362 shown in Figure 1 below.

Figure 1: Bars 360-362 from Arnold Schoenberg's *Wind Quintet*, Op. 26. The excerpt can be represented by the list [2,1,2].

In the above dodecaphonic excerpt, we can observe that the piccolo and the oboe can be seen analytically and eventually heard as a homogeneous stratum of the texture, as they share same rhythm, articulations, and expression. Even though they play different pitches, they contribute to the perception of homogeneity by means of a coordinated contrary motion (inverted serial forms).

The exact same could be said of the textural layer constituted by French horn and bassoon. Although different from the first stratum, they also share the same rhythm, articulation, and expression, establishing another distinguishable component of the musical surface. The middle layer, driven by the clarinet, is given a different character, with

attributes carefully chosen to come out as yet another component of the musical texture.

On that account, in such textural configurations, we observe relations of coordination, of homogeneity, occurring at the same time with relations of heterogeneity.

Needless to say, the ordinary three-integer-list introduced above could represent an endless number of textural situations, not only that of the Schoenberg example. Moreover, very often it isn't that straightforward to categorically determine what is coordinated or not; there are gradations of what can be regarded as uniform or heterogeneous.

Note that the list [2,1,2] is equivalent to the unordered [2,2,1], both meaning a textural setup of three strata of two pairs and a single sound resources. As such the excerpt below from the *Pierrot Lunaire*, *Raub*, bars 16-17, has an analogous textural configuration from that of the wind quintet excerpt above:

Figure 2: Bars 16-17 from Arnold Schoenberg's *Pierrot Lunaire*, *Raub*. The excerpt refers to another instance of the sonic setup [2,1,2] or [2,2,1].

In this excerpt, flute and clarinet, violin and violoncello, and reciter materialise the three strata of the texture setup [2,2,1]. If in the wind quintet excerpt the homogeneity of the coordinated parts were reinforced by a contrary motion, here the homogeneity of the coordinated strata (fl. and cl., vln. and vc.) are cemented by their timbre, as they are part of the same instrument families (woods vs. strings vs. voice). We could say that the real components are formed by homorhythm and homochromy (i.e., having a similar or same sound-color).

Each stratum, or layer, of a textural setup is called a *real component*. A real component can refer to one or more coordinated sound resources (see Berry 1987).

As a matter of fact, there are 18 ways in which the number 5 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 below:

[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]						

Now, 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].

The simplest 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 .

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 solving n choose p for p equal to 2, we will define it as a function in the following way:

$$T_2: \mathbb{N}^* \mapsto \mathbb{N}$$

$$n \mapsto \frac{n(n-1)}{2}.$$

The successive total unique pairs $T_2(n)$ when n is mapped to the first eight positive integers, that is 1, 2 ... 8 is equal to 0, 1, 3, 6, 10, 15, 21, 28. (cf. Gentil-Nunes 2009).

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

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

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

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:

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

$$[a_0 \dots a_{r-1}] \mapsto T_2(\rho) - \mathcal{A}([a_0 \dots a_{r-1}])$$

where ρ is a sum, the number of sonic resources of the LSS.

Figure 3 shows the dispersion and agglomeration rates for a section (bars 56–73) of Anton Webern's *Variations for Orchestra*, Op. 30.

Figure 3: Agglomeration and dispersion values for successive local sonic setups of bars 56-73 from Anton Webern's *Variations for Orchestra* Op. 30.

4. THE CRITERIA OF DISPERSION

The determination of heterogeneity and homogeneity relations present in a given textural configuration, when carried out in a too deterministic manner, can be limiting and naive. One of the greatest challenges of the methodology presented here is to make it flexible enough to be applicable in real situations of musical analysis, without it becoming totally arbitrary or applicable on a purely *ad hoc* basis.

To approach this type of situation, what is needed is a way to incorporate in the model a gradation or the degree of confidence in classifying a given sonic setup as more homogeneous. The representation could cover, for instance, which criteria was taken into account and their respective weight for determining the textural setups.

For that reason, we implemented a provisional, experimental strategy for dealing with situations where aspects other than rhythmic coordination happen to be more important for the analysis. When analysing such situations, the musicologist organises, in a hierarchical order, the aspects, or parameters, of the musical surface, which were important for determining the different strata of the textural configurations. For each segment of the composition under consideration an appropriate criterion is assigned.

Influenced by Wallace Berry (1987), we propose the following list of dispersion criteria, ordered by weight:

- Prevailing **heterochromy**: when the stratification of sound resources is endorsed by sound-colour and/or instrument section.
- Prevailing **heterorhythm**: when sound resources are stratified into real components based on rhythmic coordination but not endorsed by sound-colour. It may be supported by other aspects as dynamics and articulation.
- Melodic **heterokinesis**: when rhythmic coordination cannot be determined and real components are differentiated primarily by the pitch directionality.
- **Heteroarthria**: when rhythmic coordination cannot be determined but real components share same articulation (ex.: *legato* real-component vs *staccato* real-component).
- **Heterophony**: when the texture is homokinetic and homorhythmic but real components can be distinguished on the basis of intervallic disparities.

Each criterion corresponds to an integer, the weight inversely proportional to the magnitude, that is, heterochromy maps to 1, heterorhythm maps to 2, and so on. The last three criteria are significantly less common. In most analytical situations only the first two criteria would be used.

Taking into account the above criteria, the textural representation of the previous examples can be adapted in the following way:

- Schoenberg's *Wind Quintet*, bars 360-362: $[[2, 1, 2], 2]$ meaning $[[\text{Picc} + \text{Ob}, \text{Cl}, \text{Hn} + \text{Bn}], \text{'Heterorhythm'}]$.
- Schoenberg's *Pierrot Lunaire*, *Raub*, bars 16-17: $[[2, 1, 2], 1]$ meaning $[[\text{Fl} + \text{Cl}, \text{Vn} + \text{Vc}, \text{V}], \text{'Heterochromy'}]$.

5. IMPLEMENTED FUNCTIONALITY

The *Sonic Object Analysis Library* (SOAL) is an *OpenMusic* external library that we continuously develop at the University of Campinas. It is conceived to be useful for a range of analytical purposes and supports a top-down approach.

The library is modular as new functionality can be easily incorporated. SOAL allows the identification and analysis of musical structures by comparing the relative sonic qualities of a sequence of sonic setups and ultimately representing them by a *vector of relative complexity* (from 0 to 1, to the simplest to the most complex).

In this section we describe some of the tools we implemented in the SOAL package as part of the orchestration and texture analysis model.

5.1 The Weighted Number of Resources (WNR)

A segment employing four coordinated sound resources with another playing an independent part is represented by the list $[4, 1]$; to compute how many resources play in that setup, it suffices sum the list, obtaining 5. The first argument WNR requires is a sequence of such sums, or the rate of sonic resources per LSS.

The WNR function also requires a second argument and it refers to how many resources could possibly be played at any given setup or segment of the score or excerpt (determined by the SRI). In the example above, the setup $[4, 1]$ employing 5 resources could be part of a piece calling for a much larger instrumental effective. If that setup employs 5 resources of a total of 20 possible, it means that it designates only 25% of that total. The WNR function then calculates the ratios between used over total resources.

The function, however, implements a further treatment: the logarithms of the terms are used for weighting the quotients. This works for attenuating the ratios of setups where only a small number of resources is employed. The artifice helps to spot the higher points of dense setups on a WNR plot and it possibly models how we actually perceive those changes.

Note that the rate of resources per LSS (first argument) should not be greater than the total resources (second argument).

5.2 The relative voicing complexity

The relative voicing complexity, together with the relative setup complexity, is one of the two main functions that constitute the core of the model implementation. It revolves around the computation of two measurements, the agglomeration and dispersion and rates. They are used to compute intermediary values such as ratios of agglomeration and dispersion over the number of pairwise combinations of the total of sonic resources for the analysis. One of those ratios can be chosen to compute a weighted value depending on the criterion of dispersion and the number of criteria considered in the analysis. The result of that weighting is the foremost output of the function, being considered as the measure of 'relative voicing complexity' itself.

The ten output values returned by the function are:

- (i) agglomerations;
- (ii) negative agglomerations;
- (iii) dispersions;
- (iv) a experimental measure that we call textural trend;
- (v) successive ratios of agglomeration over the number of pairwise combinations of total sonic resources;
- (vi) successive ratios of dispersion over the number of pairwise combinations of total sonic resources;
- (vii) successive ratios of textural trend over the number of pairwise combinations of total sonic resources;
- (viii) agglomeration, dispersion, or textural trend weighted by criteria of dispersion and number of criteria;
- (ix) min-max normalised agglomeration, dispersion, or tendency weighted by criteria of dispersion and number of criteria; and
- (x) the number of pairwise combinations of the total of sonic resources of the analysis.

5.3 The relative setup complexity

The reasoning behind the relative setup complexity is that setup and voicing complexity are fundamentally interdependent because the number of real components (and pairwise combinations) that a setup can be structured with, is determined by how many sonic resources are available.

For instance, in a musical excerpt where only two sonic resources are being used, like in a duo, there will be a much greater chance that their parts will have a higher degree of independence than in a *tutti* of sixty instruments, which will not prevent the texture of such a duo from remaining limited despite the greater independence.

As a result, the estimations computed with 'relative voicing complexity' are only meaningful if they are made to interact with the ones obtained through the weighted number of resources.

We then use the WNR output to 'modulate' the values returned by RVC, in a similar way of what is done with the sound technique of 'frequency modulation'. It can be described by the equation

$$SRC = WNR + ((WNR \times RVC) \times p),$$

where p , is a weight, a percentage of the modulator RVC.

5.4 SOAL partitional analysis

The previous measurements are combined to form the 'SOAL partitional analysis' function, which is the main component of our implementation. It outputs every significant measurement proposed by the model. It expects as input parameters the combined list of arguments used for the RVC, RSC and WNR functions.

The function, accordingly, returns the number of outputs of the RVC function, together with the sorted and unsorted RSC output (*zipped* with bar numbers). The code for this function is given in Listings 1 below.

6. CONCLUSIONS

Our experimental model for computer-assisted analysis proposes a formal strategy for evaluating the role of orchestration and musical texture in structuring musical form. It works by looking at the symbolic level of the score's prescriptions and the acoustical level of the performed music. The analysis we put forward makes use of an application of the theory of partitions. We described how to collect and format data from the score, its mathematical background, how the implemented function works, and the kind of results it may return.

We already applied this model in a number case studies such as Anton Webern's *Symphony Op. 21*, Hermeto Pascoal's *Sinfonia em Quadrinhos* and Jean-Philippe Rameau's *Les Boréades*.

The audio branch of our will be the main point of investigation in future works which involves choosing, importing, developing and implementing appropriate audio features descriptors. The impact of an LSS' duration upon the overall musical perception should also be properly addressed in the next stages of our work. Based on our first experimental results, we hope that our tool will help musicologists shed new light in the role of orchestration in the structuration of musical form.

7. ACKNOWLEDGMENTS

This research is supported by the São Paulo Research Foundation (FAPESP) under grant 2018/04229-6 and the Brazilian National Council for Scientific and Technological Development (CNPq).

8. REFERENCES

Andrews, G. (1984) *The theory of partitions*. Cambridge University Press.

Berry, W. (1987) *Structural functions in music*. Dover, New York.

Gentil-Nunes, P. (2009) *Análise particional: uma mediação entre composição musical e a teoria das partições*. Ph.D. Thesis. Universidade Federal do Estado do Rio de Janeiro.

```
1. def soal_partitional_analysis (partitional_analysis,
2.                               total_resources,
3.                               number_of_criteria,
4.                               factor,
5.                               bar_labels,
6.                               mod_weight,
7.                               option):
8.
9.     partitions = texture_dispersion(partition_analysis)
10.    resources_per_segment = resources_per_lss(partitions)
11.    criteria_of_dispersion_per_segment = order_number(partition_analysis)
12.    voicing_complexity_multi_values = voicing_complexity(partitions,
13.                                                         resources_per_segment,
14.                                                         criteria_of_dispersion_per_segment,
15.                                                         number_of_criteria,
16.                                                         factor,
17.                                                         option)
18.    wnr = weighted_number_of_resources(resources_per_segment, total_resources)
19.
20.    # Outputs
21.    layers_per_segment      = number_of_real_components(partitions)
22.    max_total_pairw_comb    = voicing_complexity_multi_values['total_pairwise_combinations']
23.    agгло_per_segment      = voicing_complexity_multi_values['agglomerations']
24.    neg_agгло_per_segment   = voicing_complexity_multi_values['negative_agglomerations']
25.    disp_per_segment       = voicing_complexity_multi_values['dispersions']
26.    tendency_per_segment   = voicing_complexity_multi_values['tendency']
27.    aggllos_over_t_resources = voicing_complexity_multi_values['agglomerations_over_t_resources']
28.
29.    disps_over_t_resources  = voicing_complexity_multi_values['dispersions_over_t_resources']
30.    tends_over_t_resources  = voicing_complexity_multi_values['tendency_over_t_resources']
31.    weighted_option         = voicing_complexity_multi_values['weighted_option']
32.    norm_weighted_option    = voicing_complexity_multi_values['norm_weighted_option']
33.    rsc                     = relative_setup_complexity(wnr, weighted_option, mod_weight)
34.    sorted_rsc_and_bar_labels = sort_rsc_output_descendingly(bar_labels, rsc)
35.
36.    print(factor)
37.
38.    return {'wnr' : wnr,
39.           'lay' : layers_per_segment,
40.           'max' : max_total_pairw_comb,
41.           'aggl' : agгло_per_segment,
42.           'negg' : neg_agгло_per_segment,
43.           'disp' : disp_per_segment,
44.           'tend' : tendency_per_segment,
45.           'aggt' : aggllos_over_t_resources,
46.           'dist' : disps_over_t_resources,
47.           'tent' : tends_over_t_resources,
48.           'weig' : weighted_option,
49.           'neig' : norm_weighted_option,
50.           'rsc' : rsc,
51.           'srsc' : sorted_rsc_and_bar_labels}
```

Listings 1: Python code implementing the function SOAL Partitional Analysis, the main component of our model.