Formalizing Schoenberg's Fundamentals of Musical Composition through Petri Nets

A. Baratè, G. Haus, and L.A. Ludovico
LIM – Laboratorio di Informatica Musicale
Dipartimento di Informatica
Università degli Studi di Milano (Italy)
lastname@di.unimi.it

D.A. Mauro
Dept. of Computer &
Information Technology (CIT)
Marshall University (USA)
maurod@marshall.edu

ABSTRACT

The formalization of musical composition rules is a topic that has been studied for a long time. It can lead to a better understanding of the underlying processes, and provide a useful tool for musicologist to aid and speed up the analysis process. In our attempt we introduce Schoenberg's rules from Fundamentals of Musical Composition using a specialized version of Petri nets, called Music Petri nets. Petri nets are a formal tool for studying systems that are concurrent, asynchronous, distributed, parallel, nondeterministic, and/or stochastic. We present some examples highlighting how multiple approaches to the analysis task can find counterparts in specific instances of PNs.

1. INTRODUCTION

The goal of understanding the compositional processes behind the creation of music, or to mimic those processes by creating set of rules, has been pursued by many different approaches. If we focus our attention on modern attempts at building automatic composition systems, or model for the analysis of a musical piece, we can find a vast literature. For example, [1] presents different approaches aimed at encoding a musical piece. In [2] the author suggests an automatic system for score following that makes use of models in order to improve the prediction. More recent research on computer-based music modeling includes [3], addressing the relationship between programming systems and music theory and composition, and [4], that focuses on temporal dependencies modeling and applies it to polyphonic music generation and transcription.

In this context, we will explore the characteristics of Petri nets (PNs), a formal tool profitably used in Computer Science to study and describe systems that are concurrent, asynchronous, distributed, parallel, nondeterministic, and/or stochastic. When their application to the music field was first proposed, the following properties were considered: PNs are associated with a graphical form of notation that requires few symbols; they support hierarchical descriptions and the definition of macro-structures; they are able to describe music-objects processing, supporting timed representations and deterministic as well as non-deterministic models [5].

The use of a formal tool such as Petri nets for the encoding has a number of advantages with respect to other non-formal representation formats. For example, if we want to compare different "objects" encoded through Petri nets, we can rely on the theory behind, that allows to establish relationships based on the structure of the networks themselves (e.g., identify two network that share the same structure via an isomorphism) or investigate its mathematical properties.

The paper is structured as follows: Section 2 will briefly introduce the formalism of Petri nets and their applications to music, Section 3 will provide the musical background for our proposal, Section 4 will clarify why Petri nets are still a relevant tool to formalize musical composition rules, Section 5 will apply this formal tool to Schoenberg's theories, Section 6 will discuss some clarifying examples, and in Section 7 we will summarize the main strengths and weaknesses of the proposed approach.

2. AN OVERVIEW OF MUSIC PETRI NETS

A formal description of the general net theory by Carl Adam Petri would fall beyond the scope of the present paper. For details about this subject, please refer to works such as [6], [7] and [8]. For the sake of clarity, we will only summarize the key elements to let the reader understand the theoretical approach proposed in the following.

A PN is an abstract and formal model to represent the dynamic behavior of a system with asynchronous and concurrent activities. PNs are made of a combination of basic objects, falling in the following categories: *places*, *transitions*, and *arcs*. Usually represented in a graphical forms, the instances of such categories are drawn as circles, rectangles, and oriented lines respectively. *Places* and *transitions* are also referred to as *nodes*.

Arcs can have a number associated, called the *arc weight*.

PNs are not static models, rather they present an evolution from a state to another. The current state is indicated by place *marking*, namely by the number of tokens in each place. The dynamic evolution of a PN is determined by the following firing rules:

- A *transition* is enabled when all the incoming *places* of that transition present a number of tokens greater or equal to the weights of the corresponding incoming *arcs*, and - after the fire of the transition - the marking of all the output *places* will be less than or equal to their capacities;

- When a *transition* is enabled, its firing subtracts from the incoming *places* a number of tokens equal to the weights of the incoming *arcs*, and adds to each outgoing *place* a number of tokens equal to the weights of the corresponding outgoing *arc*.

The relationship between PNs and music has been investigated in a number of scientific works. To cite but a few examples, reference [9] represented a milestone on music description and processing through PNs; in [5], ScoreSynth, namely an experimental software tool for score synthesis through PNs, was presented; in more recent times, great emphasis was placed on the applicability of PNs to music analysis [10] and composition [11], even in real time environments [12].

In Music Petri nets (MPNs), that are a specialized extension of PNs, it is possible to associate music objects to *places*. A music object may be anything that could have a music meaning and that could be thought as an entity, either simple or complex, either abstract or detailed. Such an entity will present some relationship with other music objects. Consequently, with respect to traditional PNs, in MPNs the following cases can occur:

- A *place* can have no music fragment associated and no music fragment in input. In this way, it has only a structural function (e.g., a counter, a selector, a semaphore, etc.) in a given net topology, in accordance with the definition of *places* in ordinary PNs, where markings represent the state of the system;
- A *place* can host a music fragment that will be transferred to output places after the possible fire of the corresponding *transitions*. In this case, the music fragment will be delivered to output *places* after the processing operated by *transitions*;
- A *place* can receive a music fragment from either a single or multiple input *transitions*. If multiple fragments arrive simultaneously and/or a music object is already present, fragments are mixed to form a more complex music object, potentially available for outgoing *transitions*.

Moreover, in MPNs a *place* can be either enabled to play music objects or not. When an enabled *place* containing a music object receives a token, the fragment (either already present or transferred from other places) is played; when a non-enabled *place* hosts or receives music objects, its only function is to mix inputs, store music fragments and send them in output when *transitions* fire.

In MPNs *transitions* can host music algorithms (defined as abstract transformations); they can be used to process music objects in input, that are modified accordingly and then transferred in output.

As an example, one can create a simple net with a *place* that has an associated music object containing a single note (say, a C pitch) with the Play flag set to false, an output *transition* with an associated algorithm that creates a major scale in the key of the input note, and an output *place* that plays the objects thus modified (in this case, the C major scale). Then, the same net topology can be reused by changing the music object associated to the input *place*: e.g., if the original note is set to D, the D major scale is obtained; as another example, if a sequence of pitches is used as the input instead of a single note, the final result is a progression. Other examples will be provided in Section 5. For a formal description of MPNs, please refer to [13].

3. BACKGROUND

This research is based on two didactic textbooks dealing with music theory and composition authored by Arnold Schoenberg as the result of his teaching activity. In [14], the author provides simple models for beginners in composition. The main objectives of this syllabus are ear-training, the development of a sense of form, and the comprehension of the basic technique and logic of musical construction. In [15], a more advanced work that combines the analysis of masterworks with practice in the writing of music forms, Schoenberg expresses his vision on music composition.

Schoenberg's works have been conceived as a new method of achieving coordination of melody and harmony in order to make composing easier to students. It is worth noting that his pedagogical approach is not just one of theoretical speculation, but of exposing fundamental technical problems in composition and of showing how they can be solved in multiple ways. Great stress is laid upon the concept of variation, seen as the most important tool for producing logic in spite of variety.

As stated in the preface to [14], the reader should realize that such models show merely one way of approach to the technique of composing. But he or she should not in any case think that a composer would work in such a mechanical manner. What produces real music is solely and exclusively the inventive capacity, imagination, and inspiration of a creative mind – if and when a creator has something to express. A student should never write mere dry notes, at all times he should try to "express something".

4. RESEARCH QUESTIONS

As mentioned in Section 1, currently many algorithmic approaches and computer-based techniques are available to generate music or assist composers in their creative processes. In this context, why should we formalize Schoenberg's composition theories in terms of MPNs?

A first goal is to better understand Schoenberg's principles by adopting a more formal approach. In his didactic works, the Austrian composer tried to explain the way a music idea can be originated and developed, starting from the simplest structures (i.e. how to build two-measure motifs, or phrases, on a single harmony). Even if the subject is treated in a comprehensive way and through a number of clarifying examples, no notion of formalization or algorithm is explicitly given.

Moreover, as we will better explain in Section 5, the Petri-net formalization of a music excerpt is not unique, rather this formal tool allows to focus on different aspects of the score fragment and analyze its semantics from multiple angles. In this sense, the construction of the corresponding Petri net is not a mechanical operation, but it poses questions that lead to greater awareness about composition processes. The analytical valence of Petri nets applied to music scores has been discussed in other scientific works [16], highlighting both the advantages and the limits of this approach.

It is important to point out once more that the use of a formal tool presents a number of advantages. Regarding musicological tasks, the adoption of PNs in analyzing music can benefit from underlying theoretical tools. For instance, the expressive power of mathematical constructs can guide the analysis (e.g., "find all the PNs that are isomorphic to a certain test object").

After obtaining the expected formalization, results can be profitably used to generate (either manually or automatically) other music fragments sharing the same structure of the original or introducing new variants. The potential of Petri nets in a creative context has been explored in [12].

Realizing a corpus of PNs formalizing Schoenberg's approaches can be thought as a useful tool for comparative analysis. It can be envisioned an application built on a database constituted by formalizations of a collection of compositions, and the possibility to automatically analyze and compare those models with the rules presented in Schoenberg's work.

5. FORMALIZATION THROUGH PETRI NETS

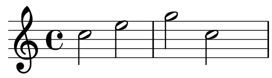
In the current section we will apply Petri nets to the formalization of the compositional processes suggested by Schoenberg. This implies the analysis and decomposition of complex musical objects, intended here as sequences of notes characterized by pitch and duration information, into simpler entities whose relationships and transformations can bring to the reconstruction of the original motif. In our approach, these relationships and transformations are formally represented by Petri nets.

It is important to point out that a given sequence of music symbols, even a very simple one, can be treated in multiple ways. For instance, let us consider a two-measure motif on a single harmony in the form of a broken chord, like the first example mentioned in [14] and shown in Figure 1a. Despite its apparent simplicity, it can be decomposed and represented in a multiplicity of ways, e.g.:

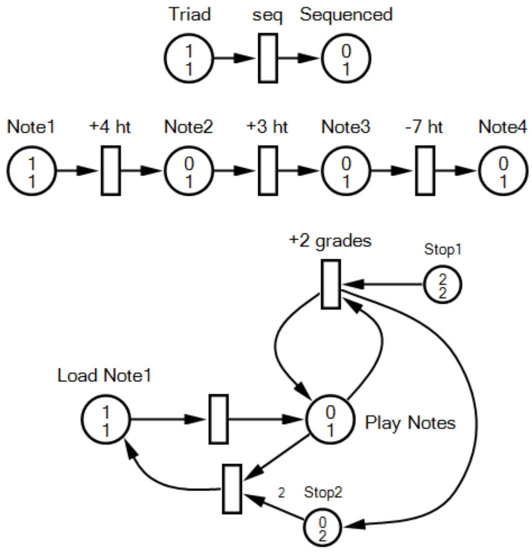
- Simply describing the whole motif as a major triad broken into a sequence of ascending half notes, with the last one transposed an octave below;

- Expressing the distance among notes in terms of halftone offset with respect to the previous pitch, resulting in the sequence [+4, +3, -7]. In this case, no recurrent pattern seems to emerge;

- Declaring such a distance in terms of number of grades over a given scale, e.g., in C major. Adopting this approach, resulting values are [+2, +2, -4]. In this case, the motif can be decomposed into two repetitions of the same ascending movement followed by the return to the original pitch. The resulting MPN supports such a behavior by introducing two additional places – graphically represented by smaller circles – that carry no music information and act as semaphores to enable/disable transitions.



(a) The original exercise.

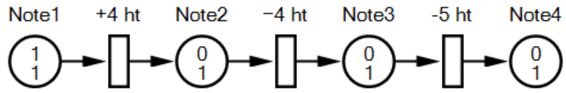


(b) Three alternative MPNs for its formalization, later referred to as case 1, 2, and 3 respectively.

Figure 1: A two-measure motif on a single harmony in the form of a broken chord (Exercise 1 excerpted from [14]).

Considering the same motif from different perspectives requires analytical skills that intrinsically foster a better comprehension of the composition process. For example, such a critical activity can unveil tonal relationships among notes, or recurrent patterns, or processes of re-elaboration and variation of the original thematic material. On the other hand, it introduces a great variability in formalization, since different musical parameters or mathematical relations can be emphasized, and the complexity of the formal description varies accordingly. In terms of Petri nets, the three cases mentioned above can be translated into as many net structures, where the musical operators associated to transitions are very heterogeneous in function and algorithmic complexity (see Figure 1b). For the sake of clarity, let us stress the difference between implementing a musical operator based on the concepts of interval and scale in order to decompose the chord on one hand (case 1), and another operator that takes a numerical value in input and produces a numerical output after a trivial sum operation on the other (case 2).

(a) The original exercise.



(b) A possible formalization through MPNs: a variant of case 2 in Figure 1b.

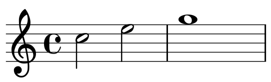
Figure 2: Exercise 2 proposed in [14].

6. EXAMPLES

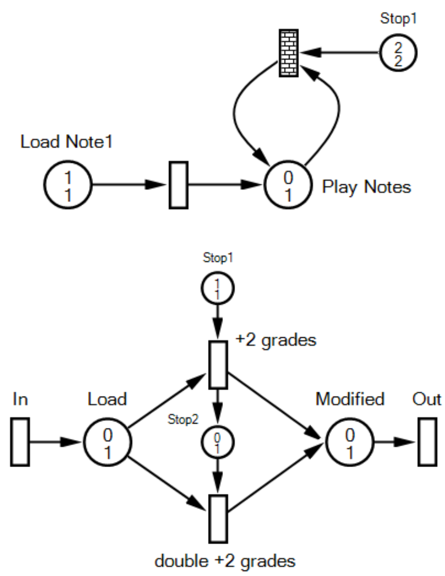
Starting from the models presented in the previous section, one can change the parameters associated to the transposing algorithms in the second net to generate all the basic examples proposed by Schoenberg, concerning both pitches and rhythmic values. For instance, adopting the halftone distance approach, Exercise 2 can be computed by changing the sequence of Exercise 1 from [+4, +3, -7] to [+4, -4, -5], as shown in Figure 2.

Similarly, Exercise 3 can be generated by removing the last *place*, and then adding a new step to the algorithm associated to transition T2 in order to double the note duration, as shown in Figure 3.

It is worth noting that, after the creation of MPN models, one can change the musical object associated to *place* Note1 and obtain the corresponding examples in several keys, as Schoenberg himself indicated as a good composition exercise in the preface of his manual. Needless to say, this behavior is strictly connected to the modeling approach in use: here it works since the musical operators associated with transitions alter halftone distances among pitches. Conversely, the third model presented in Figure 1b would transpose notes diatonically in the same scale, thus potentially changing intervals. For instance, if the musical object in *place* Note1 was a D instead of a C-pitched note, the sequence would become D–F–A–D, thus inverting the position of the major and minor third intervals in the motif. These "side effects" of PN models can be studied and profitably used to generate new motifs.



(a) The original exercise.



(b) A possible formalization through MPNs. Even if the conceptual approach can be seen as a variant of case 3 in Figure 1b, the resulting net topology, that now includes a subnet, is different.

Figure 3: Exercise 3 proposed in [14].

Figure 4: Exercise 13 and its variants proposed in [14].

Moving towards more complex and demanding examples, let us mention Schoenberg's exercises included in Part 2 – "Motive and motival features in two-measure phrases". Here the approach is different: in the discussion above the goal was to create a simple phrase by modifying a single starting note, while in the following we will focus on how to modify (more complex) musical fragments in order to generate different exercises.

For instance, let us consider Exercises 95, 96 and 97, proposed in [14] as variants of Exercise 13 and graphically listed in Figure 4. The corresponding MPNs are presented in Figure 5. The first net shows how the original phrase can be seen as the juxtaposition of four fragments, each having the duration of a half note. The second model can be obtained from the first one by changing the content of *places* Fragm2 and Fragm3, thus loading the original fragment and modifying it through suitable algorithms. In particular, the *places* of this model are represented as sub-nets, a syntactic possibility already supported in standard PNs. In this way, sub-nets can be delegated the representation of simpler models that implement the desired behavior. Exercises 96 and 97 share similar solutions.

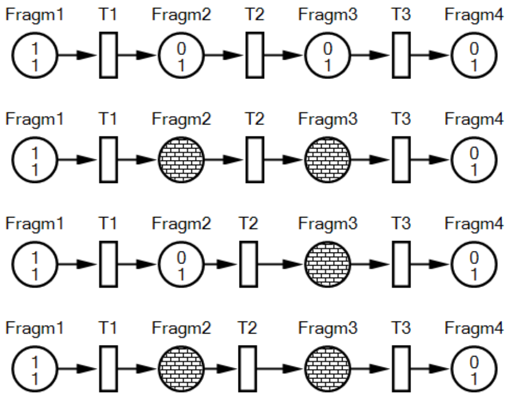


Figure 5: Possible MPNs that formalize the variants in Figure 4.

7. CONCLUSIONS

Currently, this work mainly represents a proof of concept about the applicability of MPNs to the creation and re-elaboration of motifs in music composition. Even if the mentioned examples are relatively simple and do not cover all the possibilities offered by MPNs, they should provide a broad idea of the proposed approach.

Since Schoenberg's exercises imply only basic transformations of music parameters, this didactic corpus provides learners with a valid test bed to acquire competences and skills in the use of formal description tools.

In our opinion, students in composition and musicology should be invited to create MPN models to represent Schoenberg's rules, in order to abstract from the notational aspects of the exercise and to concentrate on its intrinsic structure, on modification possibilities, on harmonic relationships, and on complex music forms intended as macro-structures.

We believe that, far from being a conclusive methodology for the complex subject of music analysis and composition, MPNs can provide a useful didactic method and a powerful artistic tool. Moreover, the creation of an extensive database of MPNs covering not only exercises but also music themes from literature can lead to more meaningful insights thanks to the possibility of performing automatic analysis on the collection.

8. REFERENCES

[1] M. Desainte-Catherine and A. Allombert, "Interactive scores: A model for specifying temporal relations between interactive and static events," *Journal of New Music Research*, vol. 34, no. 4, pp. 361–374, 2005.

[2] A. Cont, "Antescofo: Anticipatory synchronization and control of interactive parameters in computer music." in *International Computer Music Conference (ICMC)*, 2008, pp. 33–40.

[3] T. Anders and E. R. Miranda, "Constraint programming systems for modeling music theories and composition," *ACM Computing Surveys (CSUR)*, vol. 43, no. 4, p. 30, 2011.

[4] N. Boulanger-Lewandowski, Y. Bengio, and P. Vincent, "Modeling temporal dependencies in high-dimensional sequences: application to polyphonic music generation and transcription," in *Proceedings of the 29th International Coference on International Conference on Machine Learning*. Omnipress, 2012, pp. 1881–1888.

[5] G. Haus and A. Sametti, "Scoresynth: a system for the synthesis of music scores based on Petri nets and a music algebra," *Computer*, vol. 24, no. 7, pp. 56–60, 1991.

[6] C. A. Petri, "Introduction to general net theory," in *Net theory and applications*. Springer, 1980, pp. 1–19.

[7] J. L. Peterson, *Petri net theory and the modeling of systems*. Prentice Hall PTR, 1981.

[8] T. Murata, "Petri nets: Properties, analysis and applications," *Proceedings of the IEEE*, vol. 77, no. 4, pp. 541–580, 1989.

[9] G. Haus and A. Rodriguez, "Music description and processing by Petri nets," *Advances in Petri Nets 1988*, pp. 175–199, 1988.

[10] A. Baratè, G. Haus, and L. A. Ludovico, "Music analysis and modeling through Petri nets," *Computer Music Modeling and Retrieval*, pp. 201–218, 2006.

[11] A. Baratè, G. Haus, and L. A. Ludovico, "Petri nets applicability to music analysis and composition," in *Proceedings of the International Computer Music Conference'07 (ICMC 2007)*, 2007, pp. 97–100.

[12] A. Baratè, G. Haus, and L. A. Ludovico, "Real-time music composition through P-timed Petri nets," in *ICMC—SMC—2014 Proceedings, Athens 14-20 September 2014*, A. Georgaki and G. Kouroupetroglou, Eds., Athens, 2014, pp. 408–415.

[13] A. Baratè and G. M. Haus, "Structuring music information," *Music Navigation with Symbols and Layers: Toward Content Browsing with IEEE 1599 XML Encoding*, pp. 37–56, 2013.

[14] A. Schoenberg, *Models for beginners in composition: syllabus, music examples, and glossary*. G. Schirmer, Inc., 1942.

[15] A. Schoenberg, L. Stein, and G. Strang, *Fundamentals of musical composition*. Faber & Faber London, 1967.

[16] A. Baratè, G. Haus, and L. A. Ludovico, "Music analysis and modeling through Petri nets," in *Computer Music Modeling and Retrieval : 3rd International Symposium, CMMR 2005, Pisa, Italy, September 26-28, 2005 : Revised Papers*, ser. Lecture Notes in Computer Science, R. Kronland-Martinet, T. Voinier, and S. Ystad, Eds., vol. 3902. Berlin, Heidelberg: Springer-Verlag, 2006, pp. 201–218.