Representing procedural musical structures with an encoded functional grammar optimized for metaprogramming and machine learning

José López-Montes[*]
University of Granada
lopezmontes@correo.ugr.es

October 9, 2019

Abstract

We present GenoMus, a new model for artificial musical creativity based on a procedural approach, able to represent and learn the compositional techniques behind a musical score. The aim of this model is to build a framework for automatic creativity, easily adaptable to other domains beyond music. The core of GenoMus is a functional grammar designed to cover a wide range of styles, integrating traditional and contemporary composing techniques. Musical *genotypes* are defined as functional trees, able to generate musical scores described as *phenotypes*. To enable the maximal diversity of outputs, each process uses the same generic functional structure, no matter what time scale, polyphonic structure or additional characteristics are being employed. The goal of this highly homogeneous and modular approach is to simplify metaprogramming of genotypes, as well as maximize search space. Genotypes and phenotypes are encoded as normalized numeric vectors. This abstract representation of musical knowledge as pure numeric arrays is convenient for the application of different machine learning paradigms. The user interface developed for GenoMus is oriented to the exploration of augmented creativity, regardless of user expertise. However, a composer can create and alter manually genotypes and algorithms to modify automatic results. The system allows the implementation of user-defined processes, which will expand the procedures library.

Keywords: automatic musical composition · metaprogramming · procedural representation of music · artificial creativity · GenoMus

1 Introduction: composing composers

Research in artificial musical intelligence demand for formalized grammars of musical structures. Besides, a model of creative mind is required to operate these abstractions. Aesthetic criteria are extremely subjective, furthermore the details of every model of automatic composition impose, consciously or not, a limited search space. Delimiting these boundaries and setting evaluation principles can be seen as metacomposition, namely composing composers.

Composers' interest in musical language pervaded the 20th century aesthetics. Transformation and overcoming of well-established methods inherited from Romanticism led to post-tonal music. Linguistic structuralism applied to musical syntax stimulated relativization and consciousness of compositional procedures. Reversing the logic of this analytic knowledge, the methods of serial dodecaphonic music was the first step for the foundations of an inverse creative strategy: synthesize new styles from the predefinition of new rules.

---

[*]https://www.lopezmontes.es

Computer assisted composition enabled far more complex procedures, tedious or unfeasible to explore by hand. Eventually, composers began to use computers not only for analysis and calculation of complex structures, but for the automation of the creative processes themselves. That fact opened the door to a new approach to composition: a metamusical level characterized by modeling the processes within the minds of composers.

[Reflexiones sobre metacomposicion, el concepto de autoria y consideraciones pedagogicas y humanas de fondo.]

[Interes de la musica en el modelado de creatividad artificial - multidimensionalidad de la percepcion y analisis]

[Sobre la necesidad de usar el metanivel de los procedimientos antes que la partitura]

Many approaches to artificial intelligence applied to the automatic composition of music are modeled using scores as its data source...

[Complejidad del diseno de lenguajes de representacion musical en la composicion asistida por ordenador. Cita de algunas aproximaciones analogas.]

2   A functional grammar to represent musical procedures

2.1   Foundations and requirements

In any approach to artificial creativity, the representation system used is a condition that restricts the search space and determines aesthetic biases a priori, either consciously or unconsciously. On the other hand, the design of algorithms to generate music is ultimately an act of composition. With this in mind, our proposal seeks to be as open as possible, so that it can represent multiple styles. However, since the purpose of the project is not to imitate styles, but to create results of great originality and also serve as a tool for the *augmented creativity* of human composers, this grammar integrates contemporary compositional techniques such as recursive processes, controlled randomness, self-similarity, etc.

[]

[Conveniencia del paradigma de programacion funcional para la metaprogramacion. Antecedentes procedimientos compositivos como funciones (referencias a Haskell y LISP en la tradicion)]

[Necesidad de encontrar un medio de representacion adecuado a la automatizacion de analisis, a la flexibilidad de estilos, y a conjugar la programacion manual con la modularidad necesaria para las tareas automatizadas]

[Importancia de la codificacion como vector numerico como representacion abstracta de estructuras y resultados musicales para la aplicacion de tecnicas de IA.]

2.2   Musical genotypes and phenotypes

[Marco conceptual basico del paradigma genotipo-fenotipo (referencias de otros proyectos)]

[Definiciones estrictas de genotipo y fenotipo]

[Similitud con la programacion funcional: la pieza musical como funcion de funciones.]

2.3   Function types

[Tabla con los tipos de funciones. Figura ilustrando la estructura score/voice/chord]

2.4   Function libraries

2.5   Specimen data structure

[Tabla con la estructura de datos de cada par genofeno generado]

2.6   Minimal examples

3   Encoding of genotypes and phenotypes

- Proposito de la codificacion en el marco del machine learning

- Codificacion como vectores unidimensionales normalizados
- Universal valid format

$$P = \{x_n \in \mathbb{R} \mid 0 \leq x_n \leq 1\}$$

Multidimensional search space of all possible vectors with normalized parameters:

$$P^n$$

Each vector in P is mapped to a valid encoded genotype (that is a valid functional expression which generates an excerpt of music. Let G be the set of all possible valid encoded genotypes. To create genotypes, the map f works as a decision tree such as every vector $\mathbf{p} = (p_1, p_2, ..., p_n) \in P^n$ will be mapped to G:

$$f : \ P^n \to G$$

Most of times n doesn't match the number of required items to complete a valid encoded genotype. If $\mathbf{p}$ has more items than needed, they are ignored. If f needs more items than those supplied by $\mathbf{p}$, f reads $\mathbf{p}$ repeatedly from the beginning as a loop, until closing the valid encoded genotype. That implies that even vectors with a single value can be mapped to large functional expressions.

- Modularidad y posibilidad de manipulacion manual

4   Integrating traditional and contemporary techniques

- Entorno de trabajo con Max
- Ejemplo de pieza completa basado en Clapping music (La importancia de la autorreferencia)

```
s2V(                               // complete score: joins the 2 voices vertically
   vSlice(                         // voice 1: slices last cycle lost during phase lag
      vRepeatV(                    // phase G: F 13 times
         vRepeatV(                 // cycle F: E 8 times
            vConcatV(              // pattern E: C + D
               vConcatV(           // motif C: A + B
                  vMotifLoop(      // core motif A: 3 8th-notes and a silence
                     ln(           // note values
                        0.04,
                        0.04,
                        0.08),
                     lm(           // pitch (irrelevant for this piece)
                        65),
                     la(           // articulation
                        0.5),
                     li(           // intensities (last note louder for clarity)
                        50,
                        50,
                        120)),
                  vSlice(          // motif B: A with 1st note sliced
                     vAutoref(0),
                     q(1))),
               vSlice(             // motif D: C with 1st note sliced
                  vAutoref(3),
                  q(2))),
            q(8)),
         q(13)),
      q(8)),
   vConcatV(                       // voice 2: F + H
      vAutoref(7),
      vRepeatV(                    // phase H: G 12 times
         vSlice(                   // cycle G: cycle F with 1st note sliced
            vAutoref(7),
            q(1)),
         q(12))))

[ 1, 0.275535, 1, 0.534808, 1, 0.665631, 1, 0.665631, 1, 0.575462, 1, 0.575462, 1,
0.606798, 1, 0.27051, 0.51, 0.335614, 0.51, 0.335614, 0.51, 0.435614, 0, 1, 0.506578,
```

```
0.53, 0.53, 0, 1, 0.742646, 0.55, 0.553176, 0, 1, 0.36068, 0.56, 0.23, 0.56, 0.23,
0.56, 0.93, 0, 0, 1, 0.534808, 1, 0.304952, 0.57, 0, 0, 1, 0.416408, 0.58, 0.55,
0, 0, 0, 1, 0.534808, 1, 0.304952, 0.57, 0.854102, 0, 1, 0.416408, 0.58, 0.6, 0, 0,
0, 1, 0.416408, 0.58, 0.75, 0, 0, 1, 0.416408, 0.58, 0.84, 0, 0, 1, 0.416408, 0.58,
0.75, 0, 0, 1, 0.575462, 1, 0.304952, 0.57, 0.326238, 0, 1, 0.665631, 1, 0.534808, 1,
0.304952, 0.57, 0.326238, 0, 1, 0.416408, 0.58, 0.55, 0, 0, 1, 0.416408, 0.58, 0.82,
0, 0, 0, 0 ]
```

- Un ejemplo clasico con varias voces y conteniendo armonia, dinamica y articulacion
- Modularidad y posibilidad de manipulacion manual
- Ejemplos basicos de tecnicas habituales en CAC (movimiento browniano,
- Handling of recursive techniques (fibonacci, y extension del modelo a expresiones matematicas complejas)
- Puentes entra la notacion tradicional, la sintesis de sonido y la espacializacion
- Multimedia

5 Scalability

- Como conjugar universalidad de las expresiones con optimizacion para tener los vectores codificados con mayores diferencias entre si.
- Estrategias de caracterizacion de perfiles estilisticos
- El problema del mapeo de funciones y su extensibilidad
- Como establecer una base de datos de conocimiento
- Metricas automatizadas de ciertos resultados

6 Evaluation and evolution

- Esta propuesta de gramatica posibilita la implementacion y competencia de diferentes sistemas de evaluacion
- Design of evaluation methods as the crucial act of composition
- Objective vs. subjective evaluation
- What to learn?
- Evolutionary paradigm as the most promising

6.1 Evolutionary paradigm

- Analogia con el paradigma de los genotipos y fenotipos.
- El paradigma de evolucion abierta es mas adecuado a la creatividad artificial, por ser acumulativo y no plantear problemas de formato como las redes neuronales
- Planteamientos hibridos para tareas automatizables

7 Conclusions and future work

The artistic results of every algorithm designed for automated composition are strongly constrained by their own representation system of musical data. This paper presents GenoMus, a framework for the exploration of artificial musical creativity based on a generative grammar focused on the abstraction of creative processes as a metalevel of compositional tasks. We define musical genotypes as functional nested expressions, and phenotypes as the pieces created by evaluating these computable expressions. GenoMus' grammar is designed to ease the combination of fundamental procedures behind very different styles, ranging from basic to complex contemporary techniques, particularly those able to produce rich output from very simple recursive algorithms. At the same time, maximal modularity is provided to simplify metaprogramming routines to generate, assess, transform and categorize the selected musical excerpts. The system is conceived to maintain a long term interrelation with different users achieving individual musical styles. This proposed grammar can also be an

analytic tool, from the point of view of composition as computation, considering that the best analysis of a piece is the shortest precise description.

Cuestiones interesantes:

- ?Cuantas funciones primitivas son necesarias para generar musica en un determinado estilo? Hay innumerables expresiones funcionales diferentes que pueden generar la misma musica. Se puede deducir que la expresion funcional mas breve es el mejor analisis. Se pueden ver diferentes paradigmas de ensenanza/aprendizaje de la musica con estos modelos.
- ?Como puede hacerse ingenieria inversa automatizada para extraer estructuras desde la musica?

Ejemplo de citas [1, 2] y otra cita [3].

References

[1] Eleanor Selfridge-Field. Composition, combinatorics, and simulation: A historical and philosophical enquiry. In David Cope, editor, *Virtual Music: Computer Synthesis of Musical Style*, pages 187–220. The MIT Press, Cambridge, 2001.

[2] Jürgen Schmidhuber. Low-complexity art. *Leonardo, Journal of the International Society for the Arts, Sciencies and Technology*, 30(2):97–103, 1997.

[3] José López-Montes. Genomus: Prospección de técnicas de creatividad asistida por computadora mediante la metaprogramación de genotipos musicales. Master's thesis, Universidad Rey Juan Carlos, Madrid, 2013.