Computer-Representation of Music in the Research Environment

Geraint A. Wiggins

- Dolly Dhariwal

History of Music Recording

- In ancient times music was recorded for the purpose of reproduction by performers other than the composer and/or as *aides memoire* to the composer.
- Gradually as the study of music notation increased, It was used as a basis of research and teaching.
- Then in 19th century audio recording came into the picture which gave a detailed representation of the music signal but lacked the structural representation.
- In mid twenties computer were used to analyze score and audio data producing new data and storing them.

How to represent Music?

- Now the main issue here is to represent music in a way that it can be stored and made *interoperable* which means the data or processes can be exchanged.

- As the computer encoded data needs to be interpreted for using it the two attempts have been made to represent music in ways which do clearly conform to the principles of Knowledge Representation developed in the Artificial Intelligence (AI) community, in that their specifications explicitly include inference systems: a temporal logic for music (Marsden 2000) and Leibnitz, a general ontology for the representation

Difference between Knowledge Representation and Data Encoding

- Data encoding deals with the underlying mechanism of how the computer stores the data in bits.

- Whereas knowledge representation deals with what is the data stored and what does it mean, how complete it is, how accessible and manipulable it is, what can be inferred from it and how efficiently, in the sense of computational complexity.

Data Abstraction

- The primary purpose of the computer system is to manipulate data.

Reason:

- Data can be presented in the different ways, even the same data can be presented in different ways.
- Applications developed related purposes and even identical data can be expressed in different, non-interoperable encodings, and, very often, these encodings are built in to the software, so that one researcher's software can be used only on his or her particular choice of data encoding.

What it is?

- The idea is to identify and make explicit the properties of the data which are specifically relevant to a particular task or situation to be addressed by a program, and then supply library functions which will allow the program to manipulate that data *without explicit reference to its actual syntax*. This means that other data, represented in other ways, can then be used with the same program, conditional only upon the supply of appropriate library functions for that other data representation.
- **For example:** Smaill et al. represented *Syrinx* by Debussy in a rich representation using the words of the UK version of CMN

Example by Smaill

- they built a library of software functions for comparing, adding, subtracting, multiplying and supplying constants for data represented in this language
- they built a program to carry out a procedure of *paradigmatic analysis* as defined by Ruwet (1972), using the library defined over the data. The program was then used to produce an analysis of the piece similar to that of Nattiez (1975). This lead to complete representation.
- The next thing was to produce another representation of the same data using integers to represent both pitch and time values of the piece, along with another library, supplying the same operations as above for this new representation. The analysis program was then run on the new representation, using the new library, but *without any change to the analysis program itself*. The analysis resulting from this was compared with the original, and found to be indentical. Thus, the program was shown to be operable over an alternative representation encoding the necessary musical features of the data, notwithstanding the detail of the computer encoding and this lead to complete representation.

- represented a piece of music in a completely different pitch system using integers, in a way similar to the second representation above. The pieces are written for a scale with 24 divisions per octave instead of the more familiar 12 and the same alaysis program was applied without change.

- Conclusion:

  All that was necessary was the implementation of a library containing functions for the musically relevant operations on the data in terms of which the analysis program was specified.

- Abstraction Boundary:

  The theoretical line drawn between the operations of the analysis process and those which define the data types is called the *abstraction boundary*. A key issue in the design of such a system is the selection of the correct abstraction boundary.

  An abstraction boundary specified at the level of concepts and language naturally used in the process embodied in the program will generally be best for functions like *"transpose-pitch"* and "*add-integer".*

Key Features of Music Representation

- Information Content

two orthogonal8 dimensions on which a representation system for music is placed are:

1. Express Completeness:

is the extent to which the original content of the music represented may be retrieved and/or recreated from the represented knowledge

2. Structural generality:

is the extent to which information about musical structure in whatever sense and at whatever level) may be stored in the representation. For example, an audio signal is minimally structurally-general: significant work must be done to isolate.

the aim should be to increase scope for both expressive completeness and structural generality as far as

Focusing on a musical surface

- for a given musical task, there is a ‘lowest level of interest’ – that is, a level of detail below which further detail is unimportant.
- It is important to understand that the musical surface is only the *lowest* level of interest in the musical structures in a fully structurally general representation; that is, the structural information above the surface is still visible and available for reasoning.

Multiple hierarchies

- musical knowledge is *hierarchical*. A further refinement of this, emphasised by Smaill et al. (1993) and Lerdahl and Jackendoff (1983), is that musical knowledge is *multiply hierarchical*, in the sense that many different hierarchies may be constructed simultaneously on any given musical surface.

  These multiple hierarchies may be arbitrarily complex and arbitrarily intertwined however, for a given non-partitioned set of data at the musical surface, they must all join at some top point.

  One might characterise the information contained in such hierarchies as *syntactic*; Smaill et al. name them *constituent hierarchies* for this reason. Constituent hierarchies are characterised by the non-strict subset relation.

  Constituent hierarchies must carry formal and textual descriptions of the relationships they denote.

Non-hierarchical relations

- There are also relationships between hierarchies not based on structures but musical relationships such as similarity.
- Referential relations may be hierarchical, and possibly even cyclic.
- referential relation annotations should carry formal and textual descriptions of the relationship they denote, so that formation does not need to be repeatedly computed.

Description language(s)

- primitives and combination operators for describing relationships within and between musical surfaces.
- must be itself extensible, so as to allow the definition of new types of surface and their predications, and, indeed, new predications on existing surfaces.
- The extensibility must be conservative with respect to the soundness and completeness properties of the language.

Multiple domains of representation

- representations are in either the *acoustic* (or physical), the *auditory* (or perceived) or the *graphemic* (or notated) domain.
- It should be noted that there is a relationship, or at least an analogy, between the idea of Babbitt's domains.

Inference and operations

- each entity which is represented using the system must have a unique identifier, through which it may be named explicitly.
- Identity of identifier is then strictly equivalent to identity of identified, by definition. Identity, then, is not a semantic operation, but a syntactic one: two mentioned entities are identical if their syntactic identifiers are the same and not otherwise.

Continuous features

- Much past work in music representation has chosen a musical surface which avoids the difficult area of symbolic representation of values which vary continuously in time.
- The continuity is difficult to handle for several reasons, but most formally because the meaning of equality between time-variant values is difficult to define.

Representing uncertainty

In the graphemic domain, the provenance of a source may be uncertain, or editions may not be definitely attributed, or the ink on a page may be smudged; also, composers may deliberately notate chance operations in scores, and this must be represented semantically.

In the acoustic domain, the output of an algorithm for detecting the onset of notes may be probabilistic, and so may yield a probability distribution over a short time period, rather than an exact onset time.

In the auditory domain, different listeners may infer different structures from the same acoustic stimulus.

Conclusion

- In this paper, we have presented general requirements for music representation, at various levels ranging from the level of effective engineering to the philosophy of knowledge representation. Any general representation for music will require all of these features if it is truly to serve the needs of the computer-based musicologists and musicians of the future.