ICMC . Tokyo, 1993.

Deville, Y. & Van Hentenryck, P. (1991). An Efficient Arc Consistency Algorithm for a Class of CSP
   Problems. *Proceedings of the IJCAI* . Sydney, 1991.

Lurson,M. Duthen, J & Rueda, C. (1992). The PatchWork Reference Manual. *IRCAM* , 1992.

Mackworth, A. (1977). Consistency in Networks of Relations. *Artificial Intelligence* . 99-118.

Ovans, R. (1990). Music Composition as a Constraint Satisfaction Problem. *Proceedings of the ICMC* .

Schiex,T. (1992). Possibilistic Constraint Satisfaction Problems or "How to Handle Soft Constraints".
   CERT-ONERA . Personal e-mail communication.

Schottstaedt, B. (1983). Pla: A Composer's Idea of a Language. *Computer Music Journal* . 7(1).

Sidebotton, S & Havens, W. (1991). Hierarchical Arc Consistency Applied to Numeric Processing in
   Constraint Logic Programming. *CSS-IS TR 91-06* . Simon Fraser University, Burnaby, Canada.

Steele, G. (1990). *Common Lisp: The Language* . Digital Press.

Taube, H. (1991). Common Music: A Music Composition Language in Common Lisp and CLOS.
   Computer Music Journal . 15(2).

Incremental evaluation in a musical hierarchy

M. DESAINTE-CATHERINE

K. BARBAR and A. BEURIVÉ

LaBRI[1]

Université Bordeaux I

351, cours de la Libération

33405, Talence Cedex

France

myriam@labri.u-bordeaux.fr

Abstract

The work we present in this paper is a formalism of a dynamic computational model in a hierarchy. We are interested in representing musical hierarchies and bindings of characteristics (such as the mode, measure, tempo, duration, key, etc.) within them in order to provide the composer a means to verify the consistancy of the piece during the compositional process. The model transfers any modification from the composer to the representation in an incremental way, without computing again the whole hierarchy.

1   Introduction

The complexity of a musical piece can be organized in a hierarchical way based on its temporal structure. Musical characteristics (such as the mode, measure, tempo, duration, key, etc.) can be defined at any point of the hierarchy (that is any sub-piece). These characteristics are then bound together according to the temporal structure and the musical rules imposed by the composer. We are interested in representing musical hierarchies and bindings of characteristics within them in order to provide the composer a means to verify the consistancy of the piece during the compositional process.

Our work may be situated between constraints propagation techniques and hierarchical representations à la Balaban. We are interested in designing the representation and the computation model which is appropriate to it. From our point of view, a musical piece is an object that is composed of several dimensions. Classic dimensions are time, frequency, timbre and volume. The variations of the values in these dimensions are not independent from each other. The result of a musical analysis is exactly a set of correlations between variations within a single dimension and between different dimensions. In order to formalize those correlations, we define several relational operators which are dedicated to specific dimensions. The set of values in each dimension can then be structured in a hierarchical way using these operators. Hierarchical way means that the object representing the structure is not always a simple tree, but a directed acyclic graph (see the notions of shared occurrencies and repetitions of Mira Balaban (Balaban 1993)). The originality of this work relative to the others based on hierarchical representations is the addition of a semantics to the hierarchy. This semantics provides a very sound way to represent

---

[1] Laboratoire Bordelais de Recherche en Informatique – Unité de Recherche Associée au Centre National de la Recherche

some correlations in order to verify or apply them. The obvious limit of this representation is that it does not compute the correlations that are not based on an operational structure. We believe, however, that there almost always exists a structure underlying every kind of correlations.

Our initial study concerns the time dimension. Time relational operators have been widely studied (in particular, the musical concatenation of Mira Balaban (Balaban 1991), relations of Allen (Allen 1983) and their application to music by Alan Marsden (Marsden 1994)). The two operators *concatenation* and *superimposition* provide a simple model with an acceptable power of expression. We first investigated the static aspect of the model (see (Barbar, Desainte-Catherine, Miniussi 1993) and (Barbar, Desainte-Catherine 1992)) in the following way. We first transform a musical equational program defining the structure of a musical piece into a derivation tree according to an attribute grammar. This derivation structure is then considered as a data structure which represents the musical hierarchy. Each attribute in the derivation tree represents a musical characteristic and the associated semantics represents the musical rules binding these characteristics. The evaluation step computes a solution (the values of all characteristics of the hierarchy), if it exists.

This previous work provides a very sound model but is insufficient in the context of an interactive compositional environment. A dynamic model is needed. This model must transfer any modification from the composer to the representation in an incremental way, without computing again the whole hierarchy.

The work we present in this paper is a formalism of a dynamic computational model in a hierarchy. The data representation is the same than the previous one. Only the computational model has changed. This model manages modifications (giving a value to a characteristic, changing a value of a characteristic, modifying the hierarchy itself by substituting one sub-piece by another) and maintain the overall consistency of the piece. The first two operations necessitate the propagation of the modifications of a characteristic in any direction in the hierarchy. The last operation implies the management of several hierarchies at the same time. Our formalism is no longer based on attribute grammars, but on systems of equations.

In section 2, we present the syntactic aspect of a musical hierarchy. It is represented by an equational program which is given with a set of syntactic equations. We give in section 3 the musical systems or the relations between characteristics attached to nodes of a hierarchy in terms of sets of equations on these characteristics. We define the solution of a musical system in section 4. An incremental strategy for the determination of the solution is given in section 5. The section 6 contains our conclusion.

2   Equational Program

2.1   Temporal Operators

Let us denote by (t, d, f, s, v) an event, where t is the beginning time, d the duration, f the pitch, s the sound and v the volume of the event. Let $e_1 = (t_1, d_1, f_1, s_1, v_1)$ and $e_2 = (t_2, d_1, f_2, s_2, v_2)$ be two events. The operators of concatenation, denoted by ., and superimposition, denoted by |, are defined by:

$$e_1.e_2 \Rightarrow t_1 + d_1 = t_2; \quad e_1|e_2 \Rightarrow t_1 = t_2, d_1 = d_2.$$
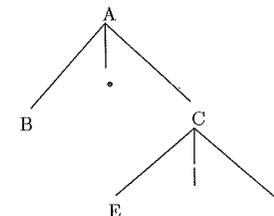
2.2   Syntactic Equations

The temporal structure of a piece is defined by the means of syntactic equations whose forms are given by general syntactic equations. For example, let us define the two syntactic equations that will be used in this paper: $e_{.} :: A = B . C$, $e_| :: A = B \mid C$ where A, B and C represent musical pieces.

The equation $e_{.}$ means that the piece C is concatenated to the piece B, i.e. it starts exactly when B ends. The piece A is the concatenation of B and C. The equation $e_|$ means that the two pieces B and C start and end at the same time. The piece A is the superimposition of B and C.

2.3   Equational Program or Hierarchy

An equational program is a set of syntactic equations. We will only consider equational programs which can be represented by a tree i.e each symbol of musical piece occurs at most one time in the left hand side

Example: Let $P = \{e_1 : A = B . C, e_2 : C = E \mid F\}$. The tree associated with the equational program P is:

3   Musical Systems

The semantics of a hierarchy is built by a kind of union (called a *cartesian union*) of the musical systems of each syntactic equation composing the equational program representing the hierarchy. In what follows, we will only study the case of the characteristics measure and duration which will be denoted, respectively, by m and d.

3.1   Musical Systems associated with Syntactic Equations

We introduce the concept of musical systems associated with a syntactic equation with the two following examples. The reader interested in the formal definition can refer to (Barbar, Desainte-Catherine, Beurivé 1994). We give two examples of musical systems for the characteristics measure.

Example 3.1 *Let $S_{e,m} = \{s_1, s_2, s_3, s_4\}$ be the set of equations systems associated with the syntactic equation $e :: A = B . C$, where*

$$(s_1) \begin{cases} m(B) \neq \varepsilon, m(C) \neq \varepsilon, \\ m(B) \neq m(C), m(A) = \varepsilon, \\ (m(A), m(B), m(C)) \in Domain(m)^3 \end{cases} \quad (s_2) \begin{cases} m(A) \neq \varepsilon, m(A) = m(B), \\ m(A) = m(C), \\ (m(A), m(B), m(C)) \in Domain(m)^3 \end{cases}$$

$$(s_3) \begin{cases} m(A) = \varepsilon, m(B) = \varepsilon, \\ (m(A), m(B), m(C)) \in Domain(m)^3 \end{cases} \quad (s_4) \begin{cases} m(A) = \varepsilon, m(C) = \varepsilon, \\ (m(A), m(B), m(C)) \in Domain(m)^3 \end{cases}$$

The value ε denotes a measure that is not constant. The musical meaning of this musical system is the following:

- *When two parts have different measures, the measure of their concatenation is not constant.*

- *When two parts have the same measure m, their concatenation has also the measure m.*

- *If a part A has got a measure m, every subpart of A gets the measure m.*

Let us now define a simple system for the measure and the superimposition operation: two parts that are superimposed have the same measure. The set of systems is reduced to the following equation system:

$$(s_5) \begin{cases} m(A) = m(B), m(A) = m(C), \\ (m(A), m(B), m(C)) \in Domain(m)^3 \end{cases}$$

Example 3.2 *The following musical system for the measure involves also the characteristics duration, denoted by d. Let $S_{e,m} = \{s_8, s_9, s_{10}\}$ be the set of equations systems associated with the syntactic equation $e :: X = Y . Z$, where*

$$(s_8) \begin{cases} m(Y) \neq m(Z), \\ d(Y) > d(Z), \\ m(X) = m(Y) \\ (m(X), m(Y), m(Z)) \in Domain(m)^3 \end{cases} \qquad (s_9) \begin{cases} m(Y) \neq m(Z), \\ d(Y) < d(Z), \\ m(X) = m(Z) \\ (m(X), m(Y), m(Z)) \in Domain(m)^3 \end{cases}$$

$$(s_{10}) \begin{cases} m(X) = m(Y) \\ m(X) = m(Z), \\ (m(X), m(Y), m(Z)) \in Domain(m)^3 \end{cases}$$

3.2 Musical Systems associated with Equational Program

We have defined the formal object representing musical systems involving syntactic equations and sets of characteristics. Now, from small pieces which are those musical systems, let us define how to build the musical system which is associated to a whole equational program. For this purpose, we introduce the *cartesian union* operation which simplifies the final definition.

Definition 3.3 *Let E_1 and E_2 be two finite sets of sets. The cartesian union of E_1 and E_2 is defined by: $E_1 \uplus E_2 = \{e_1 \cup e_2 | e_1 \in E_1, e_2 \in E_2\}$.*

Example 3.4 *Let $E_1 = \{\{1,2,3\}, \{4,5\}, \{6\}\}, E_2 = \{\{a,b\}, \{c\}\}$, then*
$E_1 \uplus E_2 = \{\{1,2,3,a,b\}, \{1,2,3,c\}, \{4,5,a,b\}, \{4,5,c\}, \{6,a,b\}, \{6,c\}\}$.

Let be P an equational program. We denote by e any syntactic equation in P and by S_e the set of equations systems associated with e. Let be Γ a set of characteristics. Then, the set of equations systems associated with the equational program P is:

$$S = \biguplus_{e \in P} (\biguplus_{\gamma \in \Gamma} s_{e,\gamma})$$

where $s_{e,\gamma}$ is the musical system associated with the equation e for the characteristics γ.

Example 3.5 *Let P be the equational program $\{e_1 : A = B.C, \ e_2 : C = E|F\}$. Then, we have:*

- *the musical systems associated with the equations e_1 and e_2 are:*

  $-s_{e_1,m} = \{s_1, s_2, s_3, s_4\}$, the equation systems given in 3.1
  $-s_{e_1,d} = \{s_5\} = \{d(A) = d(B) + d(C), d(A) \geq 0, d(B) \geq 0, d(C) \geq 0\}$
  $-s_{e_2,m} = \{s_6\} = \{m(C) = m(E), m(C) = m(F), (m(C), m(E), m(F)) \in Domain(m)^3\}$
  $-s_{e_2,d} = \{s_7\} = \{d(C) = d(E), d(C) = d(F), d(C) \geq 0, d(E) \geq 0, d(F) \geq 0\}$
  $-s_{e_1} = s_{e_1,m} \uplus s_{e_1,d} = \{s_1 \cup s_5, s_2 \cup s_5, s_3 \cup s_5, s_4 \cup s_5\}$
  $-s_{e_2} = s_{e_2,m} \uplus s_{e_2,d} = \{s_6 \cup s_7\}$

- *the musical systems associated with the program P are:*

  $$S = \biguplus_{e \in \{e_1, e_2\}} (\biguplus_{\gamma \in \{m,d\}} s_{e,\gamma}) = s_{e_1} \uplus s_{e_2} = \{s_1 \cup s_5, s_2 \cup s_5, s_3 \cup s_5, s_4 \cup s_5\} \uplus \{s_6 \cup s_7\}.$$

  $$S = \{s_{1,5,6,7}, \ s_{2,5,6,7}, \ s_{3,5,6,7}, \ s_{4,5,6,7}\}$$

  where for all $i,j,k,l : s_{i,j,k,l} = s_i \cup s_j \cup s_k \cup s_l$. As example:

  $$s_{2,5,6,7} = \begin{cases} m(A) \neq \varepsilon, m(A) = m(B), m(A) = m(C), (m(A), m(B), m(C)) \in Domain(m)^3 \\ d(A) = d(B) + d(C), d(A) \geq 0, d(B) \geq 0, d(C) \geq 0 \\ m(C) = m(E), m(C) = m(F), (m(C), m(E), m(F)) \in Domain(m)^3 \\ d(C) = d(E), d(C) = d(F), d(C) \geq 0, d(E) \geq 0, d(F) \geq 0 \end{cases}$$

4 Musical Equational Program

A musical equational program (MEP) is the main object of our model. It represents the state of the composing process at one time, that is:

- the state of the hierarchy, which is represented by a set of syntactic equations,

- the state of the musical system which is associated to the current hierarchy,

- the set of all the assignments of some parts characteristics that have either been given by the composer or either been computed from the musical system.

Definition 4.1 *A musical assignment is an equation of the form $c(A) = v$ where c is a characteristic symbol, A is a piece symbol and v a value in the domain of c.*

Example 4.2 *Let be the following MEP : $\prec \{e_1 : A = B.C, \ e_2 : C = E|F\}, \{s_{e_1}, s_{e_2}\}, \{m(A) = 3/4, d(E) = 10\} \succ$, where s_{e_1} and s_{e_2} are the musical systems of example 3.5.*

4.1 Solutions of musical equational systems

Intuitively, the solution of a MEP is the intersection of non empty solutions of all musical systems associated with the equational program.

Definition 4.3 *Let $\prec P, S, G \succ$ be a MEP. Let $sol(s)$ be the set of all the solutions of a system $s \in S$, each solution being given by a set of assignments of the form $c(A) = v$ where c is a characteristic and A is a symbol representing musical piece. Let be $sol_G(s) = \{\sigma \in sol(s) / G \subset \sigma\}$ and $sol_G(S) = \bigcup_{s \in S} sol_G(s)$. The solution of $\prec P, S, G \succ$ is the set of assignments $\bigcap_{\sigma \in sol_G(S)} \sigma$. So we will write $\prec P, S, G \succ \vdash \bigcap_{\sigma \in sol_G(S)} \sigma$.*

The solution of $\prec P, S, G \succ$ is the empty set if G does not constitute a part of some solution of S. In that case, the system $\prec P, S, G \succ$ is said to be **invalid** (or not consistant).

Example 4.4 *Let us consider the MEP $\prec P, S, G \succ$, where S is given in example 3.5 and $G = \{m(A) = 3/4, d(E) = 10\}$. Then, we have:*

$sol_G(s_{2,5,6,7}) = \{\{m(A) = m(B) = m(C) = m(E) = m(F) = 3/4, d(E) = d(F) = d(B) = 10, d(A) = d_A, d(C) = d_C\}/d_A - d_C = 10\}$, *(it contains an infinite number of solutions)*
$sol_G(s_{1,5,6,7}) = sol_G(s_{3,5,6,7}) = sol_G(s_{4,5,6,7}) = \emptyset$.
Thus, the set of all solutions of S is $sol_G(S) = \bigcup_{s \in S} sol_G(s) = sol_G(s_{2,5,6,7})$

and the solution of $\prec P, S, G \succ$ is $\bigcap_{\sigma \in sol_G(S)} \sigma = \bigcap_{\sigma \in sol_G(s_{2,5,6,7})} \sigma =$

$\{m(A) = m(B) = m(C) = m(E) = m(F) = 3/4, d(E) = d(F) = d(B) = 10\}$.

Definition 4.5 *A MEP $\prec P, S, G \succ$ is saturated if $\prec P, S, G \succ \vdash G$.*

Example 4.6 *The MEP $\prec P, S, G \succ$ given in the previous example is not saturated because the assignments $m(B) = m(C) = m(E) = m(F) = 3/4, d(F) = d(B) = 10$ do not belong to G. On the contrary, $\prec P, S, G \cup \{m(B) = m(C) = m(E) = m(F) = 3/4, d(F) = d(B) = 10\} \succ$ is saturated.*

We note that here we calculate the solutions of the musical system associated with the whole hierarchy (or tree) with respect to all assignments given by the composer. An interesting way for the determination of the solutions is the elimination of all invalid musical systems each time the composer gives an assignment.

5   The Incremental Strategy

The incremental evaluation on a musical equational program is the computation of a solution step by step. It consists of the computation of assignments which are deduced by the musical systems with initial assignments which are given by the composer. Let $\prec P, S, G \succ$ be a saturated musical equational program. A slight modification of $\prec P, S, G \succ$ implies modification of the solution. The incremental strategy consists of the computation of the new solution by modifying the old one without computing again the whole solution. Now, we give the principle of the incremental evaluation on a hierarchical structure. The nodes of the tree are denoted by $u_1, ..., u_n$. We start with a saturated MEP $\prec P, S, G \succ$ associated with the tree. Then we add a new assignment g_i on a variable of the sub-system associated with the node u_i. Then, in order to saturate $\prec P, S, G \cup g_i \succ$ i.e to calculate the solution G' (s.t $\prec P, S, G \cup g_i \succ \vdash G'$), we proceed as follows:

- we calculate the solution of the sub-system at the node u_i w.r.t the assignments $G \cup g_i$;

- we propagate to the father and the sons of the node u the assignment of the solution which concerns variables in their sub-sytems and so on.

We give a recursive fonction *sol* for the computation of the solution of the musical equational program. This is represented in the following schema:
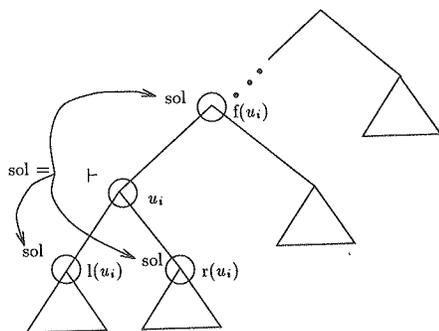
It shows the decomposition of the function *sol* at the node u_i in a resolution ($\vdash$) of the musical systems at u_i and three recursive calls to the father, left son and right son of u_i which are denoted respectively, by $f(u_i)$, $l(u_i)$ and $r(u_i)$, on the figure. The definition of the relation $\vdash$ can be given by an automata (see (Barbar, Desainte-Catherine, Beurivé 1994))

6   Conclusion

We have presented a model for representing musical pieces without repetitions by the means of a temporal hierarchy. Moreover, this model provides a way to compute automatically some musical characteristics by using equations systems and values that are given by the composer. The result is a very efficient software based on automatas solving the systems. Now, the power of expression of the model is too restrictive. It is necessary to integrate repetitions and several concurrent structurations. Those extensions will complexify the model and improve its efficiency. Now, we are currently working on the concept of abstraction of musical hierarchies for representing musical forms and items in the context of an interface for the composer. The model would then be useful for analyzing too. At last, the study of operators on other musical dimensions will increase again the power of structuration of the composer (and the power of expression of an analysis).

References

Balaban, M. (1991). Music structures: the Temporal and Hierarchical Aspects in Music. Technical report FC-035 MCS-327, Ben-Gurion University of the Negev.

Balaban, M. & Samoun, C. (1993). Hierarchy, Time and Inheritance in Music Modelling. *Languages of design*, 1(3).

Allen, J.F. (1983). Maintaining Knowledge about Temporal Intervals. *Communications of the ACM*, 26(11), pp.832-843.

Marsden, A. (1994). The Representation of Temporal Relations in Music writing in progress.

Barbar, K. & Desainte-Catherine, M. (1992). Using attribute to find solutions for musical equational programs. Technical report 92-77, LaBRI, Univ BX-1.

Barbar, K. Desainte-Catherine, M. & Miniussi, A. (1993). The semantics of a musical hierarchy. *Computer Music Journal*, 17(4), pp 30-37.

Barbar, K. Desainte-Catherine, M. & Beurivé, A. (1994). Incremental Resolution of Musical Systems. Technical report 964-94, LaBRI, Univ BX-1.