AUTOMATIC SYNTHESIS STRATEGIES FOR OBJECT-BASED DYNAMICAL PHYSICAL MODELS IN MUSICAL ACOUSTICS

Giovanni De Sanctis, Augusto Sarti, Stefano Tubaro

Dipartimento di Elettronica e Informazione
Politecnico di Milano - Piazza L. Da Vinci 32, 20133 Milano, Italy
desancti/sarti/tubaro@elet.polimi.it

ABSTRACT

Current physics-based synthesis techniques tend to synthesize the interaction between different functional elements of a sound generator by treating it as a single system. However, when dealing with the physical modeling of complex sound generators this choice raises questions about the resulting flexibility of the adopted synthesis strategy. One way to overcome this problem is to approach it by individually synthesizing and discretizing the objects that contribute to the generation of sounds. In this paper we address the problem of how to automatize the process of physically modeling the interaction between objects, and how to make it dynamical. We will show that this can be done through the automatic definition and implementation of a topology model that adapts to the contact and proximity conditions between the considered objects.

1. INTRODUCTION

In the past few years the interest in WDFs has grown a great deal, as the research in musical acoustics started to turn toward synthesis through physical modeling. This is in part due to the fact that WDFs are able to preserve many properties of the analog systems that they model, with particular reference to passivity and losslessness [1]. This renewed interest in WDFs, however, is also due to the popularity gained in the past decade by Digital WaveGuides (DWGs) [5], which can be seen as close relatives of WDFs.

WDFs are able to incorporate nonlinear elements by connecting their wave version to the adapted port of the structure. In addition to resistive nonlinearities (frictions), WDF can also accommodate reactive nonlinearities (e.g. nonlinear stiffnesses), or more general nonlinear elements with memory [4]. In order to do so, we define new waves with respect to which the description of the nonlinear elements becomes memoryless. The wave transformation is performed by dynamic multiport junctions and adaptors with memory that can be proven to be non-energetic [4]. Such multiport junctions are called *dynamic adaptors*, as their reflection coefficients are, in fact, reflection filters.

We recently showed that it is possible to use such principles in order to model physical structures in a block-wise fashion through a systematic and automatic procedure. Working in a block-wise fashion means constructing a number of individually synthesized blocks and connecting them together using a properly defined interconnection network. In this paper we show that this automatic procedure can be implemented for dynamically changing topologies, and in a very cost-effective fashion.

2. MACRO-ADAPTORS IN WAVE DIGITAL STRUCTURES

A physical structure (mechanical or fluidodynamical) can be described by an electrical equivalent circuit made of lumped or distributed elements. The equivalence can be established in a rather arbitrary fashion as a physical model is always characterized by a pair of across-through variables (e.g. voltage-current, force-velocity, pressure-flow, etc.). Wave Digital (WD) filters and WD structures [3] represent a well-consolidated solution to the problem of physically modeling structures made of lumped elements and their interconnection topology. Solutions are described in [2]. We consider here the problem of how to model the topology of interconnection in an automatic fashion. The key element for this purpose is the Macro-Adaptor (MA), which constitutes a generalization of scattering junctions in DWG and adaptors in WDF.

An N-port macro-adaptor is a non-energetic N-port with a twofold role:

- to implement the laws of continuity between interacting subsystems;
- to model changes in the wave reference resistances (wave scattering).

In principle, it is always possible to directly implement the scattering filters that constitute a macro-adaptor (direct-form synthesis). This approach, in fact, leads to a rather efficient implementation but requires a custom procedure that cannot be easily automatized. On the other hand, we can devise rather simple rules to decompose a macro-adaptor into an interconnection of elements chosen from a very limited collection of blocks, since a macro-adaptor is obtained by interconnecting together multi–port parallel and/or series adaptors. An M-port parallel (series) adaptor can always be implemented by connecting together $M-2$ parallel (series) 3-port adaptors (see Fig. 1). two MA ports may be directly connected with each other if their port reference resistances are the same;

1. a non-adapted port must be connected to an adapted one;
2. the NLE must be connected to an adapted port;
3. loop of adaptors must be avoided.

Given such rules, and considering that an adaptor can only have one adapted port, it should be quite clear that *a macro-adaptor can only accommodate up to one nonlinear element*. Should more than one NLEs be present in the system, they need to be combined into a multi-port NLE. Methods for constructing such a multi-port system are still under study.

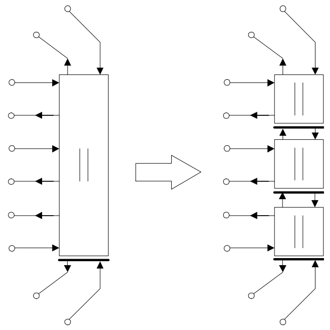

Figure 1: An M-port parallel (series) adaptor can always be implemented by connecting together $M-2$ parallel (series) 3-port adaptors

There is no reason to have adaptors connected in loop, because it is always possible to construct an equivalent structure without loops using a smaller number of adaptors. Moreover, a structure with loops does not have a spare adapted port and, still worse, lose its computability due to the rising of an indirect dependence between reflected and incident wave to an adapted port.

A further extension [4], defines a more general family of digital waves, which allow us to model a wider class of nonlinearities, such as nonlinear capacitors. This generalization of WDF principles include dynamic multiport junctions and adaptors, which synergetically combine ideas of nonlinear circuit theory (mutators) and WDF theory (adaptors). It can be easily proven [4] that, under mild conditions on their parameters, such multiport adaptors are nonenergetic, therefore the global stability of the reference circuit is preserved by the wave digital implementation. For this reason, such multiport junctions can be referred to as dynamic adaptors.

Any 3-port parallel (series) adaptor can be implemented as a standard parallel (series) WDF adaptor, whose ports are connected to 2-port scattering cells (two-port adaptors). This situation is shown for the series adaptor in Fig. 2, where $\tilde{\Gamma}_3(z)$ is the delayed part of the reflection filter of the adapted port.

It is important to notice that, although it is to construct the dynamic scatterers in such a way to avoid local instantaneous reflections between the two new elements, we cannot modify global adaptation conditions. Any 2-port scatterer can always be "pushed

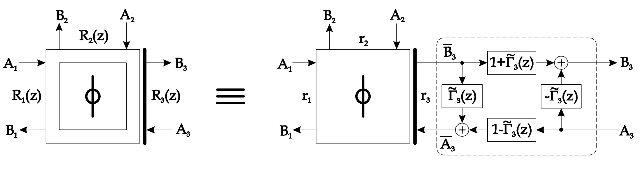

Figure 2: Any 3-port parallel (series) adaptor can be implemented as a standard parallel (series) WDF adaptor, which adapted port is connected to 2-port scattering cell.

through" a 3-port WDF adaptor, by removing it from the adapted port and inserting one like it on the other two ports, properly modifying the initial conditions. This last property allows us to model a generic macro-adaptor with memory as a memoryless macro-adaptor whose ports are connected to scatterers with memory, which will come at handy when dealing with implementational issues.

3. AUTOMATIZING THE SYNTHESIS

Some methods are already available for synthesizing macro-blocks, therefore the automatic synthesis procedure is based on the assumption that such elements are already available in the form of a collection of pre-synthesized structures. Currently, the family of blocks includes WD mutators [4] and other types of adaptors developed for modeling typical nonlinear elements of the classical nonlinear circuit theory (both resistive and reactive).

In order to devise a systematic approach to the implementation of W structures we need an appropriate data structure and a *method* that allows us to compute incident and reflected waves at each port of a macro-adaptor. If the circuit were memoryless, we would only need to apply our method to our data structure once in order to derive the solution vector (i.e. a configuration of waves that complies with the intrinsic I/O relationships of the blocks and the global continuity laws). In all practical cases of interest, however, our circuit is not instantaneous, therefore the solution vector ends up containing the system's *memory*. Once we assign such vector an initial configuration, at each iteration we update its content, to produce the next instance of the solution.

Currently two different methods are available: the first is inspired by the *tableau analysis method*, commonly employed in circuit theory to analytically determine the evolution of (analog, time-varying, linear) electrical circuits, while the second is based on a direct inspection of the numerical structure (the circuit) according to a tree-like structure that describes the interconnection topology of the elements. We refer to the first method as *Wave Tableau* (WT) method, and we call the second approach as *Binary Connection Tree* (BCT) method.

3.1. The WT method

In its first version, the WT method was based on the solution of a linear system whose vector of unknowns contains all incident and reflected waves that appear in the circuit (also those that describe the interconnection between different adaptors within the macro-adaptor), therefore the number of components turns out to be twice the number of circuit ports[2]. This method has been recently improved by constructing the WT matrix of the sole macro-adaptor, and considering as known variables the previously computed waves coming from the bipoles connected to it. The whole updating process is thus broken into two steps: one for the one-step evolution of the MA, and one for the one-step evolution of all the elements connected to the MA. This allows us to reduce the size of the linear system to be solved. Still, the WT description obtained either way is an implicit one, as the unknown vector containing both incident and reflected waves is computed at each step as a function of another vector containing just the inputs to the macro-adaptor. As a recent development of this project, it is now possible to solve the system analytically and rewrite it as an explicit state-update matrix equation in terms of a *scattering matrix* that describes the MIMO I/O relationship. This new problem formulation allows us to compute the reflected waves as an explicit function of the incident waves (which are now the system's inputs). The dimension of the WT matrix increases very rapidly with the number of bipoles that appears in the structure. As a consequence, the complexity tends to grow just as fast, even if we exploit the fact that the WT matrix tends to be quite sparse. With the introduction of the new method that brings the system into the form of a state update equation, the scattering matrix is no longer sparse and is up to 64 times smaller, therefore the latest version of the method is far more efficient than before.

To solve a circuit in the K domain means to determine the values of the across/through pairs in each one of its ports. The classical tableau analysis for analog linear circuits is based on the construction of a linear system in the $2N$ unknowns. The necessary equations to do so are divided into two groups:

- N I/O relationships of the individual bipoles, which can be seen as *local* equations (e.g. Ohm laws);
- N equations derived from the laws of continuity (*global* equations), which describe the interconnection topology (e.g. Kirchhoff laws).

A similar method can be readily derived for W structures [2] by treating the adaptors exactly like circuit elements. Indeed, with this choice the number of unknowns becomes equal to twice the number $P = 4N - 6$ of ports. Also in this case we will need $2P$ equations, half of which will come from local equations, and the other half will come from global equations.

The local relationships for W descriptions take on the form of scattering equations $\mathbf{b} = \mathbf{M}_0\mathbf{a} + \mathbf{u}$,which can be written in implicit form as

$$\begin{bmatrix} \mathbf{I} & -\mathbf{M}_0 \end{bmatrix} \begin{bmatrix} \mathbf{b} \\ \mathbf{a} \end{bmatrix} = \begin{bmatrix} \mathbf{u} \\ \mathbf{0} \end{bmatrix},$$

where $\mathbf{a}$ and $\mathbf{b}$ are the vectors of the incident and reflected waves, respectively; $\mathbf{u}$ is the vector of inputs (generators), and $\mathbf{M}_0$ is the scattering matrix. The dimension of such vectors is equal to the total number of ports, therefore in the case of a bipole, we end up with a scalar equation.

If the multiport adaptor is dynamical, then $\mathbf{b}$ depends also on the previous values of $\mathbf{a}$ and $\mathbf{b}$

$$\mathbf{b}(n) = \mathbf{M}_0\mathbf{a}(n) + \mathbf{u}(n) + \sum_{k=1}^{K_a} \mathbf{M}_k\mathbf{a}(n-k) + \sum_{k=1}^{K_b} \mathbf{M}'_k\mathbf{b}(n-k)$$

$$\begin{bmatrix} \mathbf{I} & -\mathbf{M}_0 \end{bmatrix} \begin{bmatrix} \mathbf{b}(n) \\ \mathbf{a}(n) \end{bmatrix} = \\ = \begin{bmatrix} \mathbf{u}(n) \\ \mathbf{0} \end{bmatrix} + \sum_{k=1}^{K} \begin{bmatrix} \mathbf{M}'_k & \mathbf{M}_k \end{bmatrix} \begin{bmatrix} \mathbf{b}(n-k) \\ \mathbf{a}(n-k) \end{bmatrix} \quad ,$$

where the matrix $[\mathbf{M}'_k \ \mathbf{M}_k]$ expresses this dependency. $K = \max(K_a, K_b)$ is seldom greater than 1.

We then need to specify a pair of equations for each interconnection. For example, if port i is connected to port k, then we have $a_i = b_k$, $a_k = b_i$. If P is the number of ports, then we have a total of P of equations of this sort (two ports connected together need two interconnection equations).

Once we have the scattering and the interconnection relationships of all elements, we can assemble the whole system of $2P$ equations in $2P$ unknowns

$$[\mathbf{T}_0] \begin{bmatrix} \mathbf{b(n)} \\ \mathbf{a(n)} \end{bmatrix} = \begin{bmatrix} \mathbf{u(n)} \\ \mathbf{0} \end{bmatrix} + \sum_{k=1}^{K} \mathbf{T_k} \begin{bmatrix} \mathbf{b(n-k)} \\ \mathbf{a(n-k)} \end{bmatrix} \quad ,$$

where $\mathbf{T}_0$ is the memoryless portion of the tableau matrix. This matrix is made of four $P \times P$ blocks

$$\mathbf{T}_0 = \begin{bmatrix} \mathbf{I}_P & -\mathbf{M}_0 \\ \mathbf{C} & \mathbf{I}_P \end{bmatrix} \quad ,$$

where $\mathbf{M_0}$ is the block-diagonal matrix containing the instantaneous portion of the Scattering Transfer Functions (STFs), which are the intrinsic I/O relationships of the circuit elements; $\mathbf{C}$ is the interconnection matrix; and $\mathbf{I}_P$ is the order-P identity matrix. The matrix $\mathbf{T}_k$ represents the component of the Tableau matrix that acts on those input and output samples that are delayed k time steps. In practice, the matrices $\mathbf{T}_k$ describe the dynamics (i.e. the *history*) of the network elements.

$$\mathbf{T}_k = \begin{bmatrix} -\mathbf{M}'_k & -\mathbf{M}_k \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \quad .$$

Finally, $\mathbf{u}(n)$ is the P-dimensional vector of known terms representing, for example, the generator's across variables at time n.

In principle, we should compute a solution of this system for each time sample. However, in order to speed-up the processing we can compute the inverse of T_0 in the initialization phase and compute the solution as

$$\begin{bmatrix} \mathbf{b(n)} \\ \mathbf{a(n)} \end{bmatrix} = [\mathbf{T}_0]^{-1} \left(\begin{bmatrix} \mathbf{u(n)} \\ \mathbf{0} \end{bmatrix} + \sum_{k=1}^{K} \mathbf{T_k} \begin{bmatrix} \mathbf{b(n-k)} \\ \mathbf{a(n-k)} \end{bmatrix} \right) \quad .$$

This way the computational complexity is reduced from P^3 (typical methods for the solution of linear systems) to P^2 (matrix-vector product).

With minor changes in the above approach, it is possible to solve circuits containing one nonlinearity, as long as the NLE is described in explicit W form $b = f(a)$, and is connected to the adapted port available in the macro-adaptor. This can be done by replacing the product between the coefficient of $\mathbf{T}_0$ in position $(P, 2P)$ by the incident wave $\mathbf{a}_{2P}$ at the port $2P$ with the term $f(a)$. In order to preserve the linearity of the system, this terms needs to be moved onto the other side of the equation and treated as a previously-computed variable. Indeed this is possible by exploiting the fact that the port that the NLE connects to is an adapted one. The modified system thus becomes

$$[\mathbf{T}_0] \begin{bmatrix} \mathbf{b(n)} \\ \mathbf{a(n)} \end{bmatrix} = \begin{bmatrix} \mathbf{u(n)} \\ \mathbf{0} \end{bmatrix} + \\ + \sum_{k=1}^{K} \mathbf{T_k} \begin{bmatrix} \mathbf{b(n-k)} \\ \mathbf{a(n-k)} \end{bmatrix} + \begin{bmatrix} \begin{bmatrix} \mathbf{0_{P-1}} \\ f(a_P) \\ \mathbf{0_P} \end{bmatrix} \end{bmatrix} \quad .$$

Moving this nonlinear function on the other side of the Tableau equation (removing the corresponding intrinsic description from the Tableau matrix) and treating it as a known variable means using the wave reflected by the NLE like an input to the adapted port.

By extending this last idea to all bipoles of the circuit we come to a second formulation of the WT method in which the matrix of the coefficients describes only the MA (i.e. only the adaptors within it). This is possible because all interconnections in W systems are done in such a way to avoid instantaneous (non-computable) loops, therefore the adaptation condition (no instantaneous reflection) is satisfied either by the bipole or by the port that it connects to. This allows us to split the state-update process in two phases: in the first one we compute the waves reflected by the MA (using the Tableau system driven by the waves produced by the bipoles), and the second one we compute the waves reflected by the bipoles when driven by the waves coming from the MA.

The new tableau system becomes

$$\begin{bmatrix} \mathbf{I_{3L}} & -\mathbf{M_0} \\ \mathbf{C} & \mathbf{I_{3L}} \end{bmatrix} \begin{bmatrix} \mathbf{b(n)} \\ \mathbf{a(n)} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{3L} \\ \mathbf{u(n)} \end{bmatrix} + \\ + \sum_{k=1}^{K} \mathbf{T_k} \begin{bmatrix} \mathbf{b(n-k)} \\ \mathbf{a(n-k)} \end{bmatrix} + \begin{bmatrix} \mathbf{0_{3L}} \\ \begin{bmatrix} \mathbf{0_{3L-1}} \\ f(a_{\mathbf{3L}}(n)) \end{bmatrix} \end{bmatrix}$$

In this case $\mathbf{M_0}$ is a 3×3 block-diagonal matrix (the system has only three-port adaptors), $\mathbf{C}$ describes just the $3L - N$ interconnections between adaptors, and $\mathbf{u(n)}$ is the vector of the $3L$ inputs. The input vector has a total of N non-zero elements that are placed in the second half of the vector of "known variables", which allows us to distinguish the incident waves that are coming from the bipoles. When the macro-adaptor is not instantaneous, some of the matrices $\mathbf{T}_k$ are non-zero. In conclusion, the Tableau system of the macro-adaptor is made of

- $3L$ intrinsic I/O relationships (which can be expressed as L matrix equations of the form $\mathbf{b} = \mathbf{Ma}$)
- N interconnection equations between macro-adaptor and bipoles ($a_i = u_i$)
- $3L-N$ interconnection equations between adaptors ($-b_i + a_k = 0$).

The terms on the right-hand side of the above equation, which are known at time n, may be grouped together into a single vector $\mathbf{t}(n)$.

$$\begin{bmatrix} \mathbf{I_{3L}} & -\mathbf{M_0} \\ \mathbf{C} & \mathbf{I_{3L}} \end{bmatrix} \begin{bmatrix} \mathbf{b(n)} \\ \mathbf{a(n)} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{t(n)} \end{bmatrix} \quad . \qquad (1)$$

The State-Update equation. The tableau system that describes the MacroAdaptor (MA), shown in eq. (1) emphasizes the strict relationship between the input vector $\mathbf{t}(n)$ and the vector of incident waves $\mathbf{a}(n)$. In fact, eliminating from $\mathbf{a}$ the elements that do not correspond to the inputs, we are left with $\mathbf{t}$. In other words, if we eliminate the equations that correspond to internal interconnections in the MA, it is possible to write $\mathbf{b}$ as a function of $\mathbf{t}$. The elimination of the non-necessary equations leads to the scattering equation of the MA, whose dimension is $N \times N$, just like in the analog case.

At this point we can decompose the system in the two groups of local (intrinsic) and global (extrinsic) equations. By plugging the second ones into the first ones to eliminate $\mathbf{a}(n)$, we obtain

$$\begin{cases} \mathbf{b}(n) - \mathbf{M_0}\mathbf{a}(n) = \mathbf{0} \\ \mathbf{C}\mathbf{b}(n) + \mathbf{a}(n) = \mathbf{t}(n) \end{cases}$$

$$(\mathbf{I} + \mathbf{M_0}\mathbf{C})\mathbf{b}(n) = \mathbf{M_0}\mathbf{t}(n)$$

where, for the moment, we have assumed the adaptors to be memoryless (first equation equal to zero). If the matrix $\mathbf{I} + \mathbf{M_0}\mathbf{C}$ is non-singular, then we have

$$\mathbf{b}(n) = (\mathbf{I} + \mathbf{M_0}\mathbf{C})^{-1}\mathbf{M_0}\mathbf{t}(n) = \overline{\mathbf{M}}\mathbf{t}(n) \ . \qquad (2)$$

This way we obtain a state-update equation in explicit form, where $\mathbf{t}(n)$ replaces $\mathbf{a}(n)$.

The dimension of the matrix $\overline{\mathbf{M}}$ is $3L \times 3L$, therefore it cannot be the MA's $N \times N$ scattering matrix $\mathbf{M}$ that we are looking for, as the vector $\mathbf{b}(n)$ obtained from eq. (2) contains also the waves reflected by the internal ports of the MA. The scattering matrix $\mathbf{M}$ can, in fact, be obtained by simply eliminating the equations relative to such internal ports. We can do so by using an appropriate $N \times 3L$ matrix $\mathbf{Q}$ obtained by eliminating rows from the $3L \times 3L$ identity matrix:

$$\mathbf{M} = \mathbf{Q}[(\mathbf{I} + \mathbf{M_0}\mathbf{C})^{-1}\mathbf{M_0}]\mathbf{Q^T} = \mathbf{Q}\overline{\mathbf{M}}\mathbf{Q^T} \quad .$$

The pre-multiplication by $\mathbf{Q}$ eliminates the unnecessary rows while the post-multiplication by $\mathbf{Q^T}$ eliminates the corresponding columns, in order to obtain again a square matrix. If we did not perform this column removal, the elements of such columns would multiply those elements of $\mathbf{t}(n)$ that are zero anyway. Such elements correspond to the incident waves at the internal interconnection ports. Indeed, the input vector will have to be resized accordingly, by eliminating all the zero elements

$$\mathbf{a}(n) = \mathbf{Q}\mathbf{t}(n) \quad ,$$

which results in the state-update equation of the form

$$\mathbf{b}(n) = \mathbf{M}\mathbf{a}(n) \quad ,$$

where $\mathbf{b}(n)$ and $\mathbf{a}(n)$ are now made of N elements.

3.2. The BCT method

In spite of the dramatic improvements introduced in the WT method, we decided to explore a different and novel approach, organized in iterative form. This method turns out to be the most efficient one, as it is based on a direct inspection of the numerical structure. The method starts from the incident waves to the macro-adaptor, and follows their path throughout the whole structure once every time sample. In order to generate the path, in fact, we scan a tree that describes the circuit topology. If the structure is based just on three-port junctions, the resulting connection tree turns out to be binary (hence the name binary connection tree). Like in the WT method, also with the BCT we do not necessarily need to use three-port adaptors. However, considering that any N-port adaptor can always be decomposed into the interconnection of $N-2$ three-port adaptors (see Fig. 1), we can use a BCT with no loss of generality. The BCT formally describes the interconnection topology of the adaptors under the following rules:

- the **root** corresponds to the adaptor that the nonlinear (NL) element connects to;
- the **nodes** are 3-port standard WDF adaptors and the branching topology matches the actual adaptor's interconnection topology;
- the **leaves** correspond to the bipoles.

Once the connection tree is built, the computational procedure can be constructed in two steps: a *forward scan* of the tree (from the leaves to the root), followed by a *backward scan* (from the root to the leaves). In fact, the computation begins from the memory cells, which are in the leaves of the tree and contain all the initial conditions of the system and keeps nesting function calls until we reach the root (NL element), obtaining the reflected waves at the adapted ports of each adaptor. In the backward scan, once we have the wave reflected by the NL element, all other reflected waves can be computed, reaching the leaves again and updating their content with the reflected wave of the adaptor they are connected to. In other words, following this path we always have all necessary data to compute the waves we need.

The initialization procedure follows a similar approach. Determining the initial condition means solving a set of equations, one of which is nonlinear. Indeed, the solution of this set of equation is rather simple, as it requires a search for a fixed point. The problem is to specify the set of equations starting from the connection tree. Since during this phase the reactances are formally replaced by ideal generators, it is not possible to use W variables directly, because they do not have an adapted representation and the structure would turn out to be non-computable. However, we can still use the tree structure that describes the circuit topology, which works irrespectively of whether we are working in the W domain or in the K domain. The process can again be splitted into two phases: a forward scan (from leaves to root) and a backward scan (from root to leaves). In the first phase we derive the characteristic lines that describe the relationship between current and voltage at each node. This way, during the backward scan, knowing one of the two variables, we can compute the other one using these characteristics.

One key feature of this approach is that its computational cost and memory requirements increase linearly with the number of adaptors. Of course, this improved efficiency costs in terms of evocative power of the structure.

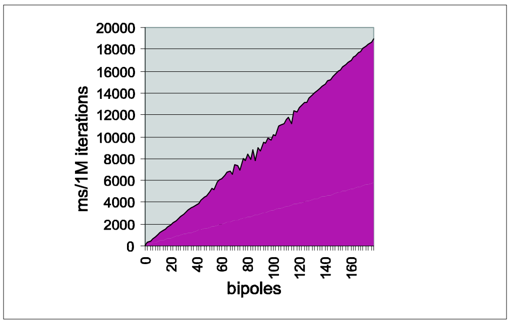

Figure 3: Execution time vs number of bipoles

4. MANAGING TIME-VARYING STRUCTURES AND TOPOLOGICAL CHANGES

Changing any model parameter in a WD structure usually affects all the other parameters as they are bound to satisfy global adaptation conditions. Temporal variations of the nonlinearities are easily implemented by employing special WD two-port elements that are able to perform a variety of transformations on the nonlinear characteristics (non-homogeneous scaling, rotation, etc.). Temporal variations of reference resistances, on the other hand, are implemented through a re-computation of the model parameters on the behalf of a process that works in parallel with the simulator. Using the BCT method, when the value of a leaf changes, the adaptors that need to be updated are only those lying on the path that link the leaf to the root (fig.4). The parameter update, however, is not computationally intensive as it is performed at a rate that is normally only a fraction of the signal rate (e.g. 100 times slower). It is important to remember, however, that abrupt parameter changes must be carefully dealt with in order not to affect the global energy in an uncontrollable fashion.

Let us consider an object that could potentially interact with a number of other objects in a sound environment. For example, we could think of a mallet that could potentially collide with a number of drum-like resonators. Indeed, this situation cannot be implemented with a fixed interaction topology. In order to be able to implement this dynamic topology, we need to be able to connect or disconnect objects on the fly. This can be achieved by exploiting the fact that a connection between systems becomes *irrelevant* when their contact condition is not satisfied.

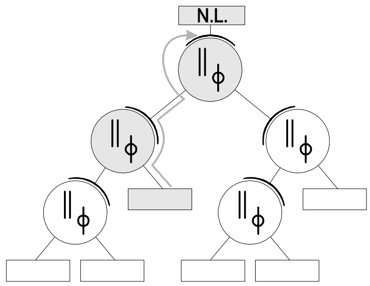

Figure 4: Tree updating after a bipole value change.

Time-varying topologies can be implemented using the WT method. The operation, however, is quite complex, as it requires replacing two independent tableau systems with a single larger one. This means that we need to replace two smaller state-updated equation with a single larger one. Doing so without producing undesirable discontinuities (clicks) in the generated sound is rather difficult, as it requires extra care in the timing of the replacement and in the initialization of the system.

Working with BCTs, in fact, is simpler, as they naturally offer an enhanced flexibility in managing topological changes. Let us consider a set of independent physical systems, each represented by a BCT. We want to assess the problem of how to go about connecting them together and how their topology changes during the interaction. Assuming, for the sake of simplicity, that the two circuits connect with each other through a single port (*interconnection port*), we would like their port to become "transparent" when the objects are isolated (no contact). This means that the port resistance is zero if it comes from a series adaptor, or infinity if it comes from a parallel one. We must remember, however, that the interconnection of two circuits could originate computability problems in the W domain, particularly if both circuits contain a NLE. In a wide variety of acoustic physical models, however, NLEs are separated by instantaneously decoupling multiports, such as DWGs, therefore they can be safely connected together.

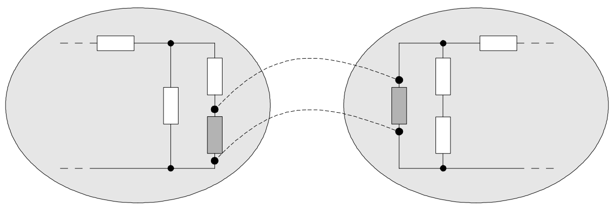

Figure 5: Interconnection between subsystems in the analog domain. The shadowed element are *pseudo-bipoles*, which act as interconnection ports

Even when we need to interconnect a linear W system with a nonlinear one, we still need to have some element that enables the connection. Since we are in a situation in which we do not need any decoupling, this interconnection element could also be memoryless. In a linear circuit the root of the BCT could be any of the bipoles (if have a BCT and have it *dangling* from another one of its nodes, we will end up with another BCT). If a linear circuit has an interconnection port, we can take that as the root of the BCT, so that it can act as the "shoot" (subroot) to be "grafted" to the receiving tree. Notice, however, that the state update equation does not treat the instantaneous interconnection port as a bipole, as it does not "contain" a numerical value but a pointer to another structure. If we have a set of independent physical systems, each

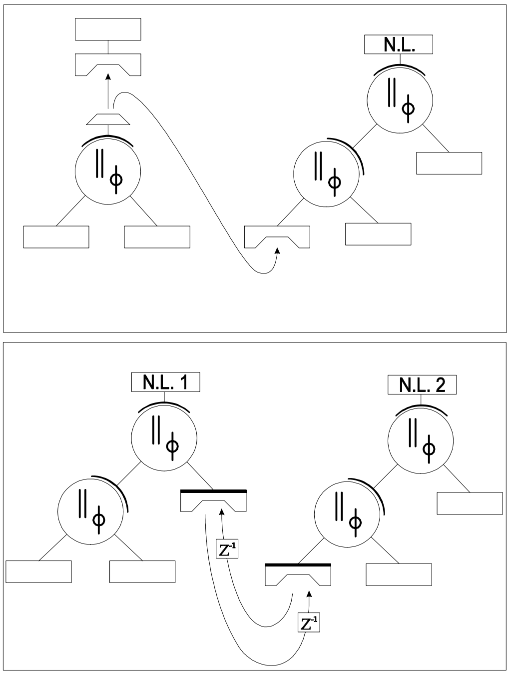

Figure 6: Memoryless (up) and dynamic (down) interconnection ports. The bold border indicates the instantaneous adaptation due to the memory

represented by a BCT. Such systems share the same environment but, for the moment, they do not interact with each other. To adopt an evocative metaphor that fits the idea of tree-like models, we can refer to the environment as the *orchard*. We want to enable interactions between trees in the orchard and how we want to give the possibility to alter the topology of trees on the fly. This problem may occur especially when dealing with sound generation for the sonification of virtual environments.

Let us consider a W hammer model interacting with the W model of an ideal string. The W hammer is made of a mass and a nonlinear spring that models the lossless and instantaneous limited compressibility of the felt. Both systems can be modeled with a single circuit but, to explain the above method, are here kept as separate through memoryless interconnection ports. During the interaction we can identify the following

1. initially the objects are far apart and their ports are disconnected. Such ports are transparent with respect to their circuits. In fact, the string port is a series one, therefore it is a short circuit; while the hammer port is a parallel one, therefore it is an open circuit.
2. When hammer and string are close to each other (proximity condition) we can establish a connection, and the string BCT can be grafted into the hammer BCT, originating a single structure. As far as the circuit behavior is concerned, however, nothing has changed, as the series adaptor is still short-circuited by the NLE, which is working on the linear portion of its characteristics with slope -1.
3. The situation changes when the hammer comes in contact with the string (contact condition), i.e. when the working point on the NLE characteristics begins changing slope. From now on, there is a non-zero power exchange between elements, therefore the hammer will begin bouncing against the string until it will be push away from it.
4. When the hammer is sufficiently far apart from the string, the proximity condition ceases to be valid, therefore the connection can be removed and the circuits are once again isolated.

Notice that although the interconnection ports and the particular behavior of the NLE (a step function in the K domain) play a similar role, irrelevant interconnections and absence of connection have consequences on the organization of the implementation. In fact, when the hammer is disconnected, it can be used elsewhere. Roughly speaking, a piano harp can use a limited amount of shared hammers

5. CONCLUSIONS

The proposed approach has proven effective for the automatic and modular synthesis of a wide class of physical structures encountered in musical acoustics. In fact, both the Wave Tableau approach and the Binary Connection Tree approach we implemented make the construction and the implementation of the interaction topology systematic. In its current state, the implementation of the described synthesis system is able to assemble the synthesis structure from a syntactic description of its objects and their interaction topology, opening the way to a first CAD approach to the construction of an interactive sound environment.

6. REFERENCES

[1] A. Fettweis: "Wave digital filters: theory and practice". *Proceedings of the IEEE*, Vol. 74, No. 2, pp. 327–270, Feb. 1986.

[2] F. Pedersini, A. Sarti, S. Tubaro: "Block-wise Physical Model Synthesis for Musical Acoustics". *IEE Electronic Letters*, Vol. 35, No. 17, Aug. 1999, p. 1418-19.

[3] A. Sarti, G. De Poli: "Generalized Adaptors with Memory for Nonlinear Wave Digital Structures". *VIII European Signal Processing Conference*, 1996, Trieste, Italy, Vol. 3, pp. 1773-6.

[4] A. Sarti, G. De Poli: "Toward Nonlinear Wave Digital Filters". *IEEE Transactions on Signal Processing*. Vol. 47, No. 6, June 1999.

[5] J.O. Smith, "Principles of digital waveguide models of musical instruments", in *Applications of digital signal processing to audio and acoustics*, edited by M. Kahrs and K. Brandenburg, Kluwer, 1998, pp. 417-466.