GENESIS : a Friendly Musician-Oriented Environment for Mass-Interaction Physical Modeling

Nicolas Castagné, Claude Cadoz

GENESIS: A Friendly Musician-Oriented Environment for Mass-Interaction Physical Modeling

Nicolas Castagne, Claude Cadoz

ACROE-ICA, INPG, Grenoble, France
email: Nicolas.Castagne@imag.fr Claude.Cadoz@imag.fr

Abstract

Mass-interaction scheme is often cited as the traditional physical modeling technique, but surprisingly some of the musical creation fields it allows have not yet been pointed out.

GENESIS is a musician-oriented graphical environment for mass-interaction physical modeling. It offers some new possibilities for music creation based on an articulation between reality and "virtuality".

This paper presents version 1.5 of the GENESIS environment we have developed in the recent years. It introduces its major functionalities and their conceptual origins. It explains the choices we made to provide attractive model representation, low and high level modeling tools, and visual output during simulation.

1 Introduction

Physical Modeling (PM) is increasingly used for musical purposes. It is generally supposed to allow (see Borin, De Poli and Sarti 1992, Jaffe 1995, Smith 1996, Pearson and Howard 1996 for example):

- A more convincing re-synthesis of real instruments' sounds;
- More generally, a better plausibility of synthesized sounds even with non-realistic chimera models. This is musically very interesting since human hearing likes being able to infer a possible cause to sounds (Risset 1990);
- An easier and more natural mapping of dynamic inputs inside models – through force or position variables;
- As a consequence of the above points more interesting sounds and better expressiveness possibilities.

Despite these common attempts, different approaches of PM present different interests. Among them, the mass-interaction paradigm:

- Allows a highly modular modeling process. A model is composed of very elementary modules. Each of these grains of matter has an easy comprehend physical behavior;
- Leads easily to multisensorial simulation Especially, visualization of the model's behavior is a natural feature with the mass-interaction paradigm;
- Enables an efficient and natural mental model of algorithms, which helps in reducing the gap between reality and "virtuality". Models are more easily internalized as representations of real objects than with more mathematical or signal processing PM approaches;
- Is not specifically dedicated to sound structures, but more generally to the modeling of matter Both low frequency and high frequency objects can be combined in a model that generates both individual sounds and musical sequences. Music formal and material contents can be thought in the same way, and different musical scales of time can be juxtaposed.

The mass interaction paradigm is thus a particularly deep approach to PM. It allows representation of physical objects as a whole, and not only of the physical phenomena that generate sound. Though one may say it is expensive in terms of processing time, it appears to be of a great interest when the purpose is not only to make new – and perhaps better – sounds, but also to let a non-physicist musician create entirely his musical material, including modeling. Being able to be thought of as real objects, mass-interaction models may support new ways of creating music based on a particularly sensitive articulation between real and virtual.

In this paper, we present the version 1.5 of the GENESIS musical creation environment that allows these creation processes. Based on the CORDIS-ANIMA system, GENESIS combines a user friendly graphical software for object design and CORDIS-ANIMA simulation engines. This paper details the recent development of the GENESIS' "lutherie"

workshop, and emphasizes the choices we made in order that it would be flexible, open and efficient.

2 Background

2.1 CORDIS-ANIMA topological language and system

General overview. The CORDIS-ANIMA language and system (Cadoz, Luciani and Florens 1993) enables the building of physical models from elementary physical modules. Such a CORDIS-ANIMA model is a network composed of mass-like objects (MAT) connected with physical relations (LIA).

The simulation space can be of any dimension. However, experiences proved that 1D-simulation-space models are of interest for music synthesis: they are at the same time *easier* to design, *quicker* to simulate and *efficient* for interesting sounds generation. We thus decided to build GENESIS on the 1D or *topological* version of CORDIS-ANIMA. In this version, the state of mass-like modules is determined along a single *movement axis.*

The basic physical MAT modules (and parameters) are: MAS (inertia) for masses and SOL (no parameter: inertia is infinite) for fixed-points.

The basic physical LIA modules (and parameters) are: RES (stiffness) for springs; FRO (damping) for friction; BUT (stiffness, damping, threshold) for space-related interactions – links which are active only when the two connected masses are in a common sphere of influence; and LIC for other nonlinear interactions.

Two macro-modules are added to the above: REF (stiffness, damping) mixes a RES and a FRO; CEL (inertia, stiffness, damping) mixes a SOL and a MAS connected with a REF. A CEL is thus an *elementary physical oscillator*, that is the simpler object able to oscillate and generate a sound signal.

CORDIS-ANIMA defines some in/out modules. SOX (sound-voice number) sends the movement of the MAT module it is connected with to loudspeakers and/or sound file. The ENX module, on the opposite, is a MAT that represents a gesture device (see below) or a gesture file. ENX is a bi-directional module: its position during simulation is taken from a file or from the gesture device, and the force it receives from the simulated model is sent outside for eventual use –in particularl force-feedback perception in the case of real time simulations.

Non-linear Interactions. The reference CORDIS-ANIMA articles leave open the question of nonlinear interactions, with a very large definition of LIC. We restricted the LIC by defining two operational non-linear interactions: the LNL and the X^3.

- *LNL: local non-linearity*

A LNL link is defined by two point-by-point curves. Between two data points, the curve can be linear or smoothed with splines, parabolas or polynomial functions. The first curve defines the force to be applied by the LNL according to the distance deltaX along the movement axis between the two connected masses. The second curve defines the force to be applied according to the relative velocities DeltaV of the two masses.

Such a non-linear link allows the modeling of complex non-linear interactions between objects. Among the simplest, for example, it is possible to simulate plucking-like interactions or bow/object-like interactions.

- *X^3 stiffness: distributed non-linearity*

We know that non-linear material behavior is of importance in sound objects. The ear often analyses it as a signature of excitation intensity.
However, a 1D model with basic mass, damping and stiffness modules is a linear model – such as a Modal Synthesis or a Wave-Guide model. We added a non-linear cubic term to stiffness (see Castagne and Cadoz, 2000, for details and reasons), so that the force is computed as:

$$F = K * deltaX * (1 + NL * deltaX^2)$$

The X^3 distributed stiffness enhances the sounds' plausibility and the transients' quality, with a very reasonable cost in term of processing time. By setting the new NL parameter, the user is able to adjust the non-linearity to be heard in sounds (see below).

2.2 The simulation engines

Genesis uses the non real-time Cordis-Off simulation engine, which is a C library. The state of the simulated object is, however, accessible during simulation. We will provide visualization tools and the possibility to listen to the sound while it is computed. With Cordis-Off, it is possible to simulate objects as complex as desired – with typically 500 to 10000 MAT or more. Gesture signal files can be used to drive ENX module inputs, if needed. Note that GENESIS may however be used with real-time and multisensorial simulation, including gesture interaction with the force feedback device designed for years in the laboratory (TGR ©, see Florens, Cadoz and Luciani 1990).

2.3 About the creative processes

The creative process supposes first the design of the physical object in a "lutherie" phase. An intuitive process can be used, but a kind of *modeling-oriented way of thinking* is often preferable. The aim is then to express *metaphors* of real objects in the CORDIS-ANIMA language.

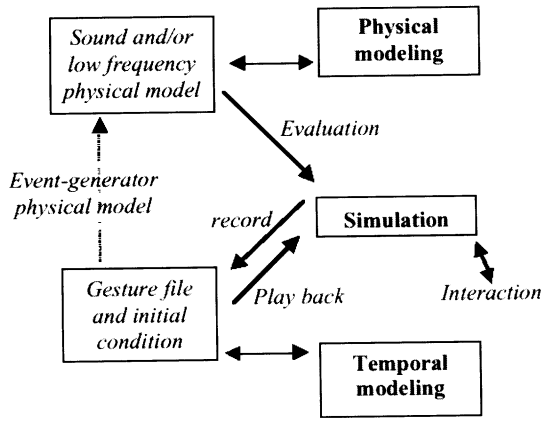


Figure 1. Aspects of a creative process

Modeling will determine the structural aspect that will be heard in the sound. During modeling, simulation is a testing and evaluation tool. When the user thinks the designed objects are interesting enough, playing can begin. Playing will determine the temporal aspect of the sound. This temporal aspect consist of initial state control and inputs to ENX modules.

When real time is possible, the latter may be achieved interactively with the TGR© force feedback interface or other devices. When real time is not possible (GENESIS' actual situation), playing has to be specified before simulation. In both cases, we need complete temporal input editing features.

As a conclusion of the above, the GENESIS environment should be conceived around the simulation (RT or non RT) paradigm, and exhibits two modes: a "lutherie" mode with modeling features, and a performance mode with more explicit time features – including gesture signal editing. The rest of the article will mainly emphasize the "lutherie" mode development.

3 Goals

Our goals designing the GENESIS environment and especially the GENESIS' "lutherie" workshop were:

- ·To allow, as a basic philosophy, operating at the atomic level. We wanted to let the user deal with elementary grains of matter and thus take advantage of the flexibility of the CORDIS-ANIMA modeling system.
- ·To design high level modeling tools to manage a group of modules as a whole when desired - which also supposes structural and parametric features;
- ·To provide a complete music creation environment, that is to make it possible for GENESIS to be the main and central software in the electroacoustic studio. In this scenario, GENESIS must support the entire creation process.
- To also offer an open environment that may interact as widely as possible with other software and standards;
- To develop a friendly user interface;
- To let GENESIS be extensible.

4 GENESIS' "lutherie" workshop

4.1 Objects representation

We first had to support the efficient mental model mass-interaction scheme allows by providing an easy-to-understand and attractive representation of designed objects. The graphical 2D representation we propose is a metaphor of a instrument-maker or "lutherie" workbench.

We tested different representations, some of which are given as options within GENESIS. Our preferred choices imply the use of shape, color, size and color intensity.

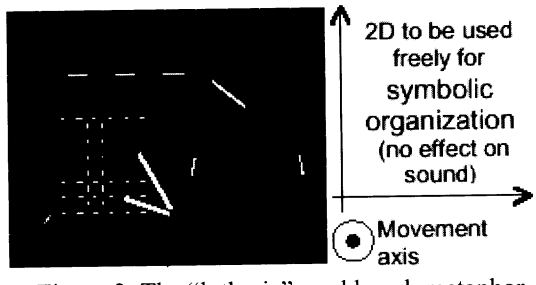


Figure 2. The "lutherie" workbench metaphor

- Shapes represent module category - circles for physical MAT modules, lines for LIA, square for input-output modules.
- Color then indicates the module behavior. For example: orange for MAS, white for REF, etc.
- Size of modules represents the scale of their main parameter. GENESIS defines a series of steps in size that correspond to a series of increasing "preferred values" for the parameter.
- For MAT modules, size is chosen according to the inertia value. The MAT size displayed is the size of the "preferred value" closest to the module's inertia.
- LIA thickness is computed in a more complex manner that takes into account the stiffness or

damping value of the LIA *and* the inertia of the two connected MAT. Thus, thickness codes the *aptitude of LIA to influence* MAT *movements*. As a result and for example, such choices allow to *guess* the typical resonant frequency of a MAT / RES / MAT system.

- Color intensity is a less perceptible display parameter. It is used to display the difference between the closest preferred values and the actual parameter value.

Such a representation optimizes the information users can easily acquire by looking at the workbench. Moreover, it is no doubt of a great help in allowing models to be perceived as objects, and not only as a mathematical or algorithmical constructions. It enhances the modeling process and a physical way of thinking.

Due to the use of the one-dimensional mass interaction scheme, masses move only on the axis perpendicular to the workbench. As a consequence, the two axes of the workbench do not have any effect on the phenomenon to be generated. They are left to organize freely the model within space, and to carry symbolic information. They may, for example, underline a behavior of a specific part of an object, symbolize time from left to right, physically express some kind of partitioning, or even allow a pictorial approach to GENESIS.

The "lutherie" workspace can be as large as needed – let's say 55*55 m^2 for example. GENESIS then provides easy to use complete zooming and panning features. Finally, note that many workbenches may be opened at the same time and that facilities to copy or move objects between workbenches are offered.

4.2 Basic editing level

Modularity is more profound with the mass-interaction scheme than with other PM approach: each module is a very small but still physically significant grain of material, and object shape and behavior emerge from built networks. GENESIS provides all the features necessary to handle the module level, in terms of both structure and parameters. They were conceived and ergonomically designed in order to support direct action, so that users interact as directly as possible with objects on the workbench during modeling (see Figure 3.).

Selection. Direct action is deeply rooted on a selection paradigm. Each module can be easily added to or subtracted from the selection, with different mechanizes of selecting: module selection, area selection, module category selection or group selection (see below) for example.

Tool Box and Structural Actions. A tool box gives access to the main actions. It proposes:

- A supply for every kind of module (MAS, SOL, RES, BUT, LNL...). By selecting one of these tools, the user can add modules to his object. By pushing on the menu, he can choose the preferred value (see §4.1) to be used when adding a module;
- A delete tool to delete the selection;
- Connection / disconnection tools;
- Different tools for selections;
- Some organization tools – which for example let
- the user move, rotate or dilate the selection in
- order to change the symbolic organization of the
- object on the workbench's 2D space;
- Full easy-to-use zooming and panning tools.

On the workbench, depending on the selected tool, the user will easily perform all the actions necessary to manipulate the network structure at the module level or selection level.

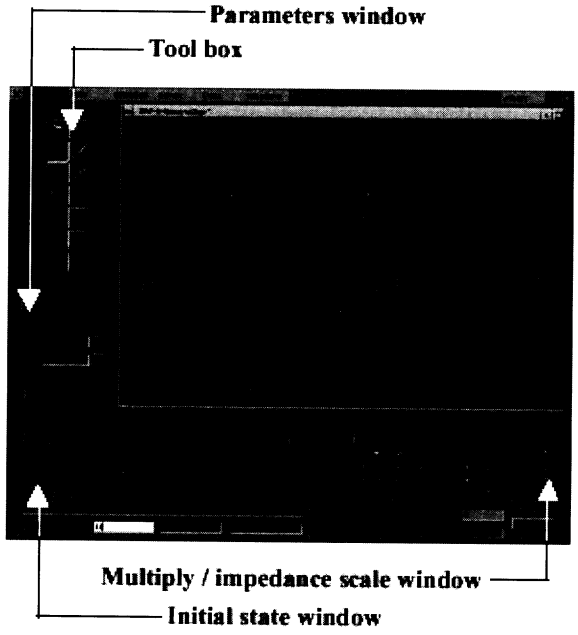


Figure 3. Direct actions features. "Lutherie" workbench and basic editing windows.

Parameter Editing. The physical parameter edition is mainly based on the homogeneous properties of matter. By selecting modules, the user will display and modify the numerical value of their parameters (inertia, damping stiffness...) provided they are homogeneous. With an optional slider the parameter can be changed around the *preferred-value* that is closest to the actual parameter value.

A *multiply* tool is also provided. It allows modifications to the *scale* of the whole selection. By applying the same factor to *all* the parameters of the selected modules, it is easily possible to change the selection's *impedance*: its intrinsic behavior will not

be modified but the way it can interact with other parts of the object will be affected.

In GENESIS, the distributed X^3 non-linearity is considered as a *final-step parameter*. The user can easily turn on/off X^3 non-linearity within the model. A control of the *perceptible* non-linear effect is provided (see below).

The non-linear LNL module required quite a special treatment: a two-level editing tool is provided.

The first level displays the two shapes of the LNL but only allows modifying its *amplitude* with four coefficients (max force and max deltaV of the velocity-relative part, max force and max deltaX of the position-relative part).

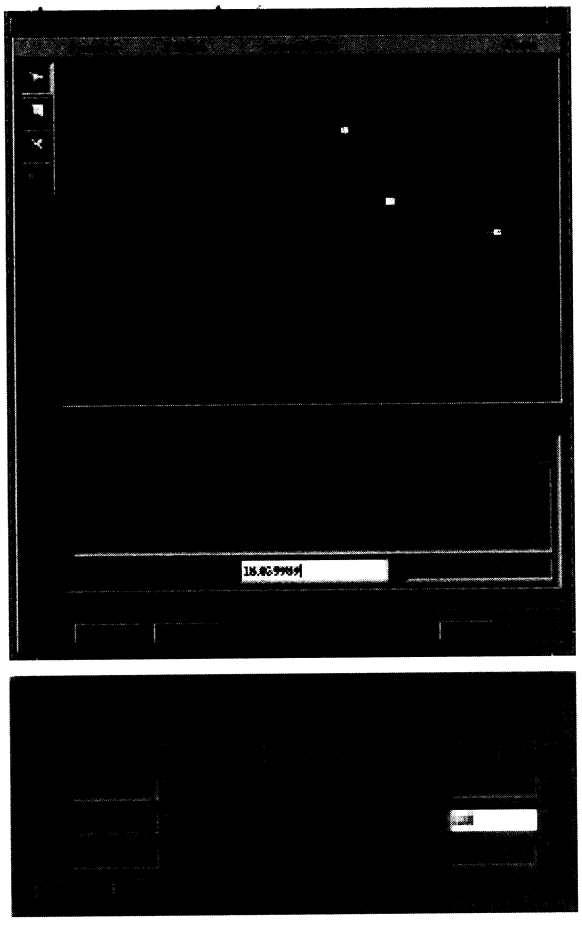


Figure 4. LNL Non-linearity edition tools Top : advanced editing. Bottom: Simple editing

The shapes of non-linearities themselves, which are less frequently modified, are edited in a second window. In this non-linearity advanced editing window, shapes may be forced to be symmetrical, smoothing is specified and absolute positions of the control-points along deltaX (resp. deltaV) and force axis are modified graphically or numerically. Note that common LNL modules are provided pre-built.

Initial State Editing. To perform simulation each MAT module needs an *initial state* (position and velocity along the movement axis) to be specified. These initial conditions differ from physical parameters: they are the very simplest way to specify a non-zero energy, and thus to perform some elementary "gestures" – i.e. dynamic inputs - on objects. They are not a property of matter, but a primitive time input. As a consequence they should not be edited by homogeneous selections, but module per module.

The use of the initial condition window is thus as follows: *first*, the user select the MAT he is interested in; the initial state of these modules is then displayed in a browser list; *second* he chooses in the browser a module, and modifies its initial position or velocity value. Note that a third redundant initial state parameter is provided. It shows, when possible, the *time* when the chosen initial state will have an effect on sound objects. For example, a MAT launched toward a sound object and connected with a BUT will strike the object after a time that GENESIS can compute. Such a *time* parameter is a very basic feature to deal with sound events. Other features will help in that sense, however.

4.3 High level editing

Modeling with the mass-interaction paradigmimplies the use of a number of elementary modules. While working, users exhibit a highly modular way of thinking, establishing dependencies and hierarchy between modules. These groups, actually, can be either *functional* (let us say: a cord, a membrane, a low frequency "player-like" oscillator, etc) or *morphologic* (groups of modules of the same category or with homogeneous parameters for example). In parallel with module-level modeling tools, GENESIS provides macro-modularity and high-level features.

Selection Groups. First of all, GENESIS proposes selection group facilities. Whenever he needs the user can nominate the current selection as a *selection* group and assign a name and comments to it.

Such a group can be easily re-selected by choosing it from a list of groups, or more *directly* by clicking on one of its modules. Grouping is non-exclusive: different groups can share some modules. We provided an easy way to choose between groups, all the groups, or even the module on its own when clicking on a module that belongs to one or more groups.

High Level Parameter Editing – Relations and Meta-Parameters. Basic physical parameters are not sufficient to summarize structure properties. To control certain behaviors, users often have to think in terms of dependencies between certain parameters of different modules, or even to define a new parameter space set of axes.

For example, we know that for a given sound structure the quality of the transients in case of a plucking excitation is connected with the inertia M of the excitator and the quotient Q between stiffness and inertia (Fourcade and Cadoz 1996, Fourcade 2001). M determines the brightness and Q determines the attack duration (or excitator/sound structure interaction duration). It would be of interest to let the user give *desired attack quality* parameters instead of explicit physical parameters, and to compute automatically the inertia and stiffness of the excitator. This example can however be generalized, which is taken into account with GENESIS' *relation* and *meta-parameter* system.

Capsules. Selection groups – as a macro-modularity facility – and relations – for high level parameter editing – are still not sufficient to take into account the user's whole macro-modular way of thinking. GENESIS proposes in addition the capsule metaphor, which allows *hierarchical* editing. A capsule is precisely a macro-module. It is composed of an unspecified number of modules but is structurally manipulated as a basic module – it may be, for instance, encapsulated in a higher-level capsule. Because a capsule is a whole, a capsule's modules are no longer accessible to user. However, a capsule's content can be edited by double clicking on it: it is then opened in a new "lutherie" workbench.

To create a capsule, one must:

- Select the modules to be encapsulated;
- Specify the capsule's name, comments, and display options. Display of a capsule can show all the inner modules – so that capsules look like ordinary networks, even though module level editing is inaccessible – or the capsule can appear as a *black box* with only access points;
- Specify which modules are to be connectable outside of the capsule, that is the "surface" of the capsule. For example, if a MAT module is connectable, it is possible to connect a LIA to it;
- Define the *control-parameters* - that is the way capsule's behavior is to be modified - and if needed their max or min and/or sets of prerecorded values.

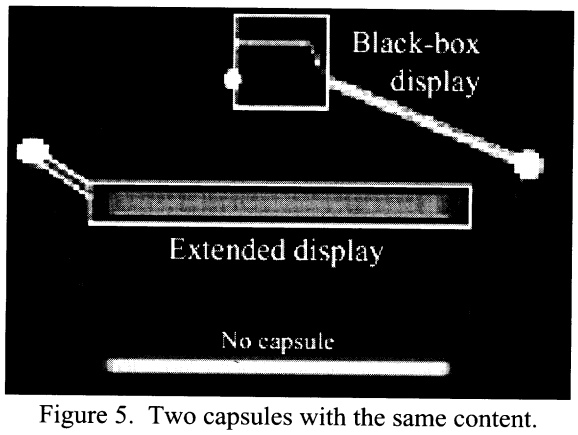


Figure 5. Two capsules with the same content.

Different choices are possible in designing a capsule's control parameters:

- Make every basic parameter visible. The problem is then that the number of parameters could be large, and the set could be meaningless.
- Define homogeneous regions. Each one of these regions will then be parameterized by a single set of inertia, damping and stiffness values.
- Leave accessible the meta-parameters that were previously defined in relation groups inside the capsule.
- Choose any combination of the above.

When defined, capsules' parameters behave as basic module parameters. They can be used, for instance, in relations.

In addition, we provide some specific capsules, with parameters of *structural level* and not only physical parameters. The *string* (parameters N: number of masses, F: frequency, DT: damping time) is such a capsule. The *modal chamber* we present below is another one.

4.4 Phenomena-oriented tools

Although GENESIS is deeply PM oriented we designed tools that allow dealing with *phenomena* to be generated. The aim is then to offer bridges between CORDIS-ANIMA models and phenomena characteristics.

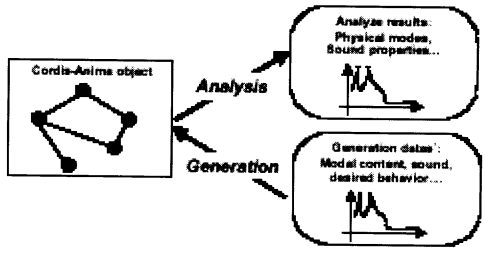


Figure 6. Analyze and generation principles

This is achieved by *analyzing* a GENESIS object in order to give information on the phenomena it may generate, or on the contrary by *generating* a model according to a given set of desired behavior and/or phenomena.

Modal Analysis and Tuning. The first tool is the modal analysis system. A 1D mass-interaction model with masses and spring-frictions is a *linear* model. It is possible to compute the modes of the structures (frequency, damping time and shape, see Incerti 1996). GENESIS provides a two-level tool for modal analysis (see Figure 7).

The first window displays the lowest-frequency mode properties. The user can then tune the object: he inputs the frequency and damping time he would like, and GENESIS multiplies damping and stiffness

throughout the structure by some factor so that desired values and real values match.

The second window displays all the modes of the structure, the global frequency-response of the structure between two points and the modal shape of a chosen mode. Tuning is then possible on each mode – so that, for example, the second mode of the structure matches specific values. Note however that tuning will multiply all the damping and stiffness inside the structure. With *tuning*, all the modes are modified at the same time. It may be of interest to handle each mode separately, which is achieved by the reciprocal problem of structure generation.

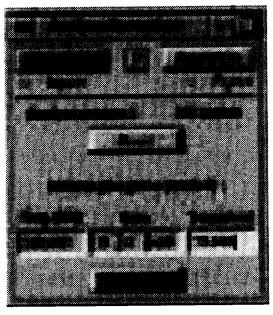
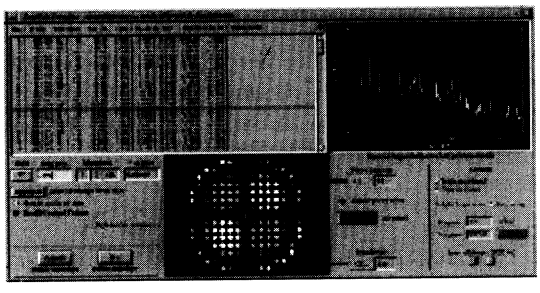


Figure 7.
Modal analysis windows.

Left: small display with fundamental mode characteristics.

Down: full display with mode list, frequency response and shape of a mode.



Structure Generation. The aim is here to generate a CORDIS-ANIMA object whose behavior matches a given set of modal data. This reciprocal problem is mathematically difficult to solve – it may have no solution, or an infinite number of solutions. We obtained however some preliminary results. The generation tool GENESIS proposes is thus able to generate an inhomogeneous cord-topology model that matches any desired (amplitude at a specific access point / frequency / damping) mode list.

The modal chamber The modal chamber is an easier but not entirely physical method for matching modal data in a CORDIS-ANIMA model.

Let's consider a set of elementary physical oscillators. The resonating properties (frequency, damping) of each oscillator can be set independently. Provided we have a transform matrix that fixes each mode's importance the set of oscillators becomes a *modal chamber* (Djoharian 1993). It can be added to a more general mass-interaction object. A specific capsule is designed within GENESIS that implements such a modal chamber as an optional feature. At this point, the modal chamber is parameterized with explicit mode frequency and damping time and explicit mode amplitude at the access point. Other controls may be added soon to provide access, when necessary, to typical modal synthesis approaches (Adrien 1991).

X^3 Perceptible Effect Input. GENESIS provides a tool to determine the value of the X^3 distributed non-linearity final-step parameter according to a desired perceptible effect. A *try* tool helps in scaling the perceptible non-linearity effect: the user provides a *desired non-linear audible intensity* (between 0 and 1), and GENESIS computes the corresponding X^3parameter value.

4.5 Other tools

Among other tools GENESIS proposes, let's note the *search facility* that will select modules in a large model according to a set of search fields (parameters value, module category, etc.), and the *divergence seeking* tool that may help debugging badly conceived or parameterized structures.

5 GENESIS' simulation window

Simulation is the main way to validate models (by observing their phenomena), and is used often within GENESIS. The two simulation engines are connected to the GENESIS graphical interface, and the user can select the one he needs.

The non real-time Cordis-Off engine is implemented as independent background programs, that can run in parallel. The data generated by the active workbench's simulator are displayed in GENESIS in the *simulation window*.

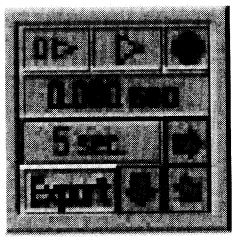
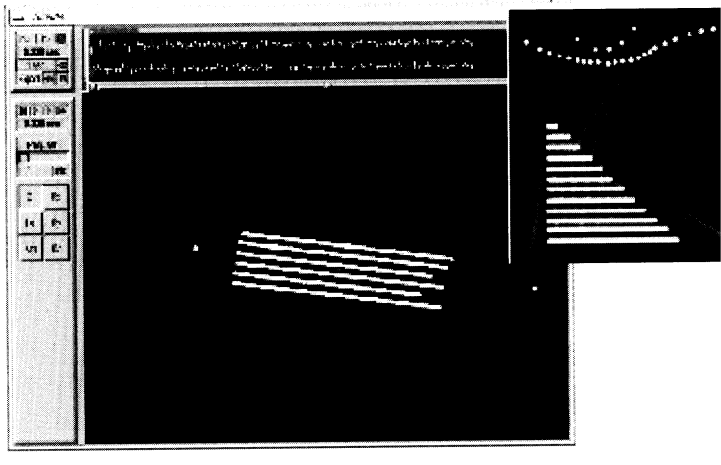


Figure 8.
The non real-time simulation window

Left: smaller aspect.

Down: large aspect with sound signal view and simulated structure view



The simulation window has different appearances according to the category of phenomenon the user want to look at: sound, sound signal, visualization or measures. The smaller aspect allows listening to the generated sound. By increasing the window's size, the user can:

- Display the sound signal, and if needed perform measurements on it.
- Display the object's movements currently calculated – and access facilities such as zooming and rotating the display, or modifying the simulation speed to slow down or accelerate observed movements. This visualization is of great help when the model's behavior needs to be understood.
- Display some measurements made on the object as time signal – for example, display dynamically the position on a LNL non-linear interaction curve.

7 Applications and results

We will give a demonstration of the software and thereby illustrate some of the creation processes it allows.

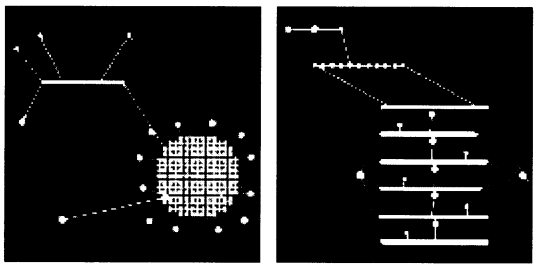


Figure 9.
Example of complex structures.

The design of Genesis has been made through years with a deep collaboration with composers: Ludger Brummer, Hans-Peter Stubbe, Giuseppe Gavaza, Claude Cadoz for example. The version 1.5 of GENESIS is now used in music creation centers for creation and formation purposes by an increasing number of people. Many musical pieces were created using the software, and some extracts and underlying physical structures will be commented briefly during presentation. As an interesting result, they demonstrate that the mass-interaction paradigm is not only a new sound synthesis technique but may have other musical uses: when wanted, it is possible to perform a whole creation process with it, especially in the GENESIS environment.

8 Conclusion and future works

Future works relative to Genesis will investigate some new features, especially those of a *performance* mode and those concerning real-time simulation. The 1.5 version of the GENESIS environment is however yet available on the SGI Unix platforms with the features described above. Users have expressed their satisfaction with using the software and investigating new creative processes based on the mass-interaction representation of matter.

10 Acknowledgments

The work of the ACROE is supported by the French Ministry of Culture and Ministry of Research.

The authors wish to acknowledge the developers who helped in designing GENESIS and all the music students and composers they worked with - especially those whose works will be heard during presentation.

References

Adrien J.M. 1991. "The Missing Link: Modal Synthesis". Representation of Musical Signal, G.. De Poli, A. Picialli, and C. Roads, eds, Cambridge, Massachusetts, MIT Press.

Borin G., De Poli G. and Sarti A., 1992: "Algorithms and Structures for Synthesis Using Physical Models" - Computer Music Journal 16(4):30-42.

Cadoz C., Luciani A. and Florens J. L., 1993: "CORDIS-ANIMA: A Modeling and Simulation System for Sound and Image Synthesis - the General Formalism." . Computer music journal 17(4):19-29.

Castagne N. and Cadoz C. 2000. "Physical modeling Synthesis: Balance Between Realism and Computing Speed" - Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFX-00), Verona, p.

Djoharian P. 1993: "Generating Model for Modal Synthesis". Computer Music Journal 17(1).

Florens J.L., Cadoz C and Luciani A 1990: "A Modular Feedback Keyboard Design". Computer Music Journal 14(2):47-51.

Fourcade P.and Cadoz, C., 1996. "Sound Synthesis by Physical Modeling: an Elementary Stricker". Proceedings of the Acusticum Forum. Acta Acustica, international Journal of Acoustics.

Fourcade P., 2001. "Etude et simulation des phénomènes percussifs dans les instruments de musique". Thèse de l'Université de la Méditérannée, France.

Incerti E., 1996. "Synthèse de sons par modélisation physique de structures vibrantes : application pour la création musicale par ordinateur". Thèse d'informatique de l'Institut National Polytechnique de Grenoble, France.

Jaffe A. 1995. "Ten Criteria for Evaluating Synthesis Techniques". Computer Music Journal 19(1):76-87.

Pearson M., and. D. M. Howard, 1996. "Recent developments with the TAO physical modelling system" Proceedings of the International Computer Music Conference. International Computer Music Association, pp. 97-99.

Risset J.C. 1990. "Modèles physiques et perception, Modèles physiques et composition". Colloque Modèles Physiques, Création Musicale et Ordinateurs vol2 p 539 - Edition de la Maison des Sciences de l'Homme – Grenoble, France.

Smith III J.O., 1996. "Physical Modelling Synthesis Update". Computer Music Journal 20(2).

Vinet, Delalande & al. 1999. "Interfaces homme-machines et création musicale". Vinet H and Delalande F, directors. ed. Hermes, Paris.