# How Can We Model Behaviours of Digital Instruments?
# Propositions in fuzzy logic from Philippe Manoury's works

**Alain Bonardi**

CICM (Centre de recherche
Informatique et Création Musicale)
Université Paris 8
2 rue de la Liberté
93526 Saint-Denis Cedex 02
France

alain.bonardi@gmail.com

**Frédéric Dufeu**

CeReNeM
(Centre for Research in New Music)
University of Huddersfield
Creative Arts Building, Queensgate
Huddersfield HD1 3DH
United Kingdom

f.dufeu@hud.ac.uk

## Abstract

For musicologists, the computing environment dedicated to the performance of an interactive work of mixed music may be an important object for analysis. Studying directly the program upon which the performance is based enables to observe with no ambiguity the consequences of incoming information, derived from the instrument or the voice, on the musical and sound processes. The aim of such an organological approach is the evaluation of the expressive range of the digital instrument on its own or combined with other ones and, beyond, the potential for the interpretation of the musical work. In cases where computer programs are available and still exploitable, their high degree of individualization raises an important issue for the musicological study of a set of works: from one composition to another, units for musical or sound processing and their interconnections are essentially variable in nature and in implementation. Although instrumental behaviours encountered throughout the repertoire of mixed music have many common properties, each study may call for an entirely new approach, with no significant benefit from earlier works.

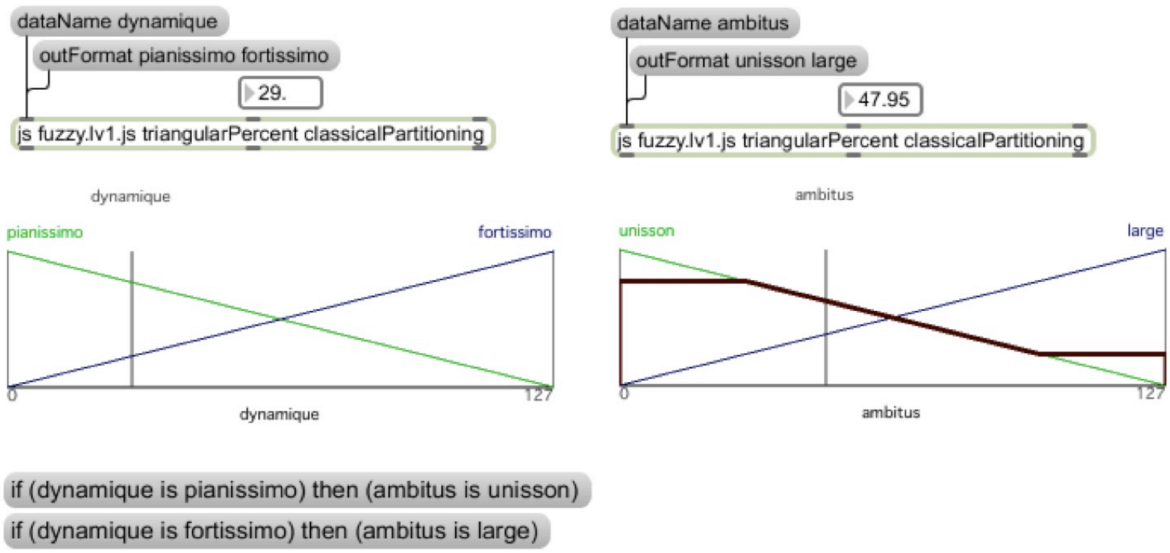
The purpose of our communication is to present perspectives for modelling behaviours of digital instruments for musical performance. We rely on an experiment calling for formalisms of fuzzy logic, applied to the third section of *Pluton* by Philippe Manoury. We use the FuzzyLib library developed by Alain Bonardi and Isis Truck for the Max environment and hence directly integrable to the patch for the performance of *Pluton*. In the framework of computing with words as described by Lofti Zadeh, this formalism enables a semantic description of the input and output phenomena of processing modules, getting back from physical measurements to a perception-based appreciation. It is also possible to make inferences at this semantic level. In fuzzy logic, it is possible to work on the musical vocabulary of dynamics and tempi (*pianissimo*, *moderato*, etc.) in a non-binary way, rather than exclusively on ranges of numeric values, and to describe transfer functions with fuzzy rules.

In the example below, two linguistic variables called *dynamique* (dynamics) and *ambitus* (range) are modelled within the third section of Manoury's *Pluton*. They have each two

classes, associated to fuzzy subsets, represented on the figure by ascending and descending functions corresponding to a degree of truth: the first function relies on notions of *pianissimo* and *fortissimo*; the second function relies on *unisson* (unison) and *large*. Two fuzzy rules have been established, connecting the two variables:

- if *dynamique* is *pianissimo*, then *ambitus* is *unisson*
- if *dynamique* is *fortissimo*, then *ambitus* is *large*

An inference engine, based on the principle of the general modus ponens, implements the reasoning and allows to deduce the range as a function of the input value for dynamics.




More generally, when a variable has five classes (for instance, for dynamics, *piano*, *mezzopiano*, *mezzoforte*, *forte*, *fortissimo*), a training algorithm has been developed by Isis Truck to shape and distribute automatically the fuzzy subsets, from experimental data directly taken from the studied patch.

This approach enables to model the controlling methods and the behaviour of the patch under terms linked to perception, and even to replace it or to contribute to its perpetuation, as it is a form of writing that is abstract from a given implementation in a particular software environment. From a particular case of man-machine interaction present in a work that is emblematic of real-time interaction, we discuss the relevance of our modelling for the analysis of mixed music and its perspectives for developed applications of formalisms that can be generalised to a large part of the digital instruments in the contemporary repertoire. This approach constitutes a contribution to the organology of real time devices for sound and music production.

## Introduction

Computing and software environments dedicated to artistic creation are of great musicological interest in the study of works of the electroacoustic repertoire. On both the composition and the performance sides, programs developed by composers with or without the support of a scientific and technical team reflect the aesthetic intentions, promises and limits involved in the resulting work. When these environments are available to the analyst and still exploitable, their thorough study enables to evaluate some important compositional decisions and, for the performance of mixed works based on a real time interaction between one or several performers and electroacoustic materials, the consequences of incoming information derived from the instrument or the voice on the musical and sound processes. Such an approach, that we call in this paper digital organology, can find its theoretical roots in Marc Battier's article "A Constructivist Approach To The Analysis of Electronic Music and Audio Art – Between Instruments and *Faktura*", published in *Organised Sound* in 2003[1]. Its relevance on actual analyses has been demonstrated in several musicological works, amongst which Kevin Dahan and Olivier Baudouin's studies and reconstructions of John Chowning's *Stria* published in a same issue of the *Computer Music Journal*[2], Michael Clarke's interactive aural analyses of Jonathan Harvey's *Mortuos Plango, Vivos Voco*[3] and Denis Smalley's *Wind Chimes*[4], or Andreas Bergsland's emulation of the sounds of Paul Lansky's *Six Fantasies on a Poem by Thomas Campion*[5]. As these research works have focused on one single work, a question arises: how can we develop analytical methods for investigating digital technologies applied to musical creation that could be generalised to a whole corpus? The high degree of individualization of pieces of software raises an important issue: from one composition to another, units for musical or sound processing and their interconnections are essentially variable in nature and in implementation. Although instrumental behaviours encountered throughout the repertoire of mixed music have many common properties, each study may call for an entirely new approach, with no significant benefit from earlier works. The two authors of this paper have contributed to some of the studies which, often concerned with preservation issues, have been undertaken to sketch typologies for digital instruments[6], as well as to consider the digital instrument as a tool for music analysis[7]. But few attempts have been made to develop, in a musicological and analytical context, methods that aim in the first instance at being generalisable to a broad set of works.

[1] Marc Battier, "A Constructivist Approach To The Analysis of Electronic Music and Audio Art – Between Instruments and *Faktura*", *Organised Sound*, 8(3), 2003, pp. 249-255.
[2] Kevin Dahan, "Surface Tensions: Dynamics of *Stria*", *Computer Music Journal*, 31(3), 2007, pp. 65-74; Olivier Baudouin, "A Reconstruction of *Stria*", *Computer Music Journal*, 31(3), 2007, pp. 75-81.
[3] Michael Clarke, "Jonathan Harvey's *Mortuos Plango, Vivos Voco*", in Mary Simoni (ed.), *Analytical Methods of Electroacoustic Music*, New York, Routledge, 2006, pp. 111-143.
[4] Michael Clarke, "*Wind Chimes*: An Interactive Aural Analysis", in Évelyne Gayou (ed.), *Denis Smalleys*, Paris, Ina-GRM, "Polychrome Portraits", n° 15, 2010, pp. 35-57.
[5] Andreas Bergsland, "The Six Fantasies Machine – an instrument modelling phrases from Paul Lansky's *Six Fantasies*", in *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME2011)*, Oslo, 2011, pp. 523-526.
[6] Serge Lemouton, Raffaele Ciavarella, Alain Bonardi, "Peut-on envisager une organologie des traitements sonores temps réel, instruments virtuels de l'informatique musicale ?" in *Proceedings of the Fifth Conference on Interdisciplinary Musicology (CIM09)*, Paris, 2009, n. p., http://cim09.lam.jussieu.fr/CIM09-en/Proceedings_files/49A-Lemouton%26al_v03.pdf (last visited June 20th, 2013); Jérôme Barthélemy, Alain Bonardi, Yann Orlarey, Serge Lemouton, Raffaele Ciavarella, Karim Barkati, "Towards An Organology Of Virtual Instruments In Computer Music", in *Proceedings of the 2010 International Computer Music Conference (ICMC2010)*, New York, 2010, pp. 369-372.
[7] Frédéric Dufeu, "L'instrument numérique comme objet d'analyse des musiques mixtes", in *Proceedings of Journées d'Informatique Musicale (JIM10)*, Rennes (France), 2010, pp. 221-227.

In this paper, we present perspectives for modelling behaviours of digital instruments for musical performance[8], many of which rely on common software environments that favour generalisation attempts, such as Max, Pure Data, or SuperCollider. We introduce an experiment calling for the formalisms of fuzzy logic, and apply it to the third section of *Pluton* by Philippe Manoury, by using the FuzzyLib library previously developed for the Max environment. In the framework of computing with words as described by Lofti Zadeh, this formalism enables a semantic description of the input and output phenomena of processing modules, allowing the analyst to get back from physical measurements to a perception-based appreciation.

## 1. Computing with words and fuzzy logic

One recurring issue encountered in the investigation of digital musical instruments is the fact that, even though the composer, the performer, the analyst and the listener consider most phenomena in musical, verbal, and semantic terms, the information carried by the software is raw digital data. This statement can directly be put in relation to the computer scientist Lofti Zadeh's reference article on the concept of computing with words (CW or CWW)[9]. As Zadeh himself states,

> In its traditional sense, computing involves (for the most part) manipulation of numbers and symbols. By contrast, humans employ mostly words in computing and reasoning, arriving at conclusions expressed as words from premises expressed in a natural language or having the form of mental perceptions. As used by humans, words have fuzzy denotations. The same applies to the role played by words in CW.[10]

In 2007, Jerry Mendel commented Zadeh's article as follows, emphasizing the close relationship between computing with words and fuzzy logic:

> Of course, he did not mean that computers would actually compute using words, a single word or phrase, rather than numbers. He meant that computers would be activated by words, which would be converted into a mathematical representation using fuzzy sets (FSs) […], and that these FSs would be mapped by means of a CWW engine into some other FS, after which the latter would be converted back into a word […]. FL [fuzzy logic] is viewed in [aforementioned] publications as the machinery that will let 'input' words, which are provided by a human, be transformed within the computer to 'output' words, which are provided back to that human, or to other humans. […] Potential applications for CWW are many and include Web-based searches, summarizations, subjective judgments, subjective decisions, etc. […] A person interacting with such a CWW computer interface would not be concerned with the CWW engine, but would only be interested in knowing the output word for their input word.[11]

---

[8] The instrumental status of electronic and digital devices for sound production and musical performance is largely discussed in the field of electroacoustic music studies, for instance in Claude Cadoz, "Musique, geste, technologie", in Hugues Genevois and Raphaël De Vivo (eds), *Les nouveaux gestes de la musique*, Marseille, Éditions Parenthèses, 1999, pp. 47-92. Throughout our paper, we adopt a broad definition where the expression "digital instrument" designates a functional unit for sound generation and processing, to which the composer provides sequences to be executed during the performance.
[9] Lofti A. Zadeh, "Fuzzy Logic = Computing with Words", *IEEE Transactions on Fuzzy Systems*, 4(2), 1996, pp. 103-111.
[10] *Ibid.*, p. 103.
[11] Jerry M. Mendel, "Computing with Words: Zadeh, Turing, Popper and Occam", *IEEE Computational Intelligence Magazine*, 2(4), 2007, p. 11.

The mapping between fuzzy logic and subsets and computing with words can interestingly be considered in the perspective of Philippe Manoury's theorisation of his own approach of the computer for real time performance, known through his conceptualisation of "virtual scores" (*partitions virtuelles*). After the technological issues linked to score following and the recognition by the machine of events played by the performer that are conform to a memorised score, as then explored by Barry Vercoe and Miller Puckette[12], Manoury elaborated a form of interaction that is based on the discrepancy between the inherently fuzzy character of dynamics and durations as understood and played by performers and the absolute values of digital encoding:

> The piano is fitted with Midi sensors that enable the computer to recognise all produced notes, durations and intensities. Such a device can recognise, for instance, 128 ways to play the keyboard, from the weakest to the loudest intensity. You can then take all the information provided by the performer (durations and intensities) and affect it to the control of some electroacoustic process. That way, dynamics can control the spatialisation, transpositions, modulations, rhythms […] of the 4X's production. Initially, I undertook a theoretical reflexion on the notion of interpretation as detected by a machine, which led to the concept of 'virtual scores'. In fact, that is about conceiving scores (heard as the set of sound events expected in a piece) for which some components are not fixed a priori, but wait for some information from the instrument to be updated. In *Pluton*, all that is produced by the 4X is derived from the piano part in real time […].[13]

From the case of Manoury's *Pluton* and, more broadly, the *Sonus ex machina* cycle[14] and his later electroacoustic works, the concept of computing with words and fuzzy logic can be applied to describe and analyse the behaviour of digital instruments, in a way that can then be extended to other composers's works for real time. For instance, other applications of the concept of virtual scores can be found in Brice Pauset's *Perspectivæ Sintagma I*, for Midi piano and real time electronics:

> […] the performance of the pianist is constantly compared to the ideal, geometric score stored in the computer. The infinitesimal gaps between the performer and the score are injected in a

---

[12] Cf. Barry Vercoe and Miller Puckette, "Synthetic Rehearsal: Training the Synthetic Performer", in *Proceedings of the 1985 International Computer Music Conference (ICMC)*, Burnaby (Canada), 1985, pp. 275-278; Miller Puckette and Cort Lippe, "Score Following in Practice", in *Proceedings of the 1992 International Computer Music Conference (ICMC)*, San Jose (CA), 1992, pp. 182-185.

[13] "Le piano est équipé de capteurs Midi qui permettent à l'ordinateur de reconnaître toutes les notes, durées et intensités qui sont produites. Un tel système reconnaît, par exemple, 128 manières différentes d'attaquer le clavier de la plus faible à la plus forte intensité. Il s'agit alors de prélever toutes les données fournies par l'interprète (durées et intensités) et de les affecter au contrôle d'un processus électroacoustique quelconque. Ainsi, les dynamiques peuvent contrôler suivant les cas la spatialisation, les transpositions, modulations, rythmes [...] de ce que fournit la 4X. Au départ, j'ai mené une réflexion théorique sur la notion d'interprétation pouvant être détectée par une machine, et qui a abouti au concept de "partitions virtuelles". Il s'agit en fait de concevoir des partitions (ce terme étant entendu comme l'ensemble des événements sonores voulus dans une composition) dont tous les composants ne sont pas fixés a priori, mais qui attendent une information venue de l'instrument pour pouvoir être exécutés. Dans *Pluton*, tout ce que produit la 4X est dérivé de la partie pianistique en temps réel […]." Philippe Manoury, "*Pluton* par Philippe Manoury", in Marc Battier *et al.* (eed.), *PMA LIB. Les musiques électroniques de Philippe Manoury*, CD-ROM, Paris, Ircam – Centre Georges Pompidou, 2003, pp. 2-3. We translate.

[14] *Sonus ex machina* is constituted with four works for instruments and real time electronics: *Jupiter* (flute, 1987), *Pluton* (piano, 1988), *La partition du ciel et de l'enfer* (flute, two pianos, orchestra, 1989) and *Neptune* (three percussionists, 1991).

> real time composition algorithm, which reproduces with synthesis the same processes as those of the written score.[15]

More generally, a modelling of digital instruments according to semantics that are linked to perception rather than to a physical evaluation of digitalised streams of numbers may be relevant to undertake and disseminate analyses, as well as to provide representations of what is going on in the electronic part of the work. In the following section, we present the FuzzyLib library for Max, which can be directly inserted into performance patches for organological and musicological purposes.

## 2. Musicological Implications of the FuzzyLib library

Developed by Alain Bonardi and Isis Truck within a collaboration between the Real-Time Musical Interactions (IMTR) team of Ircam and Université Paris 8 from 2007 to 2009, FuzzyLib is a JavaScript based library for the Max environment that "enables to implement fuzzification, uncertain reasoning and defuzzification for any number of data"[16] that can be found in an editable Max patch. The three main objects of this library (*lv*, *gmpa*, *ruleComposer*) correspond to three steps generally involved in fuzzy logic: fuzzification, reasoning and defuzzification, fuzzy rule expression. In a paper published in 2010, Bonardi and Truck elaborate on each of these steps:

> The first step in fuzzy logic is fuzzification. The purpose is the semantic representation of a phenomenon usually handled with digital values. Thanks to a range of values provided by an expert and to a certain (generally odd) number of fuzzy subsets, one can represent this phenomenon as a linguistic variable. We have designed an object named *lv*, which stands for 'linguistic variable'. […] Reasoning is computed thanks to the generalized modus ponens that enables to solve the following syllogism: if we assume A => B and we get A' close to A as input, what should the answer B' be? We have developed an object named *gmpa* […], which stands for 'generalized modus ponens application'. […] The object also provides defuzzification (by barycentre or maximum methods) for linguistic variables implied as consequents of fuzzy rules. […] We also have implemented an interface object named *ruleComposer*, associated with the 'gmpa' object to help users to write fuzzy rules and avoid syntax errors. This object automatically collects the names of the linguistic variables declared with 'lv' objects and provides help to write fuzzy rules.[17]

In the context of musicological investigation, the *lv* object along with a visualisation object ('fuzzy.FSSdrawing.js') can be inserted in a Max patch as illustrated in figure 1.

[15] "[…] le jeu du pianiste est constamment comparé à la partition idéale, géométrique, stockée dans l'ordinateur. Les décalages infimes entre l'interprète et la partition sont insérés dans un algorithme de composition en temps réel reproduisant par synthèse les mêmes processus que ceux de la partition écrite." Brice Pauset, program notes for *Perspectivæ Sintagma I*, 1997, http://brahms.ircam.fr/works/work/11027/ (last visited June 20th, 2013). We translate.

[16] "FuzzyLib – IMTR", presentation and download webpage of the FuzzyLib library, http://imtr.ircam.fr/imtr/FuzzyLib - What_is_Fuzzy_Logic.3F (last visited June 20th, 2013).

[17] Alain Bonardi and Isis Truck, "Introducing Fuzzy Logic and Computing with Words Paradigms in Realtime Processes for Performance Arts", in *Proceedings of the 2010 International Computer Music Conference (ICMC 2010)*, New York, 2010, pp. 474-475.

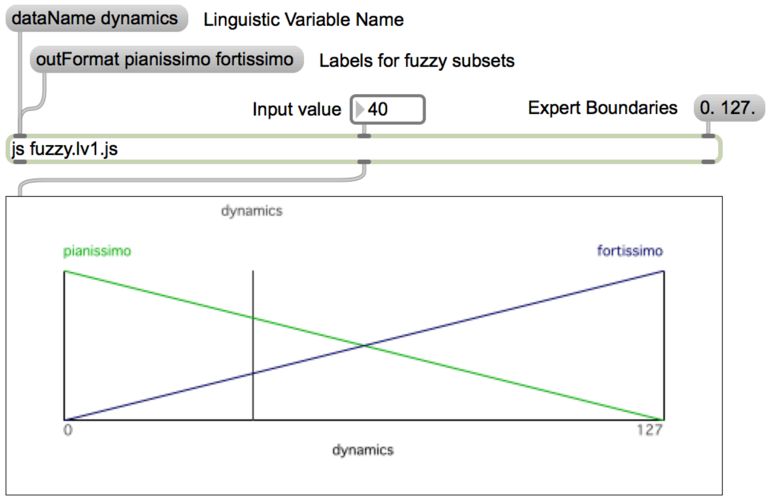



**Figure 1.** Linguistic variable object with visualisation

The *lv* objects expects three types of information: a linguistic variable name that is the practical concept to be handled (with the message selector 'dataName', here *dynamics*), a set of label names for the fuzzy subsets[18] of the linguistic variable (with the message selector 'outFormat', here *pianissimo* and *fortissimo*), and expert boundaries for the incoming value (in the third inlet of the object, here 0 and 127 corresponding to the minimum and maximum possible Midi values for velocity). The object can then receive a stream of incoming velocity values directly from the environing patch. Rather than a consideration of the 40 velocity input value, the visualisation enables to observe data according to the fuzzy subsets *pianissimo* and *fortissimo*. In an actual organological situation, a more relevant number of subsets can be set, as shown in figure 2.

As a starting point, fuzzy subsets cover uniformly the range defined by the expert, but usually the distribution of values for actual phenomena is not uniform. A training mode therefore enables to adapt the ranges of the fuzzy subsets according to a specific context. Figure 3 shows the state of the *lv* object after receiving velocity values from an actual instrumental performance and learning their distribution.

[18] A fuzzy subset is a function expressing a degree of truth of a perceptible state of the linguistic variable according to the position in the expert domain. This function is also called membership function.

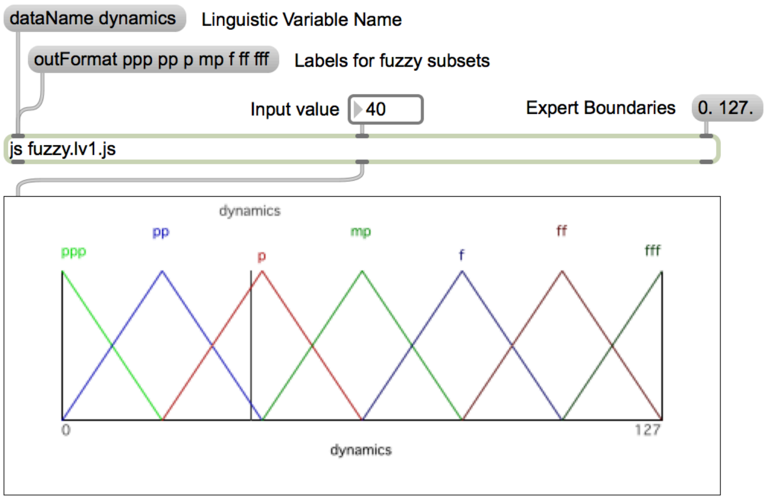



**Figure 2.** Seven fuzzy subsets corresponding to a standard range of dynamics

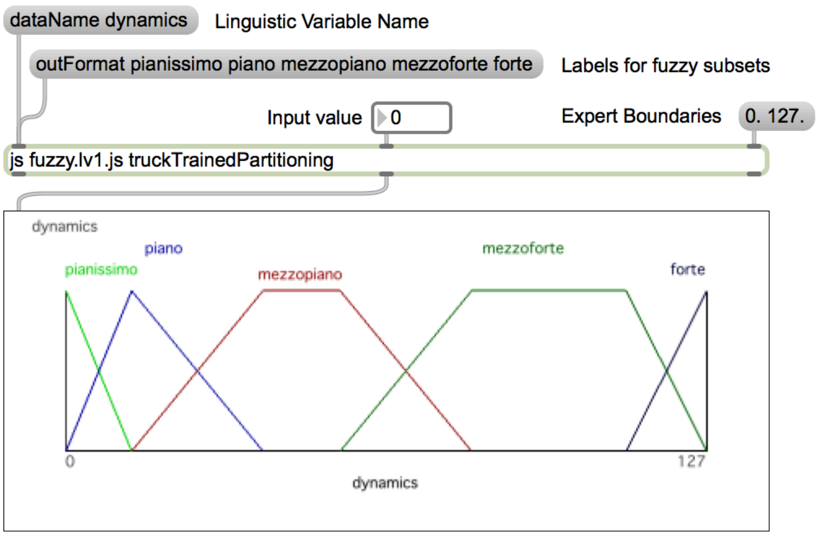



**Figure 3.** State of the *lv* object after an actual performance training

The *gmpa* (*generalized modus ponens application*) object enables to map input and output data, both expressed as linguistic variables, according to rules of inference. In the example illustrated on figure 4, velocity as input data has the linguistic name *dynamics*, with two fuzzy subsets *pianissimo* and *fortissimo*; range as output data has the linguistic name *range*, with two fuzzy subsets *unison* and *large*. Both input and output data have numeric expert boundaries 0 and 127. They are interconnected by the 'js.fuzzy.gmpa1.js' object that has received the two simple following fuzzy rules: if (dynamics is pianissimo) then (range is unison); if (dynamics is fortissimo) then (range is large). The interaction works in real time: rather than being programmatically expressed by a linear transfer function, the mapping is done through the inference rule and enables an active and perceptive consideration of both input and output according to linguistic variables.

## 3. Applications to Manoury's *Pluton*

The third section of Philippe Manoury's *Pluton* offers a useful context to integrate fuzzy logic in an actual digital instrument. This section is ruled under a semi-open form and is based on three types of sequences, plus a terminating coda. The first type is a set of recording sequences (labelled R), the contents of which are recorded within the software during the performance. The second type is constituted with three sequences described by Manoury as "functional" and labelled E, M, and S, respectively. Each of these sequences has an opening note, a closing note, and swappable notes with free dynamics for sequences E and M. The sequence E (for *échelle*: scale) enables the pianist to vary the range of the electroacoustic production: if swappable notes are fortissimo, the range is unchanged; when pianissimo, the range is compressed and tends towards an unison around e-flat. The sequence M (for *mixage*: mixing) enables, according to the same principle, to control the balance between dry recorded sound (*fortissimo*) and reverbered sound (*pianissimo*). The sequence S (for stop) simply shuts one or both of the two Markov-generated sequences, which belong to the third main type of sequences of section 3.

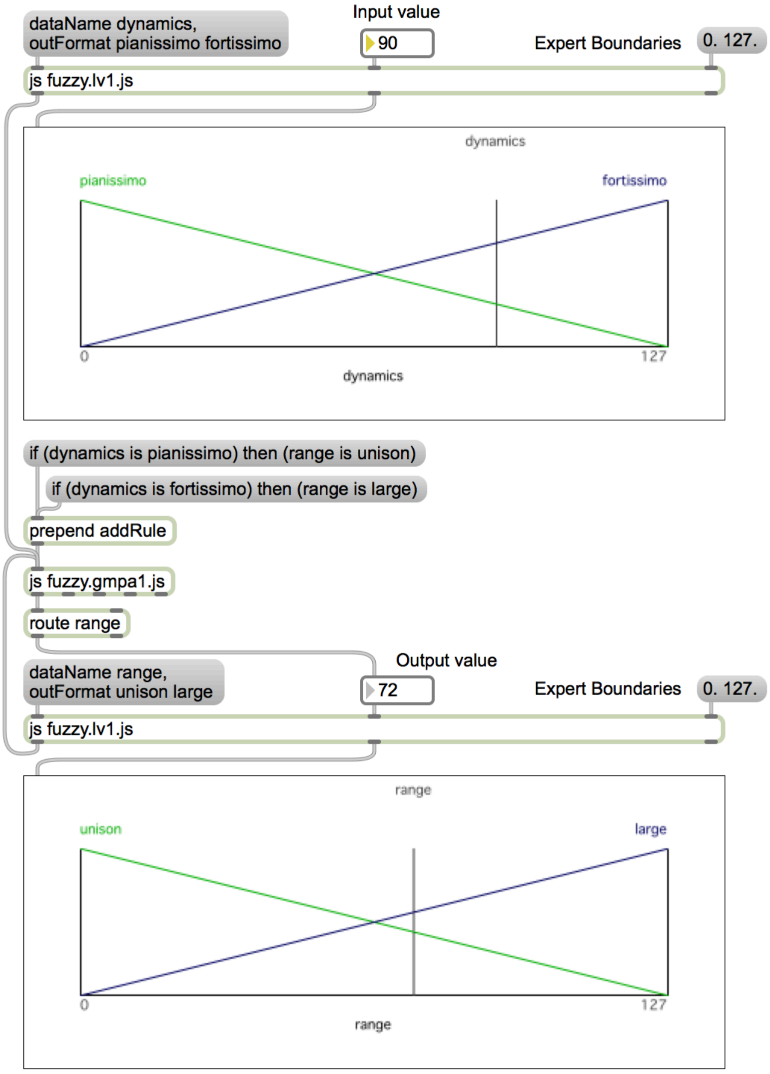



**Figure 4.** Input dynamics mapped to output range via the generalized modus ponens object

In sequences E and sequences M, the direct relationship between input velocities and electroacoustic processes raises the consideration of an integration of the FuzzyLib objects in the Max patch upon which relies the instrumental behaviour of *Pluton*. According to the configurations described above, two simple output linguistic variables and inference rules can be set: for sequence E, if (dynamics is pianissimo) then (range is large); if (dynamics is fortissimo) then (range is unison) – as in the aforementioned example. For sequence M, if (dynamics is pianissimo) then (output sound is reverbered); if (dynamics is pianissimo) then (output sound is dry). The numeric transfer functions of such mappings are visible in two tables encapsulated in the Max patch for the performance of *Pluton*[19], as seen in figure 5.

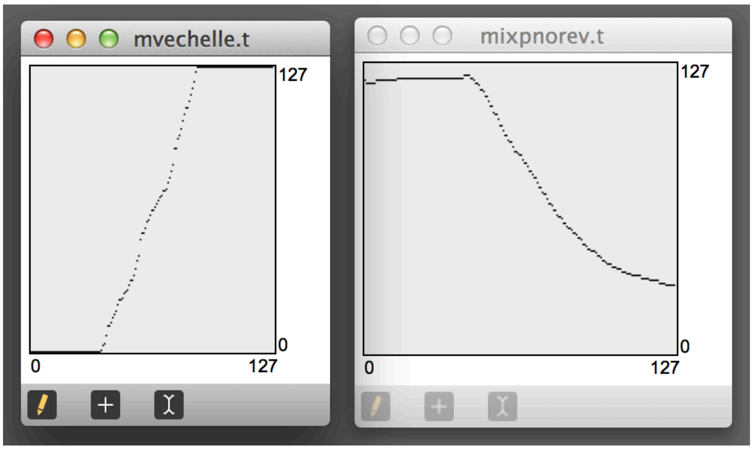



**Figure 5.** Tables mapping digital values of incoming velocity to range (left) and reverberation amount (right). The x axis corresponds to incoming velocities (0: *ppp*, 127: *fff*). On the y axis, 0 corresponds to unison (left) and no reverberation (right); 127 to unchanged range (left) and full reverberation (right).

During a performance or, more generally, any use of the patch including a rehearsal or a simulation, the original state of the program only enables to observe the information chain and to hear the actual mapping between input gestures and output sound production. A first step in the investigation of the instrumental behaviour can be the localisation of the tables and the integration of number boxes after the Midi receiver connected to the piano and before the DSP range scaler and reverb amount. However, as the patch for *Pluton* has only been designed for production purposes – as is the case for most performance programs, its exploration and modification is not optimised and a refactorisation is helpful in the investigation process. In this approach, modelling the instrument by making it more human readable, flexible and abstract from the production needs may benefit from a fuzzy logic implementation, which provides at the very level of the program linguistic and perception-based measurements, more directly exploitable for analytical purposes. Assuming that all mappings found in a given instrument can be consistently translated into linguistic variables and simple or complex inference rules, such a reimplementation enables to model the instrument so as to explore its expressive range, beyond the existing recordings of a work, leading to a better understanding of the aesthetic choices of both the composer and various performers.

[19] The Max patch discussed here is named ‘PLUTON2008_agora’ and has kindly been provided by Alain Jacquinot, then director of production at Ircam, in September 2009.

## Conclusion

As most digital instruments dedicated to the performance of contemporary electroacoustic works are implemented under various software environments and following different programming styles, studying a broad set of works and devices may call for a new approach and methodology on each case. However, if these instruments are highly individualised to the needs of one given composer and musical work, common characteristics can be found amongst large corpuses. Behaviours are often ordered according to more or less complex strategies of mappings between the information derived from the gestures of the performer and the final sound outcomes. An analysis and description of these behaviours may involve a research that requires engineering and computer science skills, but an attempt to generalise digital organology can be turned into an opportunity to develop tools that can be both applied beyond a specific case and designed in a form that is more human readable than what is directly given by the original software, hence made accessible to a wider range of musicologists. In this matter, computing with words, fuzzy logic and their instantiations within the FuzzyLib library for Max, an environment largely used throughout the electroacoustic repertoire, prove useful as they can be directly integrated in the environment, applied to any describable input and output parameter, and provide visual measurements in real time.

Although attempts to model digital instruments such as the one presented in this paper hit an important limit in the assumption that the computer environments for performances have to be both made available by the composer and still exploitable – obsolescence being a recurring issue for creation and for analysis, the general principle of integrating linguistics and fuzzy logic in the instrument itself shall not be regarded as being limited to the Max based environments. Any accessible programming language dedicated to musical creation potentially lends itself to such an approach, and libraries implementing the same features as FuzzyLib could be either adapted[20] or developed in Csound, Pure Data or SuperCollider. If the musicological investigation of the new instruments of the contemporary repertoire may be facilitated by semantic descriptors interfacing between the low-level behaviour and the analyst, this translation could also be extended towards the original development process itself, and benefit to composers, patch designers and performers in their apprehension and understanding of the potential of musical expression of their instruments.

## References

BARTHÉLEMY Jérôme, BONARDI Alain, ORLAREY Yann, LEMOUTON Serge, CIAVARELLA Raffaele, BARKATI Karim, "Towards An Organology Of Virtual Instruments In Computer Music", in *Proceedings of the 2010 International Computer Music Conference (ICMC2010)*, New York, 2010, pp. 369-372.

BATTIER Marc, "A Constructivist Approach To The Analysis of Electronic Music and Audio Art – Between Instruments and *Faktura*", *Organised Sound*, 8(3), 2003, pp. 249-255.

BAUDOUIN Olivier, "A Reconstruction of *Stria*", *Computer Music Journal*, 31(3), 2007, pp. 75-81.

[20] Fuzzy Lib has been developed in Javascript for Max but is defined as a set of object-oriented classes, enabling the code to be adapted to other languages.

BERGSLAND Andreas, “The Six Fantasies Machine – an instrument modelling phrases from Paul Lansky’s *Six Fantasies*”, in *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME2011*), Oslo, 2011, pp. 523-526.

BONARDI Alain and TRUCK Isis, “Introducing Fuzzy Logic and Computing with Words Paradigms in Realtime Processes for Performance Arts”, in *Proceedings of the 2010 International Computer Music Conference (ICMC 2010)*, New York, 2010, pp. 474-477.

CADOZ Claude, “Musique, geste, technologie”, in GENEVOIS Hugues and DE VIVO Raphaël (eds), *Les nouveaux gestes de la musique*, Marseille, Éditions Parenthèses, 1999, pp. 47-92.

CLARKE Michael, “Jonathan Harvey’s *Mortuos Plango, Vivos Voco*”, in SIMONI Mary (éd.), *Analytical Methods of Electroacoustic Music*, New York, Routledge, 2006, pp. 111-143.

CLARKE Michael, “*Wind Chimes*: An Interactive Aural Analysis”, in GAYOU Évelyne (ed.), *Denis Smalley*, Paris, Ina-GRM, “Polychrome Portraits”, n° 15, 2010, pp. 35-57.

DAHAN Kevin, “Surface Tensions: Dynamics of *Stria*”, *Computer Music Journal*, 31(3), 2007, pp. 65-74.

DUFEU Frédéric, “L’instrument numérique comme objet d’analyse des musiques mixtes”, in *Proceedings of Journées d’Informatique Musicale (JIM10)*, Rennes (France), 2010, pp. 221-227.

“FuzzyLib – IMTR”, presentation and download webpage of the FuzzyLib library, http://imtr.ircam.fr/imtr/FuzzyLib - What_is_Fuzzy_Logic.3F (last visited June 20th, 2013).

LEMOUTON Serge, CIAVARELLA Raffaele and BONARDI Alain, “Peut-on envisager une organologie des traitements sonores temps réel, instruments virtuels de l’informatique musicale ?”, in *Proceedings of the Fifth Conference on Interdisciplinary Musicology (CIM09)*, Paris, 2009, n. p., http://cim09.lam.jussieu.fr/CIM09-en/Proceedings_files/49A-Lemouton%26al_v03.pdf (last visited June 20th, 2013).

MANOURY Philippe, “*Pluton* par Philippe Manoury”, in BATTIER Marc *et al.* (ed.), *PMA LIB. Les musiques électroniques de Philippe Manoury*, CD-ROM, Paris, Ircam – Centre Georges Pompidou, 2003.

MENDEL Jerry M., “Computing with Words: Zadeh, Turing, Popper and Occam”, *IEEE Computational Intelligence Magazine*, 2(4), 2007, pp. 10-17.

PAUSET Brice, program notes for *Perspectivæ Sintagma I*, 1997, http://brahms.ircam.fr/works/work/11027/ (last visited June 20th, 2013).

PUCKETTE Miller and LIPPE Cort, “Score Following in Practice”, in *Proceedings of the 1992 International Computer Music Conference (ICMC)*, San Jose (CA), 1992, pp. 182-185.

VERCOE Barry and PUCKETTE Miller, “Synthetic Rehearsal: Training the Synthetic Performer”, in *Proceedings of the 1985 International Computer Music Conference (ICMC)*, Burnaby (Canada), 1985, pp. 275-278.

ZADEH Lofti A., “Fuzzy Logic = Computing with Words”, *IEEE Transactions on Fuzzy Systems*, 4(2), 1996, pp. 103-111.