Teaching Pd & Using it to teach: Yet Another Pd Didactic Material for Computer Music.

Alexandre Torres Porres
USP (University of Sao Paulo) - Rua
do Matao 1010 Cidade Universitaria
05508-900 - São Paulo, SP - Brasil -
Caixa-Postal: 66281
55 11 8179-6226
pores@gmail.com

ABSTRACT

The focus is on my didactic work as a Computer Music/Pd teacher, in the form of interactive patches. But, on this paper, I also present some considerations about teaching Pd, and using it to teach Computer Music. This raises issues to the Pd Community, like if we can better promote Pd by investing on more didactic materials, as well as differences in using other softwares to teach. What I say mostly concerns Brazil and similar centers, but some considerations are applicable everywhere. I start by describing the scenario of studying Pd in Brazilian Universities, then teaching it and developing didactic materials for the needs I found.

Keywords

Puredata's Tutorials and Manuals. Didactics of Puredata, Computer Music didactical Materials.

1.STUDYING PD IN BRAZIL

Around 2002, on an undergraduate *Music Production* course at UFPR (State University of Paraná), I got to study Csound. You didn't hear much about Max/Msp or Pd throughout Brazil by then (today I know they have them, like in other centers in Brazil). Anyway, Csound is of a great didactic power, but, when I got to know about Max & Pd, it felt nicer because of their friendlier non-textual programming, and their real-time capabilities.

Luckily, I had friends who studied it and helped me out. This means that Max & Pd are not things you get to see in every Music Technology course in Brazil. People have to work mostly on their own (whoever has got that spirit)! Hence, generally speaking, users of Max/Pd (and alike), in Brazil, are determined professionals who have to overcome many difficulties, not to mention (humorously speaking) partially nerds, of course.

Learning Computer Music in Brazil is a bit hard. You must be on a major City/University to begin with. Unicamp (State University of Campinas) is one of the best infra structured Universities in Brazil. During my Master's research there, I got better acquainted with Max (it is one of the places with licensed and installed copies). But even if a center in Brazil has something like Max or other stuff, it doesn't mean that a lot of people are using it and studying it. In the end, it always depends if teachers take an effort to promote it, or if students want to bother learning at all.

So, depending on the strong influence of a teacher, or on the interest of students, a major attention might be given to a specific tool (be it Max, Csound, Matlab, Pd, whatever). Some personal influences can be passed on that way, in a sense that I ended up receiving the idea that Pd was not as "good", or as "professional" as Max, for example. I don't agree with that assertion, but cannot discuss it properly here. But, by checking Pd at first, I got a bit convinced of that, because of its "uglier" interface when compared to Max's, and that still seems to the argument of those who state that.

But the thing is, even if you are lucky to have an expensive tool like Max installed on a laboratory, it doesn't mean that students will go and buy a license too for themselves. I know only of one colleague of mine who did that!

So, how can we really learn and use Max? Not only it's hard to learn, it's hard to find someone who teaches it, it's hard to find a laboratory, and it costs a lot to have it on your own. The answer is, unfortunately, by using some older and pirate/cracked version of Max. And such a reality surely contributes for not being able to produce, or develop as much as in other centers.

When I wanted to implement some research on a patch, it felt like the right thing to do was to use a free and open source tool. Specially when I was researching my topic around and got a rather bad attitude by a Max user, who told me he "wasn't giving out his code" (as if he was ruining my plans to get rich on his expense). I used Pd, and anyone can check my dissertation and code for free now. It was the right thing to do after all. I know of a friend who ended up using Max, and felt a bit regret for this reason, because he can't really share it that well.

Despite the obstacles, many researchers in Brazil use Pd, learning it by struggling on their own. Gladly, we had some classes at Unicamp, taught by a doctorate student from our research group (although a teacher at another University). So this is how it works basically, some teachers might really know how to use the latest tools available. Sometimes it depends on the will of some students, who organize and create research groups, and teach

PdCon09, July 19-26, 2000, São Paulo, SP, Brazil.

every now and then. And despite all that, we get to see some academic research.

I must say that I am reporting this from the side of a Music Department. But there are the Computer Science departments. On Music Departments, Computer Music classes are not that well structured, or uniform in content on different Universities. If a particular teacher knows better how to use Csound, that is what will be taught/used with more emphasis. And there are simply no didactic material in Portuguese for musicians (saving us from complicated math).

Unfortunately, Computer Science Departments are not collaborating as much as they could with Music Departments to form actual Music Technology research groups. So I see many research with Pd that actually come from people with background in Computer Science. I am currently taking a course on Computer Music at the Computer Science Department at USP, we are using Octave, Csound and Puredata the most. Students are mostly people from the Computer Department with hardly no background in music.

In short, studying Computer Music is a relatively specialized area of work anywhere. It is normal for it to be less structured and formalized as other more traditional courses in Brazil. Not all go to a University to study it. Hence, many musicians who end up on a Computer Music class don't even want to learn it.

There is no common program that Universities share for Computer Music, it is just incipient, and depends on the knowledge and interest of particular teachers and students. It is hard for the students who really want to study it to be completely fulfilled. People with Computer Science background do better, but don't collaborate as much as they could with people with musical background.

Now, the usage of Free/Open Software is extremely pertinent in our reality, where people end up using cracked softwares, where institutions just can't really afford a good laboratory (even if some have some software licenses, it also doesn't mean they have good hardware/recording studio facilities). By using and teaching with Free Software, we get rid of moral and legal issues. In this sense, we will do better if we invest on the usage of Puredata. "If god gave us lemons, lets make some lemonade", and that's it.

But you can't really compare "lemons" and "apples"? I know that when institution can afford it, they will just use Both Pd and Max, and whatever else they can afford. So they go for "lemonade" or "apple pie" or even "else", depending on their project's needs (they are actually becoming more and more different tools, for even different ends, and the goal of this paper to discuss that, so I will move on).

All I have to say is that Pd is a great didactic tool for us, because it is Free, and it is also a good software to develop academic research. But the didactic aspect of Pd is not necessarily useful only for a center like Brazil. Just about anyone on the world can benefit from it. And I am emphasizing the usage of Pd because it is the only free/open software with a visual interface. And I am interested in reaching musicians with little background in Computer Science. It is easier than other textual tools like Csound/Supercollider, because students can relate to Pd as they relate to some end-user-oriented-software.

But there is a drawback for Pd users, especially for newbies. Because of its open and free development, there is no good or complete documentation about it. So people can get intimidated to get started with it. This will be further discussed later on.

2.TEACHING PUREDATA

Teaching Puredata is about the first step we must take in order to start developing a local community in Brazil. But we can also just use Pd to teach Computer Music or Music Technology in general. I choose to work mostly with Pd because it is a readily accessible Free tool for beginners in Computer Music. The only problem is that beginners might be intimidated by the lack of formal documentation and manual. But if I play the role of a Pd teacher, I can help with that.

I have been reaching people interested in learning Music Technology, which is a more general aspect of Computer Music. So I do use Pd as a didactic tool, as a mean to an end, and people can get more familiar with Pd depending on their own curiosity.

I also find people interested in learning Pd as an end for itself, and who are already determined to apply it in Computer Music. So there are basically two complementary aspects here: Using Puredata to teach & Teaching Puredata.

A lot of people interested in Music Technology are much more interested in the musical aspect of it than the technological one. So it's easy to scare them with programming.

There are also those who are musicians interested in learning Pd, but feel, nevertheless, unmotivated by having to learn some technical details, as they feel that is not the role of a musician. A midterm must be found, you have to provide just "enough" to satisfy and keep them motivated. And here is where I found the need to develop my didactic material.

There are also people who really want to learn Pd, and just need a hand to overcome their somewhat lazy spirit. For this case, the tutorials that come with Pd are great, it is just a matter of using it as the didactic material, which is free, and widely used. This particular group usually just needs someone to go over the tutorial with them, and that is it. Apart of that, we got Floss Manuals [1].

But the truth is that it isn't that simple to go over the whole Puredata audio tutorial on a simple workshop. There is the book by Miller Puckette [2] that deals with the theory behind the practical audio examples of the tutorial, and it is hard for beginners with background in Music to follow it.

So there are those who are really motivated to study Pd, and just need a hand at first. These would do fine with the tutorials, or someone to go over it with them. But it is easy to find that they don't know some important details, like what is the real and imaginary part of a complex signal.

Teaching Puredata itself is actually simple, the main structure of the language (like "Depth First") requires only a few patches to explain it. By the time you master the basics, the rest requires a good background in Music Technology, or Computer Science. We cannot expect that everyone who is interested in learning Puredata already knows All about Computer Music. If any student, in a few moments, would come up and say – *ok, that is how you do FFT on Pd, what's next? How can we do wavelets too?* – they surely don't need a teacher that badly.

So, my experience of using these two books [1, 2] is that you can easily teach and train someone to a beginner's level with the tutorials and Floss Manuals. After that, it is tricky. The tutorials along with Miller's book work for people who don't need teachers that much (if not at all), and who are people with background in Computer Music. In other words, they are potential researchers and developers.

So we have a gap here between users who are beginners and developers. Maybe that is ok, if one realizes that you cannot perform miracles, and magically turn a beginner into a developer. But I just see an opportunity here to fulfill this gap. I believe if we have more and more didactic materials out there, the easier it is to promote Puredata, and its Community, which shall increase as a consequence.

Obviously, we must have some criteria, to not redundantly repeat the same topics on different books, or make some mess by generating incoherence between them. So it is needed that people who are developing different didactic materials share their thoughts, and cover the topics in a complementary and coherent way. If we can overcome these issue, the more tutorials, books, manuals and didactic materials out there, the better. The more options we have to reach different niches. This should be part of the community's goal to better document Pd, to create a complete list of documented external objects, to work for the Pdpedia project <http://wiki.puredata.info>, things like that, in a sense that this kind of work can reach out for many important objectives.

3.YET ANOTHER DIDACTIC MATERIAL

There are a lot of books out there for Computer Music. I am developing another one on this topic, with examples in Puredata. Despite the fact that I write and develop patches to show how Pd works, it is not a book on Puredata itself, but on main topics of Synthesis and DSP. The goal is to teach musicians with no background in Computer Science and Programming. Puredata comes as the interactive didactic tool, showing the theory in practice. The Book is called: Computer Music for Beginners, with examples in Puredata,

It has been developed in Portuguese, the main references (besides the tutorials) are [1-5]. Floss Manuals and Miller's book are referred as complementary reading. Floss Manuals to help beginners get better acquainted with Pd, and Miller's book as an advanced reading. A lot has also been taken from Wikipedia <http://www.wikipedia.org/>. The most part was developed in 2007. I have only used it a couple of times in my classes. But now I am sharing it and publishing it online: <http://porres.googlepages.com/pd>.

I present topics such as: *Wavetable Synthesis - DC Offset – Modulation (Amplitude, Ring, Frequency, Pulse Width) – Additive/Subtractive Synthesis - Filters – WaveShaping - Spectral Analisys (FFT/Resynthesis) - Convolution & Convolution Reverb - Vocoder, Cross Synthesis - Phase Vocoder*. The text goes over the main parameters of these topics, a bit of their history, and the theory behind it. It is surely a work in progress, that will include other topics in the near future, and that shall be updated from time to time.

Some patches are really simple, and they are supposed to be that way in order to reach their didactic objective. Sometimes it is also not the case to just present an example of "How to Implement" Amplitude Modulation in Pd, for instance. I use comments to describe some important details of the Pd implementation, but the focus is to use the comments to talk about what Amplitude Modulation is about, complementing the text on the book.

I understand that this type of material is different than other Pd books out there. This is one of the objectives, to not generate conflict or redundancy. On the contrary, there is an intent to put this among other Pd books as a complementary publication, that hopes to collaborate with them and enrich the number of didactic materials out there. In a sense that other Pd books are (and even new ones must be) part of the referenced publications.

It is great to see that a new publication by Johannes Kreidler just came out [6], especially dedicated to show examples on how to program electronic music in Pd. It will definitely come in hand!

It is important to emphasize that my book is dedicated to musicians with little background on computer Science and Programming. So an effort is made to present the topics in an accessible way. The reason behind this is that I was myself once a musician interested in learning Computer Music. And most people I get to teach are in this condition. Not only that, but I still consider myself as a musician before all, so the theory must be clear to me when I read it as well. So the text slowly builds up the necessary background to get into more complicated issues.

A good example I must bring to this paper is the Phase Vocoder, which in fact had been a technique that belonged to DSP experts for a long time, and started to become more accessible only in the nineties. I must confess that it took me quite a while to understand it, and that I drowned in many technical details with little clue out of them. It is presented in Miller's book in an objective way, but the first time I read it, it just wasn't accessible to me (how many pmusicians know what a complex conjugate is?). What I had to do was to find other sources of information, and rebuild Miller's Patch by scratch. Check the figures below.

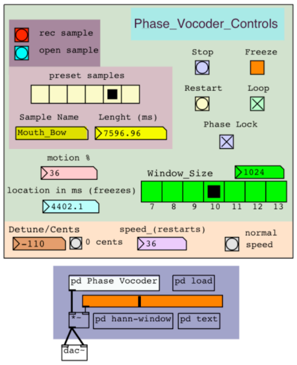

Figure 1 – Main patch window and controls

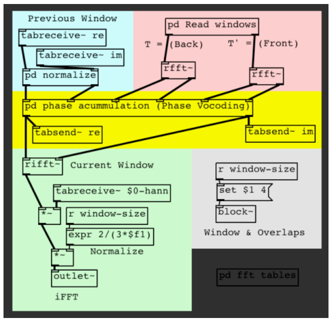

Figure 2 – pd Phase Vocoder subpatch, organized in sections.

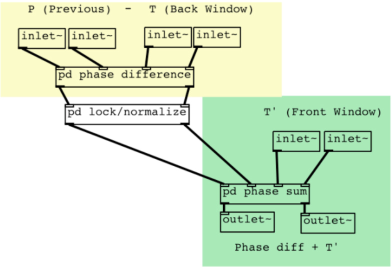

Figure 3 – pd phase accumulation subpatch window

So this is an example of a patch that comes in the tutorial and is explained in Miller's book. But struggled upon to be more accessible to beginners. Some say that things should be presented as simple as possible, but not simpler. It is surely not an easy task to find and draw this fine line. The result is that a lot of pages are dedicated for this particular topic, step by step.

Another example from my book is the part dedicated to Convolution and all the relative topics such as Vocoder and Cross Synthesis. Again, the motivation to write about it emerged from my own curiosity, as I had a lot of trouble finding out the differences between them, Actually, Convolution itself is not covered properly in the tutorial, and the other books/manuals on Pd I mentioned in this paper. And when trying to search in google about it, all I found was another person asking where to study and how to do it on Pd too.

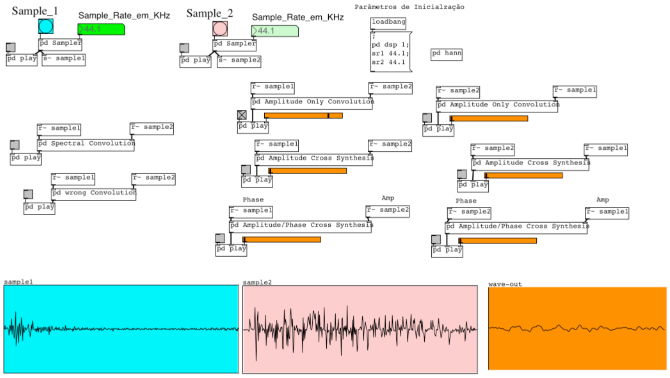

Figure 4 – Different Convolution techniques.

The figure above is an example from my book that presents two types of Convolution that are commutative, and three other examples of cross synthesis that aren't commutative (with orange sliders), so the second column of subpatches are just the inverted crossing of the loaded samples (blue and pink tables). This way, the student can check all the differences between each technique, and compare the result (orange table) by listening to it. Apart from this example, another chapter discusses Reverb Convolution, to fully cover all about this topic.

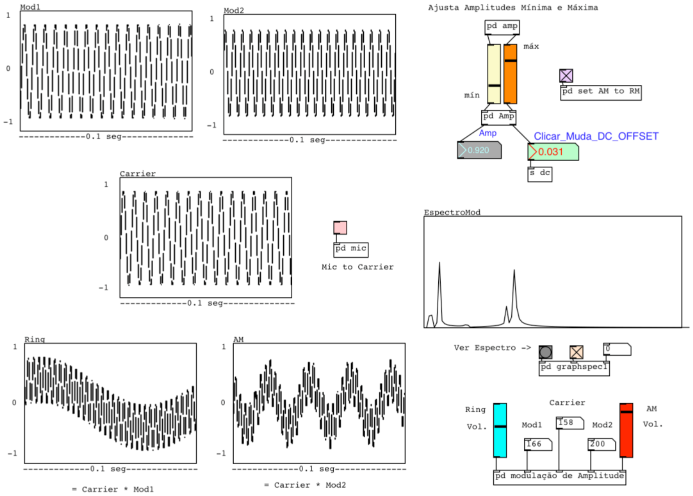

Figure 5 – Ring Versus Amplitude Modulation

The last example is on Amplitude Modulation. This patch compares and teaches about the difference in Ring Modulation (blue vertical slider on down-right corner) and Amplitude Modulation (red vertical slider on down-right corner). This is a really useful example to talk about DC Offset as well, and on the top right (yellow and orange sliders) I have a section that can manipulate DC Offset in a nice way.

4.FINAL CONSIDERATIONS

My book aims to people interested in learning Computer Music, who may or may not be acquainted with Pd. If not, other publications such as Floss Manuals play an important complementary role.

Choosing Pd for a didactical purpose is good, because it is open and free. A drawback for Pd newbies is actually the lack of a good documentation, so using it to teach can contribute against this problem. As community, we must invest in developing books and tutorials for Pd for better promoting it. Books can be on the same topic, if only they complement each other and attend different needs, the more the better. What we have is not enough yet, and some documentation projects need more attention, like Pdpedia. There is a sort of a tutorial boom now, we should enjoy the opportunity that different people want to contribute, organize and potentialize these efforts.

Another remark that points out of this paper is my belief that we should also develop plug-in packages in the same fashion as Max's pluggo. It would definitely complement the task of promoting Pd, and they could refer to tutorials and books about Pd. My Phase Vocoder, for example, could be presented as a user friendly software, or plug-in, developed in Pd. Other examples from my book could follow this trend.

5.ACKNOWLEDGMENTS

Thanks to all my teachers and friends in Brazil who struggle and collaborate to improve the Computer Music area.

6.REFERENCES

[1] Pd Floss Manuals, <http://en.flossmanuals.net/PureData>, Oßnline book.ß

[2] Puckette, Miller. Theory and Techniques of Electronic Music, <http://crca.ucsd.edu/~msp/techniques/latest/book-html> Online book.

[3] Roads, Curtis. The Computer Music Tutorial, MIT Press, Cambridge, MA, 1996

[4] Cook, Perry R. Real Sound Synthesis for Interactive Applications. A. K. Peters, Ltd. Natick, MA, USA, 2002.

[5] Moore, Richard F. Elements of Computer Music. UCSD, San Diego, CA, 1990

[6] Kreid, Johannes. Programming Electronic Music in Pd <http://www.pd-tutorial.com> Online book.