Journal of Universal Computer Science, vol. 13, no. 7 (2007), 959-969
submitted: 7/3/07, accepted: 25/7/07, appeared: 28/7/07 © J.UCS

Pipeline-scheduling Simulator for Educational Purpose

José M. Chaves-González
(University of Extremadura, Spain
jm@unex.es)

Miguel A. Vega-Rodríguez
(University of Extremadura, Spain
mavega@unex.es)

Juan A. Gómez-Pulido
(University of Extremadura, Spain
jangomez@unex.es)

Juan M. Sánchez-Pérez
(University of Extremadura, Spain
sanperez@unex.es)

Abstract: This paper presents a project that provides both, to professors and to students, a tool that is useful for studying, teaching and learning how pipelines work and how they can be scheduled in an easy and widespread way. The project is called PipeSim, and features static and dynamic pipelines with a very attractive, dynamic and intuitive interface. It is well known that pipeline and pipeline-scheduling are very relevant concepts in computer science studies and it is very important that students can learn these in an easy and reliable way. The simulator makes easy both working in depth about pipeline scheduling and working slowly paying attention in the different stages of the scheduling. However, we designed the simulator knowing that principal users would be students with no experience, so both the execution and the presentation of the results have been carefully developed. In addition to this, to check the success of PipeSim, a survey has been made among some students that used the simulator. Results reveal that this kind of applications has a great acceptance among students, thought they consider that simulators are complements to the lessons given by the professor and never a substitute for them.

Keywords: Pipeline, pipeline scheduling, reservation table, collision vector, state diagram, forbidden latency list, delay insertion, educational simulator
Categories: B.2.1, B.2.2, B.4.4, C.1.1, C.1.2, H.5.2, K.3.1

1 Introduction

Pipelining, pipelines and pipeline-scheduling are very important concepts in computer architecture. All current processors use pipelining to improve their performance; in fact, the usage of the computer resources has important influences on their final performance. So, speaking in general terms, pipeline and pipeline-scheduling topics are very outstanding in Computer Science studies and it is quite important that students in these qualifications learn those concepts in an easy and reliable way. In fact, if we have to give a motivation which encourages us to build the simulator

PipeSim was the observation of the subject "Computer Architecture and Engineering", which is given in the 4th year of Computer Science studies. This subject, as it occurs in many others, has a very large syllabus, so there is not time enough to look at some parts which contain special aspects. Among these parts are the topics about pipelining and pipeline-scheduling, which demand to solve many practical problems to acquire a complete knowledge of the stuff. However, pipeline-scheduling problems are neither small nor easy to solve in most of cases.

Pipeline-scheduling problems are usually handled on the blackboard by hand, and this obviously causes that the amount and the versatility of the exercises that both the professor and students can solve are limited. This might be the reason that explains why students are not confronted with a great set of different examples and different kinds of problems which carry them to understand all the possible situations that could happen in the scheduling of a pipeline.

Therefore, the usage of a simulator that provides an easy and fast way to solve problems connected with a user-friendly interface would be a great advantage for students. In fact, it is proved that if students have and use tools that help them to learn and study in depth any practical part of any subject, they learn easily and better, because the teacher not only has one more tool for teaching the most difficult parts of the syllabus in the lessons, but also students can practice on their own and in their own speed at home. Actually, simulators are the best way to understand some of the most difficult concepts related to computers design because these include many elements working all together, and sometimes all those concepts are quite difficult to understand or to manage without quite a lot practice and study.

If we focus on the simulator PipeSim, we can say about it that this simulator is a full interactive application that helps the user to acquire, improve and master the knowledge about pipeline scheduling with the usage of a very simple interface and with a very descriptive and didactic way of working. Besides, the simulator includes a very complete help section, which students can even use for studying the theoretical concepts of pipelining and pipeline-scheduling. Moreover, the application is completely flexible when presents results or the final information to the user. It is possible, for example, to see any of the stages of scheduling, whichever it is. You can walk through the complete simulation step by step, taking in consideration all possible details and watching concrete steps of the scheduling or, on the other hand, you can study only the final results of the simulation without presenting intermediate results.

Finally, we want to point that PipeSim is a novel didactical tool, because to our best knowledge, there is not any other resource in the specific field where PipeSim works; in fact, this was one of the main reasons which encourages us to develop it.

2 Working with PipeSim

Pipeline-scheduling simulator (PipeSim) has been built with didactical purpose. The principal target of the simulator is to give students a new tool to check and solve so many exercises as needed about pipeline scheduling in order to learn and understand that important part of the subject Computer Architecture. A good learning methodology for students could be the comparison between the scheduling problem solved manually by the student and the scheduling result given by the simulator.

Due to the usage and purpose which the simulator was thought for, we had to take care of both the simulator execution and the results presentation so that the information given by the simulator to the user is the easiest to understand and most unequivocal possible. With this aim, PipeSim has been carefully developed following the classical software engineering method [see Pressman 2002], taking into account since the very beginning of the development until the last stages the final didactical purpose of the application and the final users who were going to use the simulator.

PipeSim was developed using Borland C++ Builder 6 [Charte 2000], which is a powerful development tool which provides lots of facilities to build nice window-based applications. Referring to hardware and software requirements, PipeSim has been designed for working in a wide range of systems (current or older systems anyway). We took this as a very important point in the development of PipeSim, because students, and the different possible users that have the simulator, could have any PC at their homes and we wanted that everybody interested in the simulator could use it with no problems. For the final version, PipeSim has been optimized to perform under Windows XP, but it has been checked in a wide range of machines (the older PC successfully checked was a 300-MHz Pentium II, with 96 Mbytes of RAM and Windows 98 installed as operating system). Moreover, the PipeSim installation to the HD takes less than 5 Mbytes. So, we make things easy so that most of students (or any other people) can use PipeSim with no problem. Therefore, the simulator was developed thinking about the users who were going to work with it, and this is the reason that encouraged us to build a user-friendly interface for the application.

PipeSim is an application based on windows, very easy to use and with a well cared final appearance (buttons with descriptive icons, a lot of very clear explanations about the steps to do, complete help function, etc.) that help users to learn and study pipeline scheduling. Besides, the simulator controls all possibilities of errors that could occur during the performance, and it presents the user as many warning and error messages so that the user is informed about any irregular behaviour of the simulator.

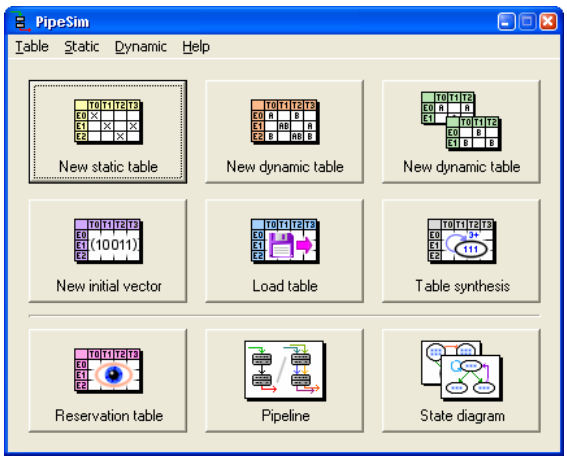

Figure 1: Simulator PipeSim main window

The simulator has a very simple main window [see Fig. 1]. The upper part of the window is occupied by the main menu of the application, and in the rest of the window there are nine buttons, grouped into three rows, according to their category. In the upper row, the system options to insert a new reservation table (RT) -static or dynamic- are situated. In the central row there are options to load a RT in the simulator. This operation can be made using a file or by means of synthesis operations. Finally, in the lower row, there are placed the most useful operations of the application (options for showing the RT loaded in the system; for seeing the hardware diagram of the pipeline obtained from the loaded RT; and for studying the state diagram obtained in the pipeline scheduling, which is the most important part of the scheduling, because it represents the occupation of the pipeline for each clock cycle). Therefore, the buttons placed in the main window have not the ability to control the entire functions that can be performed by the simulator. To access all the functions in the simulator you have to use the main menu, which has four items, which are explained in the following lines:

Menu Table: The load and save operations for static and dynamic tables are managed from this menu, besides there are options to show the loaded table and to quit the simulator.

Menu Static: Through this menu, operations which are made over static reservation tables are performed. So, this menu controls static pipeline scheduling. These operations are:

- Draw the static pipeline (hardware elements) which represents the reservation table (RT) loaded in the simulator.
- Obtain the forbidden list from this RT.
- Calculate the initial vector using the forbidden list.
- Generate and draw the state diagram obtained from the initial vector.
- Perform a complete study about the simple cycles (and their properties) presented in the state diagram.
- Insert delays in a RT that have not an optimal throughput and to synthesize a new RT taking an arbitrary cycle.

If we do not consider the last operation, the others are the sequence of operations which are necessary to solve a complete problem about static pipeline scheduling. Nevertheless, it is not necessary to go through each option, because the simulator has the capacity of going to a concrete option, running automatically the previous stages in the scheduling.

Menu Dynamic: The user can manage functions that can be made over a dynamic reservation table using this menu. The options to do the entire dynamic pipeline scheduling are here, step by step or jumping to a concrete stage of the scheduling. The dynamic scheduling is more complicated than the static one, because the pipeline performs several operations at once. PipeSim works with dynamic pipelines of 2 operations at once.

Menu Help: Help options, language configuration and “About…” option are accessible from this menu. PipeSim help is divided into a complete user manual where is explained the simulator usage and into a theoretical explanation of the concepts which support the simulator behaviour. Besides, it is possible to change the language of the simulator in any moment of the simulation from English to Spanish and vice

versa. The complete simulator works in both languages, including help files, installation system, warning and error messages, etc.

3 PipeSim main features

As we said in [Section 1 Introduction], the main aim of PipeSim is to help students to understand and to study the concepts and the method of pipeline scheduling. The simulator gives support both, to the theoretical knowledge (with a complete help section) and, over all, to the practical ability (it is possible to propose and to solve a very big amount of problems) in the part of Computer Architecture referred to pipeline scheduling. But PipeSim can be also considered an application for the analysis and the detailed study of big or very complex pipeline-scheduling problems. We can summarize the principal features of PipeSim in the following list:

- PipeSim allows the study of both static [see Fig. 2] and dynamic [see Fig. 3] pipelines, so this makes the simulator more complex and useful.

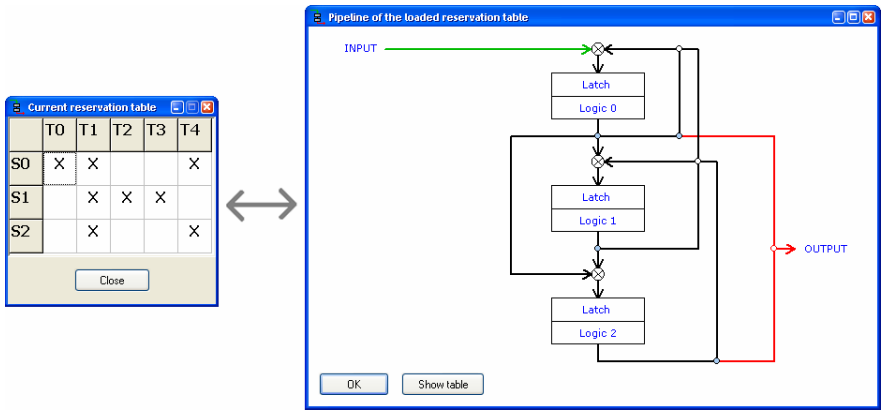

Figure 2: Static RT and its associated Pipeline generated by PipeSim

- The user can insert data in the simulator taking different ways. Users can fill in the static or dynamic reservation tables by hand, they can load files with the information of the tables, they can insert a collision vector to start the simulation or even they can synthesize a new table from a cycle of the state diagram. The state diagram represents the pipeline occupation, so the user can choose an optimal cycle to create the new table. Therefore, the simulator gives the user the possibility to start the simulation from different input data, and this provides a widespread usage of the simulator to anybody who uses it, and if we focus specifically on students, this means that they can do a bigger amount and more sorts of exercises, which is very good for their learning.
- It is possible to save in specific files reservation tables. This information can be loaded later automatically in the simulator. Moreover, PipeSim is able to save some results, for example, BMP images of the state diagrams or pipeline draws.

(These pictures can be used later for different purposes: printing, insertion in documents, professor exams…).

- The simulator offers a complete multi-language configuration, in English and in Spanish, offering the possibility to change the language in any moment of its execution, from the installation to any other time.
- The application can calculate the forbidden list of a static pipeline or the forbidden lists of a dynamic pipeline.

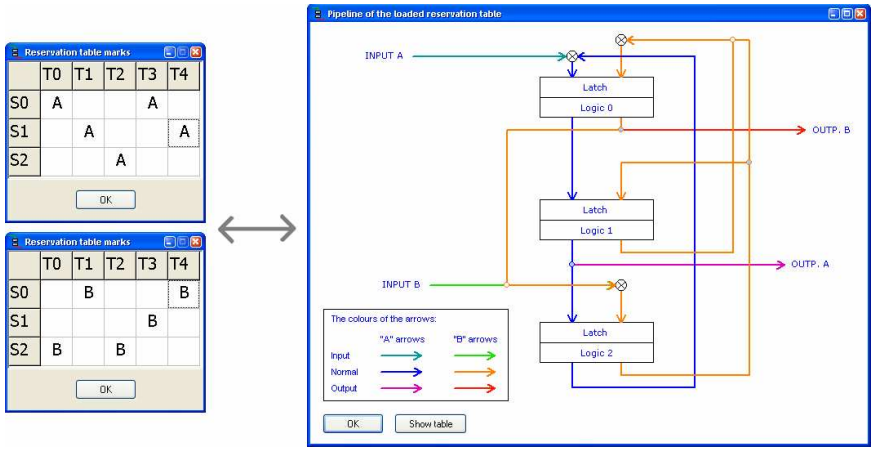

Figure 3: Dynamic RTs and the Pipeline associated to them generated by PipeSim

- It can calculate the collision vector (or –dynamic collision matrices) generated from the forbidden list (or lists –in the dynamic case).
- PipeSim can represent the hardware elements that symbolize the pipeline loaded in the simulator. The picture presents the pipeline stages and the data path followed in each clock cycle. The simulator can draw both static [see Fig. 2] and dynamic [see Fig. 3] pipelines of any reservation table up to 100 stages. It can represent different high-level hardware components such as: logic of the stages, latches, multiplexers and demultiplexers.
- The simulator obtains and represents the static or dynamic state diagram, from the collision vector or collision matrices of a loaded reservation table. This diagram represents the different states of a pipeline for a requested period of time. The diagrams are built using different colours and a clear state distribution [Fig. 4], [Fig. 5], to make it easy for reading them. The algorithm we developed for the drawing of any state diagram created from arbitrary data inserted by the user is the most complex part of PipeSim because it can draw diagrams containing up to 128 states.
- Therefore, the state diagram can become really complex. It is then very useful to see the diagram as a table (the state diagram is a graph that can be represented by its matrix). This possibility is also offered by PipeSim.

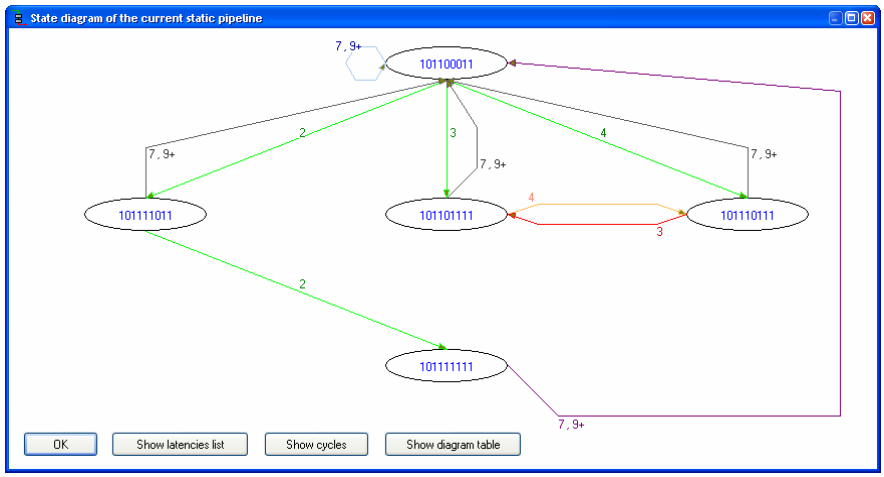

Figure 4: State diagram of a static pipeline

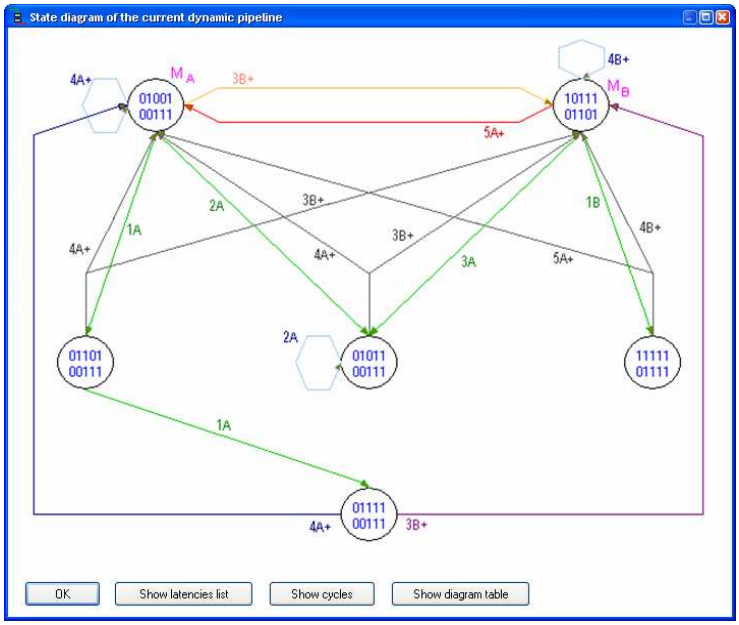

Figure 5: State diagram of a dynamic pipeline

- The application includes a complete help system, where not only the simulator usage is explained, but also the theoretical concepts of pipelining and scheduling.

- PipeSim performs a study of the pipeline throughput improvement by means of delay insertion (and it is even possible to keep the data order in the pipeline stages). [Fig. 6] shows some screens of the process and the final result for a small example. We have achieved that PipeSim increases the pipeline throughput in an efficient way. The simulator inserts the minimum amount of delays to get the optimal performance.

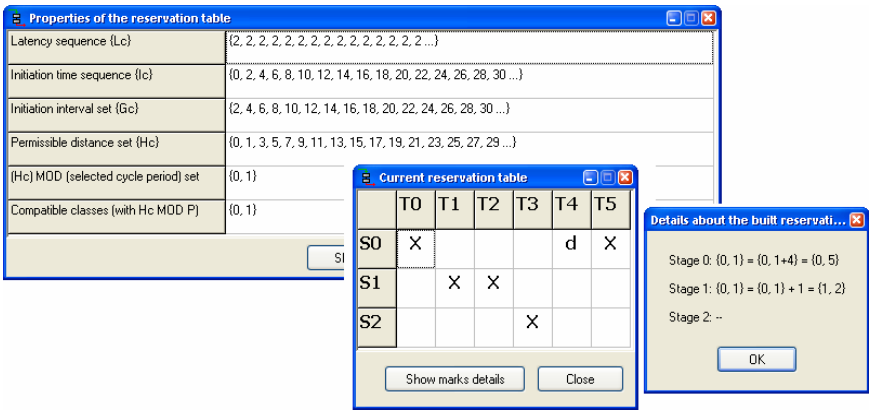

Figure 6: Delay insertion example in a static reservation table

- The simulator can synthesize new reservation tables from different input data, showing the details of the new table creation.
- The simulator performs an intensive study of all simple cycles in the state diagram (for static and dynamic pipelines). Using this tool of PipeSim, it is possible to know when a pipeline is optimal for the operations loaded in it. Besides, the simulator allows changing the data distribution in the pipeline so that the throughput became optimal if it is necessary. Therefore, PipeSim offers a complete performance study of the pipeline, and gives the user the entire detailed information about it.
- Besides, to make the simulator distribution and installation easier, PipeSim has an automatic installation assistant.
- Furthermore, we have to say that PipeSim can work with little (learning exercises), but also with huge problems (pipelines up to 100 stages, state diagrams up to 128 states). So, the simulator has the flexibility for operating with all kind of users, beginners (who want a lot of explanations and details given) and experts (who want to get a specific result or see a concrete part of the simulation).
- Finally, we want to emphasize that we had a lot of care not only in the development of PipeSim execution, but also in its interface. We took into account that the simulator would have a learning aim, so a clear and nice appearance make easy the student learning, which is the principal purpose of PipeSim.

4 Didactic experience with PipeSim

PipeSim started to be used with a good acceptation among the students of the 4th year in Computer Science for the subject "Computer Engineering and Architecture" in the academic year 2005/2006. Moreover, we have to say that this was not the first simulator used in this subject (e.g.: [Vega et al. 2001] and [Vega et al. 2004]). In fact, the success obtained with other simulators encouraged us to develop and to create PipeSim, which has been used by professors and students since its implantation following the next methodology.

The first stage is the simulator presentation, where the professor shows PipeSim to the class, explaining how the simulator can be obtained (downloading it from the web site [Vega, Chaves 2007]), how does it work, what is it used for, and solving some practical problems with it. Using PipeSim, the professor can solve more problems and the resolution takes him less time than if he does by hand. So, the concepts and methodology of pipeline scheduling is clearer and deeper in the lessons than without the simulator, because it is studied more in less time. After the teacher explanation, in what we can consider the second stage, the student can download and use PipeSim at home. S/he can do all the exercises s/he needs, and s/he can study using the simulator and its complete help system. Finally, PipeSim has a set of typical reservation tables saved in files, but the professor, in a third stage, suggests some interesting exercises that students can solve by hand and check using the simulator.

To check the acceptance of PipeSim after the first year of usage, a survey was made among some students that used the simulator. We asked about how good are the different parts of the simulator for them, if it is useful for their learning, if the management of the simulator is easy, if the theoretical concepts are well explained, if it is well-organized, if the interface is clear, which ones are the final conclusions that they have…

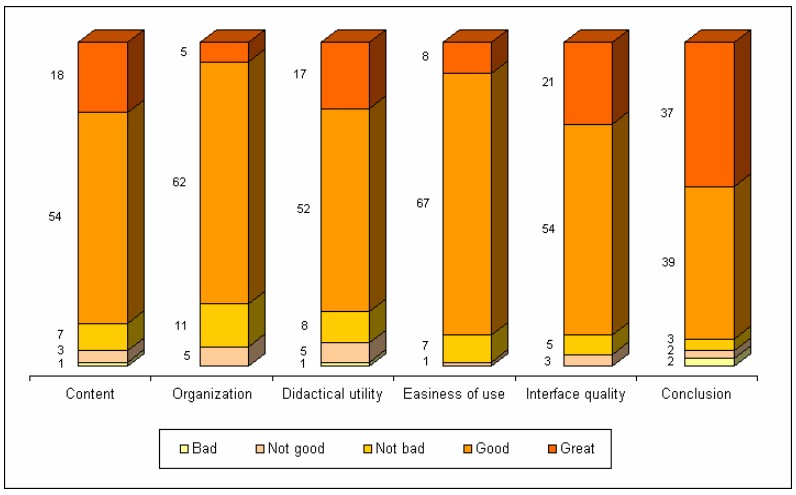

Figure 7: Students survey results

The survey was done among 83 students and the results obtained were very satisfactory for us. In [Fig. 7] some of the questions and the number of answers obtained are shown.

In the graph of [Fig. 7] a sample for the results obtained in 6 of the questions done through the survey to the students is shown. All questions about the simulator have 5 level of satisfaction, from bad (the lowest) to great (the highest). As it can be observed, the success of PipeSim among the students has been quite high, however when they have to answer the question about if they prefer to learn pipeline scheduling only using the simulator, only with the professor lessons or with both techniques, most of them prefer traditional lessons with the support of PipeSim [see Fig. 8].

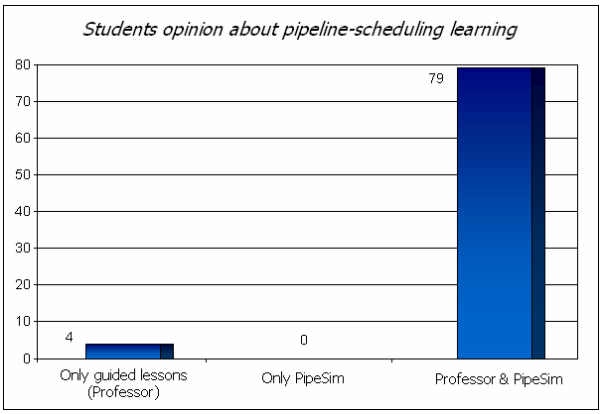

Figure 8: Students preferences in learning methodology with PipeSim

Therefore, according to the survey, PipeSim is a very successful didactical tool that gives support to traditional lessons, however for most of students it is very important that the professor gives a complete explanation and introduce them in the stuff they are studying. We can conclude that students prefer a direct human contact when they start to learn something, because this gives them an introduction and the knowledge basis that they can later improve using a simulator.

Finally, other questions of the survey state that students used PipeSim in average 11.27 times and that all students (100%) think that this kind of applications (which give didactical support) would be strongly recommended in other subjects.

5 Conclusions

In this paper we have presented a very complete and versatile simulator about pipeline scheduling. Although PipeSim could be used in other fields (research), it was developed for students, to teach them pipeline scheduling. As students are inexperienced in the beginning, we took care of explaining the methodology step by step, showing all the details in the clearest way possible, trying to give them all available help. So, although PipeSim is used for the professor to solve problems in

class, it is a tool thought mainly for students. Moreover, it is possible to go further in the topic and simulate real problems, because PipeSim has enough power to do it. Therefore, the simulator is useful to learn how the pipeline scheduling is done, learning and understanding it gradually. But it is useful too for studying more complex cases about scheduling in real pipelines and maybe for doing some research.

To check the students' acceptance, we made a survey among the students who used the simulator, and we can conclude that PipeSim was a complete success among the students. At present, PipeSim is a didactic tool which is used every year among students in the subject Computer Architecture.

Finally, we have to say that the simulator is free downloadable for educational and research purpose from [Vega, Chaves 2007].

Acknowledgements

This work has been developed in part thanks to the OPLINK project (TIN2005-08818-C04-03). José M. Chaves-González is supported by research grant PRE06003 from Junta de Extremadura (Spain).

References

[Charte 2000] F. Charte: "Programación con C++ Builder 5"; Anaya Multimedia (2000).

[Pressman 2002)] R.S. Pressman: "Ingeniería del Software: Un Enfoque Práctico"; McGraw-Hill (2002).

[Vega et al. 2001] M.A. Vega, J.M. Sánchez, J.A. Gómez: "An Educational Tool for Testing Caches on Symmetric Multiprocessors"; Microprocessors and Microsystems, 25(4): 187-194 (2001).

[Vega et al. 2004] M.A. Vega, J.C. Burgueño, J.M. Sánchez, J.A. Gómez: "Web Simulator for the Teaching of Instruction Dynamic Scheduling"; 6th International Symposium on Computers in Education, Cáceres, Spain, pp. 1-8 (2004).

[Vega, Chaves 2007] M.A. Vega, J.M. Chaves: "PipeSim" (2007). http://arco.unex.es/pipesim/

[Zargham 1996] M.R. Zargham: "Computer Architecture: Single and Parallel Systems"; Prentice-Hall (1996).