4th International Conference on Launcher Technology "Space Launcher Liquid Propulsion"
3-6 December 2002 – Liege (Belgium)

CARINS : A FUTURE VERSATILE AND FLEXIBLE TOOL FOR ENGINE TRANSIENT PREDICTION

Gerard Ordonneau*, Gerard Albano, John Masse*****

*ONERA, ,
Fundamental and Applied Energetics Department
BP 72
29, av de la Division Leclerc
92322 CHATILLON Cedex FRANCE
Tel : +33 1 4673 4333
Fax : +33 1 4673 4147
E-mail: Gerard.Ordonneau@onera.fr ;
**CNES,
Rond-Point de l'Espace,
F-91023 EVRY Cedex
Tel : +33 1 6087 7148
Fax
E-mail: Gerard.Albano@cnes.fr
*** Appedge
18-22 rue d'Arras
92000 NANTERRE
Tel : +33 1 4782 9505
Fax : +33 1 4782 9505
Email : john.masse@appedge.com

CENTRE NATIONAL D'ETUDES SPATIALES

CARINS : A FUTURE VERSATILE AND FLEXIBLE TOOL FOR ENGINE TRANSIENT PREDICTION

Gerard ORDONNEAU*, Gerard ALBANO, John MASSE*****
ONERA, BP 72 - 29, av de la Division Leclerc , F-92322 CHATILLON Cedex
** CNES, Rond-Point de l'Espace, F-91023 EVRY Cedex
*** Appedge, 18-22 rue d'Arras, F-92000 NANTERRE
Gerard.Ordonneau@onera.fr ; Gerard.Albano@cnes.fr, john.masse@appedge.com

Abstract

This paper illustrates the capabilities of the “symbolic manipulation” method which combines advantages and eliminate drawbacks of “classical” simulation methodologies. The aim of the CARINS project is to apply this technique to liquid propellant rocket engine systems, composed of pneumatic, hydraulic, mechanical and combustion subsystems, which are also used in a lot of industrial fields where can be found many critical systems for safety like embedded engine. This article presents the first step of CARINS software development, including the general structure and the computing methodology. This should provide a useful tool for modelling, simulating, and analysing dynamical systems.
Furthermore, CARINS is an Open System for simulations that allows the user to implement its own equations and connections. With this respect, CARINS combines the advantages of symbolic manipulation offered by Computer Algebra System and the power of numerical computing.

Keywords : Liquid rocket engine, numerical simulation, computer algebra, CARINS, transient, combustion, hydraulic, pneumatic.

Introduction

CARINS is a versatile and flexible tool for engine transient prediction in liquid rocket propulsion. This project was initiated two years ago by CNES, in partnership with ONERA, and it involves several laboratories.

Motivation and Objectives

Although numerical simulation is more and more used to predict operation of practical systems, it remains one of the main challenges of next years. In many fields, simulation will reduce drastically design development cost. For instance, simulation will enable to build digital mock-up of systems in order to :

- understand static or dynamic behaviour ;
- conduct fault tolerance tests for critical application when only “virtual” design is available, and to prepare test campaigns for real hardware ;
- conduct parametric studies in order to choose optimal design.

The main motivation of the CARINS project is to give to engineers and research teams a new powerful tool for reproducing time evolution of physical parameters which characterise the propulsion system behaviour for space launchers, or only part of it, during all the mission phases (start-up and shutdown transients, chilldown phase, operating point modification, main stage, etc.).

Several attempts have been made to use commercial softwares for liquid propellant rocket engine (LPRE) studies. But in many cases, the goals were not totally achieved because, either physical models were not implemented, or it was too difficult to implement them in the requested time. So, CNES had already participated in such software developments in the 90's[1,2]. For the new tool, in addition to usual specifications, CNES included two extra items which make the uniqueness of CARINS : first, the tool must be an open software where the users will be able to operate easily at the lowest level of programming for new models development, and second, the software must be free of licensed-tools, in order to control as much as possible the CARINS development and future upgrades, and to easily distribute the software to its partners, as research laboratories, if necessary.

Moreover, the software structure must take into account complex physical phenomena involved in LPRE transients.

Physical phenomena and propulsion models

Liquid propellant rocket engines systems to be considered varies from simple pressurised engines of low thrust for satellite applications to complicated stage combustion cycle engines for heavy launcher applications. Every LPRE system can be viewed as the assembly of a lot of components such as tanks, pipes, valves, orifices, vessels, pumps and turbines, injectors, combustion chamber for gas generator or main thrust chamber, nozzles, etc[3-6].

In these components, very different thermodynamic conditions and physical phenomena are observed during engine runs. Propellant can be liquid at low temperature in cryogenic tanks or hot gas in combustion chambers. Two-phase flows can be encountered in regenerative circuit where the liquid and its vapour are present ; inert gases can be melted with liquid propellants during purge sequence. Besides the fluids, mechanical pieces are subject to forces and movements like pump rotors and regulator pistons for instance. Nevertheless, the propellant flows and the mechanical movements obey to general conservation laws. So, physical models will have to reproduce the correspondent compliant, inertance, resistive and propagation effects.

Nevertheless, characteristic times are of primary importance in transient simulation and modelling can take advantage of this point[7]. For instance, if one is interested in water hammer effect due to rapid valve opening, where the inertance and the compliance of the fluid are combined to lead to wave propagation, one dimensional model is required with partial differential equations[8,9]. In other application with slower transient, the liquid can be modelled as incompressible fluid. In this case, the lumped parameter method, solving ordinary differential equations, is accurate enough[7]. Moreover, for some applications like chemical kinetics for instance, the characteristic times may vary. Thus, simulation tools should take into account all these various situations, and also those not yet imagined.

However, simulation requires two steps from the engineer's point of view. The first one is the modelling, including the choice of the relevant model for every component of the process and the assembly of these components. The second one is the resolution of the problem.

General concept

Background

Up to now, industrial companies use two kinds of simulation tools with graphical user interface (GUI) to conduct studies and design systems in the main fields of engineering : mechanics, thermo-hydraulics, automatic control, electrical engineering or electronics.

- First kind of tools is "black-box" tools. This software include both the modelling and the simulation steps. Description of mathematical behaviour of elementary components with interconnection between them, and resolution of the resulting set of equations are tightly linked. Main advantages of such an approach are the availability of well-known general tools and their use by non-expert people very quickly. Simulation becomes popular thanks to such tools. But now, as these tools have been designed ten or twenty years ago, some drawbacks appear like the lack of performance of simulator (in terms of CPU-time) and the weak opening of such tools.

- Second kind of tools consists in writing a dedicated simulator independent of GUI. They are very efficient and they can be used in real-time environment or for parametric analysis to find optimal solutions. But generally the parsers are poor for manipulating user algebraic expressions or not easy to use to introduce a complex new model.

Confronted with difficult and unforeseen situations[10], we propose to setting up a third way with Automatic Model Generation in using Symbolic Manipulation which combines advantages of both previous methods. Several examples of software allow to consider use of Computer Algebra as a successful approach for simulation pre-processing[11-14].

The symbolic manipulation allows to split simulation in two independent steps :

- Modelling, where the engineer build his process by choosing the components, connecting them, and generating the dedicated simulator using a computer algebra system (CAS) ;
- solving the problem by running the simulator.

MAXIMA is the computer algebra system (CAS) used in CARINS.

Such kind of software, combining GUI and CAS, allows to manipulate not only numbers but formulas, equations, mathematical expressions and user expressions. So the modelling task can be viewed as the CAS customisation to a specific engineering domain using the GUI. Coding of formulas provided by the user is also done by such a software, as they can learn how to translate mathematical expressions into numerical languages.

CARINS: Symbolic technique for intelligent modelling and simulation

The goal of CARINS is to propose such a technology, that is an intelligent modelling and simulation tool using symbolic manipulation, to CNES and its partners in order to better fulfil end-users needs. CARINS is graphical and interactive and encourages the user to try (fig1).

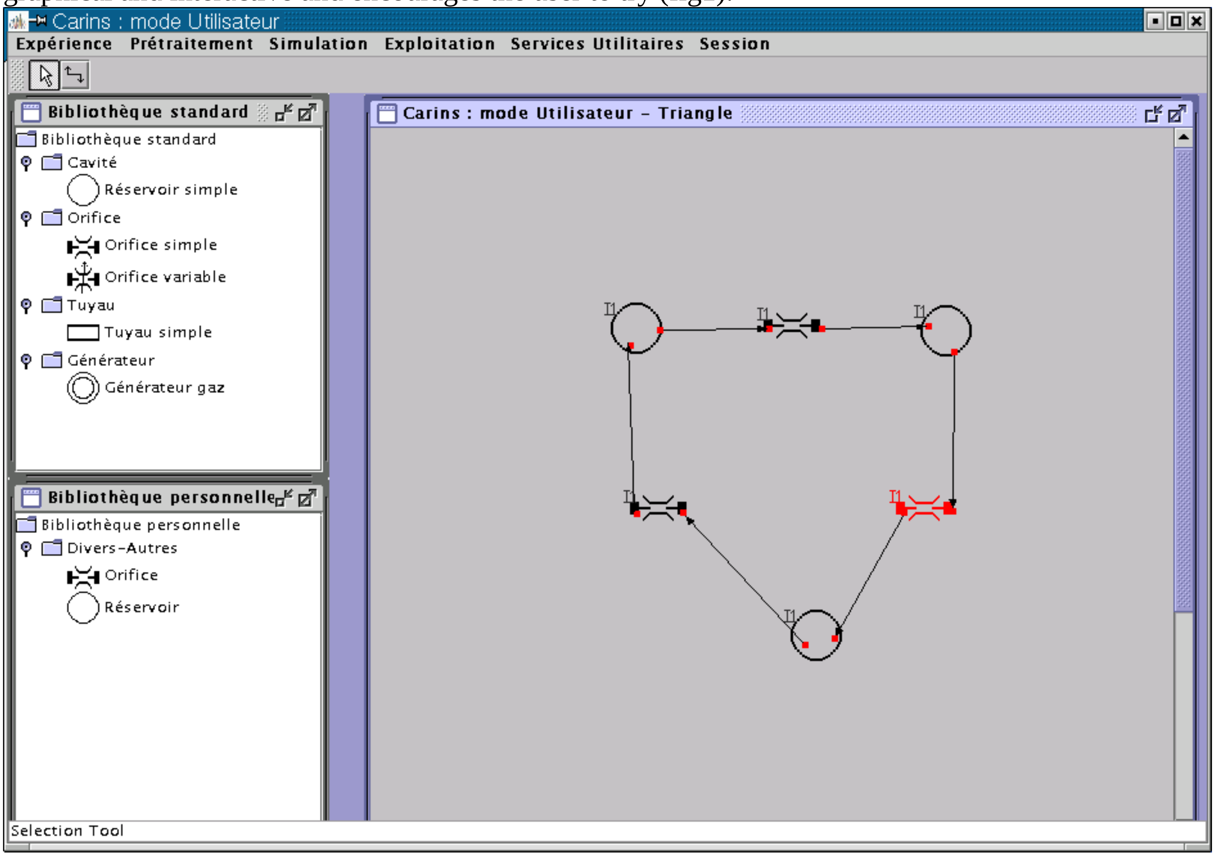

Figure 1 : CARINS GUI

The development basis will of course be the SIMPA[11] software but we propose to explore some other directions to increase the potentiality of the tool :

- Build up of an intelligent GUI by integration of a CAS (Computer Algebra System) into an existing GUI. This will allow to check user mathematical input (equations, validity domain for variables, dimensions) as they are entered through the GUI.
- Build up of an intelligent code generator. This code generator gives the possibility to generate codes in different languages like FORTRAN, C, CACSD (like Scilab).
- Set-up a pre-processing for the integration strategy which will be able to deal with phenomena like discontinuities, hysteresis and so on.
- Connect the components in several ways like block diagrams and also with bi-directional links, because the flow direction can change and the information source changes also. Bloc diagram description does not allow it.
- Easily introduce a new component with a new equation set and its own connections.
- Possibility to simulate "hybrid systems" and to perform co-simulation, co-design and co-verification when the modelling are heterogeneous. These technical capabilities (i.e. to be a host for applications) are very useful when the user has some complicated system or wants to verify and validate in parallel several axes of modelling.
- Advanced “Report Generation” functionality which mixes drawing of system topology and mathematical typesetting of components modelling. Such task is usually very time-consuming.

Architecture of CARINS

Using SIMPA experience, CARINS is structured around three main entities, whose development will allow to meet the aforementioned goals :

- an easy-to-use GUI, realised using a object-oriented method ;
- a model library : basic element models are organised under mathematical structure (state variables, internal variables, parameters, equations) which will ensure their generic status and gives to the software the best evolution capabilities. This structure is obtained by building models from general law from energetics and mechanics. Parameters are used to describe physical phenomena of smaller scale ;
- an Automatic Model Generator (AMG) using a computer algebraic system under GNU license (MAXIMA).

The engineer describes his system through CARINS's GUI (see figures 2 and 3) in which he can enter numerical data or analytical formulas built upon variables or expressions that are part of the system. These inputs must be specified in the MAXIMA syntax form.

After the system modelling is completed and checked, the MAXIMA Computer Algebra System performs the scheme analysis (interconnection) to build a optimised set of equations and then automatically generates a numerical Fortran simulator of this system.

> *Remark: We chose the Fortran language because it is more comprehensive than C language for numerical computationsbut in the future we will generate C or use "f2c" that allows to build real-time simulators more easily.*
> *In the present time our objective is to design or validate systems which ask for intensive simulation with in nominal or extreme conditions for parameters. The fortran language is efficiency for that.*

MAXIMA is the application engine which simplifies, optimises mathematical formulas and solves the interconnection between individual components. Interconnection resolution is done through evaluation and substitution for the connection equations in the global system. When MAXIMA finds an algebraic loop, it shows the equations and variables in the loop.

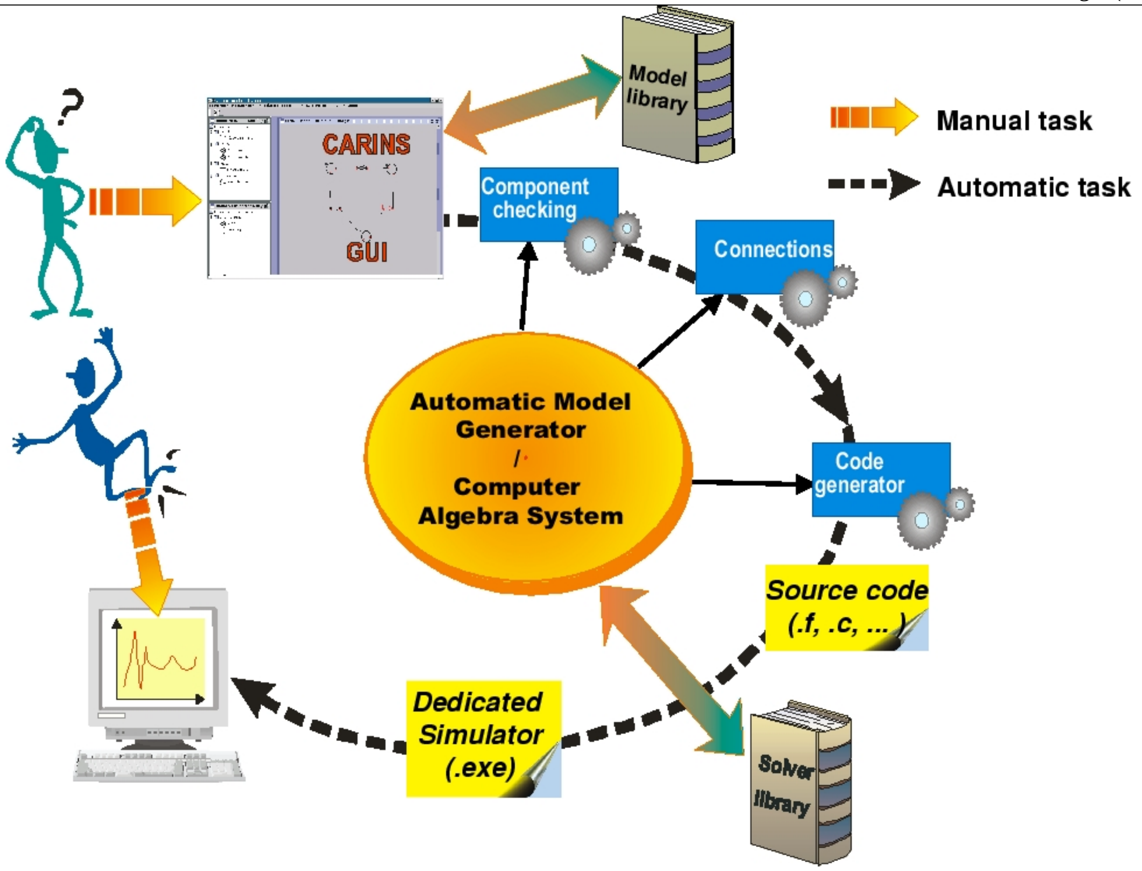

Figure 2 : CARINS Structure

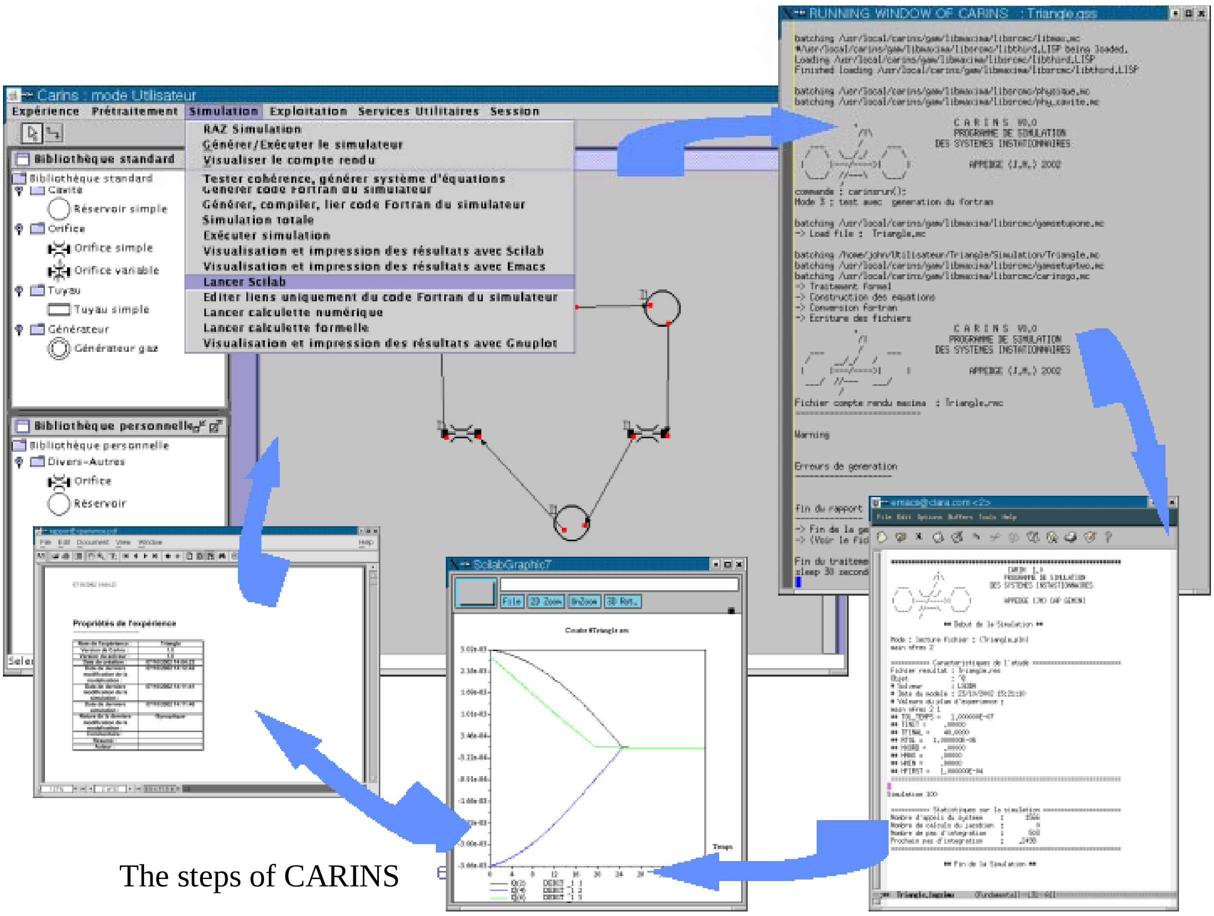

Figure 3 : simulation loop

From each component model and from user equations and connecting rules, MAXIMA built the equations of the global system using symbolic parameter management and additional analyses to realise a consistency checking on the global system.

The last step is to automatically generate the optimised code of the simulator that is then linked with a solver of ordinary differential equations (ODE).

The generated simulator

The main characteristic of the generated simulators is its efficiency because it is specific to the studied system. Dedicated simulators perform a more accurate and more efficient simulation than “black box simulators".

For each modelled process, MAXIMA generates a global virtual code (before the effective code generation), which is translated into Fortran with a MAXIMA Fortran generator The virtual code generation brings flexibility into the building of the simulator : then one can solve the system with the solver of his choice.

CARINS offers LSODA[a] and LSODES[b] solvers and linpack library to solve the ODE system. Other solvers will be added to CARINS in the future.

Integration strategy: Fast simulation

The mathematical models often contain discontinuities. Typical examples for the occurrence of discontinuities are :

- clock-actuated controllers: the right hand side of the differential equation depends on the system state at explicitly given instants ;
- friction : dry friction leads to jump discontinuities in the right hand side ;
- hysteresis : the right hand side depends on the history of motion ;
- events of co-simulation (synchronisation).

CARINS is able to handle all these kinds of discontinuities by zero crossing detection. Few software packages are able to correctly compute these types of discontinuities or to implement effective integration methods for such situations. The common turnaround is to smooth discontinuities. This smoothing approach leads to artificially stiff systems and introduces higher derivatives.

MAXIMA is a helpful parser to really identify the discontinuities functions (sign, timer, logical statement, ..) and to improve the differential equation system organisation for accurate root finding problem, that is switching points localisation.

To sum up briefly, the heuristics of integration strategy is based on the definition of Boolean electronic components: D-type with flip-flops with clock enable and D-type transparent latches. The switching functions keep the value they have at the last validated step during the computation of the next step by the solver. Thus, the ODE system has a continuous representation and the numerical resolution is very fast. If no discontinuity flag is set, the new step is validated, the switching functions are updated and computation goes on. When a discontinuity flag is set, the switching point location procedure starts until it is found with a

[a] LSODA : livermore solver for ordinary differential equations, with automatic method switching for stiff and nonstiff problems. LSODA solves the initial value problem for stiff or nonstiff systems of first order ode-s

[b] LSODES : Livermore solver for ordinary differential equations with general sparse jacobian matrix.

given accuracy. After the event is processed, the new step is validated and the integration is restarted with the new ODE system.

The main advantage of this method is that it may be applied to any integration methods for ODE systems.

Conclusion

The objective of the CARINS project is to develop a new generation software able to simulate LPRE transients. It is intended to be used for a large variety of engine cycle and propellants. In this paper, we presented the methodology and the structure of this tool. CARINS is designed to be a powerful tool to fulfil industrial needs. It will increase flexibility in mathematical information management and end-user benefits.
The use of this kind of software will probably increase during the coming years as it allows research teams to test easily new physical models and it facilitates transfer to industrial users.

References

1. J. Masse, J.S. Darrozès;, *Modélisation et simulation des systèmes pneumatiques dynamiques*. Conference on Propulsive Flow in Space Transportation System, CNES, Bordeaux (September 1995)
2. J. Masse, *Maple and Numerical methods applied to Transient Analysis of Pneumatic Systems*, ???
3. Liu Kun, Zhang Yulin, *A Study on Versatile Simulation of Liquid Propellant Rocket Engine System Transients*, AIAA 2000-3771
4. K. Van Hooser, J. Bailey, A. Majumdar, *Numerical Prediction of Transient Axial Thrust and Internal Flows in a Rocket Engine Turbopump*, AIAA 99-2189
5. J.R. Mason, R.D. Soutwick, *Large Liquid Rocket Engine Transient Performance Simulation System*, N91-24340 (1989)
6. M.P. Binder, *An RL10A-3-3A Rocket Engine Model Using the Rocket Engine Transient Simulator (ROCETS) Software*, AIAA 93-2164
7. A. Kanmuri, T. Kanda, Y. Wakamatsu, Y. Torri, E. Kagawa, K. Hasegawa, *Transient Analysis of Lox/LH2 Rocket Engien (LE-7)*, AIAA 89-2736
8. P. Corvisier, C. Jacquot, M. Feidt, G. Albano, *Numerical simulation of unsteady non-ideal gas flows. Application to Van der Waals gas*, France 4th International Conference on Launcher Technology "Space Launcher Liquid Propulsion", 3-6 December 2002 – Liege (Belgium)
9. J. Keppeler, E. Boronine, F. Fassl, *Startup Simulation of Upper State Propulsion System of Ariane 5*, 4th International Conference on Launcher Technology "Space Launcher Liquid Propulsion", 3-6 December 2002 – Liege (Belgium)
10. B. Legrand, G. Albano, P. Vuillermoz, *Start-up transient modelling of pressurised tank engine : AESTUS application*, 4th International Conference on Launcher Technology "Space Launcher Liquid Propulsion", 3-6 December 2002 – Liege (Belgium)
11. J.Masse, *Le Calcul Formel et le Calcul Numérique pour la simulation des Systèmes Dynamiques*, congrès 2AO96 (ESIEE), Novembre 1996.
12. J.Masse, *Simulation en Mécanique: Apport du Calcul Formel*, Journée Franco-Allemandes de la S.F.M : Mécanique et intelligence artificielle, Novembre 1996.
13. E. Eich ,C.Fuher, *Numerical Methods in Multi-body Dynamics*. FRANCOSIM , ROANNE mai 1992
14. G. Levey, J.Masse, *Joint Use of Maple and Basile :Using Algebra in CACSD*, Présentation à la Première conférence Européenne sur le Calcul Formel en théorie des systèmes, 13-15 Mars 91 Paris, Springer et Verlag N 165.