Contributions of the cosimulation in the chain of design of an electropneumatic system

Xavier Brun, D Morand, S. Scavarda, Daniel Thomasset

CONTRIBUTIONS OF THE COSIMULATION IN THE CHAIN OF DESIGN OF AN ELECTROPNEUMATIC SYSTEM.

X. BRUN, D. MORAND, S. SCAVARDA, D. THOMASSET
Laboratoire d'Automatique Industrielle
http://www.insa-lyon.fr/Laboratoires/lai.gb.html
INSA Lyon, Bâtiment Saint Exupéry
25 Avenue Jean Capelle, 69621 Villeurbanne Cedex, France
Tel: (33) 4 72 43 88 81 Fax: (33) 4 72 43 85 35
email: xavier.brun@lai.insa-lyon.fr

ABSTRACT

This paper focuses on a "shared simulation" procedure for the modelling and analysis of an electropneumatic system. A "shared simulation" or "cosimulation" can be defined as the common simulation between specialized softwares, where parts of the model have been created. During procedure, information exchange is realized between simulation environments thanks to a communication interface. Such principle is explained and commented for a particular case: positioning control laws study of an electropneumatic cylinder, with "cosimulation" done between a fluid power simulation software (AMESim [1]) and the Matlab/Simulink software. Experimental and simulation results are compared, with possible parallel applications given. Advantages and drawbacks of "cosimulation" are discussed.

Keyword: cosimulation, experimental and simulated results, electropneumatic.

Notation

p	pressure in the cylinder chamber [Pa]
x_d	desired position [m]
x, v, a	position, velocity, acceleration[m],[m/s], [m/s^2]
$q_m(u,p)$	mass flow [kg/s]
F_{ext}	external force due to atmospheric pressure [N]
k	time index for numerical sampling [s]
T_e	sampling period [s]
$U_P/_N$	servo-distributors voltages [V]
$U^e_{P/N}$	control law voltage at the equilibrium set [V]
$u_{P/N}$	control law voltage before servo-distributors dynamic stage [V]
K_x, K_v, K_a	position, velocity and acceleration feedback gains [V/m], [V/m/s], [V/m/s^2]

INTRODUCTION

Over the last twenty years, the numerical simulation has become a very useful and attractive solution for development of new product. Nowadays, a lot of simulation software are used for sizing problem, rapid prototyping, specifications improvement or testing different strategies of control. To effectively verify a control method, control development should be separated from software development. In order to achieve that purpose, a hardware-in-the-loop simulation can be used. This method provides interface between PC based data acquisition and the system sensors and actuators. Together with graphics, real-time control simulation software, a hardware-in-the-loop simulation can provide a relatively simple and inexpensive solution for rapid prototyping of feedback control laws.

Another approach is to use the cosimulation principle where two kinds of software are necessary. The first one has to simulate the model of the process and the second one is useful to calculate the control law. One of the aims of the cosimulation is to help the simulation specialists in numerical and modelling fields to work with specialists in automation. Indeed more and more complex softwares are needed to simulate a specified process. But the control algorithm which is implemented is written in C code or with specific softwares, used for program acquisition cards. So the programming language of software simulation and software control are very often different.

Thus, this paper focuses on an interface between the industrial specialized software: AMESim. (Advanced Modelling Environment for performing Simulations of engineering systems) and the Matlab / Simulink software. The application field concerns the control of an electropneumatic actuator. The system modelling is developed with AMESim and the control law is calculated with Simulink. When the control strategy is validated, the implementation is directly possible on every dSpace acquisition card because they can be programmed directly from the Simulink model. Hence an improvement in development time and programming simplicity. Today, the development of network performance combines with this kind of cosimulation could emerge on a stronger collaboration between simulation teams and control teams.

This article presents an elaborate model of an in-line electropneumatic servodrive, controlled by two three-way servo-distributors. State feedbacks with fixed or scheduled gains are studied. For point to point specifications, the cosimulation results are compared with experimental results. So, the advantages and drawbacks of the cosimulation are discussed.

COSIMULATION PRINCIPLE

Nowadays, engineers must compose with complex systems, dealing with several scientific fields at the same time when they design a component or a device. Such a work requires a good control of many knowledges, hence the interconnection between various specialists in order to simulate a complete system. Complexity of any field and accuracy of simulation's results drove software companies to be more specialized in a particular area of the engineering field (ex: ADAMS for multi-body dynamics, Fluent for computational fluid dynamics, AMESim for fluid power and electromechanical devices modelling). But, how to understand the complete behavior of a mechatronic system? Having a global simulation tool may be rather complicate to handle and modify as long as the project evolves. Therefore, connecting simulation software is the answer to complex simulation through different fields. This principle is called "cosimulation", or in other words "shared simulation", as process model and analysis are shared between software. Specific parts of the system are modeled within different simulation environments. During simulation process, each environment exchange information in order to calculate its own dynamics and send out outputs to other ones at regular step time. One can see that such a methodology for simulation is rather close to the hardware-in-the-loop (HILS) simulation one, as it deals with software-software communication instead of software-hardware communication. Indeed, the whole process is virtually presented in a "cosimulation" whereas in a HILS some part are physical devices, connected to the software. Such studies have been often realized as recent one [Choi, 2000]. At this stage "cosimulation" seems to be a good transition between complete simulation and practical tests, as it allows to keep separated parts of the model that can be tested and optimized in a HILS. Such an example will be shown further for an electropneumatic system. Before a short explanation is given about the AMESim-Simulink cosimulation methodology and principle, chosen for this experimentation. This is based on previous cosimulation presentation [3].

AMESim (trademark of Imagine S.A.) is an Advanced Modelling Environment for performing Simulations of engineering systems. Based on several years industrial experience, this software is more dedicated to fluid power systems and mechanical devices. Its graphical interface allows the users to design fluid circuits and mechanical components with ISO or well-known symbols. Principle quality of AMESim is its powerful solver, completely transparent for users, making choice between algorithms so as to generate the best response for a given system.

Simulink is a Matlab (trademark of the Mathworks inc.) package for dynamics system simulation. It also provide a graphical interface in order to design device models. Through the years, Simulink impose itself as a classical tool in the automation community. Coupling these two simulation tools enables users to model complex physical systems in AMESim and calculate elaborated control laws in Simulink at the same time, as shown in figure 1.

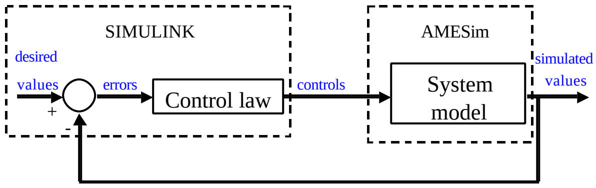

Fig. 1 Cosimulation loop presented as a control feedback loop.

For this simulation task, three possible methods can be followed as shown in figure 2 :

1. Import AMESim model in Simulink as a S-Function (C-code program interpretable by Simulink) and then simulate within only Simulink environment. This corresponds more to a "shared modelling".
2. Import Simulink controller model in AMESim and simulate within only AMESim environment. This is possible thanks to Real Time Workshop (RTW) transducer which translates Simulink graph in C-code (interpretable by AMESim).
3. Cosimulation run : instead of creating a function or program that can be load in one or another environment, just make both models. Device on AMESim, controller on Simulink and links both thanks to the AMESim cosimulation interface. Thus, a dynamical feedback loop is created between both environments with information transmitted regularly during the cosimulation execution. This method is in a complete agreement with the logical loop of figure 1. With a first result obtained through the simulation process, the control law can be translated in C-code with RTW and applied to a test bench via a

dSpace acquisition and control card. Hence, its direct application to real system.

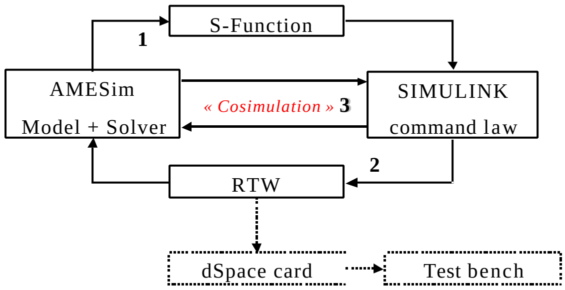

Fig. 2 Pseudo (1-2) and real (3) cosimulation principles.

Advantages of AMESim solver are lost in solution 1, as you need to choose among Simulink algorithms for the simulation. This requires a quite good knowledge of the system before simulating it. For stiff problems, the question becomes rather hard and another difficulty is added for such a principle. In solution 2, a complete recompilation must be generated for any changes in the imported control model. With "cosimulation" methodology subsystems will only exchange inputs and outputs at defined step times during the simulation process. Both solvers are in charge of their own subsystem, preventing any numerical problems. Results analysis can be finally performed within each software environment, taking advantage of conviviality and specific tools depending on the studied variables or states. Thus a maximum potential is kept through this methodology. Moreover, this cosimulation formalism is in a complete agreement with hardware-in-the loop simulation. It becomes very simple to implement the control law design on Simulink to a test bench via a dSpace card.

MODEL

The system under consideration (see figure 3) is an in-line electropneumatic servodrive controlled by two three-way servo-distributors. The stroke length is equal to half a meter and the total moving load is 17 kg.

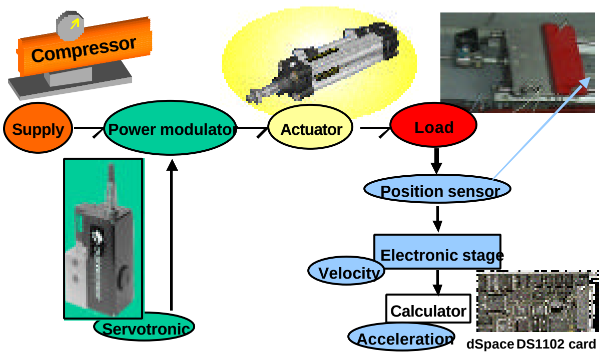

Fig. 3 Principe and example of an electropneumatic system.

Taking account of the work by [4,5,6], the electropneumatic actuator model has been created on AMESim. Each phase is summarized hereafter.

(1) Servo-distributors

Under previous work's considerations, the servo-distributor model has been separated in two blocks, a dynamical stage and a static one. The dynamical part has been identified by a third order transfer function. The static part has been implemented as a values table relative to the mass flow-pressure-control law and obtained by previous experimental measurements [4]. The whole model is introduced in AMESim but can be seen as follows.

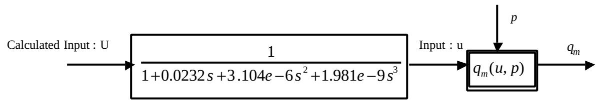

Fig. 4 servo-distributor model.

(2) Jack

The jack model has been designed with AMESim pneumatic components for variable volume pneumatic chamber, pneumatic piston and moving mass with friction, it is shown on figure 5. A polytropic model from AMESim submodels has been chosen for piston chambers. Stiction, Coulomb and viscous forces have been implemented for a better accuracy of the sliding mass. Complete explanations and theoretical laws can be found in [5] and in the AMESim technical documentation [1].

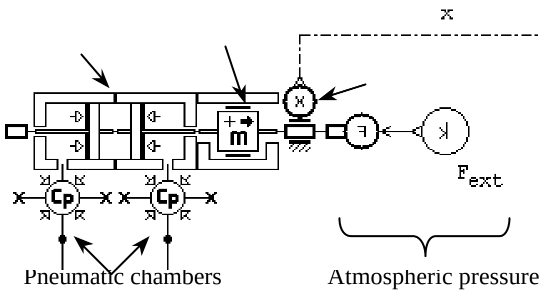

Fig. 5 Pneumatic jack model in AMESim.

(3) Sensors

In the pneumatic field, the conventional position control law consists in position, velocity and acceleration feedback. Using acceleration feedback, instead of pressures feedback or differential pressure feedback, can be justified by the fact that an external perturbation force quickly influences the acceleration. Nevertheless an accelerometer sensor is not conceivable in a lot of applications due to electric wiring. So the acceleration signal has to be derivative. Moreover in the economic aim the velocity is obtained by position derivation. So a double derivation is necessary, which is a difficult problem because it is well known that derivation amplify the noise and leads to a phase difference. After testing experimentally different methods, better results are obtained with a first analog derivation of the

position signal. Then a second numerical derivation gives the acceleration.
In order to get closer to the real data acquisition process, the dSpace card calculation have been modeled within Simulink diagram. Hence, only position is given by physical model from AMESim, whereas velocity and acceleration are computed. Velocity is obtained after an analogic derivation, actually the input signal is filtered and then submitted to differentiation block, while acceleration is obtained by numerical differentiation corresponding to the algorithm :

$$a(k) = \frac{v(k) - v(k-2)}{2 \times Te} \qquad (1)$$

(4) Control law

By separating the actuator model and the control law calculator, several control laws can be studied in an easy way, given that the Simulink loop must be changed. Two kinds of laws have been implemented: with a fixed gains (see equation 2) and with a scheduled gains (see equation 3). For this second one, gains values depend on the cylinder position, giving an accurate feedback all along the jack displacement, whereas fixed gains law only operates around one position state.

$$\begin{cases} U_P = U_P^e + K_x(x_d - x) - K_v v - K_a a \\ U_N = -U_P \end{cases} \qquad (2)$$

$$\begin{cases} U_P(x_d) = U_P^e + K_x(x_d)(x_d - x) - K_v(x_d)v - K_a(x_d)a \\ U_N(x_d) = -U_P(x_d) \end{cases} \qquad (3)$$

The control feedback loop has been generated within Simulink using block diagram editor. Advantages and drawbacks of each control law and gains values can be found in [6]. The whole loop model is shown in figure 6, where are added a zero order hold (the sampling period Te is equal to 4 milliseconds) and a saturation block due to physical constraints (voltage servo-distributors values are between –10 volts and + 10 volts).

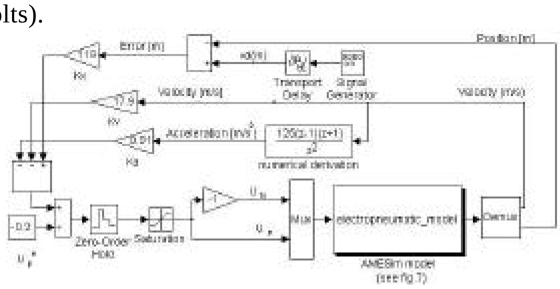

Fig. 6 Control feedback loop design with Simulink block diagram for a cosimulation protocol.

(5) Global model

The cosimulation procedure implies data exchange between the two environments. On one hand, AMESim gives to Simulink the jack position and velocity, on the other hand it receives the computed control law. For such a communication protocol, AMESim creates a specific S-Function that will be the link between both software. This last function is just inserted in the control feedback loop of Simulink, framed on figure 6 and corresponds to the electropneumatic model of figure 7. In AMESim model such links are designed and represented by interface boxes shown in figure 7.

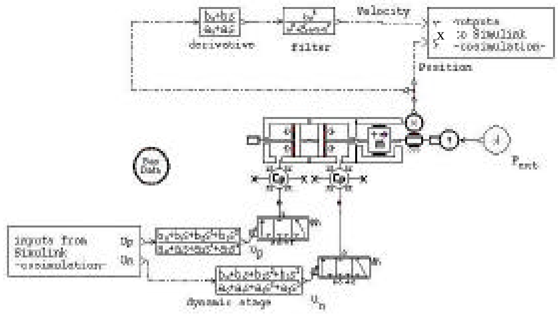

Fig. 7 Cosimulation model within AMESim for a cosimulation protocol.

RESULTS

Thanks to the addition of AMESim and Simulink facilities in a cosimulation, a more accurate model has been designed. A complete dynamical model has been tested and compared to the physical one, for several control laws : fixed gains and scheduled gains. Figure 8 presents the position response for fixed gains control law, desired position is a step between –100 mm and +100 mm of the jack middle. Cosimulation and bench test responses can be directly evaluated for a desired input signal. Simulation result is greatly close to the bench test behavior, proving the model’s accuracy although its complexity due to component model detail. More precise results can be seen on table 1, such as time response and static error. In order to have this detail level, bench test derivation for speed and acceleration have been simulated within Simulink environment. In figure 9, it can be seen that experimental and simulated velocity are rather close. Unstability and discontinuity calculation are skipped at this order of derivation for this study.

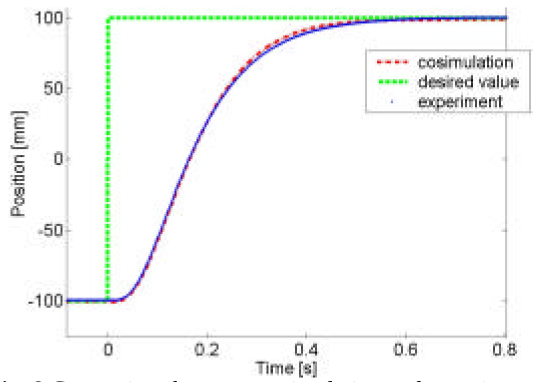

Fig. 8 Comparison between cosimulation and experimental results for the position response with fixed gains control law.

	Cosimulation	Experiment
Static error (mm)	1.0	0.3
Relative error (%)	1.0	0.3
Maximal velocity (mm/s)	895	915
Time response (ms) *	245	259

** response between 10 and 90% of the movement (ms)*

Table 1 Cosimulation and experimental result for fixed gains control law.

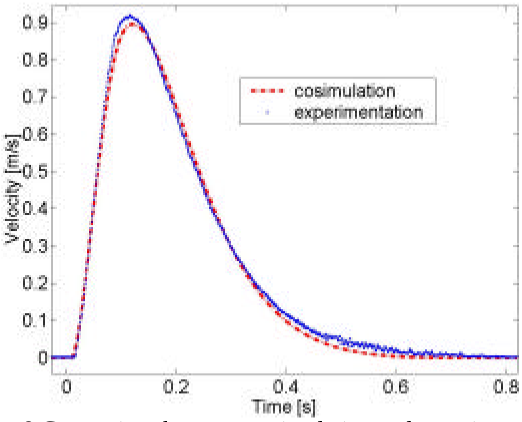

Fig. 9 Comparison between cosimulation and experimental results for the velocity state with fixed gains control law.

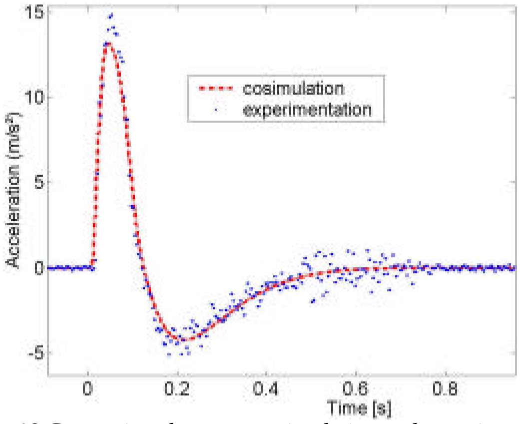

Fig. 10 Comparison between cosimulation and experimental results for the acceleration state with fixed gains control law.

Thus, cosimulation methodology and process has been validated through this study of simple control law. It can be then applied to more complex problem. A second example is presented in figure 11: actuator time response with scheduled gains control law. Here, gains value depends directly on desired position for the cylinder, hence a better control feedback all along the cylinder displacement. The response analyzed hereafter (on figure 11) is for a displacement close to the half right end of the jack (between 170 mm and 220 mm). As right chamber volume becomes rather small, dynamic of the system changes seriously. Hence, a static error occurs between simulation and experimental results. The chosen model, a polytropic one, may not be precise enough at this stage and some thermal exchange could be added in a further study.

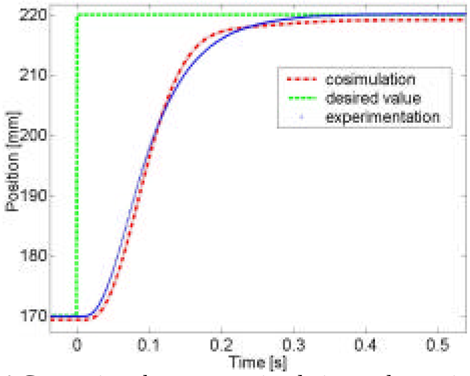

Fig. 11 Comparison between cosimulation and experimental results for the displacement with scheduled gains control law.

CONCLUSION

A new simulation procedure has been tested and validated for a complete mechatronic system behavior analysis. A fixed gains control law has been entirely examined at different levels: displacement, velocity and acceleration of the jack. A scheduled gains control law has been also considered. Through tests results comparison with simulation, cosimulation validity has been shown. With this work methodology, time gain and reduction of costly tests can be obtained. Specialists team work can be also greatly helped by connecting miscellaneous softwares. Moreover, as international network become more and more efficient and fast, the possibility of realizing a simulation process between different geographical areas could be intended in the future.

ACKNOWLEDGEMENTS

This work as been elaborated and developed in collaboration with IMAGINE S.A.

REFERENCES

[1] Advanced Modelling Environment for performing Simulations, Imagine S.A, France.

[2] Choi S.B., Choi Y.T., Park D.W., “A sliding mode control of a full-car electrorheological suspension system via hardware-in-the-loop simulation”, March 2000, Jour. of Dyn. Systems, Measurement and Control, vol. 122 pp. 114-121.

[3] Jansson A., “Co-simulation with AMESim”, 10/20/1999, AMESim workshop, Braunschweig, Germany.

[4] Sesmat S., Scavarda S., S. "Static characteristics of a three way servovalve." In: 12th Aachen Conference on Fluid Power Technology, Aachen, Germany, March 12-13, 1996.

[5] Brun X., Belgharbi M., Sesmat S., Thomasset D., Scavarda S., "Control of an electropneumatic actuator, comparison between some linear and nonlinear control laws", Jour. of Syst. and Control Engineering, 1999, Vol. 213, N°I5, p 387-406.

[6] Brun X., Sesmat S., Thomasset D., Scavarda S., “A comparative study between two control laws of an electropneumatic actuator”, ECC, Sept. 1999, Karlsruhe, Germany, CD ROM, ref. F1000-5.