INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED'07
28 - 31 AUGUST 2007, CITE DES SCIENCES ET DE L'INDUSTRIE, PARIS, FRANCE

A CO-SIMULATION BASED DESIGN METHODOLOGY FOR MECHATRONIC PRODUCTS

F. Bruno, F. Caruso, L. Falbo, M. Muzzupappa
[1]University of Calabria – Department of Mechanical Engineering – 87036 Rende (CS) - Italy

ABSTRACT
The design methodologies traditionally employed to develop mechatronic products are articulated in a sequence of phases: mechanical design, choice of actuators, sensors and other devices, design of the control system. Traditional methodologies usually end with the realisation of the physical prototypes used to perform the tests that are needed to optimise the product, and to verify the effectiveness of the design solutions adopted in the previous phases. The realisation of the physical prototypes is, usually, a complex and expensive task in which design errors and non-optimal solutions, related to the design phase, cause lateness in product launch and development costs increase.
The present paper describes a design methodology devoted to the development of mechatronic products supported by an integrated and interactive co-simulation. This methodology, in fact, allows designers from different domains (mechanical, electronic and software engineers) to follow a classic top-down approach in which, starting from the conceptual phase, they can check, at each milestone of the project, the functionality of the product at different stages and for different levels of detail.
At the end of the design process the engineers can directly interact with the mechatronic model by using the product interface implemented in the control system. The actions of the user are processed by the electronics simulation software that dialogues with actuators, motors and sensors placed in the multi-body model inside another specific simulator. So the 3D model in the multi-body software interactively responds to the actions performed by the user.

Keywords: Mechatronics, design methodology, co-simulation, CAE, CACE, top-down design.

1 INTRODUCTION
In recent years, the interest of both industrial companies and universities in mechatronics has been noticeable, and thus, mechatronics has undergone a deep development.
Mechatronic systems design is based on an interdisciplinary combination of domains: mechanical engineering, electrical engineering, electronics, and computer science. They are characterised by a tight coupling of different implementation technologies, e.g. hydraulics, mechanics, pneumatics, electromechanics, electronics and software [1].
For this reason, the development process for mechatronic products is different from others, in the sense that it spans over many closely coupled engineering domains. In recent years, existing design methodologies of mechatronic systems have undergone a progressive advancement required by the need to manage a rising of design complexity, reduce time-to-market, integrate heterogeneous fields like electronics, mechanics, and software into a single product. This has stimulated a steadily enhancement of existing design environments, tools, and methods. Such development process typically applies a subsystem-based approach. The term "subsystem based" refers to a product development strategy by which integrated systems are built from technology homogeneous subsystems (mechanics, electronics, control and software). The subsystems are developed in a concurrent manner focusing, in particular, on subsystem interfaces. Once the interfaces are designed, each subsystem is designed in a fairly traditional way. The subsystem-based approach to mechatronics is a further improvement to the traditional approach through which mechanical engineers first designed the mechanical system and then handed it over to the control engineers [2].
It is very useful to validate the project of the entire system through simulations, before building the physical mock-up. The realisation of the physical prototypes, in fact, is usually a complex and

expensive task in which design errors and non-optimal solutions, related to the design phase, cause lateness in product launch and development costs increase.
In order to simulate a mechatronic system, a multi-domain simulation environment is required. Multi-domain simulation could be achieved in two different ways: the first and more traditional way is to use a general-purpose solver to simulate the entire system, while the other way, called co-simulation, is to use different communicating solvers, to simulate each sub-system. Each of these multi-domain simulation techniques has its own advantages and disadvantages. One of the main advantages of the co-simulation is that each solver is optimised for its own specific domain.
In fact, from the software point of view, co-simulation is not only a simulation performed in two or more different physical domains, but it is a simulation in which several softwares run in parallel and continuously exchange data to verify the behaviour of each different component of the product under specific working conditions.
Furthermore, co-simulation emphasises the subsystem design approach, in which the interfaces among the sub-systems remain the same, but an inter-changeability among the possible models of a sub-system is allowed.
By using co-simulation, in fact, it is not necessary to import the several models of different domains into one multi-purpose simulation software, but each model is simulated in its own environment without any import operation. Whereas, the main disadvantage of the co-simulation approach is represented by the non trivial effort to synchronise the different solvers, though many simulators may be interfaced.
In our opinion co-simulation techniques are still not sufficiently exploited, because they are employed only at the end of the design process to validate the choices made by engineers. Whereas co-simulation can be intended as a technique which is able to support designers also in the earliest stages of the design process.
The present paper describes a design methodology devoted to the development of mechatronic products supported by an integrated and interactive co-simulation. This methodology, in fact, allows designers from different domains (mechanical, electronic and software engineers) to follow the top-down approach in which, starting from the conceptual phase, they can check the functionality of the product at different stages and for different levels of detail.
At the end of the design process, the engineers can directly interact with the mechatronic model by using the product interface implemented in the control system. The actions of the user are processed by the electronics simulation software that dialogues with actuators, motors and sensors placed in the multi-body model inside another specific simulator. So the 3D model in the multi-body software interactively responds to the actions performed by the user.
Our case study, implemented to test the proposed methodology, is the complete design of a mobile elevated working platform. The machine has four driving and steering wheels, four outriggers provided to make the machine stable and an articulated arm with six degrees of freedom. There are 22 actuators for the movement of the various parts and each of them has a sensor position for feedback control. Thanks to the availability of the integrated behavioural model of the machine, it has been possible to implement a functionality for the automatic reconfiguration of the outriggers that allows the user to move the platform by employing only three outriggers at a time.
The case study presented puts in evidence that co-simulation can be an innovative technique to develop new design methodologies devoted to the mechatronic field. Many hi-tech industries can benefit from this approach that allows designers from different domains to check their idea in an integrated model that includes the mechanics, electronics and software models.

2 STATE OF THE ART

The key of a mechatronic design is concurrent and multi-disciplinary engineering. One of the earliest approaches to the multi-disciplinary design for mechatronics was described in [3] and [4]. The methodology described in these papers was based on a software platform, called Schemebuilder, that is a design tool aimed at guiding designers through the several design options, and assists the designer in the conceptual and embodiment stages of design. It can be considered like a multi-disciplinary decision support system.
Another example of mechatronic design methodology is described in [5], in which a general model is derived to mathematically describe the concurrent design of a mechatronic system. Based on this model, a concurrent engineering approach is formally presented for mechatronic systems design.

Companies like Bosch GmbH have also made a fine effort in the formulation of mechatronics design methodologies [6]: Bosch started the initiative "Systems Engineering Mechatronic" to cope with the challenges of system design and complexity handling. Targets are high design efficiency (reduction of development time and cost) as well as high design quality (design correctness). New design methodologies and processes are currently established in different business units (automotive equipment, consumer goods and industrial equipment). Systems Engineering Mechatronic is based on the V-model for system design [6].
Artificial Intelligence (AI) and genetic algorithms are used in some design and optimisation methodologies of frontier researches: in [8] the Niching genetic algorithm (described also in [9], [10] and [11]) is used to find local and global optimal design alternatives, while [12] presents a methodology for automatic generation of robot high-level controls using a new evolutionary methodology [13].
In [14] the development of a brake by wire system is presented. The simulation is performed at the early design stages of the design process, to depict any possible design flaw. Authors point out how analysing and simulating a mechatronic system at an early design stage is of great help to find the optimum design solution. Unfortunately, they present only the case study, without deriving a general methodology nor giving any suggestions to employ simulation techniques in the different stages of the design process.
Co-Simulation has also been used within several companies. Ford Motor Company used it to simulate a Vehicle Attitude Control (VAC) System [15]. The authors present a comparison between a simulation based on a single solver; and another one based on co-simulation techniques. They report that co-simulation provides a more complete representation of both vehicle and control system by selectively using the strengths of each application, but it requires more computational resources.
Visteon Corporation used co-simulation to simulate a vehicle stability control system [15]. On the basis of the mathematical models developed, a virtual prototyping vehicle to simulate winter test events was used to validate the applicability of driveline torque controls to maintain vehicle stability. These results could serve as the upfront analysis of the control strategies and the influence of torque biasing devices on vehicle responses before building the real hardware.

3 METHODOLOGY

The basic idea of the methodology herein presented, is to enrich the classical top-down design approach with the use of co-simulation tools. In other words, the methodology establishes a cooperation among the various specialists in each engineering domain (mechanics, electronics, software, etc.) by defining some milestones in the product development process in which all the contributions from each domain are integrated and tested together.
The methodology is generally valid for a large set of mechatronic products, but it is particularly intended for the development of product characterised by a set of moving components that have to be studied in different possible configurations determined by the user and/or by a control logic. These kinds of products are frequently used in various fields like robotics (e.g.: motorised arms), automotive (e.g.: windshield wipers), etc.
At each step of this top-down design methodology the DMU is progressively defined and its components are detailed. Each version of the DMU is validated by a co-simulation that involves both the mechanical and the control model. If the result of the co-simulation is the desired one, then the designers can improve and refine their models for the next step.
At the early stages of the methodology, the designer defines some macro-components that are the main parts of the product used to sketch a preliminary kinematic model. These macro-components are initially identified with a simplified geometry formed by few primitives. Later, the macro-components are detailed with one or more components and the kinematic model is enriched with forces, torques, masses and other data to become a dynamic model. At the last step of the methodology the final geometries are modelled and inserted in the dynamic model. So it can be enriched with friction and the estimated masses are substituted with the real one calculated on the final geometries, knowing the density of the material.
The main steps and tasks of the methodology are the following.
<u>Step 0: Conceptualisation</u>

0.1 Roughly dimension the product and define the main functions and macro-components.
0.2 Identify the main performances and constraints of the product.

Step 1: Kinematic model design

1.1 Define a rough multibody model formed by the overall kinematics, constraints and degrees of freedom for each macro-component.
1.2 Define a preliminary interface between the product and the user and/or the environment.
1.3 Design a first control logic.
1.4 Connect the control logic with variables (position and velocity of each macro-component) defined in the mechanical model.
1.5 Test the kinematic co-simulation model and evaluate performance and constraint satisfaction.
1.6 Perform the necessary modification on the models until the desired performances are reached and the constraints are satisfied, as defined in task 0.2.

Step 2: Dynamic model design

2.1 Detail the multibody model by defining the dynamics of each component and estimating physical properties like forces, inertia, mass.
2.2 Refine actuators and sensors in the mechanical model.
2.3 Refine the control logic and update it to manage the new connections with the mechanical model.
2.4 Test the dynamic co-simulation model and evaluate performance and constraint satisfaction.
2.5 Perform the necessary modifications on the models until the desired performances are reached and the constraints are satisfied, as defined in task 0.2.

Step 3: Detailed design

3.1 Import macro-components in the CAD and use them as a reference for modelling in the next task.
3.2 Model the geometries of each component with the CAD.
3.3 Import the detailed geometries in the multibody simulator, substituting the macro-components previously defined.
3.4 Define friction and materials for each detailed geometry in the mechanical model.
3.5 Implement the control logic defined in task 2.3 with electronic components and software code.
3.6 Test the final co-simulation model and evaluate performance and constraint satisfaction.
3.7 Perform the necessary modification on the models until the desired performances are reached and the constraints are satisfied, as defined in task 0.2.

Modern Computer Aided Control Engineering (CACE) software also allows control engineers to extend the described methodology with a hardware-in-the-loop test. To do this, the engineers build the control hardware defined in task 3.6 and load the code on it. The hardware can be connected to a PC running the CACE software, so the real control hardware can be validated in conjunction with the virtual mechanical model. The top-down diagram of the product obtained during the application of the methodology is shown in figure 1.

4 BENEFITS AND DRAWBACKS

The main problems related to the application of this methodology are:

- The need to have a good estimation of the work required by the different specialists that collaborate in the design process. In fact, the realisation of a complete functional digital mock-up (DMU) requires that both the mechanical and control models have to be ready for each milestone in which a test of the co-simulation model is needed (tasks: 1.5, 2.4 and 3.7). A delay in the mechanical or control design could create a bottleneck in the product development process.
- The estimation of the physical properties for each macro-component (task 2.1) strongly depends on the information initially estimated and on the designers' skills.
- Mechanical models must be designed in the CAD environment and then imported to the multi-body solver. During this process, all the constraints and loads will be lost, and must be redefined within the simulation environment. This is actually a time-consuming task.

On the other hand, the advantages offered by this methodology are that:

- It is possible to execute a top-down design both on the mechanical and the control side, validating each step with the co-simulation.

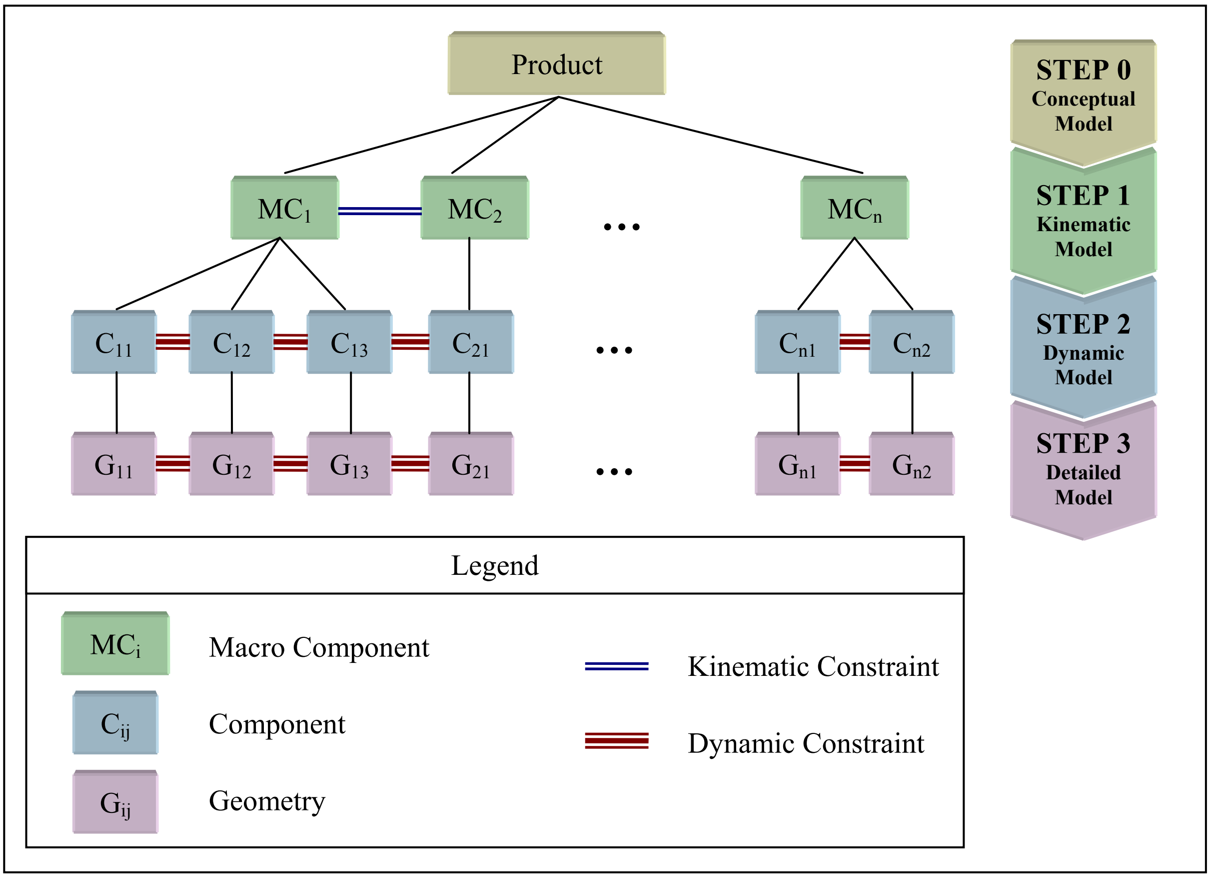


Figure 1. Schema of top-down product structure obtained applying the methodology proposed

- It is possible to reduce the occurrence of design errors in the final steps of the product development process. This is particularly true for those errors related to the misunderstandings between mechanical and control engineers.
- When the product has a user interface, this can be sketched, starting from step 1, using the CACE software. So functional and usability tests can be conducted with the participation of the user also in the first stages of the product development process.
- Using dedicated software allows the designer to achieve a very high detail in the mechatronic model, both from the mechanical and electronic points of view.

5 CASE STUDY

In this section the application of the methodology on a case study will be described. The object of our case study is a mobile elevated working platform with an articulated arm. The final CAD model of the product is shown in Figure 2.

The platform is provided with four driving and steering wheels. A great operational flexibility is guaranteed by four variable-length stabiliser legs. Every leg has 3 d.o.f. (degrees of freedom), that allow multiple support configurations. The articulated arm, conceived to easily cover a large work-volume, has 6 d.o.f.. There are 22 actuators for the movements of the various parts and each of them has a sensor position for the feedback control.

Thanks to the co-simulation results, we were also able to implement a control-logic that allows the dynamic reconfiguration of the support stabilisers, making use of the balance of the whole structure on three of the four contact points with the ground.

Figure 2. The working platform analyzed in the case study

The platform is provided with four driving and steering wheels. A great operational flexibility is guaranteed by four variable-length stabiliser legs. Every leg has 3 d.o.f. (degrees of freedom), that allow multiple support configurations. The articulated arm, conceived to easily cover a large work-volume, has 6 d.o.f.. There are 22 actuators for the movements of the various parts and each of them has a sensor position for the feedback control.

Thanks to the co-simulation results, we were also able to implement a control-logic that allows the dynamic reconfiguration of the support stabilisers, making use of the balance of the whole structure on three of the four contact points with the ground.

According to the methodology explained above, the design of the mechatronic model uses co-simulation as a test tool, in order to validate the design choices and to develop the model on gradually decreasing levels of abstraction. Interactive co-simulation with the controlled machine is realised by drawing the higher performances through the integration of the ADAMS and MATLAB/Simulink softwares (Figure 3).

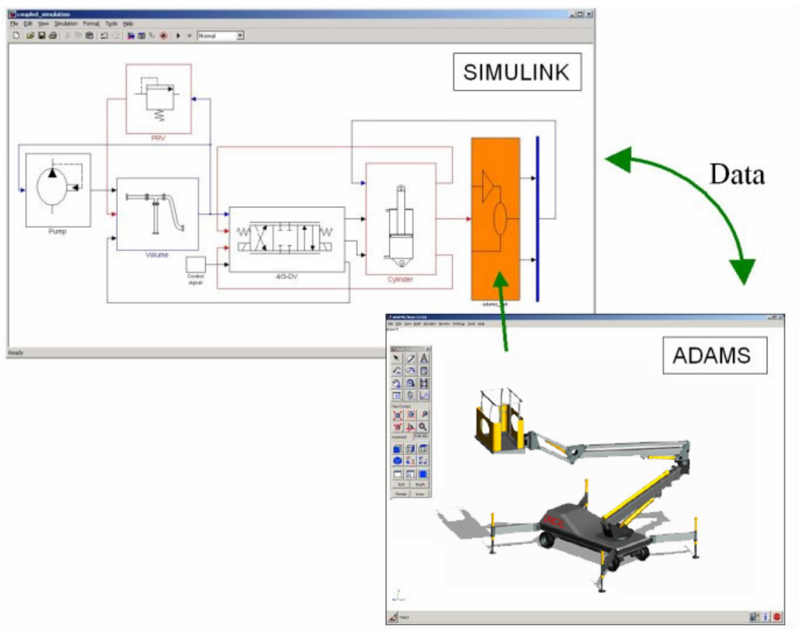


Figure 3. Software integration for co-simulation

The interface between the Matlab/Simulink environment and the Adams environment is the building block highlighted in orange called Adams-SubBlock (Figure 4). It is generated using Adams/Controls, specifying all the input and output control variables (e.g. positions, velocities, pressures, etc.) of the mechanical system that the user wants. In fact, Adams-SubBlock is the link between the control electronics developed in Simulink and the mechanical system of the machine modelled in Adams. Simulink can import it as a special building block called S-Function. From the Adams point of view, this block receives forces in input and gives the position of the actuators in output. From the Matlab point of view, the Control Block receives positions in input and returns forces in output.

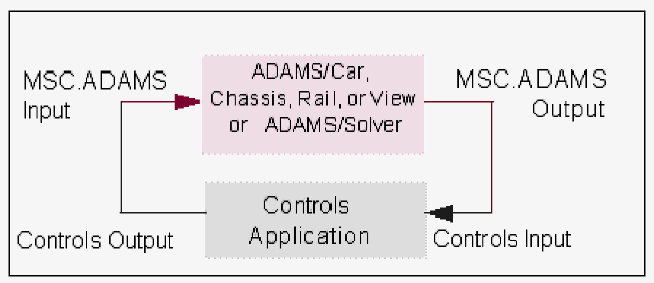


Figure 4. Inputs and outputs of the interface

Within this block, all the parameters regarding simulation and communication are defined. It is possible, in fact, to specify the time-step of the simulation, communication.
During the co-simulation, all the parts that formed the 3D model are moved according to the simulation results.

5.1 Step 0

The design idea is initially conceptualised by the rough definition of the volumes and the functional geometry of the mechanical model, represented in ADAMS by elementary solid elements (parallelepipeds, cylinders, etc.) (Figure 8a). Three fundamental sub-systems can be recognised: basket, articulated arm and base.

5.2 Step 1

The kinematics to match the functional targets of the machine is defined for each macro-component. The articulated arm is disassembled into two main groups. The first one can rotate on two axis, one vertical and the other horizontal, and it is constrained to the base. The second group is formed by two quadrilateral elements, and it is linked to the first group by a joint which allows vertical translation only. The base of the platform can be subdivided into two macro-components: the driving system and the stabilisation system. The first one is formed by four motors, acting on each wheel. A kinematic mechanism rotating the four wheels simultaneously has been developed for the steering system.
The stabilisation system can lift the platform upon four stabilising legs. In this phase only the kinematics of the multibody model is tackled. Therefore, the building of the multibody model (Figure 8b) is easy and rapid because the system dynamics is not considered. At the same time, the user interface is defined. It is formed by two joysticks, thanks to which the user can interact with the virtual model (Figure 5 on the left). In this case study an environment interface is not necessary (point 1.2 of methodology) because all the inputs of the mechatronic system come only from the user interaction. In the MATLAB/Simulink environment, the control logic is implemented. It manages the use of the joysticks connected to the control variables (position and velocity of each macro-component). These are defined in ADAMS for each kinematic constraint. The user can interact and drive the multibody model thanks to the joysticks, depending on the degrees of freedom allowed by the constraints.
Co-simulation allows one to test input devices functionality and the kinematics of the mechanical system, evaluating volumes, working-space and flexibility in positioning. One of the design targets is to maximise working-volume against idle volume and, during this phase, it is possible to evaluate the volumes, of both the working-area of the articulated arm, and the idle configuration of the machine. Supported by the visual information coming from the co-simulation output, the designer can improve kinematic mechanisms by finding alternatives to the original schema. Each leg can rotate about 180 degrees from the base, and telescopically lengthen itself to confer more stability to the platform. This

allows a flexible functioning of the system, because it is possible to reduce the room required by the platform when a wide action-radius of the arm is not required. Vice versa, when a wide action-radius of the arm is required, every leg can be extended independently. Furthermore, vehicle driving is simulated by evaluating the manoeuvrability of the platform.

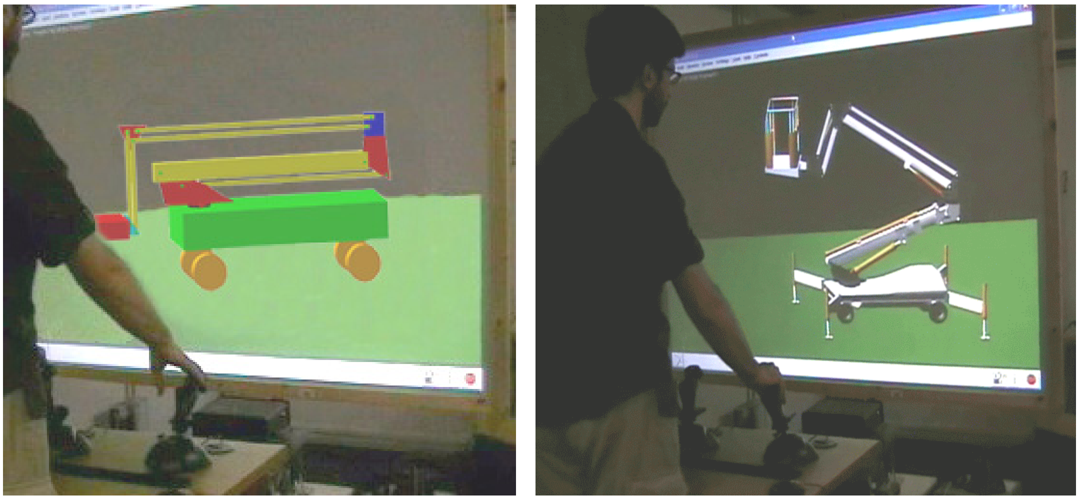


Figure 5. The methodology proposed allows designers to validate the digital mock-up of the platform interacting with a pair of joysticks, starting from step 1 with the kinematic model (image on the left), up to step 3 with the detailed model (image on the right).

5.3 Step 2

In this phase, the kinematic model is enriched with information and data to obtain a dynamic model (Figure 8c). To evaluate the dynamics of the mechatronic model, it is necessary to define every macro-component. The geometries of the hydraulic pistons used as actuators for the articulated arm are sketched for every sub-system. Subsequently, the weight of each part has to be estimated. In this way, it is possible to evaluate the dynamic behaviour, and, it is also possible to improve the control logic. Once end-stroke sensors and pressure sensors are placed, it is possible to develop a coherent control logic, taking into consideration the stability of the system and the forces and the torques that each actuator should provide. It is easy to understand that the stability of the platform is linked to the dynamics and the position of the articulated arm; therefore, the availability of the multibody model in the co-simulation environment is very useful to achieve this target. The critical condition of the anti-reversal control logic starts when the pressure of one of the legs is zero. When the critical condition is verified, all the manual controls that could worsen the equilibrium are disabled. During this phase the power of our design methodology is highlighted, because it is possible to foresee the behaviour of the whole mechatronic system before the detailed modelling. The aim of the automatic safety control is to grant the platform stability under every load condition with every possible configuration of the legs. Two of the possible working volumes, for two different leg configurations, are shown in Figure 6.
The flexibility of the legs allows the platform to adapt itself to several morphological configurations of the ground. A control logic for a dynamic re-configuration of the legs has been implemented. It allows the whole system to maintain the equilibrium upon three contact points with the ground. If the centre of gravity is within the area bounded by the three legs, the fourth one can be repositioned. For this reason the control logic allows the movement of the unloaded leg only.

5.4 Step 3

In this phase each component is defined with a detailed geometry (Figure 8d). Each part is modelled using a CAD software, using all the references developed in the previous phases as a skeleton (Figure 7).

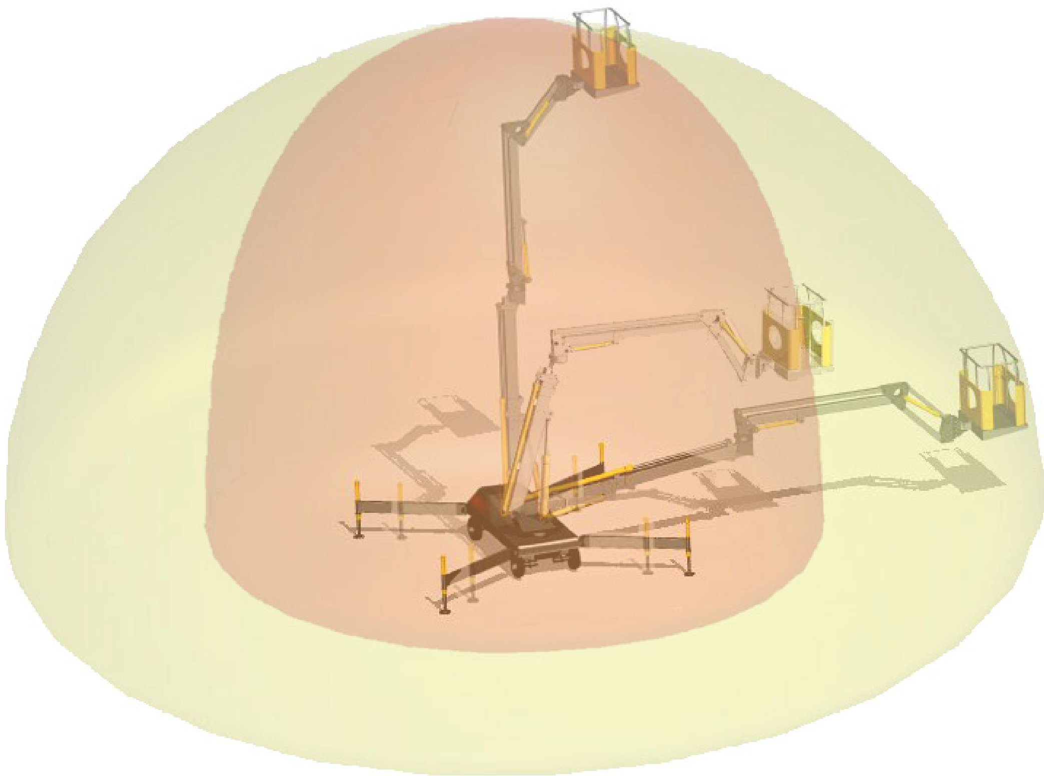


Figure 6. Two possible working volumes for two different leg configurations.

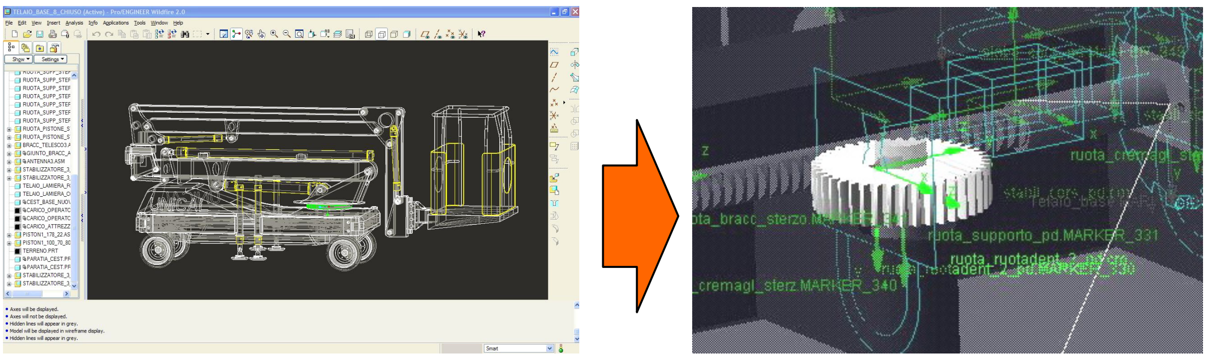
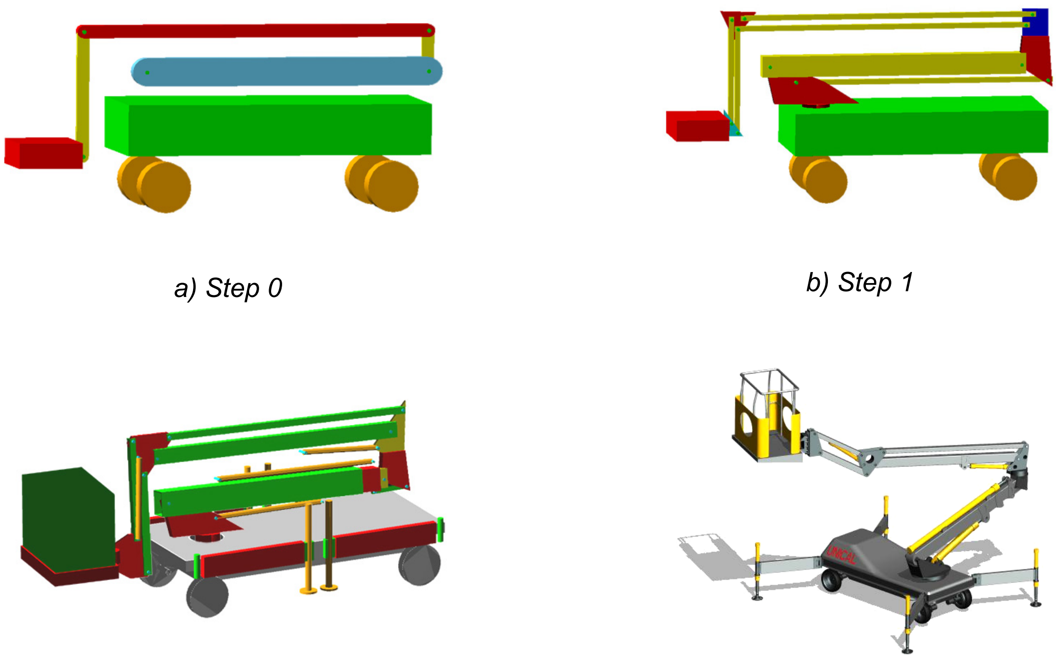


Figure 7. The detailed geometries modeled in Pro/Engineering are used to build the final multibody model of the platform in ADAMS.

Once all the parts are modelled, these are imported to the simulator, and all the mass properties and friction coefficients are automatically calculated by the simulator once the material properties are specified. After the values of the parameters in the Simulink block, relative to actuators, are updated in keeping with the new properties, the final simulation occurs. The complexity of the multi-body model is quite high, and the computational load is not affordable by a mainstream calculator. The several co-simulations performed, highlight an overall slowness in simulation. Therefore, we have chosen to split the functional digital mock-up into two co-simulation models. The first regards just the drive and the lift of the platform, while the second regards just the moving of the articulated arm.



a) Step 0 b) Step 1

c) Step 2 d) Step 3

Figure 8. The working platform at different steps of the methodology.

6 CONCLUSIONS

In the mechatronic field, co-simulation is usually employed as a tool to improve efficacy and accuracy of the models employed in the validation of the functional digital mock-ups. We think that this approach limits the advantages that engineers can obtain from co-simulation, so we have proposed a methodology that exploits the power of co-simulation techniques since the earliest stages of the design process. The main idea is to apply co-simulation in conjunction with the classical top-down design approach. The result is that the mechanical and the control models can be jointly validated during each of the three steps of the methodology described, also when the models have not been completely detailed. In particular, the methodology proposes to abstract the model initially by considering only the kinematics and later by refining the model with the dynamics and the final geometries. The control model is also progressively detailed in keeping with the results obtained at each step.
This approach improves the communication between mechanical and control engineers because they can jointly validate their ideas and assumptions at each milestone of the project, reducing the possibility of design errors occurring during the final steps of the product development process.
Moreover, this methodology could be a good support for participatory design, because functional and usability tests can be carried out, with the participation of the user, as from in the first stages of the product development process.
The proposed case-study points out that the methodology is also applicable in complex situations and that the maturity of the co-simulation tools allows designers to develop precise and complete virtual models of the products. The power of the hardware which is currently available also allows engineers to interact with the virtual model by using (when it is possible) a user interface which is very similar to the real one.

REFERENCES

[1] Isermann R., *Mechatronic systems: Fundamentals.* 2003 (Springer-Verlag)
[2] Wikander J., Törngren M. , Hanson M., Science and education of mechatronic engineering. *IEEE Robotics & Automation Magazine*, 2001, 8(2), 20-26.
[3] Bracewell R.H. and Sharpe. J.E.E. Functional descriptions used in computer support for qualitative scheme generation - "Schemebuilder" - *AI EDAM Journal.* Special Issue:

Representing Functionality in Design, volume 10, 1996 pages 333-346.
[4] Bracewell R.H., Bradley D.A., Chaplin R. V., Langdon P.M., Sharpe and J.E.E. Schemebuilder, a design aid for the conceptual stages of product design. *International conference on engineering design. Iced '93*. The hague, august 17-19, 1993.
[5] Li Q., Zhang J., Chen L., Design for Control *IEEE/ASME TRANSACTIONS ON MECHATRONICS*, Vol. 6, NO. 2, June 2001
[6] Dieterle W., Mechatronic systems: Automotive applications and modern design methodologies. *Annual Reviews in Control.* Vol. 29, 2005, page 273-277
[7] Boehm B. W., A Spiral Model of Software Development and Enhancement, *Computer*, Vol. 21, No. 5, pp. 61-72, 1988.
[8] Behbahani S., de Silva C. W., A New Multi-Criteria Mechatronic Design Methodology Using Niching Genetic Algorithm. *IEEE Congress on Evolutionary Computation.* Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada July 16-21, 2006
[9] Cho D. H., Jung H. K., Lee C. G., Induction motor design for electric vehicle using a niching enetic algorithm *IEEE Transaction on Industry Applications*, Vol. 37, No. 4, pp. 994-999, July/August 2001.
[10] Kim J. K., Cho D. H., Jung H. K., C. G. Lee, Niching genetic algorithm adopting restricted competition selection combined with pattern search method. *IEEE Transaction on Magnetics*, Vol. 38, No. 2, pp. 1001-1004, Mar. 2002.
[11] Himeno M., Himeno R., The niching method for obtaining global optima and local optima in multimodal functions. *Systems and Computers in Japan*, Vol. 34, No. 11, pp. 30-42, 2003.
[12] Sakka S., Coiffet P., Contribution to Design of Complex Mechatronic Systems. An Approach through Evolutionary Optimization. *Journal of Intelligent and Robotic Systems.* Volume 42 , Issue 1 Pages: 1 - 25 January 2005
[13] Ban, A. and Mayer, H.: PS Natural Computation Experimental analysis of evolutionary algorithms using chromosomes of variable length 2003.
[14] Roos F., Wikander J., Mechatronics design and optimisation methodology *Mekatronikmöte 2003*, Gothenburg, Sweden, August 2003
[15] Villec G. N., Cosimulation of an Automotive Control Systemusing ADAMS and Xmath. *International ADAMS User Conference* 1998
[16] Liu C.S., Monkaba V. , Tan H., McKenzie C., Lee H., SuoSae S., Driveline Torque-Bias Management Modeling for Vehicle Stability Control. *SAE Automotive Dynamics and Stability Conference and Exhibition* Detroit, Michigan May 7-9, 2002

Contact: Fabio Bruno
University of Calabria
Department of Mechanical Engineering
via P.Bucci 44/c
87036, Rende (CS)
Italy
tel.: +39 0984 494623
fax.: +39 0984 494673
f.bruno@unical.it
www.unical.it