Dynamic simulation of the motor vehicles using commercial software

Cătălin ALEXANDRU

University Transilvania of Braşov, Braşov, 500036, Romania

Abstract The increasingly growing demand for more comfortable vehicles nowadays imposes a new way of analyzing, with elaboration of models that are closer to the real systems on the vehicle. The simulation techniques allow the automotive engineers to conceive and equip virtual prototypes that permit a large-scale evaluation of the vehicle behavior. In the present paper, we attempt to carry out the simulation of the motor vehicles, using the commercial software ADAMS of MSC / MDI.

Keywords: multibody system, virtual prototype, dynamics, simulation.

1. Introduction

Two decades ago, in automotive engineering, the concept of transferring of the drawings in the computer was introduced. At the same time, major improvements were developed, enabling engineers to move from creating simple, two-dimensional drawings, to modeling three-dimensional solids.

In the last years, on the basis of advanced design prototyping software, automotive engineers have the possibility to build models of not just parts but entire mechanical systems of the vehicle, and then to simulate their behavior and optimize the design before building an expensive physical prototype. This technology, called Virtual Prototyping, is a software-based engineering process, which enables modeling, analyzing, optimizing, and simulating, respectively, the mechanical systems of the motor vehicles under real operating conditions.

Virtual Prototyping brings several advantages: it reduces the time and cost of the new product development; it reduces the product cycles; it reduces the number of expensive physical prototypes and experiment with more design alternatives. In this way, automotive engineers can quickly explore multiple design variations, testing and refining until optimizing the dynamic behavior, long before building the first physical prototype.

ADAMS (Automatic Dynamic Analysis of Mechanical Systems) of MSC - MDI is a powerful computer program for virtual prototyping of the mechanical systems, and includes a specific module, namely ADAMS/Car, to analyze the vehicle subsystems such as suspension, steering, car body, power train, brake, or anti-roll bar. The vehicle can be tested under various road conditions in the computer, performing every maneuver normally run on a test track to accurately predict handling characteristics such as body roll, ride quality including vibration and bumps, safety and performance parameters of the vehicle.

ADAMS/Car has two interfaces (i.e. modes): Standard Interface - to create and analyze assemblies of suspensions, steering subsystems and full vehicle assembly, and Template Builder Interface - to create new templates (suspension, steering, car body, power train, brake, anti-roll bar etc.) for use in the Standard Interface. The type of interface depends on the type of user access, as follows: standard users - for Standard Interface, and expert user - for Template Builder Interface.

2. Modeling Process

In ADAMS/Car, the assembly file points to subsystem files. The subsystem file points to a template file. The subsystem is simply a slightly modified version of the template; it is a small ASCII file containing hardpoint coordinate information, as well as spring & damper information etc., and a pointer to the referenced

template file. For example, the user can have a lot of different double wishbone front suspension subsystems that all reference the same double wishbone template. Each of subsystems may have different hardpoints, springs, bushings, and/or dampers, and they would all be essentially slight modifications of the referenced template.

The template is a large file (either binary or ASCII) that has complete ADAMS model information for a particular subsystem. A subsystem file has limited information, mainly the type of information that a standard (non-expert) user can modify, from the Standard Interface.

If the user wants different parts, or different types of parameterization than what is currently found in the shared database templates, the user can certainly create it own template (or modify an existing one), as an expert user in the Template Builder Interface of ADAMS/Car. The idea is to have as few templates as necessary to completely describe a subsystem model, and minor changes to these would be handled via the subsystem files, generated from the Standard Interface.

Every subsystem has a major role and a minor role. A major role describes the subsystem's function in the assembly. Examples of major roles are suspension, steering and car body. The minor role further defines the subsystem's location. For example, a minor role can be front or rear.

To create subsystems, a template that defines the topology and default data for the subsystem must be selected. At the same time, the minor role of the subsystem will be indicated. To create an assembly, different types of subsystems will be selected - one for each major role (rear and/or front suspension, steering, car body, power train, etc.).

A suspension assembly contains a suspension subsystem (front or rear suspension), and could contain, also, a steering subsystem and an anti-roll bar subsystem. A full-vehicle assembly contains two suspension subsystems (front and rear), a steering subsystem, two wheel subsystems (front and rear wheels) and a car body subsystem. It could contain, also, brake and power train subsystems.

3. Communicators

In an assembly (suspension and/or full vehicle assembly), the communication between the subsystems that form the assembly is made through different communicators. The communicators are ADAMS/Car key elements that provide the subsystems to exchange data with each other and with the testing systems. Regarding the test rigs, in ADAMS/Car two types of analyses can be made: suspension analysis, which is always a quasi-static analysis, and full vehicle analysis, which is a dynamic analysis. For each type of analysis, a test rig exists: suspension test rig and driver test rig (for entire vehicle simulation).

An assembly requires two directions of data transfer between its subsystems. To provide for these two directions of data transfer, two types of communicators are used: input communicators, which request information from other subsystems or test rigs, and output communicators, which provide information to other subsystems.

For example, a mount communicator in the rack-pinion steering template outputs the rack part name so those tie rods of suspension templates can attach to the rack. In addition, a mount communicator in the steering template inputs a part name. The steering template uses the part name to determine where to attach the steering column.

The class of a communicator indicates the kind of information it exchanges. For example, communicators of the class hardpoint exchange a location through a point name and a part name. The frequently used communicators and the information that each class exchanges are listed in table 1. The classes apply to both input and output communicators. A communicator can be either single or be part of a symmetrical pair, either left or right. Entity classes (array, differential equation, motion, parameter variable etc.) have no symmetry and, therefore, are always single.

Each communicator has a minor role that defines the communicator's position in the assembly: front, rear, trailer, inherit and any. For inherit, the minor role of the communicator will become that of the subsystem using the template.

Table 1: types of communicators in ADAMS/Car.

Mount	Part name to provide connections between subsystems
Marker	Point and part name to provide both location and part information
Joint	Joint name
Joint for motion	Joint name
Location	The location of the named point or construction frame
Motion	Motion name
Part	Part name
Orientation	The orientation of the named construction frame
Real parameter	A parameter variable name of the type real
Integer parameter	A parameter variable name of the type integer

Assigning a prefix to the name makes communicator naming: "ci" indicates an input communicator and "co" - an output communicator; "l" indicates the left side of a symmetrical pair, "r" - the right side and "s" - the single communicator. For example, an input communicator for the left side will have the prefix "cil", followed by the name (cil_suspension_mount).

4. Running Analyses

To analyze a suspension or vehicle assembly, the following must be specified:
- the assembly (model) to be tested, by opening or creating an assembly that contains the suitable components, or subsystems those make up the prototype;
- the test rig to be used to input the specified excitations on the prototype;
- the type of analysis that will be performed, by selecting one from the ADAMS/Car Simulate menu;
- the analysis inputs to be used, by typing them directly into an analysis dialog box or by selecting a load case file that contains the desired inputs from a database.

The suspension analysis allows learning how the suspension controls the wheel motions and transmits the load from the wheels to the car body. To perform a suspension analysis, first create or open a suspension assembly that contains the selected components, or subsystems, and the test rig. Then, the ranges of vertical wheel travel, steering travel and static tire contact patch loads and the number of solution steps must be specified.

During the analysis, the test rig articulates the suspension assembly in the specified number of steps and applies the inputs you specified. At each step, the suspension characteristics are calculated (for example, toe angle, camber angle, track change, wheels base change, wheel rate, etc.).

The previously created suspension subsystems will be then integrated with other subsystems to create a full-vehicle assembly. Various maneuvers on the entire vehicle can be performed to test the design of the different vehicle's subsystems and see how they influence the dynamic behavior of the vehicle.

5. Application

Virtual prototyping has become very important in a lot of applications specific to automotive industry. Every major manufacturer uses virtual prototyping in different types of applications (fig. 1), for example: suspension design, vehicle dynamics, engine design, power train design, body engineering, and so on [8].
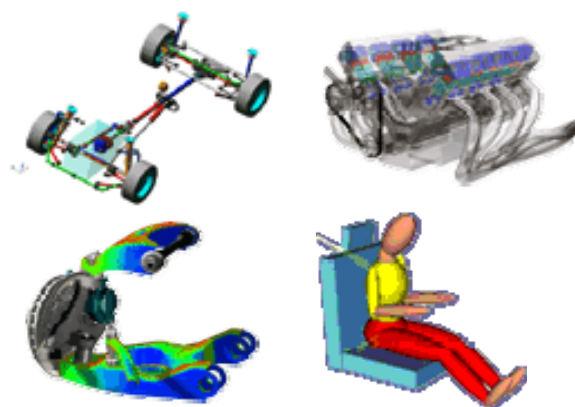

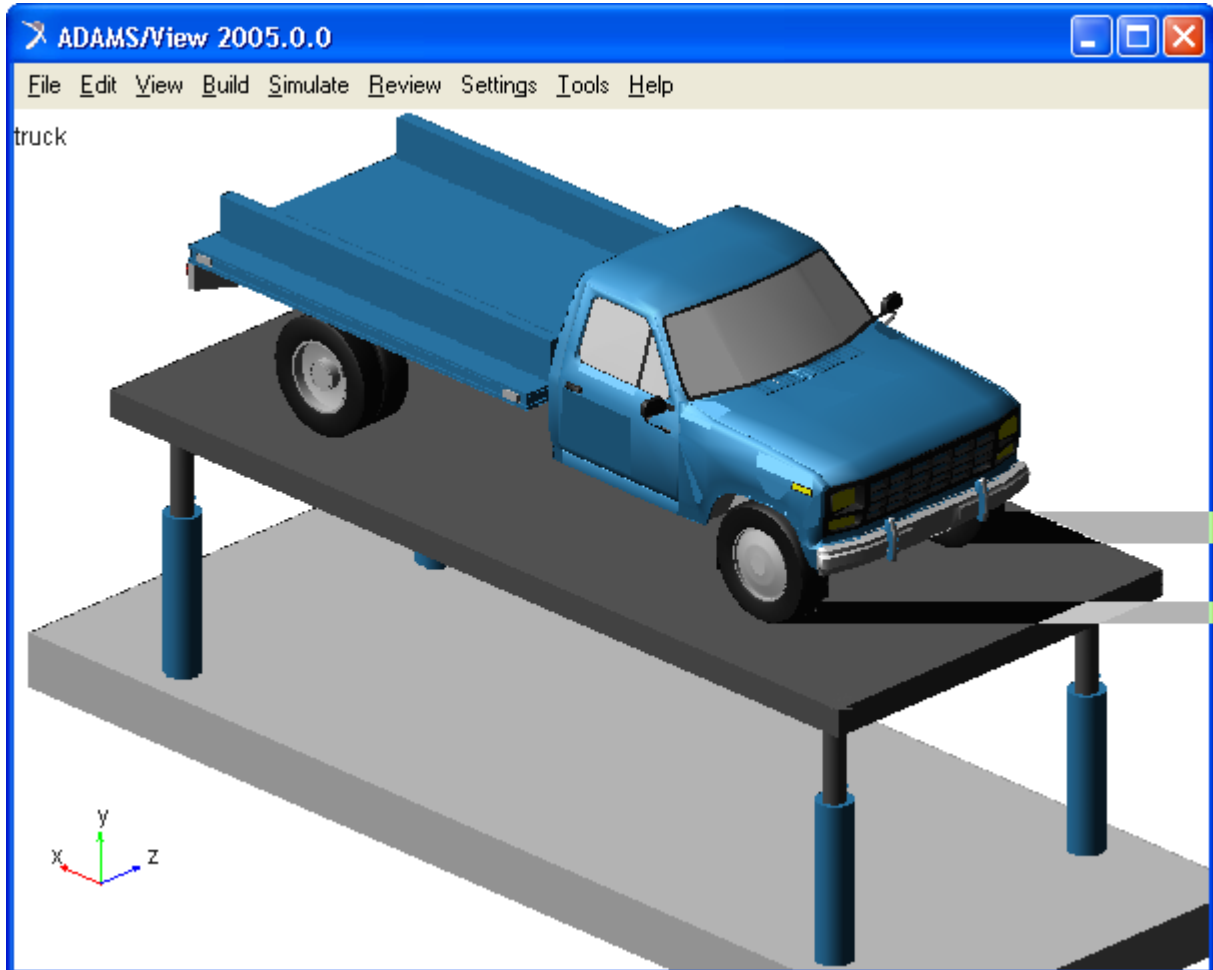Fig. 1: virtual prototyping in automotive design.

Fig. 2: virtual prototype of the light-duty truck (full-vehicle assembly).

As example, in this paper, a full-vehicle virtual prototype, which corresponds to a light-duty truck, is shown in figure 2. The prototype contains the front and rear suspension subsystems, the steering subsystem, and the car body subsystem (fig. 3). Two double-wishbone linkages are used for the front suspension. The parallel-link steering subsystem is a four-bar mechanism, with a worm steering gear. The quad-link rear axle suspension is a dependent suspension model, on which the wheels are mounted at either end of a rigid beam. The car body subsystem, which represents the base frame of the vehicle, was created by shell elements.

wheel part that is connected to the spindle mount part, which has output in the suspension system, through a revolute joint. Mount parts provide connectivity to the suspension subsystems.

The communicators used to connect the above-described subsystems are shown in table 2.

The geometric model of the vehicle was made with CAD software, the transfer to ADAMS being realized using the STEP format file. To import the geometry, ADAMS/Exchange reads the CAD file and converts the geometry into a set of ADAMS/Car geometric elements.
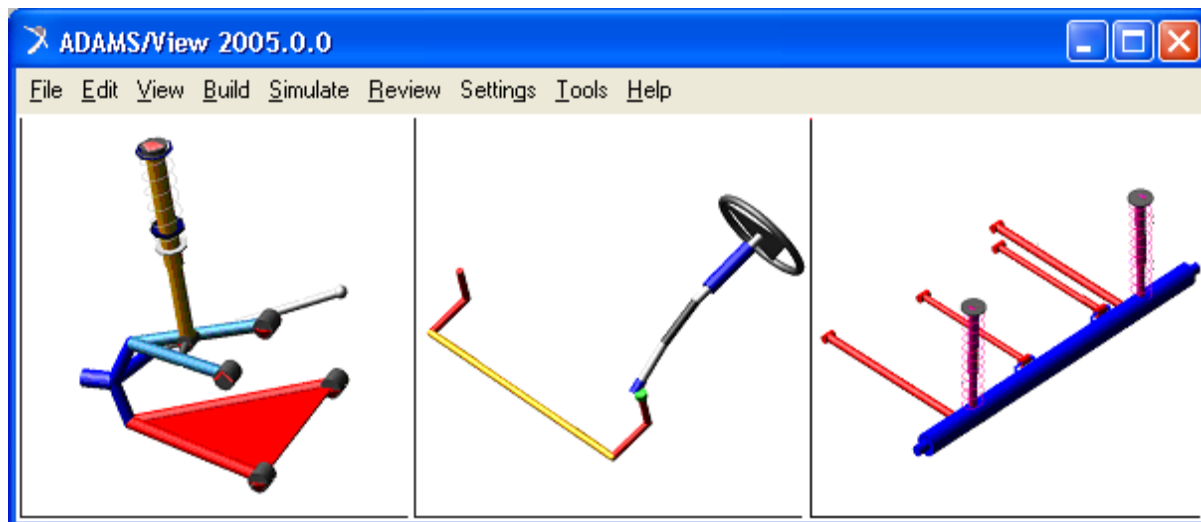
Fig. 3: front suspension, sterring and rear suspension subsystems.

Table 2: communicators between subsystems.

Communicators (l-left, r-right, s-single)			Type (Input - I / Output - O)				
			Suspension		Steering	Car body	Tire
name	front	rear	front	rear			
upper arm_body	lr	lr	I	I		O	
lower arm_body	lr	lr	I	I		O	
upper strut_body	lr	lr	I	I		O	
tierod_center link	lr		I		O		
tire_wheel carrier	lr		O				I
tire_axle		lr		O			I
steering wheel_body	s				I	O	
idler / pitman_body	lr				I	O	
input shaft_body	s				I	O	
wheel center	lr	lr	O	O			I
toe angle	lr	lr	O	O			I
camber angle	lr	lr	O	O			I

6. Results and Conclusions

The virtual prototype is analyzed in passing over bumps dynamic regime, simulating the experimental test on a hydraulic stand.

which is moved by four cylinders placed in corners. Spherical joints connect the platform with the cylinders, which execute vertical motion relative to the ground (fixed base). The motion generators, applied in the cylindrical joints, simulate the road profile. The modeling of these drivers is made in

ADAMS by Function Builder, as conditional functions of type: IF(time-0.01:50*sin(pi*time/ 0.01),50*sin(pi*time/ 0.01),0) - for the front wheels, and IF(time-0.11:0,0,-50*sin(pi*time/ 0.01))+IF(time-0.12:0,0,50*sin(pi*time/0.01)) - for the rear wheels, which correspond to the displacement over a bump of 50 mm amplitude, with the velocity of 70 km/h.

Analyzing and simulating the virtual prototype, a lot of measures that define the dynamic behavior of the truck can be obtained (in a future paper, this problem will be in detail presented). As example, in figure 4, the vertical displacements of the truck at the front & rear axle level are shown.
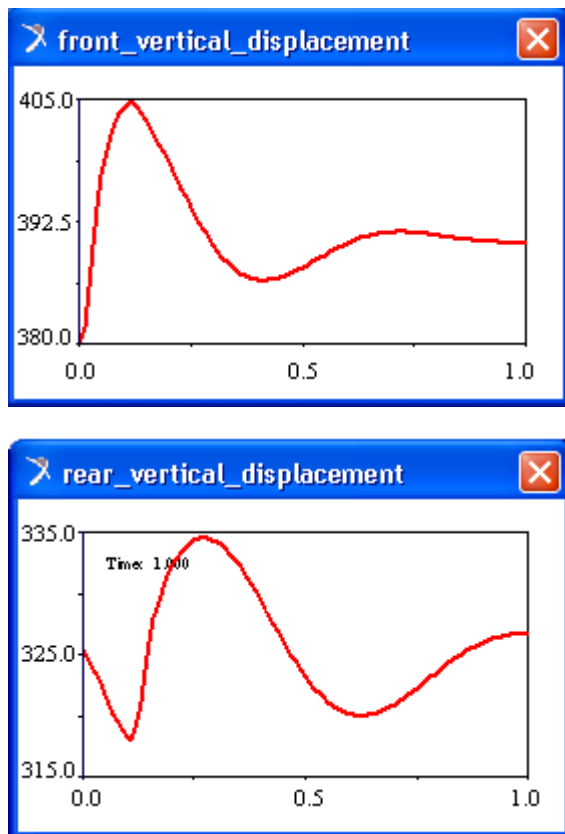

Fig. 4: vertical displacement of the truck.

References

[1] Alexandru, C., Pozna, C. Dinamica sistemelor mecanice pe baza prototipării virtuale, Editura Universităţii Transilvania, Braşov, 2003.

[2] Alexandru, C. Simularea pe calculator a sistemelor mecanice articulate, Editura Lux Libris, Braşov, 2000.

[3] Alexandru, C. The kinematic optimization of the multi-link suspension mechanisms used for the rear axle of the motor vehicles. Proceedings of the Romanian Academy, Series A , vol. 10, nr. 3, 2009, p. 244-253.

[4] Alexandru, C., Pozna, C. Dynamic modeling and control of the windshield wiper mechanisms. WSEAS Transactions on Systems, vol. 8, nr. 7, 2009, p. 825-834.

[5] Alexandru, C., Alexandru, P. The virtual prototype of a mechatronic suspension system with active force control. WSEAS Transactions on Systems, vol. 9, nr. 9, 2010, p. 927-936.

[6] Niculescu, A.I., Dumitriu, D., Sireteanu, T., Alexandru, C. The new self-adjustable shock absorber-VZN. Proceedings of the 31st FISITA World Automotive Congress, Yokohama, Japan, 2006.

[7] Haug, E. J. Computer aided kinematics and dynamics of mechanical systems, Allyn and Bacon, Massachusetts 1989.

[8] Alexandru, C., Pozna, C. Virtual prototype of a dual-axis tracking system used for photovoltaic panels. Proceedings of the 2008 IEEE International Symposium on Industrial Electronics - ISIE, 2008, p. 1598-1603.

[9] Alexandru, C., Comşiţ, M. The energy balance of the photovoltaic tracking systems using virtual prototyping platform. Proceedings of the 5-th IEEE International Conference on the European Electricity Market - EEM, 2008, Lisbon, p. 253-258.

[10] Alexandru, C. Software platform for analyzing and optimizing the mechanical systems. Proceedings of the 10th IFToMM International Symposium on Science of Mechanisms and Machines - SYROM, 2009, p. 665-677.