

WINDSHIELD WIPER MECHANISMS DESIGN IN MECHATRONIC CONCEPT

ALEXANDRU CĂTĂLIN, POZNA CLAUDIU

University “Transilvania” of Braşov, Product Design & Robotics Department

Abstract: Determining the real behavior of the windshield wiper mechanisms is a priority in the design stage since the emergence of the computer graphic simulation. The paper attempts to analyze and simulate the windshield wiper mechanisms by using a complex virtual prototyping platform, which includes the following software: CATIA (realizing the 3D solid model), ADAMS/View (analyzing and optimizing the mechanical model), and ADAMS/Controls & MATLAB/Simulink (controlling the mechanism). For applying the design strategy, we considered a tandem pattern double-lever system, which is approached in the concurrent engineering concept, by integrating the control system in the mechanical model at the virtual prototype level.

Keywords: windshield wiper mechanism, control system, dynamics, virtual prototype.

1. INTRODUCTION

The windshield wiper mechanisms are vehicle-specific systems in which the wiping motion is transferred from the wiper motor to the pivot-shaft assemblies via linkages. A compact wiper system consists of the following components: wiper motor with thermo-switch, wiper gearing, motor crank, steel base-plate, crank linkage, pivot-shaft assembly with oscillating crank, and second pivot-shaft assembly with plate (for parallel wipe pattern), respectively. The linkage forces are supported by the sheet metal of the vehicle's bodywork. For the present-day vehicles, the following wiper systems are frequently used: single-lever systems with parallel wipe patterns, single-lever systems with sector wipe patterns, opposed-pattern double-lever systems with parallel wipe patterns, opposed-pattern double-lever systems with overlapping sector wipe patterns, tandem-pattern double-lever systems with overlapping sector wipe patterns, and tandem-pattern three-lever systems with extra-wide overlapping sector wipe patterns, respectively [10].

The modern design process of the windshield wiper mechanisms involves conceptual and functional design, command & control, digital mock-up, virtual prototyping and testing [5]. The conceptual design has as main objective to establish the best product concept (in the given conditions, by performing an efficient management of information picked by the science, technology, economy, market, culture, legislation, policy etc.). The functional design involves identifying, modeling and evaluating the operational performances of the windshield wiper systems, and the deviations from the imposed characteristics, with other words the mode in which the mechanism responses to the design requirements.

In these terms, the paper attempts to analyze and simulate the windshield wiper mechanisms by using a complex virtual prototyping platform, which includes the following software: CATIA (realizing the solid model of the system), ADAMS/View (analyzing and optimizing the mechanical model), and ADAMS/Controls & MATLAB/Simulink (controlling the wiper mechanism). For applying this design strategy, we considered a tandem pattern double-lever system, which is approached the concurrent engineering concept, by integrating the control system block in the mechanical model at the virtual prototype level.

2. DESIGN PROCESS

Generally, the analysis of the windshield wiper mechanisms involves three specific mechanical models: kinematic, inverse dynamic and dynamic. The kinematic model contains the rigid parts, which are connected through geometric constraints, and the geometric parameters that define the mechanism; the input is made using kinematic restrictions, which controls the motion of the driving elements. The inverse dynamic model includes the kinematic model and, in addition, the external & internal loading; this model is used to determine the turning moment applied by the driving motor. The dynamic model includes the inverse dynamic model, but the input is made through the above-determined torque; the aim is to evaluate the real behavior.

Determining the real behavior is a priority in the design stage of the windshield wiper mechanisms. Important publications [2, 3, 7, 8] reveal a growing interest on analysis methods for multi-body systems, which may facilitate the self-formulating algorithms, having as main goal the reducing of the processing time in order to make possible real - time simulation. The modeling & analysis of the windshield wiper mechanisms by using the MBS method involves the following steps [1]: defining the wiper mechanism as multibody system; establishing the coordinate frames of the bodies; defining the geometric model of the mechanism; defining the equation system that describe the geometric and kinematic constraints from the wiper mechanism; formulating the motion's differential equations using different formalisms; defining the mass and inertia characteristics of the bodies; establishing the reaction forces and torques from the wiper mechanism; solving the algebraic and differential equation system that describes the dynamic behavior.

In the last decade, a new type of studies was defined through the utilization of the MBS environments: Virtual Prototyping. This technology consists mainly in conceiving a detailed model and using it in a virtual experiment, in a similar way with the real case. No longer is it necessary to wait months to build a hardware prototype, instrument it, run tests on it, and make a small number of expensive modifications to it in order to assess proposed design changes [6]. Basically, the virtual prototyping platform includes CAD, MBS and FEA programs [1]. The CAD environment is used to create the geometric model of the system, which contains information about the mass & inertia properties of the rigid parts. The MBS software, which represents the central component of the virtual prototyping platform, is used for analyzing, optimizing and simulating the kinematic and dynamic behavior of the mechanical system. The FEA software is used for modeling flexible components, which allows capturing inertial and compliance effects during simulation, and predict loads with greater accuracy, therefore achieving more realistic results.

The windshield wiper mechanisms are mechatronic systems, which integrate mechanics, electronics and information technology. In the typical design process of a mechanical system with controls, the mechanical designer and the controls designer work from the same concept, but use different sets of software tools. The result is that each designer produces a model for the same problem. Each design is then subject to verification and testing, and the first time the two designs are brought together is during physical prototype testing. If a problem occurs during the interaction between the controls design and the mechanical design, the engineers must refine the control design and/or the mechanical design, and then go through the entire verification process. Integrating the control system in the mechanical model, the two designers can share the same model; they can also verify from one database the combined effects of a control system on a nonlinear, non-rigid model. The physical testing process is greatly simplified, and the risk of the control law being poorly matched to the real system is eliminated [4].

In these terms, the virtual prototyping platform has to include a control software product, which directly exchanges information with the MBS software. This simulation process creates a closed loop in which the control inputs from the control application affect the MBS simulation, and the MBS outputs affect the control input levels. In the concurrent engineering philosophy, the simulation algorithm of the windshield wiper mechanisms involves the following steps [9]: regarding the MBS software: designing – realizing the mechanical model (including bodies, joints, elastic elements, forces); analyzing the dynamic model; identifying the inputs and outputs, which complete a closed loop between the mechanic model and the control system; exporting the model; regarding the control software: importing the mechanical model; explaining the trajectory and synthesizing the reference signals; designing the control system block diagram; designing the controller and the interface electric circuits; simulating the mechatronic system.

3. CASE STUDY

In order to apply the above-described algorithm, a tandem pattern double-lever wiper system (corresponding to a domestic passenger car) has been considered. The mechanism contains two four-bar spatial linkages (fig. 1): ABCD - command mechanism, from the motor crank to the left wiper arm & lamella, and DC'FE - connection mechanism, which transmits the revolute motion to the right arm. The motor crank (1) and the left & right wiper arms (3, 5) are connected to ground / car body (0) using revolute joints A, D, and E. The crank linkages (2, 4) are connected to the motor crank, respectively to the wiper arms, using spherical joints B, F and cylindrical joints C, C', respectively. The mechanism has one degree of mobility, according to the Grubler count [2]:

$$\text{DOM} = 6 \cdot n - \Sigma r_g = 6 \cdot 5 - 29 = 1, \quad (1)$$

where “n” represents the mobile parts (bodies) and “ Σr_g ” is the number of geometric constraints.

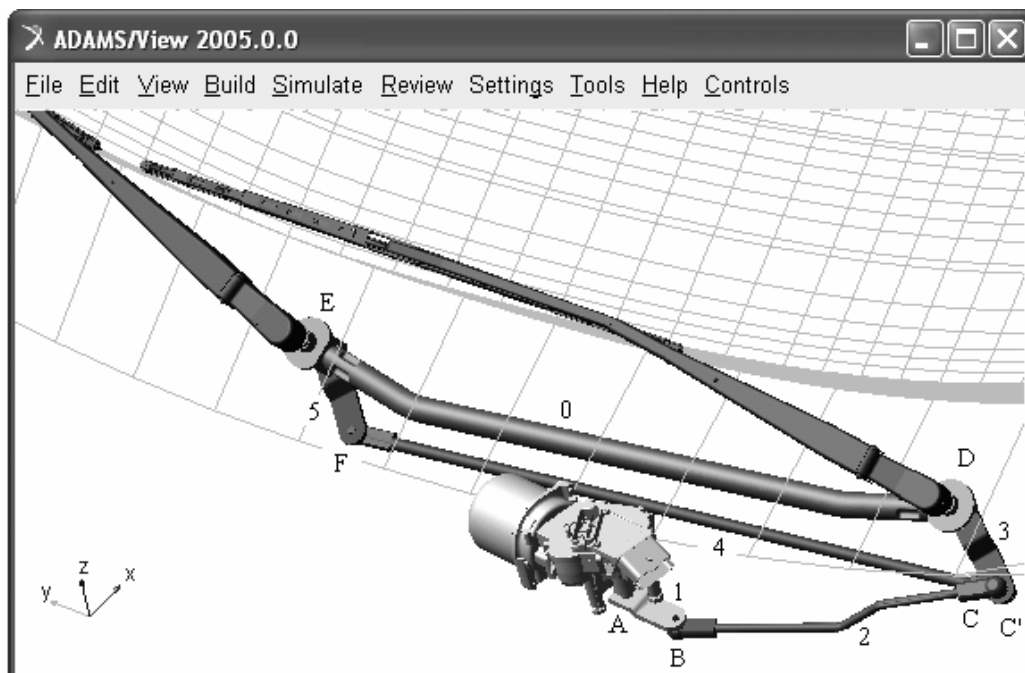


Fig. 1

The kinematic and inverse dynamic models are driven using a motion generator, which controls the angular velocity of the motor crank, $\omega_1(t)$. For the dynamic model, the kinematic constraint is replaced by the motor torque applied to the input crank. Driving the mechanism by using the motor torque is more realistic than attaching a motion generator to the input crank and driving the motion directly. By applying a torque, we can look at issues related to motor size in an actual mechanical system. The dynamic model includes also the friction forces that act on the flexible lamellas, which depend on the friction coefficient between rubber and windshield, and the normal force generated by the spring mounted between the wiper arm and the oscillating crank. The direction of the friction force depends on the sign of wiper arm's velocity, which is modeled in ADAMS by the SIGN function (using the specific Function Builder module) [9]. This function transfers the sign of an expression representing a numerical value to the magnitude of another expression representing a numerical value, as follows: $\text{SIGN}(a1, a2) = \text{ABS}(a1)$ if $a2 \geq 0$, $\text{SIGN}(a1, a2) = -\text{ABS}(a1)$ if $a2 < 0$. In our case, “a1” represents the friction force's magnitude, and “a2” is the angular velocity of the wiper arm. In these terms, there is the following expression for the left/right friction force: $\text{SIGN}(\text{value}, \text{model.left/right_wiper_arm_angular_velocity})$, where „value” is the design variable that define force magnitude.

For determining the torque that move the motor crank, a control system is developed using ADAMS/Controls and MATLAB/Simulink. ADAMS/Control is a plug-in to ADAMS/View that allows integrating motion

simulation and controlling system design in the virtual model. ADAMS/Controls allow connecting the ADAMS model to block diagrams that are developed with the control application. The four-step process of combining controls with a mechanical system involves: build the virtual mechanical model, which includes all necessary geometry, constraints, forces; identify the ADAMS inputs and outputs - the outputs describe the variables that go to the controls application, and the inputs describe the variables that come back into ADAMS; build the block diagram - build the control system block diagram with a control software, and include the ADAMS plant in the block diagram; simulate the combined mechanical model and control system.

To drive the mechanism, a DC electric motor is used, so that the objective of the controlling process is to control the angular velocity of the motor crank, which is perturbed with the crank turning moment. In these conditions, the input to the mechanical model is the angular velocity of the motor crank, and the output, which will be transmitted to the controller, is the crank turning moment. ADAMS/Controls and MATLAB/Simulink communicate by passing state variables back and forth. Therefore, it is necessary to define the model's input and output variables, and the functions that those inputs and outputs reference, with a set of ADAMS state variables. For the input state variable, representing the angular velocity of the motor crank, the run-time function is 0.0 during each step of the simulation, because the velocity will get its value from the control application. The run-time function for the input variable is VARVAL (angular velocity), where VARVAL (Variable Value) is an ADAMS function that returns the value of the given variable. For the output state variable, representing the crank turning moment, the run-time function returns the sum of torques on motor crank at location (a marker placed in the joint between motor crank and chassis).

The next step is for exporting the ADAMS plant files for the control application. The Plant Input refers the input state variable (angular velocity), and the Plant Output refers the output state variable (control torque). ADAMS/Controls save the input and output information in an "*.m" file (specific for MATLAB); it also generates a command file (*.cmd) and a dataset file (*.adm), which will be used during the simulation process. With these files, the control system block is created in MATLAB/Simulink, in order to complete the link between the controls and mechanical systems. ADAMS/Controls and MATLAB communicate by passing state variables back and forth. ADAMS accepts the control inputs from MATLAB and integrates the mechanical model in response to them. At the same time, ADAMS provides the control torque information for MATLAB to integrate the Simulink model.

The mechatronic system is composed from the following components: the actuator, which in this case is a controlled DC motor, the gear box, the wiper mechanism, the measurement system (sensor), and the controller. In this case, the plant can be defined in several ways: the subsystem composed from the actuator, the gear box and the windshield wiper; the actuator and the gear box. For solving the problem (designing the controller), we have considered the second case, from the following reasons: the plant is a linear system for which is easy to design the control system; the gear box ratio is big and will decrease the nonlinearity of the perturbations. The dynamic model of the motor, which is an electro-mechanical system (fig. 2), is given by the following relations:

$$V = R_a i + L \frac{di}{dt} + E, J \ddot{q}_m = K_M i - b \dot{q}_m - T, E = K_b \dot{q}_m, \quad (2)$$

where: V - the input voltage, T - the torque, R_a - the resistance, L - the inductance, i - the current, K_b - the contra-electromotor constant, q_m - the rotor position, $\dot{q}_m$ - the angular velocity of the rotor, J - the inertial moment, K_M - the motor torque coefficient, b - the viscous friction.

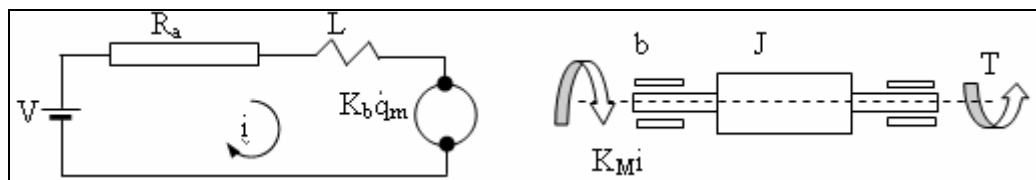


Fig. 2

The structure of the dynamic system model is shown in figure 3, in which there are mentioned the interactions parameters between the three subsystems: DC actuator with gear box, wiper mechanism, and the interaction with

outside; the controlled parameter of the DC motor is the angular velocity (ω). According with this diagram, we will control only the DC motor, which is perturbed with the torque T . The computation of T is based on the dynamic model of the mechanism on which acts outside torques. For detailing the previous diagram, we have transformed the motor's equations (1) with the Laplace operator, as follows:

$$\Omega_m(s) = G_V(s) \cdot V(s) - G_T(s) \cdot T(s), \quad G_V(s) = \frac{K_M}{LJs^2 + (JRa + bL)s + (bRa + K_b K_M)}, \quad (3)$$

$$G_T(s) = \frac{Ls + Ra}{LJs^2 + (JRa + bL)s + (bRa + K_b K_M)},$$

where: $\Omega_m(s)$ - the Laplace transform of ω_m , $V(s)$ - the Laplace transform of V , $T(s)$ - the Laplace transform of T ; $G_V(s)$ and $G_T(s)$ - the transfer function.

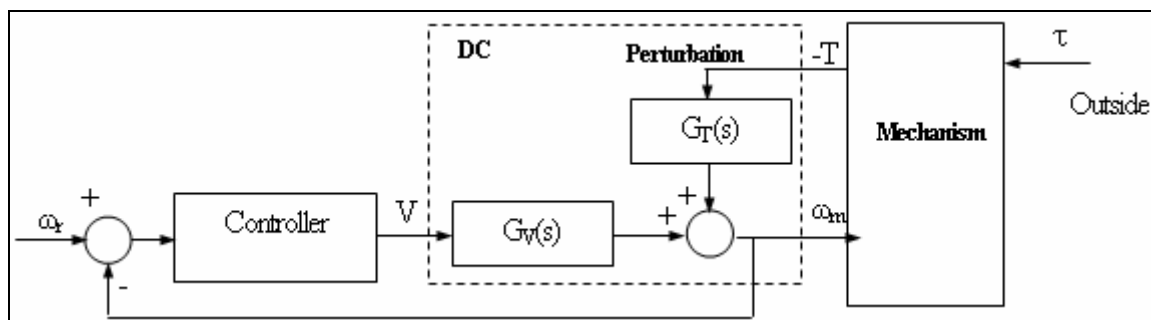


Fig. 3

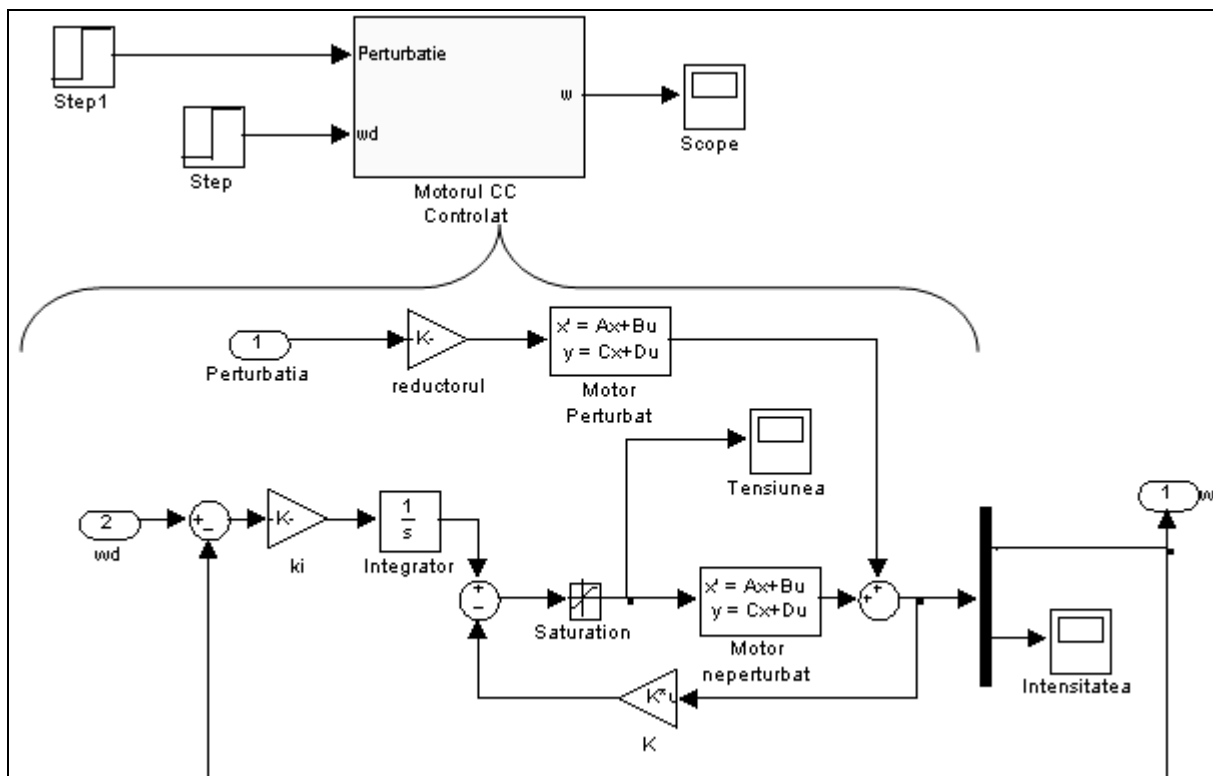


Fig. 4

The transfer function of the plant is G_V , which has not poles in the origin, and for this reason the steady state error for a step input will be different from zero if we use a proportional controller. In order to obtain a zero error

steady state is recommended to use a proportional integer controller. For these conditions, the simulation diagram was realized in MATLAB/Simulink (fig. 4). With this control system block and the MBS model of the double-lever mechanism shown in figure 1, a lot of results have been obtained in order to evaluate the dynamic behavior of the windshield wiper system. For example, in figure 5 there are presented the time - history variations of the angular velocity of the motor crank, and the crank turning moment. The results correspond for two complete rotations of the motor crank ($2 \times 360^\circ$). The control system generates the angular velocity of the motor crank, which, in the regime phase, has the value $\omega \approx 360$ [deg/sec].

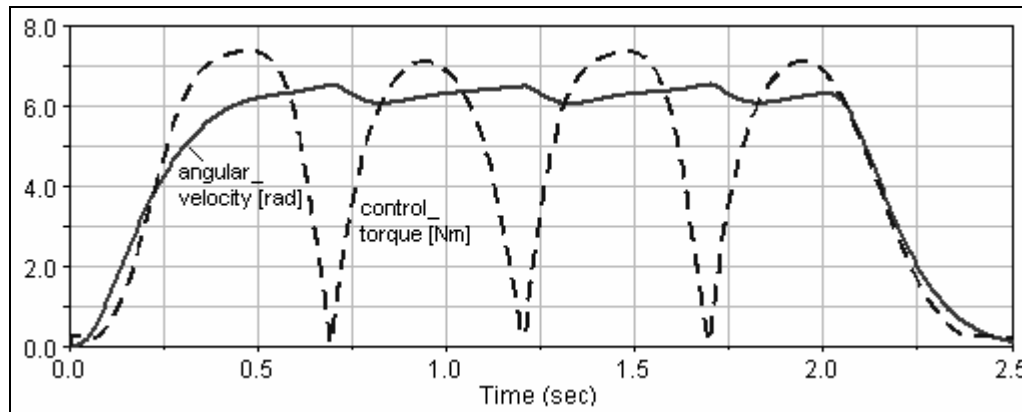


Fig. 5

4. FINAL CONCLUSIONS

The application is a typical example of virtual prototyping of the windshield wiper systems. One of the most important advantages of this kind of simulation is the possibility to perform virtual measurements in any point/area and for any parameter (motion, force). This is not always possible in the real cases due to the lack of space for transducers placement or lack of appropriate transducers. This helps us to make quick decisions on any design changes without going through expensive prototype building & testing. With virtual prototyping, behavioral performance predictions are obtained much earlier in the design cycle, thereby allowing more effective and cost efficient design changes and reducing overall risk substantially. Concluding, the virtual prototyping brings several advantages: reduce the time and cost of new product development, reduce the product cycles, reduce the number of expensive physical prototypes and experiment with more design alternatives, increase quality and efficiency, and finally improve product.

REFERENCES

- [1] Alexandru, C., Pozna, C., *Dynamics of Mechanical Systems using Virtual Prototyping Tools*, Transilvania University Publisher, Braşov, 2003.
- [2] Haug, E.J., *Computer Aided Kinematics and Dynamics of Mechanical Systems*, Allyn and Bacon, 1989.
- [3] Haug, E. J., *Virtual Prototyping Simulation for Design of Mechanical Systems*, Transaction ASME, no. 117, p. 63-70, 1995.
- [4] Mătieş, V., Mândru, D., *Mechatronic Technology and Education*, Todesco, Cluj Napoca, 2001.
- [5] Pahl, G., Beitz, W., *Engineering Design*, Springer-Verlag, 1996.
- [6] Ryan, R., *Functional Virtual Prototyping*, Mechanical Dynamics Inc., 2001.
- [7] Schiehlen, W.O., *Advanced Multibody Systems Dynamics*, Kluwer Academic, 1993.
- [8] Shabana, A., *Dynamics of Multibody Systems*, John Wiley & Sons, 1988.
- [9] *** *Getting Started using Adams/View & Controls*, MSC Software, 2005.
- [10] *** *Windshield Wiper Systems for Commercial Vehicles*, Robert Bosch Publisher, 2001.