FLEXIBLE BODY INTEGRATION AND CO-SIMULATION OF DOUBLE WISHBONE SUSPENSION SYSTEM

HAREESHA[1], PRASHANTHA.B.Y[2], BHARATH.M.S[3]

[1] PG Student, Department of Mechanical Engineering, BIET, Karnataka, India
[2] Assistant Professor, Mechanical Department, SSE, Mukka,Mangalore, Karnataka, India
[3] Assistant Professor, Mechanical Department, GMIT Mandya, Karnataka, India

---***---

Abstract - *The main aim of this paper is to how quickly set up the virtual prototyping of Double Wishbone Suspension and Steering Model System to check the dynamic behavior of system for a given input signal at the center of the wheel as vertical ground load to measure Toe angle and wheel height. For this Double Wishbone Suspension System as an example, a Co-simulation control method and Flexible Body integration method is introduced to research multi - body dynamics. Using Kinematic and Newton - Euler and Lagrange method used to establish the Dynamics Model of Double Wishbone Suspension. The simulation results indicate that the Double Wishbone Suspension and Steering Model System have preferable response characteristics. The co - simulation method is more effective. To establish the Dynamic behavior of the Tie Rod of Double Wishbone Suspension system in terms of Mode shapes.*

Key Words: *Multi-body Dynamics, Double wish bone suspension system, MSC ADAMS, NASTRAN-PATRAN, PID Controller, MATLAB/SIMULINK Co-Simulation.*

1. INTRODUCTION

In automobiles, Suspension system consisting of springs shock absorbers and linkages that connect the vehicle to the wheels and allows relative motion between the wheels and the vehicle body. Double Wishbone suspension is an independent suspension design using two wishbone shaped arms to locate the wheel shows in Figure 1. Each wish bone or arm has two mounting points to the chassis and one joint at the knuckle. The shock absorber and coil spring mount to the wishbones to control vertical movement.

Double wishbone designs allow the engineer to carefully control the motion of the wheel throughout the suspension travel, controlling such parameters as camber angle, caster angle, toe pattern, roll center height, scrub radius. The steering arm is attached to the wheel. Also, the most important role played by the suspension system is to keep the wheels in contact with the road all the time one of the functions of the suspension system is to maintain the wheels in proper steer and camber attitudes to the road surface.
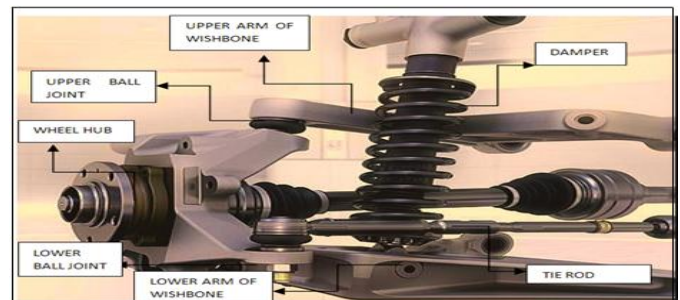

Fig-1: Double Wishbone Suspension System

It should react to the various forces that act in Dynamic condition. These forces include longitudinal (acceleration and braking) forces, lateral forces (cornering forces) and braking and driving torques. All the dynamic parameters are considered while designing the suspension system, especially the behavior of the suspension for various loading cases.

In this paper, a dynamical model of the Double wishbone suspension system with steering model is built up in ADAMS/view shows in Figure 2, the PID control model was built in MATLAB/SIMULINK, and Tie Rod of Double wish bone suspension was imported from NASTRAN-PATRAN in the form of Flexible body file (MNF) for flexible body integration to carry Dynamic Analysis. The Double wishbone suspension system model and the control model are integrated through ADAMS/Control Module, and the co-simulation carried out for the given vertical loads at the wheel center and a measure Toe angle and Wheel Height for an applied force input signal from MATLAB/SIMULINK.
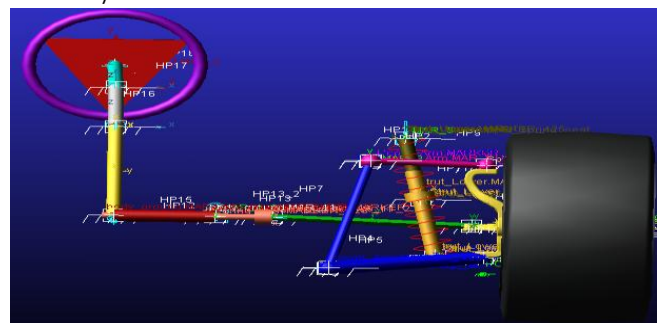

Fig-2: Double Wishbone Suspension Model in ADAMS

2. KINAMATICS AND DYNAMIC MODEL OF DOUBLE WISHBONE SUSPENSION

The topological graph of double-wishbone suspension describes in Figure 3. The ability to adjust many aspects of its kinematics makes the double-wishbone suspension popular on high-performance vehicles; its load-handling capabilities also make it suitable for use on the front axle of medium and heavy vehicles. The upper and lower control arms are connected to the wheel carrier with spherical joints (S), and to the chassis with revolute joints (R). One end of the tie rod is connected to the wheel carrier with a spherical joint; a universal joint (U) at the other end connects it to either the rack (on the front axle) or the chassis (on the rear axle). The system is modeled using joint coordinates as labeled in Fig.3.

$$q = {}^{w}\theta, \zeta, \eta, \xi, {}^{u}\theta, \grave{}\theta, \alpha, \beta, s \qquad [1]$$
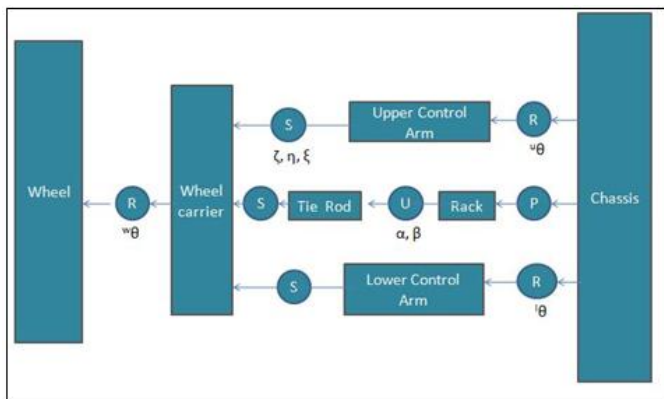

Fig-3: Double Wishbone Suspension Topological Graph

The configuration of the uni-versal joint is specified by its angles of rotation about the global Z-axis (α) and the rotated X^0-axis (β); $\{\zeta, \eta, \xi\}$ represents the 3-2-1 Euler angles associated with the spherical joint between the upper control arm and the wheel carrier. Neglecting the motion of the chassis, this system has three degrees-of-freedom: the rotation of the wheel (${}^{w}\theta$), the displacement of the rack (s), and the vertical displacement of the wheel carrier (z). Note that a rack displacement is an input to the model, which reduces the number of degrees-of-freedom to two.

We proceed with a purely joint coordinate formulation to minimize the number of modeling coordinates. The Multibody library used to generate the following six constraint equations, three for each independent kinematic loop:

$$\varphi_{1,2,3} (\zeta, \eta, \xi, {}^{u}\theta, \grave{}\theta)=0, \quad \varphi_{4,5,6} (\zeta, \eta, \xi, {}^{u}\theta, \alpha, \beta, s)=0 \qquad [2]$$

Drawing from the results of the preceding examples, we employ the following strategy:
1. Assign one of $\{\zeta, \eta, \xi\}$ to the independent coordinate vector q_i.
2. Compute the two remaining spherical joint angles iteratively using two of the six constraint equations (2).

3. Compute the four remaining generalized coordinates ${}^{u}\theta$, $\grave{}\theta$, α, β recursively by tri-angularizing the remaining constraint equations.

The first objective is to derive a triangular system to solve for q_{d1} = ($ {}^{u}\theta$, ${}^{l}\theta$, α, β) recursively, given values of $\{\xi, \zeta, \eta\}$. We begin by expressing φ_1 in the following form:

$$A_1 \sin \grave{}\theta + B_1 \cos \grave{}\theta = C_1 \qquad [3]$$

Where: co-efficients A_1, B_1, and C_1 are functions of $\{\xi, \zeta, \eta, {}^{u}\theta\}$. The following solutions can then be obtained for $\sin({}^{l}\theta)$ and $\cos({}^{l}\theta)$

$$\sin(l_\theta) = \frac{A_1 C_1 \pm B_1 \sqrt{A_1^2 + B_1^2 - C_1^2}}{A_1^2 + B_1^2} \qquad [4]$$

$$\cos(l_\theta) = \frac{A_1 C_1 \pm B_1 \sqrt{A_1^2 + B_1^2 - C_1^2}}{A_1^2 + B_1^2} \qquad [5]$$

only the first three of which are assumed to be known at this stage of the solution. We obtain an expression for ${}^{u}\theta$ as a function of $\{\xi, \zeta, \eta\}$ by substituting (4) and (5) into φ_2 which, once simplified, can be expressed in the following form:

$$A_2\sin({}^{u}\theta)+B_2\cos({}^{u}\theta)+D_2\sin({}^{u}\theta)\cos({}^{u}\theta)+E_2\cos^2({}^{u}\theta)=C_2 \qquad [6]$$

Where: Co-efficients are functions of $\{\xi, \zeta, \eta\}$. Note that the second-order trigonometric terms appear upon eliminating the square roots introduced by (4) and (5). Although (6) can be solved for $\sin({}^{u}\theta)$ and $\cos({}^{u}\theta)$, the solution is quartic and would require iteration to evaluate numerically. Instead, we note that the coefficients in D_2 and E_2 are significantly smaller than those appearing elsewhere, so expect these terms to have negligible effect on the numerical solution. Thus, we use the approximations $D_2\approx0$ and $E_2\approx0$ to obtain an equation analogous to (3), which results in solutions for ${}^{u}\theta$ of the form shown in (4) and (5). Evaluating numerically over the full range of motion of the wheel carrier, we find that this approximation introduces an error of less than 5.7×10^{-4} [rad] (0.1%) in the solution for ${}^{u}\theta$, which is deemed to be an acceptable compromise given our real-time intentions. A similar strategy can be employed to triangularize φ_4 and φ_5. In particular, φ_4 can be expressed as follows:

$$A_4\sin(\beta) +B_4\cos(\beta) =C_4 \qquad [7]$$

Where: A_4, B_4, and C_4 are functions of $\{\xi, \zeta, \eta, {}^{u}\theta, \alpha\}$. Solutions for $\sin(\beta)$ and $\cos(\beta)$ can then be obtained as before. Upon substitution of these solutions into φ_5, we obtain an equation of the following form:

$$A_5\sin(\alpha)+B_5\cos(\alpha)+D_5\sin(\alpha)\cos(\alpha)+ E_5\cos^2(\alpha)= C_5 \quad (5.16) \qquad [8]$$

Where: The solutions of the form shown in (4) and (5) are obtained. As will be verified below, the error introduced by this approximation—less than 4.5×10^{-4} [rad] (1.5%) in the solution for α—does not significantly degrade the quality of the results. In

summary, we have generated the following triangular system to solve for q_{d1} = $^u\theta$, `θ, α, β recursively, given values of {ξ, ζ, η}:

$^u\theta$ = f (ξ, ζ, η)   [9]

`θ = f (ξ, ζ, η, $^u\theta$)   [10]

α = f (ξ, ζ, η, $^u\theta$)   [11]

β = f (ξ, ζ, η, $^u\theta$, α)   [12]

Fig-4: Kinematic solution flow for double-wishbone suspension

Kinematic simulations are performed by fixing the chassis to the ground and applying motion drivers to the independent coordinate ξ and rack displacement s, as shown in Figure 4. The input applied to spherical joint angle ξ drives the wheel carrier through its full range of vertical motion, as shown in Figure 4, and compare the computational efficiency of three solution approaches:

1. Computing q_{d1} and q_{d2} using the block-triangular solution shown in Figure 4.
2. Computing q_d using Newton's method, iterating over the original constraint equations (1) and using Gaussian elimination with partial pivoting to invert the 6 × 6 Jacobian matrix numerically.
3. Computing q_d using Newton's method, iterating over the original constraint equations (1) and using a symbolic Jacobian inverse.

2.2 LAGRANGIAN DYNAMICS FOR A MULTIBODY SYSTEM

Newton-Euler equations are expressed in generalized coordinates; multibody dynamics is a straightforward extension Of a single rigid body the kinetic and strain energies are

Kinetic Energy $T = \frac{1}{2}\{\dot{q}\}^T[M]\{\dot{q}\}$ [13]

Strain Energy $U = \frac{1}{2}\{\{q\}^T[M]\{q\}$ [14]

The Lagrange's equations;

$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q_t}}\right) + \left(\frac{\partial U}{\partial q_t}\right) = Q$ [15]

2.1 Toe Angle

Toe angle is the measure of how far inward or outward the leading edge of the tire is facing, when viewed from the top. Toe is measured in degrees and is generally a fraction of a whole degree. It has a large effect on how the car reacts to steering inputs as well as on tire wear. Aggressive toe angle will cause the tire to develop "feathering" across its surface. Toe-in is when the leading part of the tire is turned inwards towards the center of the car shown in Figure 5. This makes the tires want to push inward, which acts to improve straight line stability of the car as it's traveling down the road; particularly at high speed (highway).Toe-out is when the leading part of the tire is turned outwards away from the center of the car. This makes the tires want to separate from each other.
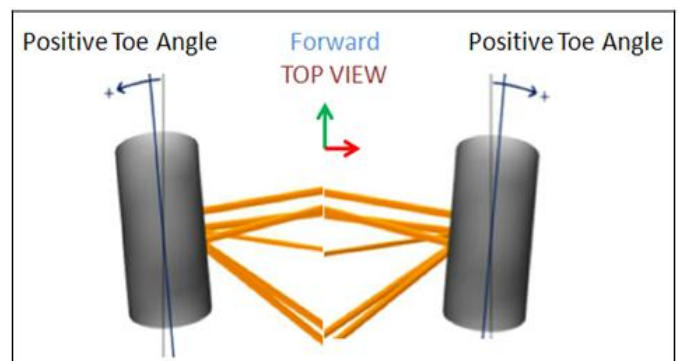

Fig-5: Toe Angle

3. CO-SIMULATION BETWEEN ADAMS AND MATLAB/SIMULINK

Co-simulation between ADAMS and MATLAB /SIMULINK means that build multi-body system in ADAMS, output parameters related to system equation and then import information from ADAMS into MATLAB/SIMULINK and set up control scheme in ADAMS/CONTROL Module. During calculation process, there exchanges data between Virtual Prototype and Control Program, where, ADAMS solves the Mechanical System Equation and MATLAB solves the Control System Equation. They both complete the whole control process. The flow chart of co-simulation described in Figure 5. In this case the vehicle is modeled on ADAMS-View shown in Figure 10, whereas the PID Controller was modeled in SIMULINK. The model built in ADAMS, as a sub-system, need to be imported into MATLAB/SIMULINK, on which SIMULINK constructs the co-simulation system shows in Figure 7. Initially exchange data between ADAMS and MATLAB/SIMULINK through ADAMS/CONTROL interface and Fig.6 describes the inside of the block Adams _sub. And Figure 8, 9 show such dynamic response for a given input signal in MATLAB/SIMULINK. Figure 11, 12 shows dynamic response in ADAMS.
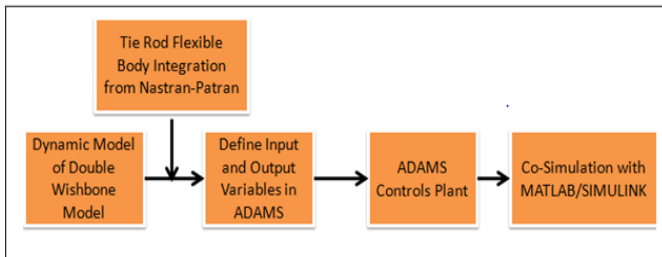
Fig-5: Co-Simulation and Flexible body integration flow chart of ADAMS, MATLAB/SIMULINK, and NASTRAN-PATRAN
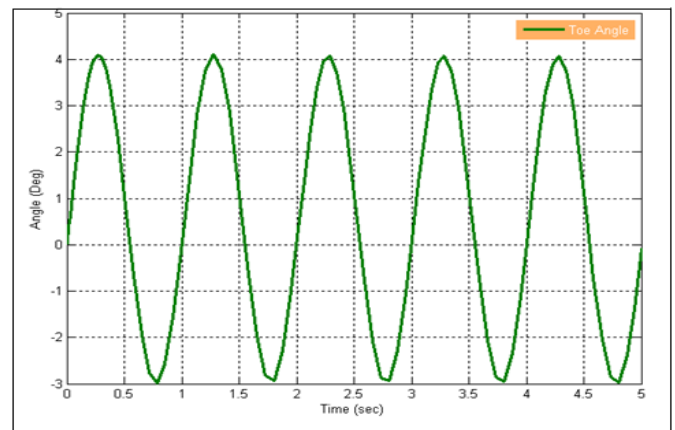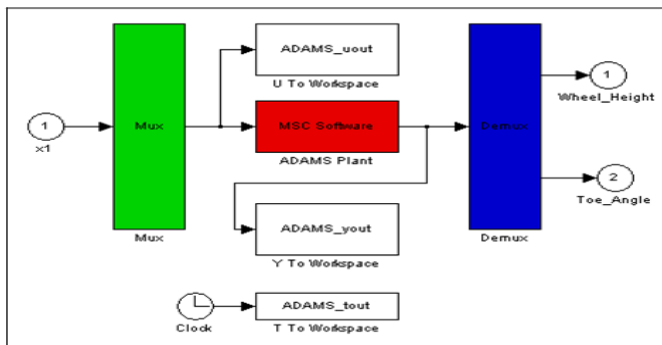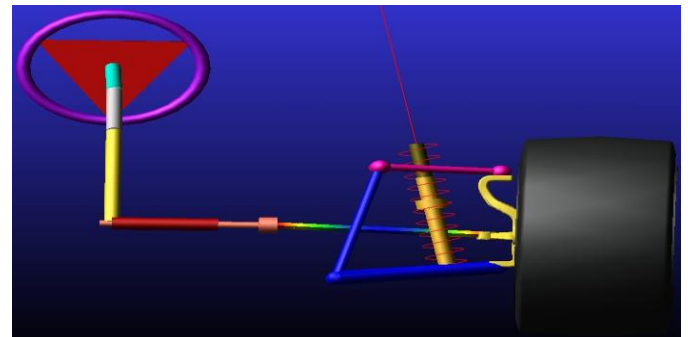

Fig-6: ADAMS Sub System Module

Fig-7: Control Scheme of the Co-simulation

Fig-8: Response of Sine signal on Wheel Centre

Fig-9: Response of Toe Angle

Fig-10: Double Wishbone Suspension Model in ADAMS

Fig-11: Applied Input Signal ADAMS (Wheel Height)

Fig-12: Toe Angle Result from ADAMS

4. FLEXIBLE BODY INTEGRATION

The View Flex module in Adams/View enables users to transform a rigid part to an MNF-based flexible body using embedded finite element analysis where a meshing step and linear modes analysis will be performed in MSC NASTRAN-PATRAN of Tie Rod are shows in Figure 13, allowing one to create flexible bodies without leaving Adams/View and without reliance on Finite Element Analysis software. Also, it's a streamlined process with much higher efficiency than the way users have traditionally generated flexible bodies for Adams in the past also convenient control over modal participation and damping.
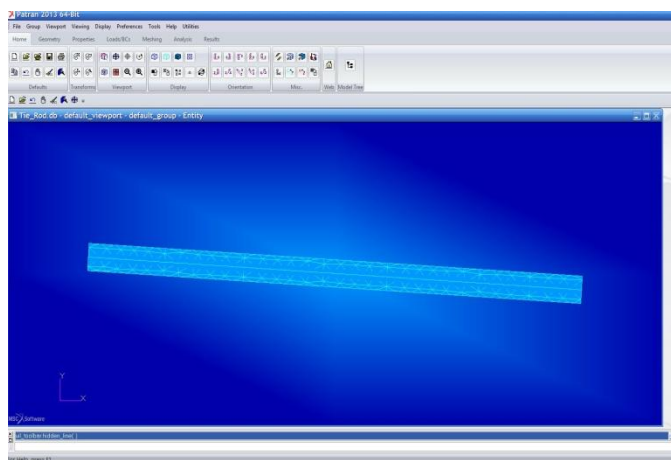

Fig-13: MNF Format Tie Rod in PATRAN

4.1 Dynamic Analysis of Tie Rod

Modal analysis is usually used to determine the Natural Frequencies and Mode shapes of a component. It can be used as the starting point for Dynamic Analysis. The Finite Element Analysis codes usually used several mode extraction methods. The number of modes was extracted and used to obtain the Tie rod stress histories, which is the most important factor in analysis.

4.2 The Equation of Motions of the Suspension System

Based on a Kinetic model of a vibration system, a mathematical model can be developed to accomplish the modal analysis. It is critical for modal analysis to develop the mathematical model. Accuracy degree of a model should affect analysis result directly. A general kinetic model is

$$[M]\{\ddot{x}(t)\}+[C]\{\dot{x}(t)\}+[K]\{x(t)\}= \{F(t)\} \qquad [16]$$

Where, [M] =Total mass matrix,
      [C] = Equivalent damping matrix,
      [K] = Stiffness matrix,
      X (t) = Displacement,
      F (t) = External load.

A model analysis equation when an external exciting force is zero [F (t) =0].

$$[M]\{x(t)\}+[C]\{x(t)\}+[K]\{x(t)\}= 0 \qquad [17]$$

Equation (17) shows that vibration amplitude of a Tie Rod should be attenuated during process of vibration.

4.3 Mode Shapes

Using this method to obtain the 8 modes of the Tie Rod, which are shown in Table.1 and the corresponding shape of the mode, are shows in Figure 14-21.
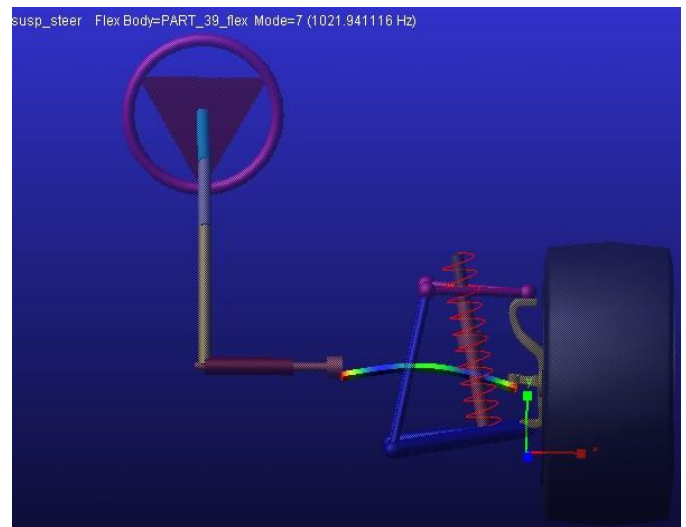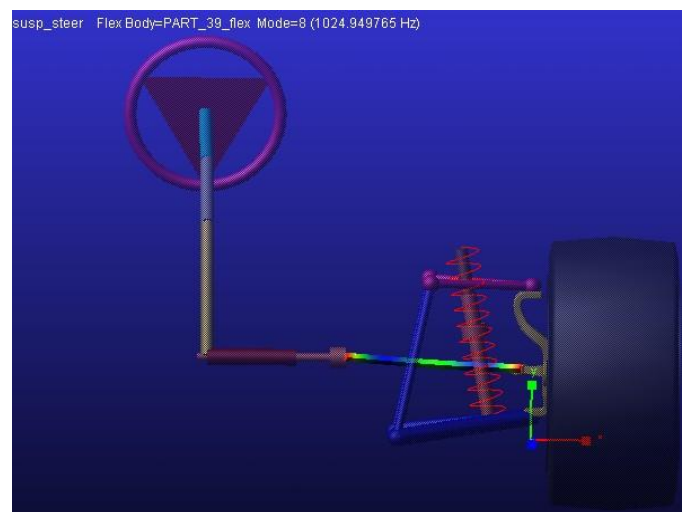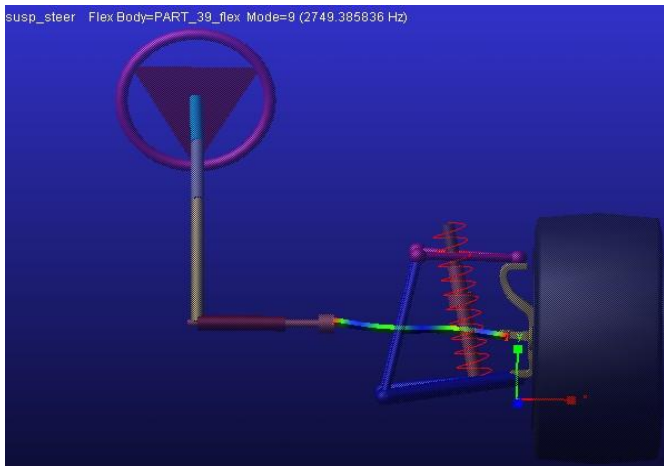

Fig-14: Mode-7

Fig-15: Mode-8

Fig-16: Mode-9

Fig-17: Mode-10

Fig-18: Mode-11

Fig-19: Mode-12

Fig-20: Mode-13

Fig-21: Mode-14

Table -1: Natural Frequency of Tie Rod

Mode No.	Frequency in Hz
7	1021.9
8	1024.9
9	2749.3
10	2814.0
11	5203.6
12	5305.6
13	7073.3
14	8333.3

5. CONCLUSIONS

In this study, build co-simulation model of Double Wishbone Suspension and Steering Model System respectively in ADAMS and MATLAB, design the interface between mechanical system and control system and get the response performance of Double Wishbone Suspension and Steering Model System. The system has good tracking ability and co-simulation result verifies that. Also, its dynamic performance meets the design requirement. What's more, modeling in ADAMS, it is beneficial to avoid solving dynamic equations and to observe motion process visualized. Using MATLAB /SIMULINK toolbox, we operate the whole process simply and program efficiently and quickly. And we get lots of design parameters from co-simulation, which is useful for subsequent research. Co-simulated method takes advantages between ADAMS and MATLAB/SIMULINK software's, with Enhancing Dynamic Performance of Double Wishbone Suspension and Steering Model System, improving efficiency, and saving Time. For the complex control system, it is a good solution to solve the problem. The Results of the Frequency from modes shapes are used for Re-design process.

REFERENCES

[1] Mike Blundell and Damaion Harty, Multibody System Approaches to Vehicle Dynamics, 2004.

[2] Gu, M.Y., R.R. Qin and D.Y. Yang, 2005. Co-simulated Research method of robot arm dynamics in ADAMS and MATLAB. Mach. Des., 1: 227-228.

[3] Li, S.Q. and H. Le, 2011. Co-simulation study of vehicle ESP system based on ADAMS and MATLAB. J. Softw., 6(5): 866-950. Li, H.,

[4] B.H. Fan and Q. Liu, 2007a. Research on co-simulation based on ADAMS and MATLAB. Sci. Technol. Inform., 8: 152.

[5] M. Arnold, B. Burgermeister, and A. Eichberger. Linearly implicit time integration methods in real-time applications: DAEs and stiff ODEs. Multibody System Dynamics, 17(2–3):99–117, 2007.

[6] Li, S.H., S.P. Yang and H.Y. Li, 2007b. Semi-active control co-simulation of automotive suspension based on ADAMS and MATLAB. J. Syst. Simul., 19(10): 2304-2307.

[7] Liu, G.J., M. Wang and B. He, 2009. Co-simulation of autonomous underwater vehicle based on ADAMS and MATLAB/SIMULINK. Chinese J. Mech. Eng., 45(10): 22-29.

[8] Zouhaier Affi., ADAMS/Simulink interface for Dynamic Modeling and Control of Closed Loop Mechanism.Tunisia.

[9] MSC ADAMS Flexible Body Integration Tutorials.

[10] Lu You-Fang. Flexible multi-body dynamics[M].Beijing: Higher Education Press, 1996