Chapter 8
A Framework for Developing Efficient Vehicle-Bridge Interaction Models Within a Commercial Finite Element Software

Omar R. Abuodeh and Laura Redmond

Abstract The development of vehicle-bridge interaction (VBI) models is a popular technique to characterize the dynamic properties of vehicles and bridges. However, there is inherent complexity in the development of VBI models which must account for multibody dynamics of vehicles, structural dynamics of bridges, and vehicle-bridge contact relationships. Within the literature, three distinct frameworks exist for handling the model complexity of VBI: (1) hard coding the equations of motion of the vehicles and bridges while numerically coupling them in an arbitrary programming language; (2) developing a co-simulation that leverages a commercial finite element (FE) software to model the bridge and separately solve for the equations of motion of a vehicle; or (3) develop high-fidelity representations of both the bridge and vehicle through a commercial FE environment alone. While these unique frameworks offer accurate and reliable results, there exists a trade-off between frameworks that require sophisticated coding of the user but result in high computational speed (1 and 2) and frameworks with simple implementation but longer computation times (3). The present study attempts to bridge this gap and formulate a computationally efficient framework for implementing VBI modeling into a commercial FE software (Abaqus). This requires minimal coding by the user and could then be used by both the industrial and the research community. A VBI model of a passenger vehicle traveling across a bridge is modeled using the bicycle model concept and Euler beam element formulations within Abaqus. Robust node-to-surface contact algorithms within Abaqus are implemented to couple the vehicle to the bridge during the analysis. The dynamic responses of both the vehicle and bridge are verified with data from the literature. The study primarily follows a 2D scheme but concludes with a discussion of how these methods can be extended to a 3D scheme.

Keywords Vehicle-bridge interaction · Commercial finite element · Model verification · Abaqus

8.1 Introduction

Traditional techniques for inspecting damage in bridges rely on subjective assessments that lack resolution and often lead to inconsistent observations [1]. This motivated the research community to migrate toward other methods of quantifying structural damage such as examining changes in its modal/dynamic properties [2]. One recent health monitoring method, known as drive-by health monitoring (DBHM), involves using vehicle-mounted accelerometers to monitor bridge vibration through the acceleration readings taken from the vehicle [3, 4]. Afterward, the measured signals can be analyzed via physics-based [2, 5] or data-driven machine learning techniques [6]. In physics-based approaches, measured signals are often compared to the signals produced by a program-automated finite element (FE) model of the bridge-vehicle system that iteratively simulates different damage scenarios on a bridge, and then a machine learning (ML) algorithm is used to quantify the most likely location and severity of the damage in the bridge [1, 2, 5]. The FE models often consist of a moving sprung mass (i.e., vehicle body) on multiple 1D beam elements where a vehicle-bridge interaction (VBI) system is required to couple the vibration of these two bodies. Within the literature, three distinct frameworks exist for handling the model complexity of VBI: (1) hard coding the equations of motion of the vehicles and bridges while numerically coupling them in an arbitrary programming language [1, 2, 6–9]; (2) developing a co-simulation that leverages a commercial finite element (FE) software to model the bridge and separately solve for the equations of motion of a vehicle [10–12]; or (3) develop high-

O. R. Abuodeh (✉) · L. Redmond
Glenn Department of Civil Engineering, College of Engineering, Clemson University, Clemson, SC, USA
e-mail: oabuode@g.clemson.edu

H. Y. Noh (eds.), *Dynamics of Civil Structures, Volume 2*, Conference Proceedings of the Society for Experimental Mechanics Series, https://doi.org/10.1007/978-3-031-05449-5_8

fidelity representations of both the bridge and vehicle while employing a built-in contact formulation through a commercial FE environment alone [12–15].

While these unique frameworks offer accurate and reliable results, there exists a trade-off between frameworks that require sophisticated coding of the user but result in high computational speed (1 and 2) and frameworks with simple implementation but longer computation times (3). For instance, Yang et al. [7] manually coded in the VBI system for a sprung mass traveling across Euler beam elements to study the feasibility of extracting the bridge's natural frequency from the acceleration data measured from the sprung mass. However, the VBI system used does not account for nonlinear geometry or inelastic material definitions. In particular, Yu et al. [10] sought to overcome these shortcomings by linking MATLAB to Ansys, a commercial FE software with a suite of material definitions, and independently solving the equations of motion of the vehicle to obtain time histories of the tire locations and their respective contact forces. The time histories of these contact forces are applied to the bridge in a separate transient analysis using Ansys. Once the analysis is completed, the resulting time histories of the bridge nodal displacements are applied back to the vehicle model in MATLAB to solve the equations of motion of the vehicle and recompute the new tire forces that begin the analysis cycle again. This process continues until the difference between the tire forces computed in MATLAB and Ansys are minimal. The previous framework is limited to a specialized audience who is proficient in automating FE models with separate programming languages. In addition, it requires multiple iterations for the model to converge for a single vehicular trip, which makes it undesirable for applications like FE model updating in DBHM with ML algorithms.

Developing a VBI system entirely within a user-friendly commercial FE software would simplify the procedure and can be useful to a broader audience. However, to the authors' knowledge, the available published articles that analyze VBI systems completely in a commercial FE software are too computationally expensive for use in physics-based ML algorithms for DBHM [12–15]. For instance, Kwasniewski et al. [15] carried out an extensive 3D FE model of a VBI system involving a heavy truck and a selected highway bridge in Florida within LS-DYNA. The bridge deck, girders, steel reinforcement, and prestressed strands were all included in the model using elastic material definitions. The truck was completely replicated within the FE environment such that the tire was modelled using shell elements with two layers, an elastic rubber material and a fabric material for tire cord, while also employing an airbag option that simulates internal pressure in all tires. As a result, approximately 420,000 elements were generated for this study, including multiple point constraints and contact algorithms used during the dynamic analysis. The researchers reported a good correlation between field measurements and FE analysis. However, fully replicating a field test can prove to be time-consuming and difficult since appropriate modeling strategies must be followed to prevent instabilities that result from nodal misalignment.

The present study attempts to bridge the aforementioned gaps by completely proposing a framework to construct an efficient VBI entirely within a commercial FE software (Abaqus). A quarter-car and half-car models are completely modeled in Abaqus where the VBI is defined using a robust node-to-surface interaction command. Bridge/vehicle data reported in published articles [1, 7] are used to verify the proposed modeling framework.

8.2 Model Development

The most common method for developing a VBI model is to model the vehicle as a sprung mass, hereafter referred to as a quarter-car model, with two degrees of freedom; vertical motion of the center of mass and tire point of the vehicle model. An extension of the quarter-car model that accounts for additional modes is the half-car model, which uses the bicycle concept and has six degrees of freedom, two of which are vertical bounce and pitching motion of the half-car body and the remaining four are vertical motions of the front wheel, rear wheel, front tire, and rear tire. Both of these models are summarized in Fig. 8.1, where the subscripts "q," "*v*," "*t*," "*R*," "*F*," "*S*," and "*R*" are the quarter-car, half-car, tire, rear, front, rear, sprung, and un-sprung, respectively, and u and θ are vertical and pitch degrees of freedom, respectively. M and I are the mass and mass moment of inertia of the vehicles, respectively. K, C, and N are the stiffness, damping, and contact node of the vehicles, respectively. For the bridge model, E_b is the elastic modulus of the beam element, I_b is the moment of inertia of the beam element, A_b is the area of the beam element, μ_b is the mass per length of the beam element, and V is the constant velocity the vehicle drives.

In Abaqus, the vehicle and bridge bodies are modelled independently, and contact is represented using one of Abaqus's interaction definitions. Finally, the vehicle body is pushed with a user-given displacement across the bridge within a time period that is equal to the user-given velocity, while gravity is being applied to the mass of the vehicle.

The bridge elements are discretized with 2-node B23 Euler beam elements which use cubic interpolation functions with consistent mass matrix formulations. The quarter-car body is discretized with 4-node CPS4R plane stress elements and prescribed a rigid body definition. The half-car body is discretized using 2-node B21 linear Timoshenko elements which

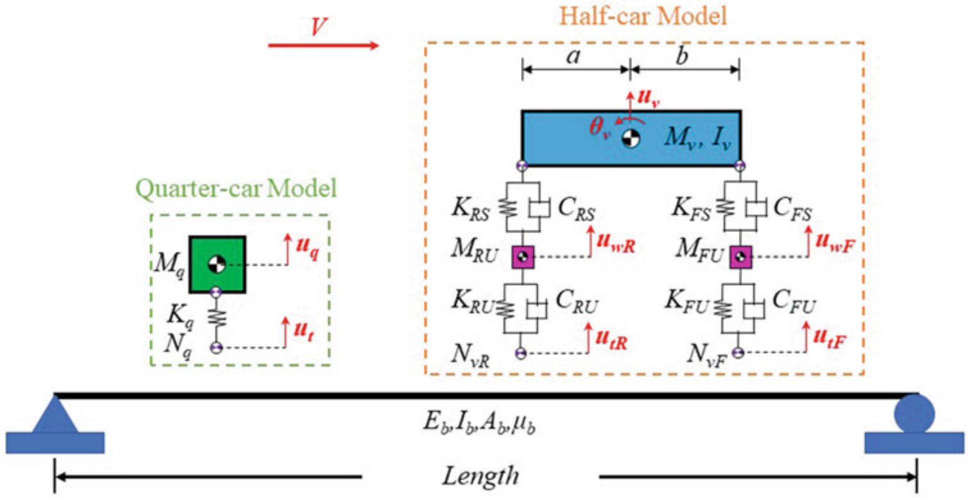

Fig. 8.1 A schematic of a Quarter-car model on a simply supported beam

use lumped mass formulations and are also prescribed a rigid body definition. The stiffness of the quarter-car and half-car models are modelled using SPRING2 elements, and the damping of the half-car model are modelled using DASHPOT2 elements [16]. Essentially, the bridge and vehicle elements are assigned density values to obtain desired inertial properties for the dynamic analysis. For defining density in a vehicle body, if the user desires the vehicle's center of mass to be located away from its midpoint, uniform density cannot be prescribed. Instead, the beam elements should be partitioned into multiple segments each with a user-defined density that would cumulatively yield the desired total mass, mass moment of inertia, and center of mass location. For a half-car model with its center of gravity at the midpoint between the wheels, it is important to note that more than one element should be used, since the mass moment of inertia calculated from the user-given density can deviate from the theoretical one for courser meshes [16]. In addition, depending on the vehicle configuration employed, the nodes that connect to the spring elements should be defined as either *pin* or *tie* nodes, where the pin nodes have only translational degrees of freedom associated with the rigid body (quarter-car model) and the tie nodes have translational and rotational degrees of freedom associated with the rigid body (half-car model) [16].

Furthermore, the vehicle and bridge bodies are coupled together using the *contact pair* formulation. In this study, node-to-surface contact pairs are employed where the tire node is the slave node and the beam surface is the master surface. The "hard" contact pressure-overclosure relationship is used to minimize penetration and avoid excessive contact chatter during the analysis. Similar contact formulations were used in a previous work [17], where Yao et al. constructed a framework for including surface roughness in a commercial FE software and verified it only against a quarter-car model.

Prior to analyzing the problem, the type and number of numerical methods, hereafter referred to as *steps*, should be defined. The vehicle needs to be in vertical static equilibrium before traveling across the beam elements. This requires the user to define gravity for the vehicle throughout all the steps. However, placing a *DASHPOT element within the initial static step results in long convergence times to reach static equilibrium. Therefore, the "`*Model change`" command [16] is used to temporarily remove the dampers during this step followed by adding them back in the next step. The model consists of three steps: two static steps and one implicit dynamic step. The first static step is defined to settle the vehicle body to its static equilibrium position vertically using gravity while removing DASHPOT2 elements. Afterward, the next static step is defined to add back the DASHPOT2 elements while maintaining static equilibrium. Finally, the last implicit dynamic step is defined to push the vehicle with a user-given velocity. During all the steps, the gravity load is only applied to the vehicle body of interest using the "`*Dload`" [16]. Figure 8.2 shows the "`*Dload`" and "`*Model change`" commands that are used in this study where their description can be found in [16]. It is worth noting that the first two steps are not required for a vehicle body without any dampers, similar to the work described in [17].

The Hilber-Hughes-Taylor time integration scheme is used during the implicit dynamic step where the integration parameters α, β, and γ are used for the direct integration of the equations of motion [16]. The β and γ integration parameters are part of the Newmark family where values of 1/4 and 1/2 follow the trapezoidal rule, which is numerically non-dissipative

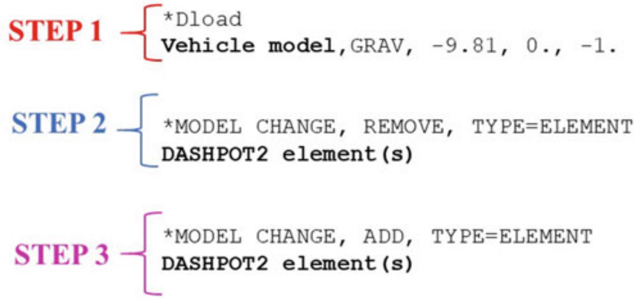

Fig. 8.2 Input file commands for adding gravity and changing element presence

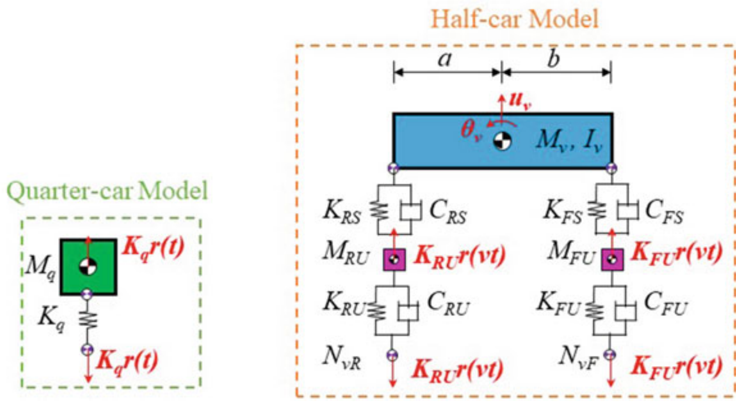

Fig. 8.3 Surface profile idealization for commercial FE software

in the presence of high-frequency vibration problems like contact penetrations/chatter [18]. Therefore, the α integrator parameter was introduced to provide numerical damping where the new relationships are described in Eqs. (8.1) and (8.2).

$$\beta = \frac{1}{4}(1-\alpha)^2 \tag{8.1}$$

$$\gamma = \frac{1}{2} - \alpha \tag{8.2}$$

Including surface roughness into VBI models is necessary to reflect noise present in physical tests of coupled bridge-vehicle systems. In a typical hard-coded FE model, a surface roughness profile is generated where the velocity and acceleration values are derived from the generated surface profile. The displacement, velocity, and acceleration are then used to calculate the contact force between the vehicle and the bridge within the equation of motion in a time-dependent scheme. Yao et al. [17] developed a simpler framework that mimics this dynamic response where it can be easily employed in a commercial FE software like Abaqus. An equivalent force with the magnitude equal to the product of the spring stiffness by the surface elevation is applied on both the contact point and the vehicle body to simulate the dynamic response that is observed when the vehicle drives on a rough surface. Further information regarding the development of this framework can be found in [17]. Figure 8.3 summarizes the framework in which the stiffness notation is similar to that of Fig. 8.1, whereas $r(vt)$ is the surface elevation at the position of the tire during its travel.

Material damping is incorporated into the model to represent realistic material behavior. To incorporate material damping in beam elements, Abaqus contains a material definition for applying the Rayleigh damping method in which two factors must be defined: α_R for mass proportional damping and β_R for stiffness proportional damping [16]. Equations (8.3) and (8.4) are used to compute these factors, where ξ is the damping ratio, ω_1 is the first mode's natural frequency, and ω_2 is the second mode's natura l frequency. The natural frequencies of beam elements can be computed using one of Abaqus's built-in eigenvalue extraction methods: Lanczos, automated multi-level substructuring (AMS), and subspace iteration [11]. The Lanczos solver is used in this study.

$$\alpha_R = \xi \frac{2\omega_1\omega_2}{\omega_1 + \omega_2} \tag{8.3}$$

$$\beta_R = \xi \frac{2}{\omega_1 + \omega_2} \tag{8.4}$$

8.3 VBI Verification Study

In this study, two VBI models, a quarter-car model and a half-car model, are created using the proposed approach in Abaqus, and their results are compared to the dynamic response of hard-coded VBI models taken from the literature [1, 7]. For the quarter-car model, the following properties taken from Yang et al. [7] are used to model the beam: length $L = 25$ m, $E_b = 2.75 \times 10^{10}$ N/m^2, $\mu_b = 4800$ kg/m, $I_b = 0.12$ m^4, and no damping. The vehicle properties are $M_q = 1200$ kg, $K_q = 500{,}000$ N/m, $V = 10$ m/s, and zero damping. The natural frequencies of the bridge (ω_b) and vehicle (ω_v) can be computed individually using the built-in frequency step in Abaqus and are $\omega_b = 2.08$ Hz (first mode's natural frequency) and $\omega_v = 3.25$ Hz. A surface profile was not defined during the quarter-car simulation.

For the half-car model, the following properties are used from Locke [1] to model the beam: $L = 21.3$ m, $E_b = 200 \times 10^9$ N/m^2, $\mu_b = 5600$ kg/m, $\xi = 3\%$, and $I_b = 0.0842$ m^4. The properties of the vehicle are $M_v = 12{,}404$ kg, $I_v = 172{,}160$ kg/m^2, $M_{FU} = 725.4$ kg, $M_{RU} = 725.4$ kg, $K_{FS} = 727{,}812$ N/m, $K_{RS} = 1{,}969{,}034$ N/m, $K_{FU} = 1{,}972{,}900$ N/m, $K_{RU} = 4{,}735{,}000$ N/m, $V = 10$ m/s, $a = 3$ m, and $b = 3$ m. Similar to the quarter-car model, the natural frequencies are computed using the frequency step and are $\omega_b = 6.00$ Hz (first mode's natural frequency), $\omega_{v1} = 1.27$ Hz (pitch with front unsprung bounce), $\omega_{v2} = 2.17$ Hz (pitch mode with rear unsprung mass bounce), $\omega_{v3} = 9.75$ Hz (front unsprung masses bounce), and $\omega_{v4} = 15.4$ Hz (mode rear unsprung bounce). In both models, the beams are discretized into 50 elements following a trial-and-error Scheme. A road profile is generated using the power spectral density (PSD) method defined by ISO-8608 standards [19] where Road Class A was used with a displacement PSD (G_d) of 32×10^{-6} m^{-1} with a spatial frequency (n_0) of 0.1 cycles/m. The spatial frequency band spans from 0.001 to 10 cycles/m at an increment of $1/L$. During the implicit dynamic step, a common issue that users face when employing contact algorithms in Abaqus is contact chatter, which is when a slave node falls off a master surface [16]. To overcome this issue, the *moderate dissipation* application is used to stabilize the model and reduce contact chatter [16] where $\alpha = -0.41421$, $\beta = 0.5$, and $\gamma = 0.91421$, while the time step used was 0.001 seconds. A sensitivity analysis was carried out to test different numerical damping values and found the recommended use of the moderate dissipation application removed contact chatter with and resulted in negligible change in vehicle response and change in bridge response.

Figure 8.4 shows the results that were extracted from the quarter-car simulation for both the proposed FE model and Yang's FE model [2], where Fig. 8.4a, b are the acceleration signals extracted at midspan of the bridge and center of mass

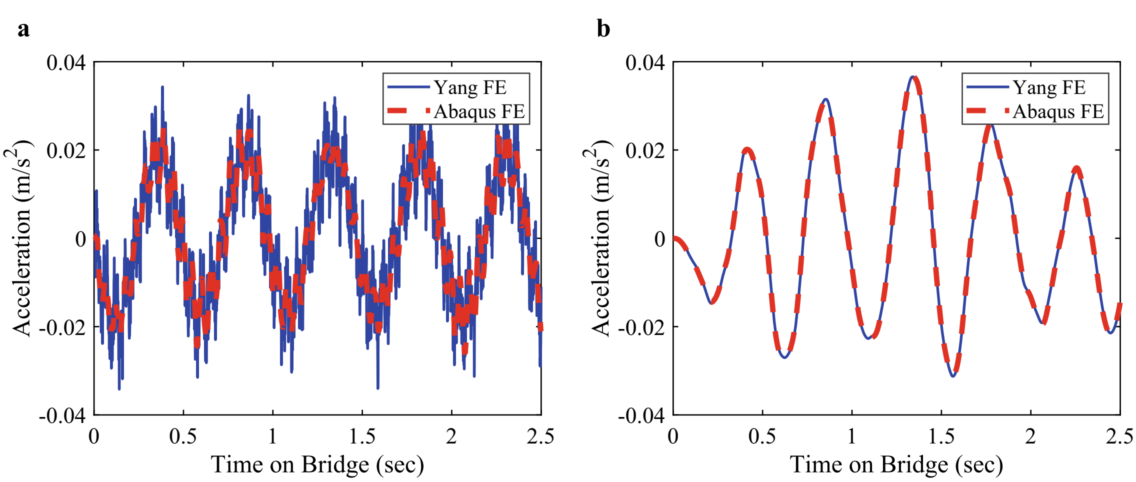

Fig. 8.4 Quarter-Car dynamic response curves. (**a**) Acceleration signal of bridge midspan. (**b**) Acceleration signal of sprung mass

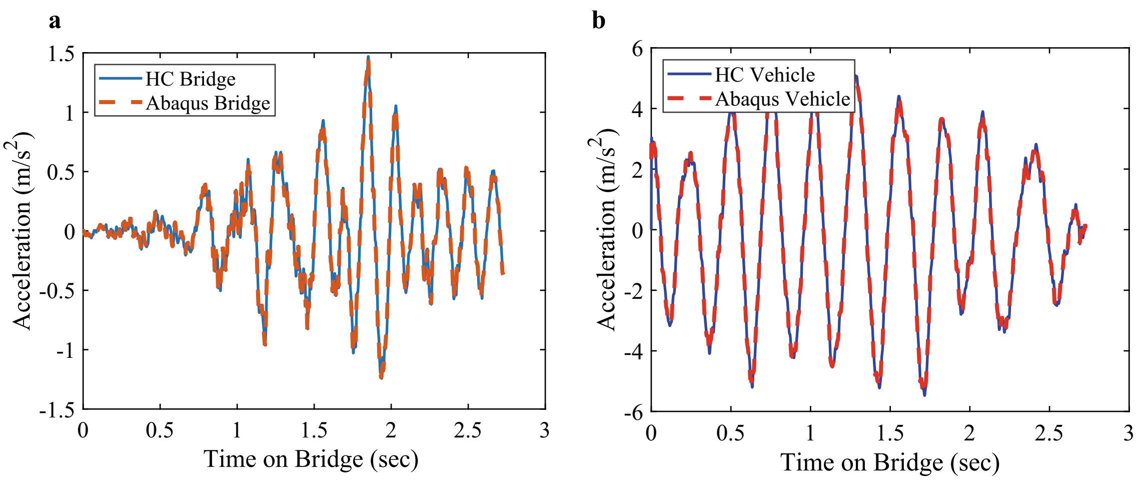

Fig. 8.5 Half-Car dynamic response curves. (**a**) Acceleration signal of bridge at midspan. (**b**) Acceleration signal of vehicle

of the vehicle, respectively. It can be observed that the acceleration at midspan of the bridge for both models are not fully in line with each other. This is attributed to the method employed when distributing the contact force of the tire to the adjacent nodes of a beam element. Yang et al. [2] used the cubic Hermitian interpolation function for the transverse displacement of the element to compute the displacement, velocity, and acceleration of two nodes based on the position of the vehicle within contacted element. However, the proposed Abaqus FE model uses a node-to-surface contact formulation in which the contact force is resolved using a hard contact pressure-overclosure relationship and is based on the tire's contact force as a function of its penetration [11]. This means that Abaqus internally generates a stiffness matrix for the contact area during the analysis in which the equivalent contact force is computed relative to the node of that contacted element, thus causing a smoothing effect on the acceleration response, as shown in Fig. 8.5a. Furthermore, discrete Fourier transform (DFT) is carried out to map acceleration signals into their respective frequency domains for further verification. The computed peak spectrum amplitudes and their corresponding frequencies are 0.0113 m/s^2 and 2.20 Hz in both models, respectively, for the vehicle acceleration signals and 0.00880 m/s^2 and 1.99 Hz, respectively, for the bridge acceleration signals in both models.

Figure 8.5 shows the results that were extracted from the half-car simulation for both the proposed FE model and the hard-coded (HC) FE model [1], where Fig. 8.5a, b are the acceleration signals recorded at midspan of the bridge and center of mass of the vehicle, respectively. As opposed to the quarter-car simulations, the half-car simulation demonstrates bridge acceleration curves that are more in line, as shown in Figs. 8.4a and 8.5a. This is attributed to the effects that material damping and surface roughness have on the dynamic response of the bridge where the sharp amplitudes that are supposed to occur in the HC FE model are smoothened. Similar to the quarter-car simulation, the DFT of the signals are computed where the peak spectral amplitudes and frequencies are 13.7 m/s^2 and 3.66 Hz, respectively, for the vehicle signals in both models, respectively, and 0.0947 m/s^2 and 6.22 Hz, respectively, for the bridge signals in both models.

8.4 Conclusion

This study proposes an efficient framework to construct a VBI model entirely within Abaqus using minimum coding from the user. The implementation of such a framework can be advantageous to users who are interested in exploring the realm of VBI to include nonlinear effects while maximizing computation efficiency. In addition, this study can be expanded to a three-dimensional outlook to increase the user's parameter space by including more modal shapes when processing the data using physics-based or ML-driven approaches. The presented framework is composed of representing the vehicle and bridge bodies using Abaqus's built-in element definitions; defining appropriate material definitions that reflect the physical aspects of the problem; employing a node-to-surface contact formulation that is responsible for coupling the vehicle and bridge; and defining the numerical methods used to solve a typical VBI problem. As a result, the proposed approach was successful in producing FE models that agree with the VBI models employed in published literature. The following can be concluded from the present work:

- Abaqus consists of element and material libraries that can be used to capture the dynamic response of VBI models.
- The use of node-to-surface contact formulations in Abaqus is successful in coupling the vehicle to the bridge.
- The proposed three-step methodology of deleting and adding dashpot elements allows the vehicle to reach static equilibrium quickly before traveling across the bridge.
- Representing surface profiles using equivalent forces is an accurate and simple technique for recreating the effects of surface roughness on the dynamic response of a vehicle body within commercial FE software packages.
- Moderate dissipation in the dynamic analysis step was sufficient to remove contact chatter without significantly influencing coupled vehicle-bridge dynamics.

Acknowledgments Omar Abuodeh was supported by Clemson's NRT RIES program and the National Science Foundation under grant #1633608. Such financial aid is greatly acknowledged.

References

1. Mokalled, S., Locke, W., Abuodeh, O.R., Redmond, L., McMahan, C.: Drive-by health monitoring of highway bridges using Bayesian estimation technique for damage classification. Struct. Control Health Monit. **29**, e2944 (2021)
2. Locke, W., Mokalled, S., Abuodeh, O.R., Redmond, L., Mcmahan, C.S.: An intelligently designed AI for structural health monitoring of a reinforced concrete bridge. Proceedings of The Concrete Industry in the Era of AI, 1–9 (2021)
3. Lin, C.W., Yang, Y.B.: Use of a passing vehicle to scan the fundamental bridge frequencies: an experimental verification. Eng. Struct. **27**(13), 1865–1878 (2005)
4. Miyamoto, A., Yabe, A.: Bridge condition assessment based on vibration responses of passenger vehicle. J. Phys. Conf. Ser. **305**, 1–10 (2011)
5. Lam, H.-F., Yang, J.-H., Au, S.-K.: Markov chain Monte Carlo-based bayesian method for structural model updating and damage detection. Struct. Control. Health Monit. **25**(4), 1–22 (2018)
6. Locke, W., Sybrandt, J., Redmond, L., Safro, I., Atamturktur, S.: Using drive-by health monitoring to detect bridge damage considering environmental and operational effects. J. Sound Vib. **468**(3), 1–17 (2020)
7. Yang, Y.B., Lin, C.W., Yau, J.D.: Extracting bridge frequencies from the dynamic response of a passing vehicle. J. Sound Vib. **272**(3–5), 471–493 (2004)
8. Siringoringo, D.M., Fujino, Y.: Estimating bridge fundamental frequency from vibration response of instrumented passing vehicle: analytical and experimental study. Adv. Struct. Eng. **15**(3), 417–433 (2012)
9. Kim, C.W., Kawatani, M., Kim, K.B.: Three-dimensional dynamic analysis for bridge-vehicle interaction with roadway roughness. Comput. Struct. **83**(18–19), 1627–1645 (2005)
10. Yu, H., Wang, B., Li, Y., Zhang, Y., Zhang, W.: Road vehicle-bridge interaction considering varied vehicle speed based on convenient combination of Simulink and ANSYS. Shock. Vib., 1–14 (2018)
11. Wang, L., Kang, X., Jiang, P.: Vibration analysis of a multi-span continuous bridge subject to complex traffic loading and vehicle dynamic interaction. KSCE J. Civ. Eng., 1–10 (2015)
12. Li, Y., Xu, X., Zhou, Y., Cai, C.S., Qin, J.: An interactive method for the analysis of the simulation of vehicle-bridge coupling vibration using ANSYS and SIMPACK. Proc. Inst. Mech. Eng. F J. Rail Rapid Transit. **232**(3), 663–679 (2018)
13. Lu, X., Kim, C., Chang, K.: Finite element analysis framework for dynamic vehicle-bridge interaction system based on ABAQUS. Int. J. Struct. Stab. Dyn., 1–36 (2020)
14. Shui-rong, G., Lu, L., Shui-sheng, C., Hui, Z.: Research on models of a highway bridge subjected to a moving vehicle based on the LS-DYNA simulator. J. Highw. Transp. Res. Dev. **8**(3), 76–82 (2014)
15. Kwasniewski, L., Li, H., Wekezer, J., Malachowsk, J.: Finite element analysis of vehicle-bridge interaction. Finite Elem. Anal. Des. **42**, 950–959 (2006)
16. Smith, M.: ABAQUS/Standard User's Manual. Dassault Systems Simulia Corp (2020)
17. Yao, Z., Sheng, Z., Tjhen, L.S.: A simple approach for simulating the road surface roughness involved in vehicle-bridge interaction systems. Int. J. Struct. Stab. Dyn. **18**(7), 1–10 (2018)
18. Hilber, H.M., Hughes, T.J.R., Taylor, R.L.: Improved numerical dissipation for time integration algorithms in structural dynamics. Earthq. Eng. Struct. Dyn. **5**, 283–292 (1977)
19. ISO-8608, Mechanical vibration — road surface profiles — reporting of measured data