TTB-2D

User guide

Author: Daniel Cantero

Document version: 11 October 2022

Summary

TTB-2D software has been developed to perform analyses of Train-Track-Ballast-Bridge systems. The TTB-2D has been validated against a similar Abaqus model (Cantero et al., 2016). The user can easily define the configuration of the traversing vehicles together with the infrastructure mechanical properties by modifying the inputs. Additionally, the user can also specify some calculation options and post-processing configurations. After successfully completing the simulation, the software plots the results in graphical form and stores relevant information in new variables.

This software can be used to correctly predict the complex vehicle-bridge interaction phenomenon of railway bridges. The user can model a wide range of vehicle convoys, railway tracks and bridge types by correctly defining the appropriate model properties. The simulation results of TTB-2D mainly focus on the bridge response. However, some results can be obtained for the vehicle and track behaviour.

License

Warranty and liability

1. Introduction

The software performs the dynamic analysis of a train moving along a railway track, which partly rests on a bridge. For short the acronym TTB-2D (Train-Track-Bridge in 2D) is used to refer to the software throughout this document.

This document is a mixture of comments, explanations and guidelines to use this model. The author acknowledges that the user manual is not complete, since it does not provide information about all possible options and variables defined in the model. However, the following text should explain the relevant details for somebody to be able to use the model. This is, only the input, options and output will be discussed. Little or no explanation is provided of the underlying calculations. The program is written in Matlab. It relies on the use of a variable type called "structures". The user needs to be familiar with this variable type before running the program.

The text is divided into two sections, namely, model description and user guide. The model description section presents an explanation of the implemented model, lists most the parameters considered and briefly gives information on the numerical solver procedure. The user guide aims to give some indications on how to use the software.

2. Model description

Correctly modelling dynamic Vehicle-Infrastructure interaction is a complex task. The model needs to couple the dynamic responses of each of the subsystems (vehicle, track, ballast and bridge), which in turn can be described using simple to very complex and detailed models. Generally, the number of variables in a model grows with its complexity. If the number of variables is large, defining the actual numerical values for each of them becomes a complex task on itself. Therefore, it is very important to find the correct trade-off between model complexity and solution accuracy. The models in TTB-2D are commonly accepted in relevant literature, proven to give accurate solutions while maintaining the number of variables to a minimum.

Figure 1 presents an overview of the model implemented in TTB-2D. A train convoy (composed of as many vehicles as the user specifies) travels on a railway track in the left to right direction at a given constant speed. The track includes various elements in order to model all the main components of railway infrastructure, which in top-down order are: rail, pad, sleeper, ballast and sub-ballast. One part of the track is located on a bridge that has been modelled as a beam. Particular modelling details are considered between the bridge and the track. For instance, the user can specify if ballast should be added or not on the bridge. Additional descriptions for each of the subsystems is given in the subsections below.

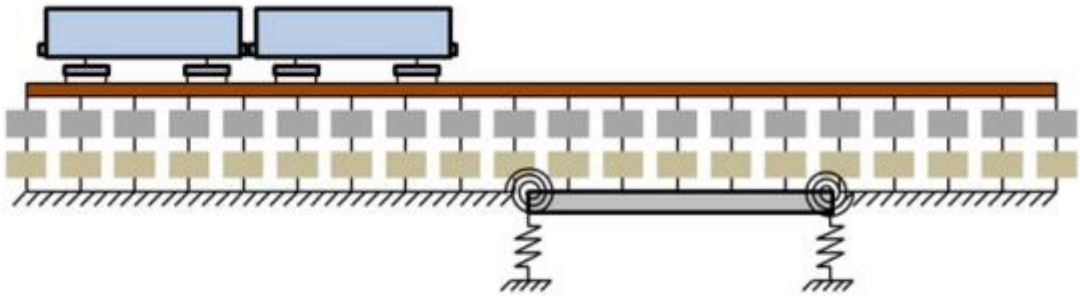

Figure 1: TTB-2D model overview

2.1 Vehicle

A train convoy is defined in TTB-2D as a concatenation of individually defined vehicles. Each vehicle is constructed using the same model, which is fully parameterised. Thus, establishing appropriate values for the variables, the model can describe a wide range of vehicle types, including heavy locomotives, freight wagons and passenger vehicles.

The implemented model for the vehicle consists of a combination of lumped masses, rigid bars, springs and dashpots as presented in Figure 2. This model is extensively used in related literature. The wheels, represented as masses on the rail, are connected to the bogies by the primary suspension. The secondary suspension links the bogies and the main body mass. Each suspension system is made of a combination of linear springs and viscous dampers. The main body and bogies are represented as rigid bars with mass and moments of inertia properties.

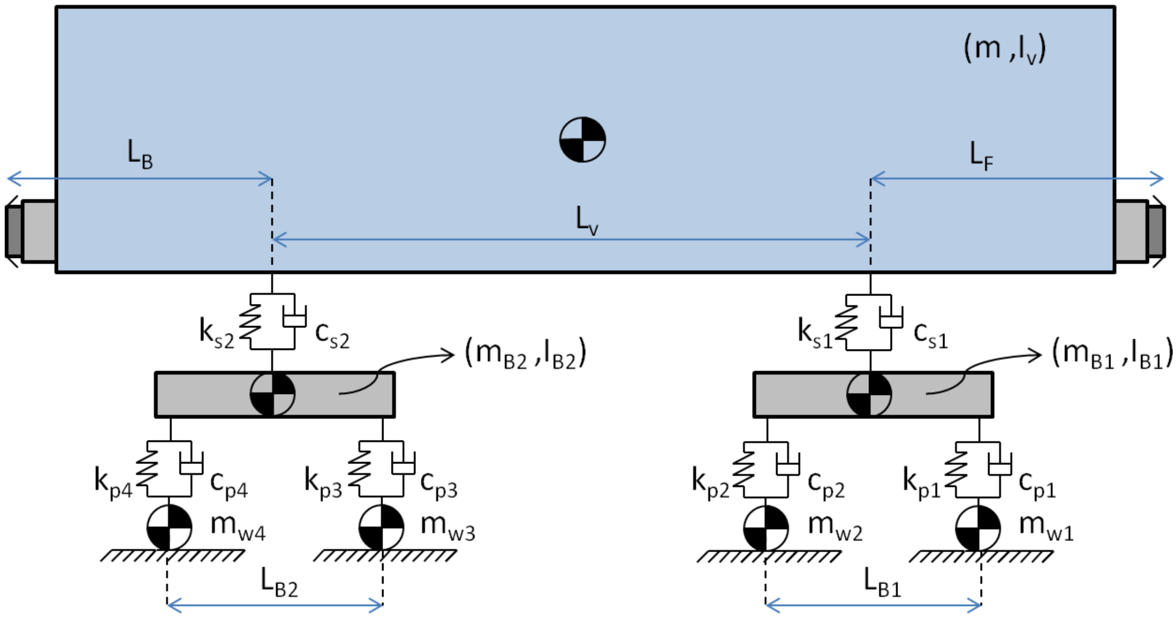

Figure 2: Vehicle model

The user needs to define the variables in Table 1 for each vehicle. A value needs to be assigned for each of these mechanical and geometrical properties in order to run TTB-2D. The user might find appropriate values in (Cheng et al., 2011; Continental, 2006; Ferrara, 2013; Fryba, 1996; James, 2003; Savini, 2010; Zhai et al., 2009) amongst others.

Table 1: List of vehicle variables

Variable	Description
m	Main body mass
I_v	Main body moment of inertia
L_v	Main body length (axle to axle)
L_F	Additional front length
L_B	Additional back length
m_{Bi}	Mass of i-th bogie
I_{Bi}	Moment of inertia of i-th bogie
L_{Bi}	Bogie length
m_w	Total mass of i-th axle and wheels
k_{pi}	Primary suspension vertical stiffness of i-th axle
c_{pi}	Primary suspension vertical viscous damping of i-th axle
k_{si}	Secondary suspension vertical stiffness of i-th axle
c_{si}	Secondary suspension vertical viscous damping of i-th axle

These variables can easily be related to the variables in the script following the comments.

2.2 Bridge

The bridge is modelled as an Euler-Bernoulli beam using a Finite Element Model (FEM) discretization. Figure 3 shows a sketch of the model together with the associated variables.

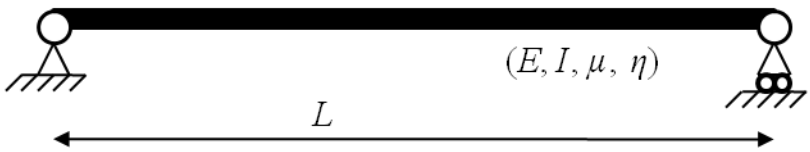

Figure 3: Sketch of bridge model

Table 2 lists the variable symbols and descriptions for the bridge model. A value needs to be assigned for each of these properties in order to run TTB-2D. The user might refer to (Bjorklund, 2004; Lin et al., 2005; Shahbaz, 2011; Xia et al., 2003) amongst others to find appropriate values.

Table 2: List of beam variables

Variable	Description
L	Bridge span
I	Section's second moment of area
μ	Mass per unit length of bridge
η	Damping ratio

In addition, the user can define other structural configurations, by specifying the desired boundary conditions at different support locations. It is possible to define multiple supports located along the beam, as shown in Figure 4. For each support, the user can specify the value of the vertical stiffness and rotation stiffness. If the value of any of these stiffnesses is set to -1, then the corresponding vertical displacement (or rotation) is fixed (See Table 3).

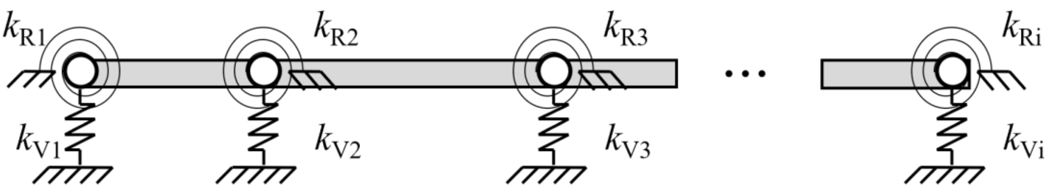

Figure 4: Beam with several supports. Each support with possible vertical and rotational stiffness

Table 3: List of support variables

Variable	Description
k_{Vi}	Vertical stiffness at support *i* (-1 = stiff, 0 = none, >0 = stiffness value)
k_{Ri}	Rotational stiffness at support *i* (-1 = stiff, 0 = none, >0 = stiffness value)

2.3 Track

The track is modelled as a beam resting on periodically spaced sprung mass systems (see Figure 5). This is a generally accepted model that correctly predicts the dynamic vertical displacements of the track system. The rail is modelled by beam elements in a FEM framework, whereas the masses represent the sleepers and ballast. Beam and masses are connected by spring/dashpot systems that represent the pad, ballast and sub-ballast.

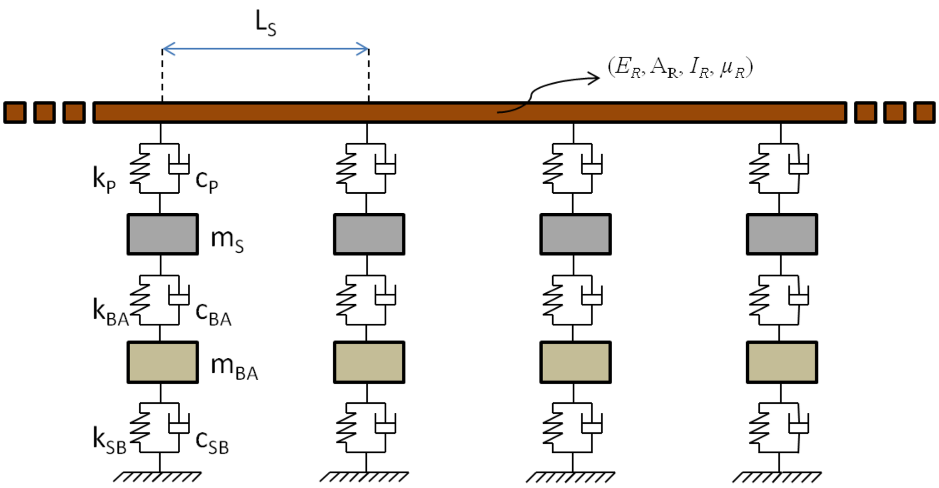

Figure 5: Sketch of track model

Table 4 lists the variable symbols and descriptions for the track model, subdivided into each of the components. A value needs to be assigned for each of these properties in order to run TTB-2D. The user might refer to (Feng, 2011; Ferrara, 2013; Savini, 2010) amongst others to find appropriate values.

Table 4: List of track variables

Variable	Description
	Rail
E_R	Young's modulus of rail material
A_R	Rail section's area
I_R	Rail section's second moment of area
μ_R	Mass per unit length of rail
	Pads
k_P	Vertical stiffness of pad
c_P	Vertical viscous damping of pad
	Sleepers
L_S	Distance between sleepers
m_S	Mass of each sleeper
	Ballast
m_{BA}	Mobilized ballast mass
k_{BA}	Ballast vertical stiffness
c_{BA}	Ballast vertical viscous damping
	Sub-Ballast
k_{SB}	Sub-ballast vertical stiffness
c_{SB}	Sub-ballast vertical viscous damping

2.4 Track on Bridge

In the particular case when the track is on top of the bridge structure, the model is defined as a combination of bridge (Figure 3) and track (Figure 5) models with some modifications. Two alternatives can be implemented in TTB-2D, depending on whether ballast on bridge should be included or not.

2.4.1 With ballast

Figure 6 presents the implemented model that represents the situation where the railway track is on the bridge including some ballast. In this case the masses representing the ballast rest directly on top of the bridge. Note that different variable names have been assigned to the ballast mechanical properties compared to the track (only) model in Figure 5, since the ballast layer on the bridge will be generally of smaller thickness and thus will feature different mechanical properties. New variable symbols and descriptions are listed in
Table 5.

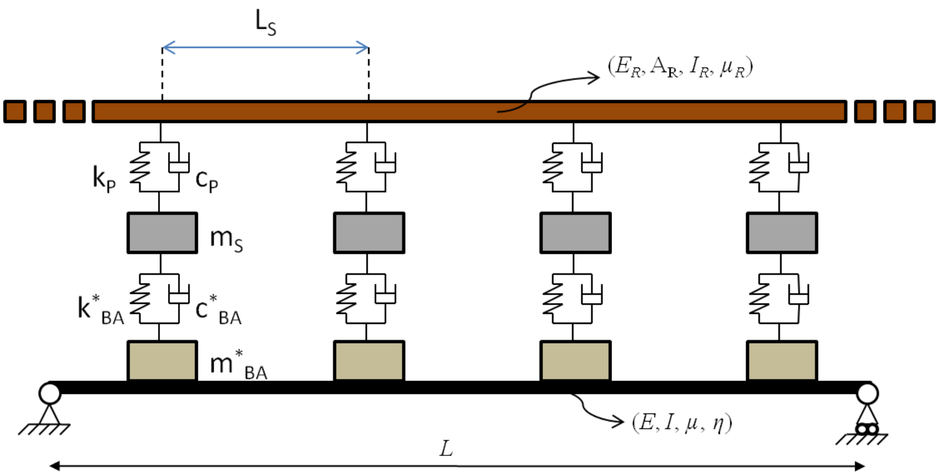

Figure 6: Sketch of track on bridge model, with ballast

Table 5: List of ballast on bridge variables

Variable	Description
	Ballast on bridge
m^*_{BA}	Mobilized ballast mass
k^*_{BA}	Ballast vertical stiffness
c^*_{BA}	Ballast vertical viscous damping

2.4.2 No ballast

An alternative model is defined to represent the case where the sleepers rest directly on top of the bridge (See Figure 7). The connection between sleeper and bridge is through another pad, termed here, Pad Under sleeper (PU). New variable symbols and descriptions are listed in Table 6.

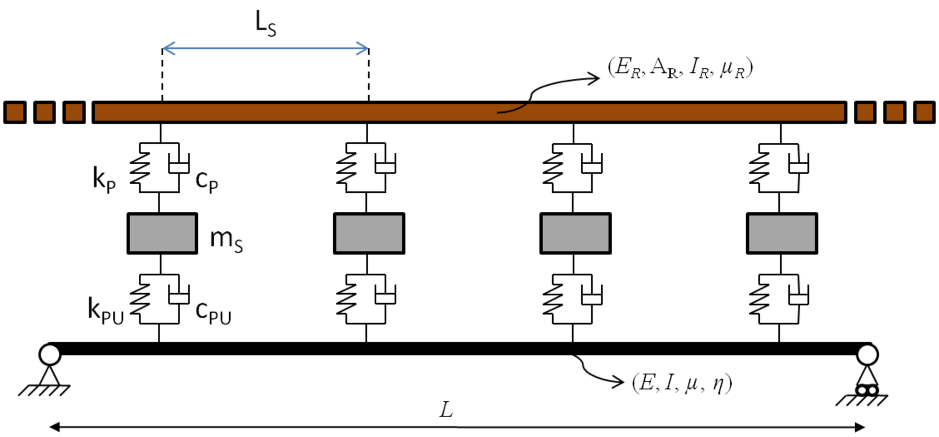

Figure 7: Sketch of track on bridge model, without ballast

Table 6: List of pad under sleeper variables

Variable	Description
Pad under sleeper	
k_{PU}	Pad under sleeper vertical stiffness
c_{PU}	Pad under sleeper viscous damping

2.5 Numerical solution

The models presented above for the vehicle and infrastructure are linear. However, when the models interact, the systems are then coupled together and the problem becomes non-linear. To solve the coupled equations of motion TTB-2D updates the system matrices for each time step. The dynamic problem is then solved by direct integration using the Newmark-β algorithm.

3. User guide

In order to run TTB-2D for the first time the user can use the script *A00_Run.m*, which already includes the definition of all the inputs and options. This main script calls the rest of the functions and script listed in the folder.

Train

- It is possible to define a train composed of as many consecutive vehicles (wagons) as desired. Simply define the properties of each wagon in a different element of the *Train.Veh* structure. For example, the properties of the first vehicle are in *Train.Veh(1)*, for vehicle 2 in *Train.Veh(2)*, and so on.

Track

- All properties related to the track definition are stored in the structure *Track*.

Bridge

- All properties related to the bridge definition are stored in the structure *Beam*.

Profile (Irregularities)

- Current version defines a perfectly smooth profile when selecting the option *Calc.Profile.Type=0*
- The user can also define a the irregularities in terms of Power Spectra Density (PSD) definitions. In this case, then *Calc.Profile.Type=1*, and the software generates a random irregularity profile based on the specified PSD. The script already includes examples of standard PSD definitions found in literature and railway authorities.

Mesh properties

- A number of variables have to be defined to specify the mesh to be used in the model.
- *Beam.Mesh.Ele.num_per_spacing* = The number of elements to be used in the main beam member for each sleeper spacing.
- *Track.Rail.Mesh.Ele.num_per_spacing* = The number of elements to be used in the rail for each sleeper spacing.
- *Track.Rail.Options.num_add_sleepers* = Number of additional sleepers to include left of the vehicle at simulation start.
- *Calc.Profile.minL_Approach* = The length of track to be considered before the bridge.
- *Calc.Profile.minL_After* = The length of track to be considered after the bridge.

Additional options

- There are some additional calculation options to be defined:
- *Calc.Options.calc_beam_sections* = Specify here the section (or multiple sections) where the program should calculate the response in the main beam member.
- *Calc.Solver.max_accurate_frq* = This variable is used to specify the size of the time step to be used in the numerical solver. It is expressed in terms of Hz. It specifies up to what

frequency the solver should provide accurate results. The time step is equal to the inverse of twice the selected frequency.
- *Calc.Options.VBI* = Option to connect or disconnect the Vehicle-Bridge Interaction (VBI). If 0 then there is no VBI included, and the solution is the moving load problem. If equal to 1, then vehicle and bridge interact, and the model solves the coupled problem.
- *Calc.Options.redux* = Option to reduce the size of the model. If the value is set to 1, then the model assumes that there is no track below the vehicle at the start of the simulation. Only as the vehicle moves it enters the track model. This simplification reduces the size of the model but introduces only some small initial vibrations to the vehicle when each axle enters the track. Select a value = 0, to avoid this simplification.

Plotting options
- The user can specify what plots of the results should be displayed at the end of the simulation. Read the comments in the script for further information.
- *Calc.Plot.Model.P00_ModelVisualization* = Option to generate a schematic plot of the model defined by the user. If this figure is generated the execution of the program is paused until the user reviews the figure and presses Enter on the keyboard. After that, TTB-2D proceeds and performs the simulation of the specified model.

Calculations
- Once all the inputs and options are specified, the script continues with the actual calculations to solve the model. This script performs all the calculations including:
    - Checks the correctness of some inputs. However, note that these checks do not cover all possible errors that the user might produce during the input definition.
    - Creates the system matrices of vehicle, track and beam, together with additional auxiliary variables needed for the calculation.
    - Solves the coupled equations of motions by direct integration.
    - Generates a series of outputs based on the nodal displacements, including contact forces, deformations, bending moments, shear forces and accelerations.

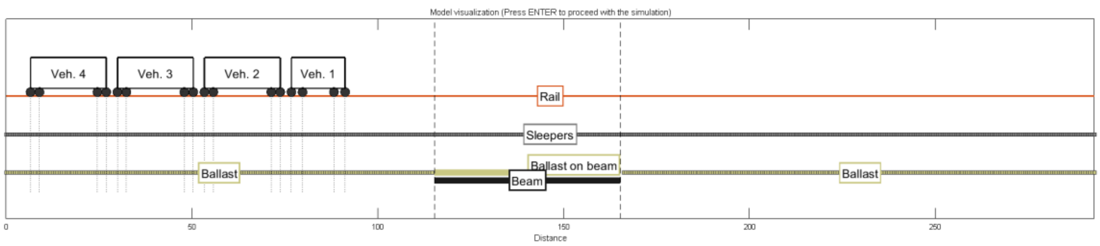

Figure 8: Example of schematic plot of a model when the option *Calc.Plot.Model.P00_ModelVisualization* = 1. The figure shows that the model to be simulated consists of a train with one locomotive and 3 wagons. The figure also shows the dimensions of the track+beam model defined for this particular example.

3.1 Working with the output

All the results of the finished simulation are included in the Matlab structure *Sol*. The results are organized in various sub-structures.

The easiest way to work with the results from the model is to use the arrays obtained at the sections. See the structure/substructure:

*Sol.Beam.(*Load Effect*).sections_t*

Also, the user can directly extract the results from the nodal displacements of the whole model. To facilitate this task, the user should use following substructure:

*Model.Mesh.DOF.(*Name of member*)*

These variables specify the DOF that correspond to the particular member of interest. With this information it is straightforward to obtain the response of a member in time. For instance, if the user wants to get the acceleration time-history response of all the DOFs of the sleepers at the approach, then select: *Sol.Model.Nodal.A(Model.Mesh.DOF.sleepers_app,:)*

3.2 Miscellanea

Below are some comments to different aspects of the model:

- If correct contact forces are desirable as an output of the simulation, then the *redux* option needs to be switched off. When redux is on, calculations are much faster but contact force is not too reliable because of the "jump" experienced by the train while entering the track.
- The coupling of the vehicle and track is performed in *B65*, however the model uses the *B65_faster version*. The original *B65* script is left in the folder for reference on how the coupling is performed step-by-step.

References

Bjorklund, L. (2004). Dynamic analysis of railway bridge subject to high speed trains. Master thesis at Royal institute of technology, Stockholm, Sweden.

Cantero, D., Arvidsson, T., OBrien, E.J., Karoumi, R. (2016). Train–track–bridge modelling and review of parameters. *Structure and Infrastructure Engineering*. Vol. 12, No. 9, pp. 1051-1064, 2016.

Cheng, Y.C., Chen, C.H. and Yang, C.J. (2011). Dynamic analysis of high-speed railway vehicles excited by wind loads. International journal of structural stability and dynamics, Vol. 11, No. 6, pp. 1103-1118.

Continental (2006). Primary and secondary suspension systems for rail vehicles.

Feng, H. (2011). 3D-models of railway track for dynamic analysis. Master thesis at Royal institute of technology, Stockholm, Sweden.

Ferrara, R. (2013). A numerical model to predict train induced vibrations and dynamic overloads. PhD thesis, at University of Reggio Calabria and University of Montpellier 2.

Fryba, L. (1996). Dynamics of railway bridges. Academia Praha.

James, G (2003). Analysis of traffic load effects on railway bridges. PhD thesis at Royal institute of technology, Stockholm, Sweden.

Lin, C.C., Wang, J.F. and Chen, B.L. (2005). Train-induced vibration control of high-speed railway bridges equipped with multiple tuned mass dampers. Journal of bridge engineering, pp. 398-414.

Savini, G. (2010). A numerical program for railway vehicle-track-structure dynamic interaction using a modal substructuring approach. University of Bologna, Italy.

Shahbaz, R. (2011). Parametric study of bridge response to high speed trains. Master thesis at Royal institute of technology, Stockholm, Sweden.

Xia, H., Zhang, N. and De Roeck, G. (2003). Dynamic analysis of high speed railway bridge under articulated trains. Computers and structures, Vol. 81, pp. 2467-2478.

Zhai, W., Wang, K. and Cai, C. (2009). Fundamentals of vehicle-track coupled dynamics. Vehicle System Dynamics, Vol. 47, No. 11, pp. 1349-1376.