OTREC

OREGON TRANSPORTATION RESEARCH AND EDUCATION CONSORTIUM

FINAL REPORT

Combined Seismic Plus Live-Load Analysis of Highway Bridges

OTREC-RR-11-20
October 2011

A National University Transportation Center sponsored by the U.S. Department of Transportation's Research and Innovative Technology Administration

COMBINED SEISMIC PLUS LIVE-LOAD ANALYSIS OF HIGHWAY BRIDGES

Final Report

OTREC-RR-11-20

by

Prof. Michael H. Scott
and
Minjie Zhu

Oregon State University

for
Oregon Transportation Research
and Education Consortium (OTREC)
P.O. Box 751
Portland, OR 97207

OTREC
OREGON TRANSPORTATION RESEARCH AND EDUCATION CONSORTIUM

October 2011

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
OTREC-RR-11-20		

4. Title and Subtitle	5. Report Date
COMBINED SEISMIC PLUS LIVE-LOAD ANALYSIS OF HIGHWAY BRIDGES	October 2011
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Michael H. Scott	

9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
Oregon State University Structural Engineering 220 Owen Hall Corvallis, OR 97331	11. Contract or Grant No. 2009-261

12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
Oregon Transportation Research and Education Consortium (OTREC) P.O. Box 751 Portland, Oregon 97207	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

The combination of seismic and vehicle live loadings on bridges is an important design consideration. There are well-established design provisions for how the individual loadings affect bridge response: structural components that carry vertical live loads are designed to remain well within the linear-elastic range while lateral load carrying components are designed to yield under large seismic excitations. The weight of the bridge superstructure is taken in to account as dead load in structural analysis for seismic loads; however, the effects of additional mass and damping of live loads on the bridge deck are neglected. To improve the design of highway bridges for multi-hazard effects of seismic plus live load, many questions arise and are addressed in this project via numerical simulations of short span bridges. Further extensions of this research can be extended to long span bridges whose seismic response is more heavily influenced by vehicle mass on the bridge deck.

17. Key Words	18. Distribution Statement
Seismic, bridge design, structural analysis, live load	No restrictions. Copies available from OTREC: www.otrec.us

19. Security Classification (of this report)	20. Security Classification (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	40	

ACKNOWLEDGEMENTS

This project was funded by the Oregon Transportation Research and Education Consortium (OTREC) with match funding from the Oregon Department of Transportation (ODOT). The authors are thankful for this opportunity provided by OTREC and ODOT.

DISCLAIMER

TABLE OF CONTENTS

EXECUTIVE SUMMARY **1**
1.0 MODELING APPROACHES **3**
1.1 BRIDGE MODEL 3
1.2 VEHICLE MODEL 3
2.0 SCRIPTS FOR COMBINED ANALYSIS **7**
2.1 BRIDGE MODEL 7
2.2 VEHICLE "CONTAINERS" 9
2.3 SEISMIC AND GRAVITY LOADS 10
2.4 STRUCTURAL DAMPING 11
2.5 MOVING VEHICLES 13
2.6 COMBINED ANALYSIS 14
2.7 OVERVIEW OF THE BRIDGE ANALYSIS APPLICATION 15
3.0 EXAMPLES **19**
3.1 SUPPORTED BEAM SUBJECTED TO A MOVING VEHICLE 19
3.2 EXPERIMENT OF A SMALL-SCALE BRIDGE MODEL UNDER A MOVING MASS 20
3.3 EXPERIMENT OF A LONG-SPAN BOX GIRDER BRIDGE SUBJECTED TO MOVING VEHICLES 25
3.4 THE LONG-SPAN BOX GIRDER BRIDGE SUBJECTED TO MOVING VEHICLES PLUS EARTHQUAKE 27
4.0 CONCLUSIONS **29**
5.0 REFERENCES **31**

LIST OF TABLES

Table 3.1: Comparison between Numerical and Experimental Results of Maximum Deflection (mm) at the point 7/16l 21

LIST OF FIGURES

Figure 1.1: Bridge model 3
Figure 1.2: The sprung mass model 4
Figure 1.3: The structural expression of the sprung mass model 4
Figure 1.4: Bridge model with containers 5
Figure 2.1: Tcl script for creating bridge model 7
Figure 2.2: Tcl script for defining section 8
Figure 2.3: Tcl script for saving and recording 8
Figure 2.4: Complete script for creating bridge and vehicle model 9
Figure 2.5: Tcl script for applying gravity loads 10
Figure 2.6: Tcl script for applying seismic loads 11

Figure 2.7: Tcl script for applying structural damping 11
Figure 2.8: Tcl script for moving vehicle properties 12
Figure 2.9: Moving the vehicle properties across the bridge 13
Figure 2.10: Tcl script for combined analysis 14
Figure 2.11: Overview of the application 15
Figure 2.12: The driver of the application 16
Figure 3.1: Moving vehicle on a simply supported beam 19
Figure 3.2: The midpoint displacement from the analytical solution (left) and the bridge analysis application (right) 20
Figure 3.3: The midpoint acceleration from the analytical solution (left) and the bridge analysis application (right) 20
Figure 3.4: Experimental model 22
Figure 3.5: Experimental (left) and Numerical (right) results for entrance speed V1 22
Figure 3.6: Experimental (left) and Numerical (right) results for entrance speed V2 22
Figure 3.7: Experimental (left) and Numerical (right) results for entrance speed V3 23
Figure 3.8: Experimental (left) and Numerical (right) results for entrance speed V4 23
Figure 3.9: Experimental (left) and Numerical (right) results for entrance speed V5 23
Figure 3.10: Experimental (left) and Numerical (right) results for entrance speed V6 24
Figure 3.11: Experimental (left) and Numerical (right) results for entrance speed V7 24
Figure 3.12: Experimental (left) and Numerical (right) results for entrance speed V8 24
Figure 3.13: Elevation and the cross section of the box girder bridge 25
Figure 3.14: Experimental (left) and Numerical (right) results for vehicle speed 20km/h 26
Figure 3.15: Experimental (left) and Numerical (right) results for vehicle speed 30km/h 26
Figure 3.16: Experimental (left) and Numerical (right) results for vehicle speed 40km/h 26
Figure 3.17: Experimental (left) and Numerical (right) results for vehicle speed 50km/h 27
Figure 3.18: Experimental (left) and Numerical (right) results for vehicle speed 60km/h 27
Figure 3.19: Plot of maximum displacements of the bridge induced by earthquake only (left) and combined earthquake and live load (right) 28
Figure 3.20: Plot of maximum accelerations of the bridge induced by earthquake only (left) and combined earthquake and live load (right) 28

EXECUTIVE SUMMARY

The seismic analysis is a conventional part of structural analysis (Chopra, 2001). All the finite element programs implement it as a basic function. The effects of moving vehicles are always of special concerns in bridge engineering. Previously, vehicles were often approximated as moving loads, which in many cases allows a finite element program to implement it without difficulties. But as increasingly larger transportation vehicles are manufactured, bridges are subjected to larger and heavier loads so that the inertia effect of the vehicles can no longer be neglected.

To consider this effect, various vehicle models were developed by researchers. Thus the problem of vehicle-bridge interaction becomes an issue of great concern which has been addressed extensively in recent literature (Lu et al., 2009; Li et al., 2008; Xia et al., 2006; Ju et al., 2006; Kwasniewski et al., 2006; Broquet et al., 2004; Kim and Kawatani, 2006; Yang and Wu, 2002; Yang and Wu, 2001; Nassif et al., 2003; Lee and Yhim, 2005; Yang et al., 1997). However, the combined effect of theses loads has not been adequately addressed (Matsumoto et al., 2004; Yau and Fryba, 2007; Fryba and Yau, 2009; Sogabe et al., 2007).

In analyzing the VBI (Vehicle Bridge Interaction) system, two sets of equations of motion have to be written each for the vehicles and for the bridge. The two equations are coupled together by the interaction forces at contact points. One way to solve the two sets of equations is through an iterative procedure (Broquet et al., 2004; Xia et al., 2006). For example, by first assuming the weights of vehicles as interaction forces, one can solve the bridge equations to obtain the displacements at contact points and then proceed to solve the vehicle equations for improved values of interaction forces. This is the first cycle of iteration. For this method, the convergence rate is likely to be low when dealing with large problems.

Another way is to merge two equations by eliminating the contact forces between vehicle wheels and the bridge to one equation (Yang and Wu, 2001; Kim and Kawatani, 2006; Li et al., 2008; Lu et al., 2009). This is an efficient way to solve VBI problem, especially for computing the bridge responses. The only drawback is that one should develop his own program to solve the condensed equations. To do the combined analysis, the function of seismic analysis needs to be added into the program. This could be very time consuming and unnecessary, since the tools for seismic analysis in existing finite element programs have been proven reliable and stable for years.

The last option would be to incorporate the vehicle model in existing finite element software by defining the vehicles as structural components. The advantage is that all the tools for structural analysis, including seismic analysis, are available to use. But the problem is that most finite element programs assume the structure is invariant in a same analysis. In other words, vehicles defined as structural components are unable to move. Taking these into account, an idea is advanced in this study; that is, to move the vehicle properties instead of vehicle itself. This aim can be achieved with the help of OpenSees and Tcl scripting language.

OpenSees (Open System for Earthquake Engineering Simulation) is a software framework aimed at simulating the seismic response of structural and geotechnical systems. OpenSees has been

developed as the computational platform for research in performance-based earthquake engineering at the Pacific Earthquake Engineering Research Center (PEER), and is also the simulation component for the NEESit since 2004. OpenSees is open source and designed in a modular fashion to support the finite element method with loose coupling of analysis and model building components. Developers do not need to know everything that is in the framework, allowing them to make improvements or create applications in their areas of expertise.

Most users of OpenSees build models and conduct analysis via the string-based Tcl scripting language. Tcl is fully programmable with the control structures, variable substitutions and procedures that are necessary to automate routine operations using scripts (Welch, 2000). The aim of Tcl is to serve as a glue language that assembles software building blocks into customized applications. This is accomplished by allowing developers to extend the Tcl interpreter with commands that suit the needs of an application. In the case of OpenSees, the Tcl interpreter is extended with commands to define the nodes, boundary conditions, elements, loads, mass and solution strategies of a finite element analysis (Mazzoni et al., 2009).

The objective of this report is to describe the method for combined seismic and live-load analysis of bridges in OpenSees, and to outline how Tcl is used as a glue to construct a bridge analysis application from finite element analysis building blocks in the OpenSees framework. The presentation begins with modeling approaches of the bridge and vehicles. The subsequent sections present Tcl scripts that define bridge and vehicle models, apply seismic and gravity loads, and perform the analysis. Analytical and experimental examples are shown to demonstrate the method and the application, and then the conclusion is made.

1.0 MODELING APPROACHES

1.1 BRIDGE MODEL

The bridge is modeled as a series of nodes and beam-column elements as shown in Figure 1.1. The equation of the forced vibration of a bridge under external loads and support excitations is

$$M_b\ddot{U}_b + C_b\dot{U}_b + R_b(U_b) = f_b + p_{eff} \quad \textbf{(2-1)}$$

Where M_b and C_b respectively represent mass and damping matrices of the bridge, R_b is the resisting force vector, f_b is the external force vector applied on the bridge (e.g., gravity), p_{eff} is the effective earthquake forces (Chopra, 2001), and U_b is the displacement vector of the bridge.

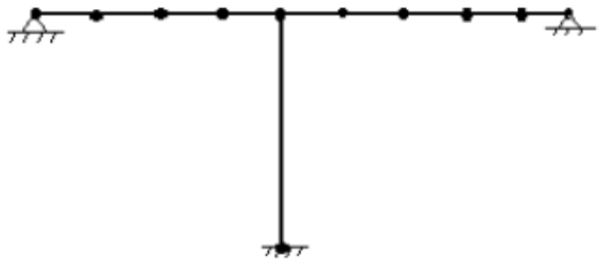

Figure 1.1: Bridge model

1.2 VEHICLE MODEL

The simplest model for considering inertia effect of moving vehicles is the moving mass model (Yang and Wu, 2001; Kozar, 2009). However, for heavy vehicles, the vertical acceleration of the large mass makes it necessary to consider the effect of the vehicles' suspension systems. The simplest model in this regard is a lumped mass supported by a spring-dashpot unit, often referred to as the sprung mass model (Yang and Wu, 2001), as shown in Figure 1.2. Though there are more sophisticated models, the sprung mass model is sufficient for this research because only the bridge response is of interest.

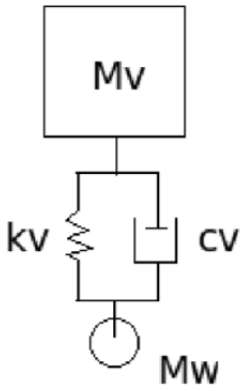

Figure 1.2: The sprung mass model

The sprung mass model can be expressed in structure as two nodes which represent the vehicle body and wheels linked together by a vertical element which represents the spring (Figure 1.3).

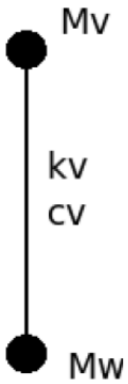

Figure 1.3: The structural expression of the sprung mass model

Instead of moving the vehicle model, the vehicle properties will be moved with time. However, vehicle properties are always associated with a concrete vehicle model. In order to move the vehicle properties, enough vehicle models need to be defined for moving the properties in or out (Figure 1.4). These vehicle models don't actually represent real vehicles, but only "containers" for holding moving properties. As real vehicles moving on the bridge, the vehicle properties are moved from one container to another. Therefore, some of the "containers" are empty when vehicles move away and some of them are full when vehicles move in.

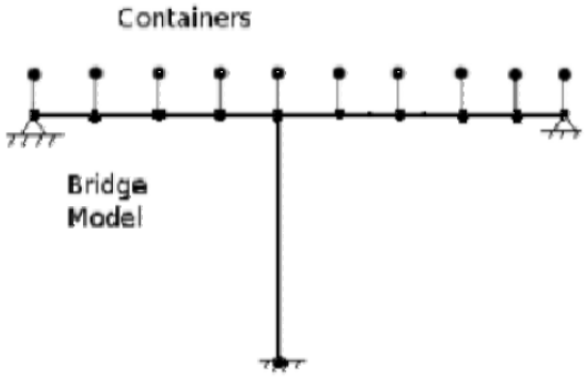

Figure 1.4: Bridge model with containers

If we let U_c denote the displacement vector of the nodes of containers, an enlarged bridge equation will be obtained as

$$\begin{bmatrix} M_b & 0 \\ 0 & M_c \end{bmatrix}\begin{bmatrix} \ddot{U}_b \\ \ddot{U}_c \end{bmatrix}+\begin{bmatrix} C_{bb} & C_{bc} \\ C_{cb} & C_{cc} \end{bmatrix}\begin{bmatrix} \dot{U}_b \\ \dot{U}_c \end{bmatrix}+\begin{bmatrix} R_b(U_b,U_c) \\ R_c(U_b,U_c) \end{bmatrix}=\begin{bmatrix} f_b \\ f_c \end{bmatrix}+p_{eff} \qquad \textbf{(2-2)}$$

Where $M_c, C_{bc}, C_{cb}, C_{cc}, R_c$ indicate properties which are associated with nodes of containers, f_b and f_c are force vectors. The partial derivative of the resisting forces R is required to find dynamic equilibrium at each time step of a simulation:

$$K=\frac{\partial R}{\partial U}=\begin{bmatrix} \partial R_b/\partial U_b & \partial R_b/\partial U_c \\ \partial R_c/\partial U_b & \partial R_c/\partial U_c \end{bmatrix}=\begin{bmatrix} K_{bb} & K_{bc} \\ K_{cb} & K_{cc} \end{bmatrix} \qquad \textbf{(2-3)}$$

There are two problems with introducing the "containers":

- Empty containers do not actually exist on the bridge and therefore will change the original stiffness and geometry of the bridge. This can be overcome by using a special element in OpenSees.

- To accurately track the moving vehicles, a large number of "containers" have to be defined. The static DOF (Degree of Freedom) is increased a lot. But the mass of empty containers is zero. So these DOFs won't be included in the dynamic analysis.

2.0 SCRIPTS FOR COMBINED ANALYSIS

In the following sections, details about the model building and combined seismic and live-load analysis in OpenSees are presented. The Tcl procedures for bridge and vehicle model building, model saving, load applying, and results recording demonstrate the Tcl syntax and compose the bridge analysis application presented herein.

2.1 BRIDGE MODEL

A mesh of nodes and beam elements of the bridge is generated by the Tcl script CreateModel.tcl as shown in Figure 2.1.

CreateModel.tcl

```
#Define Bridge Model
DefineSection $BridgeSection $ColumnSection
for { set i 0 } { $i < $Nspan } {incr i} {
  set nele [lindex $Nele $i]
  set lele [expr [lindex $LSpan $i]/$nele]
  for {set j 1} {$j<=$nele} {incr j} {
    set Lbridge [expr $Lbridge+$lele]
    incr BridgeNode
    node $BridgeNode $Lbridge 0.0
    incr ElementCounter
    element dispBeamColumn <element and node tags> <material properties>
  }
  if {$i != [expr $Nspan-1]} {
    incr ColumnNode
    node $ColumnNode $Lbridge -$Height
    incr ElementCounter
    element dispBeamColumn <element and node tags> <material properties>
    fix $ColumnNode 1 1 1
  } else {
    fix $BridgeNode <right support>
  }
}
set OldMass [expr $TotalMass/($BridgeNode-$BridgeNode0+1)]
for {set i $BridgeNode0} {$i <= $BridgeNode} {incr i} {
  mass $i $OldMass $OldMass 0.
}
```

Figure 2.1: Tcl script for creating bridge model

The script invokes the “node,” “element,” “mass” commands added to the Tcl interpreter by OpenSees and a procedure “DefineSection” defined in a separate script (Figure 2.2). The length and the number of elements of each span are contained in two lists “LSpan” and “Nele.” Variables “ElementCounter,” “BridgeNode” and “ColumnNode,” starting from “ElementCounter0,” “BridgeNode0” and “ColumnNode0,” can assign a unique tag to each element and node. Two nested loops are used to iterate over spans to define the bridge nodes and elements one by one. Variable “Oldmass” stores the original mass of each node.

DefinSection.tcl

```
proc DefineSection {BridgeSection ColumnSection} {
  source DefineUnits.tcl ;
  set E [expr 2.87e6*$kN/$m**2];
  set A 1;
  set I [expr 2.9*$m**4];
  section Elastic $BridgeSection $E $A $I;
  section Elastic $ColumnSection $E $A $I;
}
```

Figure 2.2: Tcl script for defining section

SaveModel.tcl

```
#Node Coordinates
foreach node [getNodeTags] {
  set x  [format "%.2f" [nodeCoord $node X]]
  set y  [format "%.2f" [nodeCoord $node Y]]
  set z  [format "%.2f" [nodeCoord $node Z]]
  puts $FileID "$node $x $y $z"
}
#Element Nodes
foreach elem [getEleTags] {
  puts $FileID "$elem [eleNodes $elem]"
}
#Modal Analysis
foreach w [eigen $Nmode] {
  puts $FileID [expr sqrt($w)]
}
#Support Nodes
for {set i $ColumnNode0} {$i <= $ColumnNode} {incr i} {
  puts $FileID $i
}
puts $FileID "$BridgeNode0\n$BridgeNode"
```

CreateRecorder.tcl

```
foreach recordertype $Recorders {
foreach n $nodes($recordertype) {
foreach d $DOF($recordertype) {
  set filename "$ProjectName/$ProjectName.$recordertype"
  append filename "_node$n"
  append filename "_dof$d"
  append filename "_$Type"
  if {[string first E $Type]>=0} {append filename "_$GMRecords"}
  if {$Index!=0} {append filename "_$LoopName\($Index\)"}
  if {[string equal $n all]} {
    recorder Node -file $filename <nodeRange> -dof $d $recordertype
  } else {
    recorder Node -file $filename -node $n -dof $d $recordertype
  }
}
}
}
```

Figure 2.3: Tcl script for saving and recording

After the bridge model is created a Tcl script SaveModel.tcl will be invoked, including a modal analysis to save the basic information of the bridge (e.g., node and element tags, nature frequencies of the bridge, etc.). Another script CreateRecorder.tcl will then invokes the "recorder" command to record the time history responses for specific nodes. The two scripts are shown in Figure 2.3.

2.2 VEHICLE "CONTAINERS"

The vehicle "containers" introduced in the last chapter are also defined in the script CreateModel.tcl. The complete script CreateModel.tcl is shown in Figure 2.4.

CreateModel.tcl

```
#Define Bridge Model
DefineSection $BridgeSection $ColumnSection
for { set i 0 } { $i < $Nspan } {incr i} {
  set nele [lindex $Nele $i]
  set lele [expr [lindex $LSpan $i]/$nele]
  for {set j 1} {$j<=$nele} {incr j} {
    set Lbridge [expr $Lbridge+$lele]
    incr BridgeNode
    node $BridgeNode $Lbridge 0.0
    incr ElementCounter
    element dispBeamColumn <element and node tags> <material properties>

    ###Define vehicle containers###
    incr BodyNode
    node $BodyNode $Lbridge 0.0
    incr ElementCounter
    element zeroLength $ElementCounter $BridgeNode $BodyNode -mat 1 -dir 2
    fix $BodyNode 1 0 1

  }
  if {$i != [expr $Nspan-1]} {
    incr ColumnNode
    node $ColumnNode $Lbridge -$Height
    incr ElementCounter
    element dispBeamColumn <element and node tags> <material properties>
    fix $ColumnNode 1 1 1
  } else {
    fix $BridgeNode <right support>
  }
}
set OldMass [expr $TotalMass/($BridgeNode-$BridgeNode0+1)]
for {set i $BridgeNode0} {$i <= $BridgeNode} {incr i} {
  mass $i $OldMass $OldMass 0.
}
```

Figure 2.4: Complete script for creating bridge and vehicle model

In order to minimize the influence of the "containers" on the bridge, a special element called "zeroLength" is used. The "zeroLength" element connects two nodes at the same location with previously defined materials. So this element has zero length and, therefore, has almost no influence on the bridge structure. The "UniaxialMaterial" command is used to define the material for the "zeroLength" element. If an elastic uniaxial material with tangent E and damping tangent eta is associated with the "zeroLength" element, E and eta are interpreted as the ratio of force to deformation and the rate of deformation. While in other context, E and eta are interpreted as the ratio of stress to strain and rate of strain. This makes it convenient to represent the springs and dampers. This is another merit of the "zeroLength" element.

For no consideration of jump between the wheel and the bridge, the wheel nodes should be defined on the bridge. For simplicity, using the bridge nodes directly as the wheel nodes can minimize the extra nodes and reasonably set the number of "containers" equal to the number of bridge nodes. To guarantee the structure's stability, the body nodes need to be fixed in the horizontal and rotational directions.

2.3 SEISMIC AND GRAVITY LOADS

Script GravityLoads.tcl (Figure 2.5) applies the gravity loads on the bridge nodes by invoking the "pattern" command with the "Plain" type in OpenSees.

GravityLoads.tcl

```
#apply gravity loads
proc GravityLoads {GlobalData ModelData} {
  pattern Plain -1 Constant {
    for {set i $BridgeNode0} {$i <= $BridgeNode} {incr i} {
      load $i 0.0 [expr -$Oldmass*$G] 0.0
    }
  }
}
```

Figure 2.5: Tcl script for applying gravity loads

To apply seismic loads, a procedure seismicLoads also uses the "pattern" command, but with the "MultipleSupport" type to forcibly apply the ground motion records at bridge supports as shown in Figure 2.6. The ground motion records are from the PEER strong motion database. A script ReadSMDFile.tcl, which can be found at the OpenSees homepage (http://opensees.berkeley.edu), converts a PEER strong motion database file to OpenSees format. Vertical and horizontal ground motions can be applied via two "pattern" commands.

SeismicLoads.tcl

```
#Apply seismic loads
proc SeismicLoads {GlobalData ModelData} {
  if {[llength $GroundMotionH]!=0} {
    foreach  member $GroundMotionH {
      ReadSMDFile $member $member.data DT
    }
    set accelH <series command>
    set velH <series command>
    set dispH <series command>
    pattern MultipleSupport -2 {
      groundMotion 1 Plain -accel $accelH -vel $velH -disp $dispH
      for <loop for all supports> {
         imposedMotion $supportnode 1 1
      }
    }
  }
  if {[llength $GroundMotionV]!=0} {
    foreach  member $GroundMotionV {
      ReadSMDFile $member $member.data DT
    }
    set accelV <series command>
    set velV <series command>
    set dispV <series command>
    pattern MultipleSupport -3 {
      groundMotion 2 Plain -accel $accelV -vel $velV -disp $dispV
      for <loop for all support> {
         imposedMotion $supportnode 2 2
      }
    }
  }
}
```

Figure 2.6: Tcl script for applying seismic loads

2.4 STRUCTURAL DAMPING

The script for applying structural damping is shown in Figure 2.7.

AddDamping.tcl

```
set w [eigen frequency 2]
set w1 [expr sqrt([lindex $w 0])]
set w2 [expr sqrt([lindex $w 1])]
set a1 [expr 2.*($w1*$zeta1-$w2*$zeta2)/($w1*$w1-$w2*$w2)]
set a0 [expr 2.*$w1*$w2*($w1*$zeta2-$w2*$zeta1)/($w1*$w1-$w2*$w2)]
rayleigh $a0 $a1 0.0 0.0
```

Figure 2.7: Tcl script for applying structural damping

MovingVehicle.tcl

```
proc MovingVehicle {GlobalData ModelData ith rpos rbodypos} {
  upvar $rpos pos
  upvar $rbodypos bodypos
  eval $GlobalData
  eval $ModelData
  set t [expr $ith*$dt]
  set loc [expr $Speed*$t]
  for {set i 0} {$i < $NAxle} {incr i} {
    if {$loc<=0 || $loc>=$Lbridge} {
      set pos($i) $BridgeNode0
      set bodypos($i) $BodyNode0
    }
    else {
      while {$loc>[nodeCoord $pos($i) X]} {
        incr pos($i)
        incr bodypos($i)
      }
    }
    set factor <relatvie location>
    mass [expr $pos($i)-1] <wheel mass on the left container>
    mass $pos($i) <wheel mass on the right container>
    mass [expr $bodypos($i)-1] <body mass on the left container>
    mass $bodypos($i) <body mass on the right container>
    pattern Plain $i Constant {
      load $pos($i) 0.0 [expr -(1.0-$factor)*$AxleMass($i)*$G] 0.0
      load [expr $pos($i)-1] 0.0 [expr -$factor*$AxleMass($i)*$G] 0.0
      load $pos($i) 0.0 [expr -(1.0-$factor)*$BodyMass($i)*$G] 0.0
      load [expr $pos($i)-1] 0.0 [expr -$factor*$BodyMass($i)*$G] 0.0
    }
    if {$i!=[expr $NAxle-1]} {set loc [expr $loc-$AxleSpacing($i)]}
  }
}
```

Figure 2.8: Tcl script for moving vehicle properties

Rayleigh damping is used in OpenSees. The damping matrix is calculated in OpenSees as

$$D = \alpha_M M + \beta_K K_{current} + \beta_{Kinit} K_{init} + \beta_{Kcomm} K_{comm} \quad \textbf{(3-1)}$$

where, D is the damping matrix, M is the mass matrix, $K_{current}$ is the stiffness matrix at current state, K_{init} is the stiffness matrix at initial state and K_{comm} is the stiffness matrix at last committed state; α_M, β_K, β_{Kinit} and β_{Kcomm} are coefficients for those matrices. In this application, only the stiffness at current state is considered. So β_{Kinit} and β_{Kcomm} are set to zero. The coefficients α_M and β_K, which is denoted as a0 and a1 in the script, are obtained from

$$\alpha_M = 2\frac{\omega_1\omega_2(\omega_1\zeta_2 - \omega_2\zeta_1)}{\omega_1^2 - \omega_2^2}$$
$$\beta_K = 2\frac{\omega_1\zeta_1 - \omega_2\zeta_2}{\omega_1^2 - \omega_2^2} \quad \textbf{(3-2)}$$

Where, ω_1 and ω_2 are the first and second frequencies of the bridge, which are obtained in the script SaveModel.tcl (Figure 2.3), ζ_1 and ζ_2 are the damping ratios of the first and second modes which are set in the input file.

2.5 MOVING VEHICLES

A Tcl script MovingVehicle.tcl that puts the properties of vehicles in containers at specified locations is shown in Figure 2.8. Figure 2.9 shows how vehicles properties are moved across the bridge.

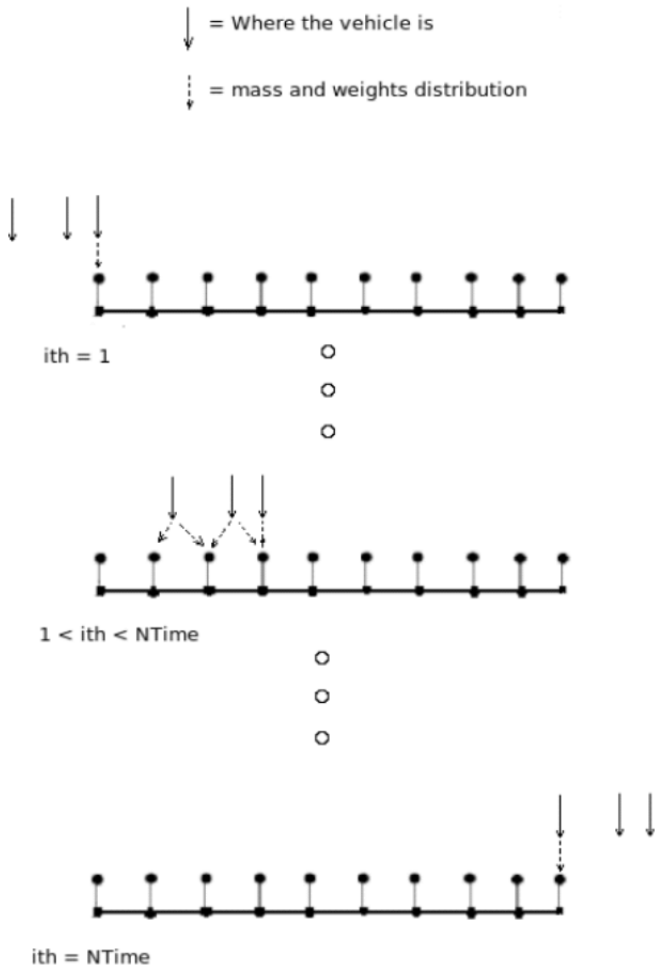

Figure 2.9: Moving the vehicle properties across the bridge

The arrays “AxleMass” and “AxleSpacing” contain the mass and relative spacing of vehicle wheels. Another array “BodyMass” contains the mass of vehicle bodies. A while loop is used to determine where the vehicle is. The location of a vehicle won’t be always just right on a vehicle container. Usually, the exact location is between two containers. A variable “factor” stores the relative location of the vehicle with respect to the nearest two containers. The vehicle’s mass and weight are assigned to the two end containers according to the “factor.”(based on a linear function?) The variable “factor” makes sure that the farther the node is from the vehicle, the less mass and weight will be assigned on it. Since a container uses the bridge node as the wheel node, the new mass of the bridge node should be the sum of its original mass “OldMass” and the vehicle’s wheel mass.

2.6 COMBINED ANALYSIS

The analysis is triggered by a Tcl script CombinedAnalysis.tcl (Figure 2.10).

CombinedAnalysis.tcl

```
if {[string first G $Type]>=0} {GravityLoads $GlobalData $ModelData}
if {[string first E $Type]>=0} {SeismicLoads $GlobalData $ModelData}
if { $zeta1!=0 && $zeta2!=0 } {AddDamping $GlobalData}
for {set ith 0} {$ith<=$NTime} {incr ith} {
  if {[string first M $Type]>=0} {
    MovingLoads $GlobalData $ModelData $ith pos bodypos
  }
  #analysis options
  <set analysis options>
  #analyze
  if {$ith!=0} {
    analysis Transient
  } else {
    integrator LoadControl 0
    analysis Static
  }
  if {[analyze 1 $dt]<0} {
    puts "Analysis failed at time [expr $i*$dt]"
    return
  }
  #Remove wheel mass and weight from bridge
  if {[string first M $Type]>=0} {
    for {set i 0} {$i < $NAxle} {incr i} {
      remove loadPattern $i
      if {$pos($i) != $BridgeNode0} {
        mass $pos($i) $OldMass $OldMass  0.0
        mass [expr $pos($i)-1] $OldMass $OldMass 0.0
        mass [expr $bodypos($i)-1] 0.0 0.0 0.0
        mass $bodypos($i) 0.0 0.0 0.0
}  }  } } }
```

Figure 2.10: Tcl script for combined analysis

Before entering the time loop, the scripts GravityLoads.tcl and SeismicLoads.tcl (Figure 2.5 and Figure 2.6) are invoked to apply gravity and seismic loads on the bridge. Then the script AddDamping.tcl (Figure 2.7) is called to add damping on the structure. At each time step, procedure MovingVehicle (Figure 2.8) will be invoked first to assign the properties for all vehicles in certain containers. Then the script sets the analysis options. The constraint equations are enforced in the analysis through the "constraints" command. There are several constraints handlers in OpenSees. Since the ground motions are enforced at the bridge supports that produce a multi-point constraint, the "Transformation" method is chosen. This method reduces the size of the system for multipoint constraints and is also recommended for transient analysis. The "integrator" command sets the Newmark method with $\beta = 0.25$ and $\gamma = 0.5$ for time integration. After the "analyze" command invokes the structural analysis, the last thing is to remove the vehicle's properties from containers. The weights and mass of the vehicles are removed and the mass of bridge nodes set to original values. This process repeats with a constant time increment until the end time is reached.

2.7 OVERVIEW OF THE BRIDGE ANALYSIS APPLICATION

A global picture of how the bridge analysis application works is shown in Figure 2.11.

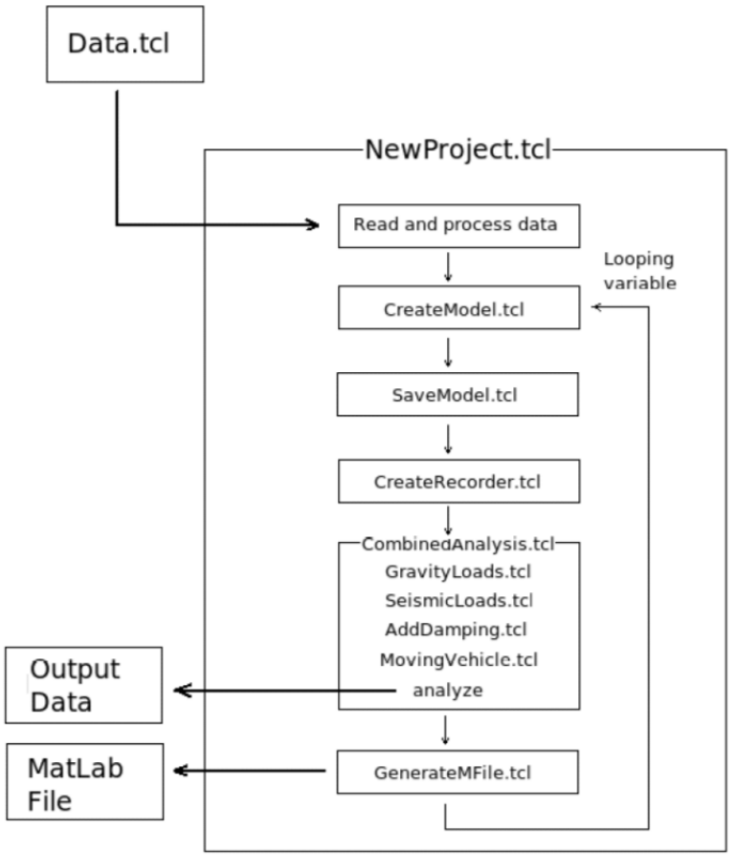

Figure 2.11: Overview of the application

The entry point of the application is the NewProject.tcl (Figure 2.12), which works like a driver for invoking procedures introduced above and all the input and output jobs.

NewProject.tcl

```
proc NewProject {ProjectName} {
  #Read all data
  if {[catch {open "$ProjectName/Data.tcl" r} DataFile]} {
    puts "Can't open file $ProjectName/Data.tcl!"
    return -1
  }
  if {[catch {read $DataFile} Data]} {
    puts "Can't read file $ProjectName/Data.tcl!"
    close $DataFile
    return -1
  }
  close $DataFile
  #Process data
  set l1 [string first "##########Section Information##########" $Data]
  set l2 [string first "##########Looping Data##########" $Data]
  set GlobalData [string range $Data 0 [expr $l1-1]]
  set SectionInfo [string range $Data $l1 [expr $l2-1]]
  set LoopData [string range $Data $l2 end]
  eval $SectionInfo
  eval $LoopData
  set LoopTimes [llength $LoopValues]
  if {$LoopTimes>0} {
    set i0 1
  } else {
    set i0 0
  }
  #Analysis
  for {set i $i0} {$i<=$LoopTimes} {incr i} {
    wipe
    puts "$i: $LoopName = [lindex $LoopValues [expr $i-1]]"
    if {$i0==1} {
      append GlobalData "\n set $LoopName [lindex $LoopValues [expr $i-1]]"
      set ModelData [CreateModel $GlobalData $ProjectName]
      SaveModel $GlobalData $ModelData $ProjectName
      CreateRecorder $GlobalData $ProjectName $i $LoopName $ModelData
      CombinedAnalysis $GlobalData $ModelData
      wipe
      GenerateMFile $ProjectName
    }
}
```

Figure 2.12: The driver of the application

The only parameter for the procedure NewProject is "ProjectName," which is also the name of the working directory. An input file Data.tcl should be prepared in the working directory in advance. Input data in this file are read to the memory and processed to three parts: the global data, section information and looping variable. If the looping variable "LoopValues" is set with a

list of values, the analysis will be repeated over the list of values (e.g., does 100-times analysis with vehicle speed from 10km/s to 100km/s). After the bridge and vehicle models are defined by evaluating the CreateModel.tcl script shown in Figure 2.4, the model is saved and recorders are created by executing the SaveModel.tcl and CreateRecorder.tcl scripts in Figure 2.3. The analysis is then done in the script CombinedAnalysis.tcl in Figure 2.10. At last, a script GenerateMFile.tcl is invoked to generate a Matlab file for displaying the results. The output files include output data of time history responses of specified nodes and a Matlab file.

3.0 EXAMPLES

To verify the present method and application, three examples are compared to the analytical solutions or experimental results. Another example shows the combined analysis.

3.1 SUPPORTED BEAM SUBJECTED TO A MOVING VEHICLE

Figure 3.1 shows a simply supported beam subjected to a moving vehicle.

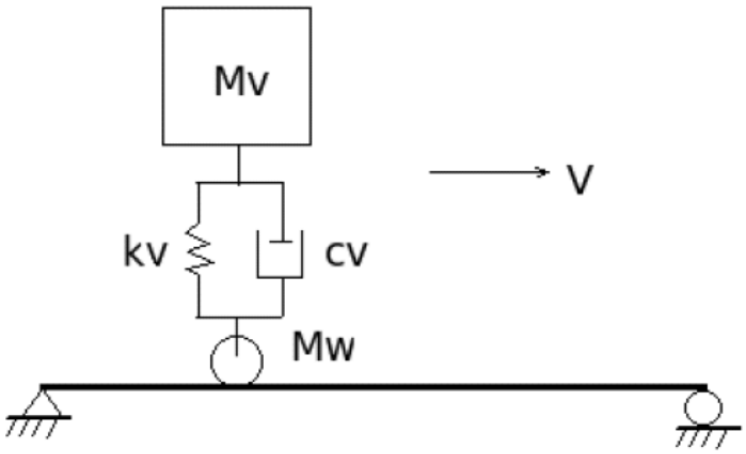

Figure 3.1: Moving vehicle on a simply supported beam

This example, as well as its analytical solution, is originally given in Yang and Wu (2001). The length of the beam is 25 meters. A displacement-based beam element is used in the analysis with Young's modulus $= 2.87 \times 10^6 kN/m^2$, moment of inertia $= 2.9m^4$ and mass $= 2.303t/m$. The vehicle is moving at speed $= 27.78m/s$ with body mass $M_v = 5.75t$ and suspension spring constant $k_v = 1595kN/m$. For demonstration purpose, the effect of the dashpot damping and the wheel mass will be neglected, i.e. $M_w = 0$ and $C_v = 0$.

The midpoint displacement obtained from the analytical solution and the bridge analysis application is plotted in Figure 3.2.

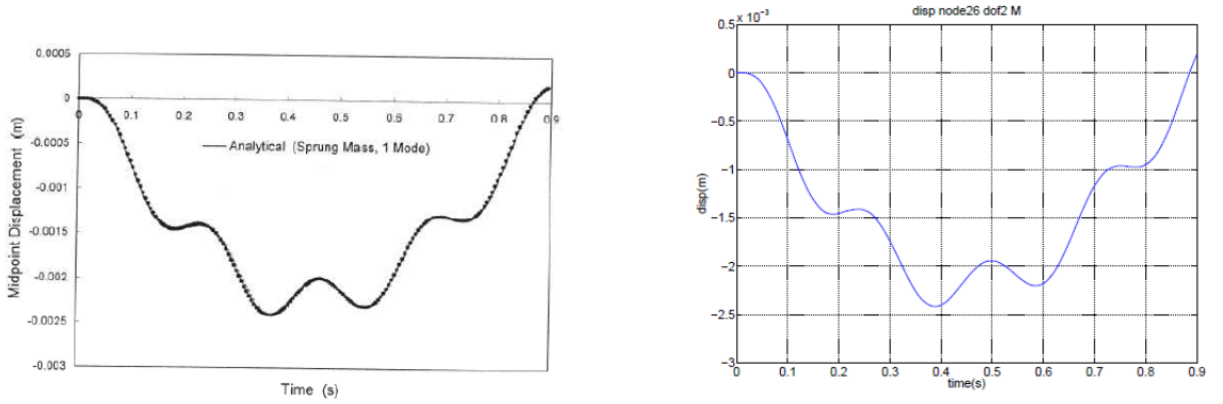

Figure 3.2: The midpoint displacement from the analytical solution (left) and the bridge analysis application (right)

The analytical solution considers only the first mode of vibration (Yang and Wu, 2001), but the time history analysis carried out by the application includes higher modes. As can be seen, the results from the application agree well with the analytical solution. For the response of midpoint displacement, the first mode is overwhelming to other modes. However, the vertical acceleration of the midpoint of the beam in Figure 3.3 shows that the higher modes can result in apparently oscillatory responses to the acceleration. The maximum acceleration is higher when higher modes are included. So it is necessary to take into account the higher-mode effects of the beam for acceleration analysis.

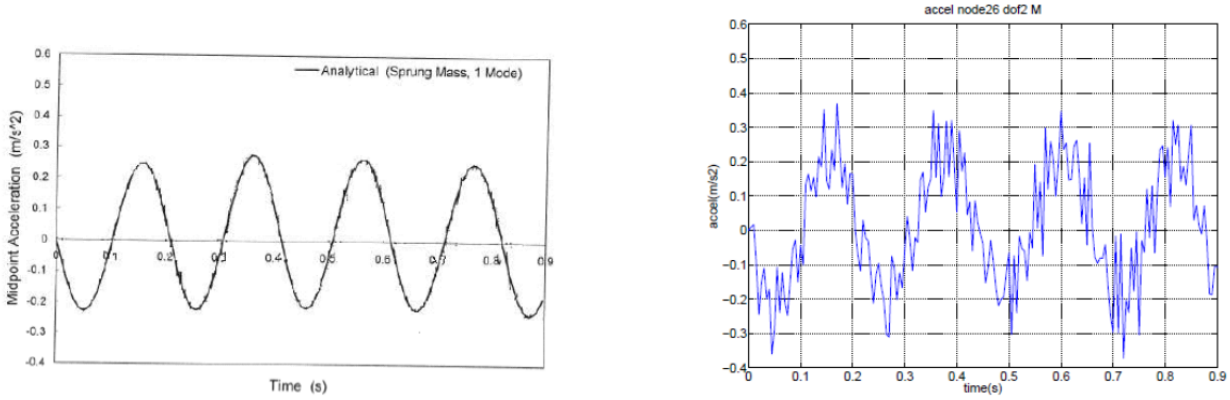

Figure 3.3: The midpoint acceleration from the analytical solution (left) and the bridge analysis application (right)

3.2 EXPERIMENT OF A SMALL-SCALE BRIDGE MODEL UNDER A MOVING MASS

This experiment (Bilello et al., 2004) was carried out on a reduced scale model as shown in Figure 3.4. The beam is simply supported along premachined shallow notches. The weight of the beam, including the rail, is 19.79N, corresponding to a linear mass density of 1.847kg/m. The

bending stiffness is $EI_m = 162.6N \cdot m^2$. The length of the beam is 1.0715 meters. A plywood ramp is placed next to the model to accelerate a moving mass prior to entering the model. The moving mass is a 1.954N steel disk. Since a moving mass is used, the spring and body mass are neglected by setting $M_v = 0$ and $C_v = 0$. Eight tests labeled V 1 − V 8 were performed with eight entrance speeds varying from $v \approx 3km/h$ to $v \approx 7.6km/h$.

According to Bilello et al. (2004), the measured response (i.e., displacement) is basically due to the first beam mode that agrees with the conclusion of the last example. The results of the beam deflection at point x = 7/16l for entrance speeds V1-V8 from the experiments and the bridge analysis application are shown in Figures 3.5-3.12. The comparison of maximum deflection at the point x = 7/16 l between the experiment and the bridge analysis application with damping ratios $\zeta = 0.05$ and $\zeta = 0.01$ is also shown in Table 3.1.

Table 3.1: Comparison between Numerical and Experimental Results of Maximum Deflection (mm) at the point 7/16l

Test	Speed(km/h)	Experimental	Numerical		Difference (%)	
			$\zeta = 0.05$	$\zeta = 0.01$	$\zeta = 0.05$	$\zeta = 0.01$
V1	3.06	0.747	0.7714	0.7857	3.27	5.18
V2	4.12	0.756	0.7742	0.7913	2.41	4.67
V3	4.63	0.736	0.7758	0.7947	5.41	7.98
V4	5.18	0.770	0.7803	0.7999	1.34	3.88
V5	5.72	0.739	0.7846	0.8075	6.18	9.27
V6	6.35	0.712	0.7704	0.7917	8.20	11.19
V7	6.92	0.743	0.7918	0.8155	6.57	9.75
V8	7.59	0.787	0.8014	0.8282	1.83	5.23

The absolute maximum error between the numerical and experimental results is 8.2% for $\zeta = 0.05$ and 11.19% for $\zeta = 0.01$, both in test V6. Though not shown here, the absolute maximum error of test V6 for $\zeta = 0.1$ is 7.52% and for $\zeta = 0.2$ is 8.08%. Based on that, one can estimate that the approximate damping ratio of that experiment is close to 0.1(not necessary. Damping is not the only factor causing the difference). In general, the measured response is smaller than the numerical results and does not appear to vary significantly over the speed range, while the numerical results monotonically and slightly increase with speed except at test V6. One fact that can be observed from both the experimental and numerical analysis is that the test V6 gives the smallest response among all the tests. This probably is an example of cancellation (Yang et al., 1997).

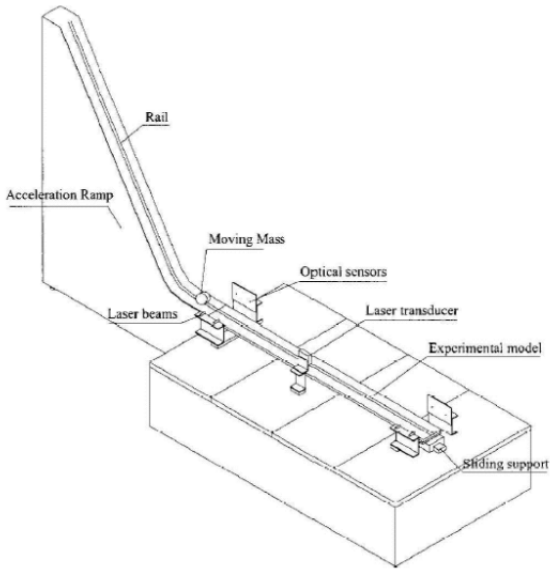

Figure 3.4: Experimental model

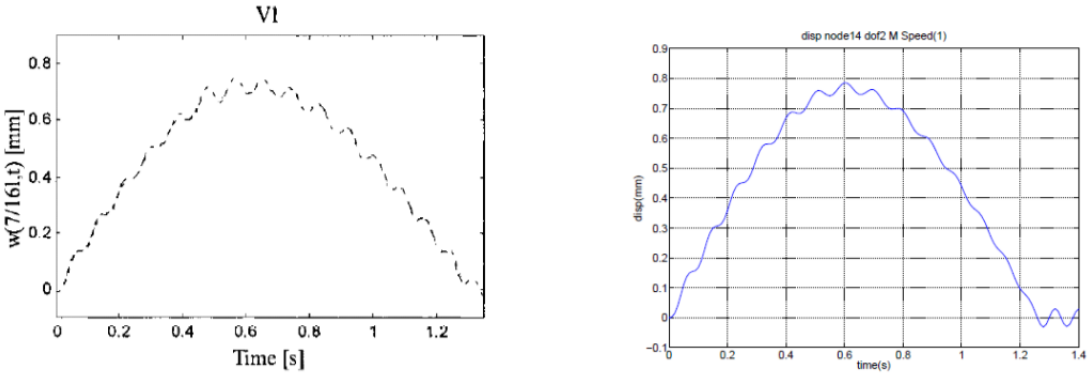

Figure 3.5: Experimental (left) and Numerical (right) results for entrance speed V1

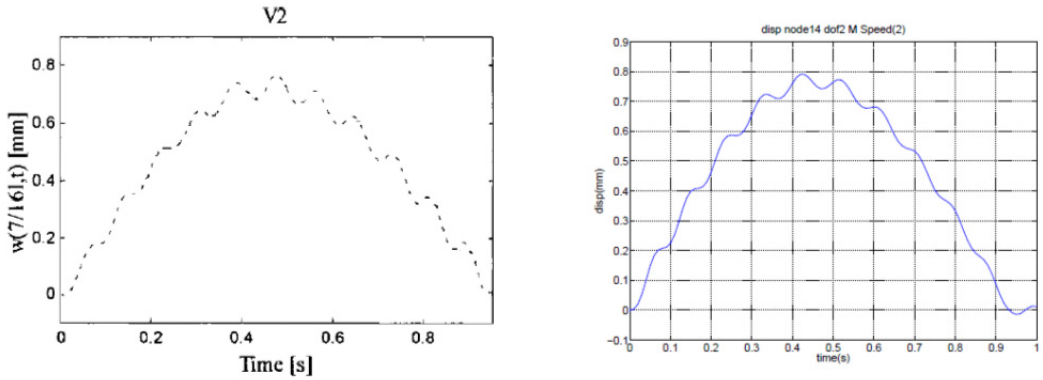

Figure 3.6: Experimental (left) and Numerical (right) results for entrance speed V2

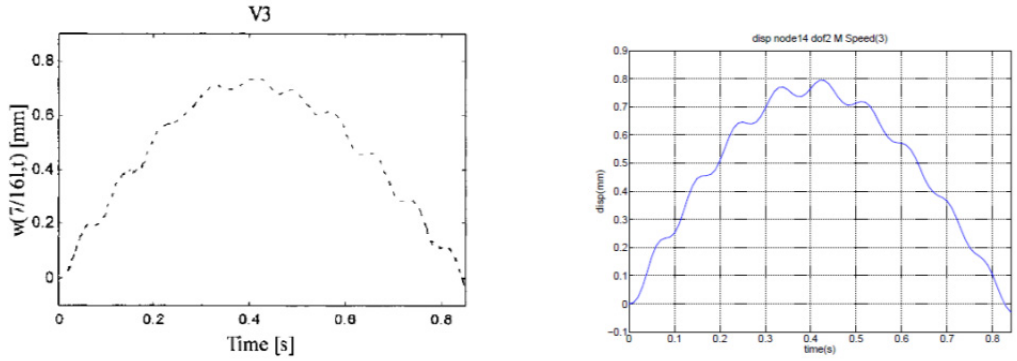

Figure 3.7: Experimental (left) and Numerical (right) results for entrance speed V3

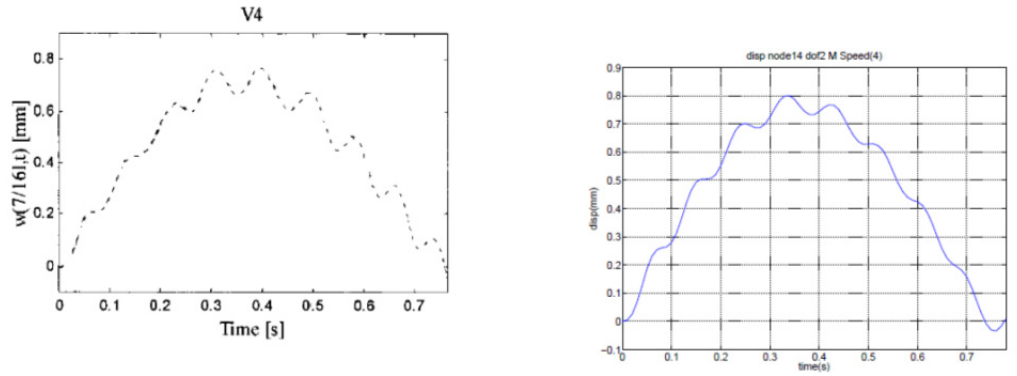

Figure 3.8: Experimental (left) and Numerical (right) results for entrance speed V4

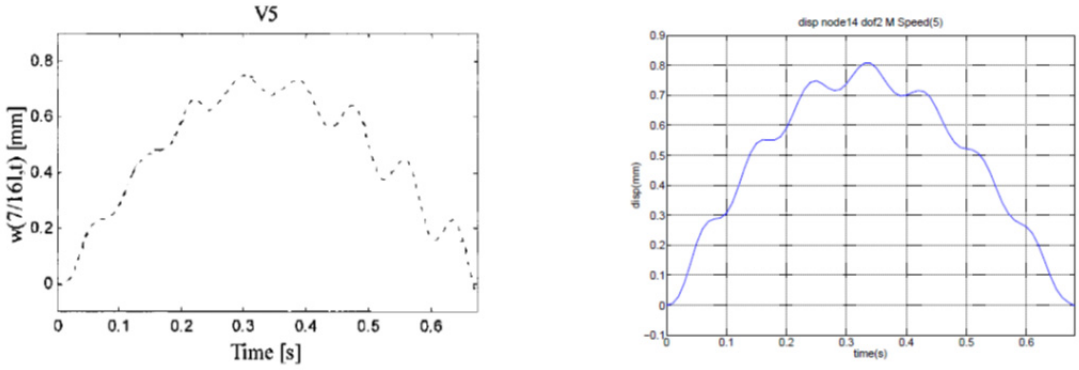

Figure 3.9: Experimental (left) and Numerical (right) results for entrance speed V5

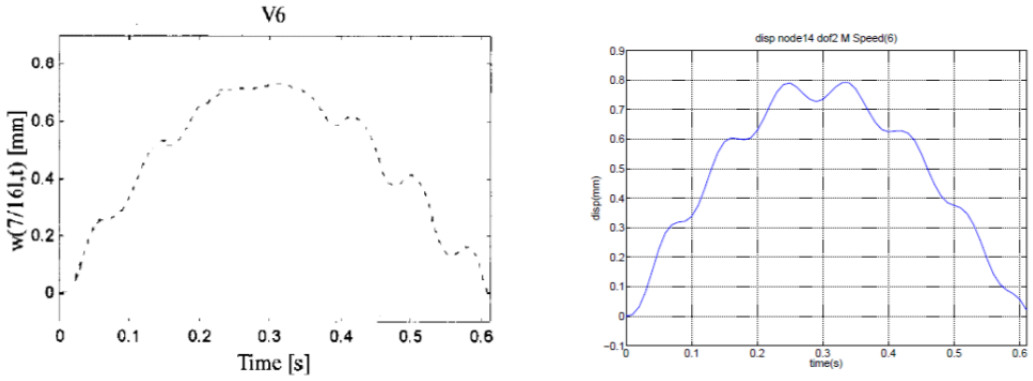

Figure 3.10: Experimental (left) and Numerical (right) results for entrance speed V6

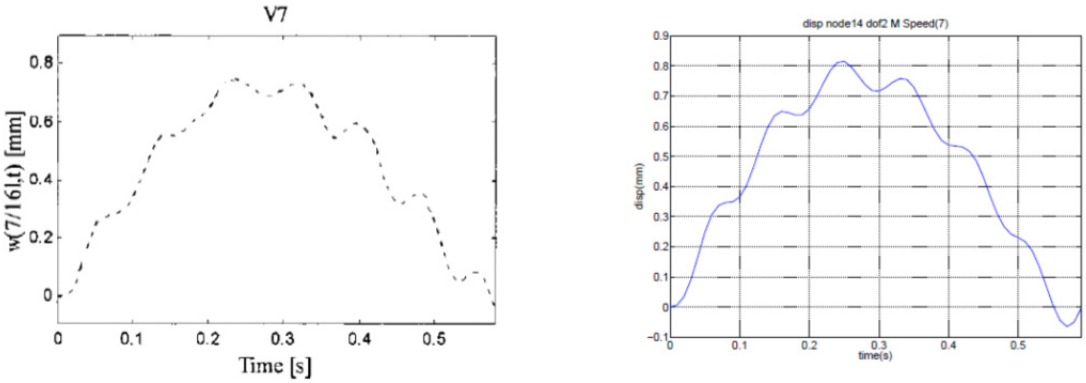

Figure 3.11: Experimental (left) and Numerical (right) results for entrance speed V7

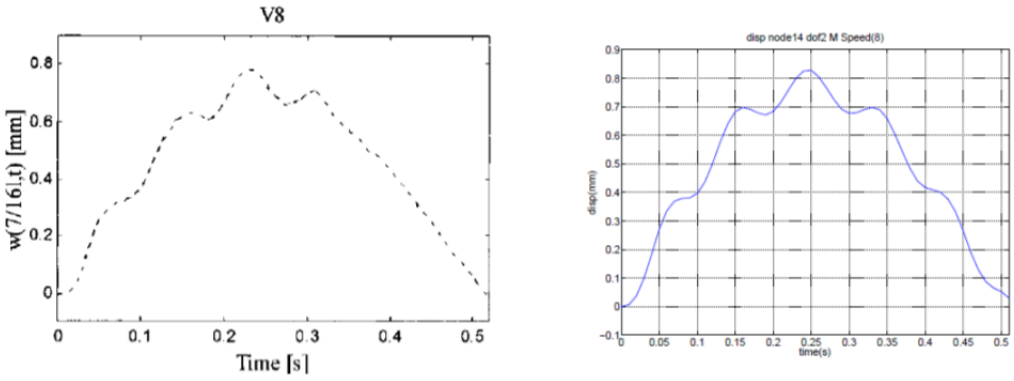

Figure 3.12: Experimental (left) and Numerical (right) results for entrance speed V8

3.3 EXPERIMENT OF A LONG-SPAN BOX GIRDER BRIDGE SUBJECTED TO MOVING VEHICLES

This experiment is performed on a real long-span bridge across the Han River in Seoul, South Korea (Lee and Yhim, 2005), as shown in Figure 3.13.

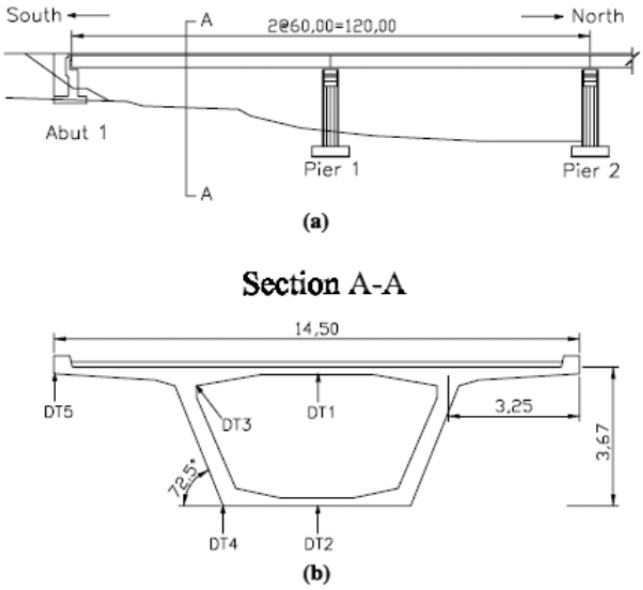

Figure 3.13: Elevation and the cross section of the box girder bridge

There are two spans of the bridge with length L = 60m for each, the density of the bridge is $\mu = 2.5kN/m^3$, Young's modulus E = 2.88GPa. The bridge consists of typical box sections. The equivalent moment of inertia of the section is $I = 37.25m^4$. The moving vehicle has a weight of 24.0kN. In the experiment, the dynamic responses at the cross section are measured in five locations. The measured results from location DT2 at the center of the span with speed from 20km/h to 60km/h are compared with the current method, as shown in Figures 3.14-3.18.

As shown in the figures, the results obtained by the current method are in good agreement with those from experiments. Different speeds of the vehicles have a small influence on both the experimental and numerical results. This is likely because the bridge is very long compared with the last example where the influence of the vehicle speed is perceptible. A similar phenomenon can also be observed; the absolute maximum displacements obtained from the experiments are smaller than the numerical results.

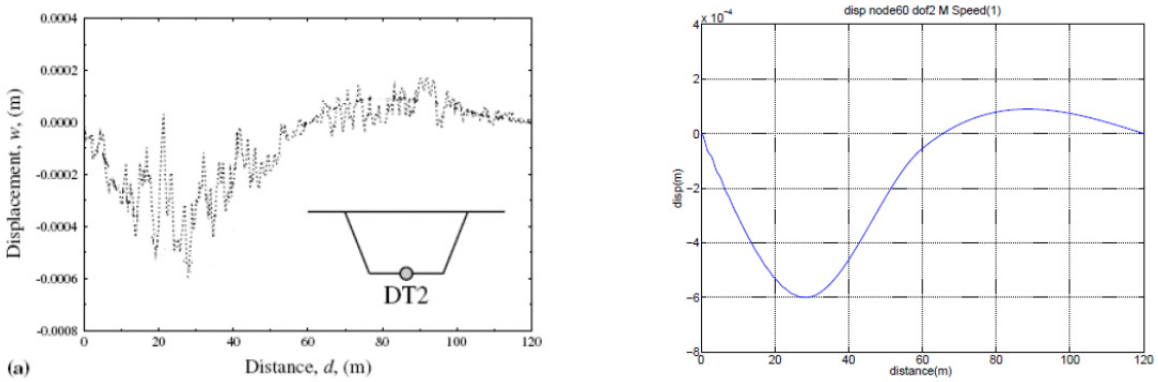

Figure 3.14: Experimental (left) and Numerical (right) results for vehicle speed 20km/h

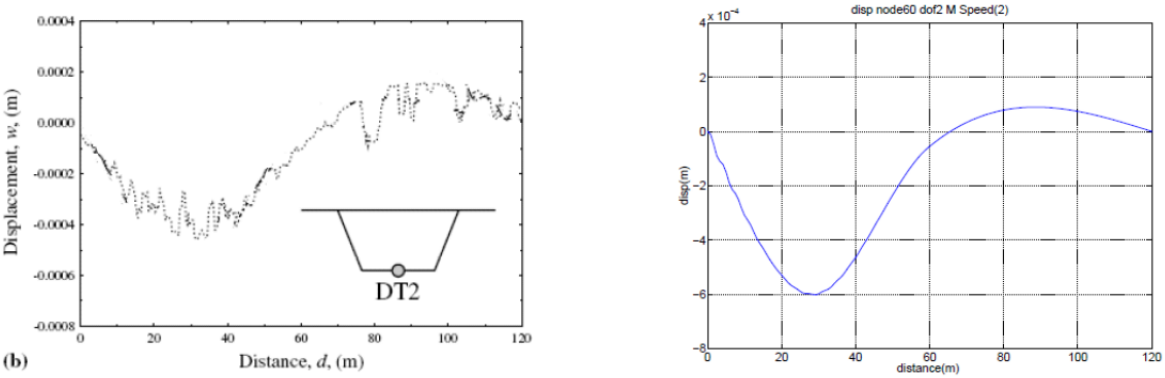

Figure 3.15: Experimental (left) and Numerical (right) results for vehicle speed 30km/h

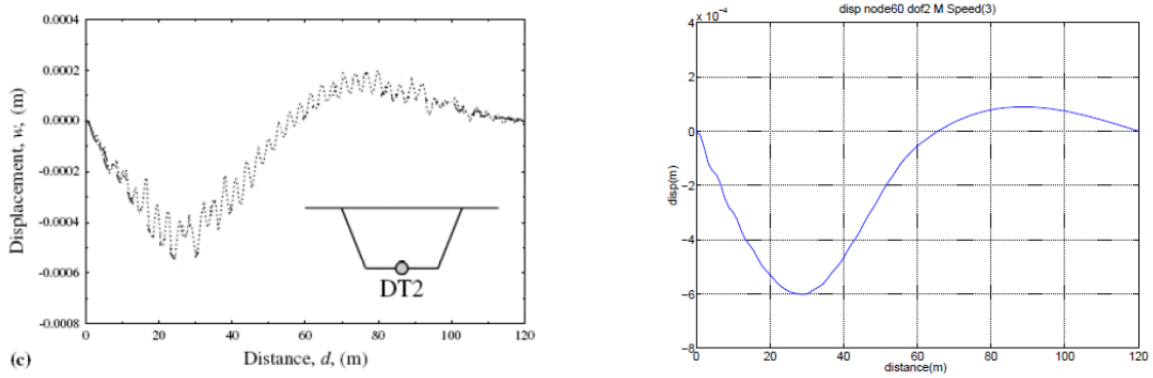

Figure 3.16: Experimental (left) and Numerical (right) results for vehicle speed 40km/h

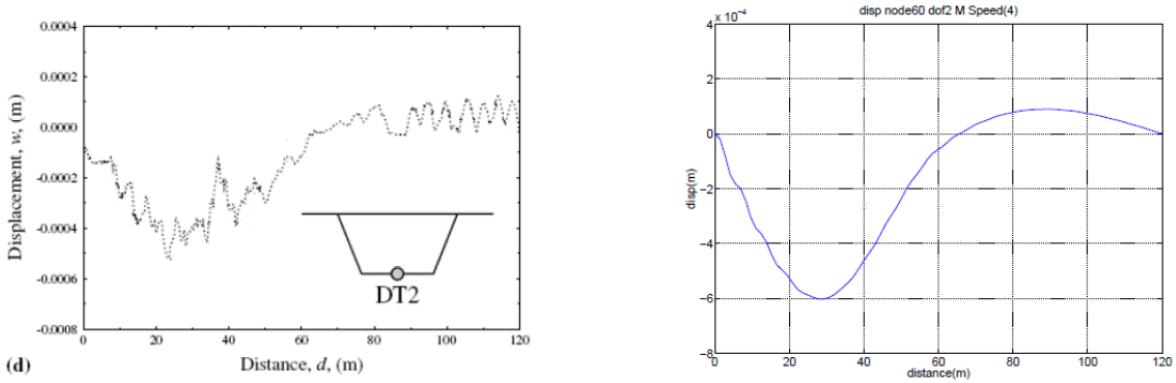

Figure 3.17: Experimental (left) and Numerical (right) results for vehicle speed 50km/h

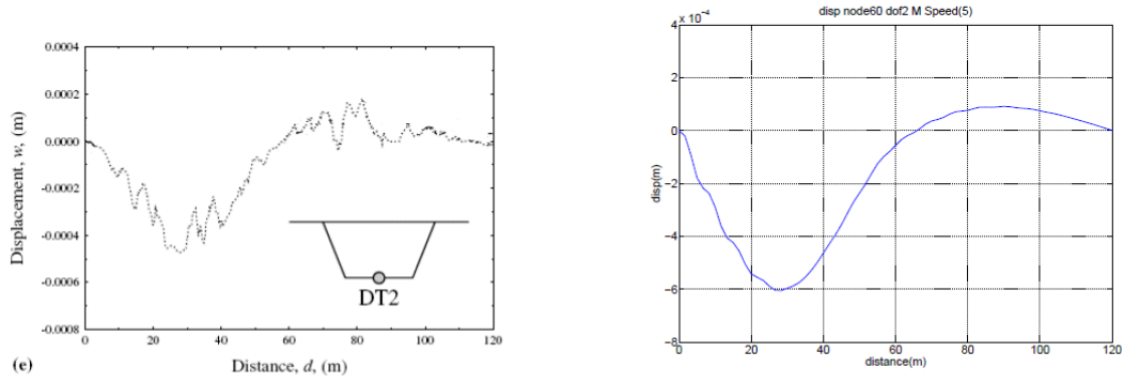

Figure 3.18: Experimental (left) and Numerical (right) results for vehicle speed 60km/h

3.4 THE LONG-SPAN BOX GIRDER BRIDGE SUBJECTED TO MOVING VEHICLES PLUS EARTHQUAKE

In addition to the last example, the earthquake excitation will be considered in this example. The ground-motion records of the Kobe earthquake from the PEER strong ground-motion database (http://peer.berkeley.edu/products/strong_ground_motion_db.html) is applied at the bridge supports.

To address the influence of the moving vehicle, the maximum displacements of the bridge induced only by an earthquake and by an earthquake plus live load are plotted in Figure 3.19. The vehicle speed varies from 20km/m to 60km/h. The left figure in Figure 3.19 is the plot considering only seismic loads, which is symmetric for the two spans because the bridge model is symmetric and the same ground motions are applied at the supports. As one can imagine, the peak value of the bridge displacements appears at the midpoint of the span. By considering the moving vehicle, the right figure in Figure 3.19 becomes unsymmetrical and has torsion of the 3D plot at lower speeds. In general, the results of the combined analysis are larger than the earthquake-only analysis. However, the responses around the mid-span for the lower-speeds portion are increased more than the high-speeds portion, while decreased around the supports. The different results between higher and lower speeds cause the torsion of the plot.

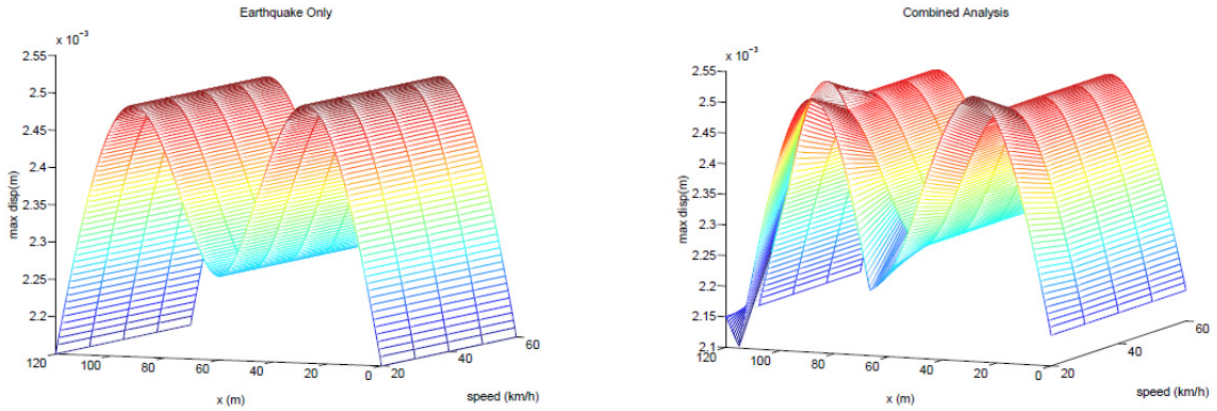

Figure 3.19: Plot of maximum displacements of the bridge induced by earthquake only (left) and combined earthquake and live load (right)

The maximum accelerations of the bridge induced only by an earthquake and by an earthquake plus live load are plotted in Figure 3.20. The maximum accelerations of supports remain constant since the same ground motions are used for all of the analysis. Considering that the moving vehicle doesn't significantly change the acceleration response of the bridge, the acceleration induced by a moving vehicle on a long-span bridge is trivial. There exist multiple resonant peaks along the bridge due to the higher modes that are caused by the earthquake excitations.

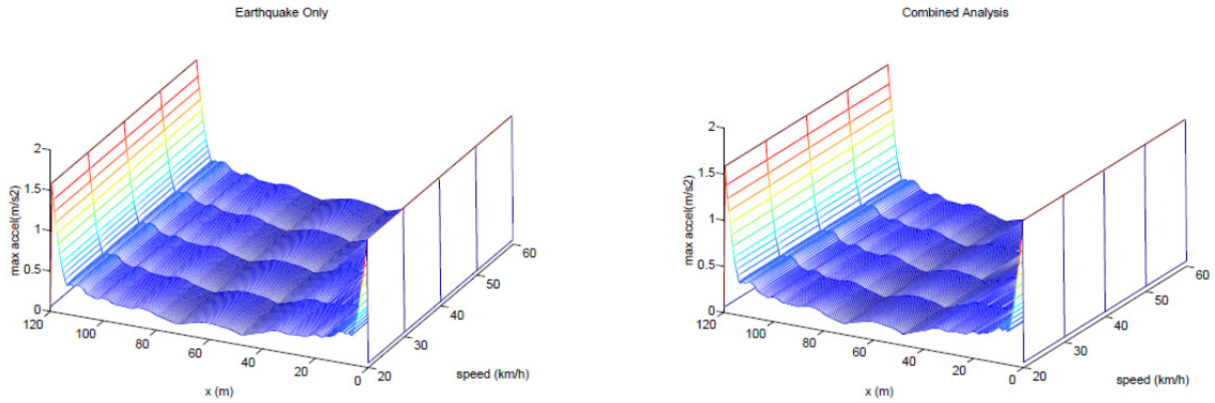

Figure 3.20: Plot of maximum accelerations of the bridge induced by earthquake only (left) and combined earthquake and live load (right)

4.0 CONCLUSIONS

A method for incorporating a vehicle model in regular finite element software was presented. An application for implementing this method was developed by using Tcl scripting language to glue (combine ?) the finite element analysis modules of OpenSees together to create customized procedures for the combined seismic and live-load analysis.

The idea of creating unmoved vehicle "containers" to hold "moving" properties makes it possible to incorporate vehicle models in a finite element program. Therefore, researchers can use existing tools of structural analysis and focus on the aspect of the structure. The examples show that the sprung mass model for the vehicle is sufficient if only structural responses are concerned. The feasibility and capability of the proposed method and application have been demonstrated in the examples by comparing with analytical and experimental results. It is concluded that the displacement responses of the bridge under moving vehicles are due to the first bridge mode for a short-span bridge. In the experiment's case, the results obtained from this method tend to be larger than those from the experiments. For a long-span bridge, vehicle speed has small influences on both the displacement and acceleration responses.

The use of Tcl scripting language makes the application easier to implement and flexible for multiple purposes. As an interpreted language, Tcl scripts do not need to be compiled to run. All the steps in the application are transparent to the user, who is free to experiment with different types of bridges or even different vehicle models.

5.0 REFERENCES

Bilello, C., Bergman, L. A., and Kuchma, D. (2004). “Experimental investigation of a smallscale bridge model under a moving mass.” Journal of Structural Engineering, 130(5), 799–804.

Broquet, C., Bailey, S. F., Fafard, M., and Bruhwiler, E. (2004). “Dynamic behavior of deck slabs of concrete road bridges.” Journal of Bridge Engineering, 9(2), 137–145.

Chopra, A. K. (2001). Dynamics of Structures: Theory and Applications to Earthquake Engineering. Prentice Hall, second edition.

Fryba, L. and Yau, J.-D. (2009). “Suspended bridges subjected to moving loads and support motions due to earthquake.” Journal of Sound and Vibration, 319, 218–227.

Ju, S.-H., Lin, H.-T., Hsueh, C.-C., and Wang, S.-L. (2006). “A simple finite element model for vibration analyses induced by moving vehicles.” International Journal for Numerical Methods in Engineering, 68, 1232–1256.

Kim, C.-W. and Kawatani, M. (2006). “Effect of dynamics on seismic response of steel monorail bridges under moderate ground motion.” Earthquake Engineering and Structural Dynamics, 35, 1225–124.

Kozar, I. (2009). “Security aspects of vertical actions on bridge structure.” International Journal for Computer-Aided Engineering and Software, 26, 145–165.

Kwasniewski, L., Wekezer, J., Roufa, G., Li, H., Ducher, J., and Malachowski, J. (2006). “Experimental evaluation of dynamic effects for a selected highway bridge.” Journal of Performance of Constructed Facilities, 20(3), 253–260.

Lee, S.-Y. and Yhim, S.-S. (2005). “Dynamic behavior of long-span box girder bridges subjected to moving loads: Numerical analysis and experimental verification.” International Journal of Solids and Structures, 42, 5021–5035.

Li, H., Wekezer, J., and Kwasniewski, L. (2008). “Dynamic response of a highway bridge subjected to moving vehicles.” Journal of Bridge Engineering, 13(5), 439–447.

Lu, F., Lin, J., Kennedy, D., andWilliams, F. (2009). “An algorithm to study non-stationary random vibrations of vehicle-bridge systems.” Computers and Structures, 87, 177–185.

Matsumoto, N., Sogabe, M., Wakui, H., and Tanabe, M. (2004). “Running safety analysis of vehicles on structure subjected to earthquake motion.” QR of RTRI, 45(3), 116–122.

Mazzoni, S., McKenna, F., H.Scott, M., and L., G. (2009). Open System for Earthquake Engineering Simulation User Command-Language Manual. Pacific Earthquake Engineering Research Center, University of California, Berkeley.

Nassif, H. H., Liu, M., and Ertekin, O. (2003). "Model validation for bridge-road-vehicle dynamic interaction system." Journal of Bridge Engineering, 8(2), 112.

Sogabe, M., Ikeda, M., and Yanagisawa, Y. (2007). "Train-running quality during earthquakes and its improvement for railway long span bridges." QR of RTRi, 45(3), 183.

Welch, B. B. (2000). Practical Programming in Tcl and Tk. Perntice Hall, Upper Saddle River, NJ.

Xia, H., Han, Y., Zhang, N., and Guo, W. (2006). "Dynamic analysis of train-bridge system subjected to non-uniform seismic excitations." Earthquake Engineering and Structural Dynamics, 35, 1563–1579.

Yang, Y.-B. and Wu, Y.-S. (2001). "A versatile element for analyzing vehicle-bridge interaction response." Engineering Structures, 23, 452–469.

Yang, Y.-B. and Wu, Y.-S. (2002). "Dynamic stability of trains moving over bridges shaken by earthquakes." Journal of Sound and Vibration, 258(1), 65–94.

Yang, Y.-B., Yau, J.-D., and Hsu, L.-C. (1997). "Vibration of simple beams due to trains moving at high speeds." Engineering Structures, 19(11), 936–944.

Yau, J. and Fryba, L. (2007). "Response of suspended beams due to moving loads and vertical seismic ground excitations." Engineering Structures, 29, 3255–3262.

P.O. Box 751
Portland, OR 97207

OTREC is dedicated to stimulating and conducting collaborative multi-disciplinary research on multi-modal surface transportation issues, educating a diverse array of current practitioners and future leaders in the transportation field, and encouraging implementation of relevant research results.