Full Length Research Paper

Development of two-dimensional solver code for hybrid model of energy absorbing system

Z. Ahmad[1]* and J. Campbell[2]

[1]Faculty of Mechanical Engineering, Universiti Teknologi Malaysia, 81300, Johor Bahru, Malaysia.
[2]School of Engineering, Cranfield University, Cranfield, Bedfordshire MK43 0AL, United Kingdom.

This paper treats the dynamic and energy absorption responses of nonlinear spring-mass system using a developed two dimensional solver code. The influence of spring stiffness and various mass configurations on the impact response was investigated using validated models. Results are quantified in terms of important impact response and indicate that an assembly of spring-mass system can be employed to examine the impact response of energy absorption system prior to the detailed numerical analysis and experiment. The developed solver code will be useful for educational purposes to preliminarily understand the behavior and impact response of energy absorbing system without learning a piece of complicated nonlinear commercial finite element codes namely LS-DYNA. The developed code could facilitate the early stage of evaluation for impact applications.

Key words: Spring-mass, dynamic, solver, energy absorption, finite element, impact.

INTRODUCTION

For several decades, an increased emphasis has been placed on crashworthiness as a structural design requirement for occupant-carrying vehicles. Structural systems in vehicles are continuously subjected to increasing safety requirement since vehicles are exposed to multidirectional impact conditions namely frontal, rear, side and roll over impacts. Although, destructive testing provides the most reliable check of the structural performance in an impact event, trial and error approach is impractical as it is relatively expensive and time consuming. The most important decision in the development of a new crashworthy structure is made in the early stage of design so that significant economic gains can be achieved by using a reasonably accurate theoretical method to predict the relevant collapse properties (Deb et al., 2004; Sheh et al., 1992; Pifko and Winter, 1981; Deb and Ali, 2004; Wu and Yu, 2001). In order to meet crashworthiness criteria, it is indispensible

that an adequate crashworthiness evaluation method may be used as early as possible in the design process. In general, various types of reliable crash analysis using experimental, theoretical, hybrid, finite element and analytical models can be conducted to examine the performance of energy absorption system in terms of energy absorption capacity, deceleration effect, dynamic force experienced by occupants and so forth (Jones, 1989; Jones and Wierzbicki, 1983). There have been numerous studies on the crash analysis of vehicular system available particularly in using experimental and finite element models as a primary approach (Benson, 1992; Wu and Cheng, 1997). However, most of the investigation emphasize on the detailed analysis rather than on the preliminary design concept in the early stage of design. It has been established that a hybrid analysis can promote a simple way of analyzing the effectiveness of energy absorption system with minimum effort and

*Corresponding author. E-mail: azaini@fkm.utm.my. Tel: +60 7 553 4647. Fax: +60 7 556 6159.

computational time (Wu and Yu, 2001; Ruan and Yu, 2003).

Hybrid analysis integrates a component testing and a simplified finite-element-based analysis. This integration of experimental and analytical approach significantly reduces development time and it also permits the study of a much wider variety of designs compared with conducting experimental testing alone (Gouddreau and Hallquist, 1982). In addition, this method requires less computer time than finite element methods leading to shorten design cycles and reduce development costs as well as providing useful preliminary design information relatively quickly or gross estimates of structural response. Moreover, by using this method, design errors can be easier to rectify before producing a full-scale model. Consequently, simplified hybrid models enable a quick parametric study to identify an optimum distribution of crush components, thus setting priority design targets. Hybrid method is also known as combined experimental and numerical methods in which a structure is divided into a number of masses which is connected each other using subassemblies that are treated as beams or nonlinear spring elements (Pifko and Winter, 1981; Hofmeister, 1978). A hybrid approach relies on experimental data gained separately from the load-deflection curves of full scale tests of each structural component in conjunction with a theoretical model (Liaw et al., 1988; Kecman, 1997). The use of a relatively simple theoretical model aids better understanding of involved collapse phenomenon. If the load-deflection response (stiffness characteristic) of components (beam or nonlinear spring elements) is known, a single or multi degree of freedom mass-spring systems can be possibly developed to simplify a model of the structure without concern of the material used (Liaw et al., 1988). The load carrying capacity of a structure is assumed to be insensitive to the deformation rate (Kirkpatrick et al., 2000). This approach, however, may give a limited accuracy of the results. In a pioneer research on hybrid model, Herridge and Mitchell (1978) have modeled a vehicle collision with two dimensional crash model. It appears that the model may anticipate the behavior of the structures in a brief manner. Sheh et al. (1992) have developed a simple hybrid model, namely a lumped parameter (LP) model in order to simply perform a frontal vehicle crash simulation. The model mainly consists of lumped masses and non-linear springs for a system with a single degree of freedom and the experimental data were used to assist the development of models. It is evident that by using this model, acceptable levels of safety in passenger automobile can be initially analyzed. In another study, Liaw et al. (1988) have used three dimensional hybrid models consist of lumped masses, nonlinear spring and beam using computer KRASH program. The nonlinear properties of the corresponding components were obtained from the static crush test and then as an input for the simulation using the computer program KRASH.

The present study treats the dynamic and energy absorption responses of crashworthy system under two dimensional impact loading. Dynamic response and feasibility of component assembly in energy absorbing system has been examined by varying component masses, crush characteristic of individual components and impact velocity. A solver code developed by using MATLAB programming (Thomson, 1988; Venkataraman, 2002), validated using existing commercial finite element code LS-DYNA was employed to investigate the relative effect of varied parameter on the dynamic response of the energy absorption system. Overall the results demonstrate the advantages of using the developed code as an analysis tool aids for academic and industrial usages. The primary outcome of this study is a simple solver code for educational purposes to preliminarily understand the behavior and impact response of energy absorbing system without learning a piece of complicated nonlinear commercial finite element codes namely LS-DYNA. The developed code could facilitate the early stage of design evaluation for impact applications specifically in automotive and aviation industries.

DEVELOPMENT OF HYBRID SOLVER AND COMPUTATIONAL MODELLING

Lagrangian computational method

The main contribution of this paper is the development of solver code using MATLAB algorithm. This programming solver was developed in accordance with the explicit Lagrangian computational method (Benson, 1992; Gouddreau and Hallquist, 1982). The solutions are advanced in time using an explicit integration scheme and they must be resolved accurately in both space and time domains (Cook et al., 1989; Jonsén et al., 2009). Recently the development of Lagrangian hydrocode is continued for a wide range of application namely automotive and aviation industries. Fundamentally, Lagrangian hydrocode applies Lagrange formulation for spatial discretization (Hallquist, 2006). Spatial discreti-sation can be carried out in either the Lagrangian or Eulerian framework. This hybrid solver code is a simplified hydrocode thereby becoming easier and straightforward. In particular, the time derivatives are material and thus simple partials of Lagrangian frame where coordinates are assigned to material points. As iterated in Benson (1992), a general iteration of Lagrangian code is outlined as follows.

(i) Stress, pressure, hourglass forces at t^n in each element, calculate the force at the node and followed by acceleration calculation,
(ii) Integration of acceleration to get velocity at $t^{n+1/2}$,
(iii) Integration of velocity to get displacement at t^{n+1},

(iv) Integration of constitutive model for the strength from t^n to t^{n+1},
(v) Calculation of shock viscosity and hourglass viscosity from $u^{n+1/2}$,
(vi) Updating internal energy based on work done between t^n and t^{n+1},
(vii) Pressure is calculated from the equation of state,
(viii) New time step is calculated,
(ix) Advance the time and return to initial step.

where t^n and t^{n+1} are full time intervals while $t^{n+1/2}$ and $u^{n+1/2}$ are half time intervals.

From the above iteration, it should be noted that the present study has only calculated the force, thus ignoring fifth and seventh steps in consideration of the type of elements used and complicated calculation for pressure, shock and hourglass viscosity. Second, third, fourth, sixth, eighth and ninth steps were accordingly devoted in this solver code. For the fourth step, the evaluation of force deflection curve was performed rather than the integration of constitutive model for the strength. For clarity of the developed solver code, the subroutines used are outlined as follows.

(i) Set-up problem: Input variables, material models, material properties, define curves,
(ii) Initialize problem: Calculate stiffness, critical timestep, state initialization, initial length, element direction vector
(iii) Solution: Solver,
(iv) Plotting results: Displacement, velocity and acceleration.

The equations used in the solver are based on Lagrangian formulation, equation of motion and fundamental spring-mass system.

Central difference time integration

Computational method using Lagrange formulation can solve the position of the mesh at discrete points in time (Cook et al., 1989). The solution is advanced from t^n to time t^{n+1} without any iterations and the time step $\Delta t^{n+1/2}$. In order to minimize the storage required, the solution is stored for only one time t^n within the program and the initial solution of the time step is overwritten by the solution at the end of the step. The central different method was used to advance the position of the mesh in time and it is based on second order accurate central difference approximation. From the Taylor series derivation, the integration rule can be carried out. The integration rule for the Lagrange formulation is expressed by Equations (1) and (2). In the present study, discrete element was used and do not involve any element meshes.

$$x^{n+1} = x^n + u^{n+\frac{1}{2}} \Delta t^{n+\frac{1}{2}} \tag{1}$$

$$u^{n+\frac{1}{2}} = u^{n-\frac{1}{2}} + \dot{u}^n \Delta t^n = u^{n-\frac{1}{2}} + (F/m)\Delta t^n \tag{2}$$

Herein, x, u and $\dot{u}$ are the displacement, velocity and acceleration respectively. In Equation (2), F and m are denoted to force and mass, respectively.

Central difference method is prevalently used in explicit direct integration methods to integrate the equations of motion in time (Benson, 1992; Gouddreau and Hallquist, 1982; Cook et al., 1989). Equation of motion is evaluated at the central time. In this present solver, the damping matrix was ignored as the system is an undamped system. Thereby, the equations of motion can be expressed by Equations (3) and (4) for the linear problem and nonlinear problem, respectively. In those equations, M, K and f^{ext} are a group of matrices for mass, stiffness and external force, respectively.

$$[M]\{\dot{u}\} + [K]\{x\} = \{f^{ext}\} \tag{3}$$

$$[M]\{\dot{u}\} + [K(x)]\{x\} = \{f^{ext}\} \tag{4}$$

For linear problems, the stiffness matrix is constant in time while for the nonlinear problem the stiffness matrix is a function of displacement as shown in Equuation (4). The central difference algorithm calculates the acceleration and velocity at the full time intervals, t^{n-1}, t^n, and t^{n+1} and the velocity at the half time interval, $t^{n-1/2}$ and $t^{n+1/2}$. In addition, this algorithm has a finite stable time step (Hallquist, 2006; Livermore Software Technology Corporation, LS-DYNA Keyword User's Manual, 2003). The critical time step must be determined to keep the time step for stability. Theoretically, central difference method is a second order accuracy with a single evaluation. Therefore, it can save the computational cost compared to second order Runge-Kutta. It is evident that the chosen method is problem dependences. In general, direct integration method is more expedient for a complicated nonlinearity. However, central difference method is more preferable for nonlinear dynamic problems (Stronge and Yu, 1993).

Specifications of solver code

Geometry

In this study, discrete element and nodal mass were employed for the spring and component masses, respectively. This type of element can represent a large system when subjected to a gross motion. In addition, discrete element and masses provide a capability for modeling a simple spring-mass system and can be used for more complicated mechanism. It should be noted that the discrete element is deemed to be massless and
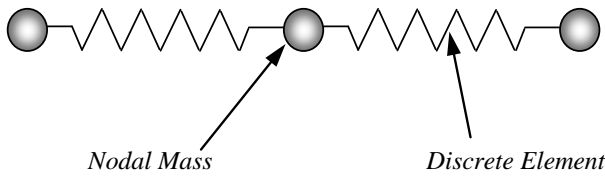
Figure 1. The discrete element and nodal masses for a spring-mass system.

Figure 2. Material model of nonlinear elastic.

consists of two nodes for each element. In particular, the nodal masses are specified at this unconstraint node or joining node to separate the element. Figure 1 shows the discrete elements and nodal masses for a spring-mass system consisting of two springs and three masses.

Force-displacement curve is an element property that can individually be included to represent the stiffness of the discrete element. By using discrete elements, various stiffness properties can accordingly be assigned to each element. The stiffness property is represented by stiffness matrix for a linear problem while load curve property is employed for a nonlinear problem. The nodal masses represent the crash component such as engine, bumper, sub floor, landing gear and other crashworthy components. Furthermore, element and nodal forces can easily be determined by using discrete element and nodal masses. A force of the local element coordinates can be expressed by the following equation.

$$F_{node} = \sum_n F_{element}$$
(5)

n = Elements attached to node.

Thereafter, kinematics responses namely acceleration, velocity and displacement for each nodal point mass can be computed directly with the time increment.

Material model

For the spring mass system, basic material models were defined excluding the strain rate effect and complicated equation of state. This programming solver code offers independent strain rate material models for the discrete elements such as linear elastic, nonlinear elastic and general nonlinear. Linear elastic discrete element is a typical force-displacement relations which is similar to Hooke's law as expressed in Equation (6) where k is the element's stiffness and Δl is the change of element's length.

$$\hat{F} = k\Delta l$$
(6)

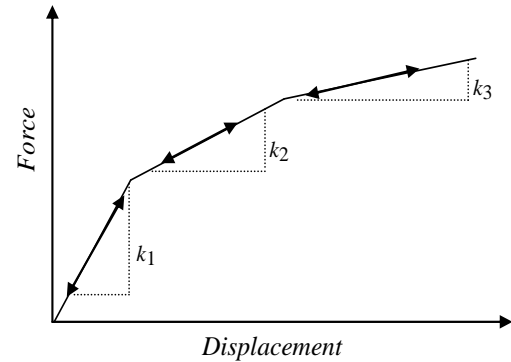Thus, the linear elastic describes the system in which the force increases linearly with the displacement. The spring element remains in a linear elastic region without yielding when subjected to loading. Nonlinear elastic discrete element exhibits a nonlinear behaviour with varying stiffness properties. Nonlinear elastic model expresses a tabulated force-displacement relation as in Equation (7) where $k(\Delta l)$ is the tabulated stiffness relation depending on the total change of element's length.

$$\hat{F} = [k(\Delta l)](\Delta l)$$
(7)

In particular, when a nonlinear elastic element starts to be loaded, it follows the loading path with varying stiffness properties. It remains in loading paths without yielding. For an unloading case, a nonlinear element is unloaded back along the loading paths. Figure 2 depicts the behaviour of the nonlinear elastic material model under loading and unloading cases.

General nonlinear material model is a primary material model of this study. This material model can offer better solution for the crash analysis. In many real situations, structure always responses under a nonlinear behaviour and involves in numerous nonlinear problems. In general, a nonlinear problem is inherently more complex to be analyzed than linear problems. It is worth noting that the model can simulate the response of structural components with the inclusion of independent and nonsymmetrical linear loading and unloading paths including the initial yield forces. It is noted that a general nonlinear element yields in either tension or compression. For instance, if the element is subjected to tension, the discrete element loads and unloads along the loading or unloading path as long as the load does not reach a yield point. Upon exceeding yield point and at all subsequent time, the element follows the specified loading and unloading curves. In addition, the displacement origin of the unloading curve is arbitrary and it is shifted to the present displacement where the element starts to unload. Then, the element reloads along the unloading path until it reaches the new updated yield point and follows the loading curve subsequently as demonstrated in Figure 3.
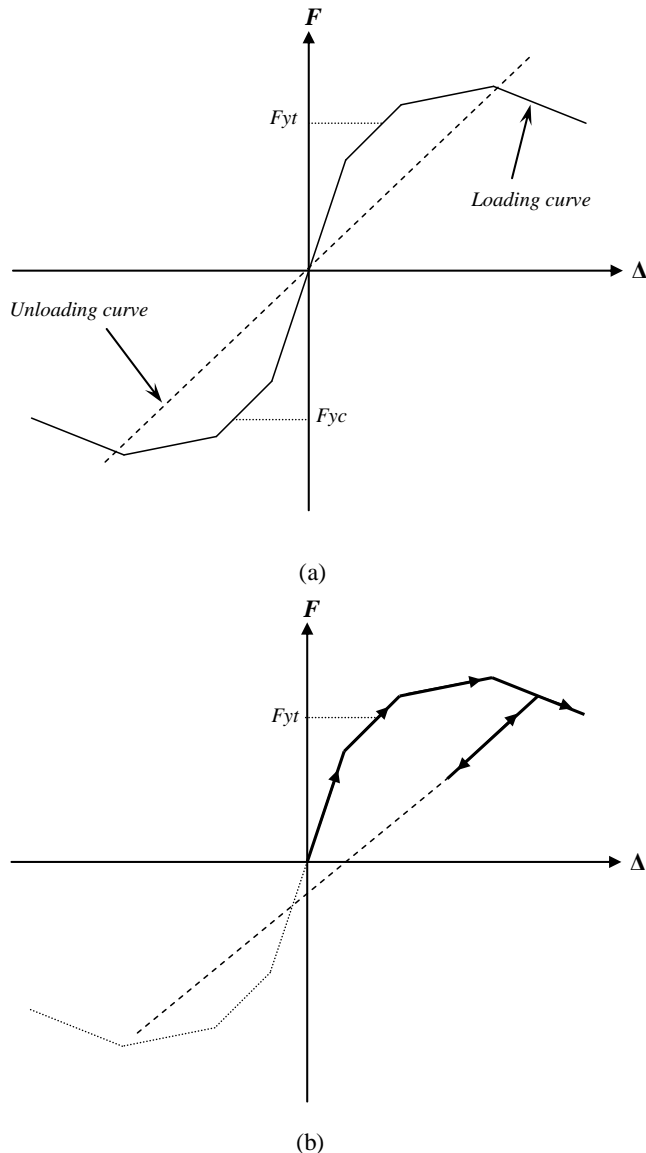
Figure 3. Material model of general nonlinear (a) Loading and unloading curve (b) Element yields in tension.

Initial, boundary and loading condition

In the programmed solver code, initial conditions are necessary in order to determine the kinematic behaviour and internal forces. The initial conditions required are initial velocity and initial coordinate. An initial velocity is needed to evaluate the motion (acceleration or deceleration) of the nodal masses. Alternatively, an initial acceleration can also be specified for the same purpose. Moreover, an initial coordinate is defined in the solver for capturing the translational motion of the element due to the change of the coordinate.

The stationary nodal masses were fully constrained in translation. The present solver also includes the prescribe motion boundary condition. This boundary condition can be devoted to a prescribe motion for nodal motion such as displacement, velocity and acceleration imposed as a prescribed function of time. This is convenient when individual or a set of nodal masses have a required velocity or acceleration time history.

There are only two types of loading conditions that have been specified in the solver namely, a concentrated load and a load with the inclusion of time variation (defined load curve). Any specified load curve has a tabulated force time relation.

Solver output

The present solver code is capable of providing beneficial design information which is pertinent to the structure crash analysis. A visualized output namely time-history plotting can interactively be presented to give a basic understanding of structure behaviour. Time history plot comprises of coordinate-time, displacement-time, velocity-time, acceleration-time and force-time. From those plots, the outputs are integrated to produce force-displacement curve. It is well known that force-displacement curve is dispensable in measuring the crush and energy absorption capacity of the system. In addition, this solver can also produce the kinetic and internal energy outputs. Therefore, the amount of dissipated energy can be evaluated for various spring-mass assemblies. A nodal mass contributes a kinetic energy while a spring discrete element contributes an internal energy.

Description of programming solver

The spring-mass system represents an assembly of crash component in structural analysis. The system consists of nodal masses, linear spring, nonlinear spring and general nonlinear spring. In developing a spring-mass system for crashworthiness analysis, it involves a topology study on the number of masses and springs to be connected and how are they connected. Mass distribution also needs to be justified in order to get an acceptable representation of the physical system.

Initially, a single spring with a pair of masses was used to understand the behaviour of the model and to validate linear, nonlinear and general nonlinear spring elements. For validation, the mass and stiffness property values were chosen for the single spring validation. Figure 4 shows a single spring-mass system in the preliminary model. Subsequently, two springs were duly used to evaluate the spring element behaviour due to the combination of linear elastic, nonlinear elastic or nonlinear elastic and general nonlinear material model. To ascertain whether the solver code was sufficiently accurate and feasible for analyzing more complicated
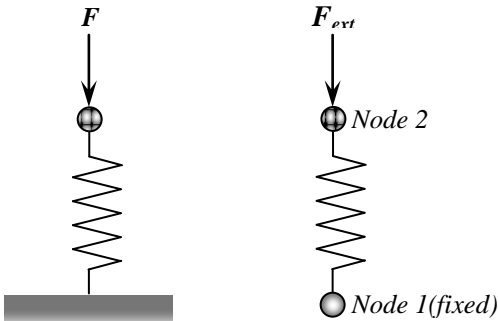
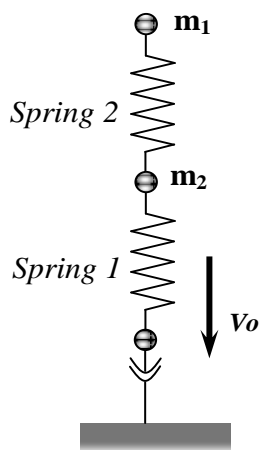**Figure 4.** A spring-mass system in the preliminary model.

Figure 5. A spring-mass system of an idealized helicopter sub floor vertical impact.
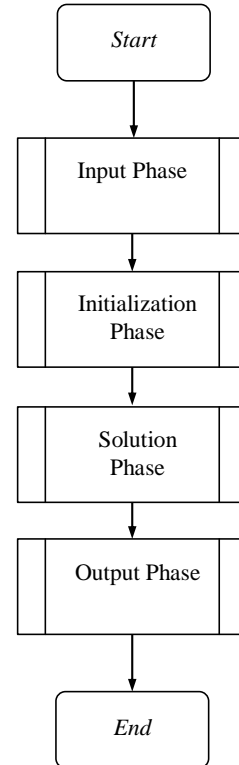

Figure 6. Programming flow for the solver code.

crash problems which consists of multi spring-masses, the code was programmed to perform analysis for multi spring-mass system representing a helicopter sub-floor structure as depicted in Figure 5. From Figure 5, the spring elements 1 and 2 represent a helicopter sub-floor and helicopter cabin structures, respectively. For the nodal masses, mass 1 represents a helicopter transmission and mass 2 represents a troop compartment.

In general, programming flow comprises several phases namely input, initialization, solutions and output phases. The detailed programming flow of the entire solver code can be illustrated in Figure 6.

VALIDATION AND DEMONSTRATION OF THE SOLVER CODE

The solver code was corroborated directly against the

numerical results obtained from LS-DYNA solver. Here, the validation is categorized for the single spring element and multi-spring element leading to the practical application of the structural crash.
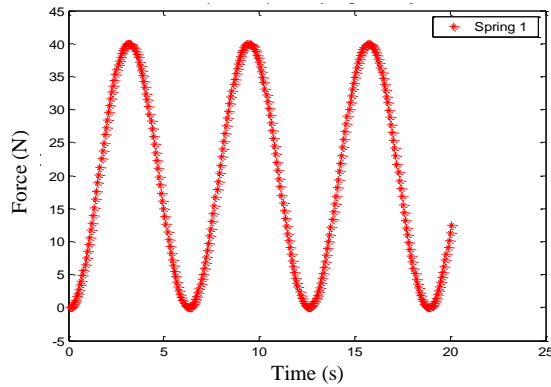
Single spring element

Single spring element was initially simulated to duly comprehend the spring element behaviour under pertinent loading condition. For the single spring element, LS-DYNA was not employed to validate the response. For validation, force-time, displacement-time and load-displacement curves are mainly of interest as those curves are sufficient to show the validity of the material model for a single spring element. Most importantly, the single spring element can present a reasonable trend prior to the subsequent validation of the multi-spring element.

Linear elastic

Figures 7 and 8 show the validation results for the single spring element. Obviously, the simulated curve trend shows curve response as expected when employing the
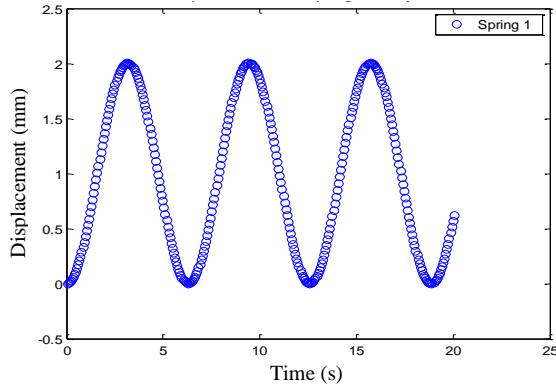
Figure 7. (a) Displacement-time (b) Force-time for a linear elastic spring element.

Figure 8. Force-displacement curve for a linear elastic spring element.

Table 1. Input data of force-displacement for nonlinear elastic material model.

Force-displacement curve	
Displacement (X coordinate) (mm)	**Force (Y coordinate) (kN)**
0.0	0.0
0.4	20.0
1.5	26.0

fundamental theory in which force and displacement oscillate accordingly to the corresponding solution time, thus resulting in proportional trend of a force-displacement curve. It is evident that the slope of the force-displacement curve is 20 N/mm which is similar to the defined spring stiffness, k in the solver code.

Nonlinear elastic

Nonlinear elastic material model can be defined by using the solver code for solving a simple nonlinear problem. Table 1 tabulates the input data of the force-displacement points. Figures 9 and 10 show accurate results plotted by the solver code for a force and displacement time history plot and a force-displacement curve respectively. Three points are indicated in Figures 9 and 10 in order to show loading and unloading paths within the solution time. In Fig. 9, the spring element is loaded rapidly to Point 1 and unloaded to Point 2. From this point onward, it is reloaded until it reaches Point 3. It is evident from Figure 10 that the material model of the solver agrees well with the postulated behavior of nonlinear elastic model shown in Figure 9. The results obviously give an accurate
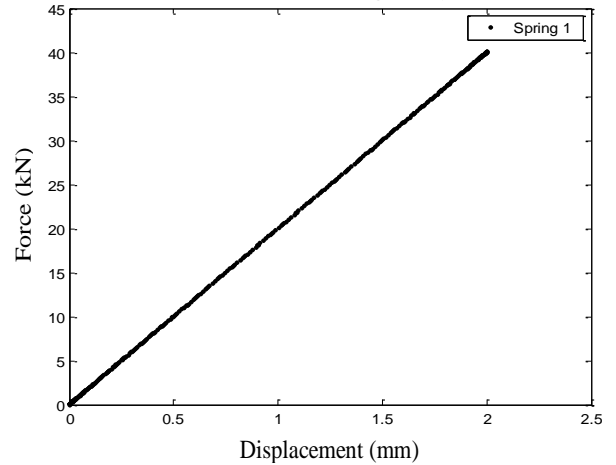
correlation between force and displacement time plot (Figure 9) with the force-displacement curve (Figure 10).
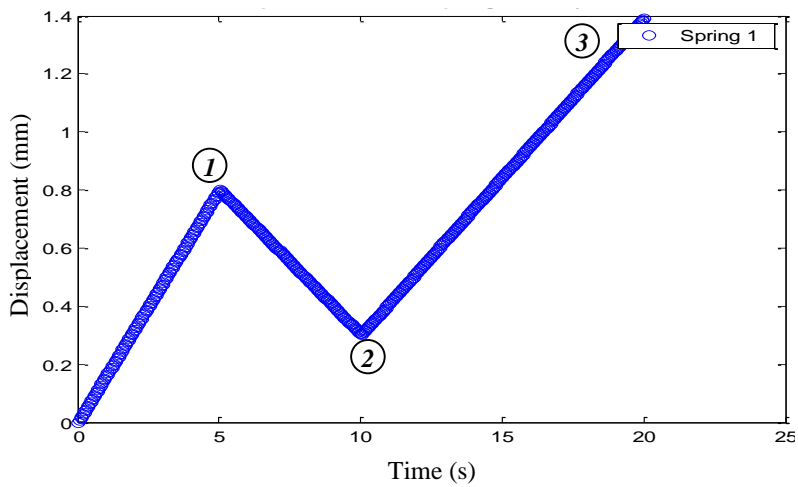
General nonlinear

The most complicated material model is a general nonlinear element. For this type of model, validation has been divided into two parts: Tension and compression loadings.

Tension loading: To ascertain whether the solver was sufficiently accurate for more complicated material model, it was virtually validated using force-displacement curve. Force-displacement curve for general nonlinear behavior is specified for both curves: Tension loading and unloading phases as indicated in Tables 2 and 3, respectively. In tension loading, States 0, 1 and 2 are involved in this solver code as demonstrated in Figure 11. Each state condition is as follows:

(i) State 0: Force has not reach tension and compression yield,
(ii) State 1: Force exceeded tension yield,
(iii) State 2: On unloading curve,
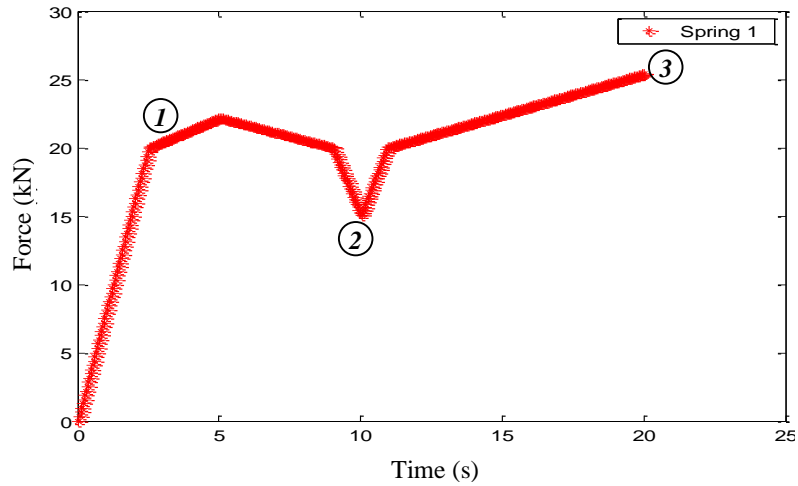
Figure 9. (a) Force-time (b) Displacement-time for a nonlinear elastic spring element.
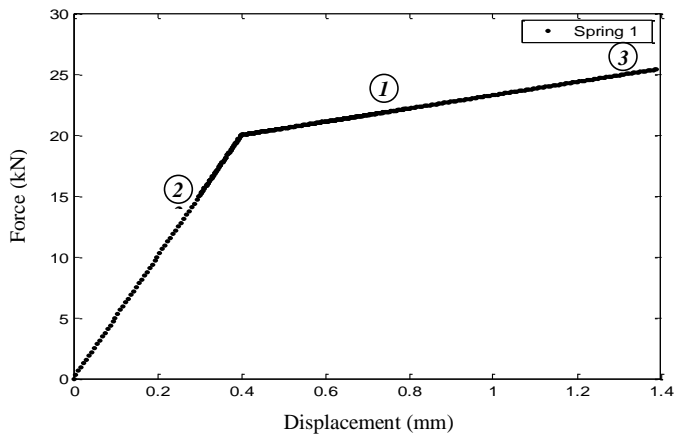

Figure 10. Force-displacement curve for a nonlinear elastic spring element.

Table 2. Input data of force-displacement curve for unloading phase in general nonlinear behaviour.

Force-displacement (Unloading curve)	
Displacement (X coordinate) (mm)	**Force (Y coordinate) (kN)**
0.0	0.0
0.1	1.0
7.0	70.0

(iv) State 3: Force exceeded compression yield.

Obviously, a reliable result is obtained from the direct comparison of the displacement-time plot and

Table 3. Input data of force-displacement curve for loading phase in general nonlinear behaviour.

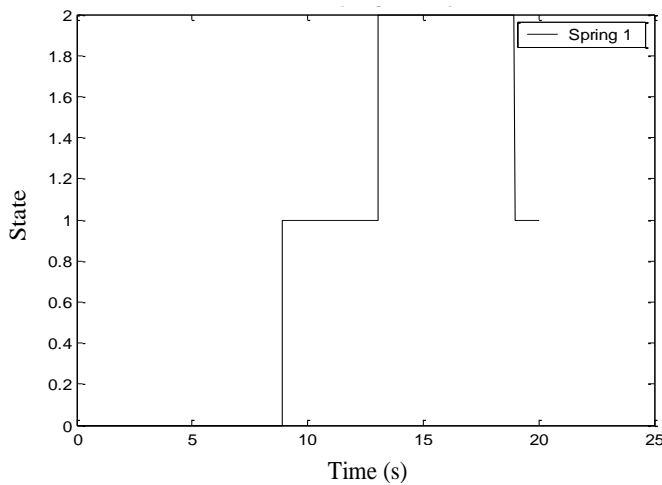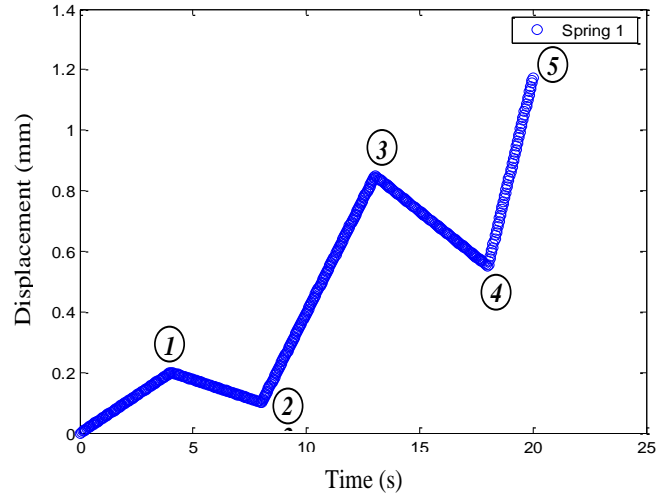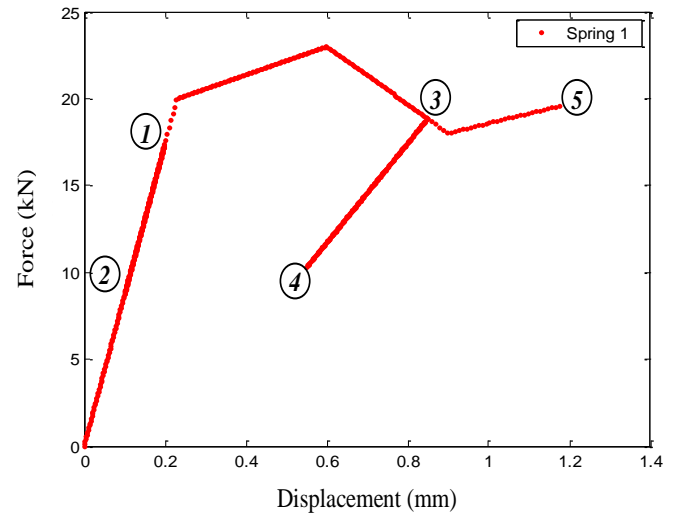| Force-displacement (Loading curve)	
Displacement (X coordinate) (mm)	Force (Y coordinate) (kN)
0.0	0.0
0.23	20.0
0.60	23.0
0.90	18.0
1.25	20.0
1.50	35.0

Figure 11. State-time plot for general nonlinear behavior under tension loading.

(a)

(b)

Figure 12. (a) Displacement–time curve; (b) Force-displacement curve for general nonlinear behavior under tension loading.

force-displacement curve (Figure 12). It should be noted that the element is subjected to general loading and unloading curves. In particular, from Point 1 to 2, the element is unloaded on the loading path as the force does not reach an initial yield force, 20 N/mm. However, the element is unloaded on the defined unloading curve upon a yield point as indicated at Points 3 to 4 in Fig.ure 12. From Points 4 to 5, the element is reloaded along the defined unloading curve as the displacement increases. Overall, the solver code shows a good capability of presenting a general nonlinear behavior of system. As such, this solver can perform the task for general nonlinear spring element when subjected to tension loading.

Compressive loading: Most of the crash structure is often exposed to compressive loading. Owing to this, this solver code was also validated under such loading condition. Table 4 tabulated input data of loading and unloading curve for the general nonlinear material model.
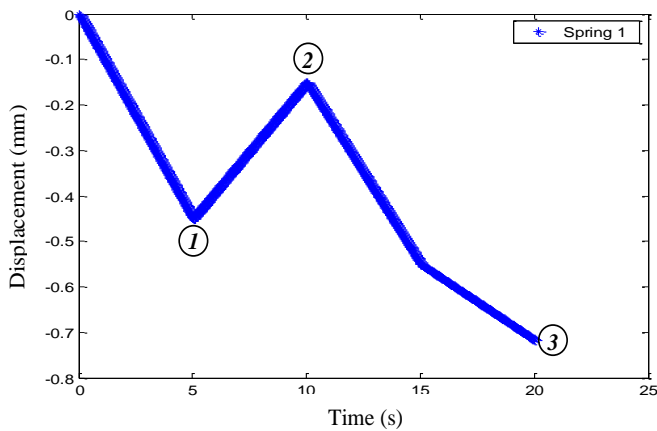
Figure 13 shows displacement-time and force-displacement curves for general nonlinear spring element when subjected to compression loading. It is evident that the curves show a good correlation between the loading and unloading conditions.

In particular, when compression load exceeds the initial yield compression, -20 N, it is unloaded on the unloading path rather than on the loading path as expected by using this solver code. This validation result is sufficient enough to show that the solver code is duly feasible and reliable for the element under compressive load. Hitherto, the solver code can subsequently be employed to a real two dimensional crash structure to show its usefulness and

Table 4. Force-displacement input data of loading and unloading curves for Spring 1 (cabin).

Displacement (X coordinate) (m)	Force (Y coordinate) (kN)
Force-displacement (Unloading curve)	
-0.006	-170000
0.0	0.0
0.006	170000
Force-displacement (Loading curve)	
-0.300	-1000000
-0.270	-500000
-0.225	-150000
-0.010	-150000
0.0	0.0

(a)
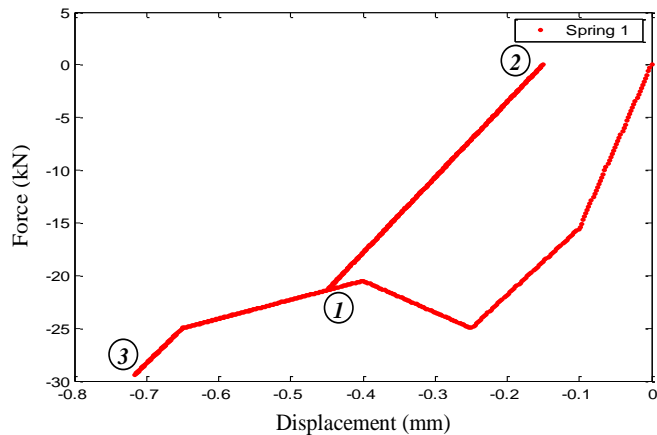
(b)

Figure 13. (a) Displacement–time curve (b) Force-displacement curve for general nonlinear behavior under compressive loading.

benefit in structural crash analysis.

Multi-spring element – two-dimensional crash solver code

For the multi-spring element, direct comparisons between the coordinate-time, velocity-time, acceleration-time and force-time curves obtained from existing finite element nonlinear code LS-DYNA and present codes are presented to show the accuracy of this solver code. For instance, a helicopter subfloor structure and frontal automotive structure crashes are demonstrated in this paper. This structural crash scenario can numerically be performed to comprehend the fundamental understanding of crashworthiness analysis since it just consists of two spring elements.

Helicopter sub-floor structure under vertical crash

The assembly of the multi-spring element shown in Figure 5 is used here. Helicopter sub-floor structures are divided into two main crash parts namely a helicopter cabin and an under-floor structure represented by spring element 1 and 2 respectively. The purpose of this crash analysis is to examine the dynamic behaviour of the structure under vertical impact load. Established force-displacement curve for the cabin and under-floor were included in the code prior to the analysis. Tables 4 and 5 show the force-displacement input data for the Spring 1 (cabin) and Spring 2 (under-floor) of helicopter sub-floor crash analysis (Liaw et al., 1988).

From the initial calculation, the undercarriage could absorb sufficient impact energy to reduce the ground impact velocity to 8.2 m/s. As such, this velocity value was defined as the initial velocity. Crash analysis has been performed by using LS-DYNA commercial code and the developed solver code. Figures 14 and 15 demonstrate the accuracy of time history plots obtained

Table 5. Force-displacement data of loading and unloading curves for Spring 2 (underfloor).

Displacement (X coordinate)(m)	Force (Y coordinate)(kN)
Force-displacement (Unloading curve)	
-0.009	-40000
0.0	0.0
0.009	40000
Force-displacement (Loading curve)	
-0.50	-76000
-0.223	-57200
-0.045	-70000
-0.009	-40000
0.0	0.0
10.0	5000

from the solver code compared with the results from commercial software LS-DYNA. In addition to this, a force-time curve is also directly compared and shown in Figure 16.

It is evident that the results obtained from the developed code, indicate good agreement between the two sets of results. The discrepancy becomes obvious upon reaching the solution time. The reason for this is due to the specified boundary condition. In LS-DYNA, rigid wall was defined as replacing translational constraint whereas in the solver code, the boundary condition was fixed. Thus, LS-DYNA code permits nodal mass 1 to bounce up and avoids any tension force occurred during the analysis. If this rigid wall is removed, the result would be better since the solver code does not take into account the contact algorithm. In order to overcome this problem, boundary conditions defined in LS-DYNA can be modified to suit with the non-contact solver code. Initially, the discrepancy has been anticipated due to the contact algorithm. As a recommendation for future research, this solver code can be improved by adding a contact boundary condition subroutine. Nevertheless, the time history plots (displacement, velocity and acceleration) present good agreement with LS-DYNA plot.

Automotive frontal structure crash

Frontal vehicle structure has a more complex spring-mass system. Typical results that produced using this solver code are presented to treat the capability of this solver code. The spring-mass model consists of 8 spring elements and 9 nodal masses as shown in Figure 17.

The important spring properties are accordingly defined to demonstrate the acceptable results of the automotive

crash. It is noteworthy that force displacement curves applied on the structure were applied arbitrary on the structure to represent the structure impacting rigid wall. The list of structural component included in this analysis is as follows.

(i) M1: Barrier mass,
(ii) M2: Bumper mass,
(iii) M3: Radiator interface mass,
(iv) M4: Engine mass,
(v) M5: Engine mounting component mass,
(vi) M6: Frontal wheel mass,
(vii) M7: Transmission component mass,
(viii) M8: Aft frame mass,
(ix) M9: Survival room mass.

In this crash analysis, the engine nodal mass and survival compartment mass have been chosen in order to evaluate dynamic behavior of the energy absorbing system. Figure 18 shows the coordinate-time, velocity-time and acceleration-time plots obtained from the solver code. More importantly, the results presented have an acceptable and reasonable finding as presented in Herridge and Mitchell (1978). It is evident that this developed solver code could successfully demonstrate the crash behavior and dynamic response of the multi complicated system in solving one dimensional crash analysis.

Overall, the present solver code has demonstrated its capability in simulating crash analysis involving multi-spring element: helicopter sub-floor structure. Although the code is not as accurate as LS-DYNA code, it can initially be employed to predict the impact response of the structure and to understand the behaviour of structure under impact loading condition.

Conclusion

The purpose of this study was to demonstrate the capability of solver code in performing two dimensional crash analysis of crashworthy system. It is evident that the solver code shows good correlation with the commercial finite element code. This satisfactory validation provides adequate confidence for promoting this code. Hence, this developed code may be employed as an educational tool. Most importantly, the code does not require learning complex pieces of nonlinear explicit code such as LS-DYNA in order to perform crash analysis. Moreover, no user manual is needed to be referred. Output response was quantified with respect to variations in the kinematic parameters and force-displacement thereby representing the energy absorption capacity of the system. The demonstrated multi-spring system shows that the dynamic and energy absorption responses of crashworthy system is significantly influenced by impact time, feasible component assembly
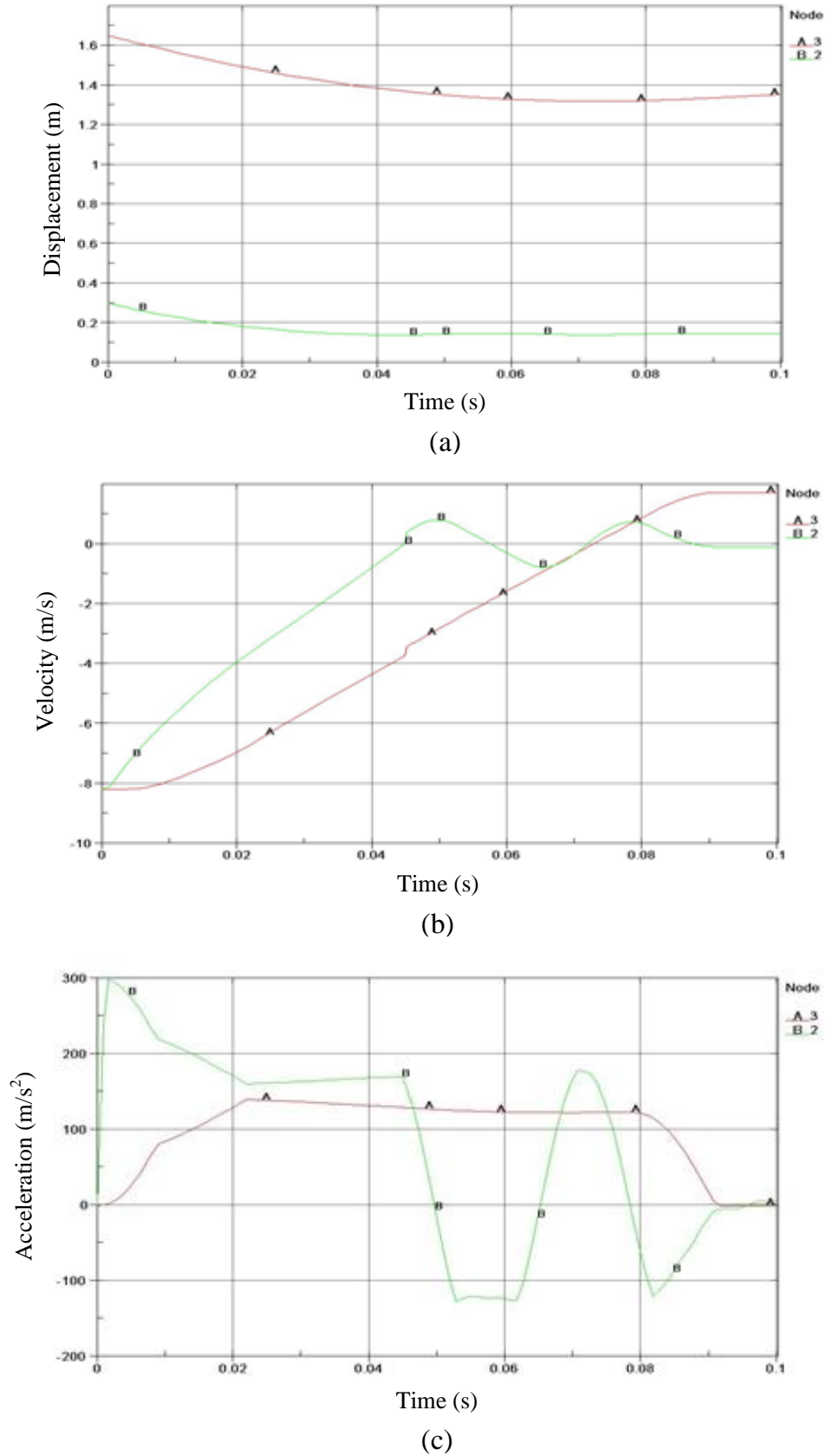
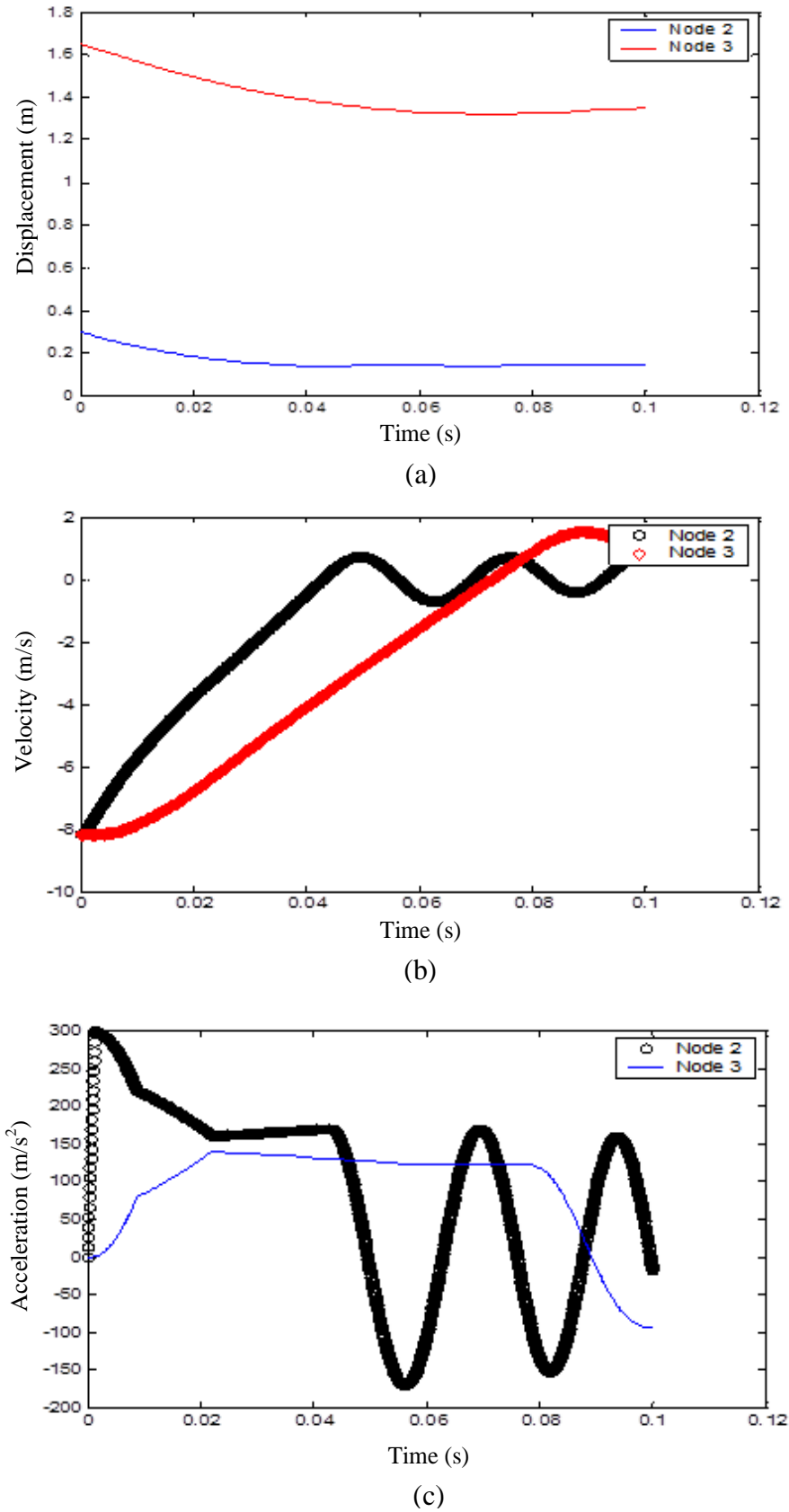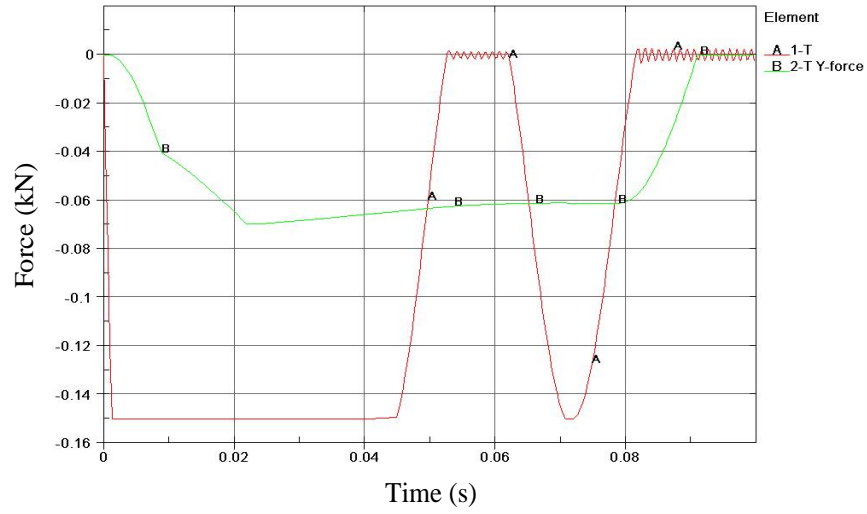**Figure 14.** (a) Displacement-time (b) Velocity-time (c) Acceleration-time of helicopter sub-floor crash (LS-DYNA*).*
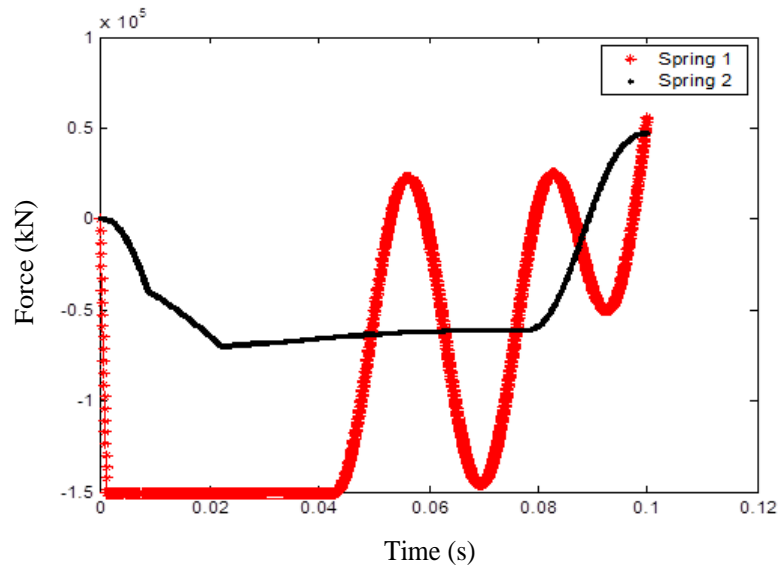
Figure 15. (a) Displacement-time (b) Velocity-time (c) Acceleration-time of helicopter sub-floor crash (solver code).

(a)

(b)

Figure 16. (a) Force-time curve (LS-DYNA) (b) Force-time curve (solver).
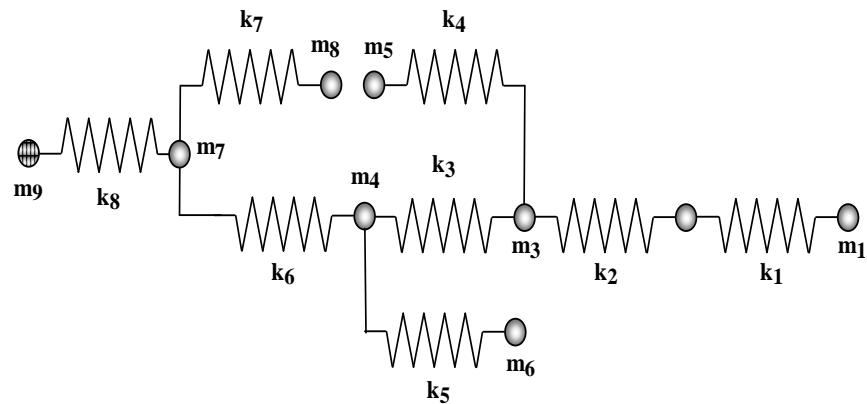

Figure 17. The spring-mass discrete elements for automotive frontal crash analysis.
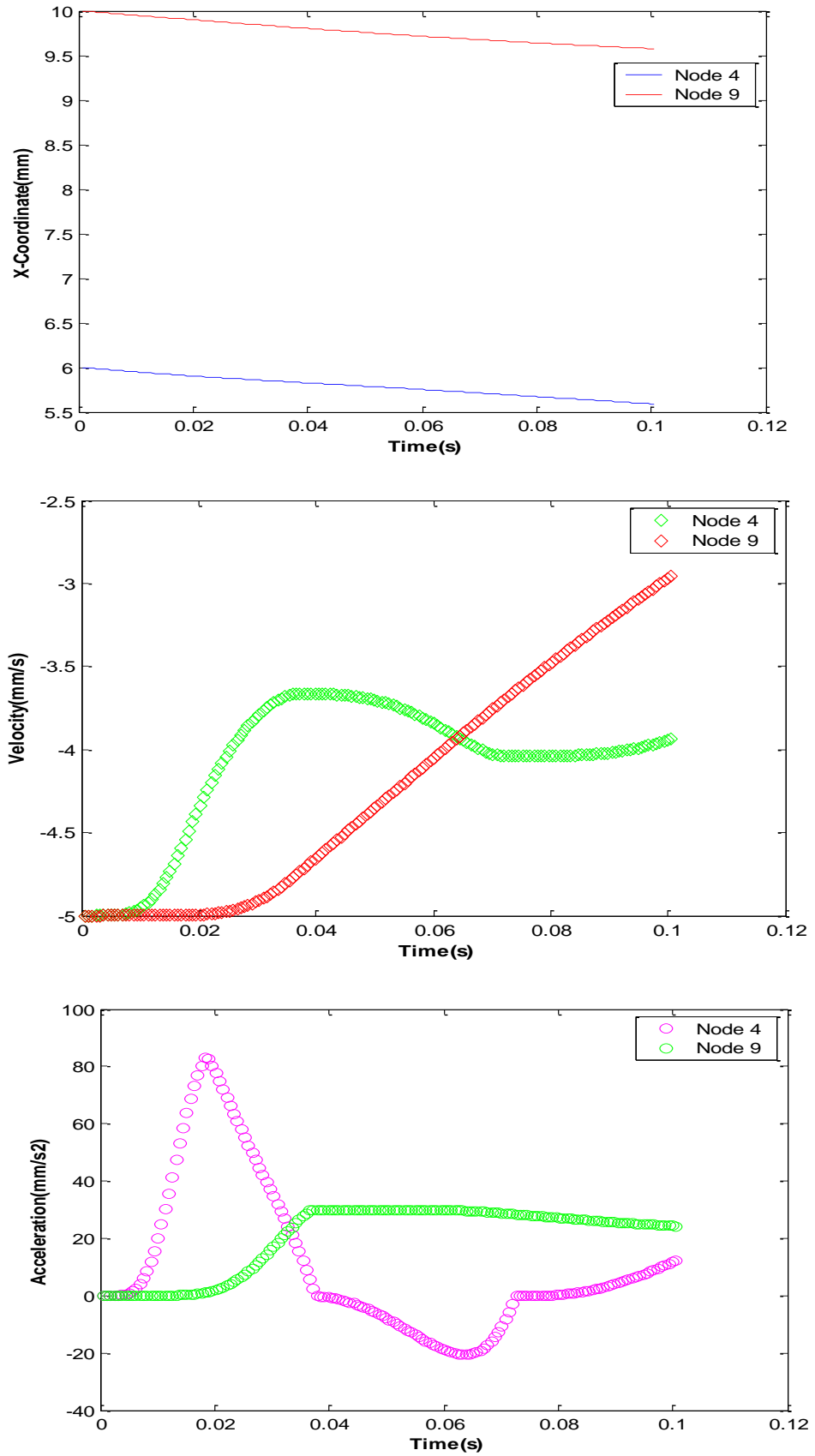
Figure 18. (a) Coordinate-time (b) velocity-time (c) acceleration-time of frontal vehicle crash.

and the stiffness of the component. It should be noted that this two dimensional solver is an innovative code to be used for the high educational level.

REFERENCES

Benson DJ (1992). Computational methods in Lagrangian and Eulerian hydrocodes. Comput. Methods Appl. Mech. Eng. 99:235-394.

Cook RD, Malkus DS, Plesha ME (1989). Concepts and applications of finite element analysis. Wiley Publication. New York.

Deb A, Ali T (2004). A lumped parameter-based approach for simulation of automotive headform impact with countermeasures. Int. J. Impact Eng. 30:521-539.

Deb A, Mahendrakumar MS, Chavan C, Karve J, Blankenburg D, Storen S (2004). Design of an aluminium-based vehicle platform for front impact safety. International J. Impact Eng. 30:1055-1079.

Gouddreau GL, Hallquist JO (1982). Recent developments in large-scale finite element Lagrangian hydrocode technology. Comput. Methods Appl. Mech. Eng. 33:725-757.

Hallquist JO (2006). LS-DYNA 3D: Theoretical manual. Livermore Software Technology Corporation. Livermore.

Herridge JT, Mitchell RK (1978). Development of computer simulation program for collinear car/car and car/barrier collisions. Am. Soc. Mech. Eng. P. 73.

Hofmeister LD (1978). Dynamic analysis of structures containing nonlinear spring. Comput. Struct. 8:609-614.

Jones N (1989). Structural impact. Cambridge University Press. Cambridge.

Jones N, Wierzbicki T (1983). Structural crashworthiness. Butterworth & Co (Publisher) Ltd: London UK.

Jonsén P, Isaksson E, Sundin KG, Oldenburg MM (2009). Identification of lumped parameter automotive crash models for bumper system development. Int. J. Crashworthiness 14(6):533-541.

Kecman D (1997). An engineering approach to crashworthiness of thin-walled beams and joints in vehicle structures. Thin-Walled Struct. 28:309-320.

Kirkpatrick SW, Simons JW, Antoun TH (2000). Development and validation of high fidelity vehicle crash simulation models. Int. Crash. Conf. 16:602-612.

Liaw JC, Walton AC, Brown JC (1988). Structural impact simulation using program KRASH. Cranfield: Cranfield Instit. Technol. pp. 69-80.

Livermore Software Technology Corporation, LS-DYNA Keyword User's Manual (2003). Livermore Software Technology Corporation. California.

Pifko AB, Winter R (1981). Theory and application of finite element analysis to structural crash simulation. J. Comput. Struct. 13:277-285.

Ruan HH, Yu TX (2003). Collision between mass-spring systems. Int. J. Impact Eng. 31:267-288.

Sheh MY, Reid JD. Lesh SM, Cheva M (1992). Vehicle crashworthiness analysis using numerical and experimental method. in International Conference on Vehicle Structural Mechanics and CAE. Warrendale: Society of Automotive Engineering Inc.

Stronge WJ, Yu TXL (1993). Dynamic models for structural plasticity. Springer.

Thomson WT (1988). Theory of Vibration with Application. Prentice Hall. Sydney.

Venkataraman P (2002). Applied optimization with Matlab programming. John Wiley & Sons Inc. New York.

Wu KQ, Yu TX (2001). Simple dynamic models of elastic-plastic structures under impact. Int. J. Impact Eng. 25:735-754.

Wu SR, Cheng J (1997). Advanced development of explicit FEA in automotive applications. Comput. Methods Appl. Mech. Eng. 149:189-199.