Three Point Bending Analysis of a Mobile Phone Using LS-DYNA Explicit Integration Method

Feixia Pan, Jiansen Zhu, Antti O. Helminen, Ramin Vatanparast
NOKIA Inc.

Abstract

In this article, the 3 point bending analysis of a mobile phone using LS-DYNA explicit integration method is discussed. Since there are a large number of contact pairs defined in the FEA model, and the FEA model is very large in a 3 point bending analysis of a phone, it is much more convenient to use the explicit method than the implicit method. However, using explicit procedure to a quasi-static analysis requires some special consideration. Since a quasi-static solution, is by definition, a long-time solution. It often requires an excessive number of small time increments. It is computationally impractical to conduct the simulation in its natural time scale. In real analysis, the quasi-static event is artificially accelerated by two approaches to reduce the computation time. One approach is to use mass scaling. Another approach is to increase the loading rate. These two approaches are closely related and should work together. If they are properly used, the speed of the analysis could be increased substantially without severely degrading the quality of the quasi-static solution. We discuss in this article how the loading rate and mass scaling factor affect each other, how to select proper values of these two parameters, and how to use these two approaches in the 3 point bending analysis of a mobile phone.

1. Introduction

The explicit solution method is originally developed to model high-speed impact events in which inertia plays a dominant role in the solution. Out-of-balance forces are propagated as stress waves between neighboring elements while solving for a state of dynamic equilibrium. Explicit time integration method has been used extensively for phone drop test simulation. In fact, the explicit method has proven to be valuable in solving static problems as well. For certain types of static problems, it is much more convenient to use explicit method than implicit method. The explicit method could more readily resolve complicated contact problems than the implicit method. In addition, as models become very large, the explicit procedure requires less system resources than the implicit procedure. However, applying the explicit dynamic procedure to a quasi-static problem requires some special considerations. Since a static solution is, by definition, a long-time solution, it is often computationally impractical to conduct the simulation in its natural time scale, which would require an excessive number of small time increments. To obtain an economical solution, the event must be accelerated in some way. The goal is to model the process in the shortest time period in which inertial forces remain insignificant.

There are two approaches to accelerate the event. One approach is to use mass scaling. Another approach is to increase the loading rate. It should be mentioned that these two approaches are closely related. They would affect each other. If these two approaches are properly used, the speed of the analysis could be increased substantially without severely degrading the quality of the quasi-static solution; the end result of the slow case and a somewhat accelerated case would be nearly the same. Otherwise, the event would be accelerated to a point at which inertia effects dominate, the solution would tend to localize, and the results would be quite different from the quasi-static solution.

In this article, we are going to discuss the use of explicit method to solve a 3 point bending problem of a phone. Since there are a large number of contact pairs defined in the FEA model, and the FEA model is very large in a 3 point bending analysis of a phone, it is much easier to use the explicit method than the implicit method. We start with examining how the loading rate and mass scaling factor affect each other, and how to select proper values of these two parameters with the aid of a simple mass spring system. Then we discuss in more detail the 3 point bending analysis of the phone.

2. Approaches to reduce the computation time in a quasi-static analysis

In using explicit method to solve a quasi-static problem, the quasi-static event is usually accelerated by mass scaling and loading rate scaling approaches. To efficiently use these approaches, it is important to understand how the loading rate scaling and the mass scaling would affect the performance of an explicit analysis. According to [1], the following ad hoc rules have been used to determine whether a quasi-static analysis is successful:

i) The kinetic energy of the deformed structure shall not exceed a small fraction (about 5%) of its internal energy throughout most of the time period of the explicit analysis.
ii) The ratio of the kinetic energy to the internal energy shall be less than 0.1% at the steady state.
iii) The time rate of change of the internal energy shall be negligible at the steady state.
iv) The maximum out-of-plane deformation shall reach a constant value at the steady state.

Essentially whether an analysis is quasi-static or not depends on whether the dynamic vibration terms are small enough or not in the response of the system. In what follows, we use a simple mass-spring system to discuss in turn the effect of loading rate scaling and mass scaling on the performance of a quasi-static analysis.

2.1 The effect of loading rate

Figure 1 shows a mass spring system and the loading profile on that system. The load is ramped to F_0 within t_f, and then keeps at the constant value of F_0.

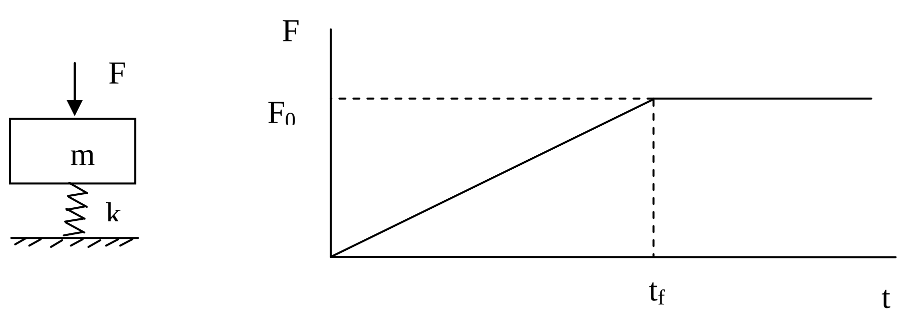

Figure 1. Mass spring system and the loading profile on that system.

The exact solution of the response of the mass spring system is as follows:

$$x(t) = \frac{F_0}{k}\left(\frac{t}{t_f} - \frac{1}{\omega_n t_f}\sin\omega_n t\right), \qquad 0 \le t \le t_f \tag{1}$$

$$x(t) = \frac{F_0}{k}\left[1 - \frac{2}{\omega_n t_f}\sin\omega_n\frac{t_f}{2}\cos\omega_n\left(t - \frac{t_f}{2}\right)\right], \qquad t \ge t_f\,, \tag{2}$$

where ω_n is the natural frequency of the system. The ratio of the kinetic energy to the internal energy is given by:

$$\frac{E_{kinetic}}{E_{internal}}(t) = \frac{mF_0}{2k^2 t_f t}\frac{(1-\cos\omega_n t)^2}{\left(1 - \frac{1}{\omega_n t}\sin\omega_n t\right)}, \qquad 0 \le t \le t_f\,, \tag{3}$$

$$\frac{E_{kinetic}}{E_{internal}}(t) = \frac{2mF_0}{k^2 {t_f}^2}\frac{\sin^2\omega_n\frac{t_f}{2}\sin^2\omega_n(t-\frac{t_f}{2})}{\left(1 - \frac{2}{\omega_n t_f}\sin\omega_n\frac{t_f}{2}\cos\omega_n(t-\frac{t_f}{2})\right)}, \qquad t \ge t_f\,. \tag{4}$$

From equations (1), (2), and (4), we find a necessary condition for the vibration terms in equations (1) and (2) be very small, and the ratio of kinetic energy to internal energy in equation (4) is less than a certain value (0.1%) at the steady state. This necessary condition is: the value of $\omega_n t_f$ be big enough. A good rule of thumb is to select t_f such that

$$\omega_n t_f \ge 20\pi\,. \tag{5}$$

This is equivalent to the relation of $\frac{t_f}{T} \ge 10$, where T is the natural period of the system [2]. This rule of thumb is also valid for a real application of complicated FEA model, except that ω_n should be replaced by the frequency of the 1st natural mode of the system.

According to equation (1), a sine wave is superimposed on the linear solution of the displacement response of the system. The frequency or period of the 1st natural mode of a real application could be obtained by running frequency analysis. It could also be estimated by a few trials of the explicit analysis, as will be discussed further in Section 3.1.

2.2 The effect of mass scaling

For a quasi-static analysis, the change in the mass of an object won't affect the deformation of that object if a solution converges. On the other hand, in an explicit analysis, the mass density of a material would affect the time step size of numerical integration. The central difference scheme, which is the most commonly used explicit algorithm, is only conditionally stable, the stability limit being approximately equal to the smallest time required for a sound wave to travel through any of the element in the mesh. That is, $\Delta t = \min\left(\frac{L_{\min}}{S}\right)$, where $L_{\min}$ is the smallest dimension in an element, and S is the speed of sound traveling through the element. It is well known that the speed of sound traveling through an element is proportional to $\sqrt{\frac{E}{\rho}}$, where E and ρ are the Young's modulus and mass density of the material, respectively. According to the above relations, artificially increasing the material density ρ by a factor of $\bar{f}^2$ decreases the wave speed by a factor of $\bar{f}$ and increases the stable time increment by a factor of $\bar{f}$. Therefore, we could efficiently reduce the computation time by increasing the mass density of those elements with relatively large E and/or small L_{min}. Both LS-DYNA and ABAQUS/EXPLICIT have feature to set a smallest time step size permitted for numerical integration [3,4]. If any elements in the FEA model could not satisfy this time step size limit, mass scaling would be done on these elements. In LS-DYNA, the parameter dt2ms in CONTROL_TIMESTEP card could be used to set the minimum time step size permitted in the analysis [3].

In simulations involving a rate-dependent material or rate-dependent damping, such as dashpots, mass scaling is the only option for reducing the solution time. In such simulations increasing the loading rate is not an option because material strain rates increase by the same factor as the loading rate. When the properties of the model change with the strain rate, artificially increasing the loading rate artificially changes the process. However, as the increase of mass causes the decrease of the frequency of the 1st natural mode of the system, excessive mass scaling without increasing the ramping time of loading could lead to erroneous solution, since the condition in (5) would not be satisfied.

On the other hand, too much mass scaling could dramatically change the mass distribution and thus the dynamic behavior of the system. Sometimes, manually scale the mass density of the parts would be better than using the feature of automatic mass scaling. We could start with uniformly scaling the mass density of the whole FEA model, and then scale certain individual elements according to the minimum time step size limit. It is more secure to keep the percentage of the added mass through individual element mass scaling low (less than 5%). On the other hand, together with the approach of increasing loading rate, if we only uniformly scale the mass density of the whole FEA model but do not scale individual element mass, then the mass scaling would not be helpful to reduce the computation time. For instance, if we scale the mass of the whole FEA model by a factor of 10, then approximately the stable time step size for numerical integration would be increased by a factor of $\sqrt{10}$, but the frequency of the 1st natural mode of the system would be decreased by a factor of $\sqrt{10}$. If we want to keep $\omega_n t_f \geq 20\pi$, then the ramping time t_f needs to be increased by a factor of $\sqrt{10}$. As a result, with both the stable time step size and the ramping time increased by a factor of $\sqrt{10}$, the total computation time would be the same as in the case of no mass scaling. In addition, it is seen from equation (4) that the ratio of the kinetic energy to the internal energy when $t \geq t_f$ is proportional to $\frac{m}{t_f^2}$. To keep the value of $\frac{m}{t_f^2}$ a constant, the increase of the global mass by a factor of 10 would require the increase of the ramping time by a factor of $\sqrt{10}$. This would lead to the same computation time.

3. Three point bending analysis of a phone

We use explicit method to conduct a 3 point bending analysis of a phone with commercially available FEA software LS-DYNA. The loading condition is as follows: The phone is facing downward. A load of 130N is applied at the center area of a circular shape with diameter of 19mm. The two ends of the phone are supported. One end is fixed, and the other end could move in the longitudinal direction (y axis). Figure 2 shows a schematic diagram of the lateral view of the 3 point bending test of the phone.

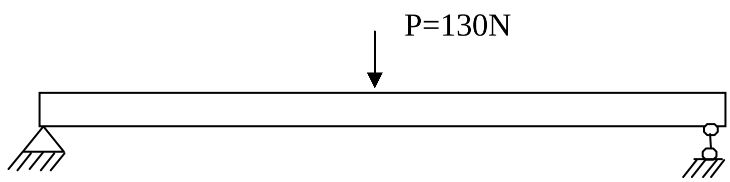

Figure 2. Schematic diagram of a phone under 3 point bending test.

3.1 Selection of mass scaling factor and ramping time for loading

Without accelerating the event, more than a week of computation time would be needed to simulate the quasi-static bending process. This is because the time step size with an explicit analysis of the phone is as small as $5x10^{-5}$ ms, and the time period for the quasi-static bending is in the order of a few seconds. We cut the computation time by mass scaling and loading rate scaling.

We have tried various mass scaling on the 3 point bending analysis of the phone. Without mass scaling, the stable time step size is about $5x10^{-5}$ ms. We set dt2ms=$-3x10^{-4}$ ms in LS-DYNA. That means the minimum time step size permitted in the analysis is $3x10^{-4}$ ms. The time step size has been enlarged by a factor of 6. We tried the following four mass scaling approaches: 1) No uniform mass density scaling was done on the whole FEA model. Through mass scaling on those elements that could not satisfy the $3x10^{-4}$ ms time step size limit, the mass added was 397% of the physical mass (M_0) of the phone. So the total mass of the FEA model was $4.97M_0$. 2) The mass density of the whole FEA model was scaled by a factor of 4, and then 31% of (4 M_0) was added to the model so the total mass of the model is 5.24 M_0. 3) The mass density of the whole FEA model was scaled by a factor of 8, and then 4.03% of (8 M_0) was added to the model so the total mass of the model is 8.32 M_0. 4) The mass density of the whole FEA model was scaled by a factor of 10, and then 1.42% of (10 M_0) was added to the model so the total mass of the model is 10.14 M_0. Table 1 lists the mass scaling details for these four cases. It turns out that in the current analysis, all these approaches could lead to convergent solution with load ramped from 0 to 130N in 30ms. However, in some other cases, the first 2 approaches may cause some trouble. It is recommended that either approach 3 or approach 4 be used.

Table 1. Four cases with different details of mass scaling

Cases	Physical mass	Mass added by uniform mass scaling	Mass added by individual mass scaling	Total mass after mass scaling
1	M_0	0	3.97 M_0	4.97 M_0
2		3 M_0	1.24 M_0	5.24 M_0
3		7 M_0	0.32 M_0	8.32 M_0
4		9 M_0	0.14 M_0	10.14 M_0

Next we discuss quantitatively the reduction of the computation time by approaches 3 and 4. Using approach 3, the mass of the model has been increased by a factor of 8.32, so the natural frequency of the system could be reduced by a factor of 2.9. As a result, the ramping time t_f may need to be increased by a factor of 2.9 ($2.9t_{f0}$). Since the time step size has been increased by a factor of 6, the computation time could be reduced to about 48% of the computation time needed without mass scaling. Similarly, using approach 4, the mass of the model has been increased a factor of 10.14, so the natural frequency of the system could be reduced by a factor of 3.2. As a result, the ramping time t_f may need to be increased by a factor of 3.2, and the computation time could be reduced to about 53% of the computation time needed without mass scaling.

Figure 3 shows a time sequence of the maximum deflection of the phone from a 3 point bending analysis. In that analysis, the mass scaling detail of Case 4 in Table 1 has been used. The ramping time of loading is set to be 50ms. It is seen that the natural period is about 3 ms. Therefore, according to equation (5), a ramping time equal or greater than 30 ms would be needed for a quasi-static analysis.

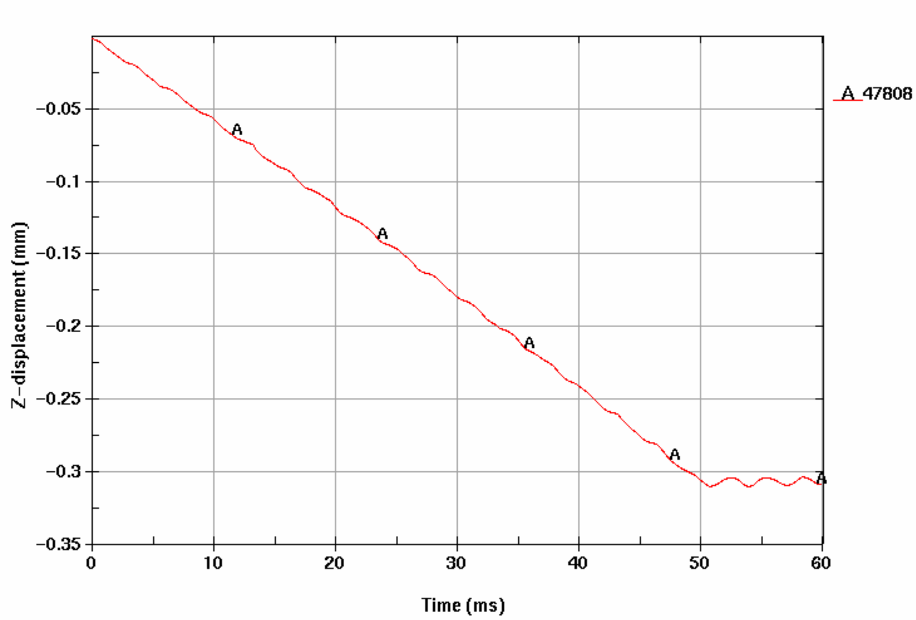

Figure 3. Time sequence of the center deflection of a phone from a 3 point bending analysis.

We have tried to use a ramping time of 1ms, 5ms, 20ms, 30ms, 40ms, and 50ms, respectively. The simulation results converge for all these ramping time periods except in the case of 1ms ramping time. In the case of ramping the load within 1ms, the kinetic energy is too big and the result could not converge. Figure 4 shows the maximum deflection of the phone versus the load applied to the center area of the phone with ramping time periods of 5ms, 20ms, 30ms, 40ms, and 50ms, respectively. It is seen that except the case with ramping time of 5ms, the response is very close to quasi-static. As a result, to obtain accurate enough solution with minimum computation time, a ramping time of 30ms is recommended for the 3 point bending analysis of the phone.

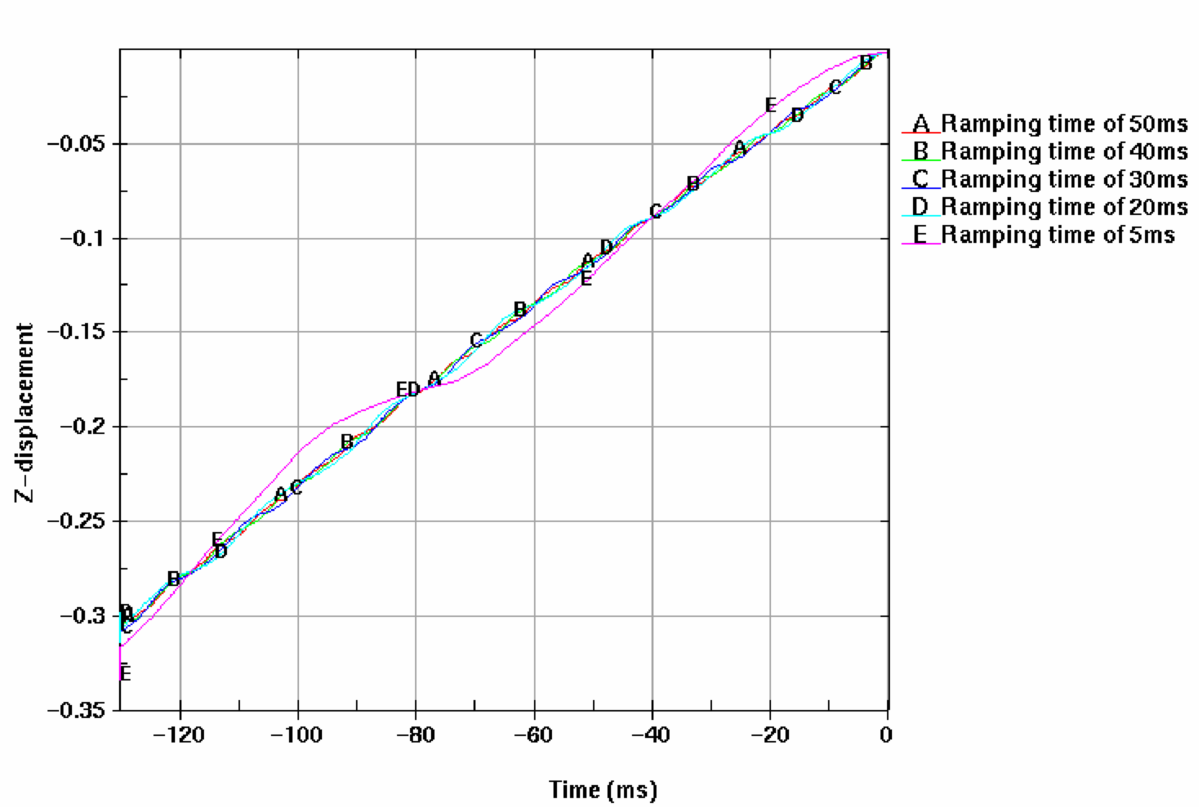

Figure 4. The maximum deflection of the phone versus the load applied to the center area of the phone with various ramping time periods.

3.2 Simulation results

Figures 5 and 6 show the contour plots of the vertical displacement of the phone from 3 point bending quasi-static analysis.

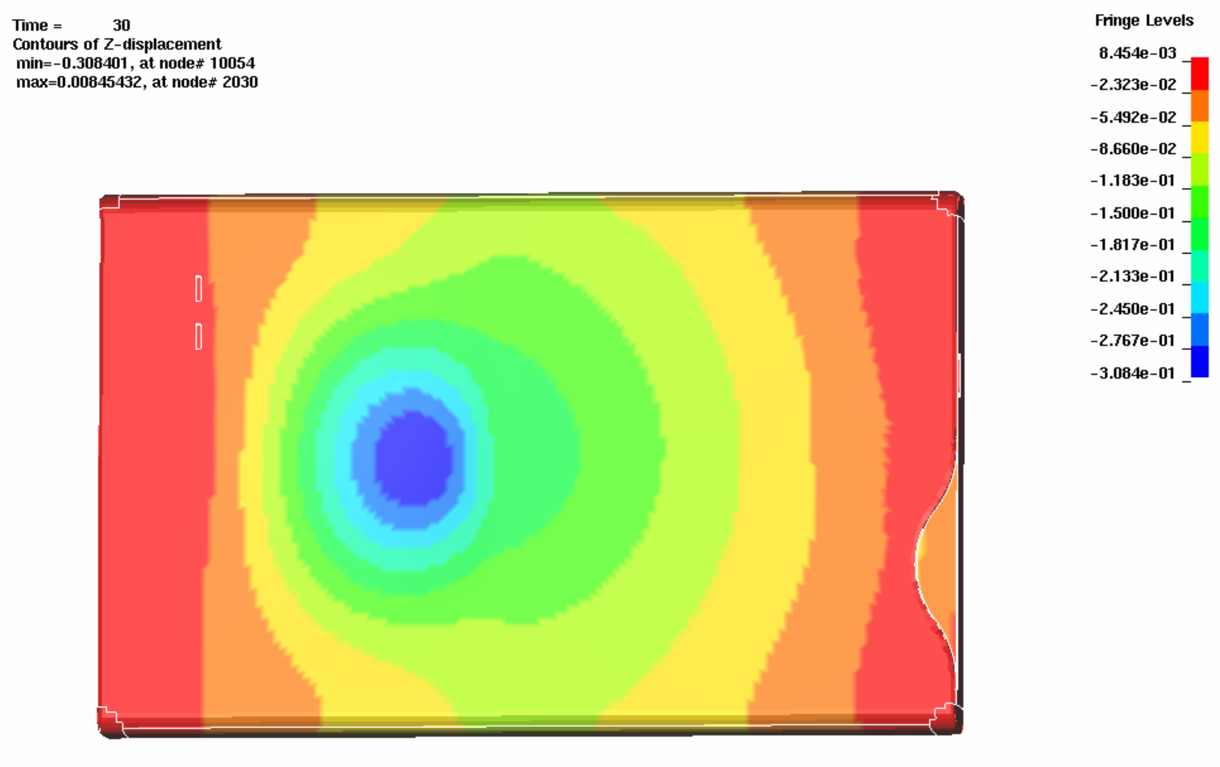

Figure 5. Contour plot of the displacement in z direction on the back of the phone from quasi-static 3 point bending analysis.

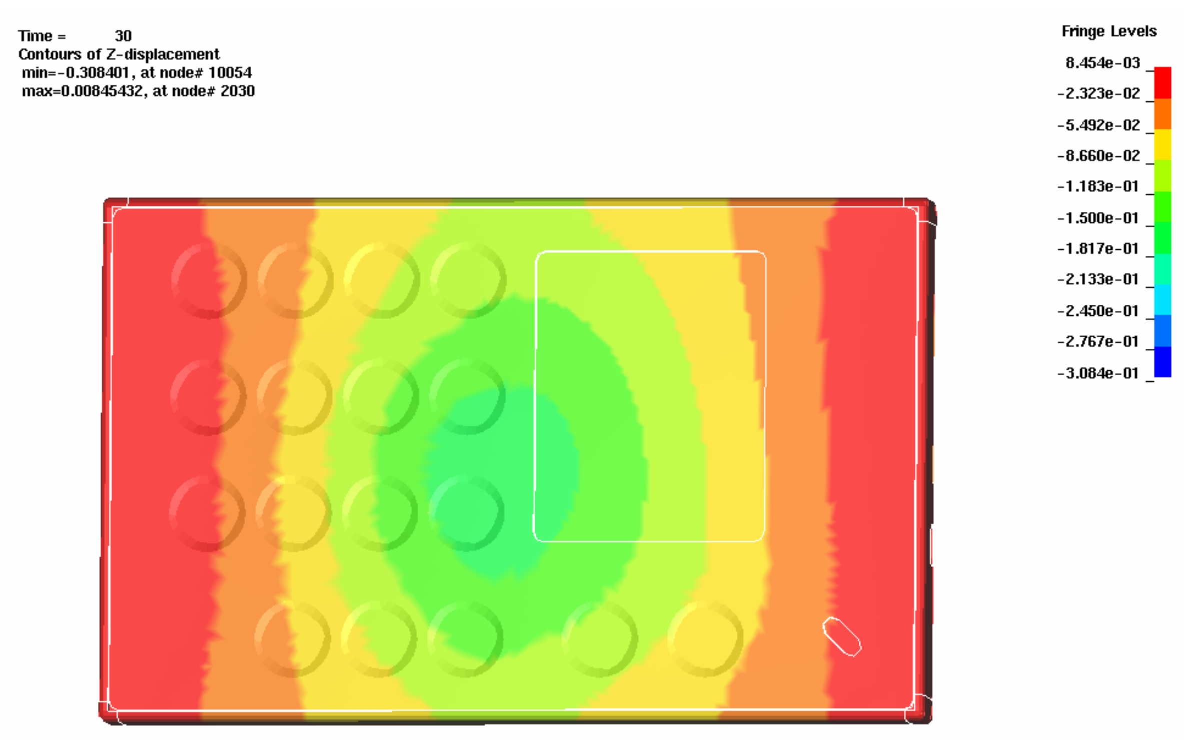

Figure 6. Contour plot of the displacement in z direction on the front of the phone from quasi-static 3 point bending analysis.

As mentioned before, a ramping time of 30 ms is recommended for the quasi-static analysis. It needs about 35 hour CPU time to complete the analysis using one HP UNIX workstation J6700. Figure 7 shows the time sequences of the kinetic energy, internal energy, total energy, hourglass energy, and external work during the 3 point bending analysis of the phone. The load is ramped within 30ms. It is seen that the kinetic energy is very small and is negligible relative to the internal energy. So is the hourglass energy. The internal energy is very close to the total energy of the system and is very close to the external work done on the system. For further clarification, Figure 8 shows the time sequence of the kinetic energy. It is seen from Figures 7 and 8 that the kinetic energy is less than 5% of the internal energy throughout most of the time period of the analysis. The ratio of the kinetic energy to the internal energy is less than 0.07% at the steady state. The internal energy reaches a steady state when the load is held at a constant value. The internal energy of the steady state is about 11 N.mm. To show that 30ms ramping time is enough long for the accuracy of the solution, we show in Figure 9 the energy time sequences during the 3 point bending analysis of the phone when using 50ms as the ramping time. With 50ms as the ramping time, it needs about 58 hour CPU time to finish the analysis using one HP UNIX workstation J6700. It is seen form Figure 9 that the internal energy of the steady state is about 11 N.mm. This shows that both the simulation results with ramping time of 30ms and ramping time of 50ms converge to the same solution. This reveals that the ramping time of 30ms is enough long for accurate solution of the 3 point bending quasi-static analysis. Based on the above discussion, we see that the quasi-static analysis with a ramping time of 30ms is successful.

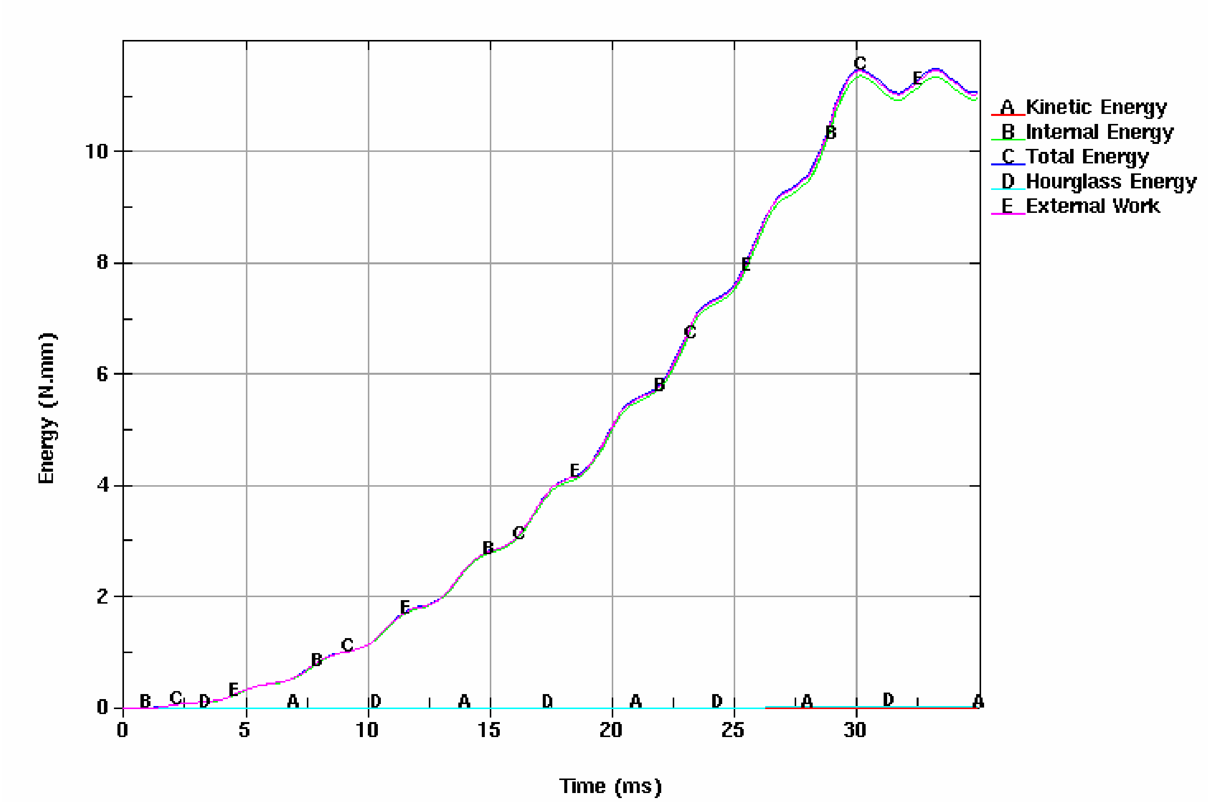

Figure 7. Energy time sequences during the 3 point bending analysis of the phone when using 30ms as the ramping time.

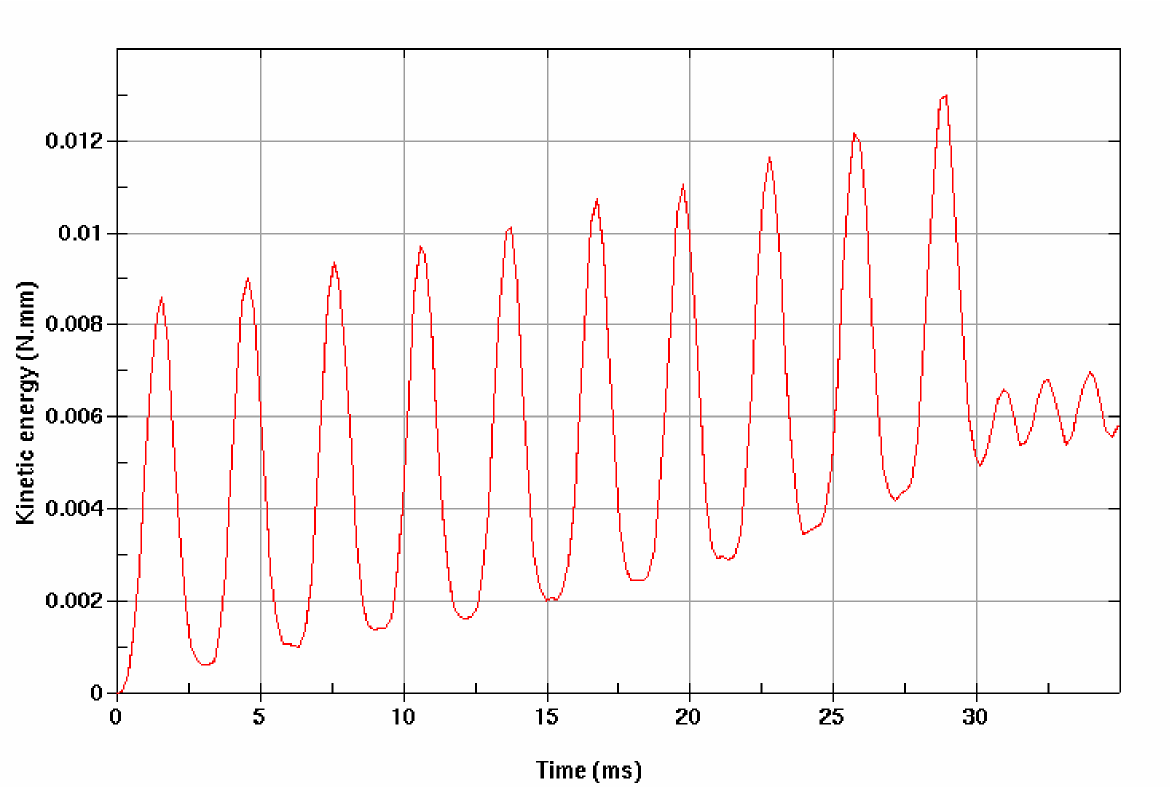

Figure 8. Time sequence of kinetic energy during the 3 point bending analysis of the phone when using 30ms as the ramping time.

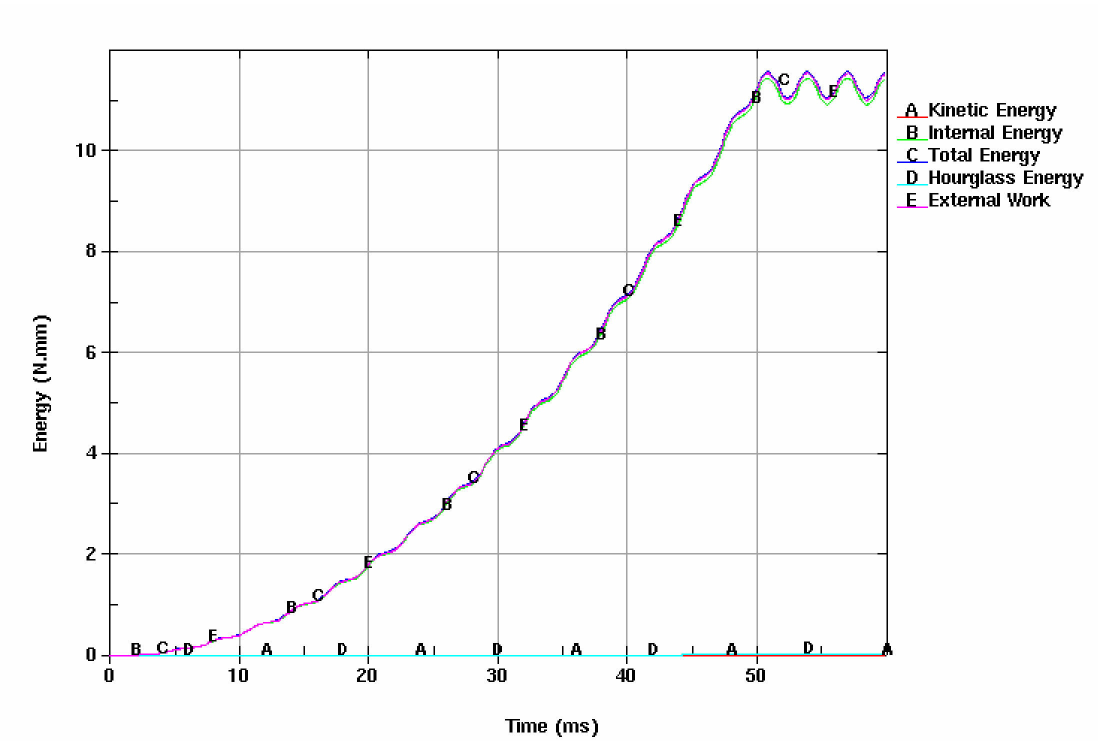

Figure 9. Energy time sequences during the 3 point bending analysis of the phone when using 50ms as the ramping time.

Figure 10 shows the time sequences of the maximum deflection on the phone during 3 point bending analysis when using 30ms and 50ms as ramping time, respectively. It is seen that the maximum deflection in the steady state is about 0.305mm with either of the ramping time.

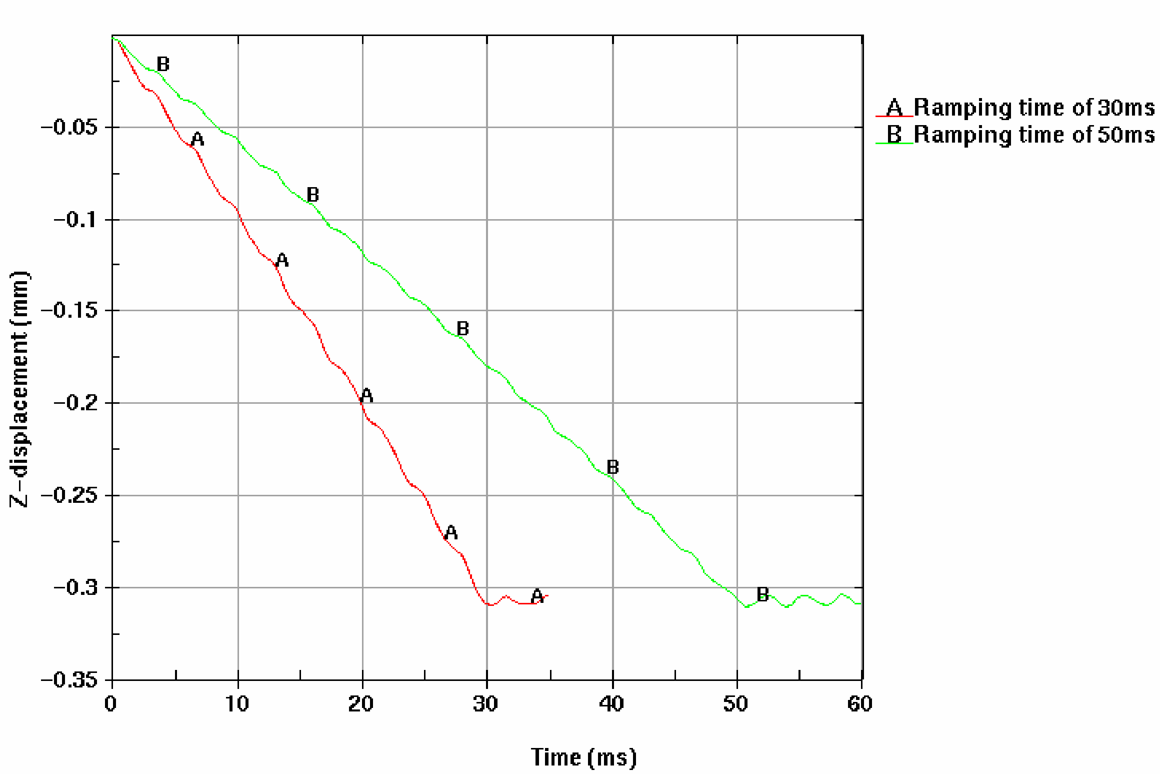

Figure 10. Time sequences of the maximum deflection on the phone during 3 point bending analysis.

In addition, we have checked the peeling stress on the solder interconnects between the electronic packages and the PWB, the 1st principal strain on LCD glass, the Von Mise stress on the PWB and the phone cover, etc.

4. Conclusions

The 3 point bending analysis of a phone has been conducted using explicit integration method. It has been shown that for a quasi-static problem with large number of potential closures/openings in contact, and with large size of FEA model, it is much more convenient to use the explicit method than the implicit method. In using explicit method to solve a quasi-static problem, the computation time could be reduced by loading rate scaling and mass scaling. The loading rate scaling and mass scaling techniques should be both considered since they affect each other. If they are properly used, the speed of the analysis could be increased dramatically without severely degrading the quality of the quasi-static solution. A good rule of thumb is to select the loading rate such that the ramping time of loading is about 10 times of the period of the 1st natural mode of the system. The frequency or period of the 1st natural mode of a real application could be estimated by running frequency analysis or by a few trials of the explicit analysis. Mass scaling could increase the stable time step size of explicit integration, but at the same time it would increase the period of the 1st natural mode of the system. Too much mass scaling could dramatically change the mass distribution and thus the dynamic behavior of the system.

Sometimes, manually scale the mass density of the parts would be better than using the feature of automatically mass scaling. In the 3 point bending analysis of the phone, we start with uniformly scaling the mass density of the whole FEA model, and then scale those individual elements that still could not satisfy the minimum time step size limit. The stable time step size of the analysis has been increased by a factor of 6 with the aid of mass scaling. Based on the mass scaling factor, a 30ms of ramping time of loading is recommended in comparison with a few seconds in real 3 point bending test. The loading rate has been accelerated by 2 orders of magnitude. According to the time sequences of the energy profile and the maximum deflection of the phone, the 3 point bending quasi-static analysis is successful. The total CPU time needed to finish the analysis is about 35 hours using one HP UNIX workstation J6700. The computation time could be reduced by using multiple CPU and/or using faster computer.

Acknowledgements

The authors would like to thank Fujii Takaharu for managing the research project. Thanks and appreciations also extend to the members in the NRC Dallas product integration group. Without the great helps of them, it would be impossible to finish this work.

References

[1] J. T. Wang, T. Chen, D. W. Sleight, and A. Tessler, Simulating nonlinear deformations for solar sail membranes using explicit time integration, 45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, Palm Springs, California, AIAA 2004-1580, April 19-22, 2004, pp. 15, (837KB).

[2] Getting started with ABAQUS/EXPLICIT keywords version, ABAQUS Version 6.4 Documentation.

[3] LS-DYNA keyword user's manual, April 2003, version 970.

[4] ABAQUS analysis user's manual, ABAQUS Version 6.4 Documentation.