The Pennsylvania State University

The Graduate School

Department of Mechanical and Nuclear Engineering

MPI ENABLED SHAPE OPTIMIZATION OF SOLID ISOTROPIC PLATES TO MITIGATE THE EFFECTS OF AIR BLAST LOADING

A Thesis in

Mechanical Engineering

by

Vikas Argod

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

May 2009

The thesis of Vikas Argod was reviewed and approved* by the following:

Ashok D. Belegundu
Professor of Mechanical Engineering
Thesis Advisor

Mary Frecker
Professor of Mechanical Engineering

Karen A. Thole
Professor of Mechanical Engineering
Head of the Department of Mechanical and Nuclear Engineering

*Signatures are on file in the Graduate School

ABSTRACT

This work presents a formal approach for optimizing the shape of freely supported isotropic plates to withstand air blast loading. Unique difficulties are presented in optimizing for short-duration dynamic loading, viz. transient dynamic response, monitoring of maximum plastic strain failure at every point in the panel over time, optimizers that can handle non-differentiable, non-convex and computationally expensive functions, and mesh distortion**.** The goal is to minimize the dynamic displacement of the plate relative to the test fixture, while monitoring plastic strain values, mass, and envelope constraints. An approach based on coupling LS-DYNA finite element software and a differential evolution (DE) optimizer is presented. Since DE involves a population of designs which are then mutated and crossed-over to yield an improved generation, it is possible to use coarse parallelization wherein a computing cluster is used to evaluate fitness of the entire population simultaneously. However, owing to highly dissimilar computing time per analysis, owing to mesh distortion and variable time step in explicit finite element analysis, implementation of the parallelization scheme is challenging. Sinusoidal basis shapes are used to obtain an optimized ‘double-bulge’ shape for a centrally located charge. The optimized shape shows vast improvement over a flat (baseline) plate of equal mass; the blast waves are smoothly deflected away and the maximum plastic strain is evenly smeared along the plate edges, indicating better utilization of material.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF NOMENCLATURE

$\mathbf{x}$	=	design variables vector
$\mathbf{x}^{\mathbf{L}}$	=	lower limit on design variables
$\mathbf{x}^{\mathbf{U}}$	=	upper limit on design variables
$\mathbf{G}$	=	vector of x-, y-, z- coordinates of nodes in FE model
$\mathbf{q}^{i}$	=	i th velocity field or trial shape change vector
w	=	z- or normal displacement
$\varepsilon_{\max}$	=	maximum plastic strain at failure for the material
ε	=	equivalent plastic strain vector
M	=	total mass of the structure
$M_{\max}$	=	upper limit for the mass of the structure
t	=	thickness of the structure (plate) at any (x, y) location in the plate
$t_{\min}$	=	Minimum thickness allowed
$\mathbf{z}$	=	vector of z-coordinate of the nodes
$\mathbf{z}^{\mathbf{U}}$	=	upper limit on z-coordinate
$\mathbf{z}^{\mathbf{L}}$	=	lower limit on z-coordinate
(det J_j)	=	Jacobian of j th hexahedral element at all the eight nodes
n_{pop}	=	Population size
n_{eval}	=	Number of Evaluations
n_p	=	Number of processors

ACKNOWLEDGEMENTS

I would like to express my gratitude towards my advisor Professor Ashok Belegundu for his guidance and support. His advice was very valuable in leading me through the often complex process of drafting and completing a thesis. I am grateful for his faith in my ability to work on this important project funded by the Army Research Office. I cannot forget the countless meetings with Professor Belegundu which have enlightened me about many qualities of a researcher. His patience in debugging the code is worth mentioning, not to forget his sense of humor.

I would also like to thank the Research Computing and Cyberinfrastructure group of Information Technology Services, led by Mr. Vijay Agarwala, which financially supported me and made my studies at Penn State possible. This group provided me with a friendly and intellectually stimulating environment. I appreciate Vijay for giving me this opportunity of working with large computational resources, which has defined a new avenue for my career. It has always been a great experience working with Vijay, a dynamic leader equipped with 'cutting edge' knowledge. Special thanks to Abdul Aziz and Jason Holmes who were always available to help me. I thank Dr. Anshuman Gupta for motivating me to study optimization and guiding me towards the 'fifth discipline'.

This work was also made possible thanks to Dr. Subramaniam Dharmarajan of Arizona State University who continuously helped me with his insights and with whom I had interesting discussions on MPI implementation. I thank Rajeev Jain and Sumanta Kumar Nayak for their cooperation and inputs in this collaborative project. I thank Professor Mary Frecker for her valuable comments on my thesis.

I would like to thank my parents for always motivating me to think big and follow my heart. I am indebted to my brother Narendra and sister-in-law Nisha for their constant support that enabled me to undertake the studies I dreamed of.

My years at Penn State would not have been the same without my friends. I thank Shyam Benegal for his encouragement and help during my grad school application process. I would like to thank Gagan Gothaniya for his timely advices, Shivakumar Jolad for being so helpful in many aspects of my life, Piyush Thakre and Archana Iyer for inspiring me to join Association for India's Development (AID) which became my home away from home. I must also acknowledge the motherly love of Jayashree Sonti, the sagely advice of Anirban Hazra and Arijit Chatterjee when I needed it most, and the many enjoyable hours spent in the company of Harisha, Umamahesh and Mahesh. Special thanks to Abhijith, Hari, Ramya, Somesh, Pratik Mankidy and all other friends from AID.

Chapter 1

Introduction

Background

This thesis presents an approach and methodology to optimize the shape of metal panels to withstand blast loading. Figure **1-1** shows a flat plate subject to an air blast load. The challenge is to optimize the shape of the plate for blast mitigation.

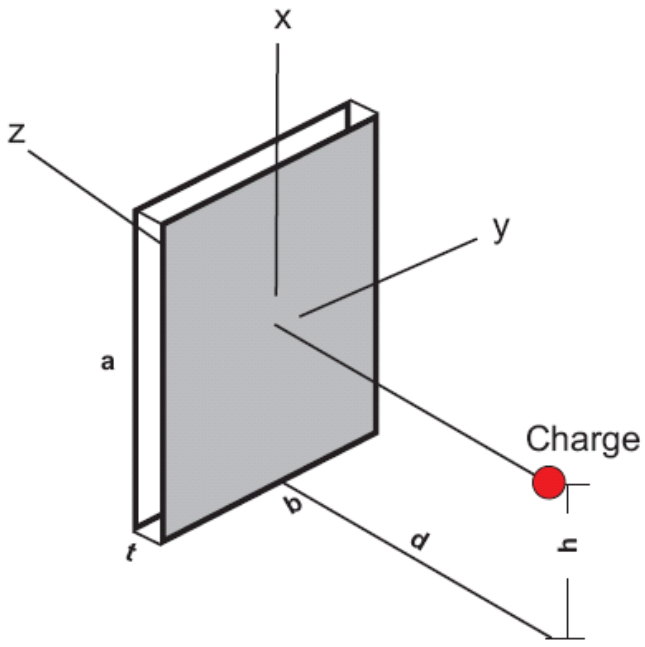

Figure **1-1**: Schematic representation of a flat Plate subject to Blast Load

Unique difficulties [1-3] are presented in optimizing for blast loading, viz. transient dynamic response, modeling the experimental setup with appropriate boundary conditions, monitoring of maximum plastic strain failure at every point in the panel over time, and optimizers

that can handle non-differentiable, non-convex and computationally expensive functions. Vastly superior structural shapes can be designed with optimization methodology compared to ad-hoc techniques. This is substantiated by a considerable amount of trial-and-test runs which are not presented here. Several military and non-military applications motivate this research. Considerable attention in journals and conferences worldwide has been given to *analysis* of metallic and composite panels subject to both blast and ballistic loads. However, very little work has been done on optimizing *shape* of panels for *blast* loads. The few papers that are available in open literature have focused on rib patterns [4] and on foam and honeycomb panels [5]. Remaining papers have focused on ballistic protection which is not the focus here. In this thesis, a rigorous shape optimization methodology is used to arrive at the best shape attainable given a discrete set of sinusoidal basis shapes. The procedure involves an integration of an optimizer with the LS-DYNA simulation code.

Use of high performance computing techniques is necessary to solve with larger number of basis shapes. Hence, shape optimization of panels subjected to blast loading has been carried out using Differential Evolution algorithm with distributed population evaluation. The population evaluation for each generation involves many explicit analyses using LS-DYNA solver. Considerable variation in computation time necessitates load balancing for the effective utilization of available computation resources. All these considerations make the methodology very robust and general. In the literature, regarding designing for impact mitigation, much greater focus has been placed on ballistic impact rather than on blast. Very few papers use formal optimization techniques. Publications related to blast damage mitigation are given in the following section.

Literature Search

In a study to find out optimal layout of the stiffeners for a given mass, Dharaneepathy and Sudhesh [4] investigated the stiffener patterns on a square plate subject to blast loads modeled using Friedlander's exponential function. This study was aimed at determining the free and forced vibration responses of a plate with different stiffener patterns. While formal optimization was not used to optimize the stiffener patterns, they demonstrated that stiffeners do provide significant advantage compared to an unstiffened panel of same weight, and that a waffle pattern is not as good as a novel pattern that they proposed. They show that blast loads cause only fundamental symmetric mode. Hence the design with maximum fundamental natural frequency and minimum blast response becomes the superior plate. Failure was not considered in their study – that is, only deflection was considered. Hou et al [9] also consider the effect of stiffener size on blast response. Xue and Hutchinson [10] and Fleck and Deshpande [11] compare blast resistance of solid versus sandwich panels (such as pyramidal truss core, square honeycomb and folded plate) which are assumed to be ductile to withstand deformation caused by the impulsive blast loads. The plates were considered to be infinitely long in one direction and fixed at the ends of the short direction. Abaqus/Explicit was used to model the blast load in [10] while an approximate analytical approach was used in [11]. For a solid plate, the impulse blast-type load is uniformly distributed across the thickness as a uniform initial normal velocity $v=\boldsymbol{I}/\rho h$, where ρ is the density of material and h is the plate thickness. But in case of sandwich plates, only face sheet towards the blast gets the impulse loading. However, initial velocity is uniform and given by $v=\boldsymbol{I}/\rho h_f$, where h_f is the thickness of face sheet. Blast in both air and water were included in the comparative study. From certain normalized displacement vs. impulse graphs, it was concluded that some of the sandwich topologies outperformed solid panels of same mass, especially in water. Two times decrease in momentum was seen with fluid structure interactions relative to that

of panels of same weight. This study shows that there is considerable potential for exploiting sandwich pattern for blast resistant designs. Highly refined three-dimensional finite element model of the sandwich plate is used to compute the blast resistance followed with an attempt to optimize to achieve maximum performance at a specified weight in [10]. This work also attempts to find near-optimization design of sandwich structures of similar properties (which are typically represented by dimensionless parameters like mass per unit area). However, no formal optimization methodology is considered. Yen, Skaags and Cheeseman [5] present an experimentally validated dynamic analysis procedure utilizing LS-DYNA and the ConWep (blast function provided in LSDYNA) air blast function with shock mitigation materials such as honeycomb or foam. The numerical results indicate that significant reduction in the maximum stress amplitude propagating within the protected components can be achieved by suitable selection of honeycomb material with proper crush strength. This work establishes the use of ConWep blast load within LS-DYNA for a reasonable accuracy. In this study, dishing deformation and spring back phenomenon were related to the observed results. It concludes that increase in saturated impulse value is due to the "dishing deformation of the front of the face sheet" which is exposed to the blast and spring back effect (in which a part of the structure travelling opposite to the direction of the blast) imparts "extra impulse" to the structure increasing a higher impulse. Liang Yang and Wu [12] investigated the optimum design of panels under blast loading by using a combined algorithm, the Feasible Direction Method (FDM) and the Backtrack Program method (BPM). However it focuses on static loading and hence not very relevant in the present context of blast effects where inertia effects play a role. Further, their paper ignores material nonlinearity.

Main and Gazonas [13] study the effect of an air blast on uniaxial crushing of a cellular sandwich plates. Unlike contemporary studies on sandwich plates trying to minimize dynamic deflections, the objective was to mitigate shock transmission through the thickness of the

sandwich plate; however the capacity to mitigate shock by a sandwich plate is limited. Initial numerical results show that this capacity of the sandwich plates of shock mitigation can be improved by varying mass fractions of the front and back face sheets. Further, it investigates physics of variation in geometry and shock mitigation capacity leading to an interesting optimization formulation to maximize mass distribution of front sheet (which maximizes impulse absorption strength) while minimizing back-face acceleration (increase in this acceleration reduces the protection provided by the core material). Icardi and Ferrero [14] study optimum fiber orientations in a laminated composite to absorb energy while maintaining stiffness. Sriram *et al* [15] studies the feasibility of using some lightweight composites (sandwich composite plates, S2-glass/epoxy laminated face sheets and aluminum foam core) in designing blast-resistant protective structures. LS-DYNA was used to simulate the blast load. The work also observes and confirms the 'dishing' phenomenon causing increased energy transfer in the sandwich composites. Variation in dishing with the variation in TNT mass and standoff distance of the blast is also studied in this work. Most of the work described above use LS-DYNA solver while Thiagarajan *et al* [16] present a numerical simulation scheme in which an in-house mutiscale model called Virtual Internal Bond (VIB) used along with Abaqus/Explicit. This scheme was used to simulate behavior of plates under various load cases involving blast load. In a study on ballistic impact behavior [17], researchers establish the relation between interfacial friction and the energy absorption of fabrics. Various boundary conditions were considered and conclude that the fabric with high friction absorbs more energy than the frictionless fabric. Friction maintains integrity of the local fabric structure at the impact region. However, this work is not directly related to present work since it considers only ballistic impact with no optimization involved. Summarizing [18], it can be noted that most of the research is mainly focused at

(i) Distributing the blast energy to much larger area to increase the area of absorption

(ii) Adjusting the overall structural stiffness and damping properties and

(iii) Re-distributing the material density and changing the material orientation to achieve better penetration protection.

In this task of designing plates for blast mitigation, achieving the best shape plate against blast has been a fairly less explored area. V-shaped plates to deflect blast energy have been proposed in army vehicle related websites, without scientific details [3]. Disadvantages of flat bottomed vehicle is that blast effects gets amplified for an explosion directly beneath. A blast wave striking bottom gets 'reflected' back at the ground thereby increasing blast over pressure (BOP). Ref [19] reviews various materials technologies for lightweight fighting vehicles, advances in fiber sizings which can provide strain rate sensitive response (strong adhesion at structural loading and weak at ballistic loading).

To summarize; In contrast with the papers cited above, this work focuses on shape, uses formal optimization techniques, incorporates boundary conditions to match those in an experimental setup, considers solid metal plates, and uses LS-DYNA for simulation. Thus, the methodology involves integrating an optimizer with the LS-DYNA simulation code.

Organization of the thesis

Chapter 2 introduces the optimization model. It also describes the finite element modeling considerations and velocity fields used for the shape optimization. In Chapter 3, optimizers and computer code development are discussed. Coarse parallelization of DE optimizer is also explained in this chapter. Numerical Results are given in chapter 4 and conclusions are given in the Chapter 5.

Chapter 2

Problem Description

Problem Definition

Shape optimization of a square aluminum panel is considered. Maximum deflection of the model is the objective function to be minimized subject to plastic strain limit, mass limit and other design related limits. Development of the objective function and constraints is explained in this section. A schematic diagram of the plate used for shape optimization is shown in Figure **2-1.**

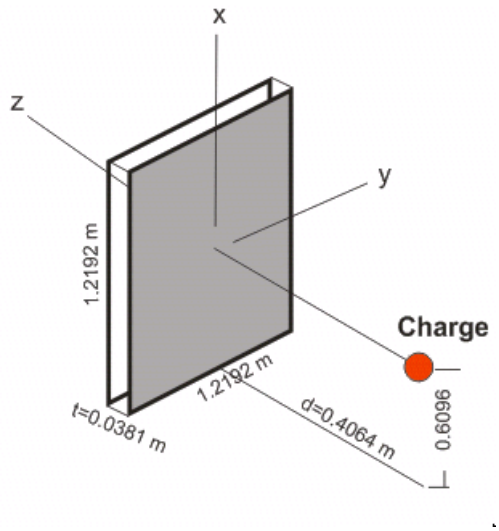

Figure **2-1**: Baseline Structural model details .

The standoff distance of the charge is taken to be 0.4064 m. It should be noted that this plate of interest is just a part of a *grip* assembly used to model the experimental condition as explained in the next section. As is the usual procedure in shape optimization problems, the final shape or design is obtained as a combination of a set of 'input' trial shapes called velocity fields

or basis shapes. The design space consists of all the possible combinations of these basis shapes. With reference to Figure **2-1**, the basic problem addressed in this work can be stated as follows:

Given a set of basis shapes that controls the shape including thickness of the plate, a mass limit for the structure, plastic strain limits representing fracture strength, a minimum thickness for the panel, and a geometric envelope within which the structure must lie, determine the best possible combination of these basis shapes that minimizes the deflection (at the first peak in time).

Mathematically, the design optimization problem can be stated as follows:

Find $\mathbf{G}(\mathbf{x})$

Minimize $\max_t w_{rms}(t)$ 2-1

subject to

$\varepsilon_j \leq \varepsilon_{\max}$ for each element j

$M \leq M_{\max}$

$thickness \geq thickness_{\min}$

$\mathbf{x}^{\mathrm{L}} \leq \mathbf{x} \leq \mathbf{x}^{\mathrm{U}}$

$det\, J_j(\mathbf{x}) \geq 0$ for each element j

$\mathbf{z}^{\mathrm{L}} \leq \mathbf{z} \leq \mathbf{z}^{\mathrm{U}}$ (geometric envelope)

where the notation is explained in the nomenclature section.

G(x) as the grid point vector representing x-, y- and z-coordinates of each node in the finite element model. It is give by the Equation **2-2**

$$\mathbf{G}(\mathbf{x}) = \mathbf{G}_{original} + \sum_{i=1}^{N_{dv}} x_i \, \mathbf{q}^i \qquad 2\text{-}2$$

Thus, design variables, x_i essentially represents the amplitude of velocity field, $\mathbf{q}^i$.

Objective function

In the objective function, $\max_t w_{rms}(t)$, represents the peak *rms* deflection over space and time perpendicular to the plate surface. As per the physics, peak over time occurs at the first response peak in time. Since the blast is a one-time phenomenon, time history of the deflection shows a downward trend, the reason why deflection at the first peak is considered in the cost function. Note that *rms* of a vector x is defined by as

$$\|x\|_{rms} = \sqrt{\frac{\sum_{i=1}^{n} x_i^{\,2}}{n}} \qquad 2\text{-}3$$

In the present formulation, the *relative* z-displacement of nodes in the plate is considered as given in the following equation

$$w_{\text{relative}} = w - w_{\text{fixture}} \qquad 2\text{-}3a$$

This equation eliminates rigid body displacement of the freely supported structure. *x*- and *y*- displacements are not significant and are not considered. The displacement is a function of time, and value at first peak is monitored. Reduction in the objective function reflects decrease in peak dynamic deflection. Thus, in Equation **2-3,** $x_i \equiv (w_i)_{\text{relative}}$, i=node number on the bask fact of the plate (away from the charge).

Constraint Equations

(i) A body subjected to a load develops stress and strain. Strain is of two types: Elastic and plastic. Elastic refers to the recoverable strain while plastic to non-recoverable. When subjected to blast, in most practical applications, the strain is mostly *plastic*. Plastic strain is a function of time and stabilizes after a certain

simulation time duration. This stabilized value, which is the peak value, is constrained. In the FE model, plastic strain values are calculated for all the elements and are monitored. Plastic strain limits are material properties and any strain above that limit causes rupture.

(ii) *M* above refers to the combined mass of the assembly. Equation to calculate mass of a hexahedral element is obtained by basic finite element principles:

$$\begin{aligned} Volume, V_e &= \iiint_V dx \cdot dy \cdot dz \\ &= \int_{-1}^{+1}\int_{-1}^{+1}\int_{-1}^{+1} |\det \mathrm{J}| \, d\xi \cdot d\eta \cdot d\zeta \end{aligned} \qquad 2\text{-}4$$

The integration in the Equation **2-4** can be performed using Gaussian quadrature. When 2x2x2 integration is considered, volume of an element can be calculated with the following Equation **2-5**

$$V_e = \sum_{\mathrm{IP}=1}^{8} \mathrm{W}_{\mathrm{IP}} \, |\det \mathrm{J}_{\mathrm{IP}}| \qquad 2\text{-}5$$

Note that W represents the weights and the determinant is calculated at Gauss points. The total volume is sum of all the elemental volume and mass is calculated as,

$$M = \left(\sum_{e=1}^{n_e} V_e \right) \cdot \rho \qquad 2\text{-}6$$

(iii) Thickness is computed from nodal coordinates of the hexahedral elements used in the FE model. Thickness should not go below a manufacturable limit and is monitored so as to be above a critical value. Thickness is calculated as the difference in z-coordinates of nodes in top and bottom surface of the plate. Note

that the plate is parallel to x- and y- place and z direction is perpendicular to the plate. 'Top' surface refers to the surface away from the charge.

(iv) Since the design space has all the combination of basis shapes, there is a need to prevent element distortion i.e. to prevent non-convex hexahedron elements. Determinant of Jacobian of distorted elements will be ≤ 0 at one or more nodes in the hex element. Hence, element distortion is prevented by requiring the determinant to be ≥ 0 at all eight nodes in each element during optimization. The Jacobian for each element is computed from element nodal coordinates. The check for distortion is done before performing finite element analysis.

(v) Geometric Envelope: Changes in the shape of the plate are restricted to lie within a box envelope. Basically, all the coordinates of any new shape is checked whether it lies inside the box. This has been termed 'Geometric envelope constraint'. This also safeguards against solutions where the new shape overlaps with the location of charge itself.

FE Modeling Considerations

Initially, a uniformly thick square aluminum plate is considered as a starting shape or baseline design. Initial studies were carried out with the baseline design to better understand the explicit finite element analysis procedure. To start with, a plate with springs along the boundaries is considered without any *grip* structure that represents experimental setup. Various boundary conditions have been studied, including one or more 'fences' of springs around the edges. Figure **2-2** shows one of those initial models with springs along the boundaries.

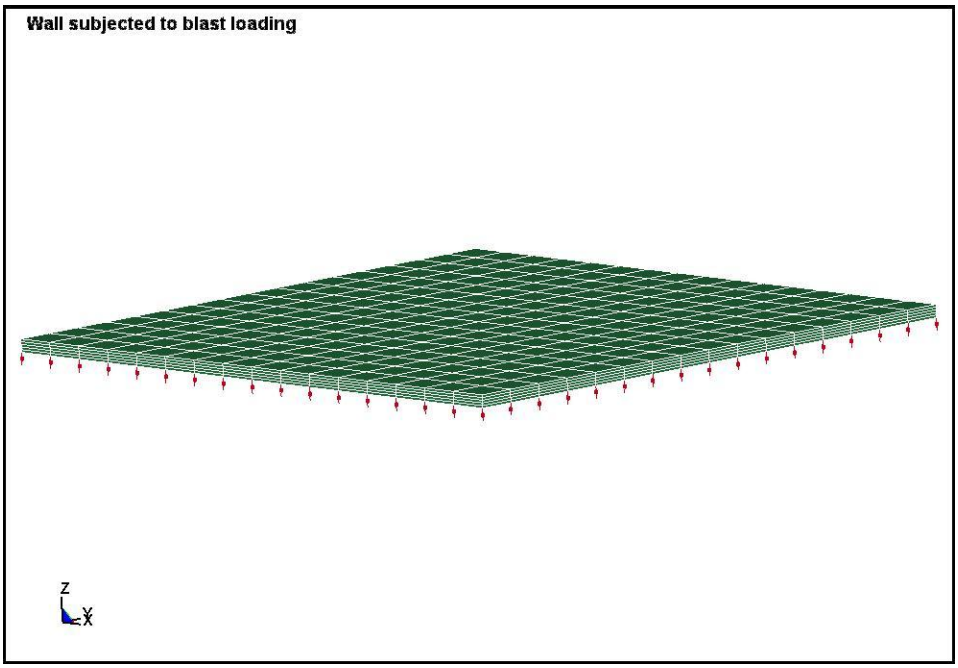

Figure **2-2**: Plate with springs along the boundaries .

Initial studies were mainly to establish the various aspects of shape optimization like development of basis shapes, selection of optimizer etc discussed subsequently. However, these boundary conditions produced unrealistic plastic strain concentrations at the corners. A grip system or assembly that holds the plate was finally adopted owing to best correlation with experimental data (Figure **2-3**).

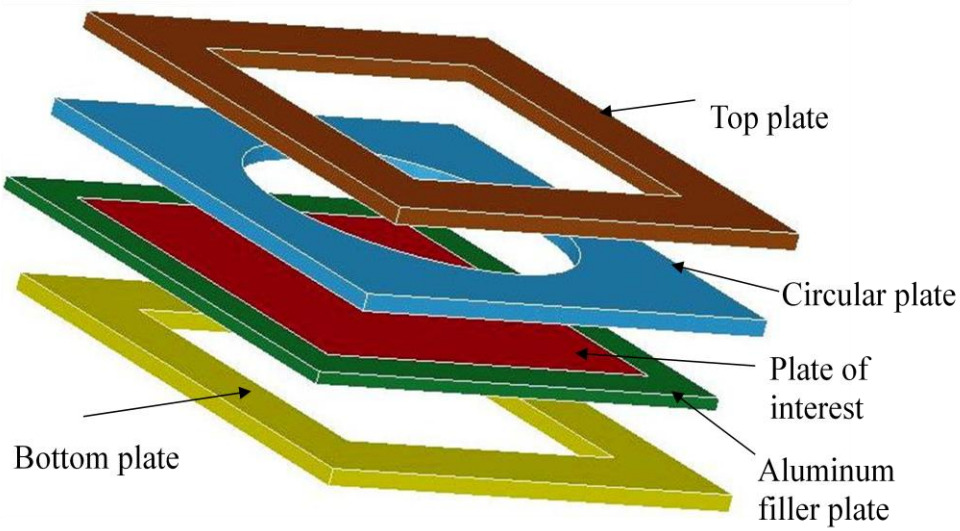

Figure **2-3**: Exploded view of the structural model

Components of the grip assembly are:

i) the 'plate of interest' (red), made of aluminum.

ii) a filler plate (blue), also made of aluminum, having same thickness as the edges of the plate of interest, necessary for the assembly.

iii) three rigid plates, whose material properties are not relevant – steel is assumed for density, and an artificially very high value for Young's modulus is used to ensure rigidity. Importantly, the circular cover plate (blue) above the plate of interest creates peak plastic strain at the center of the flat plate of interest, *as this is the location where a freely supported flat plate is experimentally observed to fail.* That is, without this circular cover plate, as with other modeling attempts mentioned above such as spring supports, unrealistic pseudo concentrations of plastic strains are observed at corners.

iv) A square portion of the model at the center of plate of interest (red) is taken to be the domain of shape optimization. This square portion lies within the circle of the cover plate (blue). Thus, shape changes (such as bulges) are introduced in this

square domain only. The region outside the domain in the plate of interest is flat, whose thickness equals that of the surrounding (blue) filler plate.

It is important to get boundary conditions to be realistic as the optimum shape will depend on it. Surfaces in contact are as follows:

(i) Bottom surface of top plate and top surface of circular plate,

(ii) Bottom surface of circular plate and top surfaces of filler plate and

(iii) Bottom surface of filler plate and top surface of bottom plate

All the above contact properties are defined using *CONTACT features and TIED_SURFACE_TO_SURFACE_ID card is used. The simulation on the baseline (i.e. flat plate) design produces maximum plastic strain at the center of the plate as also observed in the experiments. Blast load is a radially outward propagating wave caused by detonation of an explosive. After the explosion, blast waves strikes the object as incident waves. Reflected waves are generated after they hit the obstacle and start travelling back in the reverse direction which is nothing but towards the point of explosion. Hence, somewhere on its path, reflected and incident waves meet producing a single vertical wave called “Mach Stem” [14]. This causes variable loading on structures below and above this point of intersection. There are mainly two methods of producing pressure versus time histories during computational simulation [20] as given below:

1. Maintaining a database in the form of pressure versus radial distance for specified time steps. This requires a comprehensive database of blast data.
2. Using polynomial approximating functions for the peak incident pressure and shock front arrival time at a specified distance. Pressure is regarded as a function of time with an exponential decay functions. This enables us to calculate time history plot of pressure for a large range of explosive charge weights over a large range of distance.

All the functions needed as described above were established by empirical fits to airblast data [20]. Figure **2-4** represents the P-T plot of a typical blast wave. After the explosion, pressure build up to a peak value of the overpressure (P_0) which then decays to P_s. It continues to decay and reached partial vacuum and then returns to local ambient pressure. For calculation purposes only positive phase is considered.

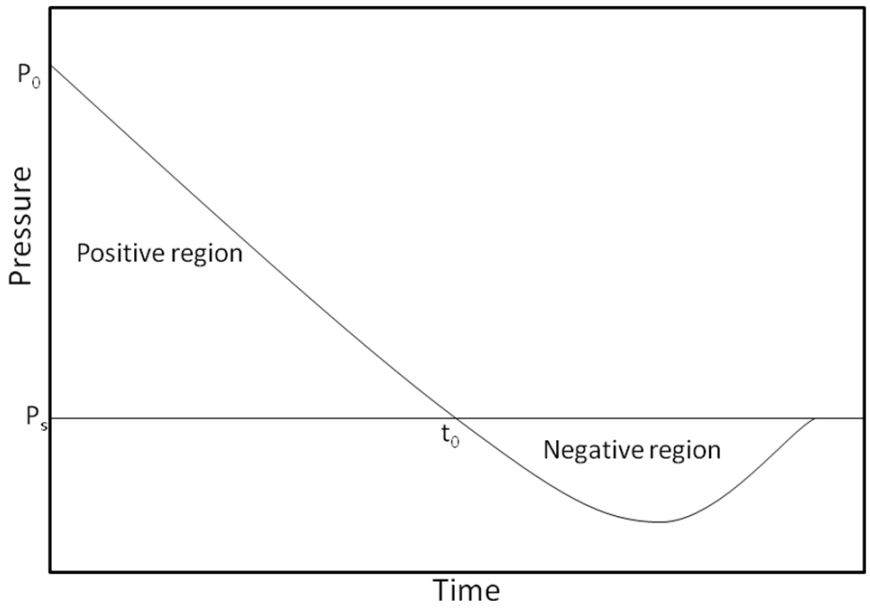

Figure **2-4**: Characteristics of a typical blast load .

Pressure at any time *t* in the positive phase is given by Equation 2-7 which is called Friedlander's equation and impulse per unit of projected area is given by Equation **2-8**. Irrespective of any blast material, most of the blast calculations are done based on effective mass of Tri-Nitro-Toluene (TNT). Using blast scaling law [15] which states that "when two charges of the same explosive and same geometry, but of different size are detonated in the same atmosphere, the shock waves produced are similar in terms of the same scaled distances", blast characteristics of any given mass of TNT can be calculated.

$$P(t) = P_o\left(1 - \frac{t}{t_o}\right)\exp\left(-\alpha\frac{t}{t_o}\right) \quad 2\text{-}7$$

$$I_s = \int_0^{t_o} P(t)dt = P_o t_o\left[\frac{1}{\alpha} - \frac{1}{\alpha^2}\left(1 - \exp(-\alpha)\right)\right] \quad 2\text{-}8$$

Scaled distance is given by Equation **2-9** and scaled time is given by Equation **2-10**

$$Z = R/W^{1/3} \quad 2\text{-}9$$

$$t_{sc} = \frac{t}{W^{1/3}} \quad 2.10$$

All the above functions are based from various studies done by Ballistic Research Laboratory (BRL) [20]. It is implemented in CONWEP algorithm which is implemented in *LOAD_BLAST function of LS-DYNA [21]. See Refs. **[9-11]** for more details on the CONWEP feature. Basically this implementation, uses Friendlander's equation (Equation **2-7**) and Hoffman's scaling law (Equation **2-9**) to calculate pressure load for a a given TNT amount and standoff distance. The inputs for *LOAD_BLAST input card used are equivalent TNT Mass, type of blast (surface or air), load curve, charge location, and surface identification for which pressure is applied. CONWEP calculates the appropriate reflected pressure values and then applies these to the appropriate surfaces by taking account the angle of incidence of the blast wave. It should be noted that the loading changes with changes in shape of the plate, although this is automatically calculated within Ls-Dyna. Blast parameters are given in Table **2-1** and blast segment (or blast area) in the assembly is indicated in Figure **2-5** (indicated by black marks) It essentially indicates the portion of the assembly exposed to blast directly and consists of portions of panel (red plate in the Figure **2-3**) and surfaces of bottom plate. Note that all the parts are assembled using

*CONTACT_AUTOMATIC_SINGLE_SURFACE feature of LS-DYNA which represents a single surface contact. Conventional values are used for various parameters namely static friction coefficient =0.3 and Dynamic coefficient = 0.15 are used

Table **2-1**: Blast load input data .

Property	*Value*
Equivalent mass of TNT	8 kg
Blast Location	(0.0,0.0,-0.4064) m
Type of Burst	Air Blast (Spherical Charge)

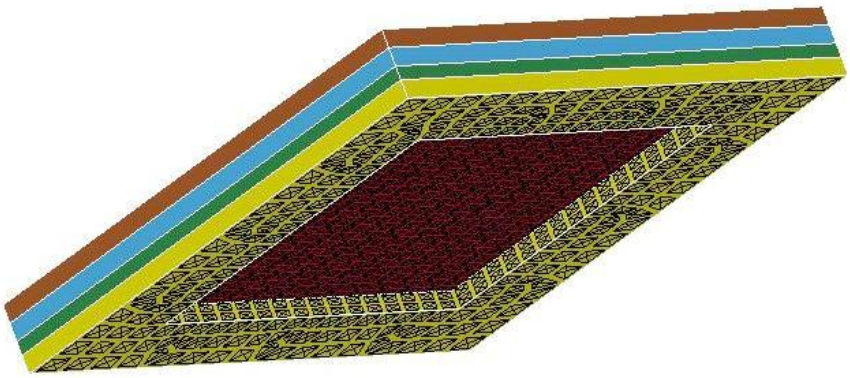

Figure **2-3**: Blast segment in the assembly .

Plate of interest (panel) and the surrounding filler plate are made of Aluminum 5083 with the bilinear material model. Material properties of the panel (and also of the filler plate) are listed in the Table **2-2** .The *MAT_PLASTIC_KINEMATIC input card is used. Other parts in the assembly are considered with *pseudo* material properties which make them *almost rigid* bodies having very minimal deflection values when subjected to the present loading conditions. After some trial and error, a high value of Young's modulus of 2080GPA seemed to be appropriate and

is used while modeling. Other properties like Density, Poisson's ration are taken to be equal to that of steel 4340.

Table **2-2**: Material Properties of Aluminum 5083 [23].

Property	*Value*
Mass Density	2700 kg/m^3
Young's Modulus	68.9 GPa
Poisson's Ratio	0.33
Yield Stress	225 GPa
Tangent Modulus	633 GPa
Hardening Parameter	1.0
Failure Strain	0.39

All parts are meshed using eight-noded solid elements. Further, it is desirable to establish a model that is computationally efficient and is accurate enough to capture the response trends correctly for iterative optimization. With respect to this problem, the displacement and plastic strain predictions are important. The panel (with springs along the boundaries) with different mesh densities is studied. Figure.**2-6** and Figure.**2-7** show how displacement and plastic strain varies with mesh densities. The mesh densities are represented as an array of three numbers [*p q* r] where, *p* represents elements along *x*-direction, *q* represents elements in *y*-direction and *r* elements in *z*-direction. Since the panel is square, in all the cases *p=q.* It can be observed that peak displacement is not sensitive to mesh refinement, while maximum plastic strain is. Computation time for this model (plate with springs) varies from 10 seconds to 820 seconds. Similar type of results are seen when the complete *grip* model is used. For such a FE analysis time varies between 90 s for the 4862-element model to 15 min for the 51902-element model,

when LS-DYNA is executed on an Intel P4-3.6 GHz machine with 3 GB RAM. The simulation time is taken to be the time required for plastic strain to stabilize at a constant value, which is taken as 11e-03 s. The coarser FE model is chosen here *with appropriately reduced plastic strain limit* for constraint evaluation in the optimization problem. This allows repeated analyses involved in the optimization procedure. The whole assembly is free to move during blast. That is, the assembly is unconstrained. To eliminate the rigid body component, the $w(\mathbf{x})$ in the objective function in (Equation 2-1) is taken to be the 'relative' displacement, obtained by subtracting the nodal displacement at a point in the armor plate from the displacement of a reference node in the grip. Since most of the parts in the grip do not experience any deflection, any displacement registered in the output file represents the motion of the assembly itself.

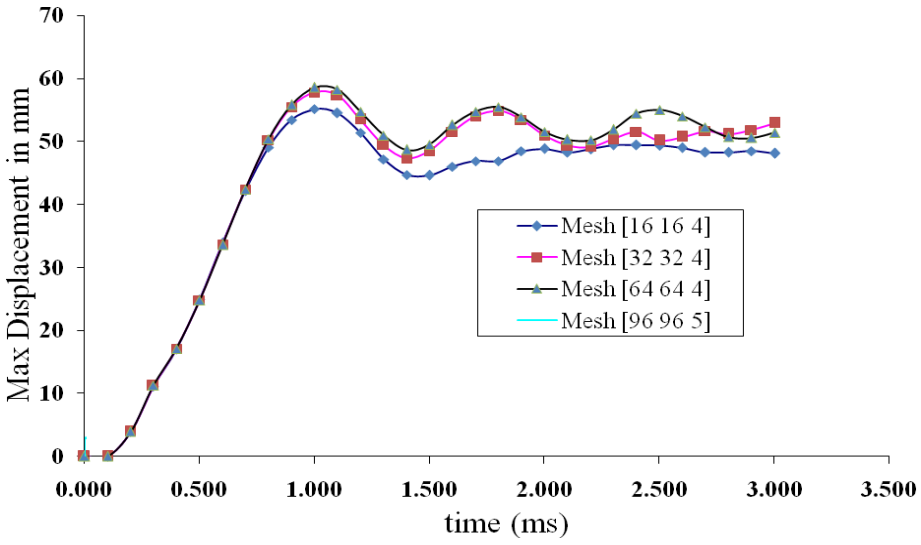

Figure **2-6**: Time history plot of max displacement for different mesh densities .

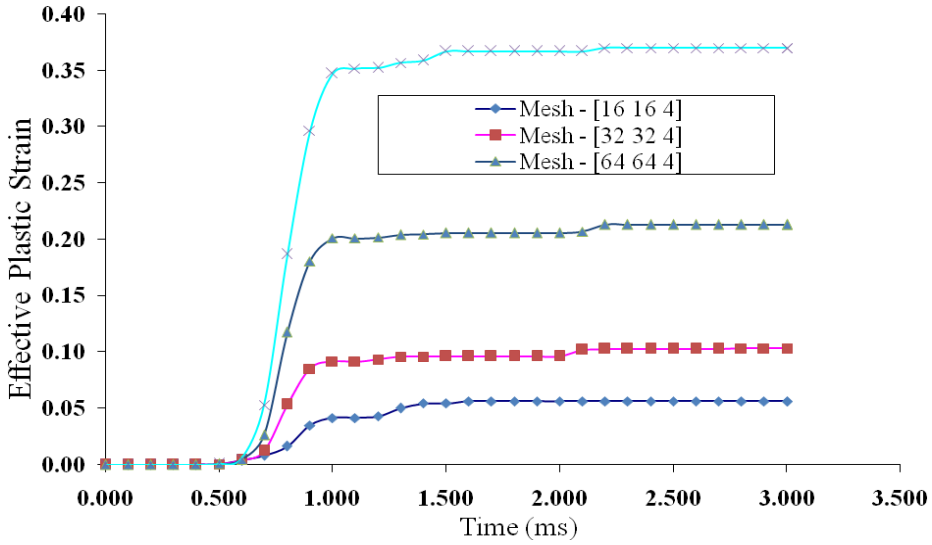

Figure **2-7**: Time history plot of plastic strain for various mesh densities .

Chapter 3

Shape Optimization

Preliminary Results

Prior to using formal optimization, ad-hoc studies were carried out to generate improved shapes. A square plate of side dimension 0.48m and thickness 0.01m with springs along the boundaries is used for these studies. Two alternative designs are considered with two orientations. Each *unit cell* in the plate is shown in the Figure **3-1**. The unit cell is flat on one side and has a depression on the opposite side.

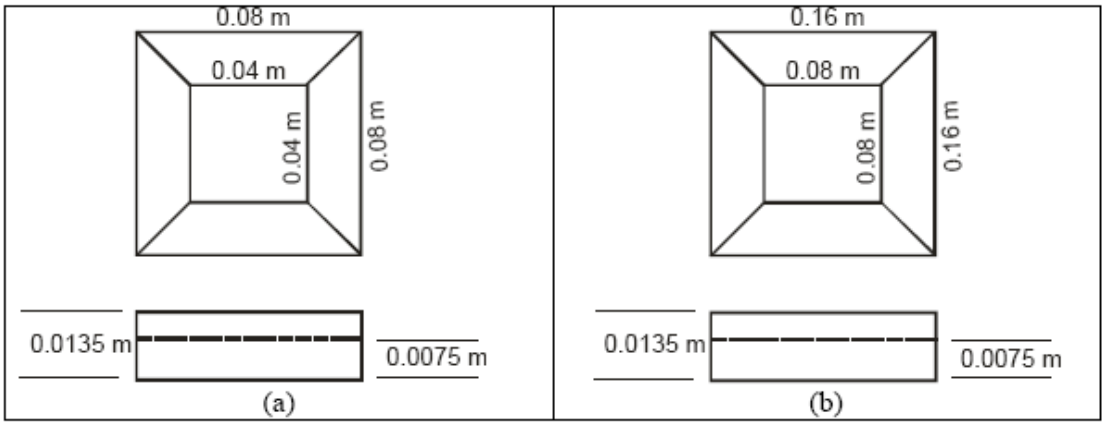

Figure **3-1**: Unit cell top view and side view .

The two orientations of the plate are shown in Figure **3-2** : (a) Blast loading is placed on the flat side and (b) blast loading on the side with depression. But LS-DYNA analysis of these two designs did not give any better results compared to flat plate. In fact, these plates were ruptured when subjected to blast. Hence, a dimpled (indented) plate, with dimples facing towards or away from the charge does not give any good result. Moving further, in order to test the effect

of *increased stiffness*, corrugated plates were subjected to blast loading. One of such corrugated plate is shown in Figure **3-3**, but this does not give any reduction in the deflection.

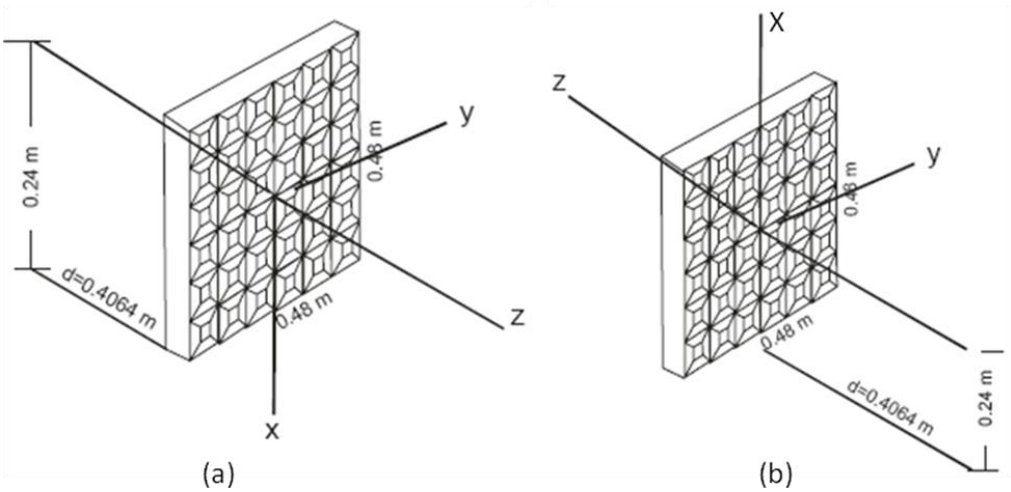

Figure **3-2**: Two orientations of the plate

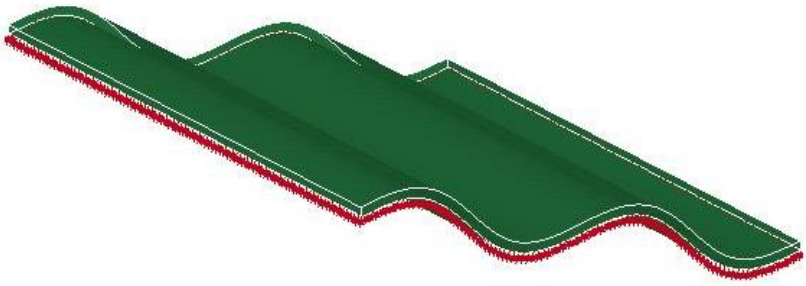

Figure **3-3**: Singly corrugated plate

These ad-hoc attempts, however, did not produce any reduction in maximum deflection and, moreover, lead to rupture levels of plastic strains. This study motivated to use formal optimization methodology. Shape optimization relates to determining the outline of a body, shape

etc. In other words, *x*-,*y*- and *z*- coordinates of the nodes or grid points in the finite element model are changed in every iteration [24].

Formal Optimization

A square portion of the model at the center of the plate is taken to be the domain of shape optimization. That is, shapes changes only occur in this region. This ensures that changes in armor plate shape do not result in changes in the grip system (see the circular blue cover plate in Figure.**2-3**). The basis of selection of the 'domain' is based on the fact that grip cannot be *specific* for a panel. However, a thickness change in the plate has to be matched by an equal thickness change in the filler plate for the assembly to function. In shape optimization, it is important to come up with a way to relate the grid points to a finite number of parameters $x_i, i=1,2,\ldots..N_{dv}$. This process is called **Mesh Parameterization**. A velocity field $\mathbf{q^1}$ relates x_1 to changes in z-coordinates of all nodes in the model as

$$\Delta \mathbf{G} = x_1 \mathbf{q^1} \qquad 3\text{-}1$$

Using the Equation.**3-1** we can arrive at an equation which gives $\mathbf{G}(x_1)$ as given below.

$$\mathbf{G}\,(x_1) = \mathbf{G_{original}} + x_1 \mathbf{q^1} \qquad 3\text{-}2$$

In case of many such design variables representing changes in many coordinates in different ways, new grid is calculated by generalizing Equation **3-2**.This leads to Equation. **3-3** , which is the key equation to implement shape optimization [24-25]**.**

$$\mathbf{G}(\mathbf{x}) = \mathbf{G}_{original} + \sum_{i=1}^{N_{dv}} x_i \ \mathbf{q}^i \qquad 3\text{-}3$$

Each x_k represents the amplitude of a 'permissible shape change vector' or what is commonly called a 'velocity field' vector $\mathbf{q^k}$. Velocity fields have nothing to do with actual velocities of the model under loading. The name comes from Equation **3-4** which is a different representation of Equation **3-4** describing the motion of the continuum with x_i as time parameter and $\mathbf{q}^i$ as velocity vector [24]. Vectors $\{\mathbf{q}^k\}$ are generated outside the iterative optimization loop. $\mathbf{G}_{\text{original}}$ is the current (flat) shape. Visualization of a $\{\mathbf{q}^i\}$ is identical to visualization of a displacement field in finite elements: $\{\mathbf{q}^k\}$ is multiplied by a magnification scalar and added to the current grid to obtain a displaced grid, except that here the displaced grid represents a new shape and is called a *basis shape*.

$$\frac{\partial \mathrm{G}}{\partial x_i} = \mathrm{q}^{\mathrm{i}} \qquad 3\text{-}4$$

The role of the optimizer is to choose $\mathbf{x}^*$ so that the corresponding shape $\mathbf{G}(\mathbf{x}^*)$ is optimum. As **x** is iteratively changed by the optimizer, the grid point coordinates **G** are updated, a FE input file is then written and an analysis is carried out to evaluate the various functions in the optimization problem. This flow of information is shown in Figure**3-4**. However this is not the exact flow chart of the code implementing shape optimization, but this is given for better understanding of shape optimization methodology

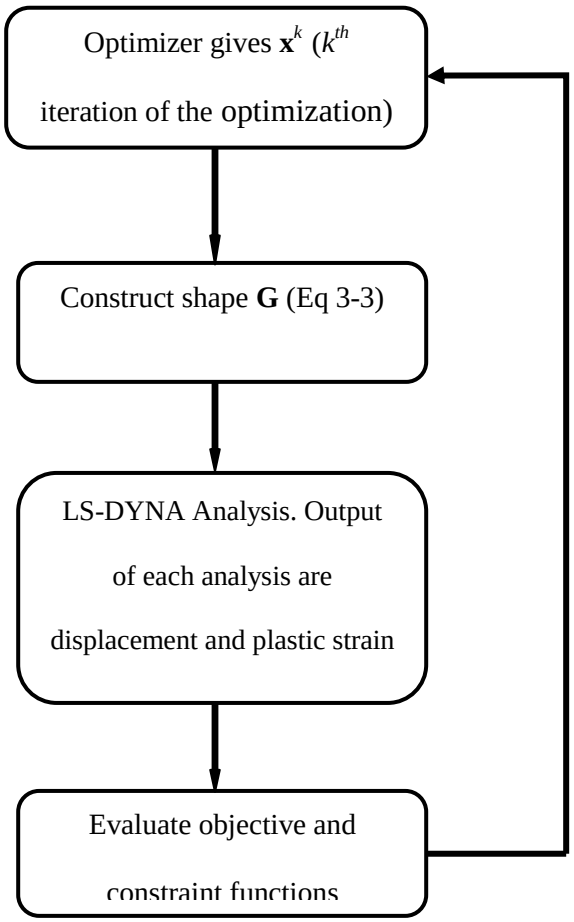

Figure **3-4**: Flow diagram of the information

Velocity Fields

Sinusoidal Velocity Fields

As explained in the previous sections, shape changes take place only in a square area, square being inscribed in the circular shape of the grip. The area with changes in the shape is given in Figure **3-5** for better understanding of the domain of shape optimization. Sinusoidal velocity fields for the top and bottom surfaces, independently, are chosen based on Equation **3-5**

$$f(m,n) = C \sin\frac{m\pi x}{L} \quad \sin\frac{n\pi y}{L} \qquad 3\text{-}5$$

where C is a suitable normalization factor and m, n are integers taking on values 1, 2, 3, etc. In this thesis, optimization results are presented for two cases of sinusoidal basis shapes: 3DV and 9DV.

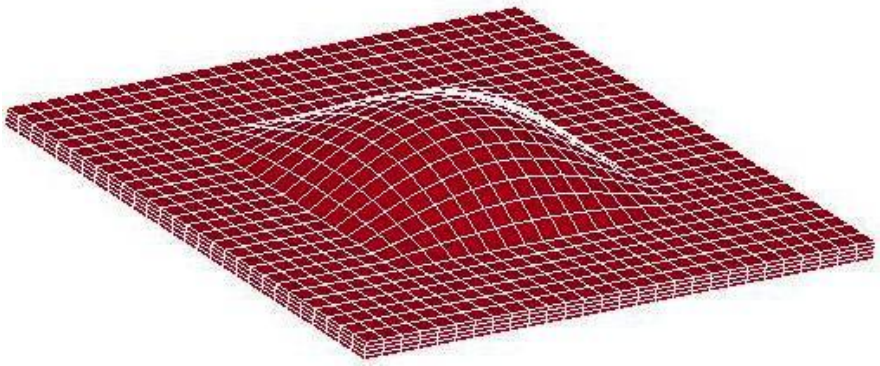

Figure **3-5**: Domain of the shape optimization .

3Design Variables:

$m = n = 1$. This gives a total of three (3) *symmetric* basis shapes corresponding to

$\mathbf{q}^1 \equiv f(1,1)$ for the top surface,

$\mathbf{q}^2 \equiv f(1,1)$ for the bottom surface,

$\mathbf{q}^3$ = thickness change

Specifically, $\mathbf{q}^1$ represents a bulge in the shape of the top surface while bottom surface is fixed (other thru-thickness nodes are moved to preserve equal spacing), $\mathbf{q}^2$ represents a bulge on the bottom surface while top surface is fixed, and $\mathbf{q}^3$ = a thickness change only (that is, middle layer

of nodes in the plate are fixed while top and bottom surfaces move in opposite directions). Thus, the design variable vector is $\mathbf{x} = [x_1 , x_2, x_3]^T$, and the optimizer tries to determine an optimum combination of these three basis shapes. Basis shape corresponding to $\mathbf{q}^1$ is illustrated in Figure **3-6** Note that the bulge can be positive or negative depending on the value of x_i

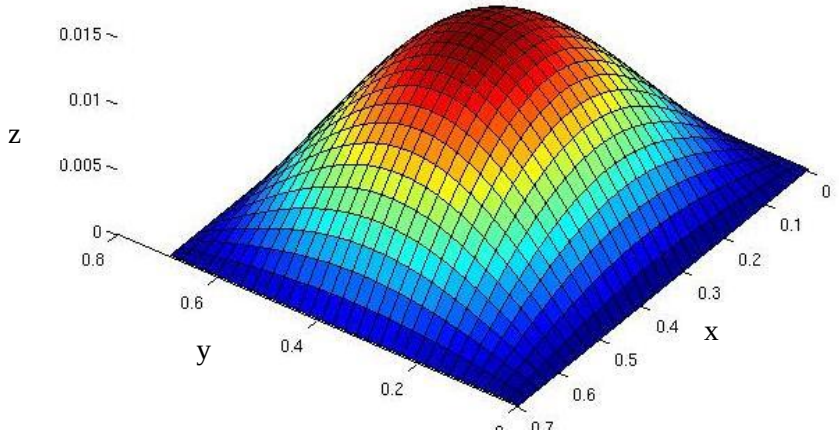

Figure **3-6**: Basis shape corresponding to top surface bulge for 3DV case

9Design Variables

$m = n = 2$. This gives a total of nine (9) basis shapes corresponding to

$\mathbf{q}^1 \equiv f(1,1)$, $\mathbf{q}^2 \equiv f(1,2)$, $\mathbf{q}^3 \equiv f(2,1)$, $\mathbf{q}^4 \equiv f(2,2)$ for the top surface,

$\mathbf{q}^5 \equiv f(1,1)$, $\mathbf{q}^6 \equiv f(1,2)$, $\mathbf{q}^7 \equiv f(2,1)$, $\mathbf{q}^8 \equiv f(2,2)$ for the bottom surface,

$\mathbf{q}^9$ = thickness change

Thus, **x** is a (9x1) vector. Unlike the 3DV Case above, presence of asymmetric basis shapes may lead to an asymmetric final optimum shape. $\mathbf{q}^1$ is same as Figure **3-6** , $\mathbf{q}^2$, $\mathbf{q}^3$ and $\mathbf{q}^4$ are given in Figure **3-7**, **3-8** and **3-9** respectively. Other basis shapes are similar to these with bottom surface taking these shapes, hence not given here.

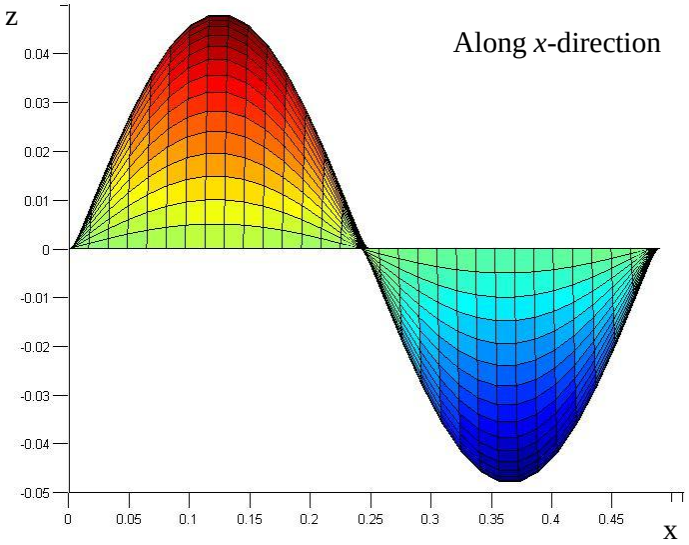

Figure **3-7**: $\mathbf{q}^2$ basis shape for 9DV case .

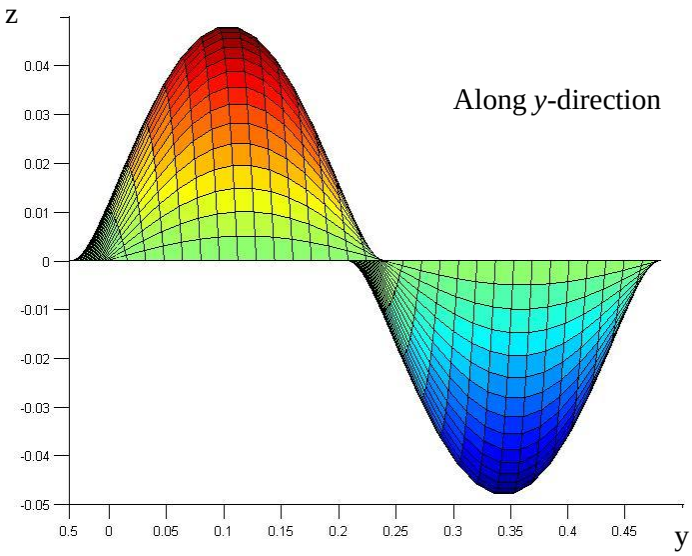

Figure **3-8**: $\mathbf{q}^3$ basis shape for 9DV case .

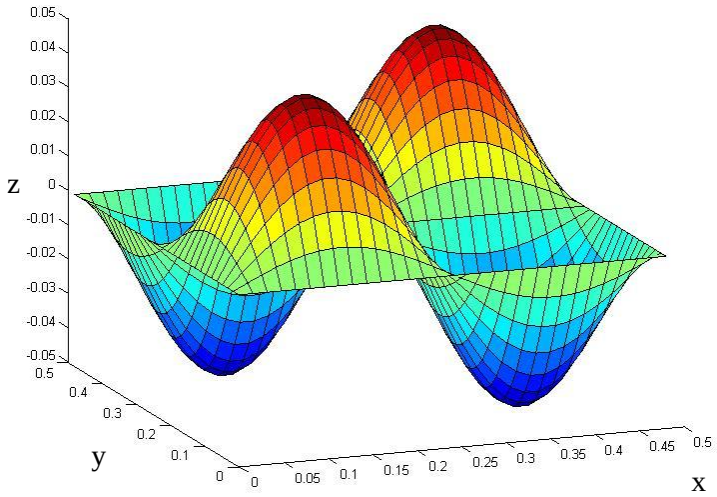

Figure **3-9**: $\mathbf{q}^4$ basis shape for 9DV case .

Local Point Load Velocity Fields

In the last section generation of sinusoidal velocity fields are explained. It is very important to note that higher the number of design variables, higher the amount of distortion caused to the finite element model by the basis shapes. Hence, a different approach of generating velocity fields is considered using local point loads. Using the closed form deflection equation for simply supported rectangular plates [27], deflection at any point (x,y) for a point load P applied at (η,ξ) can be calculated. Consider a rectangular plate as given in Figure **3-10.**

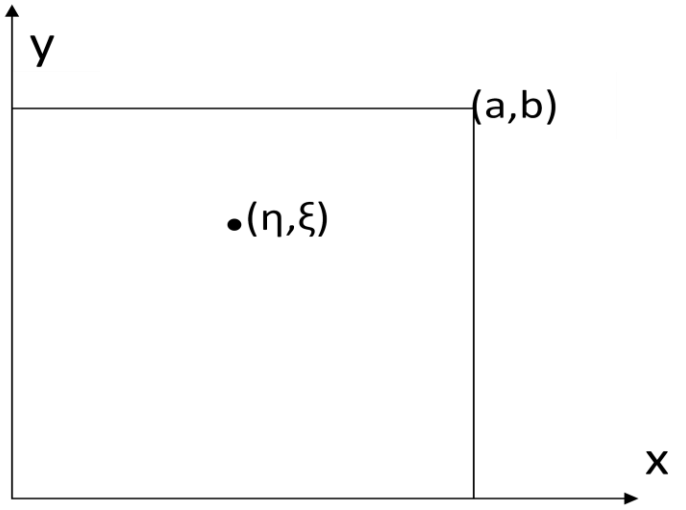

Figure **3-10**: Schematic diagram of a Rectangular plate .

Deflection, w is given by the following equation

$$w = \frac{Pa^2}{\pi^3 D}\sum_{m=1}^{\infty}\left(1+\beta_m \coth\beta_m - \frac{\beta_m y_1}{b}\coth\frac{\beta_m y_1}{b} - \frac{\beta_m \eta}{b}\coth\frac{\beta_m \eta}{b}\right)\frac{\sinh\frac{\beta_m \eta}{b}\sinh\frac{\beta_m y_1}{b}\sinh\frac{m\pi\xi}{a}\sinh\frac{m\pi x}{a}}{m^2\sinh\beta_m}$$

3-6

$$where\ \ \beta_m = \frac{mxb}{a}\ \ , y_1 = b - y\ \ , y \geq \eta$$

If $y < \eta$ the quantity y_1 must be replaced by y, and the quantity η by $\eta_1 = b - \eta$ in the above expression.

For a square plate as in the present case, a = b. Different velocity fields can be generated by considering P to be applied at different values of (η,ξ). Each location of (η,ξ) generates one velocity field. We consider five different locations of (η,ξ) corresponding to center of the plate and center of each quarter of the plate. Deflection profiles are same but with opposite sign when these locations are on the other side of the plate. Thus this leads to 5+5+1(for thickness) = 11 velocity fields ($\mathbf{q}^i$,i= 1,2,..11). Figures **3-11** to **3-15** show these velocity fields.

Figure **3-11**: Velocity field , $\mathbf{q}^1$ for a point load at the center of the square plate .

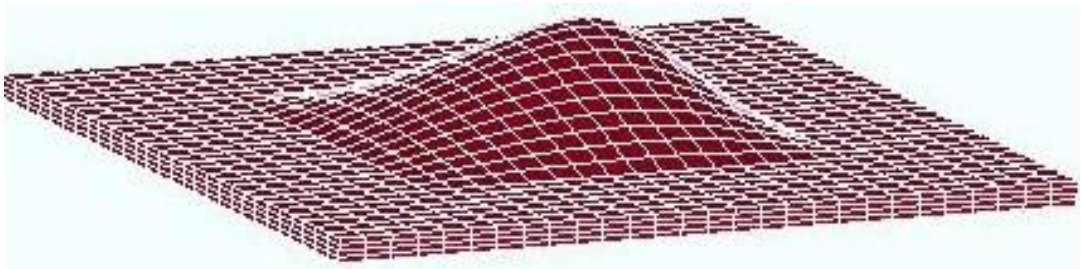

Figure **3-12**: Velocity field , $\mathbf{q}^2$ for a point load at $(\eta,\xi) = (0.75,0.75)$ (Note : (η,ξ) for farthest corner is (1,1)).

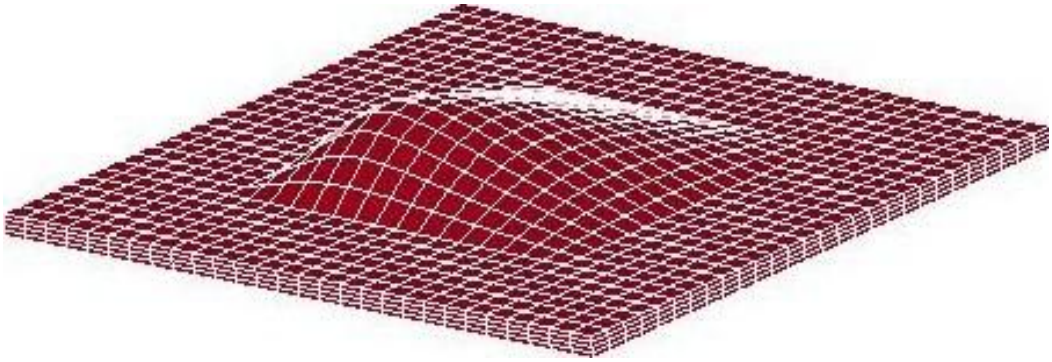

Figure **3-13**: Velocity field, $\mathbf{q}^3$ for a point load at $(\eta,\xi) = (0.25,0.25)$.

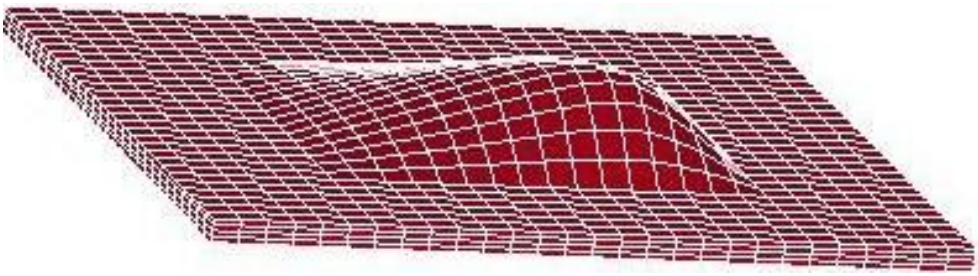

Figure **3-14**: Velocity field, $\mathbf{q}^4$ for a point load at $(\eta,\xi) = (0.75,0.25)$.

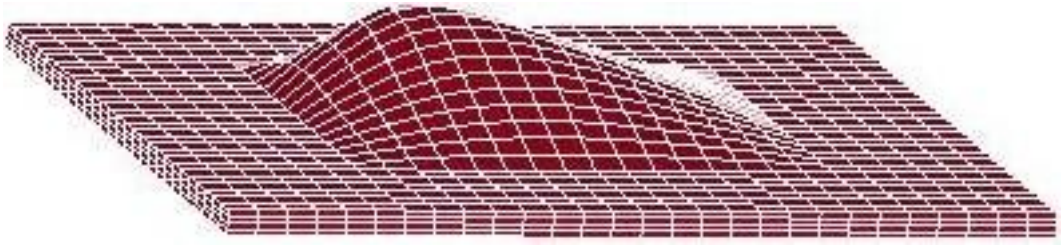

Figure **3-15**: Velocity field, $\mathbf{q}^5$ for a point load at $(\eta,\xi) = (0.25,0.75)$.

Differential Evolution with Coarse Parallelization

Computational experiments showed that downhill or descent search directions did not always lead to a reduction in the objective function even for small steps when using a gradient-based optimizer. The problem was seen to be clearly non-differentiable. The stochastic Differential Evolution (DE) optimizer was chosen for this work. From a computational point of view, DE is similar to GA. An initial population of designs is generated, function evaluations are carried out for each member in the population followed by mutation and crossover operations, leading to a new generation. Salient aspects of DE are:

(i) Function evaluation (here, a finite element analysis with Ls-Dyna) of a population of designs can be done independently, on separate processors. Thus, a computer cluster is very attractive to reduce time (as with GA), as clearly, the most time-consuming step is the function or fitness evaluation (FEA).

(ii) The total number of analyses equals product of population size and number of generations. Thus, the total computing time for a job can be ascertained ahead of time in terms of n_p = number of processors in the cluster, n_{pop} = population size, and *ngen* = number of generations (as with GA).

(iii) Real-valued design variables are stored as such, whereas in GA, a conversion to binary or other representation is usually involved.

(iv) The mutation and crossover operations are different than GA. Specifically, the following is one strategy used for mutation and crossover:

$$\mathbf{v} = \mathbf{x}_i + \lambda\,(\mathbf{x}_{\text{best}} - \mathbf{x}_i) + F\,(\mathbf{x}_{r2} - \mathbf{x}_{r3}) \qquad 3\text{-}7$$

$$\mathbf{x}_{i\_}\,\text{new} = (1\text{-}p)\,\mathbf{x}_i + p\,\mathbf{v} \qquad 3\text{-}8$$

where $\mathbf{x}_i$ = member in the population, $r2$ and $r3$ are random numbers pointing to random members in the population, λ and F are user-defined parameters, p = probability factor, say 0.5. In view of (i) above, a multi-processor computing cluster is used. Various literatures on DE parallelization is given below.

There are two popular approaches for DE parallelization. First employs fine grained parallelism, also known as Master-worker model [27]. This is the simple method where master process controls the program by assigning tasks to worker processes. Typically, any function evaluation is done by workers while master generates new population using the information from workers. The second model [28] divides population into subpopulation and associates every entire subpopulation (also called *islands*) to a process. Each subpopulation converges to its own solution which is then exchanged between processes. Periodically, best individual from each process is moved to other process (migration). Both these approaches have been implemented and used successfully in solving various types of problems [28-30]. In a report from NASA-Langley committee chaired by Biedron *et al* (1999) [31] different strategies for parallelizing an applications have been discussed. They are as follows:

1. Coarse-grained parallelization a code replicated and executed with different inputs
2. Coarse-grained parallelization by partitioning of the analysis domain
3. Fine-grained parallelization of an existing code
4. Fine-grained parallelization by recoding the solution algorithm from ground up
5. New paradigms to replace existing solution algorithms with new algorithms that are intrinsically parallel

Option # 1 is the most general implementation when there is a repeated execution of a code with different inputs. Basically, every processor is going to take different input and overall time is reduced. Example for such a problem will be solving a FE problem for different values of load. Since optimization algorithms like GA or EAs have independent evaluation of many points in the design space and due to the advantage of minimum human interface this is usually preferred to exploit the computational facilities available. Option # 2 requires dividing the domain (typically mesh or grid) among processors. This creates the need of communication of unknowns (like boundary information) among processors, which increases with the number of processors being used. This dependency creates hurdles in obtaining better speedup. Option # 3 and 4 requires careful observation and time profiling of the code to see the potential part to be re-written to parallelize it. But these are well utilized with combined with Option # 1 or 2. Option # 5 calls for invention of new algorithms. As a conclusion, coarse grained parallelism is considered as an efficient and cost effective method for search methods like GA and EAs, response surfaces and neural nets. However, in this paper, a hybrid methodology of fine and coarse grained parallelism is implemented with overall code running as fine grained but population evaluation part of DE implemented as coarse grained.

DE has been parallelized successfully on parallel virtual parallel environment in a study which showed that extent of information exchange among subpopulation associated with

every process influences the performance significantly [28]. This study also showed that mutation strategy selection is important in achieving optimum performance. In this work, population is divided into subpopulation which is then mapped to a processor. Each sub-population reaches solution independently. In order to get better solutions, best individuals are moved to other sub-populations in a ring topology by a process called 'migration'. It introduced the concept of migration constant which is randomly chosen and dictates the movement of best individual based on some rule. Hence there is probability that every best individual is given to the neighboring sub-population, 0.5 being the best for the test problems considered (sphere function, Rosenbrock's saddle, Corana's parabola, Step function, Quartic function, Shekel's foxholes and Griewangk's function). Another work on parallelizing DE aimed at large data sets especially for image processing applications with a set of patterns called *learning sets* [32] which are nothing but set of patters. Fitness calculation for the trial population is carried out by *pipeline* approach in which each process computes *local fitness* for a subpopulation. All the processes have non-blocking communication setup in *ring* topology followed by collective reduction operation used to compute *global fitness* achieving almost linear speedup for an experimental problem involving well known *lena* image. The proposed algorithm was run for varying size of DE population and number of clusters which allowed testing the algorithm using a wide range of "granularity of the communication pattern" (Granularity is measured by the length of shortest interval between two consecutive message exchanges). In a different work [33], super linear speedup was obtained by master slave parallelization of DE for two classes of problems: Hang function approximation problems and the Mackey-Glass time series prediction problem. This study basically aimed at implementation of parallel evolutionary asymmetric subsethood product fuzzy neural network in LAM/MPI cluster using master-slave approach.

In all the above mentioned parallelization approaches, time taken by each function evaluation varies in a narrow range. In the present case, different implementations have/need to

be researched owing to different computing times for each member in the population, the latter occurring due to two main reasons:

A) Due to finite element distortion. Specifically, each design in the population corresponds to a generally different shape or grid of nodal coordinates as per Eq. **2-1**. The Jacobian for each finite element is computed at the eight nodes for each hexahedral element. If it is negative at any node, then the element is distorted (non-convex) and finite element analysis (FEA) cannot and is not carried out. A suitably large objective function value is simply returned to the optimizer. Thus, the computing time is essentially zero. It is not possible to avoid this, as new member designs are obtained using a random procedure (Eq **3-7** and **3-8**). On the other hand, if the mesh is undistorted, then typically the FEA takes, say, 5 minutes for the mesh size used in this paper. Hence, a large variation in times for each function evaluation exists depending on whether the mesh is distorted or not.

B) Each function evaluation involves solving differential equations in time. The maximum time step depends on the thickness and more generally on element size and shape. The time per analysis can vary by factors of 4:1.

Two parallelization schemes have been implemented, herein called the SATR and LB. Each scheme was successively motivated by the performance of the previous scheme. Details are as given below.

Send-All-Then-Receive (SATR) Scheme

In this scheme [27], master process divides all the members of the population equally among all the processors. It first sends values of all the design variables associated with each member of the population to appropriate workers. Every worker, once it receives the

information, carries out the function evaluation. After finishing all the function evaluation assigned, each worker sends the function values to master. Even though this is a simple scheme, due to variation in the computation time for each function evaluation, there can be unnecessary wait time for the processes that have finished their function evaluations successfully or cannot evaluate due to problems with the FE model distortion. This can be illustrated with the following sample data. Let n_{pop} = 8, n_p = 4, and FEA time for each member in the population as given in Table **3-1** below. The '0 min' FE time indicates distorted meshes for which no FE is carried out.

Table **3-1**: Sample Data to Illustrate Parallel Schemes .

Member	1	2	3	4	5	6	7	8
FE time, min	2	0	6	3	0	3	2	0

From Table **4-1** we see that a single processor will take 16 min for all 8 evaluations. With 4 processors, with P1 acting both as a master and a worker, the distribution of computations is as indicated in Figure. **3-16** below. The 8 function evaluations are distributed in two stages to the 4 processors. Thus, total time for all 8 evaluations equals 6+3 = 9 min. Thus, actual speedup is 16/9 = 1.78, much less than ideal of 4.0 which would occur if each FEA takes equal time

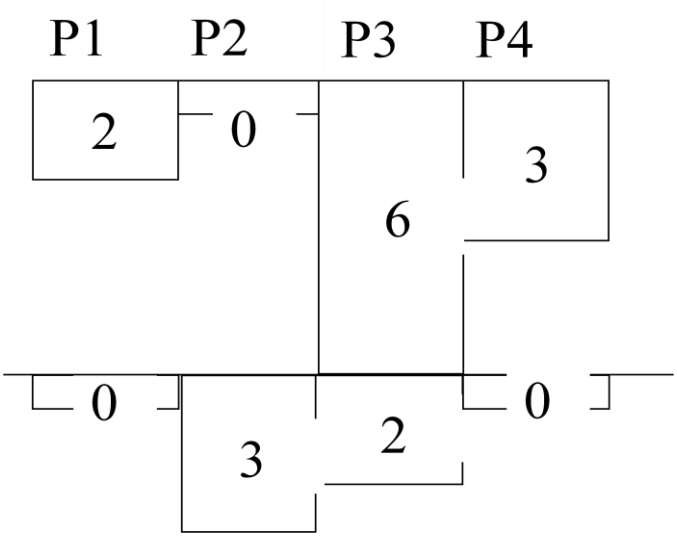

Figure **3-16**: SATR approach - numbers refer to minutes, and P*i* refers to processor *i* .

Load Balanced (LB) scheme

A modified scheme [3] is used where send and receive takes place one after another depending on how the availability of the worker process. The master process sends only one individual (of the population) to each worker initially. Once the worker process completes the function evaluation and communicates the function values to the master process, the next individual in the population is sent to the worker for evaluation. In other words, the master process continues to send and receive until all members of population have been evaluated. The overall algorithm is as follows:

Master process

1. Broadcast details of entire population (design variable values) to all the worker processes.
2. Set number of members already evaluated, $n_{sent} = 0$.
3. Loop through $i = 1,\ldots,\min(n_{pop},n_p)$.
4. Increment n_{sent}.

5. Ask process *i* to evaluate member n_{sent}.
6. End loop
7. Loop through all the members of the population, $j = 1,2,\ldots,n_{pop}$
8. Receive function values from process *i.*
9. If $n_{sent} < n_{pop}$, increment n_{sent} and ask process *i* to evaluate member n_{sent}. Else inform process *i* that there is no more function evaluation to conduct (**no-more-work** message).
10. End loop.

Worker process (only for process $i < n_{pop}$)

1. Receive the broadcast message and store all the design variables.
2. Loop till **no-more-work** message is received from the master process.
3. Receive the population index, compute and send function values to master process.
4. End loop.

It should be noted that only the master process executes DE and takes a very small compute time compared to the time taken for a typical function evaluation. On the sample data in Table **3-1**, with P1 acting as the master and P2-P4 doing the FEA, Figure. **3-17** shows the distribution of computations. We see that maximum time is 6 min. Thus, even with one processor not sharing in FEA, the speedup is 16/6 = 2.67 > 1.78 by SATR approach.

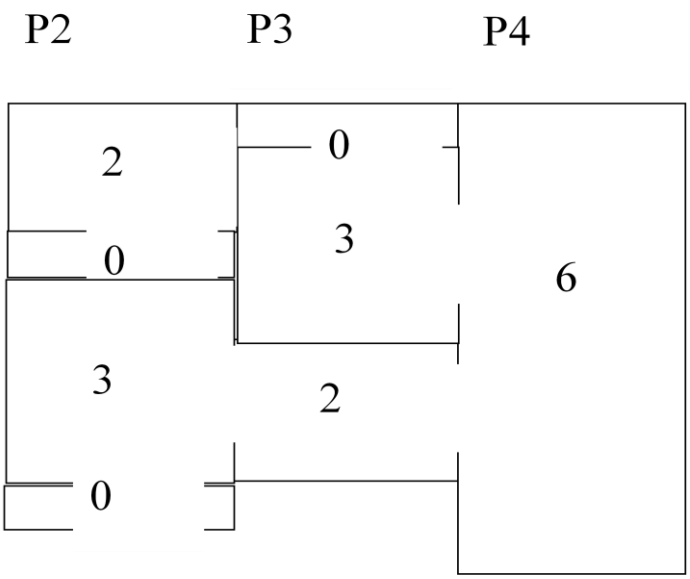

Figure **3-17**: LB (load balanced) Scheme on example; numbers refer to FEA time (min), and P*i* refers to processor *i*

This parallel algorithm was implemented and executed on the LION-XC cluster located at Pennsylvania State University's Research Computing and Cyberinfrastructure. The compute nodes are dual 3.0-GHz Intel Xeon 3160 (Woodcrest) Dual-Core Processors each with 16 GB of ECC RAM and with Myrinet interconnect. The program was written in FORTRAN and message passing was achieved via MPICH2 calls.

Computer Code Development

An ANSI Fortran code is developed which connects LS-DYNA and optimizer. Overall flow of this code is given in Figure **3-18** Function evaluation process starts with writing an input file for running LS-DYNA which is checked for mesh distortion by calculating the determinant of Jacobian. High function value is returned as the value of *fitness* for distorted mesh or if the thickness of the model is less than a critical value. For feasible design, Dyna solver is invoked.

Objective function is calculated from 'nodout' and 'elout' files written by LS-DYNA. Visualization of the result is done through LS-PrePost.

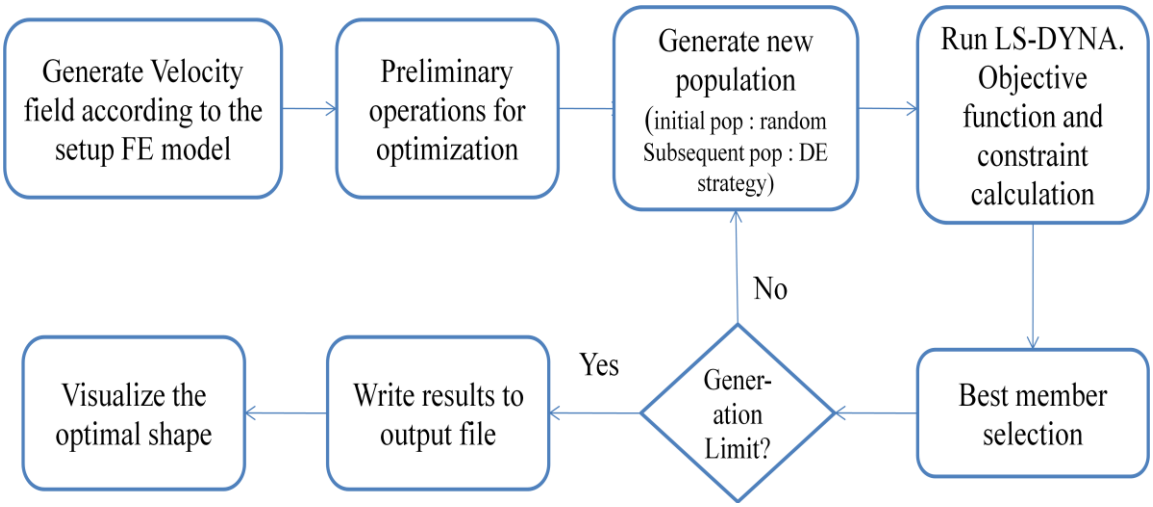

Figure **3-18**: Flow Diagram of the computer code for shape optimization

A key element of the implementation lies in the creation of two types of input files, as discussed below.

(A) DESIGN-OPT INPUT FILE

This file contains the following important inputs:

i. seed used for DE
ii. generation limit,
iii. population size,
iv. geometric envelope limits (limits for the z-coordinate of the nodes)
v. maximum mass limit,
vi. plastic strain limit,
vii. upper and lower limit of design variable,

viii. velocity fields

ix. model related inputs

(B) Ls-Dyna .k INPUT FILE

i. charge data

ii. material properties

iii. nodal coordinates and element connectivity

Typical values used are given in the Table 3-2 . Note that plastic strain limit is less than the material property, to account for the fact that a coarser mesh is being used for optimization (based on the discussion earlier in the paper). Coordinate files are written for every population generated and checked for distortion of the mesh. High function value is returned for a distorted mesh; otherwise the Ls-Dyna solver is invoked. DYNA writes nodal and element related time history outputs to ASCII files called ‘nodout’ and ‘elout’ respectively. FORTRAN routines are written to open these files and calculate relative displacement values and constraint values which are added to objective function as quadratic penalties. Based on the objective function values of all the population members and previous generations, best member is selected and stored. After the last generation, best member is written to the output file along with max displacement value and values of plastic strain. The visualization of the results is through LS-Prepost.

Table **3-2**: Typical values of input parameters used in the input file

Parameter	*Value*
Generation limit	20 to 45
Population size	30 to 200
Envelope limits (*m*)	-0.3 to 1.0 (large) -0.1 to 0.14 (small)
Mass of the assembly (*kg*)	1890.0
Plastic strain limit	0.15
Scaling factor, *C*	0.02
Seed	1170, 2349

Chapter 4

Numerical Results

Optimized Shape with 3 Design Variables

Shape optimization results for the square aluminum plate model using three design variables, or equivalently, three basis shapes is discussed in this section. Figure **3-6** gives the figure of 3DV. Initial baseline (flat plate) model and the resulting optimum shape of the panel obtained using DE is shown in Figure.**4-1** . The optimized shape turns out to be a double bulge (Figure. **4-1a** and **4-2**). Overall thickness is reduced from 0.0381m to 0.0241m with the introduction of bulges on both sides to keep total mass within limits. The result shown in the figures is for the case 'large envelope' where geometric envelope limits is fairly large (relaxed) on both sides.

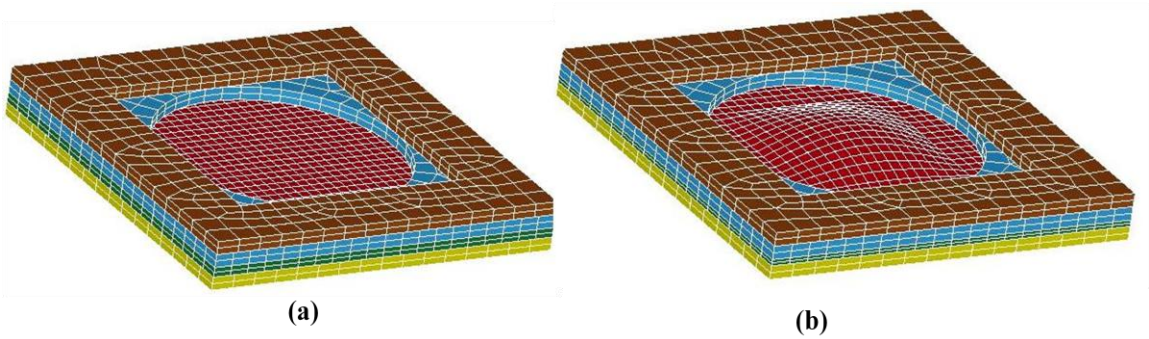

Figure **4-1**: (a) Baseline design(uniform thickness) , (b) Optimized plate-double bulge (for large envelope case).

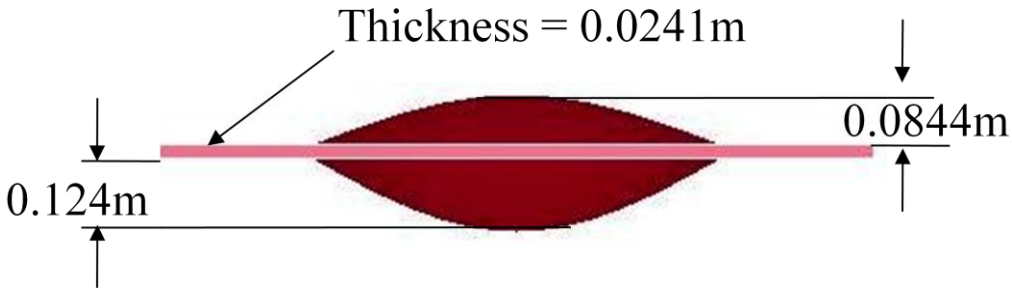

Figure **4-2**: Dimension of the optimal design panel (for large envelope case).

Lower envelope limit cannot be larger than 0.3m due to presence of the charge. With a reduced envelope limit of (-0.1, 0.14)m an optimum double bulge *is again seen* but lies within the envelope. Table **4-1** summarizes the results of the two cases of *large* and *small* envelope.

Table **4-1**: Results for baseline design and optimized designs for 3DV case .

Property	*Baseline Design*	*Large Envelope (-0.3,1.0)m*		*Small Envelope (-0.1,0.14)m*	
		Optimized Design	*Change*	*Optimized Design*	*Change*
Objective Function (*m*)	20.51E-03	4.34E-03	78.9%	4.49E-03	78.1%
Max Relative Displacement (*m*)	58.43E-03	7.32E-03	87.5%	7.68E-03	86.9%
Max plastic strain	0.1277	0.01932	84.9%	0.02028	84.1%
Total Mass (*kg*)	1872.2	1894.57	(1.2%)	1894.54	(1.2%)
Saturated impulse, (*kg-m/s)*	6254.3	5298.62	15.6%	5425.82	13.6%

The ‘large envelope’ case is slightly better compared to ‘small envelope’ case, as is to be expected. In Table **4-1**, both RMS displacement (= the objective function to be minimized) and the maximum displacement are quoted. Plastic strain plots for baseline design and optimized design is given in Figures **4-3** and **4-4** respectively. Plastic strain is maximum at center for the baseline design while it is smeared around the borders of domain for the optimized panel.

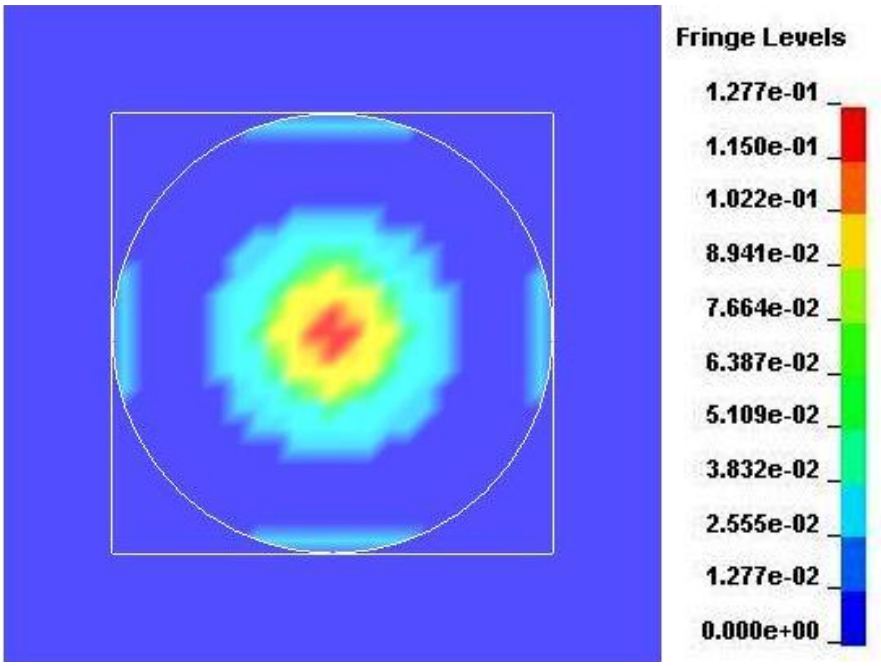

Figure **4-3**: Plastic strain plot for baseline design

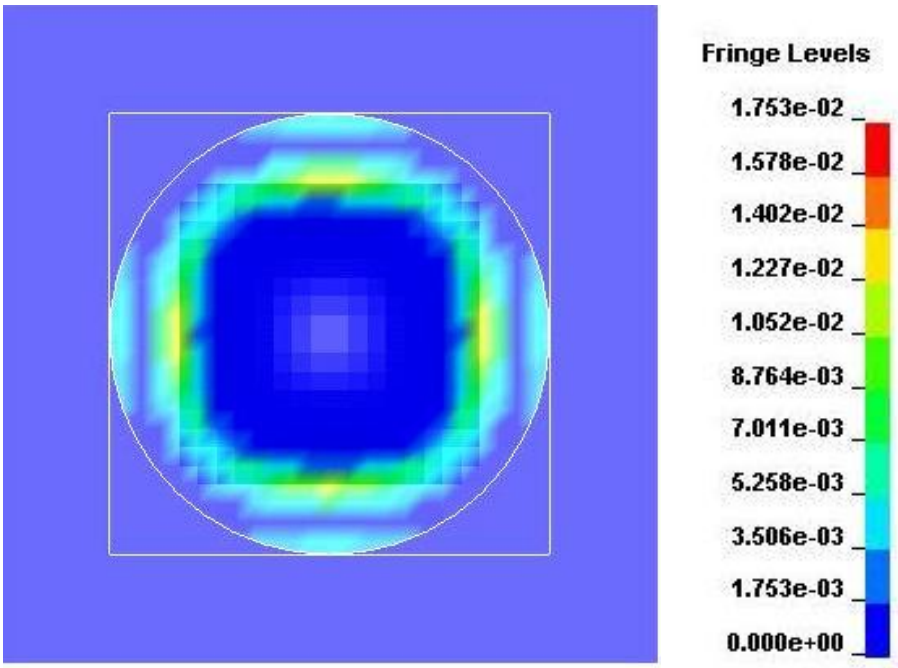

Figure **4-4**: Plastic strain plot for optimized design

Maximum relative displacement of optimized designs vs. baseline design is shown in Figure **4-5**. Displacement reaches its peak at t = 1.1e-03 s. The total impulse responses of optimized designs are compared to those of baseline design in Figure **4-6**. The saturated impulse value of 'large envelope' and 'small envelope' optimized panel designs are 15.6% and 13.6% lesser compared to baseline design, respectively. However, the reduction in displacement is near 80% which shows that there is a stiffening effect on top of blast wave flow diversion.

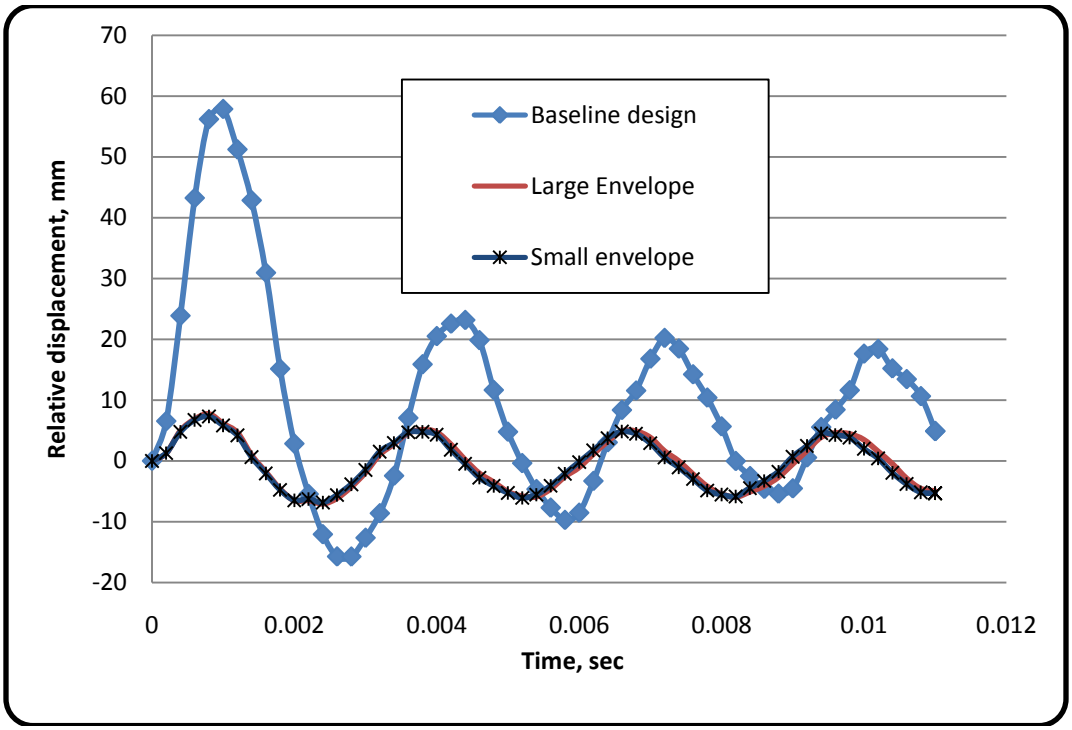

Figure **4-5**: Comparison of relative displacement .

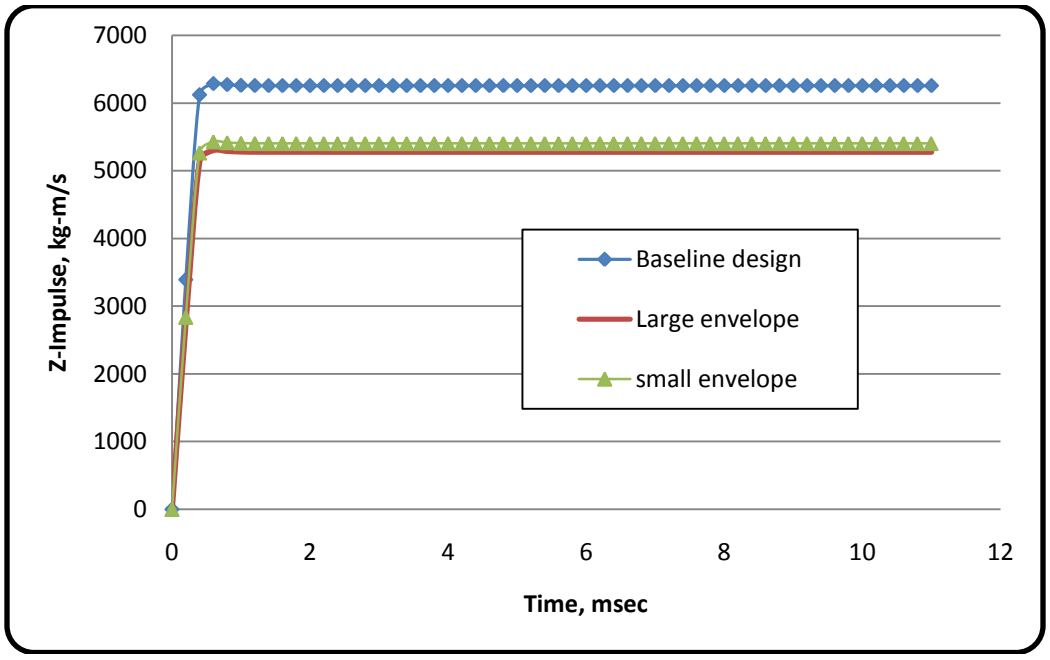

Figure **4-6**: Comparison of impulse response.

Sensitivity

Sensitivity for charge distance and charge amount is studied for 3DV case. Limited sensitivity studies were conducted to ensure that the optimum shapes did not change with ±5 % and ±10 % changes in standoff distance (z-distance of charge from plate) and also of charge amount (i.e. kg of TNT). Figure. **4-7** shows the graph of value of objective function versus standoff distance. Note that, as standoff distance varies initial objective function value also changes. For brevity, details are not given here, since the optimized shapes and results are similar except for small perturbations in values. As the standoff distance decreases, bulge towards the charge was more compared to the amount of bulge on the top face of the panel. With the increases in standoff distance, optimum shape remained same.

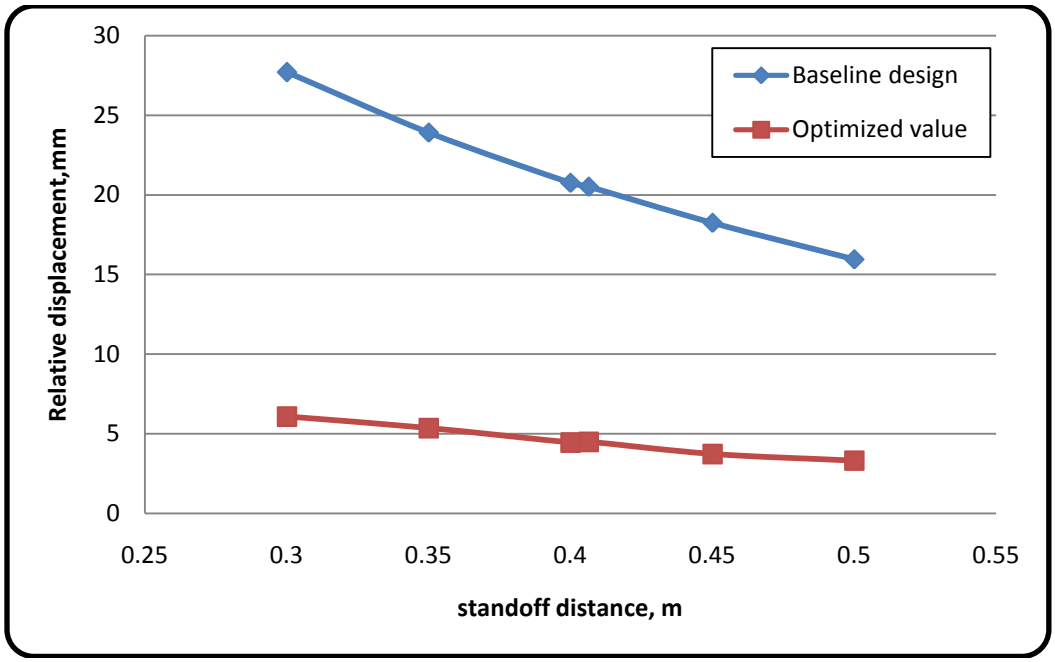

Figure **4-7**: Sensitivity of objective function for changing standoff distance

Comparison of Optimized Plate with Flat Plate of Equal Mass

It can be argued that results in previous section above are only application specific and not valid for arbitrary freely supported plates. Specifically, the mass constraint on the entire structure in Equation. **2-1** can be re-written as

$$M_{\text{plate of interest}} + M_{\text{filler plate}} + M_{\text{remainder}} \leq M_{\max} \qquad 4\text{-}1$$

Note that $M_{\text{remainder}}$, representing mass of the rigid plates, does not change during optimization, while mass of the filler plate does change owing to preserving its continuity in thickness with the plate of interest. Thus, it can be argued that mass has been transferred from the filler plate to the plate of interest. Specifically, for the 3DV case, we have

$M_{\text{plate of interest}}$ at initial baseline design = 152.9 kg ($t_{\text{filler plate}}$ = 0.0381 m)

$M_{\text{plate of interest}}$ at optimized design = 206.8 kg ($t_{\text{filler plate}}$ = 0.0241 m)

Now, a new flat plate of interest is analyzed whose mass is the same as the optimized plate, viz. 206.8 kg. This has been done by using $t_{\text{filler plate}}$ = 0.05153 m. The total mass of the structure, is maintained at 1894.5 kg, to avoid changes to total baseline impulse, by adjustment of the density of the rigid components whose properties are not relevant except with regard to providing a rigid grip to the plate. Thus, no new inertia effects are introduced. Results of the equal mass plate are given in Figures **4.8**, **4.9** and Table **4.2**. Comparing the equal mass and optimized plates, we again see a vast improvement in deflection reduction of 67% (but less than 79% as quoted in the Table 4.1). 78% reduction in maximum plastic strain and 15% reduction in impulse are again observed. Thus, there is no doubt that shape optimization has given improved structural results, likely

attributable to smearing of plastic strain and deflection of the blast wave.

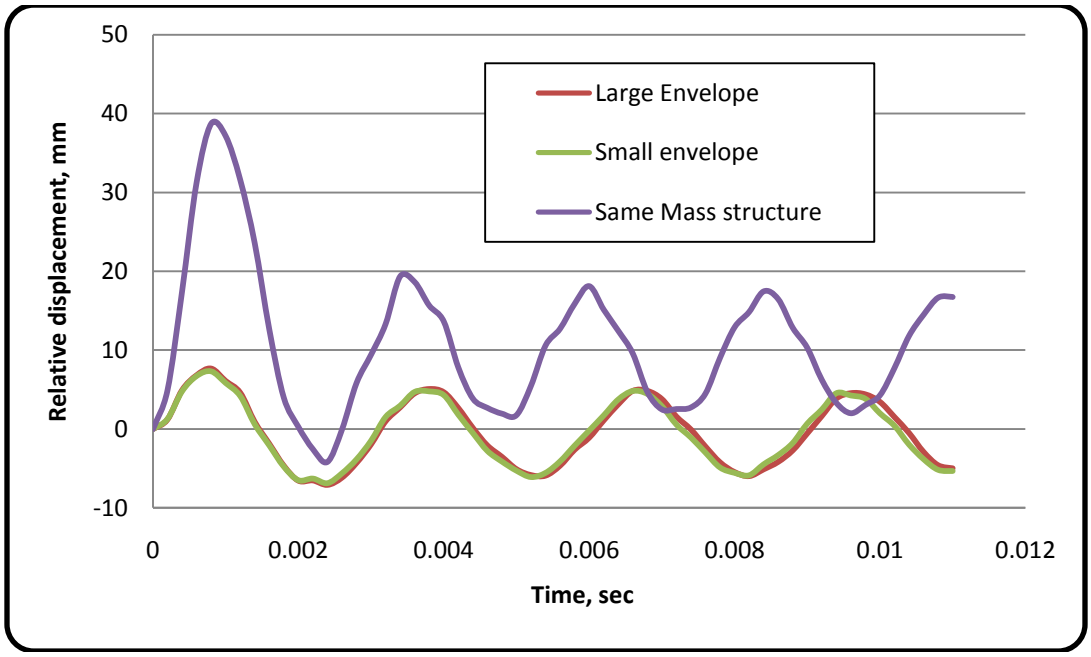

Figure **4-8**: Comparison of Relative displacement with equal mass plate

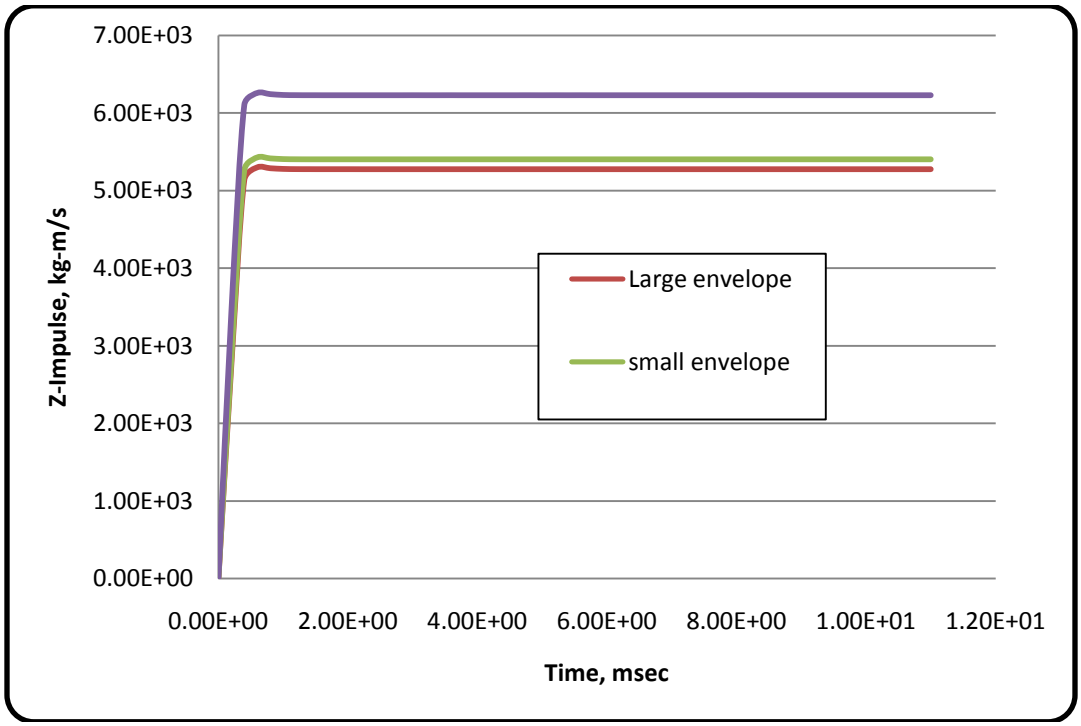

Figure **4-9**: Comparison of saturated impulse with equal mass plate

Table **4-2**: Numerical results for 3dv case in comparison with equal mass plate.

Property	**Same mass structure***	**Large Envelope (-0.3,1.0)m**		**Small Envelope (-0.1,0.14)m**	
		Optimized Design	*Change*	*Optimized Design*	*Change*
Objective Function (*m*)	1.32E-02	4.34E-03	67.12%	4.49E-03	65.98%
Max Relative Displacement (*m*)	3.88E-02	7.32E-03	81.15%	7.68E-03	80.23%
Max plastic strain	0.08739	0.01932	77.89%	0.02028	76.79%
Total Mass (*kg*)	1894.5	1894.57	0.00%	1894.54	0.00%
Saturated impulse, (*kg-m/s)*	6229.3	5298.62	14.94%	5425.82	12.90%

Optimized shape with 9 Design Variables

Optimization results with 9 basis shapes is given in this section. Refer to Figure **3.7-3.9** for figures of 9 DVs. The best shape obtained by the optimizer (repeated trials were needed) is shown in Fig. **4-10** , which is again seen to be a near-symmetric double bulge. Result is given in Table **4-3** for the small envelope case. Results obtained show only marginal improvement over 3DV Case. The 9DV Case is a very challenging task for any optimizer. It may still be that a hitherto undetermined unsymmetric sinusoidal shape lies within the design space that has a lower objective. However, we have confidence in the result obtained below owing to the use of various starting random number seeds. Further, optimal shapes that are ‘wavy’, even if discovered in the future, are hard to manufacture and may be highly sensitive to charge location.

Figure **4-10**: Optimized shape for 9 DV case .

Table **4-3**: Results for baseline design and optimized designs for 9DV case .

Property	*Baseline Design*	*Small Envelope (-0.1,0.14)m*	
		Optimized Design	*Change*
Objective Function (*m*)	20.51E-03	4.28E-03	79.1%
Max Relative Displacement (*m*)	58.43E-03	7.79E-03	86.7%
Max plastic strain	0.1277	0.02233	82.5%
Total Mass (*kg*)	1872.2	1895.3	(1.2)%
Saturated impulse, (*kg-m/s)*	6254.3	5313.58	15.0%

Optimized Shape with 11 Design variables

After experimenting with 3 and 9 DV which are sinusoidal in nature, velocity fields based on local point loads are used for shape optimization. The optimized shape using 11 design variables based on local point load deflection is given in this section. Refer to Figures **3-12 - 3.15** for detailed figures of 11 DVs. The best shape obtained by the optimizer (repeated trials were needed) is shown in Figure **4-11**, which is again seen to be a near-symmetric double bulge. Result is given in Table **4-4** for the small envelope case.

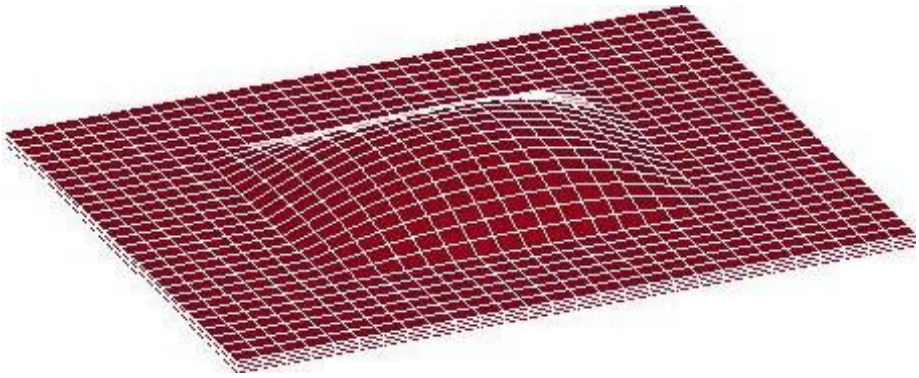

Figure **4-11**: Optimized shape with 11DV – top side .

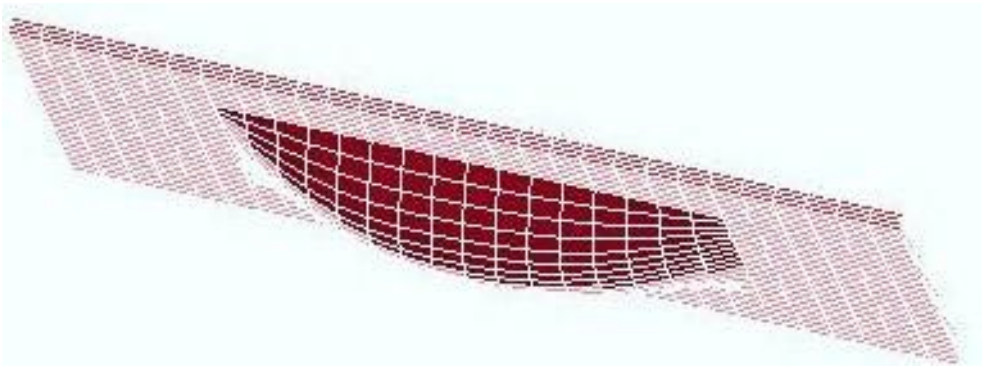

Figure **4-12**: Optimized shape with 11DV– bottom side .

Table **4-4**: Results for baseline design and optimized designs for 11DV case .

Property	*Baseline Design*	*Small Envelope (-0.1,0.14)m*	
		Optimized Design	*Change*
Objective Function (*m*)	20.51E-03	4.28E-03	78.2%
Max Relative Displacement (*m*)	58.43E-03	7.40E-03	87.3%
Max plastic strain	0.1277	0.02307	81.9%
Total Mass (*kg*)	1872.2	1894.54	(1.19)%
Saturated impulse, (*kg-m/s)*	6254.3	5384.84	14%

Comparison of 3dv, 9dv and 11dv results

Table **4-5** gives the comparison of 3dv,9dv and 11dv results.

Table **4-5**: Comparison of 3dv,9dv and 11dv results .

Property	*Baseline*	*3dv*	*9dv*	*11dv*
Objective Function (m)	2.05E-02	4.49E-03	**4.28E-03**	4.48E-03
Max Relative Displacement (m)	5.84E-02	7.68E-03	7.79E-03	**7.40E-03**
Max plastic strain	0.1277	**0.02028**	0.02233	0.023072
Total Mass (kg)	1872.2	**1894.54**	1895.3	**1894.54**
Saturated impulse, (kg-m/s)	6254.3	5425.82	**5313.58**	5384.88

As seen from the Table **4-5**. There is little change in the results owing to the fact that the optimized shape is double bulge in all the cases. The double bulge shape for the 9-DV problem is slightly unsymmetrical, owing to the choice of basis shapes used. Bulge in the side facing the charge deflects the pressure wave which reduces the impulse experienced by the plate. It is interesting to see that the optimized shape is similar in all the cases of design variables. Higher design variables were run with different random seeds and yet double bulge was the optimized shape. It should be noted that in all the above cases, optimized shape remained same when run even for higher number of generations.

Results of Parallel Computation

Distribution of Computing times

We first study the distribution of computing times for FE analysis of each member of the design population. As noted in the last chapter, the unequal FEA times creates challenges to obtain good speedup. Results for the 3-DV (three design variable) problem are used here. Figure **4.13** shows how the iteration time for every processor varies with optimization iterations with n_p = 4. The length of each bar represents the time taken by each iteration. Computations for the next generation can commence only after all the processors finish function evaluations for the current generation. This slows down the progress as some processors wait for other processors to finish their computations. The iteration time for any processor is the sum of time taken by all the function evaluations allocated to that processor. Potentially, each worker can get a different number of function evaluations as the master processor tries to balance the computational load. Next, Figure **4-14** shows the distribution of the FE (Ls-Dyna function evaluation) times. Bin represents the function evaluation time while frequency gives the number of evaluations in a given bin. Majority of the function evaluations required about 150 s. However, a significant number of members in the generation turn out to be invalid/distorted designs requiring almost zero compute time. Finally, in Figure **4-15,** the normalized computation time of iteration is plotted against the iteration number. Departure from equal-time analyses, indicated by a horizontal line, is again evident.

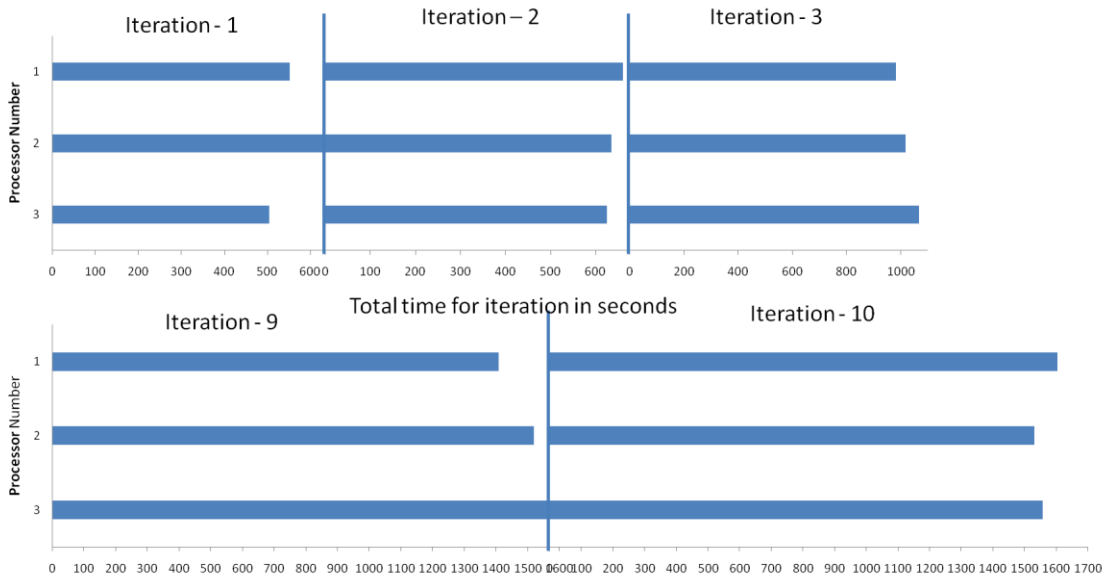

Figure **4-13**: Bar chart to show iteration time of every processor (for $n_p = 4$) for 3-DV case .

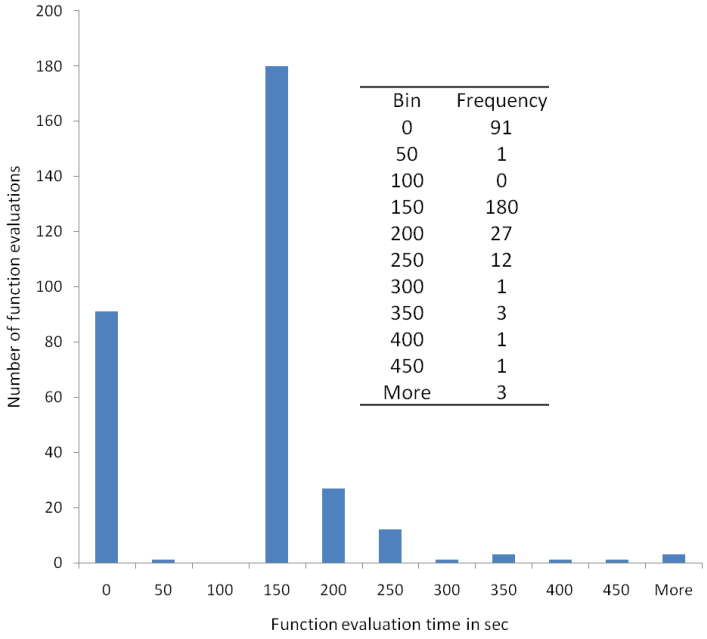

Figure **4-14**: Histogram of function evaluation times for 3-DV for LB approach.

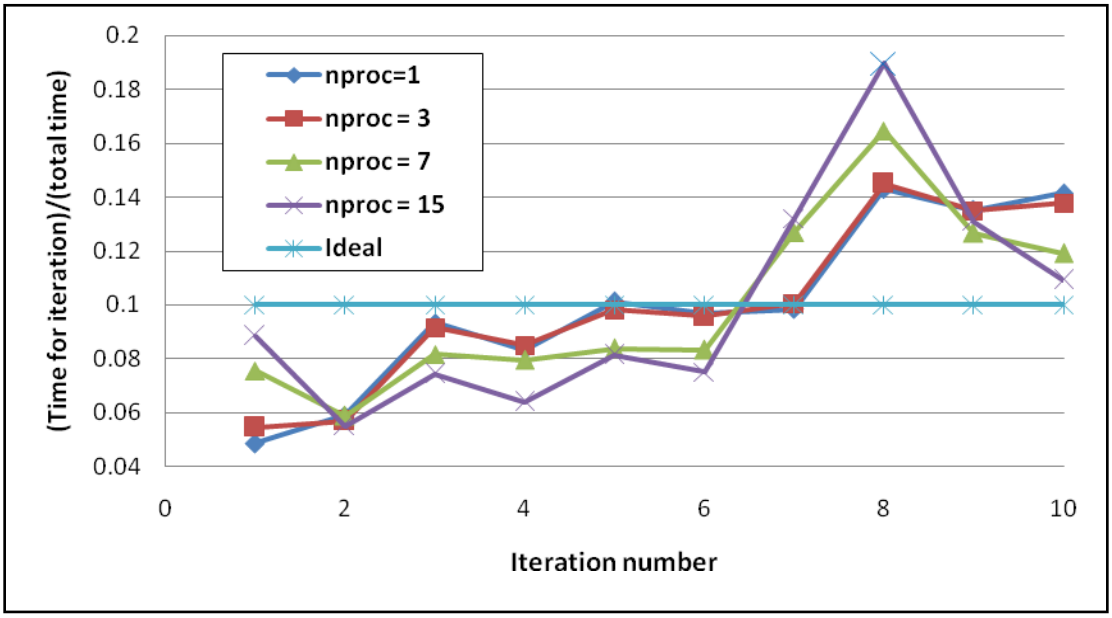

Figure **4-15**: Plot showing the variation of computation time of every iteration for different number of processors (SATR approach).

Speedup

The SATR and LB schemes were used to solve the design optimization problem. Speedups obtained by the two schemes discussed in Section 3 are shown for the 3-DV and 9-DV cases, respectively, in Figure **4-16,4-17** and corresponding Tables **4-6, 4-7.** Speedup is defined in the usual way as $S = T(n_{ref}) / T(n_p)$, where n_{ref} = number of processors used in the benchmarking and n_p = number of processors. In the 3-DV case in Table **4-6** (Figure **4-16**), $n_{ref} = 1$, while in the 9-DV case in Table **4-7** (Figure **4-17**), $n_{ref} = 4$ (the 9-DV problem cannot be solved with less processors in reasonable time). Note that the speedup with SATR, when n_p is very low, is better than LB since the latter uses one of the processors as a master which does not share in FEA. As is to be expected, LB overtakes SATR with increasing n_p. Compared to ideal speedup, both fall short, owing to the continuing presence of assorted computing times per FEA.

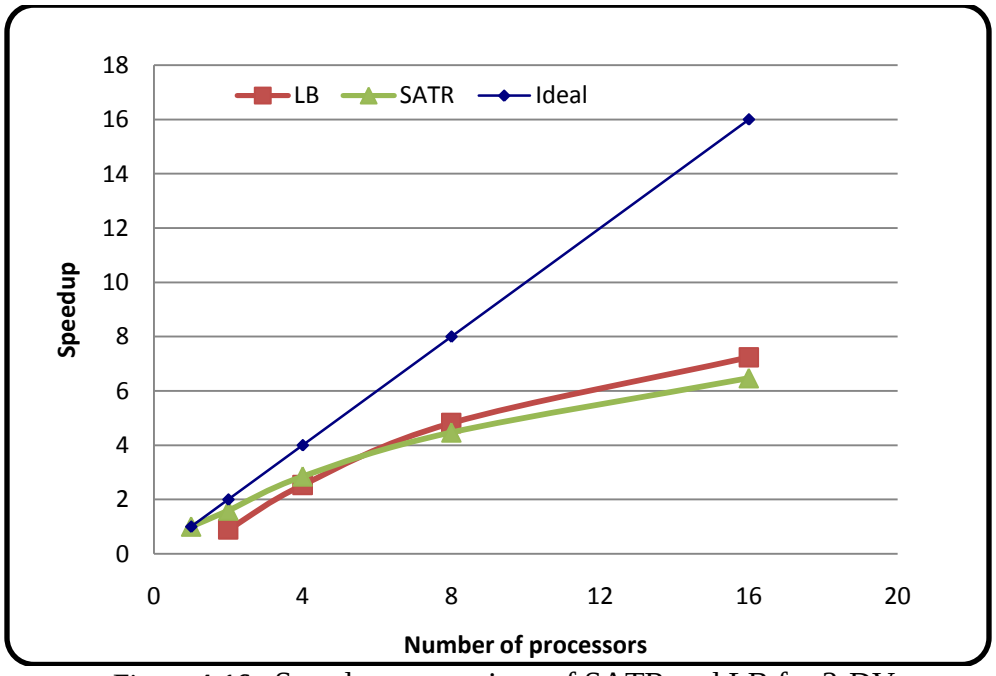

Figure **4-16**: Speedup comparison of SATR and LB for 3-DV case

Table **4-6**: Parallel computation for 3 DV, N_{pop} = 32, N_{gen} = 10

NP	*Time in mins (SATR)*	*Speedup (SATR)*	*Time in mins (LB)*	*Speedup (LB)*
1	487.6	1.0		
2	306.5	1.6	544.1	0.9
4	171.3	2.8	192.3	2.5
8	109.1	4.5	101.2	4.8
16	75.4	6.5	67.4	7.2

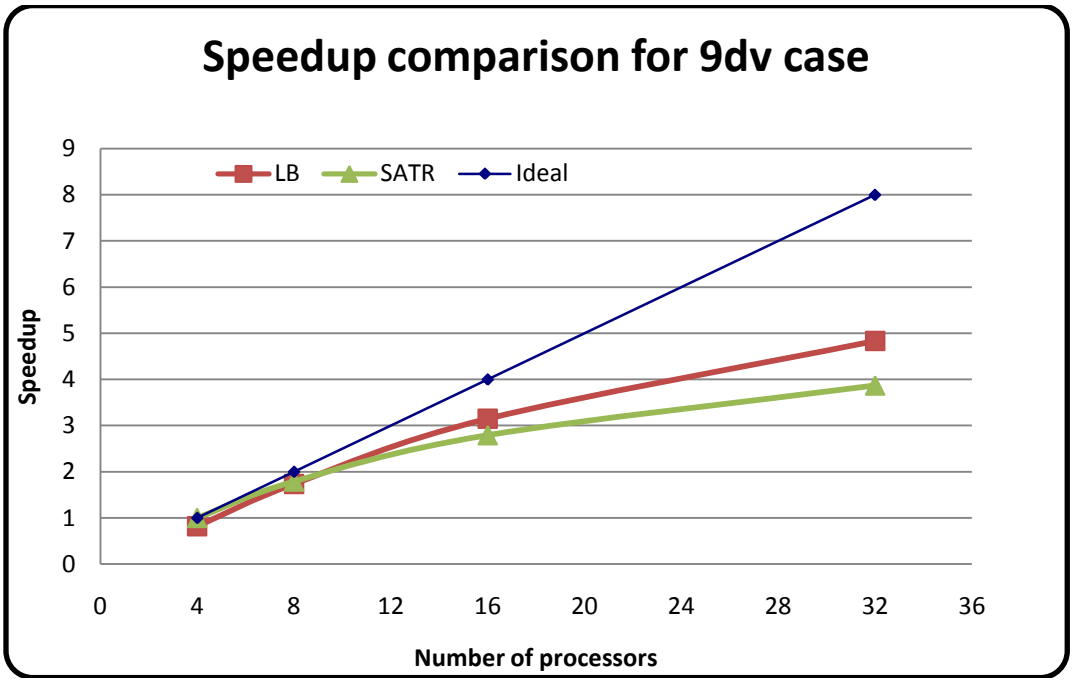

Figure **4-17**: Speedup comparison of SATR, LB for 9-DV case

Table **4-7**: Parallel computation for 3 DV, N_{pop} = 100, N_{gen} = 30

NP	*Time in mins (SATR)*	*Speedup (SATR)*	*Time in mins (LB)*	*Speedup (LB)*
4	1402.5	1.0	1701.1	0.8
8	784.2	1.8	806.0	1.7
16	502.2	2.8	445.0	3.2
32	362.4	3.9	290.0	4.8

Chapter 5

Conclusions

Shape optimization of a panel for blast load mitigation is carried out. This research is timely as very little work has been done in this area. The work has direct application to design of protective structures in military and civilian applications. After studying some heuristic designs like waffle shapes, formal shape optimization methodology was developed. Initial runs proved the nonlinear nature of the problem requiring the use stochastic optimizers like evolutionary algorithms. Ls-Dyna is coupled to a stochastic differential evolution optimizer using a modular Fortran code that has been developed. Thus, accurate responses are used at all times during optimization. Due to the stochastic nature, DE requires much more function evaluation compared to gradient based optimizers. Hence high performance clusters were used here. However, parallelization of the DE optimizer is relatively easy and makes the optimization approach viable. Population evaluation in every generation is distributed across the cluster with load balancing for an efficient utilization of the computational resources. The finite element model has been developed to reflect experimental test conditions and observed structural response.

Three and nine sinusoidal basis shapes and 11 local point load velocity fields are chosen. The basis shape is generated by a modular approach and is designed in a way that, generation of higher number of basis shape is simple. The optimum shape, a combination of these basis shapes, turns out to be a double-bulge in all the three cases, with unequal bulges on both the sides of the plate. Further, the shape and results are robust with respect to small changes in charge density or standoff distance. The bulge in the charge side of the plate diverts the spherical charge thereby reducing the effect of direct blast waves. The exact role of upper bulge is not understood so far. The panel's rms displacement which is the objective function, relative to the fixture, decreased by

78% compared to the baseline or flat plate design. Saturated z-impulse decreased by 14%. Maximum plastic strain decreased significantly as well and was well within the limit. In the process of 'fair' comparison, an optimized plate is compared with a 'same mass' flat panel. The optimized plate still shows a 67% improvement in the objective function and significant reduction in the plastic strain as before. The task of finding a global optimum in such highly nonlinear, nonconvex and computationally expensive functions is challenging. This work lays down a methodology of shape optimization against blast loading.

Future Work

After successfully implementing the shape optimization methodology for blast mitigation, ideas of future work are given below:

- This work being purely simulation based, correlation with the actual test is essential. If carried out, an **experimental blast** against double bulged plate will throw light into many aspects. However, this is out of the scope of university research and can be considered only by defense labs.
- The result of higher three, nine and 11 design variables has resulted in double bulge. But it is unclear whether this is the global minimum. Reaching at global minimum requires improved optimizer with **better constrained handling**, which also improves the speedup in the clusters. In this work, constraints are handled as penalty and are found not to be so efficient in the higher design variable domain. Improvement to the optimizer will be useful in finding global minima.
- This work assumes the charge to be at or in the vicinity of the geometric centre of the plate. To be more realistic in the design, it would be interesting to optimize for **multiple positions of the charge**. Charge can be placed at random positions

and ensemble average of RMS displacement value may be considered as objective function.

- Coupling **topology** and shape optimization models might even result in more interesting answers. Comparative performance of solid metal honeycomb, composite and other panel concepts in the same unified framework needs to be studied.

Bibliography

1. Belegundu, A.D, Argod,V., Rajan,S.D., Krishnan,K., *Shape optimization of panels subjected to blast loading modeled with LS-DYNA*, 49th AIAA/ASME/ASCE/-AHS/ASC Structures, Structural Dynamics, and Materials. AIAA 2008-2285 (2008)
2. Belegundu, A.D, Argod,V., Rajan,S.D., Jain,R., *Shape optimization of solid isotropic plates to mitigate the effects of air blast loading*, submitted to Intl J of Impact Engineering (2008)
3. Argod,V., Belegundu,A.D., Aziz,A., Agrawala,V., Jain,R., and Rajan,S.D., *MPI-enabled Shape Optimization of Panels Subjected to Air Blast Loading*, accepted at Intl J of Simulation of multidisciplinary Design Optimization (2008)
4. Dharaneepathy, M.V. and Sudhesh, K.G. *Optimal Stiffening of square plates subjected to air-blast loading.* Compuers & Structures. Vol 36 No 5, 891-899 (1990)
5. Yen, C.F., Skaggs, R., Cheeseman, B.A., *Modeling of shock mitigation sandwich structures for blast protection*. The 3rd First International Conference on structural stability and dynamics, Kissimmee, Florida, June 19-22 (2005)
6. BAE Systems, Land Systems South Africa (LSSA), RG31 http:// www. baesystems.com/ WorldwideLocations/ SouthAfrica
7. Storn, R . and Price,K., *Differential Evolution a simple and efficient adaptive scheme for global optimization over continuous spaces*. Journal of Global optimization 11, 341-359 (1997)
8. Storn, R, *System design by constraint adaptation and differential evolution*. IEEE Transactions on Evolutionary Computation ,22-34 (1997)

9. Hou, Hai-Liang, *Study on failure mode of stiffened plate and optimized design of structure subjected to blast load*, Explosion and Shock Waves [in Chinese], vol. 27, 26-33 (2007)
10. Xue, Z. and Hutchinson, J.W., *A Comparative study of impulse-resistant metal sandwich plates*, International Journal of Impact Engineering, 30, 1283–1305 (2004)
11. Fleck, N.A. and Deshpande, V.S., *The Resistance of Clamped Sandwich Beams to Shock Loading*, vol. 71, J. of Applied Mechanics, Transactions of the ASME, 386-401, (2004)
12. Liang, C.C.,Yang, M.F., Wu, P.W ,*Optimum design of metallic corrugated core sandwich panels subjected to blast loads*. Ocean Engineering, 28, 825–861 (2001)
13. Main, J.A. and Gazonas, G.A, *Uniaxial crushing of sandwich plates under air blast: Influence of mass distribution*, International Journal of Solids and Structures. 45, 2297–2321 (2008)
14. Icardi, U. and Ferrero, L., *Impact and blast pulse: Improving energy absorption of fibre-reinforced composites through optimized tailoring*. Proceedings of ESDA 2006, 8th Biennial ASME conference on Engineering Systems Design and Analysis, Torino, Italy, July 4-7 (2006)
15. Sriram, R., Vaidya, U.K. and Kim, J., *Blast Impact response of aluminum foam sandwich composites*. Journal of Material Science 41,4023-4039 (2006)
16. Thiagarajan, G., Hsia, J.K. and Walters, W. *Blast simulation of armor plate behavior using the virtual internal bond model*, 16th ASCE Engineering Mechanics Conference. University of Washigton, Seattle.July 16-18 (2003)
17. Duan,Y., Keefe, M., Bogetti, T.A. and Cheeseman, B.A., *Modeling friction effect on the ballistic impact behavior of a single-ply high-strength fabric*. International Journal of Impact Engineering.

18. SITIS Archives ,*Development of Reactive Reflector Technology for Vehicle and Crew Protection from Blast of Landmine and Improvised Explosive Device (IED)*, A07-177 (Army)
19. Cheeseman, B. A., *Protecting the Future Force: Advanced materials and analysis enable robust composite armor,* AMPTIAC Quarterly, vol. 8, no. 4, pp. 37-43 (2004)
20. Bulmash, G.and Kingery, C.N *Computational Determination of blast wave pressure-time histories*, Computers in Engineering., Proceedings of the International Computers in Engineering Conference, Volume 2 p 343-347 (1986)
21. LS-DYNA Keyword Manual version 971
22. Johnson, G.R. and Cook,W.H., *A constitutive model and data for metals subjected to large strain, high strain rates and high temperatures*, Proc. 7th Int. Symposium. on Ballistics, pp. 541-547(1983)
23. Clausen,B.T., Arild, H., Hopperstad,O. and Langseth,M, *Perforation of AA5083-H116 aluminium plates with conical-nose steel projectiles—experimental study*, International Journal of Impact Engineering.Vol 30,p367-384(2004)
24. Belegundu, A.D, and Chandrupatla, T. R., *Optimization Concepts and Applications in Engineering*, Prentice-Hall, New Jersey(1999)
25. Belegundu, A.D.and Rajan,S.D., *A Shape Optimization Approach Based on Natural Design Variables and Shape Functions,* Journal. Computer Methods in Applied Mechanics and Engineering, 66:87-106 (1998)
26. Timoshenko, S. P. and Woinowsky-Krieger, S. *Theory of plates and shells,* Engineering Societies Monographs, New York: McGraw-Hill, 2nd edition (1959)
27. Rajan, S.D. and Nguyen, D. T., *Design Optimization of discrete structural systems using MPI-enabled genetic algorithm*, Structural Multidisciplinary Optimization, 27,1-9(2004)

28. Tasoulis, D.K., Pavlidis, N.G.;Plagianakos, V.P. and Vrahatis, M.N. *Parallel Differential Evolution*

29. Plagianakos, V.P, and Vrahatis, M.N., *Parallel evolutionary training algorithm for 'hardware-friendly' neural networks*. Natural Computing ,307-322 (2002)

30. Michalewicz, Z.and Fogel, D.B., *How to solve it: Modern heuristics*. Springer (2000)

31. Biedron, R.T.; Mehrotra, P., Nelson, M.L., Preston, F.S., Rehder, J.J.;Rogers, J.L., Rudy, D.H., Sobieski,L. and Storaasli,O.O, *Compute as fast as the engineers can think!* NASA/TM-1999-209715, September (1999)

32. Kwedlo,W., *Parallelizing Evolutionary algorithms for clustering Data*. Springer-Verlag Berlin Heidelberg. PPAM 2005, LNCS 3911, pp.430-438 (2006)

33. Singh, L. and Kumar, S., *Parallel Evolutionary Asymmetric Subsethood product Fuzzy Neural Inference System with applications*. IEEE International Conference on Fuzzy Systems (2006)