

University of Bahrain
College of Engineering
Department of Civil and Architectural Engineering

OPTIMUM DESIGN OF ROOF TRUSSES BY GENETIC ALGORITHMS

BY

SAMAR NIZAR HUSSAIN WELDALI

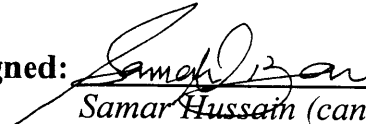
Thesis Submitted in Partial Fulfillment of the Requirement for the Degree of

MASTER OF SCIENCE IN CIVIL ENGINEERING

December 1997

DECLARATION

I declare that this work is the result of my own investigations and that it has neither been accepted nor being submitted in substance for any other degree.

Signed: 
Samar Hussain (candidate)

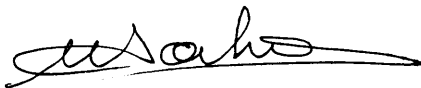
Date: 17 Dec. 1997

The oral examining committee considers the Thesis titled:

OPTIMUM DESIGN OF ROOF TRUSSES BY GENETIC ALGORITHMS

Submitted by Samar Hussain to the Department of Civil and Architectural Engineering satisfies the requirement for the Master of Science degree in Civil Engineering.

Oral Examining Committee:



Prof. M. P. Saka
(Supervisor)

Department of Civil and Architectural
Engineering, University of Bahrain



Prof. E. Hinton
(External Examiner)

University of Wales, Swansea,
United Kingdom.



Dr. Esmat Kameshki
(Internal Examiner)

Department of Civil and Architectural
Engineering, University of Bahrain

In the name of Allah, the Compassionate, the Merciful

He is Allah, the one: none but He is worthy of worship: to Him is all praise due, both in this world and in the Hereafter; Sovereignty is His, and to Him shall all of you be returned.

Surah AL-Qasas, from Ayah 69-70, The Holy Quran

To my parents and my dearest brother Waddah

ABSTRACT

Structural design is a process by which the optimum solution is sought. In any design certain criteria must be established to evaluate whether or not the optimum has been achieved. The design of a steel roof truss in particular involves in making decisions about its topology, geometry, steel sections to be used for its members and the spacing between trusses. All these are required to be selected in such a way that the final truss should be capable of resisting to the loads that may act during its life time and that the truss should be economical to produce. This decision making problem when formulated as programming problem turns out to be nonlinear programming problem in which some of the design variables are discrete in nature while others are continuous.

There are vast amount of algorithms developed in the past to obtain the solution of structural optimization problems where the design variables are to be selected from a continuous set. However, the optimum solutions obtained with these methods are not of interest to the practicing engineer. This is due to the fact that the design variables are usually discrete in nature. This necessitates to consider this feature of the design parameters in the solution techniques to be developed.

Certainly there exists number of algorithms as well which can be used in the solution of structural optimization problems where the design variables are to be selected from a discrete set. However, these are cumbersome and require tedious computer programming.

The genetic algorithm is a comparatively recent addition to the methods of optimization which is quite suitable for use in structural design due to its ability to deal with discrete design variables. These algorithms are search procedures based on the mechanics of natural selection and natural genetics. They apply the natural law of the survival of the fittest to structural optimization. They initiate a search with an initial population which consists of set of artificial creatures representing each possible design for the structure. The algorithm moves from one generation to another during which some individuals are allowed to die-off while some others are allowed to multiply. This is achieved by generating new individuals from the bits and pieces of the fittest of the old, occasionally adding new parts for good measures. This is performed through the three random genetic operators of reproduction, crossover and mutation. Although genetic algorithms are randomized, they are not simple random walk. They efficiently exploit historical information to speculate on new search points with expected improved performance. Their simplicity in approach and directness in discrete variables make them attractive in comparison with more mathematically complex methods.

In this study genetic algorithm based optimum design method is developed for roof trusses subjected to multiple loading cases. The algorithm obtains the roof truss which has the minimum weight by selecting appropriate steel sections from available section tables while satisfying the design limitations given in BS5950. The design constraints in the design problem are implemented directly from the code.

The algorithm has three versions. In the first, the optimum angle section designations are found for the truss members while the geometry and spacing between the trusses are kept fixed. In the second the height of apex is treated as design variable in addition to member sizes. The third one considers the spacing between the trusses also as an additional design variable. In all these versions, the genetic algorithm has successfully obtained the optimum solution.

The design algorithm developed is applied to find the optimum solution of four different design problems in each seven different topologies is considered for the roof truss. In the first, the geometry and spacing are kept fixed and optimum equal leg angle designations are obtained for each roof truss topology. In the second the spacing is also made variable while the geometry is kept fixed. It is found that saving was possible in the overall weight of the roof if the spacing is made a design variable. In the third, the height of apex is treated as variable allowing the geometry to change while the spacing is kept fixed. In the fourth, the height of apex and the spacing are taken as variables in addition to member designations. It is shown that the latter yields to the lightest structure for the roof. It is shown that up to 15% saving was possible in the lightest topology when height and spacing are treated as variable as compared to fixed geometry and spacing. The genetic algorithm generally obtained the optimum solution efficiently for the design problems where the height and spacing are fixed. However when those were taken as design variables, it becomes necessary to rerun the program few times to detect the optimum solution. For the geometrical and topological design problems, the genetic algorithm has the tendency to settle down with the local optimum.

ACKNOWLEDGMENTS

I would like to express my deep and sincere words of thanks and appreciation to my advisor Professor M.P. Saka for his continuous support.

Words of thanks goes to the University of Bahrain which has provided me with all the required facilities that enabled me to complete the work on my MSc. thesis.

Samar N. Hussain

TABLE OF CONTENTS

ABSTRACT	i
ACKNOWLEDGMENTS	iv
LIST OF SYMBOLS	xii
LIST OF TABLES	xx
LIST OF FIGURES	xxi

1 INTRODUCTION	1
1.1 DESIGN OF ROOF TRUSSES	1
1.2 STRUCTURAL OPTIMIZATION	2
1.2.1 Design Variables	2
1.2.2 Constraints	4
1.2.3 Objective Functions	4
1.3 STRUCTURAL OPTIMIZATION TECHNIQUES	5
1.3.1 Continuous Variable Optimization Methods	6
1.3.1.1 Unconstrained Optimization	6
1.3.1.2 Constrained Optimization Methods	7
1.3.1.2.1 Direct Methods	8
1.3.1.2.2 Indirect Methods	8
1.3.2 Discrete Variable Optimization Methods	9
1.3.2.1 Branch and Bound Methods	10

1.3.2.2 Approximation Methods	10
1.3.2.3 Genetic Algorithms	10
1.4 LITERATURE SURVEY	12
1.5 SCOPE OF THE WORK	16
2 GENETIC ALGORITHMS	19
2.1 INTRODUCTION	19
2.2 OPTIMIZATION AND GENETIC ALGORITHMS	20
2.3 WORKINGS OF A SIMPLE GENETIC ALGORITHM	22
2.3.1 Generation of Initial Population	22
2.3.2 Operators of Simple Genetic Algorithm	24
2.3.2.1 Reproduction Operator	25
2.3.2.1.1 Fitness Function	27
2.3.2.1.2 Fitness Scaling	27
2.3.2.2 Crossover Operator	28
2.3.2.3 The Mutation Operator	29
2.4 MATHEMATICAL ANALYSIS OF GAS: SCHEMA THEOREM	30
2.5 MODIFICATIONS TO THE SIMPLE GENETIC ALGORITHM	34
3 OPTIMUM DESIGN OF ROOF TRUSSES VIA GENETIC ALGORITHMS	39
3.1 INTRODUCTION	39
3.2 OPTIMUM DESIGN OF ROOF TRUSSES	40
3.2.1 The Mathematical Model	40

3.2.2 Objective Function	41
3.2.3 Design Variables	42
3.2.4 Design Constraints	42
3.2.4.1 Displacement Constraints	43
3.2.4.2 Member Capacity Constraints	43
3.3 GENETIC ALGORITHM IMPLEMENTATION	45
3.3.1 Population Generation	46
3.3.2 Reproduction Operator	49
3.3.2.1 Fitness Function	50
3.3.2.2 Fitness Function Scaling	50
3.3.3 Crossover Operator	52
3.3.4 Mutation Operator	53
3.4 OPTIMUM DESIGN ALGORITHM	53
3.5 ANALYSIS AND DESIGN OF ROOF TRUSSES	54
3.5.1 Load Cases	55
3.5.2 Truss Analysis	56
3.5.3 Truss Members Design	58
3.5.3.1 Design of a Compression Member	58
3.5.3.2 Design of a Tension Member	60
3.6 SAMPLE EXAMPLE	61
4 GENETIC ALGORITHM IMPLENTATION IN PROGRAMMING	67
4.1 INTRODUCTION	67
4.2 OPTIMUM DESIGN OF ROOF TRUSS WITH FIXED GEOMETRY	68
4.2.1 SUBROUTINE INPUT	70
4.2.2 SUBROUTINE IPOP	73

4.2.3 SUBROUTINE DECOD	74
4.2.4 SUBROUTINE EVALUATE	75
4.2.4.1 ANALYSIS Subroutine	76
4.2.4.1.1 ELEMENT Subroutine	77
4.2.4.1.2 RESOLVE Subroutine	77
4.2.4.1.3 SOLVER Subroutine	77
4.2.4.1.4 CLC Subroutine	77
4.2.4.2 CDESIGN Subroutine	77
4.2.4.2.1 CLASS Subroutine	78
4.2.4.2.2 CSTR Subroutine	78
4.2.4.2.2.1 XYLAMDA Subroutine	78
4.2.4.3 TDESIGN Subroutine	78
4.2.4.3.1 TSTR Subroutine	79
4.2.5 FITNESS Subroutine	79
4.2.6 COPY Subroutine	79
4.2.7 RANDMCS Subroutine	79
4.2.7.1 RANDUM Subroutine	80
4.3 OPTIMUM DESIGN OF ROOF TRUSS WITH VARIABLE GEOMETRY	80
4.4 OPTIMUM DESIGN OF ROOF TRUSS WITH VARIABLE TRUSS SPACING	81
5 DESIGN EXAMPLES	84
5.1 INTRODUCTION	84
5.2 DESIGN EXAMPLES WITH FIXED GEOMETRY AND SPACING	88

5.2.1 Truss Topology I	91
5.2.2 Truss Topology II	95
5.2.3 Truss Topology III	98
5.2.4 Truss Topology IV	101
5.2.5 Truss Topology V	104
5.2.6 Truss Topology VI	107
5.2.7 Truss Topology VII	110
5.2.8 Summary for Design Examples with Fixed Geometry and Fixed Spacing	113
5.3 DESIGN EXAMPLES WITH FIXED GEOMETRY AND VARIABLE TRUSS SPACING	115
5.3.1 Truss Topology I	115
5.3.2 Truss Topology II	117
5.3.3 Truss Topology III	120
5.3.4 Truss Topology IV	122
5.3.5 Truss Topology V	124
5.3.6 Truss Topology VI	126
5.3.7 Truss Topology VII	129
5.3.8 Summary for Design Examples with Fixed Geometry and Variable Truss Spacing	131
5.4 DESIGN EXAMPLES WITH VARIABLE GEOMETRY AND FIXED SPACING	133
5.4.1 Truss Topology I	133
5.4.2 Truss Topology II	135
5.4.3 Truss Topology III	138
5.4.4 Truss Topology IV	140
5.4.5 Truss Topology V	143
5.4.6 Truss Topology VI	145
5.4.7 Truss Topology VII	147

5.4.8 Summary for Design Examples with Variable Geometry and Fixed Spacing	149
5.5 DESIGN EXAMPLES WITH VARIABLE GEOMETRY AND VARIABLE TRUSS SPACING	151
5.5.1 Truss Topology I	151
5.5.2 Truss Topology II	153
5.5.3 Truss Topology III	155
5.5.4 Truss Topology IV	157
5.5.5 Truss Topology V	160
5.5.6 Truss Topology VI	162
5.5.7 Truss Topology VII	164
5.5.8 Summary for Design Examples with Variable Geometry and Variable Truss Spacing	166
5.6 SUMMARY OF ALL DESIGN EXAMPLES	168
6 SUMMARY AND CONCLUSIONS	170
6.1 SUMMARY	170
6.2 CONCLUSIONS	171
REFERENCES	174

LIST OF SYMBOLS

A_e	effective area of a cross-section
A_i	cross sectional area of the member
A_{net}	steel section net area
A_g	gross area of section
C_{max}	cost to fitness transformation coefficient
C_{mult}	number of expected copies of an individual
$\{D\}$	joint displacement vector
$f(x)$	Objective function
$F(x)$	fitness function
F'	scaled fitness function value
F_i	fitness function value for the i th individual
F_{avg}	average fitness function value
F_{cm}	compressive force in member m
F'_{max}	maximum fitness function value
F_{tk}	tensile force in member k
FF_i	fitness factor for the i th individual
$g_i(x)$	j th inequality constraint function
$h_i(x)$	i th equality constraint function
K	penalty function parameter
$[K]$	overall stiffness matrix
$[k_a]$	member stiffness matrix
L_i	Length of member i

n	population size
NJ	Number of joints
NRJ	Number of restricted joints displacements
num_i	expected number of copies for each individual
$\{P\}$	joint load vector
P_{cm}	compression resistance of member m
$P_{select\ i}$	probability of selecting the i th individual
P_{tk}	tension capacity of member k
x	design variables vector
ρ	density of steel
δ_j	vertical displacement of restricted joint j
δ_u	displacement upper bound
λ_j	slenderness of member j
λ_m	slenderness of compression member m
λ_{max}	maximum slenderness ratio
λ_{all}	allowable slenderness ratio
λ	string length
$\phi_i(x)$	modified objective function value for i th individual
$\phi(x)_{max}$	maximum modified objective function value
$\phi(x)_{min}$	minimum modified objective function value

LIST OF TABLES

Table 2.1	Mapping relationships between strings and discrete values	24
Table 2.2	Sample Problem Strings and Fitness Values	25
Table 3.1	Two sample problems, to illustrate the idea of fitness scaling	51
Table 3.2	Equal leg angle sections considered in the design of simple truss	61
Table 3.3	Detailed computations in the first generation	63
Table 3.4	Detailed computations in the first generation	65
Table 3.5	Final computations obtained for generation #50	66
Table 5.1	List of Load cases.	87
Table 5.2	List of sections designation with the required properties	90
Table 5.3	Optimum sections for topology I with fixed geometry and spacing.	92
Table 5.4	Member forces and capacities for the optimum design of the first topology.	93
Table 5.5	Optimum sections for topology II with fixed geometry and spacing.	96
Table 5.6	Member forces and capacities for the optimum design of the second topology.	97
Table 5.7	Optimum sections for topology III with fixed geometry and spacing.	99

Table 5.8	Member forces and capacities for the optimum design of the third topology.	100
Table 5.9	Optimum sections for topology IV with fixed geometry and spacing.	102
Table 5.10	Member forces and capacities for the optimum design of the fourth topology.	103
Table 5.11	Optimum sections for topology V with fixed geometry and spacing.	105
Table 5.12	Member forces and capacities for the optimum design of the fifth topology.	106
Table 5.13	Optimum sections for topology VI with fixed geometry and spacing.	108
Table 5.14	Member forces and capacities for the optimum design of the sixth topology.	109
Table 5.15	Optimum sections for topology VII with fixed geometry and spacing.	111
Table 5.16	Member forces and capacities for the optimum design of the seventh topology.	112
Table 5.17	Summary of the optimum designs of the seven truss topologies with fixed geometry and fixed spacing.	114
Table 5.18	Optimum sections for topology I with fixed geometry and variable spacing.	116
Table 5.19	Member forces and capacities for the optimum design of the first topology.	117
Table 5.20	Optimum sections for topology II with fixed geometry and variable spacing.	118
Table 5.21	Member forces and capacities for the optimum design of the second topology.	119
Table 5.22	Optimum sections for topology III with fixed geometry and variable spacing.	120

Table 5.23	Member forces and capacities for the optimum design of the third topology.	121
Table 5.24	Optimum sections for topology IV with fixed geometry and variable spacing.	122
Table 5.25	Member forces and capacities for the optimum design of the fourth topology.	123
Table 5.26	Optimum sections for topology V with fixed geometry and variable spacing.	124
Table 5.27	Member forces and capacities for the optimum design of the fifth topology.	125
Table 5.28	Optimum sections for topology VI with fixed geometry and variable spacing.	127
Table 5.29	Member forces and capacities for the optimum design of the sixth topology.	128
Table 5.30	Optimum sections for topology VII with fixed geometry and variable spacing.	129
Table 5.31	Member forces and capacities for the optimum design of the seventh topology.	130
Table 5.32	Summary of the optimum designs of the seven truss topologies with fixed geometry and variable spacing.	132
Table 5.33	Optimum sections and height for topology I with variable geometry and fixed spacing.	134
Table 5.34	Member forces and capacities for the optimum design of the first topology.	135
Table 5.35	Optimum sections and height for topology II with variable geometry and fixed spacing	136
Table 5.36	Member forces and capacities for the optimum design of the second topology.	137
Table 5.37	Optimum sections and height for topology III with variable geometry and fixed spacing.	139

Table 5.38	Member forces and capacities for the optimum design of the third topology.	139
Table 5.39	Optimum sections and height for topology IV with variable geometry and fixed spacing.	141
Table 5.40	Member forces and capacities for the optimum design of the fourth topology.	142
Table 5.41	Optimum sections for and height for topology V with variable geometry and fixed spacing.	143
Table 5.42	Member forces and capacities for the optimum design of the fifth topology.	144
Table 5.43	Optimum sections and height for topology VI with variable geometry and fixed spacing.	145
Table 5.44	Member forces and capacities for the optimum design of the sixth topology.	146
Table 5.45	Optimum sections and height for topology VII with variable geometry and fixed spacing.	147
Table 5.46	Member forces and capacities for the optimum design of the seventh topology.	148
Table 5.47	Summary of the optimum designs of the seven topologies with variable geometry and fixed truss spacing.	150
Table 5.48	Optimum sections, height and spacing for topology I with variable geometry and variable spacing.	152
Table 5.49	Member forces and capacities for the optimum design of the first topology.	152
Table 5.50	Optimum sections, height and spacing for topology II with variable geometry and variable spacing.	153
Table 5.51	Member forces and capacities for the optimum design of the second topology.	154
Table 5.52	Optimum sections, height and spacing for topology III with variable geometry and variable spacing.	156

Table 5.53	Member forces and capacities for the optimum design of the third topology.	156
Table 5.54	Optimum sections, height and spacing for topology IV with variable geometry and variable spacing.	158
Table 5.55	Member forces and capacities for the optimum design of the fourth topology.	159
Table 5.56	Optimum sections, height and spacing for topology V with variable geometry and variable spacing.	160
Table 5.57	Member forces and capacities for the optimum design of the fifth topology.	161
Table 5.58	Optimum sections, height and spacing for topology VI with variable geometry and variable spacing.	162
Table 5.59	Member forces and capacities for the optimum design of the sixth topology.	163
Table 5.60	Optimum sections, height and spacing for topology VII with variable geometry and variable spacing.	164
Table 5.61	Member forces and capacities for the optimum design of the seventh topology.	165
Table 5.62	Summary of the optimum designs of the seven truss topologies with variable geometry and variable truss spacing.	167
Table 5.63	Summary of the optimum designs for all examples	169

LIST OF FIGURES

Fig. 2.1	A sample representation of a binary string (An individual in a population)	23
Fig. 2.2	Simple reproduction allocates offspring strings using a roulette wheel with slots sized according to fitness.	26
Fig. 2.3	A simple crossover illustration showing the alignment of two strings and the partial exchange of information, using a cross site chosen randomly.	29
Fig. 3.1	Types of trusses	39
Fig. 3.2	A typical roof truss	47
Fig. 3.3	Individual String for first optimization problem	48
Fig. 3.4	Individual String for second optimization problem	48
Fig. 3.5	Individual String for third optimization problem	48
Fig. 3.6	Illustration of joint connection for both single and double angles.	59
Fig. 3.7	Example of a simple 3-bar truss.	61
Fig. 4.1	Common structure between the three algorithms.	67
Fig. 4.2	Flowchart of Program GA1.FOR	70
Fig. 4.3	An individual representation in a population	74
Fig. 4.4	Typical Truss showing the Geometry design variable YAPX.	80
Fig. 4.5	A Plan Layout showing the truss spacing design variable	82

Fig. 5.1	Typical Front Elevation and Plan view of the industrial Building	85
Fig. 5.2	The seven Truss topologies	86
Fig.5.3	Unfactored joint loads for different load cases.	89
Fig. 5.4	Member numbering for the first topology.	91
Fig. 5.5	Design history for the first topology with fixed geometry and spacing.	91
Fig. 5.6	Member numbering for the second topology.	95
Fig. 5.7	Design history for the second topology with fixed geometry and spacing.	95
Fig. 5.8	Member numbering for the third topology with fixed geometry and spacing.	98
Fig. 5.9	Design history for the third topology with fixed geometry and spacing.	98
Fig.5.10	Member numbering for the fourth topology.	101
Fig.5.11	Design history for the fourth topology with fixed geometry and spacing.	101
Fig. 5.12	Member numbering for the fifth topology.	104
Fig. 5.13	Design history for the fifth topology with fixed geometry and spacing.	104
Fig. 5.14	Member numbering for the sixth topology.	107
Fig. 5.15	Design history for the sixth topology with fixed geometry and spacing.	107
Fig. 5.16	Member numbering for the seventh topology.	110
Fig.5.17	Design history for the seventh topology with fixed geometry and spacing.	110
Fig. 5.18	Design history of the first topology with fixed geometry and variable spacing.	116
Fig.5.19	Design history for the second topology with fixed geometry and variable spacing.	118

Fig.5.20	Design history for the third topology with fixed geometry and variable spacing.	120
Fig. 5.21	Design history for the fourth topology with fixed geometry and variable spacing.	122
Fig.5.22	Design history for the fifth topology with fixed geometry and variable spacing.	124
Fig.5.23	Design history for the sixth topology with fixed geometry and variable spacing.	126
Fig.5.24	Design history for the seventh topology with fixed geometry and variable spacing.	129
Fig. 5.25	Design history for the first topology with variable geometry and fixed spacing.	133
Fig. 5.26	Design history of the second topology with variable geometry and fixed spacing.	136
Fig. 5.27	Design history of the third topology with variable geometry and fixed spacing.	138
Fig. 5.28	Design history of the fourth topology with variable geometry and fixed spacing.	140
Fig. 5.29	Design history of the fifth topology with variable geometry and fixed spacing.	143
Fig. 5.30	Design history of the sixth topology with variable geometry and fixed spacing.	145
Fig. 5.31	Design history for the seventh topology with variable geometry and fixed spacing.	147
Fig. 5.32	Design history of the first topology with variable geometry and variable truss Spacing.	151
Fig. 5.33	Design history of the second topology with variable geometry and variable truss Spacing.	153
Fig. 5.34	Design history of the third topology with variable geometry and variable truss Spacing.	155

Fig. 5.35	Design history for the fourth topology with variable geometry and variable truss Spacing.	157
Fig. 5.36	Design history for the fifth topology with variable geometry and variable truss Spacing.	160
Fig. 5.37	Design history for the sixth topology with variable geometry and variable truss Spacing.	162
Fig. 5.38	Design history for the seventh topology with variable geometry and variable truss Spacing.	164

Chapter 1

INTRODUCTION

1.1 DESIGN OF ROOF TRUSSES

Design of roof trusses is one of the earliest structural design problem that civil engineers had to deal with. Structural design in general is a process by which the dimensions of members of a structure is determined in such a way that the structure performs the expected function in a safe and economic way. The design process starts making decision about the shape of the structure. Generally, there are several framing shapes available in the practice. The designer has to select the most suitable one for his/her design. This is followed by another decision making process regarding initial sizes and shapes of the members that form the structure. The next step is to predict the loads that the structure is expected to withstand during its life time. The structure is then analyzed to determine its behavior under these loads. This behavior is required to be within the limitations described in design codes. If the initial dimensions do not produce a structure which is acceptable from strength or serviceability or economy point of view, the dimensions of members are altered and structural analysis is repeated to determine the new response. This iterative process is repeated until a satisfactory result is reached.

Structural design therefore can be defined as a decision process in which a certain objective is to be achieved while certain constraints are satisfied. As a result, it becomes possible to formulate structural design problems as a decision making problems and obtain their solutions by making use of the existing solution techniques of a separate discipline known as operation research. This discipline is of a relatively recent origin. It is basically developed after the second world war.

As its name implies, it is concerned with the planning, assessment and control of operating systems, such as industrial production or commerce. The methods developed for finding the solution of decision making problems in operation research are applied to structural design problems. This led to a new branch in structures known as structural optimization.

1.2 STRUCTURAL OPTIMIZATION

A structural optimization problem can generally be formulated as follows:

$$\begin{aligned} & \min f(\mathbf{x}) & (1.1) \\ \text{subject to} & \quad g_j(\mathbf{x}) \leq 0 & j = 1, 2, \dots, m \\ & \quad h_i(\mathbf{x}) = 0 & i = 1, 2, \dots, n \end{aligned}$$

where $\mathbf{x} = \{ x_1, x_2, \dots, x_n \}$ represents design variables, $f(\mathbf{x})$ is the objective function, $g_j(\mathbf{x})$ is the j th inequality constraint function, and m is the number of inequality constraints, $h_i(\mathbf{x})$ is the i th equality constraint function, and n is the number of equality constraints.

1.2.1 Design Variables

Design variables in structural optimization are those quantities which describe the structural system and they are decided by designer. These quantities are the overall dimensions of a structure, its geometry, its topology, types and locations of its supports, areas or moment of inertia of its members. In some cases material properties such as modulus of elasticity is also added to the design variables. It is possible to include all these parameters in the formulation of the design problem. Consequently such more leads to a complex mathematical problem solution of which is cumbersome. On the other hand, some of these parameters may be fixed due to architectural reasons and can not be included among the design variables. Those quantities that are fixed during the automated design are called “preassigned parameters” and they are not varied by the optimization algorithm. Those quantities that are not preassigned and their values are changed during the optimum design cycle are called “design variables”.

From the mathematical point of view, it is important to distinguish between continuous and discrete design variables [1]. In cases of discrete variables which are uniformly distributed over a given interval, use of continuous variable representation is often satisfactory, which is followed by selection of the nearest available discrete value. However, it should be recognized that situations arise when it will be essential to employ discrete variables. One such example is the design of steel structures where the cross sections of members are required to be selected from a set of standard steel sections.

The design variables that are varied by the optimization procedure may represent the following properties of the structure [1]:

a- Topological Design Variables:

The topology of a structure is defined as the total number of joints and members and the pattern they are connected. Hence, the design variables which represent any of the above parameters are called topological design variables.

Topology of the structure can be optimum automatically in certain cases when members are allowed to reach zero size. This permits the elimination of some uneconomical members during the optimization process.

b- Geometrical or configuration Design Variables:

The geometry of a truss or a frame is a function of the locations of joints at which members intersect. Any change in the location of a joint in a structure alters its geometry. Hence, the coordinates of joints in a structure can be taken as geometrical design variables. In some cases instead of joint coordinates, a span of a beam or height of a truss are treated as design variables. In general, the geometry of the structure can be represented by either continuous or discrete variables.

c- Cross-Sectional Design Variables:

Cross-sectional dimensions or member sizes are the simplest design variables. The cross sectional area of a truss member, the moment of inertia of a flexural member,

or the thickness of a plate are some examples of this class of design variables. In practical design, cross sectional variables may be restricted to some discrete values. Such are the rolled steel sections, which are produced in distinct sizes with unevenly spaced cross sectional properties. In such cases the design variables are permitted to take only one of a discrete set of available values.

1.2.2 Constraints

As mentioned earlier, the design process necessitates the behavior of a structure which is to be designed to be within the limitations described in design codes. It should have sufficient strength to withstand the loads and it should not deflect excessively. Hence, any design which follows recent design codes should satisfy the ultimate limit states and serviceability requirements. The values of the design variables are to be obtained in such a way that both constraints are required to be satisfied. These constraints can be expressed either as an inequality form or as an equality form. A general structural design problem therefore may include inequality constraints as written in the following

$$g(x_1, x_2, \dots, x_n) \leq 0 \quad (1.2)$$

Furthermore it may contain some equality constraint which has the form of

$$h(x_1, x_2, \dots, x_n) = 0 \quad (1.3)$$

The set of design variables which satisfy all the design constraints define a “feasible set”. If some of the constraints are not satisfied, then such set is termed “infeasible set” [2].

1.2.3 Objective Functions

There usually exists an infinite number of feasible designs. In order to find the best one, it is necessary to form a function of design variables which can be used for comparison of feasible design alternatives. The objective function is the

function whose least value is sought in an optimization procedure [1]. It is usually a nonlinear function of the variable x , and it may represent the weight, the cost of a structure, or any other criterion by which some possible designs are preferred to others [1].

Structural optimization techniques obtain the set of values for the design variables which satisfy all the constraints and attain a minimum value to objective function.

1.3 STRUCTURAL OPTIMIZATION TECHNIQUES

The available methods of structural optimization may conveniently be sub-divided into two distinctly different categories: *analytical* and *numerical* methods. The analytical methods generally employ the mathematical theory of calculus or variational methods [1]. These methods are most suitable for fundamental studies of single structural components, and they are usually not suitable to handle larger structural systems. On the other hand, numerical methods make use of techniques which are in the field of numerical mathematics called *mathematical programming* (MP). The MP methods have the advantage of wider applicability and base of resources. As a result, efficient and reliable algorithms are continually developed. In these methods, a near optimal design is automatically generated in an iterative manner. An initial guess is used as a starting point for a systematic search for better designs. The search is terminated when a certain criteria is satisfied, indicating that the current design is sufficiently close to the optimum [1]. Applying MP methods to structural design, a wide variety of problems can be considered, including :

- a) Complex structural systems subject to different failure modes in each of several load conditions.
- b) General design variables representing cross sectional dimensions, the geometry or the topology of the structure.
- c) Various constraints on the structural behavior and on the design variables.
- d) A general objective function representing cost or weight of the structure.

MP methods will find the optimum in problems where there exists a single *global optimum* [1]. In some structural applications, however, multiple *relative optima* may exist in addition to the global optimum which is sought. Local or relative minima may occur in MP problems due to the nature of both the objective function and the constraints. In such cases, the solution is liable to depend on the initial design from which the search procedure is started. This difficulty can be reduced by repeating the computations from different starting points and comparing the solutions until reasonable confidence is built up that the global optimum has been reached.

Mathematical programming methods can be divided into two classes : *continuous variables* and *discrete variables* optimization methods. Many methods are available to solve either of the two classes, each has its own characteristics, strengths and weaknesses. To choose the most effective method for any specific application, one should be familiar with the different methods and their adequacy with his or her problem.

1.3.1 Continuous Variable Optimization Methods

1.3.1.1 Unconstrained Optimization

Methods for unconstrained multi-variable optimization can be divided into search methods which use function evaluation only, and gradient methods which in addition to function evaluation require gradient information in the form of the Jacobian gradient vector [2,3].

Search methods for optimization attempt to reduce the value of the objective function $f(x)$ by the use of tests near to an estimate of the solution. The tests determine a direction of search in which the minimum is expected to lie. The minimum is then either found by a single variable search or linear search in the direction determined, or approached by taking a fixed step towards it. Since the direction determined is not necessarily correct, the process is iterative. After each minimization step further searches are carried out until a criterion which defines

when the desired minimum has been found is satisfied. Gradient methods for optimization are based on a Taylor expansion [2].

1.3.1.2 Constrained Optimization Methods

Constrained optimization deals with methods for finding the minimum of equation 1.1 subject to constraints of equations 1.2, 1.3.

In considering the variety of algorithms available, there are no reliable rules to determine which method is best. It is most important to use an algorithm that provides acceptable results for the problem.

Using constrained minimization methods, the design variables are modified successively during the design process by moving in the design space from one design point to another [1]. Most methods consist of the following four basic steps:

- a) Determination of the set of active constraints at the current design.
- b) Selection of a search direction in the design space, based on the objective function and the active constraint set.
- c) Calculation of how far to go in the direction found in step b.
- d) Convergence check which determines whether additional modifications in the design variables are required.

MP methods that are used to solve constrained optimization problems may be divided into *direct* and *indirect* methods [1,2]. *Direct* methods deal with the constrained formulation as it is while *Indirect* methods, convert the problem first into an equivalent unconstrained problem and then solve by any method of the available unconstrained techniques. The idea of indirect methods is to penalize constraints violations by adding a sequence of penalty functions to the objective function in such a way that the solutions to the resulting sequence of unconstrained problem tend to a constrained minimum. This approach results in a class of techniques termed Sequential Unconstrained Methods [1,2,3].

1.3.1.2.1 Direct Methods

The available methods which take explicit account of the constraint at each step are the methods of feasible directions, which are intended for problems with general inequality constraints, and it is assumed that the derivatives of the objective and constraint functions are available. Other methods are gradient projection and reduced gradient methods and dual methods. Details of these methods are given in reference [1,3].

1.3.1.2.2 Indirect Methods

In the indirect or sequential unconstrained methods, a constrained problem in optimization is transformed to an unconstrained problem by associating a cost or penalty which penalize violation of the constraints [1]. This cost is included in the objective function evaluation. Consider, for example, the original constrained problem in minimization form [4]:

$$\begin{aligned} &\text{minimize} && f(\mathbf{x}) \\ &\text{subject to} && g_i(\mathbf{x}) \geq 0 \quad i = 1, 2, \dots, n \end{aligned} \quad (1.4)$$

where $\mathbf{x}$ is a vector of design variables, can be transformed into the unconstrained form as

$$\text{minimize} \quad f(\mathbf{x}) + r \sum_{i=1}^n \phi [g_i(\mathbf{x})] \quad (1.5)$$

where ϕ penalty function and r penalty coefficient.

This unconstrained problem is solved with a predetermined value of r . The value of r is then reduced and the problem is solved with this new value of r . These iterations are continued until convergence is obtained.

Among the available methods are the following [1,3]:

1) Exterior penalty function methods:

In these methods all intermediate solutions lie in the infeasible region. One advantage of such methods is that the solution may be started from an infeasible point, eliminating the need for an initial feasible point. A major shortcoming is that we can not stop the search with a feasible solution before the optimum is reached.

2) Interior penalty (barrier)-function methods:

In the barrier function methods all intermediate solutions lie in the feasible region and converge to the solution from the interior side of the acceptable domain. The advantage is that one may stop the search at any time and end up with a feasible and , hopefully, usable design. One drawback is that we have to start the solution always with a feasible design.

3) The augment Lagrange Multiplier method:

Multiplier methods combine the use of penalty functions with that of Lagrange multipliers. When only Lagrange multipliers are employed, the optimum is a stationary point rather than a minimum in the Lagrangian function. When only penalty functions are used, ill-conditioning problems may be encountered during the solution process. The object in combining both is to obtain an unconstrained minimization problem that does not suffer from ill-conditioning. The conditions of optimality are included in the optimization algorithm in order to improve its efficiency and reliability. As a result, the dependency of the method on the choice of the penalty parameters is reduced.

1.3.2 Discrete Variable Optimization Methods

Most applications of optimization methods to structural design have assumed that design variables are continuous. However, many cases exist in which the engineer must consider integer or discrete values of design variables. For example, in civil engineering applications structural sections must usually be selected from an available set [8].

Methods available for discrete variable structural optimization may be divided in three categories: branch and bound, approximations using branch and bound, and genetic algorithms methods [8].

1.3.2.1 Branch and Bound Methods

The classical branch and bound method originated by Land and Doig (1960) was originally developed for convex linear programming problems. This method is theoretically correct for convex problems but has the disadvantage of high computational cost and numerical problems associated with the accuracy of nonlinear programming solutions [7,8].

1.3.2.2 Approximation Methods

Approximation methods provide efficiency but does not guarantee an optimum discrete solution. More than one approach was reported in the use of approximation method, but still these methods faces difficulties and sometimes may lead to an infeasible discrete solution [8]. In general approximation methods can provide a discrete variable optimization capability in a general purpose optimization program, but the user must be aware of its limitations.

1.3.2.3 Genetic Algorithms

Genetic algorithms differ from the classical numerical optimization techniques, they combine the concept of survival of fittest into optimization. The first dissertation to apply GAs to a pure problem of mathematical optimization was Hollestien's (1971) work, which was titled "Artificial Genetic Adaptation in Computer Control Systems". His work was concerned with optimizing functions of two variables.

In the year 1975, Holland published his influential book, "Adaptation and Natural Artificial Systems", and in that same year DeJong completed his important and pivotal dissertation, "An analysis of the Behavior of a class of Genetic Adaptive

Systems". DeJong's study is considered as a milestone in the development of GAs. Subsequent developments of GAs were all based on his work [4].

GAs are robust stochastic search and optimization techniques based on concepts which comes from population genetics. Since Holland, DeJong, and Hollestein's introduction of genetic algorithms as a method of optimization, GAs have been applied to commerce, engineering, mathematics, medicine and pattern recognition with promising results [9]. Recently, some works have efficiently applied GAs to optimum structural design. The robustness of GAs is due to their capacity to locate the global optimum in a multimodal landscape. Multimodal functions exist in engineering problems, for example, designing structures [10].

GAs work with coding of the design variables rather than the variables themselves. This makes GAs computationally suited for treating discrete design variables. GAs also can treat continuous design variables by specifying required precision. Moreover, GAs search from a population of designs, so they can provide a robust search in complex design space [4].

GAs are computationally simple, but powerful in their search for improvement, and they are not limited by restrictive assumptions about the search space, such as continuity or existence of derivatives. The GAs employ a random, yet directed, search for locating the globally optimal solution, for that they are superior to gradient descent techniques as the search is not biased towards the locally optimal solution [8]. On the other hand, they differ from random sampling algorithms due to their ability to direct the search towards relatively prospective regions in the search space.

The name GA is puzzling because the first word refers to a biological science and the second word is borrowed from computer science. An algorithm is a step-by-step procedure for accomplishing some specific task. Many problems can be

readily implemented as computer programs. Thus, an algorithm is the general description of a procedure, and a program is its realization as a sequence of instructions to a computer [11]. In the simplest form of the GA, bit strings play the role of chromosomes, with bits playing the role of genes. An initial population of individuals (bit string) is generated randomly, and each individual receives a numerical evaluation that is then used to make multiple copies of higher-fitness individuals and to eliminate lower-fitness individuals. The process is similar to a natural population of biological creatures where successive generations of creatures are conceived, born, and raised until they themselves are ready to reproduce. This population by population approach is very different than the more typical search methods of engineering optimization. In addition, the search for optimum is done by probabilistic operators, which play a great role in exploring the solution space, such as crossover (exchanging substrings of two parents to obtain two offsprings) and mutation (flipping individual bits) which are applied to the population to produce a new population, or generation of individuals. The GA is considered to be successful if a population of highly fit individuals evolves as a result of iterating this process.

GAs are best suited for unconstrained optimization problems [4,12], for that the structural optimization problem, which is a constrained one, has to be transformed to unconstrained optimization problem. Transformation methods achieve this by either using exterior or interior penalty functions [1,4].

1.4 LITERATURE SURVEY

This section represents a summary of work done in the area of structural optimization by using the genetic algorithms.

D. Goldberg and Samtani [13] has applied the GA in structure optimization of pin jointed structures. Number of design examples solved has shown the robustness

and efficiency of genetic algorithms for the solution of structural engineering optimization problems.

Jenkins [14] presented in his paper the GA as a structural design tool and considered the basic GA operations of selection, cross over and mutation and parameter scaling as well as the analysis and design interfaces. He investigated the fundamental matter of selection and design parameter sensitivity which is an aspect of structure design offering some computational advantage. Results from a number of practical applications in structural design were represented.

Rajeev and Krishnamoorthy [12] have presented a simple GA for optimizing structural systems with discrete design variables. They applied penalty- based transformation method to transform the constrained problem to unconstrained one. The concept of optimization using GA was presented in detail in a 3-bar truss problem. The application of GA to design optimization of a larger problem was illustrated using a 160-bar transmission tower.

Hajela [15] described the implementation of genetic search methods in the optimum synthesis of structural systems. The focus of his paper resides in examining the numerical characteristics of GAs. His paper also presents an adaptation of the GA approach in simultaneous analysis and design. The approach was combined with finite element analysis for the design of truss and frame structures.

Shanker and Hajela [16] described a quasi-procedural approach for the generation of near-optimal structural topologies. Such an approach is characterized by the use of algorithmic processing to enumerate the design space and this information was used in a design methodology directed by heuristic decisions. A stagewise decomposition approach was presented to decompose the given problem into a series of subproblems and generate a population of trial designs. A GA-based

search was then conducted in a discontinuous space with these designs as seeds, to obtain an optimal design for the problem.

Grierson and Pak [17] applied GA for the optimal design of skeletal building structures accounting for discrete sizing, geometrical and topological variables. An approximate design fitness evaluation technique was investigated with the aim to improve the numerical efficiency of the genetic search.

Adeli and Cheng [18] presented the optimization of space structures by integrating a GA with the penalty-function method. In the GA process, they used different types of crossover operations, and their relative merit was investigated. The integrated GA has been implemented in C-Language and was applied to optimization of three space truss structures.

Adeli [19] presented a hybrid GA that maintained the following attractive characteristics of GA. First, no line search is involved in genetic search, second, derivatives of objective function and constraints are avoided. In comparing it with the penalty-function based GA, only a few additional simple function evaluations were needed in the new algorithm. Furthermore, the trial and error approach for the starting penalty function coefficients and the process of arbitrary adjustments were avoided. The developed algorithm was general and could be applied to a broad class of optimization problems.

Adeli and Cheng [20] extended their previous work by presenting two concurrent augmented Lagrangian GAs for optimization of large structures utilizing the multiprocessing capabilities of high-performance computers. The efficiency of the algorithms has been investigated by applying them to four space structures including two high rise building structures. It was observed that the algorithms were suitable for optimization of large structures.

S. Wu and P. Chow [9] presented an application of GA to integrated discrete and configuration optimization of trusses. The optimization problem was mathematically formulated as a constrained nonlinear optimization problem with a mix of discrete sizing and continuous configuration variables.

Leite and Topping [21] presented a new recombination schemes for improving the GA operators. These features respectively increase the exploratory of the GA while simultaneously introducing selective pressure to increase the speed of convergence. These features were designed to ensure the balance between effective exploration and selective pressure to exploit the better solutions which are the main power behind the GAs. Finally, some guidelines were drawn for handling the constraints using transformation methods.

Nukulchai and Witchuwaint [22] investigated the application of GAs to optimize a 32m telecommunication tower, subjected to both the stress and displacement constraints and the load conditions specified by the structural standard for Steel Antenna Towers & Antenna Supporting structures.

Ghasemi and Hinton [23] demonstrated in their paper the use of GAs for size optimization of trusses. They introduced the concept of rebirthing and showed that it is considerably effective for problems involving continuous design variables. 3-bar, 4-bar, 10-bar, 64-bar, 200-bar, and 940-bar 2-dimensional trusses were designed using the algorithm developed. In these problems both continuous and discrete variables were considered.

Rajan [26] developed a procedure for combined sizing, shape and topology design of space trusses. He used both discrete and continuous values to define the cross-sectional areas of the members and the nodal locations. Element connectivity and boundary conditions were treated as Boolean design variables. For handling the formulation of the problem, a modified version of the traditional GA was used.

Rajeev and Krishnamoorthy [27] presented two improved methods for size, configuration, and topology optimization. The first one was a two-phase method, which in addition to arriving at better solutions, also improves the computational efficiency. The second method was based on a variable string length genetic algorithm, which was addressing the topology optimization problem.

1.5 SCOPE OF THE WORK

The literature survey carried out in the previous section reveals the fact that genetic algorithms have found a wide application in structural design. However, most of the algorithms developed do not consider design code requirements which make them not applicable from the every day practicing engineers point of view. In this study, an algorithm is developed for steel roof trusses which is based on British standards (BS 5950). The design method makes use of genetic algorithm and considers design requirements as they are defined in BS 5950. The steel sections for the truss members are selected from standard angle sections.

In this work the intention was to develop an algorithm which designs optimally two dimensional truss structures. The aim was to use a simple and efficient method which could be used by the practicing engineers without need to know deep and complicated mathematical details. The successful results of some recent works concerning a discrete optimization have encouraged the use of GAs in this work. Three algorithms were developed by using GA as the optimization design tool. The first one considers fixed geometry and the optimum problem was to optimize the weight of structure by varying the cross-sectional areas of the truss members. In the second algorithm, the height of truss was taken as a design variable in addition to cross sectional areas. Finally, truss spacing was taken in addition to previous design variables. In these three cases all the design variables were considered to be selected from a discrete set.

In chapter 1 the optimum structural design is introduced with a summary of the available optimization techniques. This is followed by the discrete optimization, in which GA method is explained in general. The last section covers a survey of the previous work carried out in the area of structural optimization which makes use of GA as a solution technique.

Chapter 2 covers the details of genetic algorithms. The basic steps of selection, cross-over and mutation are explained. The mathematical foundations of the GA is highlighted by introducing the theory of schemata. The improvements in GA operators carried out by many researchers were also included.

In Chapter 3 the formulation of optimum design problem of roof trusses based on BS 5950 requirements is explained. Transformation of the constrained problem into an unconstrained one is shown. Application of the genetic algorithm to obtain the solution of the design problem is covered. Three different design problems are considered. In the first, the geometry of the truss is kept fixed, optimum sections are determined for the members. In the second the height of apex is also taken as design variable in addition to member sizes. Hence, optimum geometry is to be determined. In the third the spacing between the trusses is also included as a design variable which made it possible to obtain the most economical spacing and geometry for a truss with a certain topology. The BS5950 requirements were imposed on the constrained optimum problem. The design procedure considers the equal angles as the sections of the truss members. Within this chapter the developed algorithm is applied to a 3-bar truss where the design process is explained in detail.

Chapter 4 contains flow charts of the computer programs developed with the relevant subroutines.

Chapter 5 includes a variety of examples solved by the algorithms developed, and contains the discussion of the results obtained.

In chapter 6, conclusions obtained from this study are given and suggestions for further study are listed.

Chapter 2

GENETIC ALGORITHMS

2.1 INTRODUCTION

Genetic Algorithms are search algorithms based on the mechanics of natural selection and natural genetics. They combine survival of the fittest among string structures with a structured yet randomized information exchange to form a search algorithm [4]. While randomized, GAs are no simple random walk. They maintain balance between exploitation and exploration by efficiently exploiting historical information to speculate on new search points with expected improved performance.

There exist a resemblance between genetic algorithms and nature, which is explained by the following. In nature, individuals best suited to competition for insufficient resources survive. Adapting to a changing environment is essential for the survival of individuals of each species. While the various features that uniquely characterize an individual determine its survival capacity, the features in turn are determined by the individuals genetic content. Specifically, each feature is controlled by a basic unit called a *gene*. The sets of genes controlling features form the chromosomes, the “keys” to the survival of the individual in a competitive environment [10].

In nature, competition among individuals for scarce resources such as food and space results in the fittest individuals dominating over weaker ones. Only the fittest individuals survive and reproduce, this natural phenomenon is called “ the survival

of the fittest ”. Hence, the genes of the fittest survive, while the genes of weaker individuals die out.

The reproduction process generates diversity in the gene pool. Evolution is initiated when the genetic material (chromosomes) from two parents recombines during reproduction. New combinations of genes are generated from previous ones; a new gene pool is created. Specifically, the exchange of genetic material among chromosomes is called crossover. Repeated selection and crossover cause the continuous evolution of the gene pool and the generation of individuals that survive better in a competitive environment [9,10].

Holland proposed GAs in the early 1970 as computer programs that imitate the evolutionary processes in nature. GAs handles a population of efficient solutions to an optimization (or search) problem. Specially, they operate on encoded representations of the solutions, equivalent to the genetic material of individuals in nature, and not directly on the solutions themselves. Holland’s GA encodes the solutions as strings of bits from a binary alphabet. Each string of 0’s and 1’s is the encoded version of a solution to that problem. Each solution is associated with a fitness value that reflects how good it is, compared with other solutions in the population [24]. The higher the fitness value of an individual, the higher its chances of survival and reproduction and the larger its representation in the subsequent generations. Recombination of genetic material in GAs is simulated through a crossover mechanism that exchanges portions between string. Another operation, called mutation, causes random alteration of the bits of strings. Mutation too has direct analogy from nature and plays the role of regenerating lost genetic material.

2.2 OPTIMIZATION AND GENETIC ALGORITHMS

Before examining the mechanics and power of a simple genetic algorithm, we must be clear about our goals when we say we want to optimize a function. The optimization seeks to improve performance toward some optimal point or points.

Note that this definition has two parts:(1) we seek improvement to approach some (2) optimal point. There is a clear distinction between the *process* of improvement and the *distination* or optimum it self. Yet, in judging optimization procedures we commonly focus only upon convergence and forget entirely about interim performance [4].

The following outlines some very fundamental ways that make GAs different from more traditional optimization and search methods, these are given as follows [4]:

1. GAs work with a coding of the parameter set, not the parameters themselves.
2. GAs search from a population of points, not a single point.
3. GAs use objective function information, not derivatives or other auxiliary knowledge.
4. GAs use probabilistic transition rules, not deterministic rules.

GAs require the natural parameter set of the optimization problem to be coded as a finite-length string over some finite alphabet. With more traditional methods we would try many values for the parameter x . With GAs, the first step of our optimization process is to code the parameter x as a finite-length string. GAs exploit coding similarities in a very general way.

In many traditional optimization methods, the search for the optimum solution to the problem starts at some random point (initial guess), and if a slight modification improves the quality of your solution, continue in that direction ; otherwise, go in the other direction. For the case of multimodal functions, where many peaks exist in the solution space, then the final solution will very much depend upon the initial guess. This might lead to a local optimum. By contrast, GAs search is conducted in a more effective manner. GAs search climbs many peaks in parallel, in which an effective and robust search is implemented and the probability of reaching to a local optimum is very much reduced as compared with traditional methods [24].

Many search techniques require much auxiliary information in order to work properly. For example, gradient techniques need derivatives in order to be able to climb the current peak. By contrast, GAs have no need for this auxiliary information. To perform an effective search for better and better structures, they only require objective function values associated with individual strings. This characteristic makes GA a more canonical method than many search schemes.

Unlike many methods, GAs use probabilistic transition rules to guide their search. The use of probability does not suggest that the method is some simple random search, GAs use random choice as a tool to guide a search toward regions of the search space with likely improvement.

Taken together, these four differences - direct use of a coding, search from a population, blindness to auxiliary information, and randomized operators contribute to a genetic algorithm's robustness and resulting advantage over other more commonly used techniques.

2.3 WORKINGS OF A SIMPLE GENETIC ALGORITHM

This section covers working of a simple genetic algorithm. The mechanics of a simple GA are surprisingly simple, involving nothing more complex than copying strings and swapping partial strings. Simplicity of operation and power of effect are two of the main attractions of the GA approach [4].

2.3.1 Generation of Initial Population

The central data structure in GAs is a population of string (usually fixed size) which is used to represent or code candidate solutions (design variables) [4].

In many applications, the candidate solutions that comprise the population are encoded as bit-strings, composed by characters copied from the binary (2-cardinal) alphabet $\{0,1\}$ [25]. This representation is analogous to chromosomes in a biological system and the binary digits, 0 and 1, are analogous to genes. The

binary representation, by its nature is convenient for implementation on computers, and each string represents a design.

A bit string representation for one design prescribes how binary strings, or bit fields for each design variable are placed end to end. Several such designs constitute a population. Fig.2.1 illustrates an example of a binary representation of a candidate solution that involves problem variables x_1 , x_2 , x_3 , and x_4 for which the bit field sizes are specified by bit string format '5,7,4,9'.

11010	0011011	0010	011010010
x_1	x_2	x_3	x_4

Fig. 2.1 A sample representation of a binary string
(An individual in a population)

The string length λ depends on the number of discrete values available to be mapped [9], i.e.

$$2^\lambda = \text{Number of discrete values} \quad (2.1)$$

For example, if there are 32 discrete values, then a binary digit substring of length five, is used for the design variables. For each string length adopted, the total capacity must be used, since each individual string must represent a real value of the corresponding variable. If the number of values of the variable lies between one string length and the next, we must choose the longer string and fill in the excess capacity by duplication. This is crude but effective. It would be normal practice to arrange the lists in ascending numerical order, but this is not essential when using the basic method. In general, non uniform string lengths are used, with longer strings being used for the variables where more information is to be represented [14].

The physical value of a design variable is determined by a mapping relationship which is illustrated by four bitstring, as shown in Table 2.1. The four-bit string can

be decoded to an unsigned decimal integer between 0 and 15, and the unsigned decimal integer can map to any one of the set of 16 discrete values. For example, the string 1010 of length five can be decoded to an unsigned decimal integer 11 as follows:

$$1010 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 11$$

Then the string 1010 is mapped to the discrete value D_{12} .

Table 2.1 Mapping relationships between strings and discrete values.

Binary digit string of length 4	Unsigned decimal integer	Discrete values
0000	0	D_1
0001	1	D_2
0010	2	D_3
1010	11	D_{12}
1111	15	D_{16}

The initial generation (generation 0) is produced by random selection of the bits in each string. A reasonable population size is to be chosen. If a small-size population is used in the genetic search, then the search will be constrained in a subspace instead of the global space because the number of different types of alternative combinations of strings is not enough to span the entire search space [20]. Thus, the global or near-global optimum may never be obtained. On the other hand, if a large-size population is used in each generation, a great amount of computational time will be required to complete the whole optimization process.

2.3.2 Operators of Simple Genetic Algorithm

GAs get their power from the genetic operators. The various genetic operators that have been identified are: reproduction, cross over, mutation, dominance, inversion,

deletion, translocation, segregation, speciation, migration and sharing. Depending on the nature of the problem and the requirements for performance, GAs can be improved by applying more and more of these operators [4]. This section concentrates on a simple GA that yields good results in many practical problems and is composed of the following three operators :

1. Reproduction
2. Crossover
3. Mutation

2.3.2.1 Reproduction Operator

The reproduction operator highlights the survival of the fittest in GAs. It is a process in which individual strings are copied according to their objective function values, f (biologists call this function the fitness function). Fitness is defined as a figure of merit (credit), which is either maximized or minimized [4]. Copying strings according to their fitness values means that strings with a higher value have a higher probability of contributing one or more offspring in the next generation.

The reproduction operator may be implemented in algorithmic form in a number of ways. Perhaps the easiest is to create a biased roulette wheel (biased selection) where each current string in the population has a roulette wheel slot sized in proportion to its fitness [4]. As an example, consider a sample population of four strings which has objective or fitness function value $f(F_i)$ as shown in Table 2.2

Table 2.2 Sample Problem Strings and Fitness Values

No.	String (i)	Fitness(F_i)	% of Total (p_{select_i})
1	01101	169	14.4
2	11000	576	49.2
3	01000	64	5.5
4	10011	361	30.9
Total		1170	100.0

Summing the fitness over all four strings, we obtain a total of 1170. The percentage of population total fitness is also shown in the table. This percentage value represents the probability of selecting an individual and is calculated by

$$P_{select_i} = F_i / \sum F \quad (2.2)$$

The corresponding weighted roulette wheel for this generation's reproduction is shown in Fig. 2.2. To reproduce, we simply spin the weighted roulette wheel thus defined four times. For the example problem, string number 1 has a fitness value of 169, which represents 14.4% of the total fitness. As a result, string 1 is given 14.4% of the biased roulette wheel, and each spin turns up string 1 with probability 0.144. Each time we require another offspring, a simple spin of the weighted roulette wheel yields the reproduction candidate. In this way, more highly fit strings have a higher number of offspring in the succeeding generation.

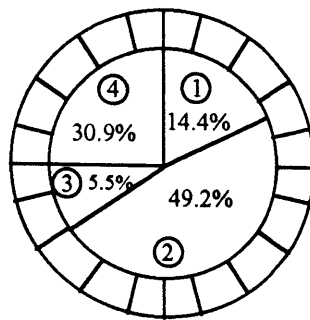


Fig. 2.2 Simple reproduction allocates offspring strings using a roulette wheel with slots sized according to fitness.

Once a string has been selected for reproduction, an exact duplicate of the string is made. This string is then entered into a mating pool, a tentative new population, for further genetic operator action.

Biased selection, may be regarded as the exploitative part in the search heuristic in GA. It is necessary but not sufficient for efficient search. It is not sufficient because the search space is not explored. Within a few generations under fitness-proportional reproduction the population will comprise only copies of the best bit string in the initial population. It is very unlikely that this bit string represents the

global solution, since the population is at any time only a small sample of the total population (search space). In general, the phenomenon whereby the fittest, yet not globally optimal, strings dominate the population is called premature convergence [25].

2.3.2.1.1 Fitness Function

In many problems, the objective is more naturally stated as the minimization of some cost function $f(x)$. It is necessary to map the underlying natural objective function to a fitness function form through one or more mappings. In normal operation research work, to transform a minimization problem to a maximization problem we simply multiply the cost function by a minus one. With GAs the following cost-to-fitness transformation is commonly used [4],

$$\begin{aligned} F(x) &= C_{\max} - f(x) && \text{when } f(x) < C_{\max} \\ &= 0 && \text{otherwise} \end{aligned} \quad (2.3)$$

There are a variety of ways to choose the coefficient $C_{\max}$. $C_{\max}$ may be taken as an input coefficient, as the largest f value observed thus far, as the largest f value in the current population, or the largest of the last k generations. Perhaps more appropriately, $C_{\max}$ should vary depending on the population variance [4].

In GAs we have the opportunity and perhaps the duty to regulate the level of competition among members of the population to achieve the ultimate algorithm performance we desire. This is precisely what is done when performing fitness scaling [4].

2.3.2.1.2 Fitness Scaling

Regulation of the number of copies is especially important in small population in genetic algorithms. At the start of GA runs it is common to have a few extraordinary individuals in a population of mediocre colleagues. If left to the normal selection rule, eqn. (2.2), the extraordinary individuals would take over a

significant proportion of the finite population in a single generation, and this is undesirable, a leading cause of premature convergence. Late in a run, there may still be significant diversity within the population best fitness. If this situation left alone, average members and best members get nearly the same number of copies in future generations, and this survival of the fittest necessary for improvement becomes a random walk among the mediocre. In both cases, at the beginning of the run and as the run matures, fitness scaling can help.

One useful scaling procedure is linear scaling [4]. Linear scaling requires a linear relationship between F' and F as given in the following:

$$F' = a \times F + b \quad (2.4)$$

where coefficients a and b may be obtained as given in reference [4]. Another scaling proposed by Goldberg [4] in which the control is on the number of offspring given to the population member with maximum raw fitness, and is given as

$$F'_{\max} = C_{mult} \times F_{avg} \quad (2.5)$$

where C_{mult} is the number of expected copies desired for the best population member. $C_{mult} = 1.2$, for typical small populations ($n=50$ to 100).

2.3.2.2 Crossover Operator

The crossover operator does the work of exchanging the information between the two selected mates which are generated by the reproduction operator, and out of this a new individual will be created. This operator is required to counter balance depletion of diversity in the population, caused by the reproduction operator. The crossover operator comprises the explorative part of search heuristic [25]. Figure 2.3 illustrates a simple crossover operation, which may be proceeded in two steps.

First, members of the newly reproduced strings in the mating pool are mated at random. Second, each pair of strings undergoes crossing over as follows: an integer

k along the string is selected uniformly at random between 1 and the string length less one $[1, \lambda-1]$. Two new strings are created by swapping all characters between positions $k+1$ and λ inclusively [4].

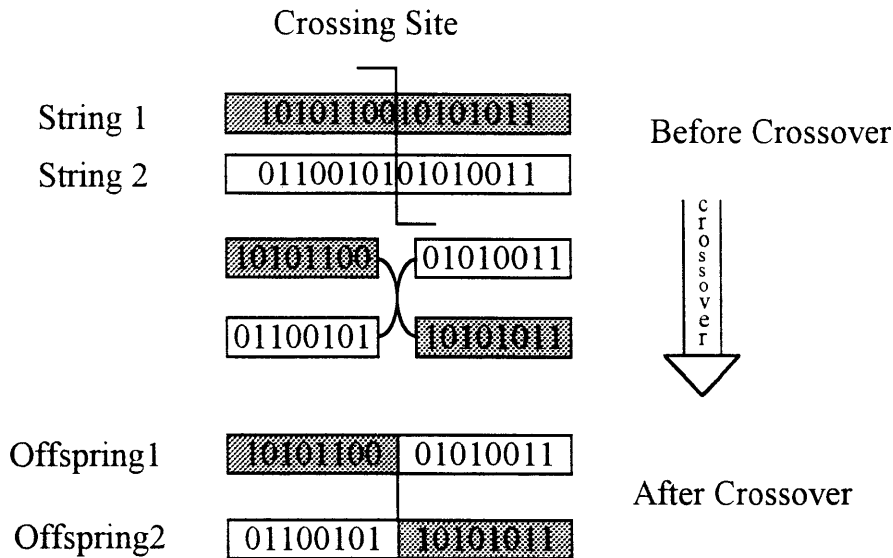


Fig. 2.3 A simple crossover illustration showing the alignment of two strings and the partial exchange of information, using a cross site chosen randomly.

Though reproduction and crossover operators look very simple, their combined action is responsible for much of GAs power. From a computer implementation point of view, they involve only random number generation, string copies, and some partial string exchanges.

2.3.2.3 The Mutation Operator

After the reproduction and crossover comes the work of the mutation operator, which plays decidedly secondary role in the operation of GAs. Mutation is needed because, even though reproduction and crossover effectively search and recombine extant notions (surviving ideas), occasionally they may loose some potentially useful genetic material (1's or 0's at particular locations). In artificial genetic systems, the mutation operator protects against such an irrecoverable loss. In the simple GA, mutation is the occasional(with small probability) random alteration of the value of a string position. In the binary coding, this simply means changing a 1

to a 0 and vice versa. By itself, mutation is a random walk through the string space. When used sparingly with reproduction and crossover, it is an insurance policy against premature loss of important notions [4].

2.4 MATHEMATICAL ANALYSIS OF GAS:

SCHEMA THEOREM

Holland and Goldberg have captured the essence (spirit) of GA mechanics by introducing the Schema Theorem. The concept of schema, the substring pattern, is introduced as the mathematical basis to clarify the good performance of GAs, and to distinguish GAs from random search methods [4].

Analysis of GAs begins with the concept of a search space. The GA can be viewed as a procedure for searching the space of all possible binary strings of a fixed length λ , denoted as

$$\{0,1\}^\lambda = \text{Set of strings of length } \lambda \quad (2.6)$$

Under this interpretation, the algorithm is searching for points in the λ -dimensional space (2.6) that have high fitness. The search space is identical for all problems of the same size (same λ), but the locations of good points will generally differ. The surface defined by the fitness of each point is sometimes referred to as the fitness landscape [4]. The longer the bit strings, corresponding to higher values of λ , the larger the search space is, growing exponentially with the length of λ . For problems with a sufficiently large λ , it is not feasible for any algorithm to examine more than a small fraction of the search space. For example, $\lambda=64$ defines a search space that is too large to search exhaustively (thoroughly) with current computer technology. Because only a small fraction of a search space this size can be examined, it is unreasonable to expect an algorithm to locate the global optimum in the space. A more reasonable goal is to search for good regions of the search space corresponding to regularities in the problem domain [11].

The genetic algorithms exploits the higher target regions of the solution space, because successive generations of reproduction and crossover produce increasing numbers of strings in those regions. The algorithm favors the fittest strings as parents, and so above average strings (which fall in target regions) will have more offspring in the next generation [24]. Holland introduced the notion of a “schema” to explain how GAs search for regions of high fitness. Schemata are patterns that are used to explain the behavior of GAs and are not processed directly by the algorithm.

A schema is a similarity template describing a subset of strings with similarities at certain positions. It is defined over the alphabet $\{0,1,*\}$ that describes a pattern of bit strings in the search space (2.6). In other words, a schema represents a subset of all possible strings that have the same bits at certain string positions [10]. As an example, consider strings with five bits. A schema $**000$ represents strings with 0s in the last three positions: the set of strings 00000, 01000, 10000, and 11000. Similarly, a schema $1*00*$ represents the strings 10000, 11000, 10001, and 11001. Each string represented by a schema is called an *instance* of the schema. The $*$ is simply a notational device that allows description of all possible similarities among strings of a particular length and alphabet [4]. The schema $*****$ represents all possible strings of five bits.

Holland schemata have 2 elementary properties, which are important for the derivation of the schema theorem. The first is called schema order, O , which is the number of fixed positions, e.g. the schema $**000$, is of order 3. The second property is called the defining length, L , of a schema, which is the difference between the last fixed position and the index of the first fixed position. Hence, the defining length of $**000$ is 2, while the defining length of $1*00*$ is 3. Any specific string is simultaneously an instance of 2^λ schemata (λ is the string length) [25].

The above leads us to the important theoretical result about GAs which is the Schema Theorem. It states that the observed best schemas are expected to receive an exponentially increasing number of samples in successive generations.

When we consider the effects of selection, crossover, and mutation on the rate at which instances of a schema increase from generation to generation, we see that proportionate selection increases or decreases the number in relation to the average fitness value of a schema. Neglecting crossover, a schema with a high average value grows exponentially to win its relevant schema competition. However, a high average fitness value alone is not sufficient for a high growth rate. A schema must have a short defining length too. Because crossover is disruptive, the higher the defining length of a schema, the higher the probability that the crossover point will fall between its fixed positions and an instance will be destroyed. Thus schemata with high fitness values and small defining lengths grow exponentially with time. This is the essence of the schema theorem, which is called the building block hypothesis, and the following equation is a formal statement of the schema theorem [4]:

$$m(H, t + 1) \geq m(H, t) \times \frac{F(H, t)}{\bar{F}(t)} \left[1 - p_c \frac{\delta(H)}{\lambda - 1} p_m o(H) \right] \quad (2.7)$$

where,

$F(H, t)$ = average fitness value of schema H in generation t

$\bar{F}(t)$ = average fitness value of the population in generation t

p_c = crossover probability

p_m = mutation probability

$\delta(H)$ = defining length of the schema H

$o(H)$ = order of the schema H

$m(H, t)$ = expected number of instances of schema H in generation t

λ = the number of bit positions in a string

The factor $p_c \frac{\delta(H)}{\lambda - 1}$ gives the probability that an instance of the schema H is disrupted by crossover, and $p_m o(H)$ gives the probability that an instance is disrupted by mutation.

The above schema theorem comprises an expression for the expected growth or decay rate of a particular schema in a population t . Simply stated, this expression is a factorization of two effects: an effect due to exploitation and an effect due to exploration.

First, exploitation is performed by selective reproduction. Consider a particular schema in a population. This schema will grow as the ratio of estimated average fitness of the schema H to the average fitness of the population. That is, a schema with fitness values above the population average will receive an increasing number of instances in the next generation, while schemata with fitness values below the population average will receive a decreasing number of instances. Importantly, this expected behavior is carried out with every schema contained in the population-in parallel and with no special bookkeeping or special memory other than the population itself. It should be noted, however, that there is a statistical error in the estimate of $F(H,t)$, called the sampling error. This may mislead the GA, especially if the population size is relatively small.

The fitness of any bit string in the population gives some information about the average fitness of the 2^l different schemas of which it is an instance, so an explicit evaluation of a population of M individual strings is also an implicit evaluation of a much larger number of schemas. This is referred to as *implicit parallelism* [24]. At the explicit level, the GA searches through populations of bit strings, but the genetic algorithm's search can also be interpreted as an implicit sampling process. Feedback from the fitness function, combined with selection and recombination, biases the sampling procedure over time away from those schemas that give negative feedback (low average fitness) and toward those that give positive feedback (high average fitness). Ultimately, the search procedure should identify regularities, or patterns, in the environment that lead to high fitness, and because the space of possible patterns is larger than the space of possible individuals (3^l versus 2^l), implicit parallelism is potentially advantageous.

The exploration is required in order to counterbalance depletion (consumption) of the population caused by exploitation (fitness proportional reproduction). Exploration comes down to variation, imposed on the strings by recombination and mutation, and leads to the best search efficiency. In order to understand the exploration effect, recall that any strings of length λ comprises 2^λ schemata. Therefore, when such a string recombines (crossover) or mutates, it switches from one set of 2^λ contained schemata to another. The original and the new schemata set will, in general, not mutually overlap completely, i.e. some of the original schemata have vanished. Thus, taking all strings of the population into account, the total impact of recombination and mutation across the population might be that a particular schema vanishes entirely from the population. Such a schema is said to be disrupted, i.e. the information it represents is lost from the population.

2.5 MODIFICATIONS TO THE SIMPLE GENETIC ALGORITHM

GAs have approved as a class of general purpose search strategies that strike a reasonable balance between exploration and exploitation. Indeed, the theoretical analysis of GAs suggest that they manage this tradeoff in a near optimal way. The power of these algorithms is derived from a very simple heuristic assumption: that the best solutions will be found in regions of the search space containing relatively high proportions of good solutions; and, that these regions can be identified by careful and robust sampling of the space [25].

Despite the demonstrated advantages of these algorithms, the feeling persists that there is much to learn about how to effectively implement a genetic algorithm. While the performance of most implementations is comparable to or better than the performance of many other search techniques, it still fails to live up to the high expectations caused by the theory. Estimates based on finite samples eventually have a sampling error associated with them. Repeated iterations of the algorithm

compound the sampling error and lead to premature loss of diversity in the population with search converging to a local optimum solution.

Over the last decade, considerable research has focused on improving GA performance by limiting these finite stochastic effects. DeJong (1975) has shown that bounding the error in the selection probabilities for each string substantially improves the performance of the algorithm. Others have tried reducing error by using more accurate selection procedures. Efficient implementations of the proportionate selection scheme such as the stochastic remainder technique and the deterministic sampling technique, have been proposed to reduce sampling errors. Selection mechanisms such as rank-based selection, and tournament selection have been proposed as alternatives to proportional selection. Crossover mechanisms such as two-point, multi-point and uniform have been proposed as improvements to the traditional single point crossover technique [10].

Selection mechanisms and scaling:

The proportionate selection scheme allocates offspring based on the ratio of a string's fitness value to the population's average fitness value. In the initial generations of the GA, the population typically has a low average fitness value. The presence of a few strings with relatively high fitness values causes the proportionate selection scheme to allocate a large number of offspring to these "super-strings", and they take over the population, causing premature convergence. A different problem arises in the later stages of the GA when the population has converged and the variance in string fitness values becomes small. The proportionate selection scheme allocates approximately equal numbers of offspring to all strings, thereby depleting the driving force that promotes better strings. Scaling mechanisms such as linear scaling, sigma truncation, and power law scaling overcome these two problems. Linear scaling of fitness works well except when negative fitness calculations prevents its use. This is most usually a problem later in the search process when most population members are highly fit. Linear

scaling computes the scaled fitness value as given in equation (2.4). For the case of power law scaling, the fitness values are re-adjusted according to

$$F' = \left[1 - \frac{F_{\max} - F}{F_{\max} - F_{\min}} \right]^3 \quad (2.8)$$

where F' is scaled fitness and F is the raw fitness.

In the year 1981 Brindle proposed six selection schemes, the following are four of them [4]:

1. Stochastic Sampling with replacement.
2. Deterministic Sampling.
3. Remainder Stochastic sampling without replacement.
4. Stochastic Tournament (competition).

Stochastic sampling with replacement is another name for roulette wheel selection, which was discussed in an earlier section.

Deterministic sampling is a scheme which work with finite population size to reduce the stochastic errors associated with roulette wheel selection. In this scheme the probabilities of selection are calculated as given in equation (2.3). Then the expected number of individuals for each string num_i is calculated as follows,

$$num_i = n \times P_{\text{select}} \left[\frac{F_i}{F_{\text{avg}}} \right] \quad (2.9)$$

where, n is the total number of individuals in a population.

Each string is allocated samples according to the integer part of the num_i value, and the population is sorted according to the fraction parts of the num_i value. The remainder of the strings needed to fill the population is drawn from the top of the sorted list.

In the *remainder stochastic* method, each string is allocated samples according to the integer part of the value obtained from eqn. (2.9). The fraction parts of the expected number values are treated as probabilities. One by one, weighted coin tosses are performed using the fractional parts as success probabilities. For example, a string with an expected number of copies equal 1.23 would receive a single copy surely and another with probability 0.23. This process continues until the population is full.

In the fourth procedure which Brindle was calling a ranking method, the selection probabilities are calculated normally and successive pairs of individuals are drawn using wheel selection. After drawing a pair, the string with higher fitness is declared the winner, inserted in the new population, and another is drawn. This process continues until the population is full.

Crossover and Mutation Mechanisms:

Because of their importance to GA functioning, much of the literature has been devoted to different crossover techniques and their analysis. The following discusses the important techniques.

Traditionally, GA researchers set the number of crossover points at one or two. In the two-point crossover scheme, two crossover points are randomly chosen and segments of the strings between them are exchanged. Two point crossover eliminates the single-point crossover bias toward bits at the ends of strings.

An extension of the two-point scheme, the multipoint crossover, treats each string as a ring of bits divided by k crossover points into k segments. One set of alternate segments is exchanged between pair of strings to be crossed [21].

Uniform crossover exchanges bits of a string rather than segments. At each string position, the bits are probabilistically exchanged with some fixed probability. The

exchange of bits at one string position is independent of the exchange at other positions.

Empirical and theoretical studies have compared the merits of various crossover operators, particularly two point and uniform crossover. At one end, uniform crossover swaps bits irrespective of their position, but its higher disruptive nature often becomes a drawback. Two-point and single-point crossover preserve schemata because of their low disruption rates, but they become less exploratory when the population becomes homogeneous.

Others have introduced adaptive probabilities for both crossover and mutation operators, and the GA that implement these operators is called adaptive genetic algorithm AGA. In the AGA the probabilities of crossover and mutation are varied depending on the fitness values of the solutions. High fitness solutions are protected while solutions with below average fitnesses are totally disrupted [31].

A related issue is the interplay between the population size and the type of crossover. Empirical evidence suggests that uniform crossover is more suitable for small populations, while for larger populations, the less disruptive two-point crossover is better. Uniform crossover's disruptiveness helps sustain a highly explorative search in small populations. The inherent diversity in larger populations reduces the need for exploration and makes two-point crossover more suitable.

Mutation operator in a simple GA occurs at fixed rates and it is a simple operator, flipping bits at random with a uniform distribution along the string. However, it is generally agreed that for most problems mutation plays a less significant role in the algorithm as the optimization proceeds. The probability of mutation can therefore be reduced as the procedure continues. Topping observed that the use of relatively high mutation rates at the start of the GA runs may be an efficient aim at preventing premature convergence [21].

Chapter 3

OPTIMUM DESIGN OF ROOF TRUSSES VIA GENETIC ALGORITHMS

3.1 INTRODUCTION

Design of a roof truss involves in making decisions about its topology, geometry, member sizes and the spacing between trusses. All these are to be selected in such a way that the final truss should be capable of resisting to the loads that may act during its life time and that the truss should be economical to be produced. In this selection, designer faces a difficult task of finding the most suitable truss among very large number of practically available sets. First decision involves in finding the topology. There exists a wide variety of different shapes in practice, some of which are shown in Figure 3.1.

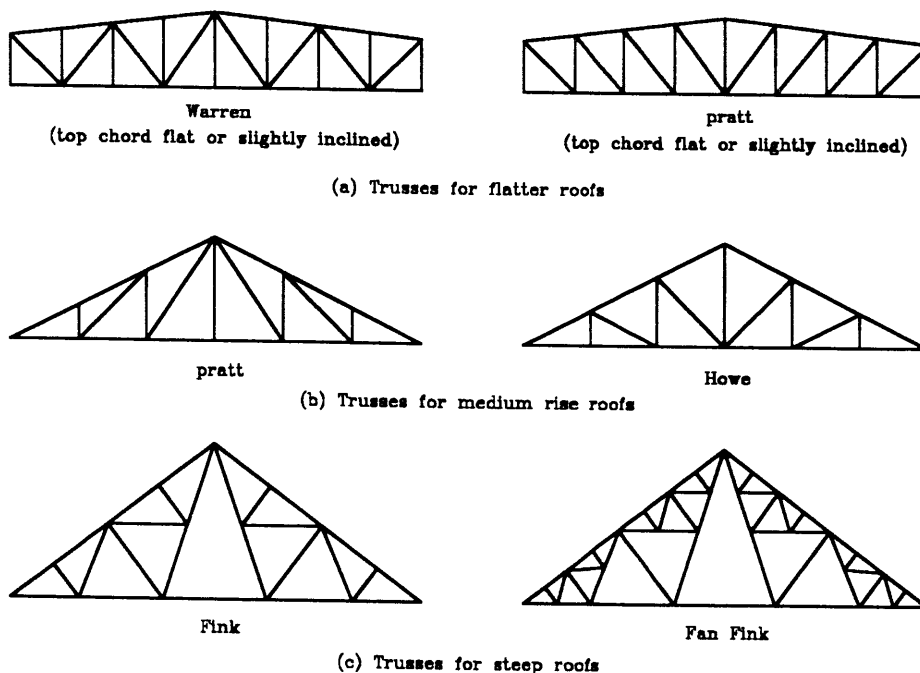


Fig. 3.1 Types of trusses

After deciding the topology, designer has to select the geometrical dimensions of the truss for a span that is to be used. Furthermore, it is also necessary to decide a suitable spacing between roof trusses. Once the topology, geometry and spacing are determined, the truss is then analyzed under various load conditions to find the forces in the members. This process requires the knowledge of cross sectional areas, if the truss is statically indeterminate. The selection of cross sectional areas is carried out from the set of standard sections which are practically available. These available sections are discrete in nature, but not continuous. Hence, the objective of the designer becomes one of obtaining the most economical truss while satisfying the safety requirements. A safe truss is the one which has members with sufficient compressive resistance and tensile capacity to withstand the compressive and tensile forces that may develop in the members due to various loading cases. In addition to this strength requirement, it is also necessary for the truss to deflect within practically acceptable limits so that excessive deformations do not cause damage in the roof cover. As it is known, the designer is not free in achieving the above requirements. He / she has to follow the design codes which intends to uniformize the design process while attaining the safety requirements.

Consequently, the design of roof trusses becomes a problem of finding a truss with the optimum topology, geometry, spacing and member sizes which are to be selected from a discrete set while satisfying the code requirements.

3.2 OPTIMUM DESIGN OF ROOF TRUSSES

The design problem described above can be modeled as an optimum design problem, where the objective can be minimizing the weight or the cost of the truss while satisfying the design code requirements.

3.2.1 The Mathematical Model

The optimization problem of a plane truss can be written as follows:

$$\text{Minimum } F = f(x_n) \quad x_n = 1, \dots, ndv$$

subject to:

$$\begin{aligned}
 g_j(x_n) &= \delta_j - \delta_{ju} \leq 0 & j &= 1, \dots, nrj \\
 g_k(x_n) &= F_{tk} - P_{tk} \leq 0 & k &= 1, \dots, ntm \\
 g_m(x_n) &= F_{cm} - P_{cm} \leq 0 & m &= 1, \dots, ncm \\
 g_l(x_n) &= \lambda_l - \lambda_{lu} \leq 0 & l &= 1, \dots, ncm
 \end{aligned} \tag{3.1}$$

The objective function, $f(x_n)$, of the optimization problem is taken as to minimize the weight of the roof truss. And the vector of design variables x_n , are selected to represent the following: the sectional designation of the truss members which is to be selected from the list of standard steel angle sections, supplied lists of practical height of roof truss apex, and spacing between trusses respectively. The constant ndv represents the total number of design variables in the optimization problem (3.1). The constraints imposed on the optimum design problem are all taken as the limit states specified by the BS5950. The first $g_j(x_n)$ constraints represents the serviceability limit state, which states that the horizontal or vertical displacement δ_j of the restricted joint is not to exceed the allowable displacement δ_{ju} . The second is the ultimate limit state which includes the satisfaction of strength criteria. This covers tension and compression capacity constraints of respective truss members. In order not to have the member reach yielding, tensile force in member F_{tk} should not exceed the tension capacity P_{tk} of the net cross sectional area of the member. And to avoid buckling of a compression member, compressive force F_{cm} in the member should not exceed the compression capacity P_{cm} of the member accompanied with the slenderness ratio constraint $g_l(x_n)$, where λ_l of compression member is to be less than the maximum slenderness λ_{lu} .

3.2.2 Objective Function

Since the objective function in this study is taken as to minimize the weight $f(x)$ for the roof truss, it can be written as :

$$f(x) = \sum \rho A_i L_i \tag{3.2}$$

Where, A_i = the cross sectional area of the i th member.
 L_i = the length of the i th member.
 ρ = the density of steel.

3.2.3 Design Variables

In this study, three different optimum design problems are considered for a fixed truss topology each having different collection of design variables. The first optimization problem deals with optimizing a roof truss with fixed geometry. In this case the design variables involved in obtaining the minimum weight of the truss are the section designation of the truss members. In order to have greater improvement in the design, the structural layout optimization, which include geometry or topology variation, was used in which the optimization process treats the coordinates of joints and the cross-sectional sizes as design variables which are optimized simultaneously. In this case the y-coordinate of truss joints are the representatives design variables for the roof truss geometry. In the third optimization problem, the spacing between trusses is the third type of design variable in addition to the previously mentioned types of design variables. In all these cases the design variables are taken as discrete values. The Steel section designations have to be chosen from a list of discrete values and the height of truss is selected from a set of discrete values instead of continuous values by supplying the program with a list of practical height values. The same applies for the spacing between trusses. Due to the fact that design variables being discrete, makes it very suitable to use genetic algorithm for obtaining the solution of optimum design problem.

3.2.4 Design Constraints

The design requirements in this study are taken from the British Standard BS5950 for the design of the truss members.

These requirements are described as listed in the following:

- 1) The maximum deflection in the truss should be less than or equal to span/200 (BS5950, clause 2.5.1)
- 2) The tension capacity of a tension member should be more than or equal to tensile force in the member (BS5950, clause 4.6.1)
- 3) The compression capacity of a compressive member should be more than or equal to compressive force in the member (BS5950, clause 4.7.4)
- 4) The slenderness ratio of compression member should be less than or equal to 180 and 350 (BS5950, clause 4.7.3).

The above design restrictions are described in more detail in the following sections. All these restrictions can be expressed as inequalities as shown in equation (3.1).

3.2.4.1 Displacement Constraints

According to BS5950, the maximum deflection of the restricted joints in a truss under unfactored imposed loads, should be less than its span/200. This requirement can be written as

$$\delta_j - \delta_u \leq 0 \quad (3.3)$$

where δ_j is the vertical displacement of restricted joint and δ_u is its upper bound specified by the BS5950 code.

3.2.4.2 Member Capacity Constraints

Application of loads to a structure generates internal forces and stresses in its elements. Due to the fact that the flexural stiffnesses of members are ignored in the analysis of trusses, members are assumed to be connected to pin joints, the only forces developed in the members are tensile or compressive forces. These forces could either be less than or equal to or greater than the member capacities, depending on the cross-sectional areas of the sections adopted for these members. If the member force is more than its capacity, such member will not be able to resist the applied force. Hence, it is necessary that the cross-sectional areas of the

members should be selected in such a way that the member force will always be less than or equal the member capacity.

For the case of tension member, the tensile capacity constraint is to be imposed, and has the following form:

$$F_{tk} - P_{tk} \leq 0 \quad (3.4)$$

where, F_{tk} = Tensile force in member k

P_{tk} = Tension capacity of member k

which is computed from $P_y \times A_e$, where P_y is the yield strength of steel, which depends on the grade and thickness of the steel section (BS5950, table 6), & A_e is the steel section effective area (BS5950, clause 4.6.3).

In the case of a compression member, the compression resistance should be more than the compressive force in the member. This can be expressed as

$$F_{cm} - P_{cm} \leq 0 \quad (3.5)$$

where, F_{cm} = Compressive force in member m

P_{cm} = Compression Resistance of member m

= $p_c \times A_g$, p_c is the compressive strength of member m
(BS5950, clause 4.7.4), A_g is the gross area of the section.

Furthermore, the slenderness ratio of a compression member should be less than the certain limits specified in BS5950. This constraint can be expressed as

$$\lambda_m - \lambda_{all} \leq 0 \quad (3.6)$$

where, λ_m = slenderness of compression member m

= L_E/r , L_E is the effective length of member m , and r is the radius of gyration of its cross section about the relevant axis.

λ_{all} = 180 for members resisting loads other than wind loads.

= 350 for any member normally acting as a tie but subject to reversal of stress resulting from the action of wind.

3.3 GENETIC ALGORITHM IMPLEMENTATION

Among the available optimization techniques mentioned in chapter 1, GA was selected as the optimization tool to obtain the optimum design of roof trusses due to its ability of finding solution to the problems with discrete variables as well as its robustness. GAs can be used directly to solve unconstrained optimization problems only, and because most of engineering problems are constrained ones, the constrained optimization problem is to be transformed into an unconstrained one before the GA can be employed. In GAs the constraints are conveniently handled by transformation methods by either using exterior or interior penalty functions. Such transformations are ideally suited for sequential searches. GAs perform the search in parallel using population of points in the search space. Hence, traditional transformations using penalty or barrier functions are not appropriate for GAs [4,12].

A formulation based on the violations of normalized constraints is proposed in this study.

The constraints given in equation 3.1 are expressed in normalized form as follows:

$$\begin{aligned} \frac{\delta_j}{\delta_{ju}} - 1 &\leq 0, \\ \frac{F_{tk}}{P_{tk}} - 1 &\leq 0, \quad \frac{F_{cm}}{P_{cm}} - 1 \leq 0, \quad \frac{\lambda_j}{\lambda_{ju}} - 1 \leq 0 \end{aligned} \quad (3.7)$$

After the constraints are all evaluated, a violation coefficient C computed in the following manner:

$$\begin{aligned} \text{if } g_j(x_n) > 0 & \quad \text{then } c_j = g_j(x_n); \quad j = 1, \dots, nc \\ \text{or if } g_j(x_n) \leq 0 & \quad \text{then } c_j = 0; \quad n = 1, \dots, ndv \end{aligned} \quad (3.8)$$

where $g_j(x_n)$ is the normalized form of the constraint, and c_i is the violation coefficient. And the total constraints violations are given by the following equation:

$$C = \sum_{j=1}^{nc} c_j \quad (3.9)$$

The above constraint function can be appended to the structural weight (objective function) as in an exterior penalty function formulation to obtain the modified objective function. The modified objective or penalty function $\phi(x)$ is written, incorporating the constraint violations as [12]:

$$\phi(x) = f(x)(1+KC) \quad (3.10)$$

where, K = A parameter which has to be judiciously selected depending on the required influence of a violated individual. A value of 10 was found suitable for the problems considered in this work [12].

The optimum solution of unconstrained function $\phi(x)$ is obtained by using the genetic algorithm. This is achieved by first generating the initial population and then applying the transformation rules which involve the reproduction, crossover and mutation operators.

3.3.1 Population Generation

GAs perform the search for optimum in parallel using population of points in the search space rather than one single point. This requires the generation of an initial population of individuals (individual means a one set of design variables that comprise a design.) by random selection of individual strings of design variables. In addition and as mentioned in chapter 2, it is fundamental to the GA structure to use an encoding mechanism for the representation of design variables.

In order to transform these design variables into string-like structures for processing by genetic operators, a simple yet effective approach is to represent them by a fixed length binary coded number. In this approach, each design variable is represented by a p -digit binary number, and placed one after another. When n

design variables represented by p strings stacked end to end, they constitute an individual which has pxn length structure.

Consider the truss; shown in Fig. 3.2, with fixed topology and geometry. In this case, the only design variables are the cross sectional designations which are taken from the steel section table. Hence, the design variables will represent 11 section designations corresponding to 11 members.

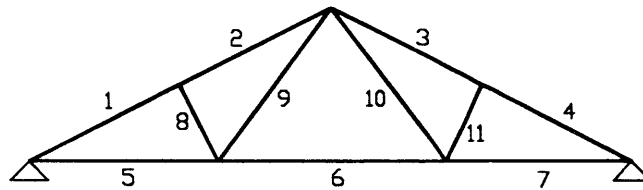


Fig. 3.2 A typical roof truss

From a practical point of view, and particularly for a truss, the members are grouped for economy as well as for esthetic reasons. Let us adopt the following design variable linking for the truss under consideration:

- Upper chord members belong to group 1.
- Lower chord members belong to group 2.
- Diagonal members belong to group 3.

As a result of this design variable linking, there are only three design variables for the optimization problem corresponding to three groups, denoted by DV_1 , DV_2 , and DV_3 .

Consider a binary string of length 5 to represent each of the three design variables. This string length has a capacity of mapping 32 discrete values such that each individual string maps a discrete value of the corresponding design variable. As there are three groups, the total length of the string becomes $3 \times 5 = 15$ with 3 substrings of length 5 as shown in Fig. 3.3.

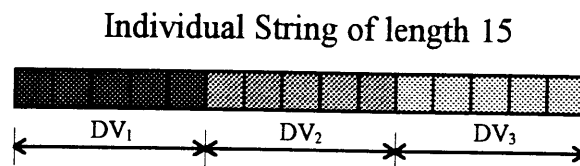


Fig. 3.3 Individual String for first optimization problem

Because the design variables are discrete, it is necessary to supply a list of values that the design variables can take. For the first optimization problem the cross sectional areas as well as other design properties are taken from steel section table for equal angle sections. A portion of the section table; obtained from [38], is reported in Table 5.2 in chapter 5.

Similarly, for the second and the third type of design variables which include geometry and truss spacing optimization, total string length of an individual will be $4 \times 5 = 20$ and $5 \times 5 = 25$ respectively.

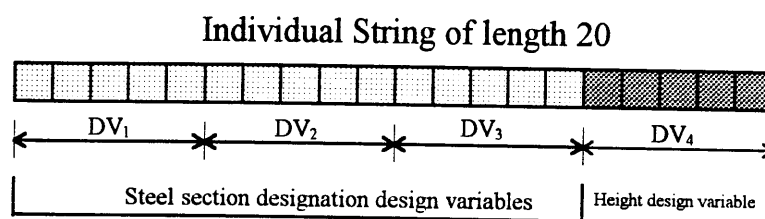


Fig. 3.4 Individual String for second optimization problem

For the second optimization problem, three of the substrings represents the cross sectional areas of member groups and the fourth represents the apex height of the roof truss as shown in Fig. 3.4. Finally, for the third optimization problem an additional string is added to the previous design variables, to represent the truss spacing design variable. The full string length is illustrated in Fig. 3.5.

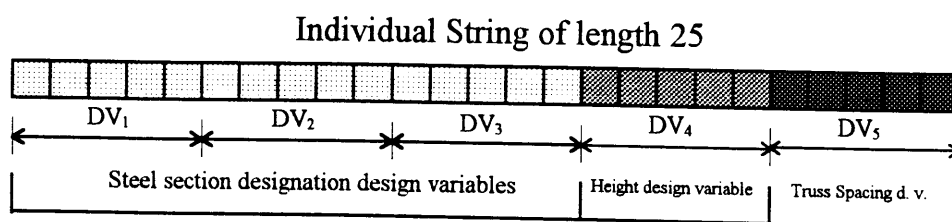


Fig. 3.5 Individual String for third optimization problem

Similar to the first optimization problem a range of height values as well as truss spacing were selected in addition to sections. Each height corresponds to a value which provides practical slope for the rafters of roof truss. In each case the encoding mechanism should map each solution to a unique or a discrete value.

To begin with the process of optimization, it is necessary to generate a population of the above defined strings. Now an important question in GA arises which is *how to choose the population size?* If a small-size population is used in the genetic search, then the search will be considered in a subspace instead of the global space. Thus, the global or near-global optimum may never be obtained. On the other hand, if a large-size population is used in each generation, a great amount of computational time will be required to complete the whole optimization process. Therefore, a judicious tradeoff (moderate population size) is necessary to achieve a near-optimum solution within a reasonable amount of processing time. The adopted population size in this work was taken as 50.

3.3.2 Reproduction Operator

The reproduction operator is the first operator in the genetic search mechanism. It is the operator that is responsible for the exploitation of the information stored in the old population and the transfer of the good characteristics to next generation. It is simply a process to decide which strings or individuals should survive and how many copies of them should be produced in subsequent generations or in the mating pool.

The reproduction operator consists of two processes: duplication and selection of mating pairs. An individual will only be copied or duplicated if it has not violated any of the imposed constraints, i.e. it has a high fitness value as compared to other individuals in a particular generation. Then the selection process selects randomly the fit individuals for mating. This operator requires deciding about an important

criteria, on which the selection will be carried out. This criteria is determined by the setting of and scaling of a fitness function.

3.3.2.1 Fitness Function

The objective function which is now the penalty function $\phi(x)$ given by eqn. (3.10) which is to be optimized. It provides the mechanism for evaluating each string or individual. The penalty function has to be converted into corresponding fitness values, where a careful consideration must be given to the fitness function formulation. The fitness function has to take into consideration all factors that establish the relative merits of a given design. It should be done in such a way that the best individual has maximum fitness. Goldberg suggests that for minimization problems, $\phi(x)$ should be subtracted from a large constant, so that all the fitness values are non negative and individuals get fitness according to their actual merit. In this way, the weight minimization problem is transformed to the fitness maximization problem according to the following equation [12]:

$$F_i = [\phi(x)_{\max} + \phi(x)_{\min}] - \phi_i(x) \quad (3.11)$$

where, F_i = the fitness of the i th individual. Fitness values are computed for all the individuals in a population. Good individuals receive higher fitness values, and poor or bad ones get lesser fitness values.

3.3.2.2 Fitness Function Scaling

There are several scaling procedures proposed in the literature [4,22] such as linear scaling, sigma truncation, power scaling, and proportionate selection scheme. In all above mentioned methods the aim is to decide how many duplications of the fit individuals is to be done as well as how to give higher chances for surviving the good individuals and getting rid from the bad ones.

The deterministic sampling scheme was adopted in this work. In this scheme an individual will receive more than one copy in the mating pool depending upon the

ratio of its fitness to the average population fitness. So the following fitness factor FF_i will be calculated for each individual in the population

$$FF_i = F_i / F_{avg} \quad (3.12)$$

Where, F_i is the raw fitness of individual i and F_{avg} is the average fitness of the current population.

Through fitness factor, the selection and duplication of the fit individuals from the current population are carried out, and they are placed in a mating pool. As the number of the individuals in the next generation is also the same, the individuals with lowest fitness perish or die off. The reproduction operator is implemented in the following manner.

After the evaluation of each individual fitness in a population they are sorted in ascending order. Later on the fitness factors are rounded off to their corresponding integer values. Doing so, the good individuals will have at least a single copy in the mating pool, whereas the bad ones will either have zero copy or sometimes; still, they will have a single copy in the mating pool depending on the overall fitness of a generation. Because the population size is kept the same in subsequent generations, the actual counts of round off values of the fitness factor should be adjusted in such a way that only the fit individuals will have copies and even if actual count of a bad individual come out to be one it will be forced to be zero. Table 3.1 illustrates the idea,

Table 3.1 Two sample problems to illustrate the idea of fitness scaling

	fitness factor	actual count	adjusted count
	1.9	2.0	2.0
	1.6	2.0	2.0
	1.3	1.0	1.0
	0.8	1.0	1.0
	0.6	1.0	0.0
	0.4	0.0	0.0
popsize	6	7	6

(a)

	fitness factor	actual count	adjusted count
	1.7	2.0	2.0
	1.3	1.0	2.0
	0.9	1.0	1.0
	0.6	1.0	1.0
	0.4	0.0	0.0
	0.1	0.0	0.0
6		5	6

(b)

3.3.3 Crossover Operator

Crossover operator is GA's crucial operator. Through this operator the search process in the solution space is implemented. If this operator is not applied in an efficient manner then premature convergence occurs, and GA process ends to an individual which is not really the best among its colleagues. In other words, a given individual will have a high expected number of offspring only if it has a high fitness relative to the rest of the population. This fitness advantage is what causes the subsequent loss of diversity. The loss of diversity is premature only when the fitness advantage is misleading; that is, when the context of the current population makes a string look better than it really is. The fundamental issue, therefore, is the extent to which the population is a representative sample of the best regions in the search space.

The only way to assure that a sample is representative is to make sure that the exploration of the space is as thorough as possible. Pairs of strings are picked at random from the mating pool, which is at this stage full of copies of fit individuals carried from the old generation, to be subjected to crossover. There exist many ways to implement this operator, which were discussed in chapter 2. The multiple crossover was used in this study, in which crossing over was carried out design variable by design variable along the string length. In other words, if an individual consist of 5 design variables then number of cross sites will be five. In this operator the bits are swapped between the mating pairs, and new offsprings are borned. This method was used because in nature there is no such thing that a super individual emerges at the end of a process of generation, but instead a group of good individuals is the aim. By this way of crossing the solution space is fully explored and all the time new combinations will come out , meanwhile the good individuals still control the generation and survive until the end of the genetic process.

3.3.4 Mutation Operator

Mutation is the third operator of a simple GA which plays a secondary role in the genetic operation. Mutation safeguards the process from a complete premature loss of valuable genetic material during the reproduction and crossover. Mutation of a bit involves flipping it: changing a 0 to 1 or vice versa. Mutation is not always effected, the algorithm invokes mutation only if a randomly generated number in the range of 0 to 1 is greater than P_m , mutation rate (Probability of mutation). Otherwise the strings remain unaltered. The value of P_m lies between 0 and 1. Probability of mutation = .001 was adopted in this study [4].

3.4 OPTIMUM DESIGN ALGORITHM

The steps of optimum design procedure by the aid of genetic algorithm are as follows:

1. Input the geometrical properties of the structure, number of design variables involved in the optimization problem, sec.3.3.1.
2. Input the data required for the generation of initial population, as GAs starts with a population of strings which are created at random. Such data include, the population size, length of substring, and a seed value for Fortran Compiler random number generator, sec. 3.3.1.
3. Encode mechanism. Each design variable is expressed as a finite length binary digit substring, and all substrings are connected head-to-tail into one single string which represents a design (individual), i.e. every string contains all design variables.
4. Decode. It involves decoding the strings to obtain the physical values of the design variables.
5. Evaluate each string (individual) in the population by undergoing an analysis and design interfaces. Out of this constraint violations are obtained. sec. 3.3.

6. Apply operators of genetic algorithm.
7. Reproduction Operator, sec. 3.3.2.
8. Calculate the fitness for each string in the population to measure the reproductive efficiency of the living creatures (strings), sec. 3.3.2.1.
9. Scale the fitness by using the deterministic sampling scheme. The rule is: The best get more copies, the average stay even, and the worst die off, sec. 3.3.2.2.
10. Selection. Strings are selected randomly as parents.
11. Crossover Operator. Each pair of parent strings exchange the bits between the randomly generated cross sites along the string length, to reproduce two new strings, sec. 3.3.3.
12. Mutation Operator, alters digits with small probability on a bit-by-bit basis. Input mutation probability to determine whether mutation should be implemented or not, sec. 3.3.4.
14. The generated population undergoes the operations carried out in steps 4 through 12 iteration after iteration.
15. Stop the genetic process after reaching a specified maximum number of generations, and report the final optimum results to an output file.

3.5 ANALYSIS AND DESIGN OF ROOF TRUSSES

The purpose of structural analysis is to determine the distribution of internal forces and displacements in structures. Analysis may be necessary in order to verify the adequacy of a structural design and to assess the behavior of the structure under load or it may be more closely integrated with the design process. It is necessary to

analyze structures under design load cases (discussed in a later section) in order to obtain a general understanding of their behavior in practice.

Equal leg angles were used in the design of members. Double angle sections are adopted for the upper and lower chord members and single angle sections used for the diagonal and vertical bars.

All design requirements considered in this study were those imposed by the BS5950 for the axially loaded compression and tension members of roof trusses.

3.5.1 Load Cases

Design of trusses requires the consideration of different load cases in order to determine the unfavorable loading case for a particular member in a truss. Load cases considered were dead, imposed, and wind loads. Dead load consists of the following possible load types for a roof truss: self weight of roof truss, purlins self weight and roof sheeting weight. It is assumed that purlins support the roof sheets at the joints between the connection of the upper chord members. For the case of the wind load, two wind directions were considered, which are wind blowing at the side of the building and the other one, wind blowing at the frontal side of the building as given in CP3 [36]. Combination of these load cases was carried out to obtain the worst load case to be used for the ultimate limit state design of truss members. In checking the strength and stability of the structure the loads should be multiplied by the relevant γ_f factors given in table(2) in the BS 5950 [37]. The factored loads should be applied in the most unfavorable realistic combination for the part or effect under consideration. The two worst load cases were taken to be

$$\begin{aligned} &1.4 \times \text{Dead Load} + 1.6 \times \text{Imposed Load} \text{ and} \\ &1.0 \times \text{Dead Load} + 1.4 \times \text{Wind Load.} \end{aligned} \quad (3.13)$$

3.5.2 Truss Analysis

Analysis of trusses involves the determination of the bar forces and deflections. The analysis is carried out by using the matrix displacement method, in which the structure stiffness matrix $[K]$ is constructed as well as the load vector $\{R\}$. A brief outline of Matrix Displacement Method (M.D.M.) is given in the following.

1. In the M.D.M. two coordinate systems are introduced in order to describe those quantities associated with the members. The coordinate system used to reference joint quantities is known as *Global Coordinate System*, the other one is known as *Local Coordinate System*, to which end loads and member deformations are referenced to.
2. The structure geometry is defined by inputting first joint coordinates in the global coordinate system as well as the member end connectivities.
3. Joint displacement vector $\{D\}$ is constructed, where each unrestrained joint in a plane truss is allowed to have a horizontal and vertical displacement. Each unrestrained joint is said to have a degree of freedom (d.o.f.) in the unrestrained direction. Accordingly the $\{D\}$ will have a dimension of

$$2 \times (NJ - NRJ) = \text{d.o.f.}, \text{ so } \{D\}_{dof} \quad (3.14)$$

where NRJ = number of restricted joints.

4. The load vector $\{P\}$ is constructed with a dimension of $\{P\}_{dof}$ which has four different values corresponding to four load cases which are $\{P\}^{\text{dead}}$, $\{P\}^{\text{imposed}}$, $\{P\}^{\text{wind1}}$, $\{P\}^{\text{wind2}}$.
5. In the formulation of M.D.M. the following three relationships are used.

I. Relationship between the end forces and end deformations (*Stiffness*):

End forces are related to end deformations by the stiffness matrix $[k_a]$ which is obtained from the linear relationship between the stress and strain. For each member in the structure the relationship in matrix form is given as:

$$\{F_i\} = [k_a]_i \{u\}_i \quad (3.15)$$

where, $\{F_i\}$ = end forces vector

$\{u\}_i$ = end deformations vector

$[k_a]_i$ = member axial stiffness matrix

$$= \frac{EA}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

II. Relationship between Global and Local Displacements (*Compatibility*):

This relationship is to ensure the continuity of the material as well as the elastic behavior of the deformation at the joints, by constructing the displacement transformation matrix $[B]$ for each member, which is given in the following matrix form as:

$$\{u_i\} = [B]_i \{D\}_i \quad (3.16)$$

where, $\{u_i\}$ = vector of member end deformations in local coordinates

$\{D\}_i$ = vector of member end deformations in global coordinates.

$[B]_i$ = Compatibility or Displacement transformation matrix.

$$= \begin{bmatrix} \cos\alpha & \sin\alpha & 0 & 0 \\ 0 & 0 & -\cos\alpha & -\sin\alpha \end{bmatrix}$$

where α is the angle between member axis and global x-axis.

III. Relationship between External loads & Member end forces (*Equilibrium*):

This relationship is obtained by equating the external work done by applied loads, which is in global coordinates to the work done by the internal member forces which is in local coordinates. This equality leads to the following equation:

$$\{P\} = [B]^T \{F\} \quad (3.17)$$

From the above relationships the overall stiffness matrix $[K]$ of the structure is constructed and is given as :

$$\{P\} = [K] \{D\} \quad (3.18)$$

where, $[K] = [B]^T [k_a] [B]$

Note that, the dimension of $[K]$ depends on the total number of degrees of freedom of the structure and $[K]$ is always a square matrix.

3.5.3 Truss Members Design

BS5950 initiates the design procedure by first finding out the classification of the sections, in order to assess the resistance to local buckling of the steel section. If the member to be designed is a tie, then the effective area of section is considered. Where as for the design of struts or compression members the design involves stability checking to ensure the occurrence of no overall buckling.

In order to start the design process, the first decision involves in deciding the type of section which is taken herein as the equal leg angle sections. The following section gives the steps to be followed with reference to the appropriate clauses in BS5950.

3.5.3.1 Design of a Compression Member

- 1) Design process of a compression member starts with section classification, which includes the check for local buckling. This check is carried out by limiting the width to thickness ratios of each element of the section. The ratios b/t and d/t are to be calculated and if they exceeded the limits for semi-compact class then the section should be taken as class 4, slender section. For a slender section, the design strength should be reduced by the appropriate factor taken from Table (8) of the BS5950 [37].

clause 3.5

Table 7
Clause 3.6
- 2) The second step is to calculate the compression resistance of the member, which requires the following information:

- I. Member joint fixity, which decides on member effective length.
- II. Type of section adopted, being either single or double.
- III. For a double section type, it is important to decide on whether the sections are connected back to back to a gusset plate or battened as shown in Fig.3.6 below.

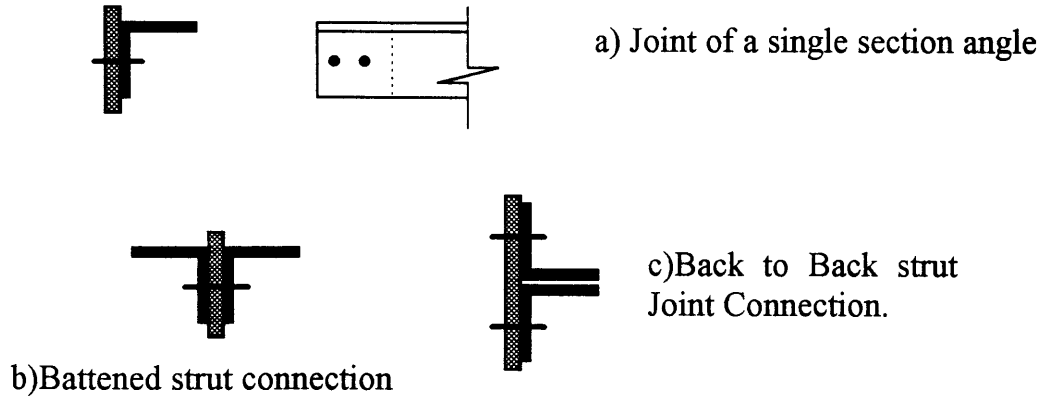


Fig. 3.6 Illustration of joint connection for both single and double angles.

Slenderness ratio:

Clause 4.7.3

The member length should be taken from center to center of intersections with restraining members. And the slenderness ratio is to be taken as given in *clause 4.7.10* for both single or double angle sections.

The code requires that the slenderness ratio of a compression member should not exceed the limit given as

$\lambda_{\max} = 180$ for members resisting loads other than wind loads.

$\lambda_{\max} = 350$ for any member normally acting as a tie but subject to reversal of stress resulting from the action of wind.

Clause 4.7.3.2

Compression resistance:

The compression resistance is the measure of the member resistance to the applied compression load, and is given as

$$P_c = A_g p_c. \quad (3.19)$$

The p_c (Compressive strength) depends on the slenderness λ , of the gross section, the design strength p_y , or the reduced strength of slender sections (cl. 3.6) and the relevant strut curve. In this study p_c is obtained from

Clause 4.7.4

Clause 4.7.5

$$p_c = \frac{P_E \cdot P_y}{\phi + \sqrt{\phi^2 - P_E \cdot P_y}} \quad (3.20)$$

where, P_E is the Euler buckling stress

P_y is the yield strength

and
$$\phi = \frac{P_y + (\eta + 1)P_E}{2}$$

where η is the Perry factor defined in BS5950 [37].

3.5.3.2 Design of a Tension Member

A tension member is designed by evaluating its tension capacity, which is given by:

$$P_t = A_e p_y \quad (3.21)$$

where A_e is the effective area of the section.

Clause 4.6.1

If the joint is welded then $A_e = A_g$ (where A_g is the section gross area). While if the joint is bolted then,

For single angles, A_e is given as:

Clause 4.6.3.1

$$A_e = a_1 + (3a_1a_2) / (3a_1 + a_2) \quad (3.22)$$

where a_1 and a_2 are net cross-sectional areas defined as given in the BS5950 [37].

For double angles, A_e is given as:

Back to back:

Clause 4.6.3.2

$$A_e = a_1 + (5a_1a_2) / (5a_1 + a_2) \quad (3.23)$$

Battened angles:

Clause 3.3.2

$$A_e = (A_g - (DH \times t)) \times 2 \quad (3.24)$$

where, DH is the diameter of the hole and t is the thickness of the connected leg.

3.6 SAMPLE EXAMPLE

The simple truss shown in Fig. 3.7 is selected to demonstrate the application of the design algorithm presented earlier. The dimensions, element grouping and external loading are also shown in the figure. The vertical displacement of joint 2 is restricted to 20mm while the design strength is taken as the steel yield stress of 275N/mm^2 .

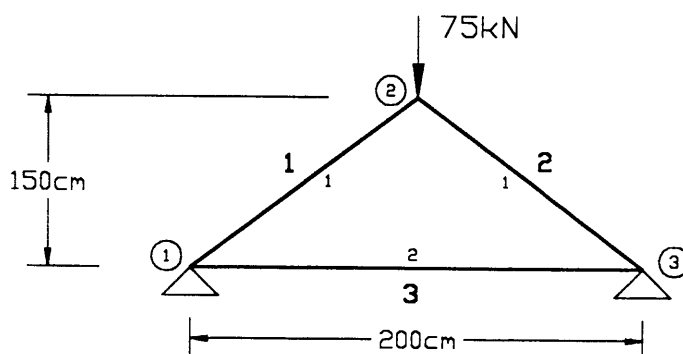


Fig. 3.7 Example of a simple 3-bar truss.

The discrete set from which the truss members sections belonging to two groups are to be selected is given in Table 3.2. Sixteen equal leg angle sections are listed in this table. The other sectional properties such as area, radius of gyration can be

Table 3.2 Equal leg angle sections considered in the design of simple truss

Sequence No.		Designation	Unit Weight (kg/m)
In decimal	In binary		
0	0000	L 45x45x4	2.74
1	0001	L 45x45x5	3.38
2	0010	L 45x45x6	4.00
3	0011	L 50x50x4	3.06
4	0100	L 50x50x5	3.77
5	0101	L 50x50x6	4.47
6	0110	L 50x50x8	5.82
7	0111	L 60x60x5	4.57
8	1000	L 60x60x6	5.42
9	1001	L 60x60x8	7.09
10	1010	L 60x60x10	8.69
11	1011	L 70x70x6	6.38
12	1100	L 70x70x8	8.36
13	1101	L 70x70x10	10.3
14	1110	L 80x80x6	7.34
15	1111	L 80x80x8	9.63

obtained from [37]. In the same table the sequence number of each section is given in decimal as well as binary form. It should be noticed that in order to represent the decimal figure sixteen, the string length in binary representation should be selected as four.

The details of the computations carried out in the genetic algorithm presented are shown in Table 3.3. The size of initial population is selected as six for the purpose of demonstration. The initial population is constructed randomly. The string length of each individual is eight. First four represents the first design variable while the second four belongs to the second variable. By converting these into decimals, one obtains the sequence number of angle section in Table 3.2 which is to be used for the design variable that binary number represents. These are shown in the third and fourth columns of Table 3.3. The truss is then analyzed under the external loads for each individual and joint displacements and capacities for each member are obtained. The maximum values of the displacement and members stresses are shown in columns 5,6 and 7 while the member strengths are given in columns 8 and 9. The unconstrained function of (3.2) is given in column 10. The constraints violation which are calculated by (3.9) are given for each individual in column 11. The modified objective function of (3.10) is given in column 12. From that the fitness is calculated from (3.11) and is located in the 13th column of the same table. The fitness factor is obtained from the fitness by applying equation (3.12) and the respective values are stored in the 14th column. These are sorted and the individual with the highest fitness gives two copies into the mating pool, while that with lowest fitness dies off. These are given by the actual counts stored in the 15th column.

Now it is time to apply the transformation process to produce the new generation. This is achieved by application of the genetic operators, which starts with the reproduction operator. This operator is the one responsible for the survival of the good individuals from the old generation and the extermination of bad ones. It

Table 3.3 Detailed computations in the first generation

GENERATION # 1

Individual Number	Initial Population	Sequence No.		$\delta_{\max}$ (mm)	Members Stresses		Members Strength		Weight (kg)	C	ϕ	Fitness	Fitness Factor	Actual Count	Mating Pool
		Group1	Group2		σ_c (N/mm ²)	σ_t (N/mm ²)	p_c (N/mm ²)	f_t (N/mm ²)							
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)	(15)	(16)
1	10010000	9	0	2.33	-68.97	143.02	49.07	275	46.41	0.8291	431.216	2639.199	1.30	1	10010000
2	11100110	14	6	1.60	-66.59	67.74	75.44	275	59.98	0.0000	59.980	3010.435	1.48	2	11100110
3	01111010	7	10	1.97	-106.17	45.13	49.06	275	57.61	2.3295	1399.656	1670.759	0.82	1	11100110
4	11110110	15	6	1.37	-51.01	67.74	79.83	275	71.43	0.0000	71.430	2998.985	1.47	1	01111010
5	00000101	0	5	2.92	-145.81	87.97	29.57	275	34.78	8.5557	3010.435	59.980	0.03	0	11110110
6	01111001	7	9	2.07	-106.17	55.03	49.06	275	51.21	2.3295	1244.166	1826.25	0.90	1	01111001

Individual No.	Mating Pool	Mate	Cross-sites		New Population
			cs1	cs2	
(17)	(18)	(19)	(20)		
1	10010000	4	2	1	10110010
2	11100110	5	3	1	11110110
3	11100110	6	2	4	01110110
4	01111010	1	2	1	01011000
5	11110110	2	3	1	11100110
6	01111001	3	2	4	01101001

involves the generation of the intermediate population with the selection of mating pairs. The intermediate population is generated from copies of the good individuals of the old population, which are given in the 16th column. The second stage in this operator is the random selection of parents, which are listed in the 17th column.

The second genetic operator involves the exchange of genetic material between the two parents by the random selection of cross sites along the string length and then swapping the bits that come after the cross site. These cross sites are given for each substring of the individual and are listed in columns 18 and 19. Out of this operator and from each couples two offsprings are produced and that will replace these parents in the new generation. The offsprings are given in the 20th column.

The last operator used in this algorithm is the mutation operator which is applied at a very low probability. Through this operator an individual is selected at random and one of its bits which is selected also at random is flipped from 0 to 1 or vice versa. The bold face of bit shown in the 20th column represents the flipped bit.

The generated new population is ready now to undergo a new genetic process in which same computations are repeated. This is shown in Table 3.4 for the second generation. Note the improvement of the quality of the violations of this generation as compared to that of the previous one. It is clear that the current generation has improved and better individuals dominates.

These calculations are repeated from generation to generation. After ceratin number of generations one of the individuals, which is the best, will continue to appear and it becomes impossible to generate another better one. Table 3.5 shows the final results obtained from the 50th generation. The final generation shows the dominance of good individuals where violations were for all equal to zero. It is clear from the results that individual #4 (14, 0) is the best among the candidate individuals of the population. This table shows that according to optimum solution members belonging to group 1 will be made out of the single equal angle section

Table 3.4 Detailed computations in the second generation

GENERATION # 2

Individual Number	Initial Population	Sequence No.		$\delta_{\max}$ (mm)	Members Stresses		Members Strength		Weight (kg)	C	ϕ	Fitness	Fitness Factor	Actual Count	Mating Pool
		Group1	Group2		σ_c (N/mm ²) (6)	σ_t (N/mm ²) (7)	p_c (N/mm ²) (8)	f_t (N/mm ²) (9)							
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)	(15)	(16)
1	10110010	11	2	2.02	-76.18	98.216	64.932	275	47.9	0.3464	213.82	1908.14	1.57	2	10110010
2	11110110	15	6	1.37	-51.01	67.739	79.832	275	71.43	0.0000	71.43	2050.53	1.69	2	10110010
3	01110110	7	6	2.18	-106.17	67.739	49.055	275	46.13	2.3295	1120.75	1001.21	0.82	0	11110110
4	01011000	5	8	2.26	-109.81	72.667	35.658	275	44.03	4.5831	2061.98	59.98	0.05	0	11110110
5	11100110	14	6	1.60	-66.59	67.739	75.441	275	59.98	0.0000	59.98	2061.98	1.70	2	11100110
6	01101001	6	9	1.74	-84.73	55.029	35.328	275	57.46	3.2359	1916.78	205.18	0.17	0	11100110

Individual No.	Mating Pool	Mate (17)	Cross-sites		New Population (20)
			cs1 (18)	cs2 (19)	
1	10110010	2	4	2	10110010
2	10110010	1	4	2	10110010
3	11110110	5	1	3	11100110
4	11110110	6	3	2	11000110
5	11100110	3	1	3	11110110
6	11100110	4	3	2	11110110

Table 3.5 Final computations obtained for generation #50

GENERATION # 50

Individual Number	Sequence No.		$\delta_{\max}$ (mm)	Members Stresses		Members Strength		Weight (kg)	C	ϕ
	Group 1	Group 2		σ_c (N/mm ²)	σ_t (N/mm ²)	p_c (N/mm ²)	f_t (N/mm ²)			
1	14	11	1.54	-66.59	61.07	75.44	275	62.22	0.00	62.22
2	12	7	1.65	-58.58	84.94	64.53	275	60.08	0.00	60.08
3	12	1	1.93	-58.59	116.19	64.53	275	55.32	0.00	55.32
4	14	0	2.30	-66.54	143.02	75.44	275	47.66	0.00	47.66
5	14	0	2.30	-66.54	143.02	75.44	275	47.66	0.00	47.66
6	14	9	1.49	-66.6	55.03	75.44	275	65.06	0.00	65.06

with sequence number 14 which is 80x80x6 and members of group 2 will have the single equal angle section with sequence number zero which is 45x45x4. This gives the minimum weight of 47.66kg. In order to confirm this result, comparison was carried out between the capacities of the smaller sections series provided by the BS5950 and those selected by the algorithm. It was found that the 70x70 series with thicknesses 10mm and 8mm provided the required capacities but were heavier, whereas thickness 6mm of same series was lighter but with insufficient capacity. On the other hand, although the tension strength is far from the tension stress developed in member 3 of the second group, but it was the smallest section provided as a candidate for the algorithm.

4.1 INTRODUCTION

The genetic algorithm based optimum design algorithm described in the previous chapter is coded in Fortran. Three different codes were written for three different type of structural optimization problems. All three algorithms used the same approach of simple genetic algorithm code skeleton of which is given in Figure 4.1.

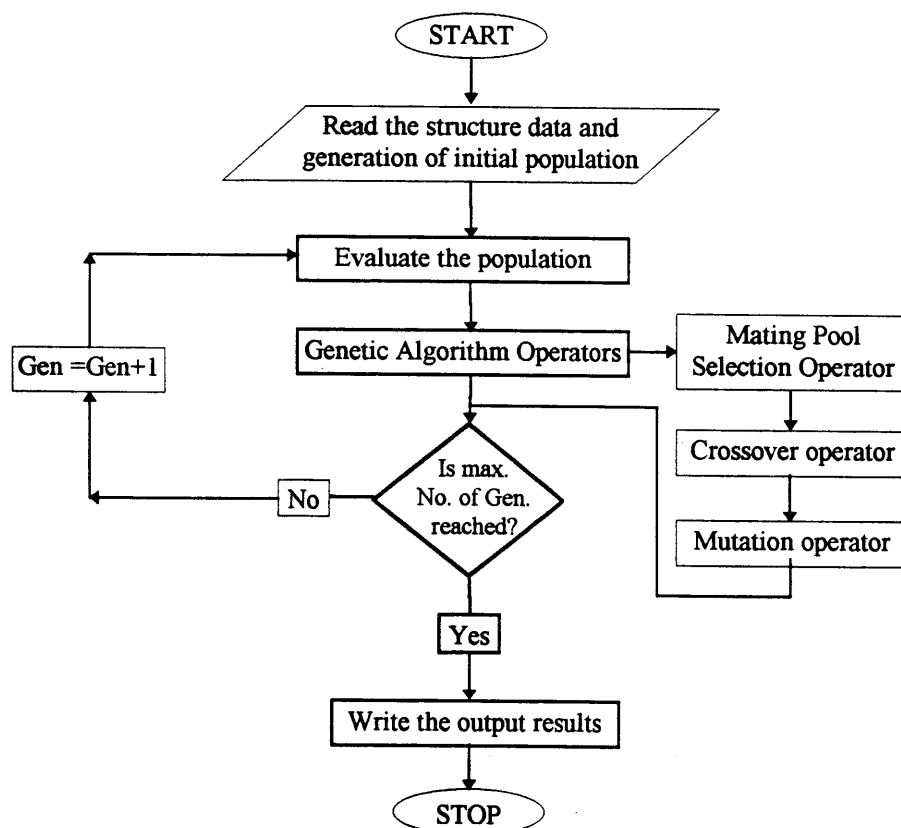


Fig. 4.1 Common structure between the three algorithms.

The three programs developed are given in the following:

1. Program name: **GA1.FOR**

Determines the optimum angle section designations for the members of roof trusses with a given topology, geometry and truss spacing.

2. Program name: **GA2.FOR**

Determines the optimum angle section designations for members and the height for roof trusses with a given truss spacing.

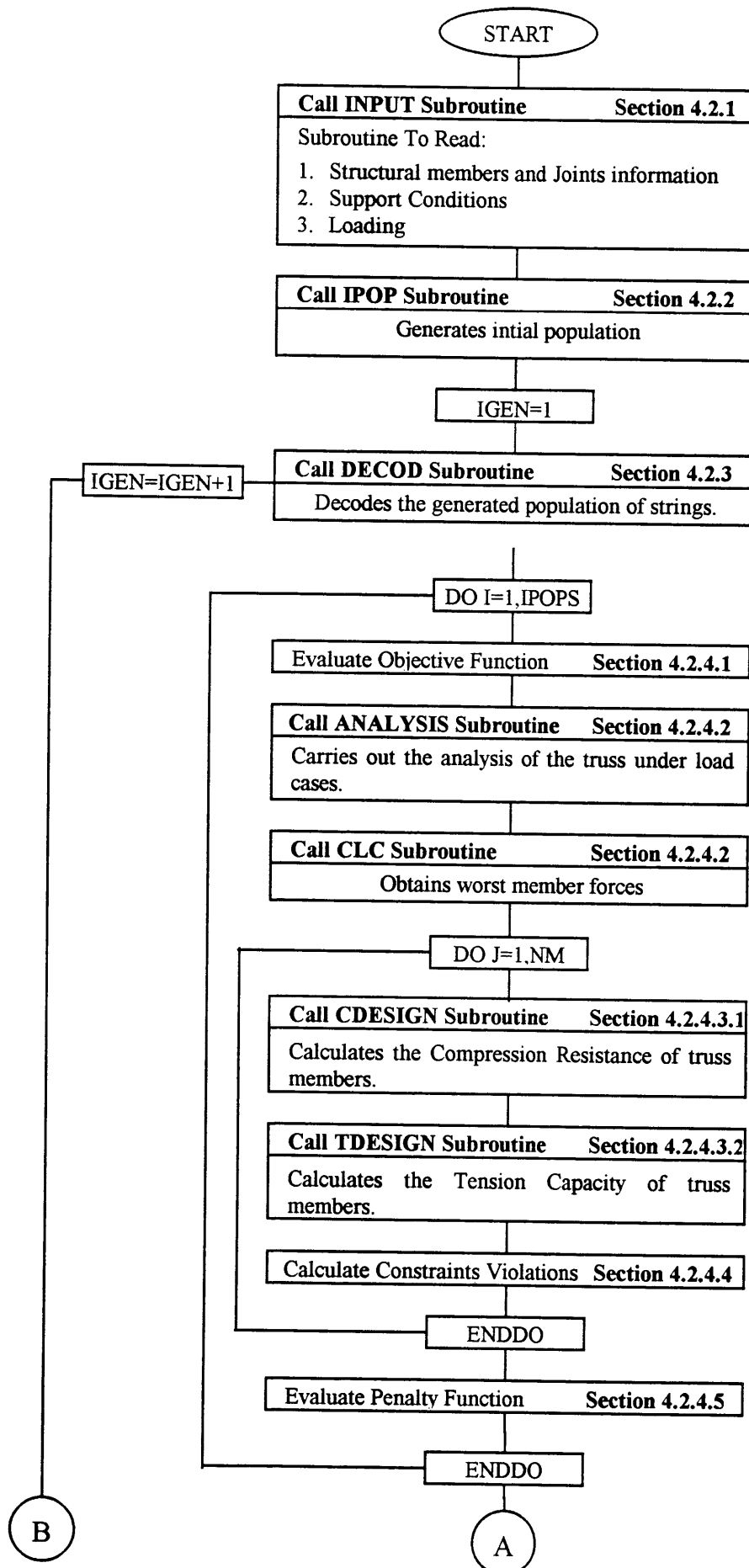
3. Program name: **GA3.FOR**

Determines the optimum angle section designations for members, the optimum height of roof truss apex and the optimum truss spacing for roof trusses with a given topology.

4.2 OPTIMUM DESIGN OF ROOF TRUSS WITH FIXED GEOMETRY

In this algorithm **GA1**, the topology and the geometry of a roof truss are taken to be fixed. The optimization procedure is carried out for different steel section designations for the truss members.

The algorithm consists of a main program which calls nineteen subroutines, each carry out a specific task for the optimum solution of the design problem. First the program calls two subroutines as shown in Fig.4.2 which are necessary for defining the structure and the generation of the initial population to be used in the genetic algorithm. After this information is being supplied to the main program, the evaluation of the individuals of the generated population is carried out by calling the Evaluate subroutine. In this subroutine, the objective function is calculated, the analysis and appropriate design subroutines are called to analyze the structure and to design each structural member. Later, the constraints are calculated as given in sec. 3.4.1. From the constraints the modified objective function is calculated which is used in the genetic process.



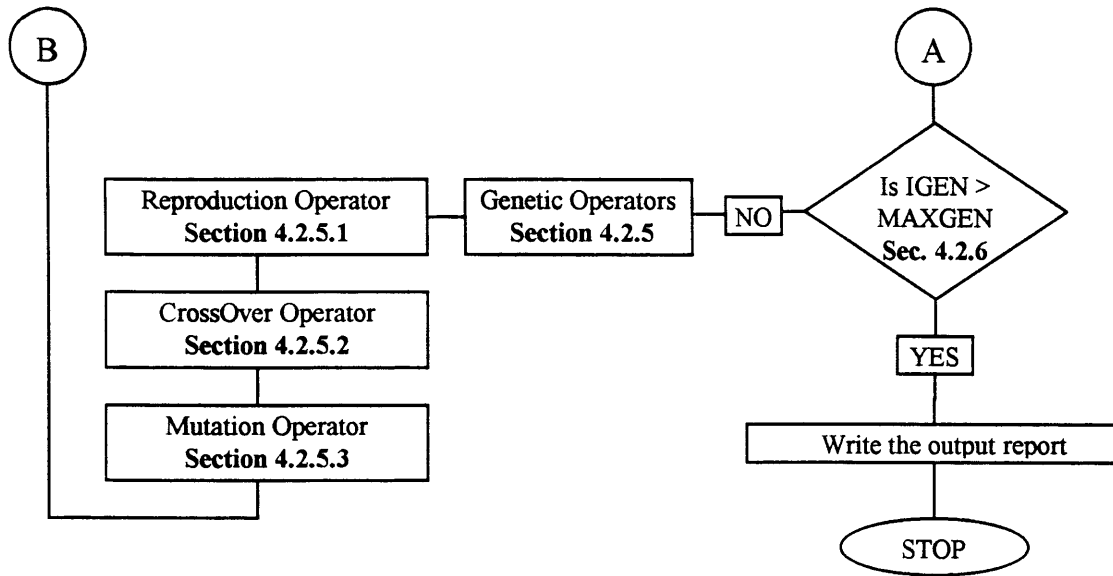


FIG. 4.2 Flowchart of Program GA1.FOR

The genetic process starts by the reproduction operator which is carried out by calling the fitness subroutine. In this subroutine the modified objective function is converted to a fitness function. This function in turn is scaled in order to obtain the actual counts of each individual in the new population. These copies of fit individuals are taken for the second operator which is the cross over operator for mating and exchanging between the bits of the mated individuals. Last operator is the mutation which is only implemented occasionally depending on its probability value. The newly generated population will further be taken for another evaluation iteration and the same is repeated generation by generation until the maximum number of generations is reached, after which the genetic process is stopped. Final results are reported to an output file, and program terminated.

4.2.1 SUBROUTINE INPUT

The INPUT subroutine covers the reading of the required structural information in which the data files required for designing the truss members as well as the definition of the structure are opened for reading. The following are the variables required for defining the structure:

- ◆ Opening the file *SECTAB* for reading the section table, which contains the cross sectional properties of the equal angle sections as given in the BS5950 section tables.
- ◆ Opening the file *INFILE* for reading the structure information, which includes the definition and the construction of the following:

1) General Structural Information:

Number of members	NM
Number of Joints	NJ
Modulus of Elasticity (kN/cm ²)	E
Yield Stress	SIGMAX
Allowable Displacement (cm)	DISPMAX
Roof Truss Span (cm)	SPAN
Roof Truss Apex height (cm)	YAPX

2) Geometrical information

Joint Coordinates x and y (cm)	[COORD] _{NJx2}
Member Connectivity	[NOD] _{2xNM}

3) Members Grouping

The user decides on the number of groups for the truss members as well as the type of the section adopted for one member whether it is a double or a single section. All members belonging to the same group adopt the same cross section designation. For that the following variables are defined:

Number of design variables	NDV
Number of groups	NG
For this type of optimization problem NDV=NG	
The array in which the section types are stored is	{TYPE} _{NDV}
This array contains either the value of 2 or 1 to represent double or single section type, respectively.	
The array in which member groups are stored is	{MGN} _{NM}

4) Support Conditions

- The following is to be read,

Number of restrained joints	NRJ
Restrained joint Number	IRNO
X-direction restrained	IXR
Y-direction restrained	IYR

IXR and IYR = 1 if restrained otherwise = 0
- ID matrix is to be constructed first by initializing its elements i.e. all its elements are equal to zero. [ID]_{2xNJ}
 It has a dimension of 2xNJ, where 2 corresponds to total number of degrees of freedom at each joint of a truss member. For restrained joints, the corresponding column in [ID] will have the value of 1 for the restrained degree of freedom, IXR or IYR. The appropriate element in [ID] is changed to 0 if the corresponding direction of the read joint number (IRNO) is restrained, otherwise a variable NEQ is incremented.
- Calculating the equation number or the total degree of freedom for sizing the stiffness matrix. NEQ
- Reading the restricted joints

Number of restricted joints	NRSJ
Restricted Joint Number	{JRN} _{NRSJ}

5) Loading

The loading is supplied to the program as an area load (kN/cm²) and different types of loading are considered herein. Those are, sheeting and purlin load, truss self weight, and wind load.

Number of Load Types	NLT
Array to store load types values (kN/cm ²)	{ALOAD} _{NLT}
Number of Loading Conditions	NLC
Number of Loaded Joints	NLJ
Loaded Joint Number	LJN
Number of joints on the top chord	NP
Roof truss spacing (cm)	FS
Array to store loaded joint numbers	{LJN} _{NLJ}

6) Data for the type of connection used for member ends

Defining whether the connection is Welded or Bolted	JOINT
1=Weld 0=Bolt	
Diameter of the hole, if bolted connection is used	DH
Thickness of Batten plate	TBP

4.2.2 SUBROUTINE IPOP

This subroutine generates the initial population of individual designs randomly as given in the following

- A seed value for random number generator subroutine is supplied. Depending on the given seed value, the random number generator picks different numbers every time when it is been called.

Seed value for initializing random number generator	ISEED
Length of Substring	LSUB
Population Size	IPOPS
Maximum String Length =NDVxLSUB	MSL
Maximum string length represents an individual in a population. Figure 4.3 shows an example of such an individual.	
Initial population matrix generated	[IPS] _{IPOPSxMSL}

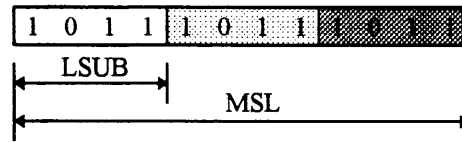


Fig. 4.3 An individual representation in a population

Each bit of the population's individuals is generated as 0 or 1 by calling the **FLIP function**.

FLIP function is a logical function that can have either a value of TRUE (1 bit) or FALSE (0 bit). The idea is exactly of that of tossing a coin and the result be either head or tail. As a result the probability of getting a head or tail is 0.5. The seed value given is taken into the Fortran random number generator to pick up a random number (R). This random number (R) could either be greater than 0.5 or less. If R is greater than 0.5 then FLIP will return FALSE otherwise TRUE.

- The maximum number of generations is given as the stopping criteria
The probability of mutation is a real number given in the range between 0 and 0.001.

MAXGEN

Pm

4.2.3 SUBROUTINE DECOD

The decoding is performed by calling the **DECOD** subroutine. In this subroutine, each individual string in the population is decoded to its representative parameters. And because each individual constitutes more than one design variable along its length, then the decoding of each substring is stored in a column in the $[MC]_{I \times OPS \times NDV}$ matrix which has dimensions of: population size as its maximum row number and number of design variables as the column number.

The binary string population is decoded by applying the following equation

$$\text{The parameter number} = 2^{LSUB}$$

The elements in the [MC] are the numbers representing the corresponding actual values of design variables, whether they are steel section designations, height of the roof truss apex or truss spacing.

4.2.4 SUBROUTINE EVALUATE

The Evaluate subroutine involves in the following computations for each individual string in the population:

1. Calculation of the Objective function:

The objective function which represents the total weight of the roof truss members (kg) is calculated for each individual in the population as given in the following equation

$$\{F\}_{\text{POPS}} = \sum_{j=1}^{\text{NM}} \text{UW} \times \{\text{XLENG}\}_j \times \{\text{TYPE}(\{\text{MGN}\}_j)\}_j \quad (4.1)$$

where, {XLENG} = contains the length of each member

{TYPE} = contains the member group numbers.

UW = the unit length mass (kg/cm) obtained from decoding the individual substring to its actual section designation.

2. Obtaining the truss member forces for each load case by calling the ANALYSIS subroutine. The most unfavorable combination of load cases are considered as the design member forces.

3. Designing of the truss members which involve design of both double and single sections for both compression and tension members. The compression resistance and tension capacity of members are calculated by calling the CDESIGN and TDESIGN subroutines.

4. Evaluation of constraint violations:

The design constraints imposed are evaluated as given in the normalized inequalities in equation 3.7. First of all, the constraint violations are computed

for the member capacities through the **CDESIGN** and **TDESIGN** subroutines calls. Then these constraint violations are added to the violations due to restricted joint displacements if these exist.

5. Calculation of the modified objective function:

The final step in the evaluation process is to compute the modified objective function, in which the constraint violations calculated are substituted in equation 3.10, as given in section 3.3.

4.2.4.1 ANALYSIS Subroutine

The analysis subroutine performs the analysis by applying the matrix displacement method through which the overall stiffness matrix $[K]_{DOF \times DOF}$ and the load vector $[R]_{DOF \times NLC}$ are constructed.

1. Construction of the overall Stiffness matrix $[K]$.

The overall stiffness matrix is assembled by calling **ELEMENT** subroutine. The overall stiffness matrix $[K]$ for structures with a large degrees of freedom has zeros in many locations which makes it sparse. Sparsity should be exploited in order to economize on computer storage space and running time. This is achieved by a scheme called bandedness [39]. In this scheme the nonzero elements are grouped into a relatively narrow band around the main diagonal by proper numbering of the structure degrees of freedom.

The band width MBAND is obtained by first taking each member in turn and adding the degree of freedom of both ends. Then the maximum value of the total number of degrees of freedom at the end joints of any member is assigned to MBAND.

2. Calculation of the load vector $\{RR\}$ for different load cases.

The elements of $\{RR\}$ are calculated by calling the **RESOLVE** subroutine.

3. Solving the stiffness equations for $\{D\}$.

4. Calculating member end forces $\{F\}$.

5. Obtaining design member forces by calling CLC subroutine.
6. Storing displacements due to un-factored imposed load to be used for serviceability check.

4.2.4.1.1 ELEMENT Subroutine

Through this subroutine, the elements in the member stiffness matrix $[k]_{4 \times 4}$ are computed. Later these elements are positioned in their appropriate locations in the overall stiffness matrix depending on the member end joint numbers.

4.2.4.1.2 RESOLVE Subroutine

In this subroutine the area loads are converted to joint loads and resolved into the x and y directions. Each joint load is positioned in the appropriate location in the $\{RR\}$ vector depending on the loaded joint number. The load values for each load case are then stored in $[R]_{NEQ \times NLC}$ where each column corresponds to a load case.

4.2.4.1.3 SOLVER Subroutine

Solution of the stiffness equations for the calculation of joint displacements $\{D\}$ is carried out by this subroutine. The equations $[K]\{D\}=\{R\}$ are solved by the Gauss elimination method. This subroutine is repeatedly called for all the load cases considered. The values of the degrees of freedom for each load case are stored in $[D]_{NEQ \times NLC}$.

4.2.4.1.4 CLC Subroutine

Through the CLC subroutine, the member end forces $[FM]$ obtained for the different load cases are combined to find the unfavorable member forces. This combination was carried out with accordance to equation 3.13 given in section 3.5.1.

4.2.4.2 CDESIGN Subroutine

The compression resistance of truss members is calculated by calling the CDESIGN subroutine. Through this subroutine, the design procedure is carried out

with reference to section 3.5.3. In order to accomplish the compression design process another three subroutines are called which are given in the following subsections.

4.2.4.2.1 CLASS Subroutine

The design process for compression members starts by first calling the CLASS subroutine. The section type at this stage should be known which is taken as the equal leg angles. Through this subroutine, sections which are subjected to compression should be classified. The ratios of section dimensions are calculated and checked with the limiting ratios for semi compact class. If these ratios exceeds the limits specified for semi compact class, then the design strength is reduced according to factors given in Table 8 of the BS5950 [37] and the section is classified as slender.

4.2.4.2.2 CSTR Subroutine

The CSTR subroutine calculates the compressive strength of a compression member. In order to calculate the compressive strength, the effective slenderness ratio is first calculated by calling the XYLAMDA subroutine.

4.2.4.2.2.1 XYLAMDA Subroutine

The effective slenderness ratio for both single and double (battened) equal leg angles is calculated through the XYLAMDA subroutine according to Table 28 in BS5950.

4.2.4.3 TDESIGN Subroutine

The tension capacity P_T of the tension members in the truss is calculated by calling the TDESIGN subroutine. This subroutine in turn calls the TSTR subroutine, in which the effective area is calculated.

4.2.4.3.1 TSTR Subroutine

Through this subroutine the effective area of a section is calculated depending on whether the connection is a welded or a bolted one. If the joints are welded then the effective area equals the gross area, otherwise the effective area is calculated as given in eqns.(3.22) and (3.24) of section 3.5.3.2 for both single and double angle section.

4.2.5 FITNESS Subroutine

The FITNESS subroutine calculates the population statistics and the fitness of each individual in a population. The statistical calculations include the determination of maximum and minimum values of the modified objective function $\phi(x)$ to be used in the evaluation of an individual raw fitness value as given in eqn. (3.11). Then the total and average fitnesses are calculated which is used to decide on the counts of the copies of individuals as given in eqn. (3.12). Then finally, the actual count for each individual in a population is calculated by adopting the fitness scaling mechanism given in section 3.3.2.2.

4.2.6 COPY Subroutine

This subroutine performs the copying of individuals into the mating pool according to their scaled fitness values. Accordingly, the old population is replaced with multiple copies of fit individuals which will replace the bad ones.

4.2.7 RANDMCS Subroutine

The selection of parents and the cross sites for the crossover operator are carried out by calling the RANDMCS subroutine. The first phase of this subroutine carries out the selection of couples by first dividing the population into two halves and then performing the selection randomly by calling the RANDUM subroutine. The selected couples are stored in the first and second columns of $[MATECS]_{(IPOS/2) \times (2+NDV)}$ matrix.

The second phase of this subroutine carries out the random selection of cross sites by calling the **RANDOM** subroutine. The number of cross sites selected depends on how many design variables exist along the string length, as explained in section 3.3.3 in chapter 3. The randomly selected cross sites are stored in columns starting from third column up to the number of design variables (NDV) in the [MATECS] matrix.

4.2.7.1 **RANDOM Subroutine**

The **RANDOM** subroutine generates a random number for all random operations required in the genetic process, such as the random selection of mates, cross sites. And finally the random selection of an individual with a bit string along that individual to be flipped for simulating the mutation operator.

4.3 **OPTIMUM DESIGN OF ROOF TRUSS WITH VARIABLE GEOMETRY**

In the algorithm developed which is called **GA2** the topology is again kept fixed, and the height of roof truss is taken as variable in addition to previous design variables. The figure below illustrates the design variable. Hence, the discrete values supplied to the data file represents the apex height of roof truss.

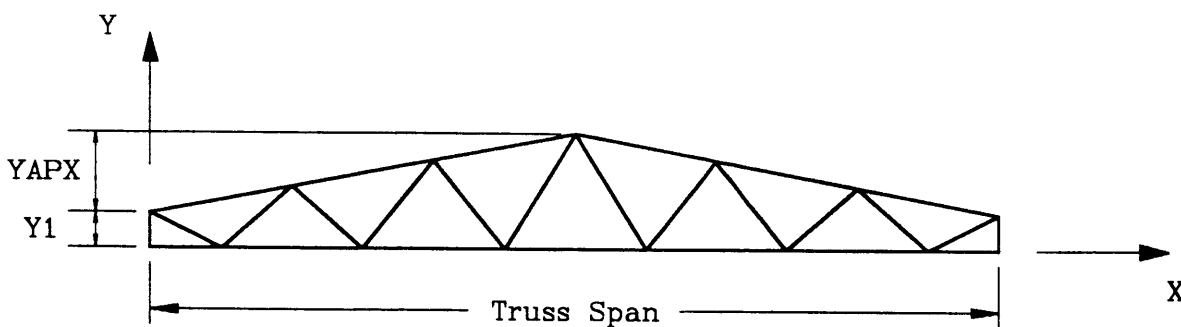


Fig. 4.4 The Geometry design variable YAPX.

The present algorithm consists of same main parts of previous algorithm except with the addition of a new subroutine **YC**. The following points lists the modifications and additions carried out in the previous algorithm in order to include the new design variable,

1. In the **INPUT** subroutine:

- Open a file DATAHT for reading the candidate discrete values for height of roof apex given in cm.
- The y-coordinate of the eave is to be read as Y1.

The y-coordinate values for joints lying on the upper chord are coded with the value of 1. On the other hand, joints lying on the lower chord will have the y-coordinate of 0.

- Number of design variables NDV includes the section designation and the height design variables. And NG which is a variables to define number of groups, includes the grouping of roof truss members chosen for economical and esthetical reasons.

2. In the **EVALUATION** process:

- The process starts by first calling **YC subroutine** which performs the calculation of y-coordinate of the upper chord joints, based on the decoded string value of the substring that represent the height design variable YAPX, which is stored in the last column of [MC]. This is followed by calculation of the lengths of truss members.
- The process continues as given in the previous section, and the output results contains the optimum values for both the section designations and the apex height of the roof truss.

4.4 OPTIMUM DESIGN OF ROOF TRUSS WITH VARIABLE TRUSS SPACING

The algorithm **GA3** is programmed to carry out the optimization of variable truss spacing in addition to previous design variables. The algorithm uses the same flow of program statements as that of **GA2**, except with some modifications and additions to encounter the new design variable. Fig. 4.5 illustrates a plan layout with the truss spacing design variable.

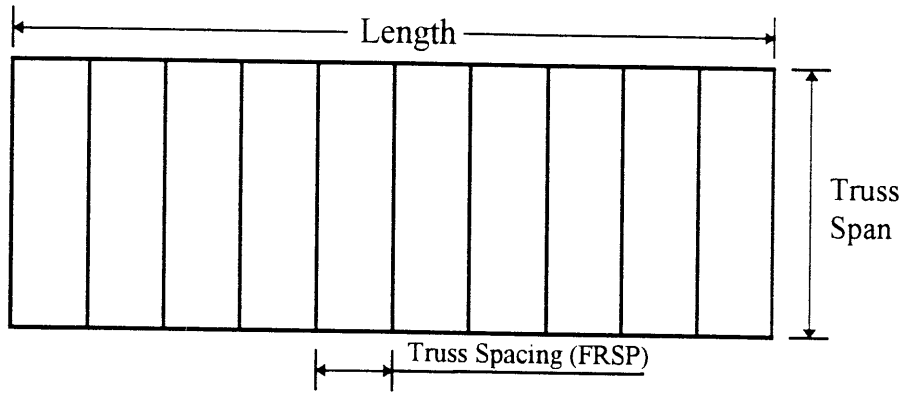


Fig.4.5 A Plan Layout showing the truss spacing design variable

The following is a list of the modifications required for the present algorithm:

1. In **INPUT** subroutine:

- A new data file is opened HT_FS for reading the candidate values for both the height and the truss spacing design variables. Hence, this data file replaces DATAHT.
- Another parameter is required in this algorithm, which is the total length of building (LENGTH) given in cm.

1. In the evaluation process:

- The evaluation process starts by calling the YC subroutine, which is followed by computation of one truss frame weight. This was the objective function in the previous algorithms. Now, the objective function will be that for calculating the whole building weight, and is given in the following equation:

$$\{F\}_{IPOPS} = \left(\frac{LENGTH}{\{FRSP\}_{IPOPS}} \right) \times \sum_{J=1}^{NM} UW \times \{XLENG\}_J \times \left\{ TYPE(\{MGN\}_J) \right\}_J \quad (4.2)$$

where, $\{FRSP\}$ is the array in which the candidate truss spacing values are stored, and the substituted value for the above equation is the decoded mapping number taken from the last column in the [MC] matrix.

- The process continues as in the previous algorithms until the maximum number of generations is reached. The results are written to an output file in which the optimum section designations for members, the optimum height of apex and the optimum spacing of the roof trusses are stored.

Chapter 5

DESIGN EXAMPLES

5.1 INTRODUCTION

The genetic algorithm based optimum design method described in the previous chapters is used to find a roof truss with the minimum weight in a single storey industrial building which is planned to cover a clear area of 50mx16m and clear height of 6m. The front elevation and plan view of the building are shown in Fig. 5.1.

The structural system selected for the building consists of two columns placed at certain intervals along the longer dimension of the area to be covered. These columns are fixed at their base and carry the roof trusses which are simply connected to these elements.

There are various forms of roof trusses used in practice. Among these seven different topologies shown in Fig. 5.2 are selected in this study as a candidate truss for the roof of the industrial building. The design method developed is employed to determine the lightest one among these candidate trusses.

The number of members in these trusses range from 25 to 35, and the number of joints range from 14 to 19. Equal leg double angle section (battened section) was adopted for the upper and lower chord members, while single equal leg angle was adopted for the diagonal and vertical members. The members are assumed to be connected by welding.

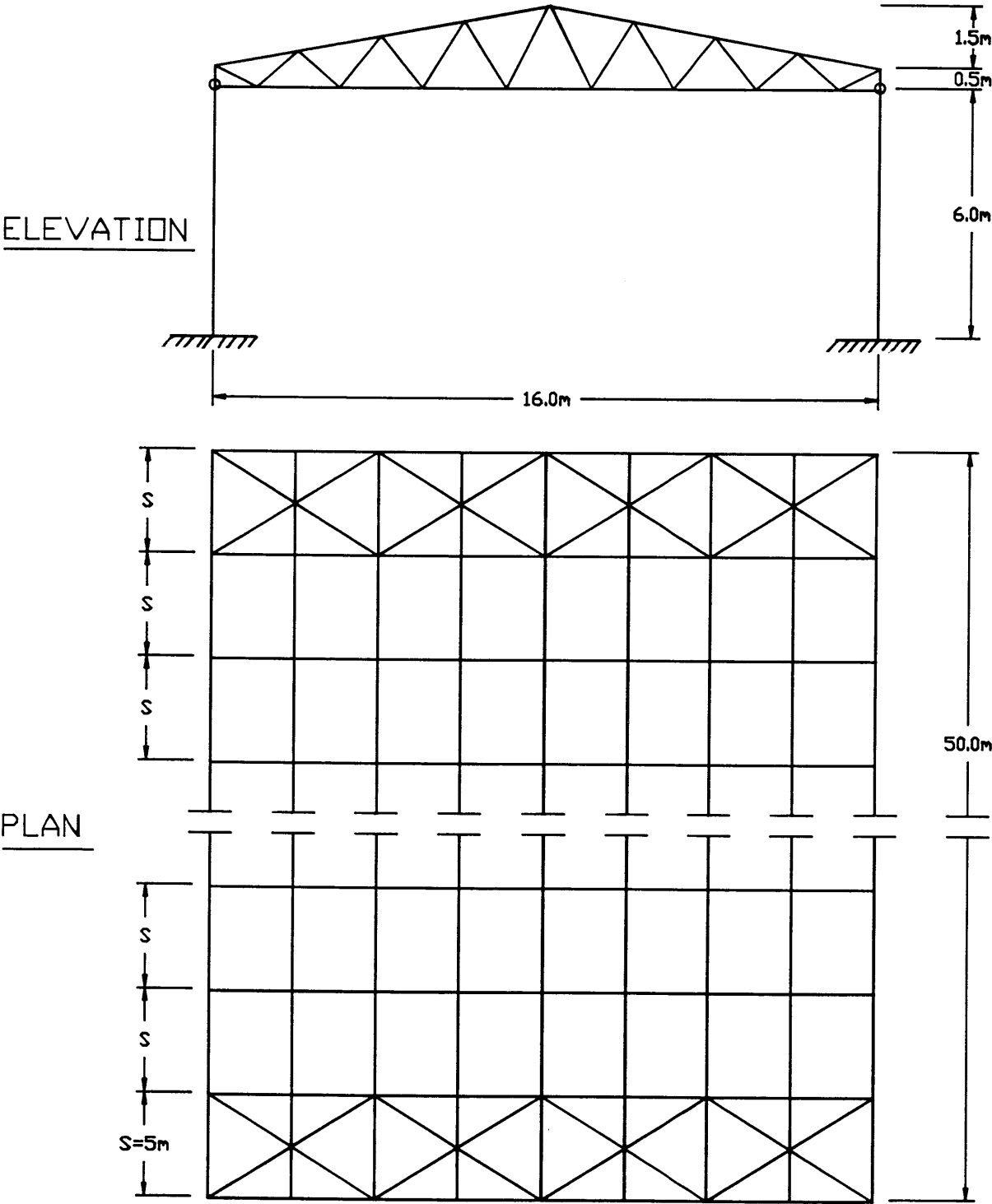
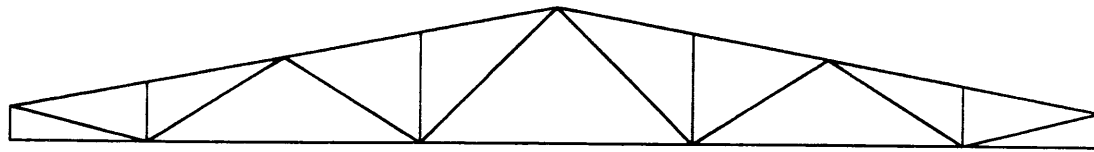
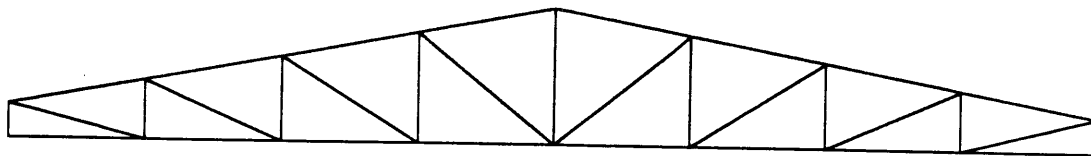


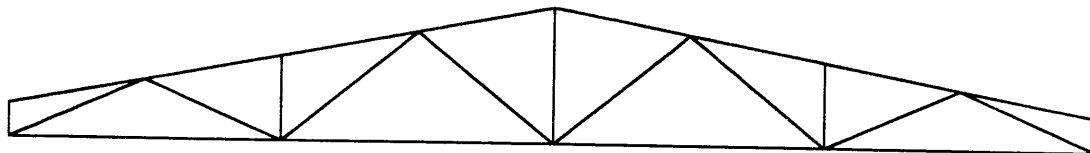
Fig. 5.1 Typical Front Elevation and Plan view of the industrial Building



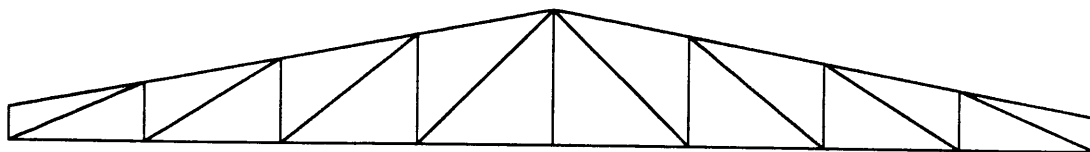
Truss I



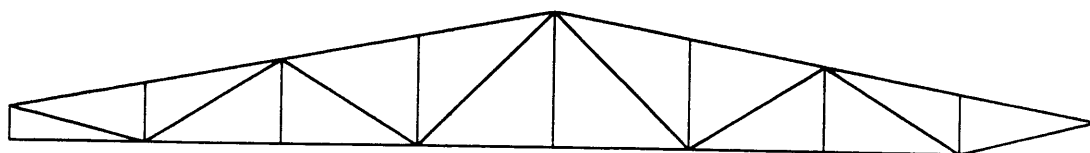
Truss II



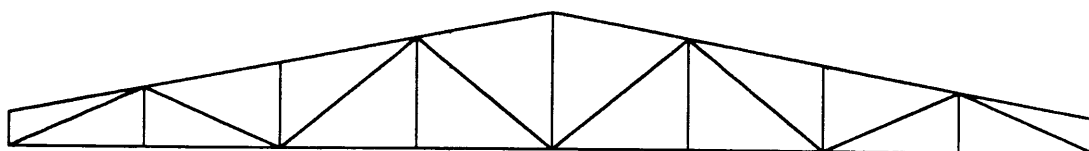
Truss III



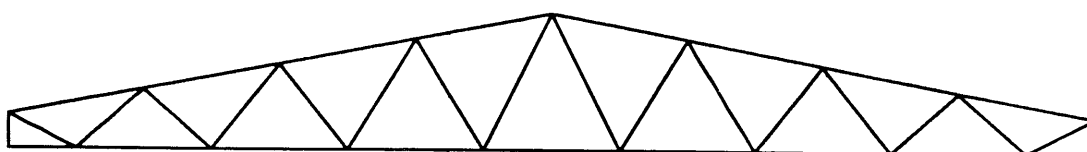
Truss IV



Truss V



Truss VI



Truss VII

Fig. 5.2 The seven Truss topologies

Material Properties:

Steel Grade	= Gr 43
Modulus of Elasticity E_s	= 205 kN/mm ²
Design Strength P_y (for	
Steel section thickness ≤ 16 mm)	= 275 N/mm ²

Loading:

The design of roof trusses necessitates consideration of different load cases. The loading cases which are taken into account in the design of a single storey building are listed in the following table.

Table 5.1 List of Load cases.

Load Case	Load Type		Load Value (kN/m ²)
Dead Load	Truss self weight	Acting on plan	0.10
	Purlins and Sheeting weight	Acting on Slope	0.35
Imposed Load	-	Acting on Slope	0.75
Wind Load	Wind acting on side wall $\alpha=0^\circ$	Left Slope	0.95
		Right Slope	0.41
	Wind acting on front wall $\alpha=90^\circ$	Acting on Slope	0.68

The purlins which supports the sheetings are assumed to be located at the joints of the upper chord. As a result no bending moment develops in the upper chord truss members. Furthermore, lateral supports are provided at the joints of the lower chord members. The wind loading is determined according to CP3 [36]. The truss is analyzed under these load cases and forces in the members are determined. By making use of relevant load factors, these different load cases are combined to obtain the most unfavorable loading conditions. Then members are either designed as compression or tension members depending on the type of force developed in each member.

Deflection:

The mid-span deflections of the trusses are checked against the upper bound of $\text{SPAN}/200$, which is given in BS5950. The code requires that this deflection should be determined due to unfactored imposed loads.

The design examples considered are collected in four main groups. In the first, the geometry of each truss and the spacing between them is fixed. The program GA1.FOR is used to find the optimum angle sections for the members of each truss. In the second, the spacing between trusses is taken as design variable in addition to member sections. The program GA3.FOR is used for finding the optimum truss. In the third, the initial spacing is used for each truss but the height of apex is treated as design variable in addition to member sizes. The program GA2.FOR is employed to obtain the optimum geometry of each truss. In the fourth, the member sizes, geometry and spacing of each truss are considered as design variables and by making use of the program GA3.FOR, the optimum geometry and spacing is determined for each truss. The results obtained in each case are summarized in Table 5.62.

5.2 DESIGN EXAMPLES WITH FIXED GEOMETRY AND SPACING

The program GA1.FOR is used to optimize the seven different roof trusses shown in Fig. 5.2.

The resolved joint loads due to different load cases considered in the design of each truss are shown in Fig. 5.3. It was observed that the unfavorable loading condition to be considered for the design of members were dead+imposed and dead+wind after applying the relevant load factors from the code.

There are four design variables in the optimum design problem with fixed geometry and spacing. These are the angle section for each group in the roof truss.

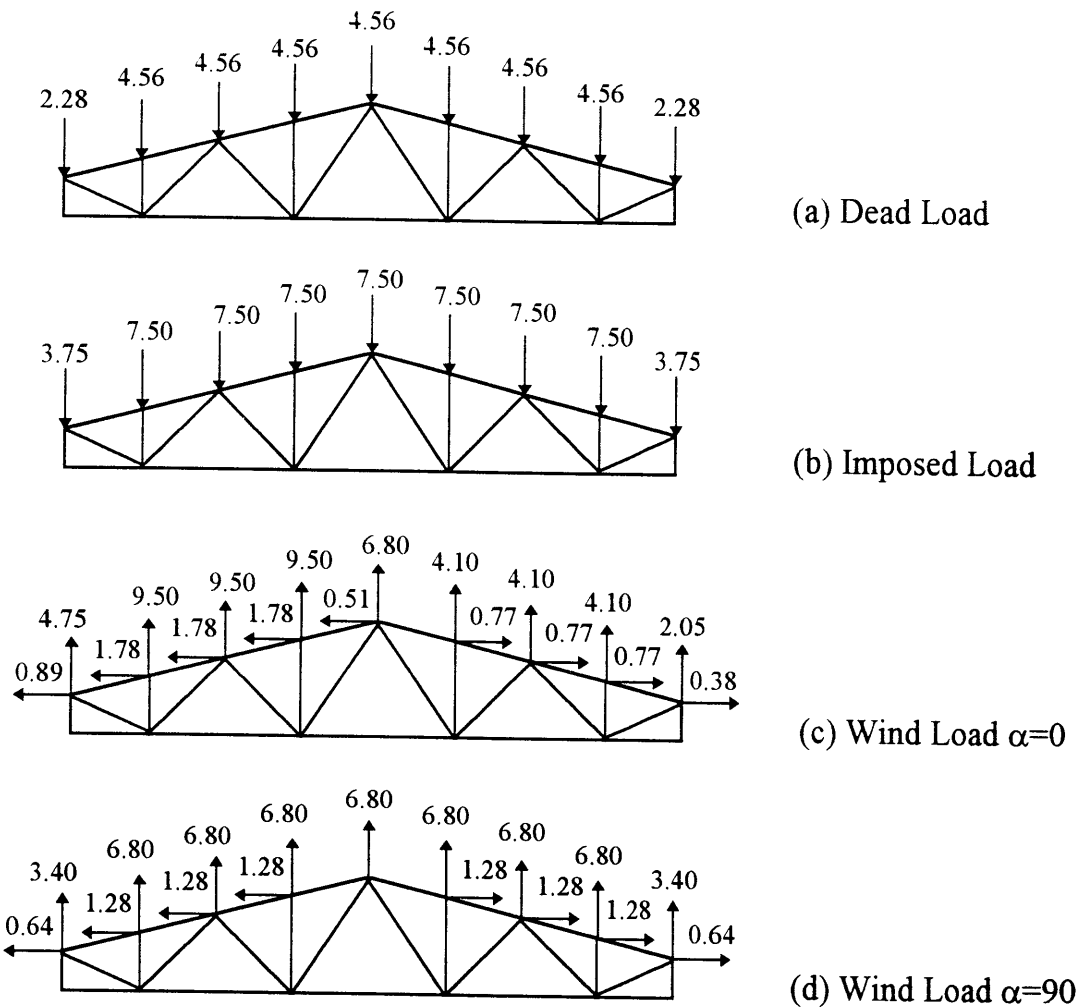


Fig.5.3 Unfactored joint loads for different load cases.

In constructing the initial population for the design, 5-bit string length is employed for coding the four design variables in binary form. The candidate sections for the design variables are the series of equal leg angle sections listed in Table 5.2. The population size is also required to be decided in genetic algorithm. Small population sizes appear to converge well provided that more generations are processed. Hence, population size was taken as 50 in each design, as 450 generations were carried out to ensure the dominance of the best individual. The initial population is created randomly by reading every time different seed values for the random number generator. Each design problem was solved more than once every time initializing the random number generator with different seed values.

Table 5.2 List of sections designation with the required properties

S. No.	Section Designation	Unit Weight (kg/m)	Area (cm ²)	Distance of Center of Gravity C _x & C _y (cm)	Second Moment of Area x-x, y-y (cm ⁴)	Radius of Gyration	
						Axis x-x (cm)	Axis y-y (cm)
1	L 45x45x4	2.74	3.52	1.24	6.58	1.37	0.883
2	L 45x45x5	3.38	4.33	1.29	7.99	1.36	0.877
3	L 45x45x6	4.00	5.12	1.33	9.30	1.35	0.872
4	L 50x50x4	3.06	3.92	1.37	9.16	1.53	0.987
5	L 50x50x5	3.77	4.83	1.41	11.10	1.52	0.979
6	L 50x50x6	4.47	5.72	1.45	13.00	1.51	0.974
7	L 50x50x8	5.82	7.44	1.53	16.50	1.49	0.968
8	L 60x60x5	4.57	5.86	1.65	19.80	1.84	1.180
9	L 60x60x6	5.42	6.95	1.70	23.20	1.83	1.180
10	L 60x60x8	7.09	9.07	1.78	29.60	1.80	1.170
11	L 60x60x10	8.69	11.10	1.85	35.30	1.78	1.160
12	L 70x70x6	6.38	8.19	1.94	37.70	2.15	1.380
13	L 70x70x8	8.36	10.70	2.02	48.30	2.12	1.370
14	L 70x70x10	10.30	13.10	2.10	58.00	2.10	1.360
15	L 80x80x6	7.34	9.36	2.17	56.00	2.45	1.570
16	L 80x80x8	9.63	12.30	2.26	72.40	2.43	1.560
17	L 80x80x10	11.90	15.10	2.34	87.60	2.41	1.550
18	L 90x90x6	8.30	10.60	2.41	81.00	2.76	1.780
19	L 90x90x8	10.90	13.90	2.50	105.00	2.75	1.770
20	L 90x90x10	13.40	17.20	2.58	128.00	2.73	1.760
21	L 90x90x12	15.90	20.30	2.66	149.00	2.70	1.750
22	L 100x100x8	12.20	15.60	2.75	146.00	3.07	1.970
23	L 100x100x10	15.00	19.20	2.83	178.00	3.05	1.960
24	L 100x100x12	17.80	22.80	2.91	208.00	3.02	1.950
25	L 100x100x15	21.90	28.00	3.02	250.00	2.99	1.940
26	L 120x120x8	14.70	18.80	3.24	259.00	3.71	2.380
27	L 120x120x10	18.20	23.30	3.32	316.00	3.69	2.370
28	L 120x120x12	21.60	27.60	3.41	371.00	3.66	2.350
29	L 120x120x15	26.60	34.00	3.52	448.00	3.63	2.340
30	L 150x150x10	23.00	29.50	4.06	635.00	4.64	2.990
31	L 150x150x12	27.30	35.00	4.14	748.00	4.62	2.970
32	L 150x150x15	33.80	43.20	4.26	909.00	4.59	2.950

5.2.1 Truss Topology I

The first topology selected for the roof truss shown in Fig. 5.4 consists of 27 members and 15 joints.

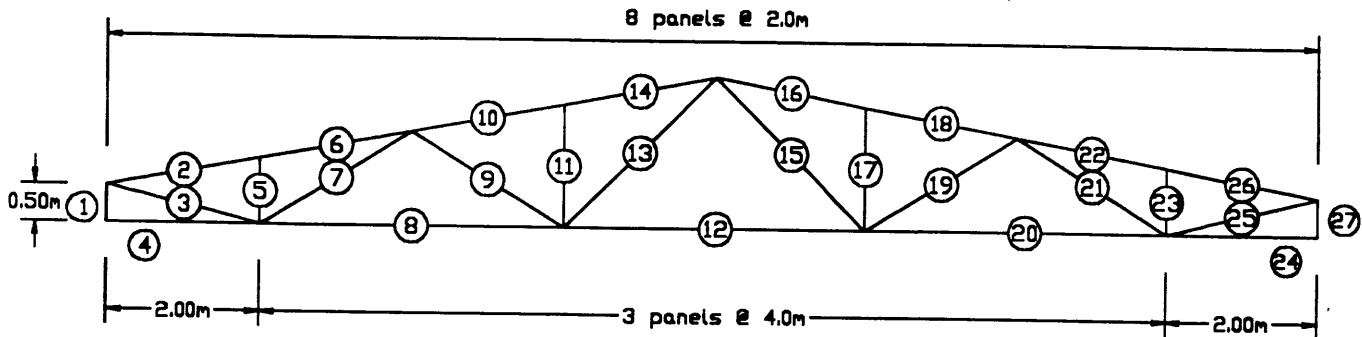


Fig. 5.4 Member numbering for the first topology.

Fig. 5.5 shows the variation of minimum weights during the generations for two different seed values. It is clear that both seed values has the same trend towards improvement of designs in each generation. Although the program was run for 450 generations, only 100 generations are included in the figure due to the fact that from 100 onward, the best individual remains the same.

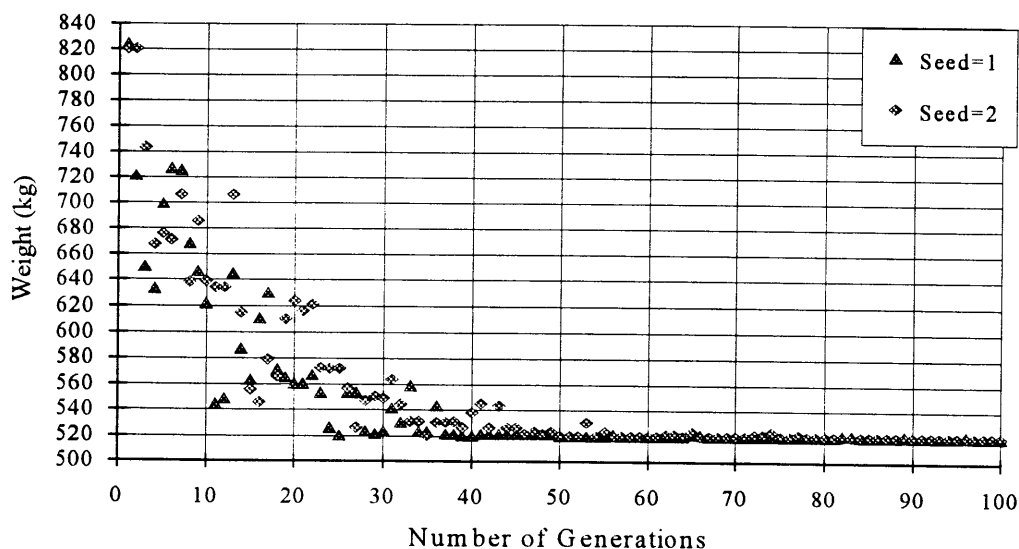


Fig. 5.5 Design history for the first topology with fixed geometry and spacing.

Table 5.3 lists the final values of the design variables for each group with the minimum weight obtained for this truss topology.

Table 5.3 Optimum sections for topology I with fixed geometry and spacing.

Group Number	1 Upper chord members	2 Lower chord members	3 Vertical members	4 Diagonal members	Weight (kg)
Designation	 70x70x6	 60x60x6	 45x45x4	 70x70x6	520.18

Table 5.4 lists the values of the compressive resistance and tension capacity constraints for each group of members.

By considering the first group, which consists of the upper chord members, the compression resistance was the dominant factor which had governed the selection of double angle 70x70x6. The members of this group had all the same length and among them member 14 has the maximum member force as given in Table 5.4. This force was only 80% of the compression resistance of member 14 which seems to be uneconomical selection for the design algorithm. However, when other available sections below 70x70x6 were considered for member 14, it was found that they are either not sufficient or heavier solution. For example, section 60x60x6 has compression capacity of 160kN which makes it inadequate and that sections 60x60x8 and 60x60x10 do have sufficient compression capacity but they provide heavier design for the member. Hence 70x70x6 is the optimum section. On the other hand, and due to second load case, the upper chord had to carry a maximum tension force which is only 12% of the tension capacity of that same section.

The algorithm has selected 60x60x6 for the lower chord members which are basically tension members. The maximum tensile force in these members is 176.41kN which is only 46% of the tension capacity of the selected double angle section 60x60x6. But on the other hand, when the same group was subjected to second load case they act as compression members and the maximum compression force developed was 95% of the compression resistance of selected section for member length of 400cm.

Table 5.4 Member forces and capacities for the optimum design of the first topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I(kN)	D+W(kN)	D+I(kN)	D+W(kN)
Upper Chord Members	2	203.49	96.25	-149.66	53.25	213.30	450.45
	6	"	"	-149.61	55.76	"	"
	10	"	"	-172.60	55.69	"	"
	14	"	"	-172.57	58.22	"	"
	16	"	"	-172.57	51.01	"	"
	18	"	"	-172.60	49.21	"	"
	22	"	"	-149.61	42.67	"	"
	26	"	"	-149.66	40.89	"	"
Lower Chord Members	4	200	0.00	0.00	-5.67	0.00	0.00
	8	400	185.79	176.41	-62.43	382.25	65.77
	12	"	"	147.01	-39.81	"	"
	20	"	"	176.41	-46.51	"	"
	24	200	0.00	0.00	0.00	0.00	0.00
Verticals	1	50.00	55.55	-73.53	26.95	73.86	96.80
	5	87.50	84.37	-18.37	9.20	53.80	"
	11	162.50	156.43	-18.38	9.21	22.40	"
	17	"	"	-18.38	5.29	22.40	"
	23	87.50	84.37	-18.37	5.29	53.79	"
	27	50.00	55.55	-73.53	19.83	73.86	"
Diagonals	3	206.16	126.98	150.79	-52.39	225.23	72.55
	7	235.85	145.27	-34.76	8.57	58.62	225.23
	9	"	"	-7.95	9.72	"	"
	13	282.84	174.21	31.94	-20.29	225.23	43.18
	15	"	"	31.95	-9.78	"	"
	19	235.85	145.27	-7.95	-3.59	58.62	0.00
	21	"	"	-34.76	10.38	"	225.23
	25	206.16	126.98	150.79	-40.27	225.23	72.55

The members in the third group are the vertical members which are all in compression due to D+I and in tension due to D+W. The maximum compression force is 99.6% of the compression resistance of the selected single angle section 45x45x4 for the member number 1 which has the length of 50cm. This indicates that the compression resistance was dominant in the selection of this angle. On the

other hand, the maximum tensile force in this group is only 27% of that section tension capacity.

The diagonal members act in both compression or tension due to one load case. The maximum compression force developed was 72% of the compression resistance of the selected single angle section 70x70x6 for the member number 3 with the length of 206.16cm. Although, this value indicates ineffective selection of sections for the diagonals at the first look, further investigation reveals the opposite. Among the angles designations 60x60, the ones with thickness of 10mm and 8mm provides enough compression resistance! such as 79kN and 65kN respectively for the length of the member. But unfortunately they both are heavier than 70x70x6. The angles with thicknesses 6mm and 5mm within this series provides lighter sections but with insufficient compression resistance such as 50.5kN and 52.4kN respectively. On the other hand, this group has to carry a maximum tension force which is only 67% of the single section angle tension capacity.

As expected the deflection limitation was not dominant in the design, the maximum deflection obtained from the analysis due to unfactored imposed load was 1.1 cm. When this value is compared with the allowable deflection of 8 cm (1600/200), it is only 15% of the allowable.

From all above discussion it was clear that the compression resistance was the dominant factor for the selection of angle section for all the groups of the members of truss topology I.

5.2.2 Truss Topology II

The second topology selected for the roof truss consists of 18 joints 33 members shown in Fig. 5.6.

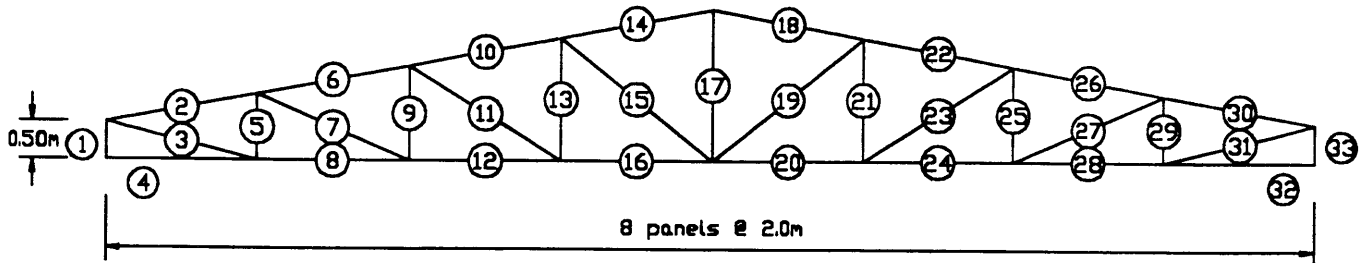


Fig. 5.6 Member numbering for the second topology.

The truss is optimized twice with the developed algorithm. In each design different seed values were used which makes the initial population completely different in each design. Optimum design was obtained from early generations and remained the same until the end of the generations. This is shown in Fig. 5.7 which contains the first hundred generations.

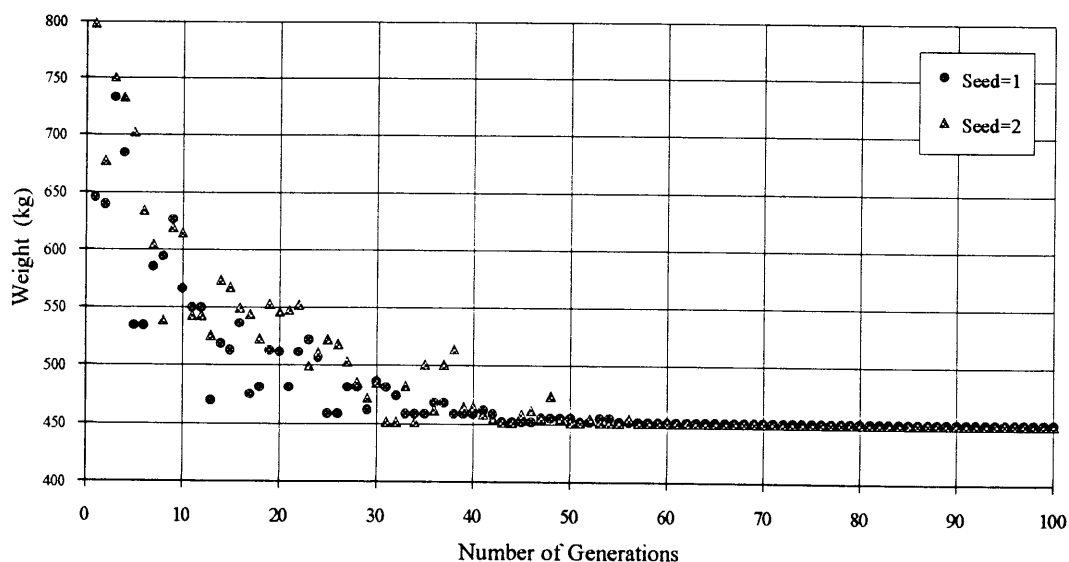


Fig. 5.7 Design history for the second topology with fixed geometry and spacing.

The final values for the design variables with the minimum weight of the second topology are listed in Table 5.5. The sections selected by the program for the different groups of the truss members were same as those of truss I except for the

second group of members for which double angle section 50x50x4 was selected by the program. This leads to a lighter weight for this truss topology.

Table 5.5 Optimum sections for topology II with fixed geometry and spacing.

Group Number	1 Upper chord members	2 Lower chord members	3 Vertical members	4 Diagonal members	Weight (kg)
Designation	 70x70x6	 50x50x4	 45x45x4	 70x70x6	451.54

In this topology, the lower chord members have all same length of 200cm with member 12 carrying the maximum compression force due to D+W load case, as given in Table 5.6. Due to having lower chord members shorter than those in the previous topology, the selection for a smaller section was obvious. By referring to the table of section capacities the series of 60x60 sections provides heavier designs and obviously the 50x50 series with 5mm, 6mm and 8mm thicknesses provide the sufficient capacities but are not lighter in weight as compared with the selected section. Finally, the 45x45 series provide lighter design but with insufficient capacities.

The maximum deflection obtained for this truss topology was 1.57cm in the optimum design which was only 19% of the allowable.

Table 5.6 Member forces and capacities for the optimum design of the second topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I(kN)	D+W(kN)	D+I (kN)	D+W (kN)
Upper Chord Members	2	203.49	96.25	-149.69	53.25	213.30	450.45
	6	"	"	-179.54	61.57	"	"
	10	"	"	-172.60	55.69	"	"
	14	"	"	-149.56	43.97	"	"
	18	"	"	-149.56	44.33	"	"
	22	"	"	-172.60	49.21	"	"
	26	"	"	-179.54	50.06	"	"
	30	"	"	-149.69	40.90	"	"
Lower Chord Members	4	200	0.00	0.00	-5.67	0.00	0.00
	8	"	121.5	146.97	-56.71	215.6	71.82
	12	"	"	176.37	-62.41	"	"
	16	"	"	169.59	-54.16	"	"
	20	"	"	169.59	-43.89	"	"
	24	"	"	176.37	-46.50	"	"
	28	"	"	146.97	-39.26	"	"
	32	"	0.00	0.00	0.00	0.00	0.00
Verticals	1	50.00	55.55	-73.53	26.95	73.86	96.80
	5	87.50	84.37	-36.74	12.76	53.79	"
	9	125.00	120.33	-12.86	3.17	34.05	"
	13	162.50	156.43	4.24	-5.16	96.80	22.40
	17	200.00	192.53	36.69	-11.23	"	15.64
	21	162.50	0.00	4.24	1.90	"	0.00
	25	125.00	120.33	-12.86	3.84	34.05	96.80
	29	87.50	84.37	-36.74	9.81	53.79	"
	33	50.00	55.55	-73.53	19.83	73.86	"
Diagonals	3	206.16	126.98	150.56	-52.31	225.23	72.55
	7	218.30	134.46	32.08	-7.90	"	66.36
	11	235.85	145.27	-7.93	9.72	58.62	225.23
	15	257.69	158.72	-29.14	18.47	50.61	"
	19	"	"	-29.13	8.92	"	"
	23	235.85	145.27	-7.93	-3.60	58.62	0.00
	27	218.30	134.46	32.08	-9.58	225.23	66.36
	31	206.16	126.98	150.56	-40.22	225.23	72.55

5.2.3 Truss Topology III

The third truss topology selected for the roof truss is shown in Fig. 5.8 which consists of 25 members and 14 joints. This truss topology is similar to that of truss I, but with the change in the connection of the diagonals with the verticals, which leads to less number of members.

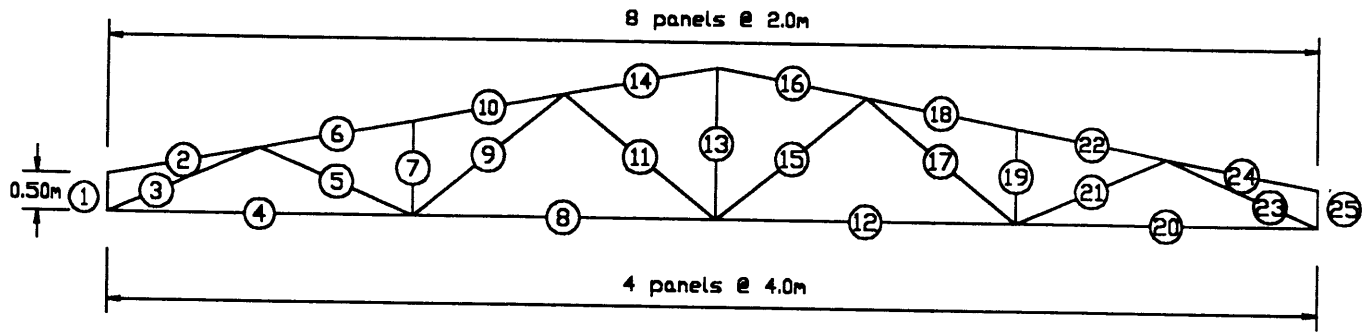


Fig. 5.8 Member numbering for the third topology with fixed geometry and spacing.

For this topology, it became necessary to run the program GA1.FOR four times in order to obtain the same weight at least from two runs. This is achieved at the third and the fourth run. Both final designs converge to the same individual. The history of these designs during the first 100 generations are shown in Fig. 5.9. The same individual has dominated the next 350 generations.

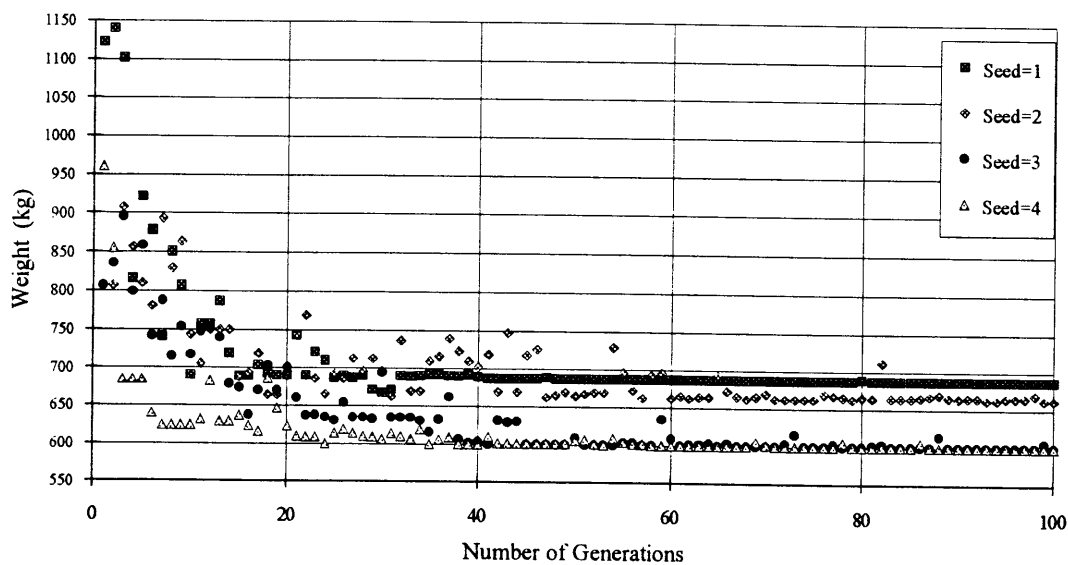


Fig. 5.9 Design history for the third topology with fixed geometry and spacing.

The final values obtained for the truss topology III are shown in Table 5.7. The program has selected the same sections of the first topology for the first three groups. Whereas a larger section was selected for the fourth group of members as compared to that of previous topologies. Consequently this leads to a heavier design.

Table 5.7 Optimum sections for topology III with fixed geometry and spacing.

Group Number	1 Upper chord members	2 Lower chord members	3 Vertical members	4 Diagonal members	Weight (kg)
Designation	 70x70x6	 60x60x6	 45x45x4	 90x90x8	603.76

The selection of the single section angle 90x90x8 for the diagonals, was because of having a maximum compression force of 160.72kN in member 3 of this group. This necessitates a larger compression resistance for the member with a length of 218.3cm. The maximum compression force was due to the D+I load case and it was 98% of the compression resistance of the selected section.

For the lower chord members, the maximum compression force was only 86% of the compression resistance of the selected double angle section 60x60x6 with the member length of 400cm. The double angle section 60x60x5 has the compression resistance of 55.3kN for the effective length of 400cm which is less than the required resistance. Hence, the algorithm has selected the most suitable section for the member.

The maximum deflection for this truss topology was 1.15 cm which is only 14% of the allowable deflection.

Table 5.8 Member forces and capacities for the optimum design of the third topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to D+I (kN) D+W (kN)		Member Capacities due to D+I (kN) D+W (kN)	
Upper Chord Members	2	203.49	96.25	-0.10	1.30	213.30	450.45
	6	"	"	-179.53	61.57	"	"
	10	"	"	-179.50	64.09	"	"
	14	"	"	-149.56	43.97	"	"
	16	"	"	-149.56	44.33	"	"
	18	"	"	-179.50	51.86	"	"
	22	"	"	-179.53	50.06	"	"
	24	"	"	-0.10	0.93	"	"
Lower Chord Members	4	400	185.79	147.01	-56.73	382.25	65.77
	8	"	"	169.62	-54.17	"	"
	12	"	"	"	-43.90	"	"
	20	"	"	147.01	-39.27	"	"
Verticals	1	50	55.55	-9.19	4.60	73.86	96.80
	7	125	120.33	-18.38	9.20	34.05	"
	13	200	192.53	36.69	-11.23	96.80	15.64
	19	125	120.33	-18.38	5.29	34.05	96.80
	25	50	55.55	-9.19	2.65	73.86	"
Diagonals	3	218.3	104.83	-160.72	55.82	163.58	382.25
	5	"	"	32.24	-7.95	382.25	163.58
	9	257.69	123.75	8.80	-10.65	"	128.88
	11	"	"	-29.16	18.51	128.88	382.25
	15	"	"	-29.13	8.93	"	"
	17	"	0.00	8.79	3.89	382.25	0.00
	21	218.3	104.83	32.24	-9.63	"	163.58
	23	"	"	-160.72	42.93	163.58	382.25

5.2.4 Truss Topology IV

The fourth topology selected for the roof truss is shown in Fig. 5.10, consists of 33 members and 18 joints. It is similar to topology II, except that the diagonals are reversed.

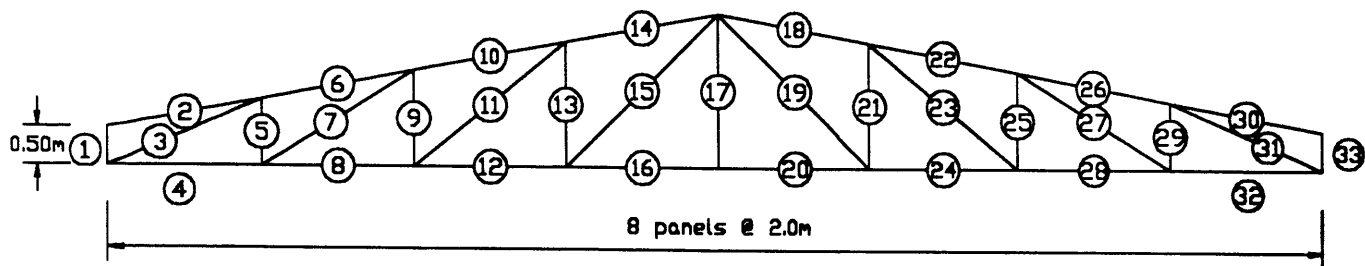


Fig.5.10 Member numbering for the fourth topology.

In this design, the program was run 4 times, each time with a different initial population obtained from using different seed values. The initial population constructed from seed value 1 obtained the minimum weight of 648kg, and seed value 2 yield to the minimum weight of 580kg. Whereas for the seed values 3 and 4, the least weight of the truss come out to be 554.61kg. The variation of the truss weight during the generations for each seed value is shown in Fig. 5.11.

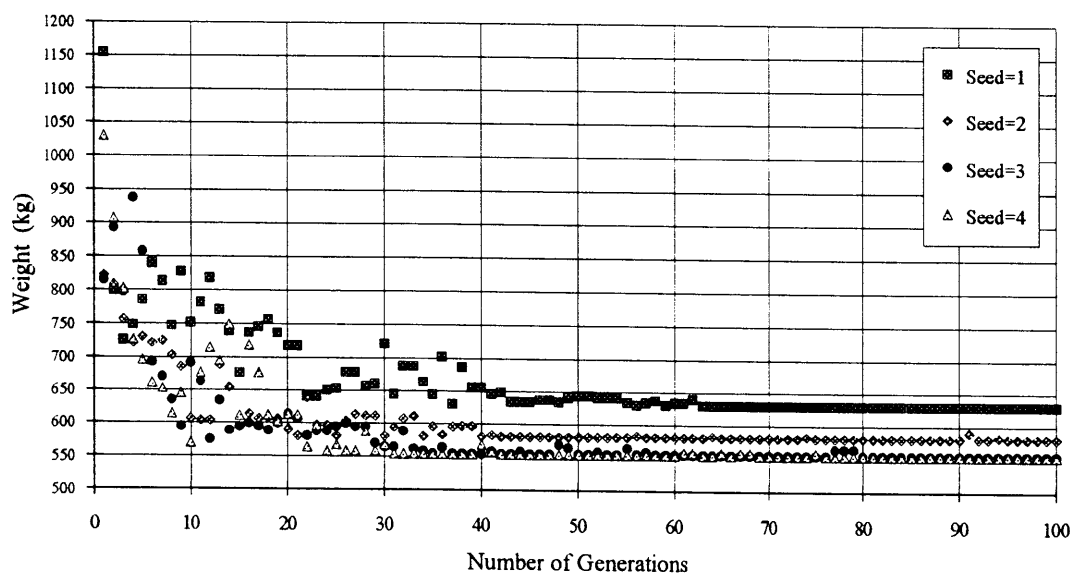


Fig.5.11 Design history for the fourth topology with fixed geometry and spacing.

The optimum sections obtained for each group and the minimum weight of the truss is given in table 5.9. The sections selected by the algorithm yield a lighter design as compared with the previous topology. It has the same sections selected for the first and fourth groups. On the other hand, the design obtained here was heavier than the first two topologies.

Table 5.9 Optimum sections for topology IV with fixed geometry and spacing.

Group Number	1 Upper chord members	2 Lower chord members	3 Vertical members	4 Diagonal members	Weight (kg)
Designation	 70x70x6	 50x50x4	 50x50x4	 90x90x8	554.61

The single angle section 50x50x4 was selected for the third group to resist the maximum compression force developed in member 13 which has a length of 162.5cm, as given in Table 5.10. The series of 45x45 angle sections with thicknesses 6mm and 5mm provides heavier designs, meanwhile the section with thickness 4mm of this series provides a resistance of 22.4 kN for that same length of the member which is not sufficient to resist the maximum compression force. For the diagonals the same compression force was developed as compared with the previous topology. For which the same section was adopted.

The maximum deflection obtained for this truss topology was 1.59cm, which is only 20% of the allowable.

Table 5.10 Member forces and capacities for the optimum design of the fourth topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I (kN)	D+W (kN)	D+I (kN)	D+W (kN)
Upper Chord Members	2	203.49	96.25	-0.13	1.31	213.30	450.45
	6	"	"	-149.64	55.77	"	"
	10	"	"	-179.51	64.09	"	"
	14	"	"	-172.58	58.22	"	"
	18	"	"	-172.58	51.02	"	"
	22	"	"	-179.51	51.87	"	"
	26	"	"	-149.64	42.69	"	"
	30	"	"	-0.13	0.94	"	"
Lower Chord Members	4	200.00	121.50	147.00	-56.71	215.6	71.82
	8	"	"	176.37	-62.41	"	"
	12	"	"	169.59	-54.16	"	"
	16	"	"	146.98	-39.80	"	"
	20	"	"	"	"	"	"
	24	"	"	169.59	-43.89	"	"
	28	"	"	176.37	-46.50	"	"
	32	"	"	146.97	-39.26	"	"
Verticals	1	50.00	52.88	-9.19	4.60	77.21	107.80
	5	87.50	77.06	18.38	-4.53	107.8	61.04
	9	125.00	107.65	-5.51	6.70	42.45	107.80
	13	162.50	139.94	-22.61	14.35	29.01	107.80
	17	200.00	0.00	0.00	0.00	0.00	0.00
	21	162.50	139.94	-22.60	6.92	29.01	107.80
	25	125.00	107.65	-5.51	-2.46	42.45	0.00
	29	87.50	77.06	18.38	-5.49	107.80	61.04
	33	50.00	52.88	-9.19	2.65	77.21	107.80
Diagonals	3	218.3	104.83	-160.81	55.85	163.58	382.25
	7	235.85	113.26	-34.67	8.54	146.90	"
	11	257.69	123.75	8.82	-10.66	382.25	128.88
	15	282.84	135.83	31.99	-20.31	"	111.48
	19	"	"	"	-9.79	"	"
	23	257.69	0	8.82	3.88	"	0.00
	27	235.85	113.26	-34.67	10.36	146.90	382.25
	31	218.30	104.83	-160.81	42.96	163.58	"

5.2.5 Truss Topology V

The truss topology shown in Fig. 5.12 is similar one to truss II with different arrangement of diagonals. The total number of members in this truss is 33 and total number of joints is 18. This topology is selected as the fifth candidate for the roof truss.

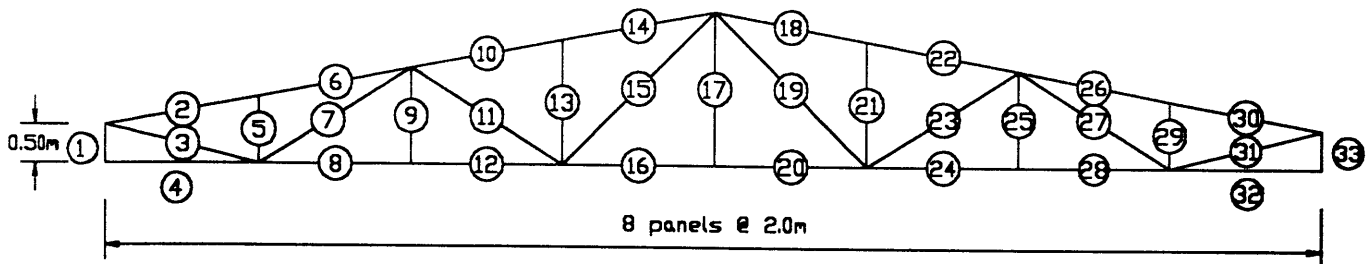


Fig. 5.12 Member numbering for the fifth topology.

The truss was designed twice starting from two different initial populations. The same minimum was obtained from two independent seed values of 1 and 2. The history of the design is shown in Fig. 5.13.

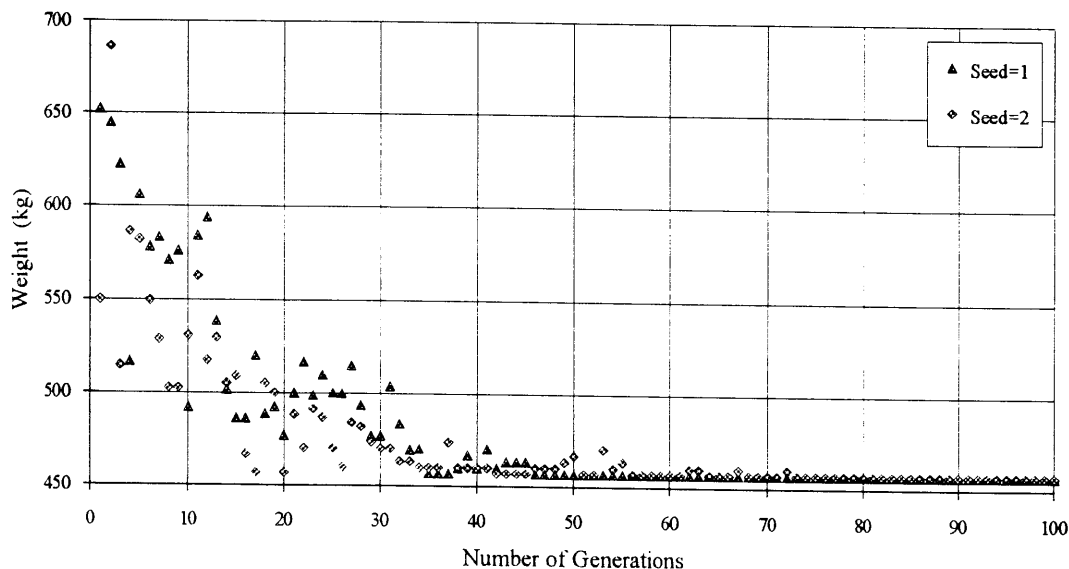


Fig. 5.13 Design history for the fifth topology with fixed geometry and spacing.

The optimum angle sections arrived at the final design are given in Fig. 5.11 with the minimum weight for this truss topology. The sections selected were exactly same as those for truss II, but the weight in this truss was slightly heavier.

Table 5.11 Optimum sections for topology V with fixed geometry and spacing.

Group Number	1 Upper chord members	2 Lower chord members	3 Vertical members	4 Diagonal members	Weight (kg)
Designation	 70x70x6	 50x50x4	 45x45x4	 70x70x6	456.99

The member forces and other properties of the optimum truss is given in Table 5.12. The increase in the weight was due to longer diagonal members in this topology. Further more the existence of three zero force members in the truss has added to the weight which become heavier than topology II. The maximum deflection was 1.6 cm, which is only 20% of the allowable.

Table 5.12 Member forces and capacities for the optimum design of the fifth topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to D+I (kN) D+W (kN)		Member Capacities due to D+I (kN) D+W (kN)	
Upper Chord Members	2	203.49	96.25	-149.69	53.26	213.30	450.45
	6	"	"	-149.64	55.77	"	"
	10	"	"	-172.60	55.69	"	"
	14	"	"	-172.57	58.22	"	"
	18	"	"	-172.57	51.02	"	"
	22	"	"	-172.60	49.21	"	"
	26	"	"	-149.64	42.70	"	"
	30	"	"	-149.69	40.89	"	"
Lower Chord Members	4	200	0.00	0.00	-5.67	0.00	0.00
	8	"	121.50	176.37	-62.41	215.6	71.82
	12	"	"	"	"	"	"
	16	"	"	146.97	-39.80	"	"
	20	"	"	"	"	"	"
	24	"	"	176.37	-46.50	"	"
	28	"	"	"	"	"	"
	32	"	0.00	0.00	0.00	0.00	0.00
Verticals	1	50.00	55.55	-73.53	26.5	73.86	96.80
	5	87.50	77.06	-18.37	9.20	53.79	"
	9	125.00	107.65	0.00	0.00	0.00	0.00
	13	162.50	139.94	-18.38	9.21	22.40	96.80
	17	200.00	0.00	0.00	0.00	0.00	0.00
	21	162.50	139.94	-18.38	5.29	22.40	96.80
	25	125.00	107.65	0.00	0.00	0.00	0.00
	29	87.50	77.06	-18.37	5.29	53.79	96.80
	33	50.00	52.88	-73.53	19.83	73.86	"
Diagonals	3	206.16	126.98	150.60	-52.33	225.23	72.55
	7	235.85	145.27	-34.81	8.58	58.62	225.23
	11	"	"	-7.94	9.72	58.62	225.23
	15	282.84	174.21	31.94	-20.29	225.23	43.18
	19	"	"	31.95	-9.78	"	"
	23	235.85	145.27	-7.94	-3.59	58.62	0.00
	27	"	"	-34.81	10.39	"	225.23
	31	206.16	126.98	150.60	-40.22	225.23	72.55

5.2.6 Truss Topology VI

The truss topology given in Fig. 5.14 is similar to truss IV with different arrangement of the diagonals. This truss topology consists of 33 members and 18 joints.

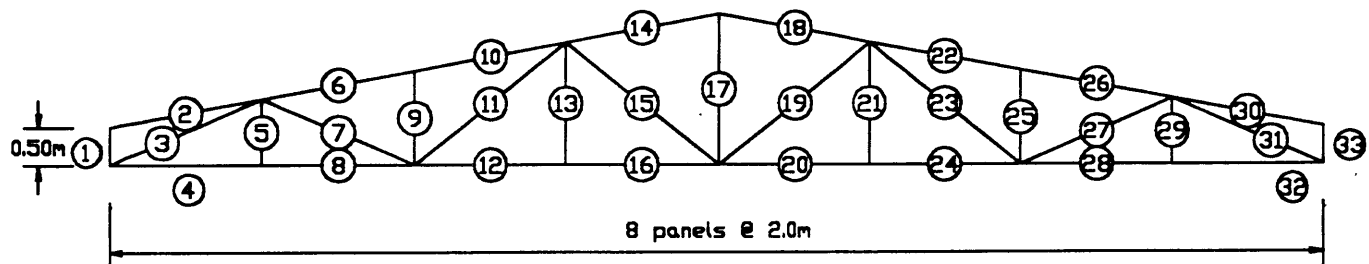


Fig. 5.14 Member numbering for the sixth topology.

Similar to the third topology, it was necessary to run the program four times for this topology as well. The history of these designs during the generations are shown in Fig. 5.15. Three of the runs represented by three different seed values obtained close results while the first seed value was far from them.

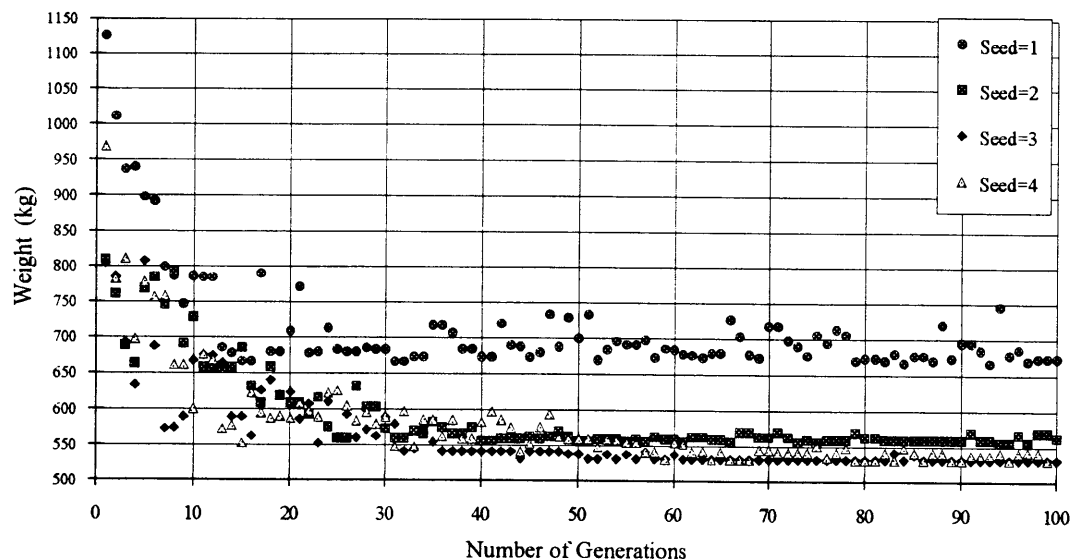


Fig. 5.15 Design history for the sixth topology with fixed geometry and spacing.

Table 5.13 lists the values of the final design variables with the minimum weight of this truss topology. The sections selected here were same as those for truss IV except for groups 2 and 3 for which the smallest section was selected. Consequently this design was lighter compared to that of truss IV.

Table 5.13 Optimum sections for topology VI with fixed geometry and spacing.

Group Number	1 Upper chord members	2 Lower chord members	3 Vertical members	4 Diagonal members	Weight (kg)
Designation	 70x70x6	 45x45x4	 45x45x4	 90x90x8	531.70

The reason for selecting a smaller size for second and third groups in this truss was due to the following:

The maximum compression force 56.71kN developed in the lower chord members which was carried by member 4. For that the double section angle 45x45x4 was selected providing required resistance of 58.94kN. The third group was subjected to a maximum compression force carried by member 9 with an effective length of 125cm. The single section angle 45x45x4 has the required resistance to that compression force for this member. The maximum deflection was 1.69cm, which was only 21% of the allowable deflection.

Table 5.14 Member forces and capacities for the optimum design of the sixth topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I (kN)	D+W (kN)	D+I (kN)	D+W (kN)
Upper Chord Members	2	203.49	96.25	-0.14	1.32	213.30	450.45
	6	"	"	-179.54	61.57	"	"
	10	"	"	-179.51	64.09	"	"
	14	"	"	-149.56	43.97	"	"
	18	"	"	-149.56	44.34	"	"
	22	"	"	-179.51	51.87	"	"
	26	"	"	-179.54	50.06	"	"
	30	"	"	-0.14	0.95	"	"
Lower Chord Members	4	200	132.19	146.96	-56.71	193.60	58.94
	8	"	"	"	"	"	"
	12	"	"	169.57	-54.15	"	"
	16	"	"	"	"	"	"
	20	"	"	"	-43.89	"	"
	24	"	"	"	"	"	"
	28	"	"	146.96	-39.26	"	"
	32	"	"	"	"	"	"
Verticals	1	50.00	55.55	-9.19	4.60	73.86	96.80
	5	87.50	0.00	0.00	0.00	0.00	0.00
	9	125.00	120.33	-18.38	9.20	34.05	96.80
	13	162.50	0.00	0.00	0.00	0.00	0.00
	17	200.00	192.53	36.69	-11.23	96.80	15.64
	21	162.50	0.00	0.00	0.00	0.00	0.00
	25	125.00	120.33	-18.38	5.29	34.05	96.80
	29	87.50	0.00	0.00	0.00	0.00	0.00
	33	50.00	55.55	-9.19	2.65	73.86	96.80
Diagonals	3	218.30	104.83	-160.86	55.86	163.58	382.25
	7	"	"	32.29	-7.96	382.25	163.58
	11	257.69	123.75	8.83	-10.66	"	128.88
	15	"	"	-29.16	18.50	128.88	382.25
	19	"	"	-29.16	8.93	"	"
	23	"	0.00	8.83	3.88	382.250	0.00
	27	218.30	104.83	32.29	-9.64	"	163.58
	31	"	"	-160.86	42.97	163.58	382.25

5.2.7 Truss Topology VII

The truss topology shown in Fig. 5.16 consists of 35 members and with 19 joints.

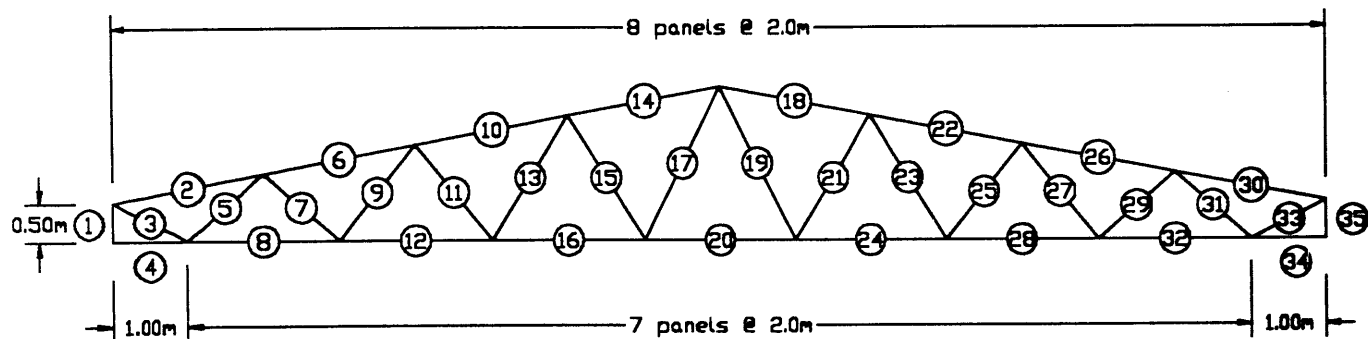


Fig. 5.16 Member numbering for the seventh topology.

The program was run 5 times until same minimum was obtained from the two seed values of 4 and 5.

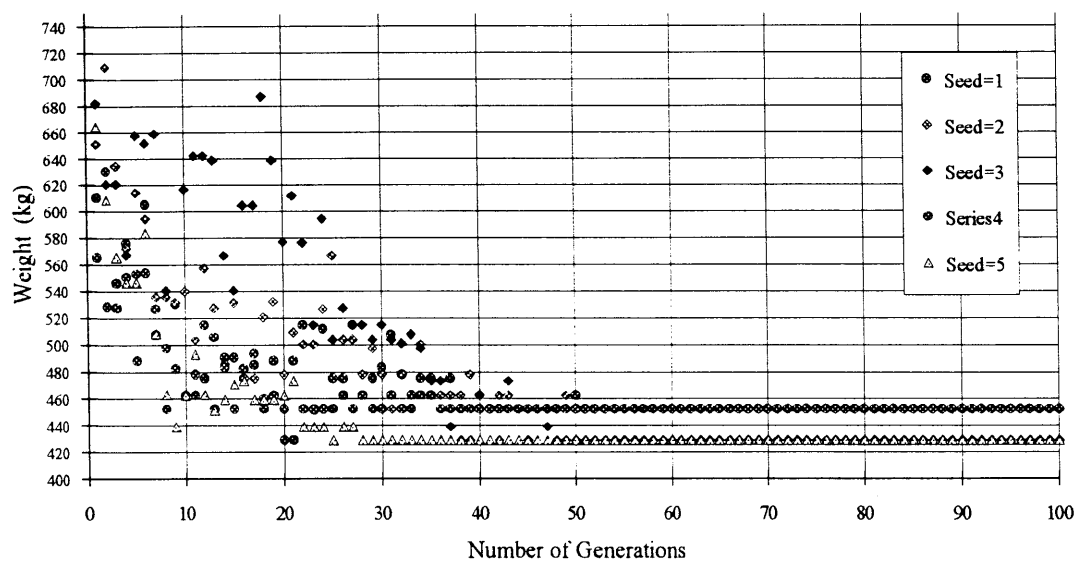


Fig.5.17 Design history for the seventh topology with fixed geometry and spacing.

The optimum sections obtained for the members of this truss topology are shown in Table 5.15 with the minimum weight of 429kg. This truss topology has provided the lightest weight among all seven trusses considered.

Table 5.15 Optimum sections for topology VII with fixed geometry and spacing.

Group Number	1 Upper chord members	2 Lower chord members	4 Diagonal members	Weight (kg)
Designation	 70x70x6	 50x50x4	 60x60x5	429.30

Table 5.15 lists the final analysis and design results. It is apparent that the member forces developed in the first and second groups were more or less of same order of magnitude as those in previous topologies. For that, the sections selected were same. Whereas for the diagonals, the single section angle 60x60x5 was selected which provides a compression resistance of 75.42kN. This is sufficient to resist the maximum compression force 71.03kN which developed in the member number 5 having the length of 132.88cm. The maximum deflection was 1.5cm which is 19% of the allowable deflection.

Table 5.16 Member forces and capacities for the optimum design of the seventh topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I (kN)	D+W (kN)	D+I (kN)	D+W (kN)
Upper Chord Members	2	203.49	96.25	-95.29	34.36	213.30	450.45
	6	"	"	-167.23	59.18	"	"
	10	"	"	-175.61	59.34	"	"
	14	"	"	-159.88	50.16	"	"
	18	"	"	-159.88	47.13	"	"
	22	"	"	-175.61	50.36	"	"
	26	"	"	-167.23	47.03	"	"
	30	"	"	-95.29	26.36	"	"
Lower Chord Members	4	100	0.00	0.00	-5.67	0.00	0.00
	8	200	121.5	146.97	-56.71	215.60	71.82
	12	"	"	176.37	-62.41	"	"
	16	"	"	169.59	-54.16	"	"
	20	"	"	146.97	-39.80	"	"
	24	"	"	169.59	-43.89	"	"
	28	"	"	176.37	-46.50	"	"
	32	"	"	146.97	-39.26	"	"
Diagonals	34	100	0.00	0.00	0.00	0.00	0.00
	1	50.00	49.02	-73.52	26.95	124.36	161.15
	3	111.80	81.32	104.14	-36.18	161.15	89.90
	5	132.88	95.72	-71.03	24.67	75.42	161.15
	7	"	"	22.98	-5.66	161.15	75.42
	9	160.08	115.31	-19.88	4.91	59.07	161.15
	11	"	"	-5.83	7.39	"	"
	13	190.80	137.44	5.30	-6.78	161.15	45.35
	15	"	"	-23.66	15.11	45.35	161.15
	17	223.61	161.07	22.66	-14.39	161.15	35.01
	19	"	"	22.66	-6.94	"	"
	21	190.80	137.44	-23.66	7.25	45.35	161.15
	23	"	"	5.30	2.63	161.15	0.00
	25	160.08	115.31	-5.82	-2.84	59.07	0.00
	27	"	"	-19.88	5.93	"	161.15
	29	132.88	95.72	22.98	-6.87	161.15	75.42
	31	"	"	-71.03	18.97	75.42	161.15
	33	111.80	81.32	104.14	-27.82	161.15	89.90
	35	50.00	49.02	-73.52	19.83	124.36	161.15

5.2.8 Summary for Design Examples with Fixed Geometry and Fixed Spacing

The optimum angle sections obtained for each candidate topology are given in Table 5.17. The results show that the lightest topology among them was the seventh one and heaviest was the third one. Although truss III consists of the least number of members with no redundant members, it was still the heaviest.

In the design of these seven topologies the compression resistance was the governing factor. And selection of a section was based on the magnitude of the compression force and the length of the member that is carrying that force.

For topology III ,IV and VI and particularly for diagonals group, the sections selected was the largest, 90x90x8. By referring back to the tables of members forces, it is clear that the maximum compression force that controlled the selection of that section was that due to D+I load case, not due to D+W load case, which was the case of other topologies. Hence, those three topologies were the heaviest.

The sections selected for all topologies and in particular for the first groups, were more or less close to each others, the difference comes in the fourth group where the smallest section was that for seventh topology which led to producing the lightest topology.

Table 5.17 Summary of the optimum designs of the seven truss topologies with fixed geometry and fixed spacing.

TRUSS TOPOLOGIES	Upper Chord Double Section	Lower Chord Double Section	Verticals Single Section	Diagonals Single Section	Weight (kg)
	70x70x6	60x60x6	45x45x4	70x70x6	520.18
	70x70x6	50x50x4	45x45x4	70x70x6	451.54
	70x70x6	60x60x6	45x45x4	90x90x8	603.76
	70x70x6	50x50x4	50x50x4	90x90x8	554.61
	70x70x6	50x50x4	45x45x4	70x70x6	456.99
	70x70x6	45x45x4	45x45x4	90x90x8	531.70
	70x70x6	50x50x4	-	60x60x5	429.30

5.3 DESIGN EXAMPLES WITH FIXED GEOMETRY AND VARIABLE TRUSS SPACING

In the previous group of examples, the geometry and the spacing between the trusses were fixed. In order to investigate the effect of variable truss spacing, the same set of seven truss topologies once more considered. Each topology is designed by using program GA3.FOR where the spacing between trusses is taken an additional design parameter. The values of 300., 320., 340., 360., 380., 400., 410., 420. 430. 440. 450. 460. 470. 480. 490. 500. 510. 520. 530. 540. 550. 560. 570., 580., 590., 600., 620., 640., 650., 660., 680., and 700. in centimeters, are selected for possible truss spacing. They make a set of 32 values which makes it possible to use the bit string length for the spacing as 5.

It is apparent that when spacing is taken as variable in the design problem the overall weight of the roof rather than an individual truss has to be taken as an objective function as given in eqn (4.2), where the number of trusses represent only the number of intermediate trusses. It is also apparent that the variable spacing necessitates the inclusion of the weight of purlins in the overall weight of the roof. Their weight increases with the increase of the spacing. However, in this study, in order not to complicate the design problem due to existence of large variety of different type of purlins used in the practice, their weight is not included in the overall roof weight.

5.3.1 Truss Topology I

The first truss topology was designed four times in each having the seed value of 1,2,3 and 4. The design history during the generations are shown in Fig. 5.17. The overall weight of the roof obtained were very close to each other with all seed values. However, seeds 2 and 4 produced the least weight which is 4118.37kg and the optimum truss spacing is 680cm as given in Table 5.18.

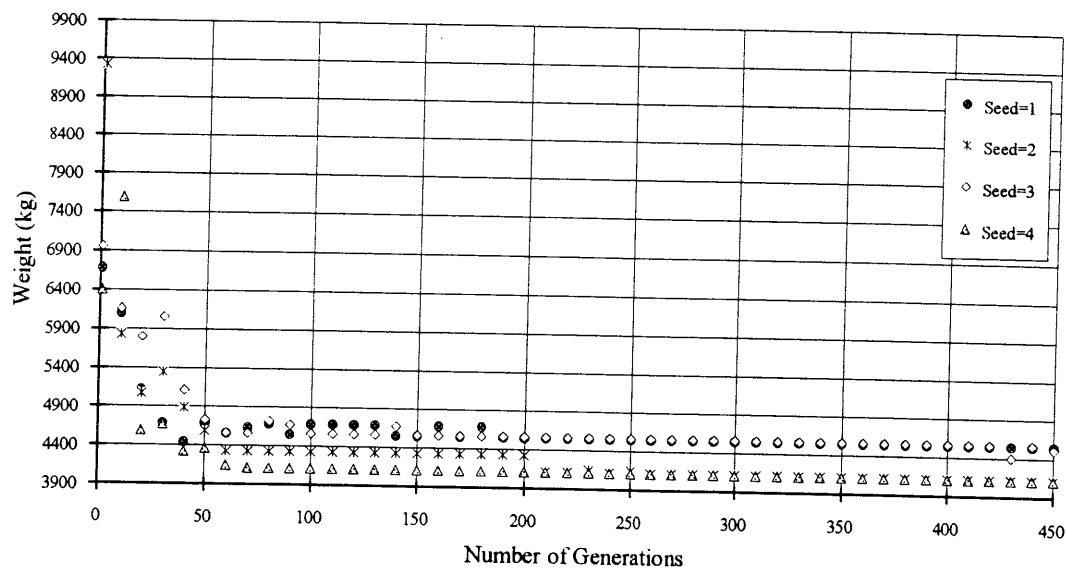


Fig. 5.18 Design history of the first topology with fixed geometry and variable spacing.

The design obtained here for one truss is heavier than that obtained in the fixed spacing problem for the same truss topology. This result is obvious because the trusses in this example will be loaded more due to the increase of spacing from 500cm to 680cm. But in overall the weight here is lighter due to less number of trusses, seven trusses, to cover the whole length of the building, as compared to nine trusses in the case of fixed spacing.

Table 5.18 Optimum sections for topology I with fixed geometry and variable spacing.

Group Number	1	2	3	4	Best Truss Spacing (cm)	Weight (kg)	
	Upper chord members	Lower chord members	Vertical members	Diagonal members		One Truss	All Trusses
Designation	┐L70x70x6	┐L70x70x6	L 50x50x5	L 70x70x6	680	588.34	4118.37

The selected sections for the different groups of members were all larger than those selected for fixed spacing for same truss topology, except for group#4, where the same section was still satisfactory. The maximum deflection was 1.44cm.

Table 5.19 Member forces and capacities for the optimum design of the first topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to D+I (kN) D+W (kN)		Member Capacities due to D+I (kN) D+W (kN)	
Upper Chord Members	2	203.49	88.14	-202.58	72.10	244.43	514.80
	6	"	"	-203.00	75.71	"	"
	10	"	"	-234.54	75.71	"	"
	14	"	"	-234.76	79.24	"	"
	16	"	"	-234.76	69.40	"	"
	18	"	"	-234.54	66.87	"	"
	22	"	"	-203.00	57.93	"	"
	26	"	"	-202.58	55.34	"	"
Lower Chord Members	4	200.00	0.00	0.00	-7.71	0.00	0.00
	8	400.00	160.23	239.90	-84.90	450.45	99.62
	12	400.00	160.23	199.92	-54.14	450.45	99.62
	20	400.00	160.23	239.90	-63.25	450.45	99.62
	24	200.00	0.00	0.00	0.00	0.00	0.00
Verticals	1	50.00	53.03	-100.00	36.65	103.68	132.82
	5	87.50	77.56	-24.99	12.51	80.20	"
	11	162.50	141.09	-24.99	12.51	36.44	"
	17	162.50	141.09	-24.99	7.20	36.44	"
	23	87.50	77.56	-24.99	7.20	80.20	"
	27	50.00	53.03	-100.00	26.97	103.68	"
Diagonals	3	206.16	126.98	205.18	-71.28	225.22	72.55
	7	235.85	145.27	-47.26	11.64	58.62	225.22
	9	235.85	145.27	-10.82	13.22	58.62	225.22
	13	282.84	174.21	43.45	-27.59	225.22	43.18
	15	282.84	174.21	43.44	-13.30	225.22	43.18
	19	235.85	145.27	-10.82	-4.88	58.62	0.00
	21	235.85	145.27	-47.26	14.11	58.62	225.22
	25	206.16	126.98	205.18	-54.80	225.22	72.55

5.3.2 Truss Topology II

The history of the generations for the design of truss topology II with three different seed values is shown in Fig.5.19. The minimum weight was obtained in early generations from seed value 2 which was 3666.12kg.

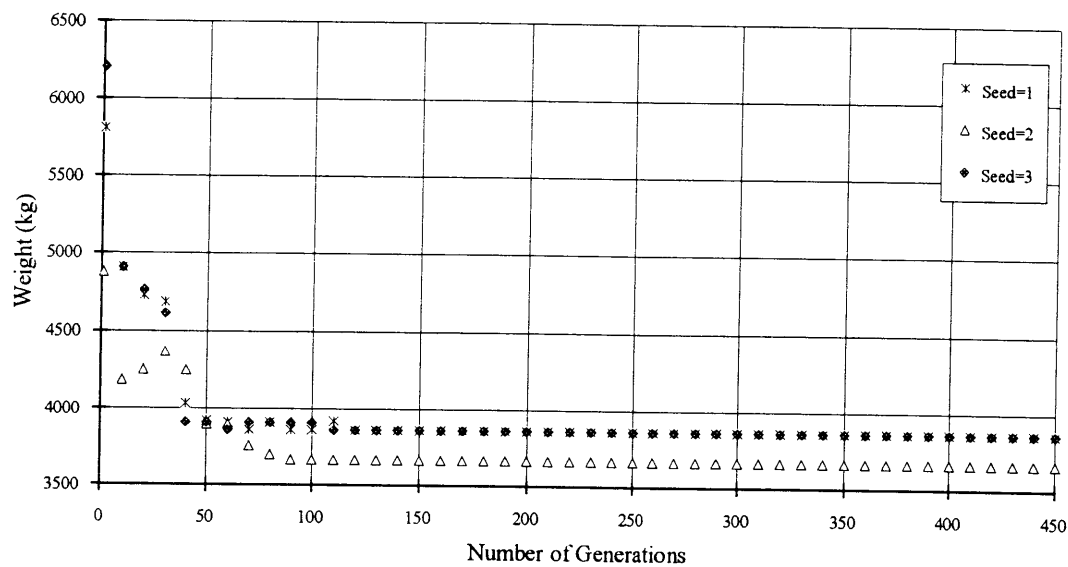


Fig.5.19 Design history for the second topology with fixed geometry and variable spacing.

The best truss spacing obtained was 560cm which has led to a truss which is slightly heavier than the same truss when spacing was fixed to 500cm. This increase in weight was due to the increase in loading carried by one truss which has affected the selection of a slightly larger section for the third group only. The rest of the groups capacities of the sections selected for the fixed spacing were in the margin of that increase in the member forces in this case. The member forces and capacities of this truss are given in Table 5.21.

Table 5.20 Optimum sections for topology II with fixed geometry and variable spacing.

Group Number	1 Upper chord members	2 Lower chord members	3 Vertical members	4 Diagonal members	Best Truss Spacing (cm)	Weight (kg)	
						One Truss	All Trusses
Designation	∟L 70x70x6	∟L 50x50x4	L 45x45x5	L 70x70x6	560	458.27	3666.12

The maximum deflection for this truss topology was 1.74cm.

Table 5.21 Member forces and capacities for the optimum design of the second topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to D+I (kN) D+W (kN)		Member Capacities due to D+I (kN) D+W (kN)	
Upper Chord Members	2	203.49	96.25	-167.65	59.65	213.30	450.45
	6	"	"	-201.09	68.96	"	"
	10	"	"	-193.31	62.37	"	"
	14	"	"	-167.50	49.25	"	"
	18	"	"	-167.50	49.65	"	"
	22	"	"	-193.31	55.11	"	"
	26	"	"	-201.09	56.07	"	"
	30	"	"	-167.65	45.80	"	"
Lower Chord Members	4	200.00	0.00	0.00	-6.35	0.00	0.00
	8	"	121.50	164.61	-63.52	215.60	71.82
	12	"	"	197.53	-69.90	"	"
	16	"	"	189.93	-60.66	"	"
	20	"	"	189.93	-49.15	"	"
	24	"	"	197.53	-52.08	"	"
	28	"	"	164.61	-43.97	"	"
	32	"	0.00	0.00	0.00	0.00	0.00
Verticals	1	50.00	55.74	-82.35	30.18	90.70	119.07
	5	87.50	84.84	-41.16	14.30	65.78	119.07
	9	125.00	121.15	-14.41	3.55	41.45	119.07
	13	162.50	157.50	4.75	-5.78	119.07	27.23
	17	200.00	193.84	41.11	-12.58	119.07	19.01
	21	162.50	0.00	4.75	2.12	119.07	0.00
	25	125.00	121.15	-14.41	4.31	41.45	119.07
	29	87.50	84.84	-41.16	11.00	65.78	119.07
	33	50.00	55.74	-82.35	22.21	90.70	119.07
Diagonals	3	206.16	126.98	168.63	-58.59	225.22	72.55
	7	218.30	134.46	35.93	-8.85	225.22	66.36
	11	235.85	145.27	-8.89	10.88	58.62	225.22
	15	257.69	158.72	-32.63	20.69	50.61	225.22
	19	257.69	158.72	-32.63	9.99	50.61	225.22
	23	235.85	145.27	-8.89	-4.03	58.62	0.00
	27	218.30	134.46	35.93	-10.73	225.22	66.36
	31	206.16	126.98	168.62	-45.04	225.22	72.55

5.3.3 Truss Topology III

The third topology is designed three times from three different seed values as shown in Fig. 5.20 with seed values 2 and 3 obtaining the least overall weight of 5206.61kg from early generations.

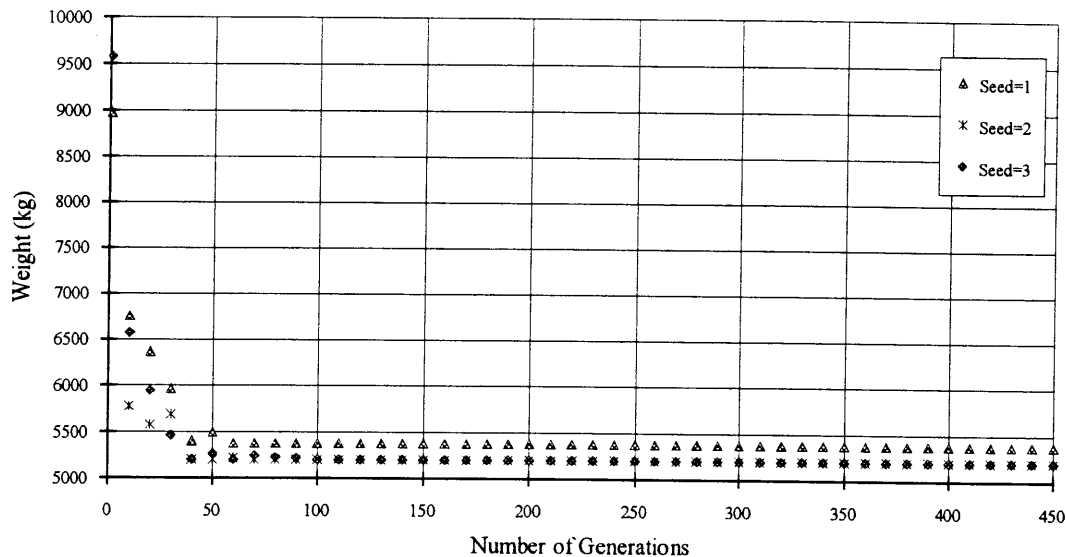


Fig.5.20 Design history for the third topology with fixed geometry and variable spacing.

The best spacing obtained for this truss topology was 640cm which requires seven trusses to cover the whole given length of the building. This results in a truss which is heavier than that obtained for same topology with fixed spacing. Although, the weight of one truss is heavier, the overall weight of the trusses required in this case is lighter as compared to that of fixed spacing where nine trusses were required which summed up to an overall weight of 5433.84kg.

Table 5.22 Optimum sections for topology III with fixed geometry and variable spacing.

Group Number	1	2	3	4	Best Truss Spacing (cm)	Weight (kg)	
	Upper chord members	Lower chord members	Vertical members	Diagonal members		One Truss	All Trusses
Designation	└┐L80x80x6	└┐L70x70x6	L45x45x4	L100x100x10	640	743.80	5206.61

Due to the increase in spacing, more load is carried by each truss. Consequently, the member forces increase and sections selected for the different groups of members become larger.

The maximum deflection was 1.25cm.

Table 5.23 Member forces and capacities for the optimum design of the third topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to D+I (kN) D+W (kN)		Member Capacities due to D+I (kN) D+W (kN)	
Upper Chord Members	2	203.49	96.25	0.71	1.39	514.80	450.45
	6	"	88.14	-229.37	78.68	244.43	514.80
	10	"	"	-229.60	82.02	"	"
	14	"	"	-191.50	56.30	"	"
	16	"	"	-191.50	56.74	"	"
	18	"	"	-229.60	66.35	"	"
	22	"	"	-229.37	63.96	"	"
	24	"	96.25	0.71	0.97	514.80	450.45
Lower Chord Members	4	400.00	160.23	188.16	-72.61	450.45	99.62
	8	"	"	217.11	-69.34	"	"
	12	"	"	217.11	-56.19	"	"
	20	"	"	188.16	-50.26	"	"
Verticals	1	50.00	55.55	-11.77	5.89	73.86	96.80
	7	125.00	120.33	-23.52	11.78	34.05	96.80
	13	200.00	192.53	46.96	-14.38	96.80	15.64
	19	125.00	120.33	-23.52	6.77	34.05	96.80
	25	50.00	55.55	-11.77	3.39	73.86	96.80
Diagonals	3	218.30	94.67	-204.88	71.17	257.51	528.00
	5	218.30	94.67	40.80	-10.04	528.00	257.51
	9	257.69	111.76	11.11	-13.60	528.00	206.83
	11	257.69	111.76	-37.29	23.69	206.83	528.00
	15	257.69	111.76	-37.29	11.42	206.83	528.00
	17	257.69	0.00	11.11	5.03	528.00	0.00
	21	218.30	94.67	40.80	-12.19	528.00	257.51
	23	218.30	94.67	-204.88	54.73	257.51	528.00

5.3.4 Truss Topology IV

Truss topology IV was optimized four times by using the program GA3.FOR in each time using a different seed value. The minimum weight obtained using the four independent seed values were very close to each other. The least of these was obtained from seed value 4 as shown in Fig. 5.21 with a value of 3666.12kg.

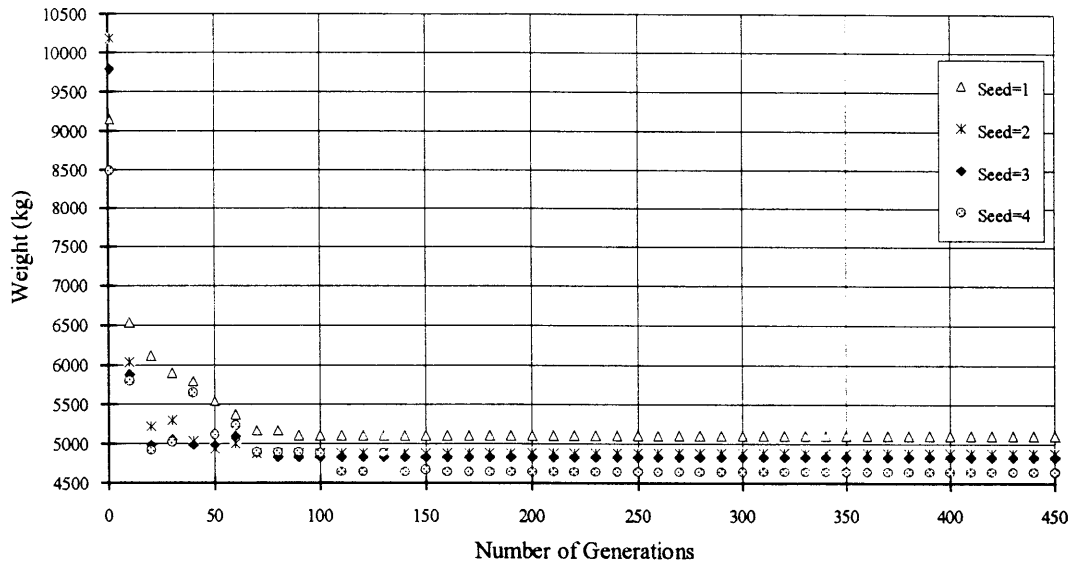


Fig. 5.21 Design history for the fourth topology with fixed geometry and variable spacing.

The best truss spacing obtained for this truss topology was 560cm as given in Table 5.24. This truss spacing had produced a slightly heavier truss than that obtained for same topology but with fixed spacing. On the other hand, the total weight of the eight trusses required in this case to cover the whole given length of the building, is lighter than the one obtained for fixed spacing, where the total weight was 4991.49kg.

Table 5.24 Optimum sections for topology IV with fixed geometry and variable spacing.

Group Number	1	2	3	4	Best Truss Spacing (cm)	Weight (kg)	
	Upper chord members	Lower chord members	Vertical members	Diagonal members		One Truss	All Trusses
Designation	┐L70x70x6	┐L50x50x4	L50x50x4	L100x100x8	560	458.27	3666.12

The member forces and capacities for this truss are given in Table 5.25. The increase in member forces due to the optimum spacing 560cm, was in the

Table 5.25 Member forces and capacities for the optimum design of the fourth topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I (kN)	D+W (kN)	D+I (kN)	D+W (kN)
Upper Chord Members	2	203.49	96.25	-0.14	1.47	213.30	450.45
	6	"	"	-167.59	62.46	"	"
	10	"	"	-201.05	71.78	"	"
	14	"	"	-193.28	65.20	"	"
	18	"	"	-193.28	57.14	"	"
	22	"	"	-201.05	58.09	"	"
	26	"	"	-167.59	47.82	"	"
	30	"	"	-0.14	1.05	"	"
Lower Chord Members	4	200.00	121.50	164.61	-63.52	215.60	71.82
	8	"	"	197.53	-69.90	"	"
	12	"	"	189.93	-60.66	"	"
	16	"	"	164.61	-44.58	"	"
	20	"	"	164.61	-44.58	"	"
	24	"	"	189.93	-49.15	"	"
	28	"	"	197.53	-52.08	"	"
	32	"	"	164.61	-43.97	"	"
Verticals	1	50.00	52.88	-10.29	5.16	77.21	107.80
	5	87.50	77.06	20.58	-5.07	107.80	61.04
	9	125.00	107.65	-6.17	7.51	42.45	107.80
	13	162.50	139.94	-25.32	16.08	29.00	107.80
	17	200.00	0.00	0.00	0.00	0.00	0.00
	21	162.50	139.94	-25.32	7.75	29.00	107.80
	25	125.00	107.65	-6.17	-2.76	42.45	0.00
	29	87.50	77.06	20.58	-6.15	107.80	61.04
	33	50.00	52.88	-10.29	2.96	77.21	107.80
Diagonals	3	218.30	94.19	-179.51	62.35	198.99	429.00
	7	235.85	101.76	-38.66	9.52	181.51	429.00
	11	257.69	111.19	10.08	-11.98	429.00	161.80
	15	282.84	122.04	35.82	-22.75	429.00	141.99
	19	282.84	122.04	35.82	-10.97	429.00	141.99
	23	257.69	0.00	10.08	4.28	429.00	0.00
	27	235.85	101.76	-38.66	11.55	181.51	429.00
	31	218.30	94.19	-179.51	47.95	198.99	429.00

margin of the carrying capacities of the same sections selected for the same truss topology with fixed spacing. Except for the fourth group where a larger section, 100x100x8, was selected to resist the increase in the compression force, 179.51kN, developed in member 3. This required to provide a compression resistance of 198.99kN for member length of 218.3cm. The maximum deflection in Truss topology IV was 1.77cm.

5.3.5 Truss Topology V

The history of the generations for the design of truss topology V with three different seed values is shown in Fig.5.22. The minimum weight was obtained in early generations from seed values 1 and 2 as 3652.48kg.

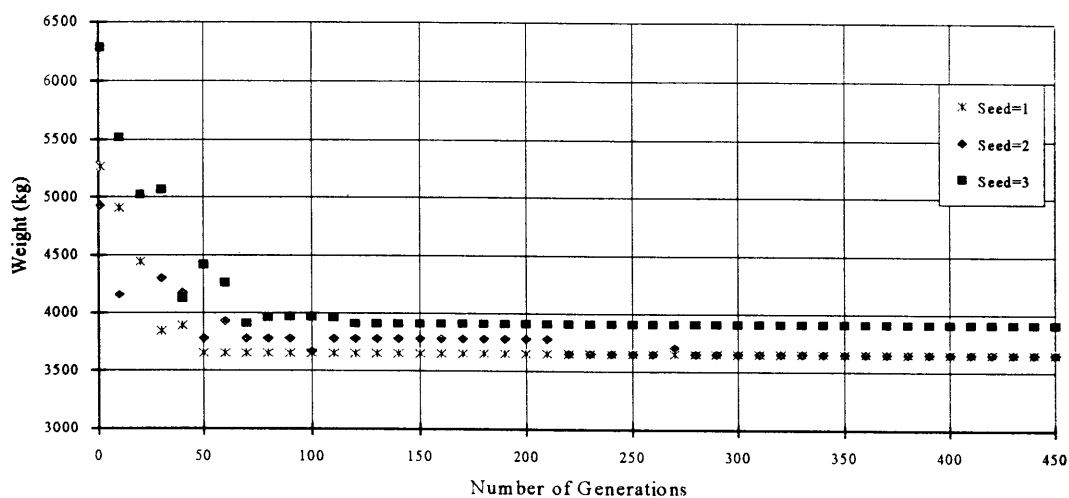


Fig.5.22 Design history for the fifth topology with fixed geometry and variable spacing.

The best truss spacing obtained for truss V was 680cm as given in Table 5.26. This truss spacing requires seven intermediate trusses to cover the whole given length of

Table 5.26 Optimum sections for topology V with fixed geometry and variable spacing.

Group Number	1	2	3	4	Best Truss Spacing (cm)	Weight (kg)	
	Upper chord members	Lower chord members	Vertical members	Diagonal members		One Truss	All Trusses
Designation	┐L80x80x6	┐L50x50x5	┐L50x50x5	┐L70x70x6	680	521.78	3652.48

Table 5.27 Member forces and capacities for the optimum design of the fifth topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I (kN)	D+W (kN)	D+I (kN)	D+W (kN)
Upper Chord Members	2	203.49	88.14	-202.40	72.04	244.43	514.80
	6	"	"	-202.82	75.66	"	"
	10	"	"	-234.50	75.70	"	"
	14	"	"	-234.72	79.24	"	"
	18	"	"	-234.72	69.39	"	"
	22	"	"	-234.50	66.86	"	"
	26	"	"	-202.83	57.88	"	"
	30	"	"	-202.40	55.30	"	"
Lower Chord Members	4	200.00	0.00	0.00	-7.71	0.00	0.00
	8	"	122.11	240.00	-84.93	265.65	91.40
	12	"	"	"	-84.93	"	"
	16	"	"	"	-54.16	"	"
	20	"	"	"	-54.16	"	"
	24	"	"	"	-63.28	"	"
	28	"	"	"	-63.28	"	"
	32	"	0.00	0.00	0.00	0.00	0.00
Verticals	1	50.00	53.03	-100.00	36.65	103.68	132.82
	5	87.50	77.56	-24.99	12.51	80.20	132.82
	9	125.00	0.00	0.00	0.00	0.00	0.00
	13	162.50	141.09	-24.99	12.51	36.44	132.82
	17	200.00	0.00	0.00	0.00	0.00	0.00
	21	162.50	141.09	-24.99	7.20	36.44	132.82
	25	125.00	0.00	0.00	0.00	0.00	0.00
	29	87.50	77.56	-24.99	7.20	80.20	132.82
	33	50.00	53.03	-100.00	26.97	103.68	132.82
Diagonals	3	206.16	126.98	204.98	-71.22	225.22	72.55
	7	235.85	145.27	-47.31	11.66	58.62	225.22
	11	235.85	145.27	-10.81	13.22	58.62	225.22
	15	282.84	174.21	43.44	-27.59	225.22	43.18
	19	282.84	174.21	43.44	-13.30	225.22	43.18
	23	235.85	145.27	-10.81	-4.88	58.62	0.00
	27	235.85	145.27	-47.31	14.13	58.62	225.22
	31	206.16	126.98	204.98	-54.75	225.22	72.55

the building, which produced an overall weight of 3652.48kg. When compared with same topology with fixed spacing, the number of trusses required were 9 and the overall weight produced was 4112.9kg which was clearly heavier than the existing one.

In order to resist the increase that occurred in member forces, the best sections selected for this truss were all larger than those obtained for the same truss with fixed spacing. Except for the fourth group where same section resistance was in the margin of carrying the maximum compression force developed in member 3 of that group. The maximum deflection for truss topology V was 1.83cm.

5.3.6 Truss Topology VI

The history of the generations for the design of truss topology V with three different seed values is shown in Fig.5.23. The minimum weight was obtained in early generations from seed values 2 and 3, which was 4717.87kg.

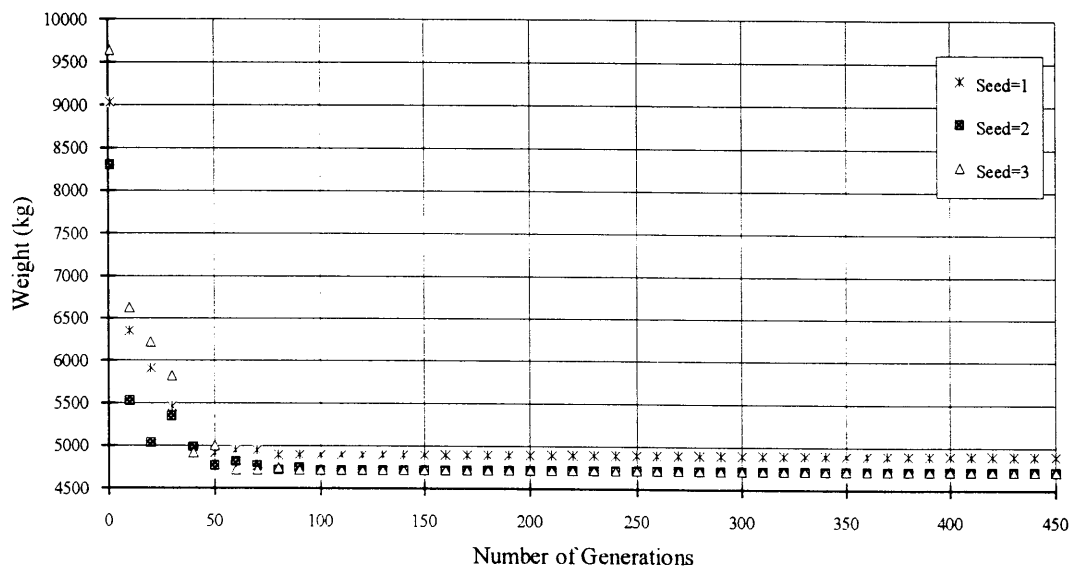


Fig.5.23 Design history for the sixth topology with fixed geometry and variable spacing.

The best truss spacing obtained for this truss topology was 660cm, which has produced the overall weight of 4717.87kg for 7 trusses to cover the whole length of

the building. On the other hand, for same topology but with fixed spacing of 500cm the overall weight was 4785.3kg which was slightly heavier than that obtained here.

Table 5.28 Optimum sections for topology VI with fixed geometry and variable spacing.

Group Number	1	2	3	4	Best Truss Spacing (cm)	Weight (kg)	
	Upper chord members	Lower chord members	Vertical members	Diagonal members		One Truss	All Trusses
Designation	L80x80x6	L50x50x5	L45x45x4	L100x100x10	660	673.98	4717.87

The sections selected here were all larger than those selected for same topology when spacing was fixed, except for the third group, where the same section still provides the sufficient resistance to resist the increase in the maximum compression force of this group. Finally, the maximum deflection for this truss topology was 1.72cm.

Table 5.29 Member forces and capacities for the optimum design of the sixth topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to D+I (kN) D+W (kN)		Member Capacities due to D+I (kN) D+W (kN)	
Upper Chord Members	2	203.49	96.25	0.99	1.35	514.80	450.45
	6	"	88.14	-236.44	81.11	244.43	514.80
	10	"	"	-236.68	84.56	"	"
	14	"	"	-197.48	58.06	"	"
	18	"	"	-197.48	58.51	"	"
	22	"	"	-236.68	68.39	"	"
	26	"	"	-236.44	65.93	"	"
	30	"	96.25	0.99	0.93	514.80	450.45
Lower Chord Members	4	200.00	122.11	194.12	-74.91	265.65	91.40
	8	"	"	194.12	-74.91	"	"
	12	"	"	223.99	-71.53	"	"
	16	"	"	223.99	-71.53	"	"
	20	"	"	223.99	-57.97	"	"
	24	"	"	223.99	-57.97	"	"
	28	"	"	194.12	-51.85	"	"
	32	"	"	194.12	-51.85	"	"
Verticals	1	50.00	55.55	-12.13	6.08	73.86	96.80
	5	87.50	0.00	0.00	0.00	0.00	0.00
	9	125.00	120.33	-24.26	12.15	34.05	96.80
	13	162.50	0.00	0.00	0.00	0.00	0.00
	17	200.00	192.53	48.43	-14.83	96.80	15.64
	21	162.50	0.00	0.00	0.00	0.00	0.00
	25	125.00	120.33	-24.26	6.98	34.05	96.80
	29	87.50	0.00	0.00	0.00	0.00	0.00
	33	50.00	55.55	-12.13	3.49	73.86	96.80
Diagonals	3	218.30	94.67	-211.10	73.33	257.51	528.00
	7	218.30	94.67	42.00	-10.34	528.00	257.51
	11	257.69	111.76	11.41	-14.02	528.00	206.83
	15	257.69	111.76	-38.45	24.44	206.83	528.00
	19	257.69	111.76	-38.45	11.77	206.83	528.00
	23	257.69	0.00	11.41	5.20	528.00	0.00
	27	218.30	94.67	42.00	-12.55	528.00	257.51
	31	218.30	94.67	-211.10	56.39	257.51	528.00

5.3.7 Truss Topology VII

The design history for truss topology VII, is shown in Fig. 5.24, where the algorithm with three different seed values was run and the minimum weight was obtained from seed value 3 which was 3618.36kg.

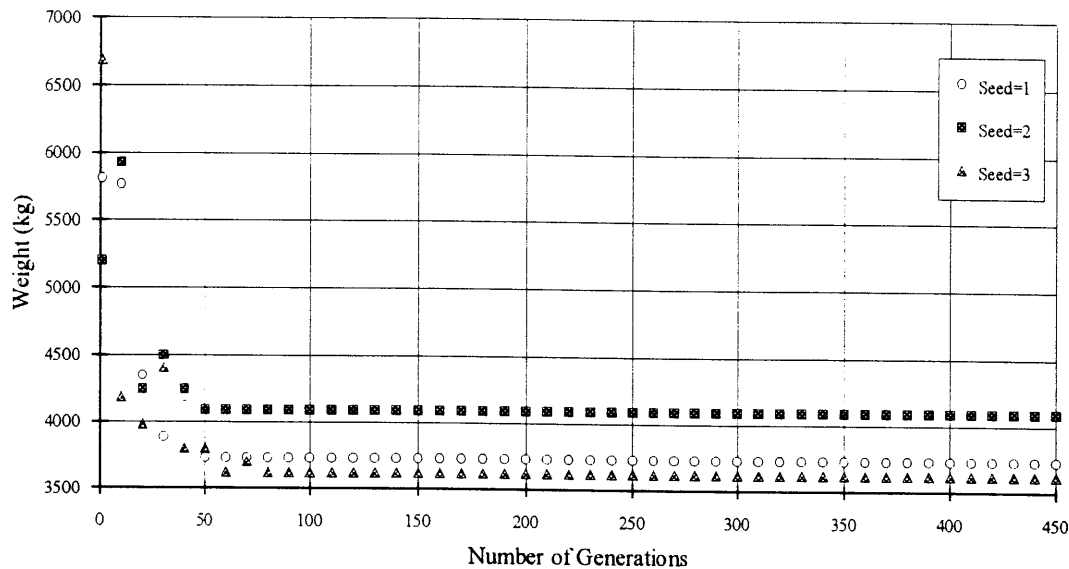


Fig.5.24 Design history for the seventh topology with fixed geometry and variable spacing.

The best truss spacing obtained for this truss topology was 560cm, this spacing produced an overall lighter design as compared to that obtained for fixed spacing, which was 3863.7kg for nine trusses.

Table 5.30 Optimum sections for topology VII with fixed geometry and variable spacing.

Group Number	1	2	4	Best Truss Spacing	Weight (kg)	
	Upper chord members	Lower chord members	Diagonal members		One Truss	All Trusses
Designation	┐┐ 70x70x6	┐┐ 50x50x4	L 60x60x6	560	452.30	3618.36

The increase in the members forces due the increase in truss spacing was in the margin of the carrying capacities of the sections selected for same topology with fixed spacing. Except for the third group where a slightly larger section of same

Table 5.31 Member forces and capacities for the optimum design of the seventh topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I(kN)	D+W (kN)	D+I (kN)	D+W (kN)
Upper Chord Members	2	203.49	96.25	-106.73	38.49	213.30	450.45
	6	"	"	-187.29	66.28	"	"
	10	"	"	-196.68	66.47	"	"
	14	"	"	-179.06	56.18	"	"
	18	"	"	-179.06	52.78	"	"
	22	"	"	-196.68	56.41	"	"
	26	"	"	-187.29	52.67	"	"
	30	"	"	-106.73	29.53	"	"
Lower Chord Members	4	100.00	0.00	0.00	-6.35	0.00	0.00
	8	200.00	121.50	164.61	-63.52	215.60	71.82
	12	"	"	197.53	-69.90	"	"
	16	"	"	189.93	-60.66	"	"
	20	"	"	164.61	-44.58	"	"
	24	"	"	189.93	-49.16	"	"
	28	"	"	197.53	-52.08	"	"
	32	"	"	164.61	-43.97	"	"
	34	100.00	0.00	0.00	0.00	0.00	0.00
Diagonals	1	50.00	49.13	-82.35	30.18	154.24	191.13
	3	111.80	81.32	116.57	-40.50	191.13	110.27
	5	132.88	95.72	-80.25	27.86	91.97	191.13
	7	132.88	95.72	26.23	-6.47	191.13	91.97
	9	160.08	115.31	-21.82	5.38	71.57	191.13
	11	160.08	115.31	-6.78	8.34	71.57	191.13
	13	190.80	137.44	6.40	-7.69	191.13	54.69
	15	190.80	137.44	-26.69	16.92	54.69	191.13
	17	223.61	161.07	25.31	-16.12	191.13	42.09
	19	223.61	161.07	25.31	-7.75	191.13	42.09
	21	190.80	137.44	-26.69	8.17	54.69	191.13
	23	190.80	0.00	6.40	2.78	191.13	0.00
	25	160.08	115.31	-6.78	-3.10	71.57	0.00
	27	160.08	115.31	-21.82	6.52	71.57	191.13
	29	132.88	95.72	26.23	-7.82	191.13	91.97
	31	132.88	95.72	-80.25	21.44	91.97	191.13
	33	111.80	81.32	116.57	-31.14	191.13	110.27
	35	50.00	49.13	-82.35	22.21	154.24	191.13

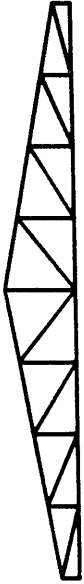
series was selected to resist the increase in the compression force, 80.25kN, that developed in member 5 of length 132.88cm, by providing the required resistance of 91.97kN. The maximum deflection of this truss topology was 1.66cm.

5.3.8 Summary for Design Examples with Fixed Geometry and Variable Truss Spacing

The optimum designs obtained for the seven different topologies are summarized in Table 5.32. It is clear that the weight of one truss has increased here as compared to that obtained for fixed spacing, where spacing was fixed to 500cm. But in over all, the total weight of trusses produced in this case was for all topologies lighter than the previous cases. This was due to best spacing selected for each topology being larger than that used in previous examples which reduces the number of trusses. The total number of trusses for the spacing of 5m was 9.

As with the previous cases, the lightest structure was produced from topology VII, and the heaviest was truss III.

Table 5.32 Summary of the optimum designs of the seven truss topologies with fixed geometry and variable spacing.

TRUSS TOPOLOGIES	Upper Chord Double Section	Lower Chord Double Section	Verticals Single Section	Diagonals Single Section	Optimum Truss Spacing (cm)	Weight of one Truss (kg)	No. of Trusses	Over all Weight (kg)
	80x80x6	70x70x6	50x50x5	70x70x6	680	588.34	7	4118
	70x70x6	50x50x4	45x45x5	70x70x6	560	458.27	8	3666
	80x80x6	70x70x6	45x45x4	100x100x10	640	743.80	7	5207
	70x70x6	50x50x4	50x50x4	100x100x8	560	580.47	8	4644
	80x80x6	50x50x5	50x50x5	70x70x6	680	521.78	7	3653
	80x80x6	50x50x5	45x45x4	100x100x10	660	673.98	7	4718
	70x70x6	50x50x4	-	60x60x6	560	452.25	8	3618

5.4 DESIGN EXAMPLES WITH VARIABLE GEOMETRY AND FIXED SPACING

In order to investigate the effect of change in the geometry of the trusses, the same set of trusses were designed once more by considering the height of apex as variable while fixing the spacing to 5m. For this, the values of 50., 55., 60., 65., 70., 75., 80., 85., 90., 95., 100., 105., 110., 115., 120., 125., 130., 135., 140., 145., 150., 155., 160., 165., 170., 175., 180., 185., 190., 195., 200., and 210 in centimeters are selected as possible values for the height of each truss. They make a set of 32 values which makes it possible to use the bit string length for the height as 5. The coordinates of joints on the upper chord is then computed accordingly while keeping the height of eaves as 50cm.

5.4.1 Truss Topology I

The first truss topology was designed four times from four different seed values. The variation of minimum weights during the generations are shown in Fig. 5.25, where the history of the generations is plotted in intervals of 10 and up to 450 generations. The least weight for this truss topology was obtained with seed values 1 and 4.

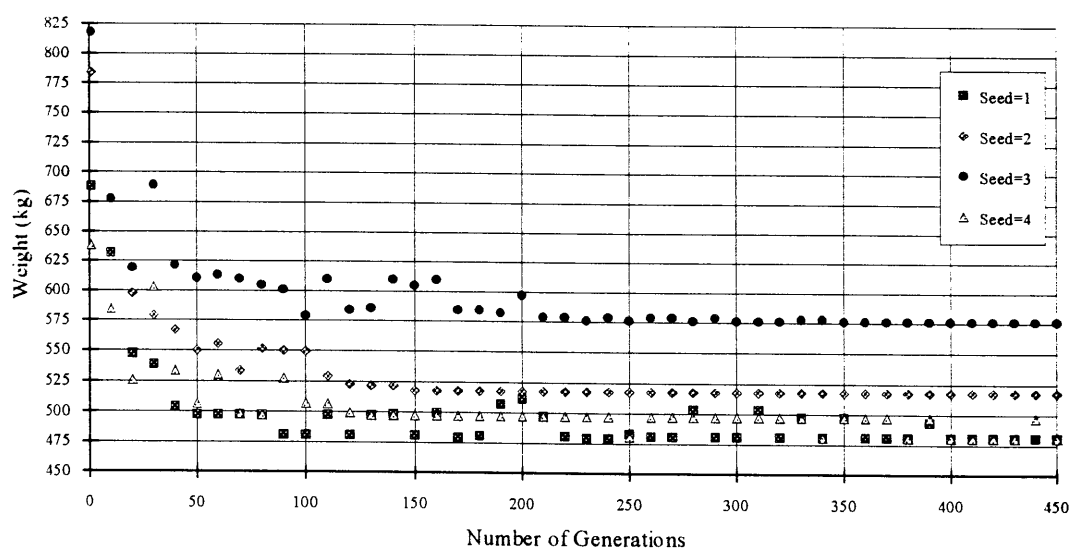


Fig. 5.25 Design history for the first topology with variable geometry and fixed spacing.

The best height for this truss topology was 185cm which has produced the optimum weight of 481kg as given in Table 5.33. The height obtained here has led to an overall reduction in the member forces as given in Table 5.34 where maximum compression force had dropped for the members of every group.

Table 5.33 Optimum sections and height for topology I with variable geometry and fixed spacing.

Group Number	1 Upper chord members	2 Lower chord members	3 Vertical members	4 Diagonal members	Best Height (cm)	Weight (kg)
Designation	∟ 60x60x6	∟ 60x60x6	∟ 50x50x4	∟ 60x60x6	185	481

Consequently, the weight of this truss came out to be lighter than that of the same topology (520kg) when geometry was fixed to apex height at ridge of 150cm. This reduction of weight was due to selecting smaller sections for first and fourth groups. The second group of members were not affected length wise from the change in height because they lie on the bottom chord. The sections for these members were same as that selected for the fixed geometry. Although it looks that the resistance is more than enough, but the smaller sections were not providing the sufficient compression resistance. For the verticals, the section 50x50x4 was selected which was larger than that of the fixed geometry. Although the resistance of the section 45x45x4 for length 50cm was enough to resist the compression force, the difference between the weight of these two sections was only 2kg. The maximum deflection was 1.05cm which is far from the allowable deflection.

Table 5.34 Member forces and capacities for the optimum design of the first topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to D+I (kN) D+W (kN)		Member Capacities due to D+I (kN) D+W (kN)	
Upper Chord Members	2	205.28	108.52	-136.56	48.03	156.03	382.25
	6	"	"	-137.07	51.38	"	"
	10	"	"	-150.05	48.67	"	"
	14	"	"	-150.32	51.94	"	"
	16	"	"	-150.32	45.61	"	"
	18	"	"	-150.05	43.27	"	"
	22	"	"	-137.07	39.50	"	"
	26	"	"	-136.56	37.09	"	"
Lower Chord Members	4	200.00	0.00	0.00	-6.99	0.00	0.00
	8	400.00	185.79	155.11	-55.33	382.25	65.77
	12	"	"	125.41	-33.56	382.25	65.77
	20	"	"	155.11	-40.05	382.25	65.77
	24	200.00	0.00	0.00	0.00	0.00	0.00
Verticals	1	50.00	52.88	-73.71	26.64	77.21	107.80
	5	96.25	83.26	-18.41	9.41	56.91	"
	11	188.75	162.55	-18.40	9.40	22.76	"
	17	188.75	162.55	-18.40	5.43	22.76	"
	23	96.25	83.26	-18.41	5.43	56.91	"
	27	50.00	52.88	-73.71	19.71	77.21	"
Diagonals	3	206.16	148.50	137.33	-46.71	191.13	48.22
	7	245.57	176.90	-25.74	5.78	35.83	191.13
	9	245.57	176.90	-10.81	10.52	35.83	191.13
	13	308.59	222.29	32.34	-20.34	191.13	23.88
	15	308.59	222.29	32.34	-10.25	191.13	23.88
	19	245.57	176.90	-10.81	4.12	35.83	191.13
	21	245.57	176.90	-25.74	8.21	35.83	191.13
	25	206.16	148.50	137.33	-36.17	191.13	48.22

5.4.2 Truss Topology II

The history of the generations for the design of truss topology II with four seed values is shown in Fig. 5.26. The minimum weight was obtained from seed values 1 and 4 with best height of 120 cm.

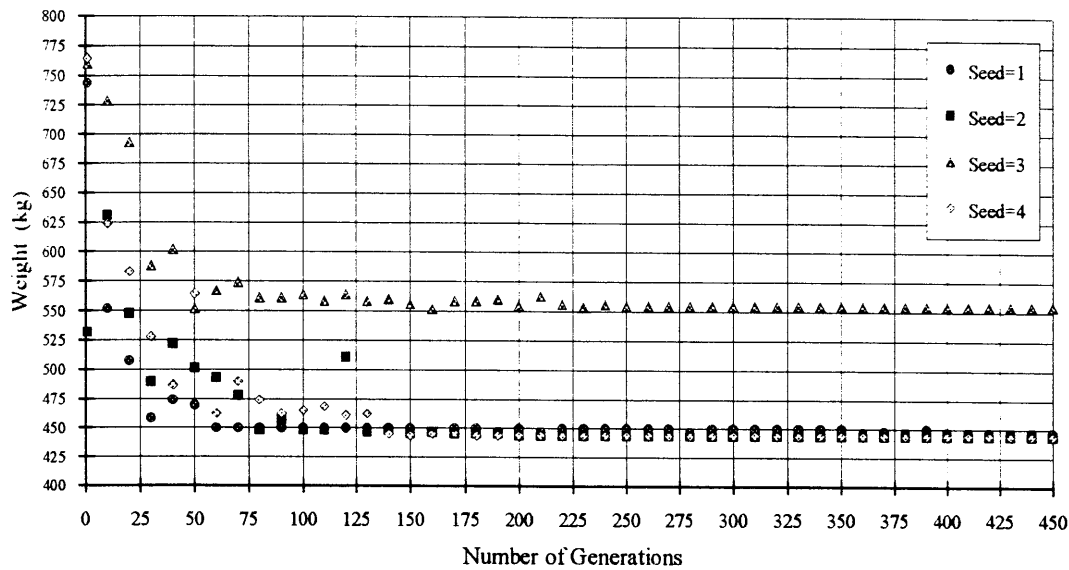


Fig. 5.26 Design history of the second topology with variable geometry and fixed spacing.

Table 5.35 shows the optimum sections selected by the algorithm for the different groups of members. The minimum weight obtained here is lighter than that obtained for the fixed geometry and spacing where apex height was 150cm.

Table 5.35 Optimum sections and height for topology II with variable geometry and fixed spacing

Group Number	1 Upper chord members	2 Lower chord members	3 Vertical members	4 Diagonal members	Best Height (cm)	Weight (kg)
Designation	└┐ 70x70x6	└┐ 50x50x4	L45x45x4	L 70x70x6	120	443

The best height obtained results in an increase in the member forces as seen from Table 5.36. At the same time, the lengths of members have shortened due to smaller height value. The increase in the maximum compression forces for each group was in the margin of the capacities of the same sections selected for the fixed geometry. For that the same sections were selected as those of the fixed geometry.

Table 5.36 Member forces and capacities for the optimum design of the second topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I (kN)	D+W (kN)	D+I (kN)	D+W (kN)
Upper Chord Members	2	202.24	95.84	-161.62	57.99	214.40	450.45
	6	"	"	-202.03	69.50	"	"
	10	"	"	-198.66	63.86	"	"
	14	"	"	-174.68	50.42	"	"
	18	"	"	-174.68	50.68	"	"
	22	"	"	-198.66	56.13	"	"
	26	"	"	-202.03	56.20	"	"
	30	"	"	-161.62	44.31	"	"
Lower Chord Members	4	200.00	0.00	0.00	-4.53	0.00	0.00
	8	200.00	121.50	160.48	-61.09	215.60	71.82
	12	"	"	200.08	-70.34	"	"
	16	"	"	196.51	-62.69	"	"
	20	"	"	196.51	-51.95	"	"
	24	"	"	200.08	-53.52	"	"
	28	"	"	160.48	-43.28	"	"
	32	200.00	0.00	0.00	0.00	0.00	0.00
Verticals	1	50.00	55.55	-73.41	27.17	73.86	96.80
	5	80.00	78.42	-40.15	14.15	57.85	96.80
	9	110.00	105.89	-15.85	4.10	40.87	96.80
	13	140.00	134.77	1.97	-4.21	96.80	28.58
	17	170.00	163.65	33.43	-9.97	96.80	20.75
	21	140.00	107.22	1.97	2.48	96.80	140.80
	25	110.00	105.89	-15.85	4.49	40.87	96.80
	29	80.00	78.42	-40.15	10.83	57.85	96.80
	33	50.00	55.55	-73.41	19.92	73.86	96.80
Diagonals	3	206.16	126.98	164.25	-57.92	225.23	72.55
	7	215.41	132.68	42.35	-10.94	225.23	67.77
	11	228.25	140.59	-4.21	8.76	61.81	225.23
	15	244.13	150.37	-29.17	19.04	55.39	225.23
	19	244.13	150.37	-29.17	8.70	55.39	225.23
	23	228.25	140.59	-4.21	-5.10	51.80	61.81
	27	215.41	132.68	42.35	-12.00	225.23	67.77
	31	206.16	126.98	164.25	-44.29	225.23	72.55

Note that the maximum tensile force in member 12 of the second group has reached to 93% of the tension capacity of the double angle section 50x50x4. Also note that the compression resistance was more than enough for member 3 belonging to the diagonals, which has a length of 206cm. For this group of members, if either section 60x60 with thicknesses 10 or 8 was selected, they produce heavier designs. On the other hand section 60x60x6 has insufficient strength to resist the maximum compression force carried by that member. Hence, section 70x70x6 provided the best selection for this group of members. The maximum deflection of this truss topology was 1.99cm

5.4.3 Truss Topology III

The design history of the third topology is shown in Fig. 5.27 where the variation of minimum weight is plotted against the generations. This truss topology was optimized three times every time with different seed value. The minimum weight for this truss topology was 600.29kg, obtained from seed values 2 and 3.

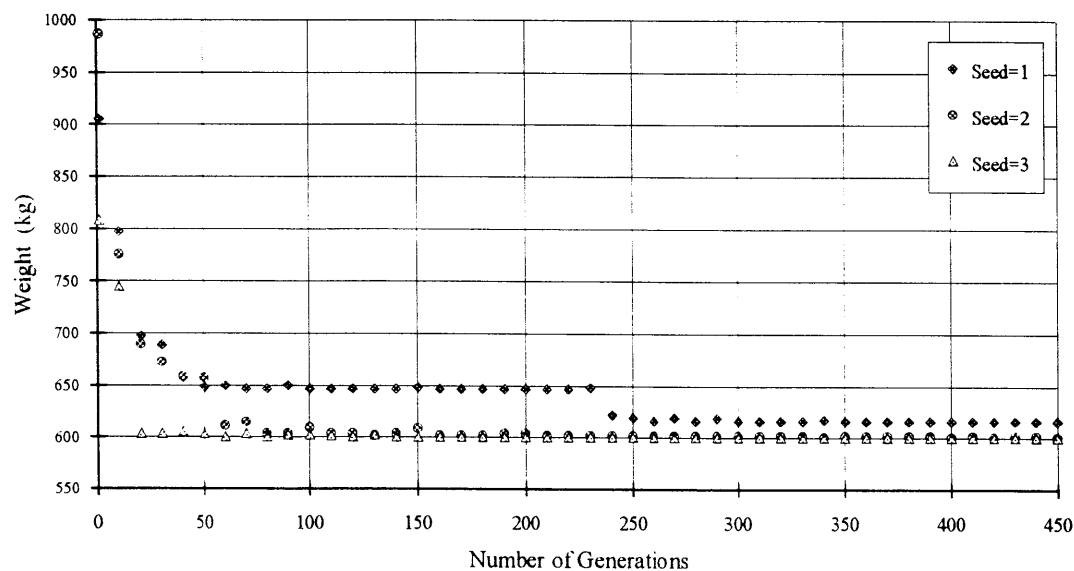


Fig. 5.27 Design history of the third topology with variable geometry and fixed spacing.

The optimum height obtained by the algorithm for this topology was 140cm. Table 5.37 shows the optimum sections for the different groups of members which

Table 5.37 Optimum sections and height for topology III with variable geometry and fixed spacing.

Group Number	1 Upper chord members	2 Lower chord members	3 Vertical members	4 Diagonal members	Best Height (cm)	Weight (kg)
Designation	∟ 70x70x6	∟ 60x60x6	∟ 45x45x4	∟ 90x90x8	140	600.29

Table 5.38 Member forces and capacities for the optimum design of the third topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to D+I (kN) D+W (kN)		Member Capacities due to D+I (kN) D+W (kN)	
Upper Chord Members	2	203.04	96.25	0.12	1.14	450.45	450.45
	6	"	96.11	-186.46	64.02	213.69	"
	10	"	"	-186.50	66.39	"	"
	14	"	"	-157.08	45.89	"	"
	16	"	"	-157.08	46.22	"	"
	18	"	"	-186.50	53.66	"	"
	22	"	"	-186.46	51.95	"	"
	24	"	96.25	0.12	0.81	450.45	450.45
Lower Chord Members	4	400.00	185.79	151.24	-58.09	382.25	65.77
	8	"	"	177.73	-56.74	"	"
	12	"	"	177.73	-46.35	"	"
	20	"	"	151.24	-40.54	"	"
Verticals	1	50.00	55.55	-9.19	4.58	73.86	96.80
	7	120.00	115.52	-18.36	9.15	36.16	96.80
	13	190.00	182.90	35.71	-10.83	96.80	17.12
	19	120.00	115.52	-18.36	5.26	36.16	96.80
	25	50.00	55.55	-9.19	2.63	73.86	96.80
Diagonals	3	217.31	104.36	-163.41	57.07	164.58	382.25
	5	217.31	104.36	34.73	-8.70	382.25	164.58
	9	253.03	121.51	7.83	-10.32	382.25	32.48
	11	253.03	121.51	-29.27	18.65	132.48	382.25
	15	253.03	121.51	-29.27	8.88	132.48	382.25
	17	253.03	126.05	7.83	4.26	382.25	125.30
	21	217.31	104.36	34.73	-10.20	382.25	164.58
	23	217.31	104.36	-163.41	43.80	164.58	382.25

shows no difference in the section designations as compared to those obtained for the fixed geometry. This is of no doubt, because the best height did not vary much from that of the fixed geometry which had a minor effect on the member lengths and a slight increase in member forces. This slight increase was in the margin of the carrying capacities of those sections. The maximum deflection for this truss topology was 1.24cm.

5.4.4 Truss Topology IV

For the fourth topology, five different seed values were used to obtain the best design as shown in Fig. 5.28. The five of the seed values have approached to almost the same value. However, the minimum weight is obtained with seed values 3 and 5 which has the value of 546.38kg.

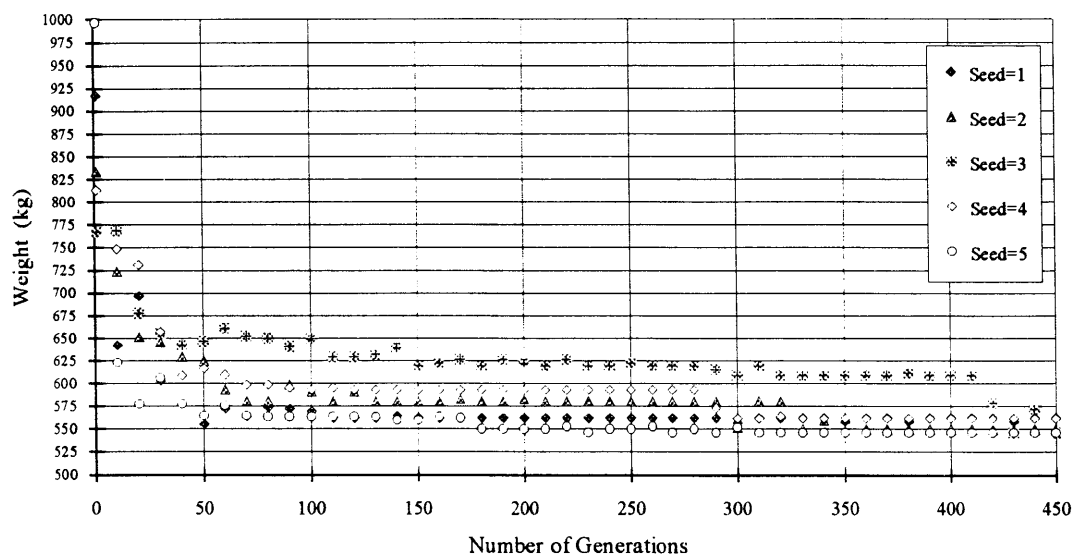


Fig. 5.28 Design history of the fourth topology with variable geometry and fixed spacing.

Best height obtained for this truss topology was 140cm which has produced a minimum weight of 546.38kg as given in Table 5.39. The minimum obtained here is again lighter than that obtained for same topology with fixed geometry.

Table 5.39 Optimum sections and height for topology IV with variable geometry and fixed spacing.

Group Number	1 Upper chord members	2 Lower chord members	3 Vertical members	4 Diagonal members	Best Height (cm)	Weight (kg)
Designation	┘┘ 70x70x6	┘┘ 50x50x4	L45x45x4	L 90x90x8	140	546.38

The increase in member forces due to the height 140cm was balanced with the shortening of members which kept the change within the margin of the carrying capacities of the sections selected for the fixed geometry. Except for the third group where the single section angle 45x45x4 was selected. This was due to the decrease in the lengths of the members, which required a smaller section to resist the maximum compression force that has developed in member 13 with the length of 155cm. The maximum deflection for this truss topology was 1.73cm.

Table 5.40 Member forces and capacities for the optimum design of the fourth topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I (kN)	D+W(kN)	D+I (kN)	D+W (kN)
Upper Chord Members	2	203.04	124.12	0.16	1.13	450.45	314.60
	6	"	96.11	-153.51	57.15	213.69	450.45
	10	"	"	-186.48	66.39	"	"
	14	"	"	-180.50	60.53	"	"
	18	"	"	-180.50	52.99	"	"
	22	"	"	-186.48	53.65	"	"
	26	"	"	-153.51	43.69	"	"
	30	"	124.12	0.16	0.80	450.45	314.60
Lower Chord Members	4	200.00	121.50	151.21	-58.08	215.60	71.82
	8	"	"	183.61	-64.83	"	"
	12	"	"	177.68	-56.73	"	"
	16	"	"	154.62	-41.99	"	"
	20	"	"	154.62	-41.99	"	"
	24	"	"	177.68	-46.34	"	"
	28	"	"	183.61	-48.66	"	"
	32	"	"	151.21	-40.53	"	"
Verticals	1	50.00	55.55	-9.19	4.58	73.86	96.80
	5	85.00	82.38	19.43	-4.88	96.80	55.13
	9	120.00	115.52	-4.59	6.28	36.16	96.80
	13	155.00	149.21	-21.91	14.00	24.23	96.80
	17	190.00	0.00	0.00	0.00	0.00	0.00
	21	155.00	149.21	-21.91	6.65	24.23	96.80
	25	120.00	115.52	-4.59	-2.68	36.16	100.76
	29	85.00	82.38	19.43	-5.70	96.80	55.13
	33	50.00	55.55	-9.19	2.63	73.86	96.80
Diagonals	3	217.31	104.36	-163.11	56.97	164.58	382.25
	7	233.24	112.01	-37.53	9.41	149.26	382.25
	11	253.03	121.51	7.98	-10.36	382.25	132.48
	15	275.86	132.48	31.74	-20.34	382.25	115.98
	19	275.86	132.48	31.74	-9.63	382.25	115.98
	23	253.03	106.33	7.98	4.22	382.25	382.25
	27	233.24	112.01	-37.53	11.02	149.26	382.25
	31	217.31	104.36	-163.11	43.72	164.58	382.25

5.4.5 Truss Topology V

Five independent seed values were used to carry out the optimization of truss topology V. The optimum design was achieved from seed values 1 and 5. The minimum weight was obtained after the first 120 generations as shown in Fig. 5.29.

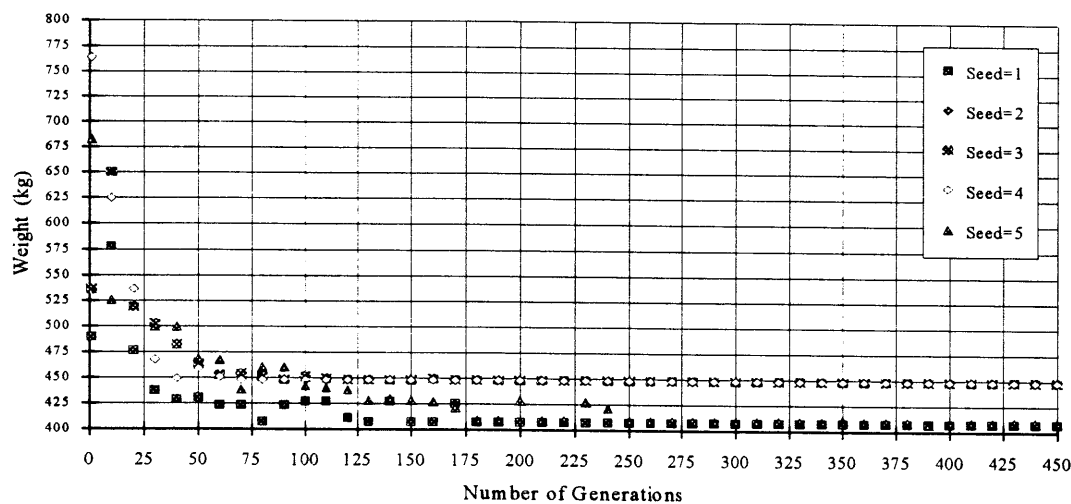


Fig. 5.29 Design history of the fifth topology with variable geometry and fixed spacing.

The optimum values for the design variables are listed in Table 5.41. Best height obtained for this truss topology was higher than that for fixed geometry which has resulted in a decrease in the member forces. Consequently, the selected sections were smaller for the different groups except for the diagonals where a slightly larger section was selected. This was due to increase in length of verticals without having a reduction in compression forces of the members in the group. Finally, the maximum deflection occurred in this truss topology was 1.49cm.

Table 5.41 Optimum sections for and height for topology V with variable geometry and fixed spacing.

Group Number	1 Upper chord members	2 Lower chord members	3 Vertical members	4 Diagonal members	Best Height (cm)	Weight (kg)
Designation	┐┐ 60x60x6	┐┐ 45x45x4	L50x50x4	L 60x60x6	175	408.01

Table 5.42 Member forces and capacities for the optimum design of the fifth topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I (kN)	D+W (kN)	D+I (kN)	D+W (kN)
Upper Chord Members	2	204.73	108.31	-140.13	49.46	156.45	382.25
	6	"	"	-140.36	52.53	"	"
	10	"	"	-155.81	50.46	"	"
	14	"	"	-155.93	53.49	"	"
	18	"	"	-155.93	46.95	"	"
	22	"	"	-155.81	44.78	"	"
	26	"	"	-140.36	40.33	"	"
	30	"	"	-140.13	38.14	"	"
Lower Chord Members	4	200.00	0.00	0.00	-6.61	0.00	0.00
	8	"	"	160.59	-57.15	193.60	58.94
	12	"	"	160.59	-57.15	"	"
	16	"	"	130.85	-35.15	"	"
	20	"	"	130.85	-35.15	"	"
	24	"	"	160.59	-41.73	"	"
	28	"	"	160.59	-41.73	"	"
	32	"	0.00	0.00	0.00	0.00	0.00
Verticals	1	50.00	52.88	-73.65	26.74	77.21	107.80
	5	93.75	81.49	-18.41	9.35	58.08	107.80
	9	137.50	0.00	0.00	0.00	0.00	0.00
	13	181.25	156.09	-18.40	9.35	24.34	107.80
	17	225.00	0.00	0.00	0.00	0.00	0.00
	21	181.25	156.09	-18.40	5.39	24.34	107.80
	25	137.50	0.00	0.00	0.00	0.00	0.00
	29	93.75	81.49	-18.41	5.39	58.08	107.80
	33	50.00	52.88	-73.66	19.75	77.21	107.80
Diagonals	3	206.16	148.50	140.67	-48.16	191.13	48.22
	7	242.71	174.83	-28.06	6.51	36.57	191.13
	11	242.71	174.83	-10.10	10.31	36.57	191.13
	15	301.04	216.85	32.26	-20.30	191.13	24.97
	19	301.04	216.85	32.26	-10.12	191.13	24.97
	23	242.71	174.83	-10.10	3.84	36.57	191.13
	27	242.71	174.83	-28.06	8.78	36.57	191.13
	31	206.16	148.50	140.67	-37.21	191.13	48.22

5.4.6 Truss Topology VI

For truss topology VI the program was run seven times until the same minimum was achieved. The best height and minimum weight for this truss topology were obtained with seed values 1 and 7 as shown in Fig.5.30. It can be noted that six of the seed values yielded very close results except for seed 6 which resulted in the heaviest design.

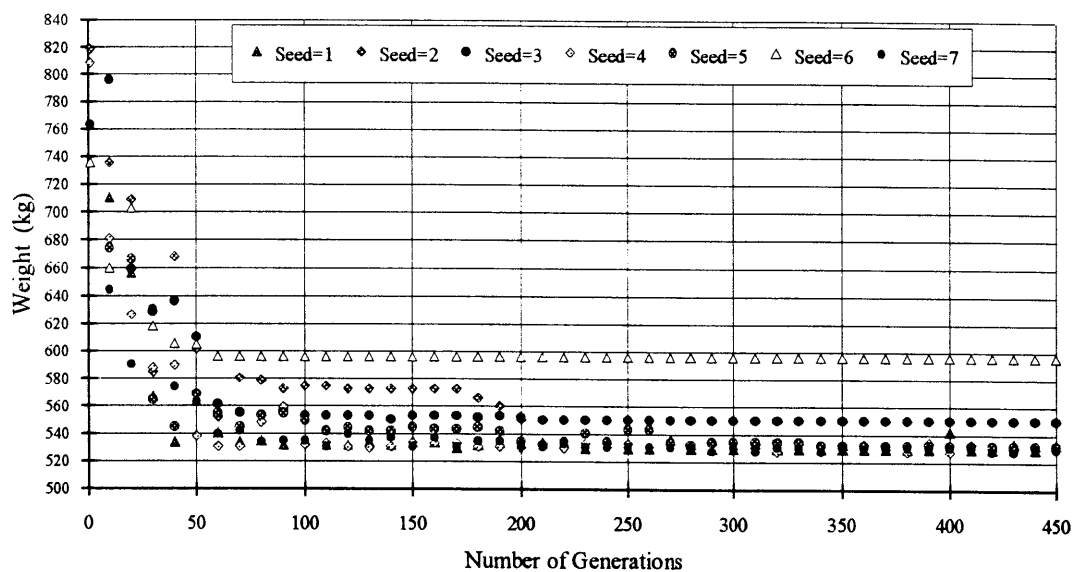


Fig. 5.30 Design history of the sixth topology with variable geometry and fixed spacing.

The sections selected for the best height 145cm were those selected for fixed geometry where height was 150cm. This height has led to a small reduction in the member lengths, at the same time small increase in the member forces. This increase in member forces was still in the margin of the capacities of the sections. This is why the algorithm selected the same sections. The maximum deflection of this truss topology was 1.75cm.

Table 5.43 Optimum sections and height for topology VI with variable geometry and fixed spacing.

Group Number	1 Upper chord members	2 Lower chord members	3 Vertical members	4 Diagonal members	Best Height (cm)	Weight (kg)
Designation	└L70x70x6	└L 45x45x4	└L 45x45x4	└L 90x90x8	145	529.68

Table 5.44 Member forces and capacities for the optimum design of the sixth topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I (kN)	D+W (kN)	D+I (kN)	D+W (kN)
Upper Chord Members	2	203.26	108.88	1.25	0.81	450.45	498.85
	6	"	96.18	-182.29	62.59	213.50	450.45
	10	"	"	-182.55	65.14	"	"
	14	"	"	-153.22	44.90	"	"
	18	"	"	-153.22	45.21	"	"
	22	"	"	-182.55	52.64	"	"
	26	"	"	-182.29	50.81	"	"
	30	"	108.88	1.25	0.54	450.45	498.85
Lower Chord Members	4	200.00	132.19	149.04	-57.38	193.60	58.94
	8	"	"	149.04	-57.38	"	"
	12	"	"	173.52	-55.41	"	"
	16	"	"	173.52	-55.41	"	"
	20	"	"	173.52	-45.08	"	"
	24	"	"	173.52	-45.08	"	"
	28	"	"	149.05	-39.88	"	"
	32	"	"	149.05	-39.88	"	"
Verticals	1	50.00	55.55	-9.19	4.59	73.86	96.80
	5	86.25	0.00	0.00	0.00	0.00	0.00
	9	122.50	117.92	-18.38	9.18	35.08	96.80
	13	158.75	0.00	0.00	0.00	0.00	0.00
	17	195.00	187.71	36.28	-11.06	96.80	16.36
	21	158.75	0.00	0.00	0.00	0.00	0.00
	25	122.50	117.92	-18.38	5.28	35.08	96.80
	29	86.25	0.00	0.00	0.00	0.00	0.00
	33	50.00	55.55	-9.19	2.64	73.86	96.80
Diagonals	3	217.81	104.60	-162.57	56.61	164.08	382.25
	7	217.81	104.60	33.70	-8.38	382.25	164.08
	11	255.35	122.62	8.67	-10.57	382.25	130.68
	15	255.35	122.62	-29.24	18.55	130.68	382.25
	19	255.35	122.62	-29.24	8.91	130.68	382.25
	23	255.35	128.42	8.68	3.96	382.25	121.76
	27	217.81	104.60	33.70	-9.98	382.25	164.08
	31	217.81	104.60	-162.57	43.50	164.08	382.25

5.4.7 Truss Topology VII

The design history for truss topology VII, is shown in Fig.5.31, where three different seed values were used for the three runs of the program. The minimum weight was obtained from seed value 2 which is 400.07kg with the optimum height of 185cm.

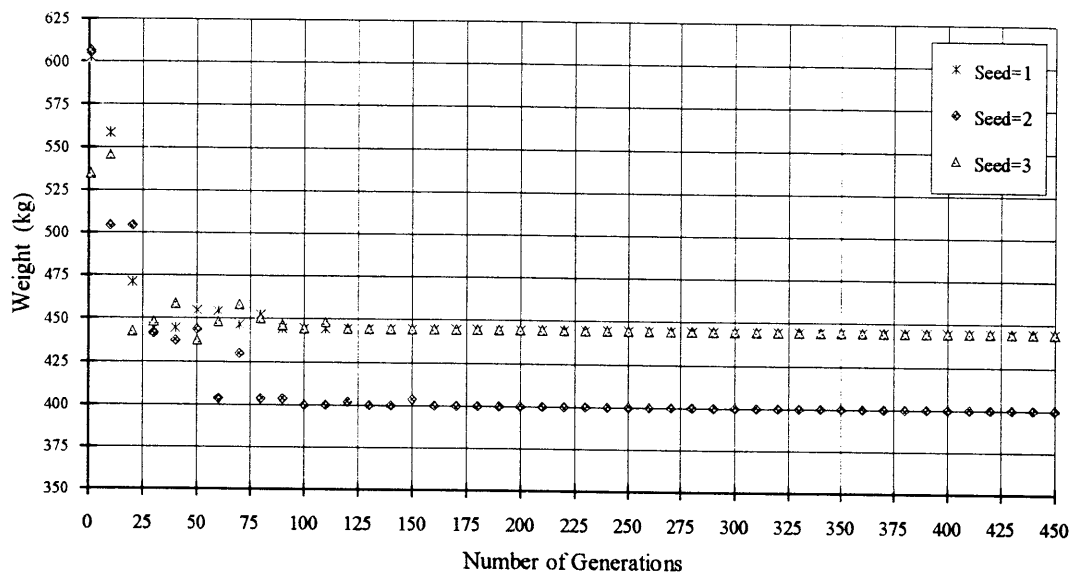


Fig. 5.31 Design history for the seventh topology with variable geometry and fixed spacing.

The minimum weight obtained in this case is slightly lighter than that obtained when geometry was fixed to height 150cm. Due to this increase in height, member forces had reduced for which smaller sections were selected for the different groups of members except for third group where same section was selected. The maximum deflection of this truss was 1.77cm.

Table 5.45 Optimum sections and height for topology VII with variable geometry and fixed spacing.

Group Number	1 Upper chord members	2 Lower chord members	3 Diagonal members	Best Height (cm)	Weight (kg)
Designation	┐┐ 60x60x6	┐┐ 45x45x4	L 60x60x5	185	400.07

Table 5.46 Member forces and capacities for the optimum design of the seventh topology.

Groups	M. No.	Length	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I	D+W	D+I	D+W
Upper Chord Members	2	205.28	108.52	-89.40	31.99	156.03	382.25
	6	"	"	-149.84	53.02	"	"
	10	"	"	-153.84	52.45	"	"
	14	"	"	-138.32	44.41	"	"
	18	"	"	-138.32	41.81	"	"
	22	"	"	-153.84	44.73	"	"
	26	"	"	-149.84	42.36	"	"
	30	"	"	-89.40	24.67	"	"
Lower Chord Members	4	100.00	0.00	0.00	-6.99	0.00	0.00
	8	200.00	132.19	133.92	-52.54	193.60	58.94
	12	"	"	155.07	-55.31	"	"
	16	"	"	146.34	-46.75	"	"
	20	"	"	125.37	-33.55	"	"
	24	"	"	146.34	-36.70	"	"
	28	"	"	155.07	-40.03	"	"
	32	"	"	133.92	-35.28	"	"
	34	100.00	0.00	0.00	0.00	0.00	0.00
Diagonals	1	50.00	49.02	-73.70	26.64	124.36	161.15
	3	111.80	81.32	98.17	-33.39	161.15	89.90
	5	138.80	99.98	-63.83	21.71	71.51	161.15
	7	138.80	99.98	17.71	-3.99	161.15	71.51
	9	174.09	125.40	-14.70	3.30	52.24	161.15
	11	174.09	125.40	-8.74	8.51	52.24	161.15
	13	213.60	153.87	8.10	-7.87	161.15	37.78
	15	213.60	153.87	-24.85	15.62	37.78	161.15
	17	255.39	183.97	23.84	-14.99	161.15	27.89
	19	255.39	183.97	23.84	-7.55	161.15	27.89
	21	213.60	153.87	-24.85	7.87	37.78	161.15
	23	213.60	153.87	8.10	-3.08	161.15	37.78
	25	174.09	125.40	-8.74	3.33	52.24	161.15
	27	174.09	125.40	-14.70	4.69	52.24	161.15
	29	138.80	99.98	17.71	-5.63	161.15	71.51
	31	138.80	99.98	-63.83	16.82	71.51	161.15
	33	111.80	81.32	98.17	-25.86	161.15	89.90
	35	50.00	49.02	-73.70	19.71	124.36	161.15

5.4.8 Summary for Design Examples with Variable Geometry and Fixed Spacing

The optimum sections obtained for the four groups of members as well as the optimum height for each of the seven different topologies are summarized in Table 5.47.

As with the previous two groups, trusses III, IV and VI here also produced the heaviest designs among the seven topologies. The excess weight these trusses produced was because of the large compression member forces that has developed in their diagonal members due to D+I load case. Consequently, larger sections were selected for those members that belong to the fourth group.

Similar to the previous two groups where geometry was fixed with both fixed and variable spacing, the lightest topology comes out to be the seventh one. Furthermore taking the geometry as variable the lightest topology here is still lighter than that obtained for same topology in the previous cases. Likewise, the optimum weight obtained for all topologies in this case are lighter than those obtained for fixed geometry.

The percentage weight saving due to variation of geometry come out to be 6.5% from that for fixed geometry and fixed spacing.

Table 5.47 Summary of the optimum designs of the seven truss topologies with variable geometry and fixed truss spacing.

TRUSS TOPOLOGIES	Upper Chord Double Section	Lower Chord Double Section	Verticals Single Section	Diagonals Single Section	Optimum Height (cm)	Weight (kg)
	60x60x6	60x60x6	50x50x4	60x60x6	185	481
	70x70x6	50x50x4	45x45x4	70x70x6	120	443
	70x70x6	60x60x6	45x45x4	90x90x8	140	600
	70x70x6	50x50x4	45x45x4	90x90x8	140	546
	60x60x6	45x45x4	50x50x4	60x60x6	175	408
	70x70x6	45x45x4	45x45x4	90x90x8	145	530
	70x70x6	50x50x4	-	60x60x5	185	400

5.5 DESIGN EXAMPLES WITH VARIABLE GEOMETRY AND VARIABLE TRUSS SPACING

As a final group of examples, the very same set of trusses were designed by considering the geometry as well as the spacing as design parameters. In this case the number of design variables in each truss increases to six, each of which is represented with 5-bit string length. The same values that were selected in the previous sections for the height of apex and the spacing were also used in this group of design examples.

5.5.1 Truss Topology I

The first truss topology was designed four times in each having the seed values of 1,2,3 and 4. The optimum design was obtained from seed 3 as shown in Fig. 5.32.

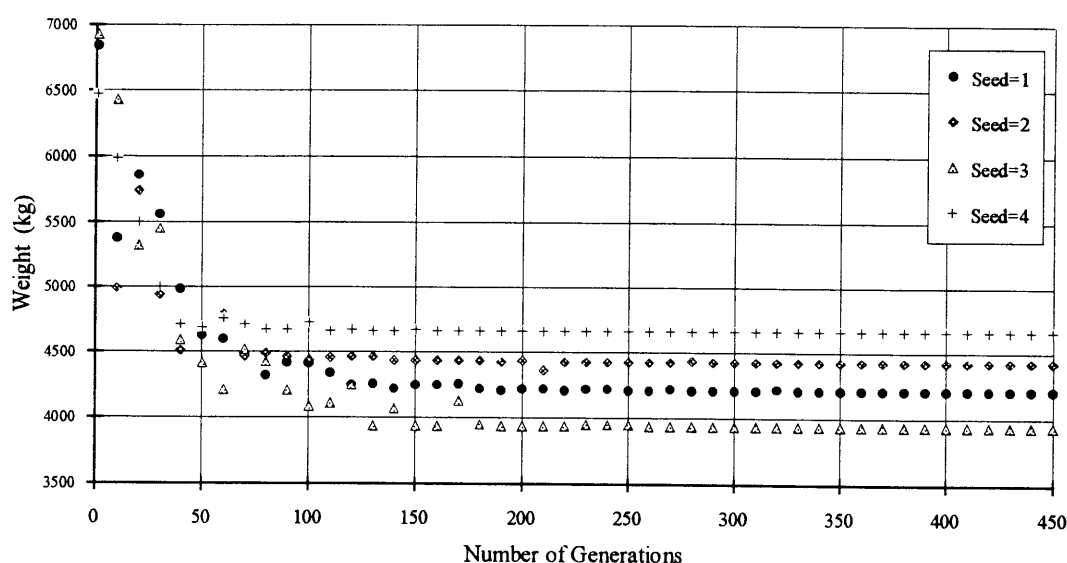


Fig. 5.32 Design history of the first topology with variable geometry and variable truss Spacing.

With the spacing being variable, the best height for truss topology I came out to be 170cm as shown in Table 5.48. When compared with the previous group, the overall weight which is 3930.81kg in this case is lighter compared to 4329kg. Although the weight of one truss has increased, but in overall weight of the roof has decreased.

Table 5.48 Optimum sections, height and spacing for topology I with variable geometry and variable spacing.

Group Number	1	2	3	4	Best Height (cm)	Best Truss Spacing (cm)	Weight (kg)	
	Upper chord members	Lower chord members	Vertical members	Diagonal members			One Truss	All Trusses
Designation	∟70x70x6	∟70x70x6	∟50x50x5	∟70x70x6	170	650	562.83	3939.81

Table 5.49 Member forces and capacities for the optimum design of the first topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I (kN)	D+W (kN)	D+I (kN)	D+W (kN)
Upper Chord Members	2	204.47	96.57	-184.62	65.27	212.43	450.45
	6	"	"	-184.87	69.12	"	"
	10	"	"	-206.59	66.86	"	"
	14	"	"	-206.72	70.67	"	"
	16	"	"	-206.72	62.01	"	"
	18	"	"	-206.59	59.28	"	"
	22	"	"	-184.87	53.04	"	"
	26	"	"	-184.62	50.28	"	"
Lower Chord Members	4	200.00	0.00	0.00	-8.35	0.00	0.00
	8	400.00	160.23	212.61	-75.57	450.45	99.62
	12	"	"	173.95	-46.81	"	"
	20	"	"	212.61	-55.42	"	"
	24	200.00	0.00	0.00	0.00	0.00	0.00
Verticals	1	50.00	53.03	-95.71	34.82	103.68	132.82
	5	92.50	81.14	-23.92	12.11	76.81	"
	11	177.50	154.11	-23.90	12.10	31.51	"
	17	177.50	154.11	-23.90	6.97	31.51	"
	23	92.50	81.14	-23.92	6.98	76.81	"
	27	50.00	53.03	-95.71	25.69	103.68	"
Diagonals	3	206.16	126.98	185.78	-63.80	225.22	72.55
	7	241.30	148.63	-38.28	9.01	56.46	225.22
	9	241.30	148.63	-12.56	13.25	56.46	225.22
	13	297.32	183.13	41.82	-26.37	225.22	39.59
	15	297.32	183.13	41.82	-13.05	225.22	39.59
	19	241.30	148.63	-12.56	4.79	56.46	225.22
	21	241.30	148.63	-38.28	11.85	56.46	225.22
	25	206.16	126.98	185.78	-49.24	225.22	72.55

The sections selected by the algorithm here were larger, in order to resist the increase in member forces that developed due to the increase in loading per truss due to the new spacing.

5.5.2 Truss Topology II

For the second topology, six different seed values were used as shown in Fig. 5.33. The six seed values obtained close results to the optimum design which was achieved from seed value 5.

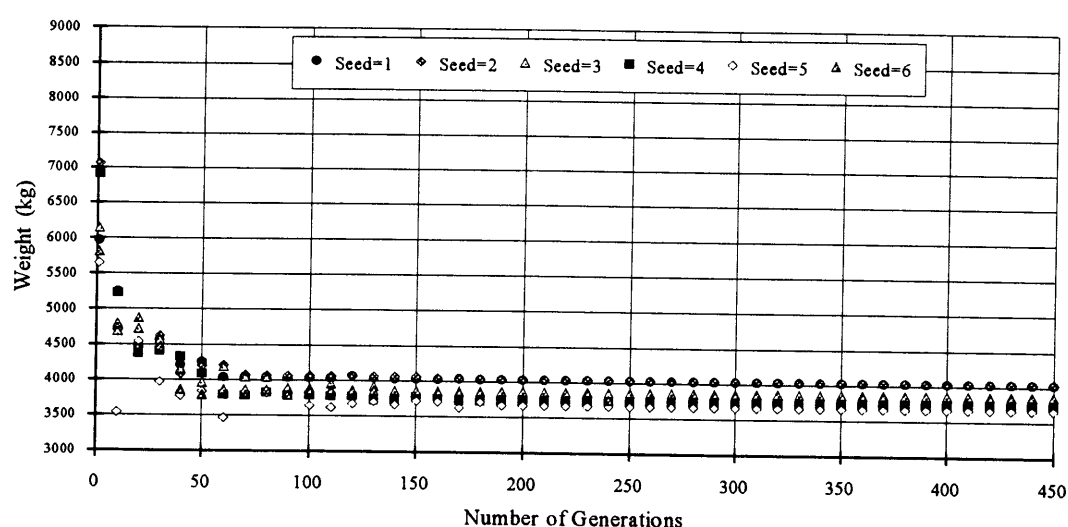


Fig. 5.33 Design history of the second topology with variable geometry and variable truss Spacing.

The best height obtained for this truss topology was 170cm same as truss I. Whereas best truss spacing here was 660cm as given in the Table 5.50 below.

Table 5.50 Optimum sections, height and spacing for topology II with variable geometry and variable spacing.

Group Number	1	2	3	4	Best Height (cm)	Best Truss Spacing (cm)	Weight (kg)	
	Upper chord members	Lower chord members	Vertical members	Diagonal members			One Truss	All Trusses
Designation	L80x80x6	L50x50x5	L50x50x5	L70x70x6	170	660	522.70	3658.89

Table 5.51 Member forces and capacities for the optimum design of the second topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to D+I (kN) D+W (kN)		Member Capacities due to D+I (kN) D+W (kN)	
Upper Chord Members	2	204.47	88.42	-188.04	66.47	243.66	514.80
	6	"	"	-220.79	75.53	"	"
	10	"	"	-209.87	67.90	"	"
	14	"	"	-180.58	53.81	"	"
	18	"	"	-180.58	54.36	"	"
	22	"	"	-209.87	60.22	"	"
	26	"	"	-220.78	61.64	"	"
	30	"	"	-188.04	51.22	"	"
Lower Chord Members	4	200.00	0.00	0.00	-8.48	0.00	0.00
	8	"	122.11	183.87	-71.61	265.65	91.40
	12	"	"	215.97	-76.77	"	"
	16	"	"	205.32	-65.59	"	"
	20	"	"	205.32	-52.24	"	"
	24	"	"	215.97	-56.29	"	"
	28	"	"	183.87	-48.74	"	"
	32	"	0.00	0.00	0.00	0.00	0.00
Verticals	1	50.00	53.03	-97.19	35.35	103.68	132.82
	5	92.50	81.14	-45.96	15.78	76.81	132.82
	9	135.00	117.21	-14.84	3.49	48.57	132.82
	13	177.50	154.11	7.18	-7.53	132.82	31.51
	17	220.00	191.01	50.80	-15.86	132.82	21.77
	21	177.50	154.11	7.18	-2.73	132.82	31.51
	25	135.00	117.21	-14.84	4.60	48.57	132.82
	29	92.50	81.14	-45.95	12.18	76.81	132.82
	33	50.00	53.03	-97.19	26.09	103.68	132.82
Diagonals	3	206.16	126.98	188.49	-64.73	225.22	72.55
	7	220.35	135.73	34.72	-8.14	225.22	65.39
	11	241.30	148.63	-12.74	13.45	56.46	225.22
	15	267.41	164.71	-38.23	24.08	47.54	225.22
	19	267.41	164.71	-38.23	11.93	47.54	225.22
	23	241.30	148.63	-12.74	4.86	56.46	225.22
	27	220.35	135.73	34.72	-10.78	225.22	65.39
	31	206.16	126.98	188.49	-49.96	225.22	72.55

Due to the increase in spacing the sections selected herein were larger except for the fourth group where same section was selected. This is because the increase in the maximum compression force of this group was in the margin of the compression resistance of the section that was selected when spacing was fixed to 500cm.

5.5.3 Truss Topology III

The history of generations for the design of truss topology III with five different seed values is shown in Fig. 5.34. Seed values 2,3, and 5 obtained the same optimum weight of 3658.89kg.

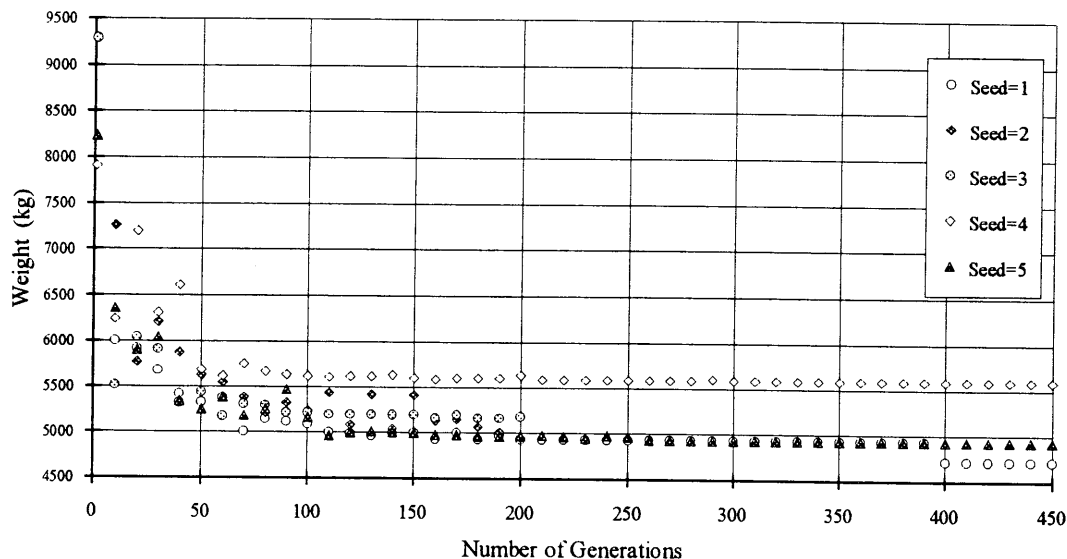


Fig. 5.34 Design history of the third topology with variable geometry and variable truss Spacing.

The optimum apex height selected by the algorithm for this truss topology was 180cm for the best truss spacing of 650cm as shown in Table 5.52. The overall weight of the intermediate trusses here was lighter than that obtained for same topology with fixed spacing where the overall weight was 5402.61kg.

Table 5.52 Optimum sections, height and spacing for topology III with variable geometry and variable spacing.

Group Number	1	2	3	4	Best Height (cm)	Best Truss Spacing (cm)	Weight (kg)	
	Upper chord members	Lower chord members	Vertical members	Diagonal members			One Truss	All Trusses
Designation	∟ 70x70x6	∟ 70x70x6	∟ 50x50x4	∟ 100x100x8	180	650	673.09	4711.65

Table 5.53 Member forces and capacities for the optimum design of the third topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I (kN)	D+W (kN)	D+I (kN)	D+W (kN)
Upper Chord Members	2	205.00	88.42	0.82	1.72	450.45	514.80
	6	"	96.74	-209.94	71.74	211.96	450.45
	10	"	"	-210.23	75.84	"	"
	14	"	"	-170.80	51.25	"	"
	16	"	"	-170.80	51.79	"	"
	18	"	"	-210.23	61.59	"	"
	22	"	"	-209.94	58.65	"	"
	24	"	88.42	0.82	1.20	450.45	514.80
Lower Chord Members	4	400.00	160.23	176.37	-69.02	450.45	99.62
	8	"	"	194.07	-62.00	"	"
	12	"	"	194.07	-48.92	"	"
	20	"	"	176.37	-46.56	"	"
Verticals	1	50.00	52.88	-11.97	6.10	77.21	107.80
	7	140.00	120.57	-23.95	12.20	36.31	107.80
	13	230.00	198.07	50.97	-16.07	107.80	16.22
	19	140.00	120.57	-23.95	7.04	36.31	107.80
	25	50.00	52.88	-11.97	3.52	77.21	107.80
Diagonals	3	221.42	95.53	-194.67	66.43	195.79	429.00
	5	221.42	95.53	31.56	-7.20	429.00	195.79
	9	272.44	117.55	14.90	-15.07	429.00	149.82
	11	272.44	117.55	-37.49	23.65	149.82	429.00
	15	272.44	117.55	-37.49	11.82	149.82	429.00
	17	272.44	117.55	14.89	-5.70	429.00	149.82
	21	221.42	95.53	31.56	-9.97	429.00	195.79
	23	221.42	95.53	-194.67	51.39	195.79	429.00

Due to the increase in spacing the sections selected for the different groups of members for this case were larger than those obtained for same topology with variable height and fixed spacing. Except for the first group where same section was still capable of resisting the increase in the compression force developed in member 10 of this group.

5.5.4 Truss Topology IV

The history of generations for the fourth truss topology is shown in Fig.5.35 where five different seed values were used. All of the seed values obtained close results and the optimum design was reached from the seed value of 4.

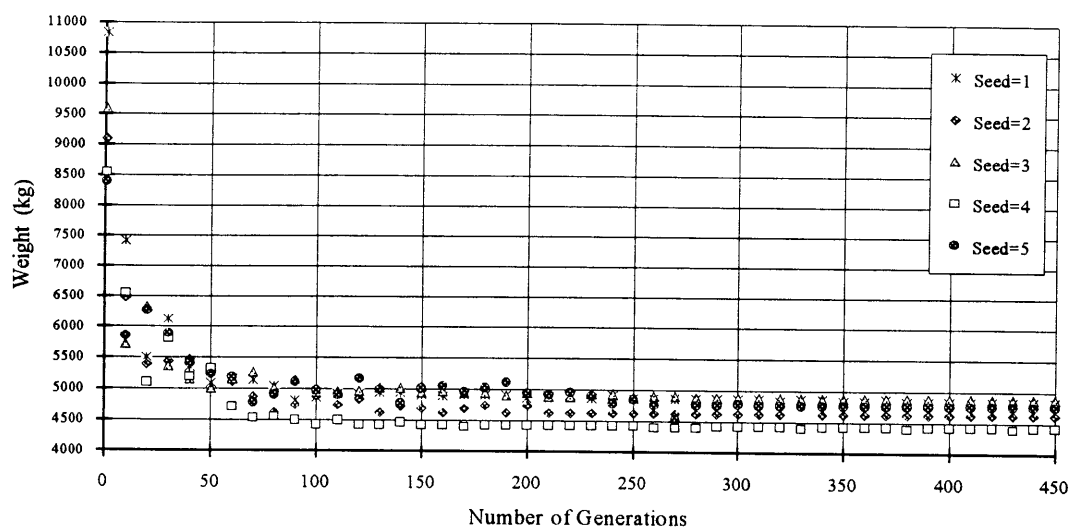


Fig. 5.35 Design history for the fourth topology with variable geometry and variable truss Spacing.

The overall weight obtained for this topology was lighter than that obtained for same topology with fixed spacing where the overall weight there was 4917.42kg obtained for height 140cm. Here the best height was 190cm with best spacing of 640cm. There is an increase in height as well as in truss spacing. The increase in truss spacing led to increase in load carried by each truss and at the same time reduction in the total number of trusses required to cover the whole length, and on the other hand the increase in the height led to reduction in member forces. All these together led to overall reduction of the total weight of this truss topology.

Table 5.54 Optimum sections, height and spacing for topology IV with variable geometry and variable spacing.

Group Number	1	2	3	4	Best Height (cm)	Best Truss Spacing (cm)	Weight (kg)	
	Upper chord members	Lower chord members	Vertical members	Diagonal members			One Truss	All Trusses
Designation	∟L70x70x6	∟L50x50x4	∟L60x60x6	∟L100x100x8	190	640	631.87	4423.07

The increase in the loading did not affect the selection of the sections for the first and second groups of members where the same sections were still sufficient to provide resistance to resist the increase in the compression forces. On the other hand for the third and fourth groups, the increase in height and loading caused an increase in the lengths of the members of these groups as well as their member forces. Consequently larger sections were selected in order to provide the required resistance in this case.

Table 5.55 Member forces and capacities for the optimum design of the fourth topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to D+I (kN) D+W (kN)		Member Capacities due to D+I (kN) D+W (kN)	
Upper Chord Members	2	205.56	110.51	0.77	1.82	450.45	610.5
	6	"	96.93	-173.67	65.12	211.47	450.45
	10	"	"	-200.49	72.57	"	"
	14	"	"	-188.94	65.49	"	"
	18	"	"	-188.94	57.56	"	"
	22	"	"	-200.49	59.02	"	"
	26	"	"	-173.67	50.11	"	"
	30	"	110.51	0.77	1.28	450.45	610.5
Lower Chord Members	4	200.00	121.5	169.30	-66.57	215.60	71.82
	8	"	"	195.15	-69.69	"	"
	12	"	"	183.75	-58.70	"	"
	16	"	"	157.20	-41.99	"	"
	20	"	"	157.20	-41.99	"	"
	24	"	"	183.74	-45.85	"	"
	28	"	"	195.15	-50.21	"	"
	32	"	"	169.30	-44.50	"	"
Verticals	1	50.00	49.13	-11.80	6.05	154.24	191.13
	5	97.50	72.84	18.75	-4.14	191.13	121.95
	9	145.00	104.45	-10.98	10.57	82.20	191.13
	13	192.5	138.67	-31.87	20.07	53.92	191.13
	17	240.00	0	0	0	0	0
	21	192.50	138.67	-31.87	10.15	53.92	191.13
	25	145.00	104.45	-10.98	4.20	82.20	191.13
	29	97.50	72.84	18.75	-6.03	191.13	121.95
	33	50.00	49.13	-11.80	3.49	154.24	191.13
Diagonals	3	222.50	96.00	-187.62	63.60	194.68	429.00
	7	247.03	106.59	-31.07	6.83	171.13	429.00
	11	277.59	119.77	16.3	-15.35	429	145.88
	15	312.41	134.80	41.47	-26.11	429.00	122.36
	19	312.41	134.80	41.47	-13.21	429.00	122.36
	23	277.59	119.77	16.30	-6.18	429.00	145.88
	27	247.03	106.59	-31.07	10.04	171.13	429.00
	31	222.5	96.00	-187.62	49.31	194.68	429.00

5.5.5 Truss Topology V

Five independent seed values were used to carry out the optimization of the fifth truss topology. The optimum design was obtained from seed value 4 as shown in Fig.5.36.

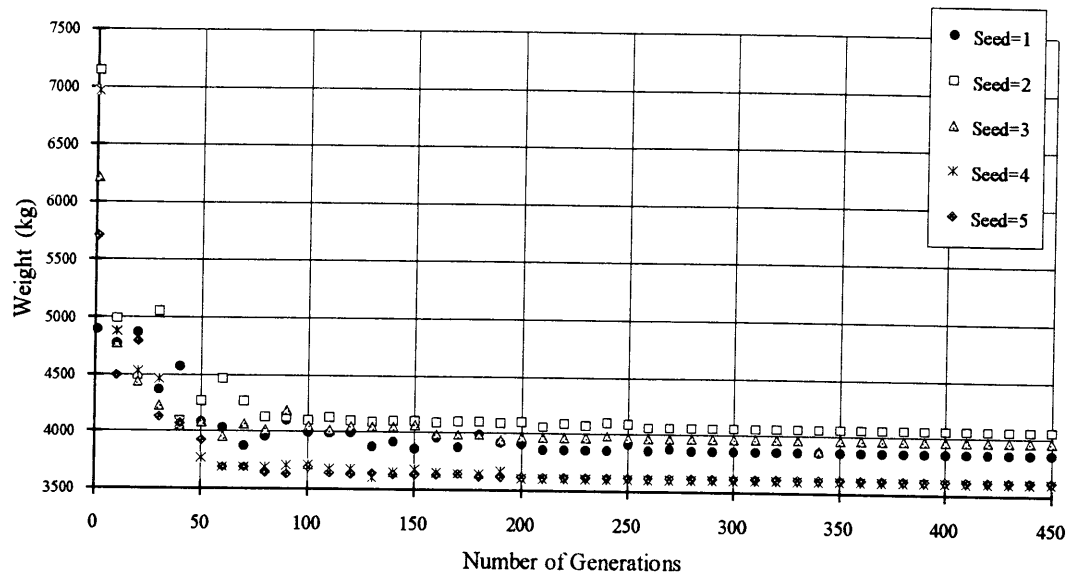


Fig. 5.36 Design history for the fifth topology with variable geometry and variable truss Spacing.

For this truss topology, the overall weight of trusses obtained for best height of 130cm and truss spacing of 640cm did not produce a significantly lighter trusses as compared to same topology with variable height and fixed spacing. In this case the increase in spacing has reduced the total number of trusses to 7, as compared to 9 which had an overall weight of 3672.09kg.

Table 5.56 Optimum sections, height and spacing for topology V with variable geometry and variable spacing.

Group Number	1	2	3	4	Best Height (cm)	Best Truss Spacing (cm)	Weight (kg)	
	Upper chord members	Lower chord members	Vertical members	Diagonal members			One Truss	All Trusses
Designation	└┐L80x80x6	└┐L50x50x5	L50x50x5	L70x70x6	130	640	514.68	3602.79

Table 5.57 Member forces and capacities for the optimum design of the fifth topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to D+I (kN) D+W (kN)		Member Capacities due to D+I (kN) D+W (kN)	
Upper Chord Members	2	202.62	87.89	-200.17	71.67	245.10	514.80
	6	202.62	87.89	-200.87	74.75	245.10	514.80
	10	202.62	87.89	-241.68	77.82	245.10	514.80
	14	202.62	87.89	-242.05	80.77	245.10	514.80
	18	202.62	87.89	-242.05	70.60	245.10	514.80
	22	202.62	87.89	-241.68	68.48	245.10	514.80
	26	202.62	87.89	-200.87	57.02	245.10	514.80
	30	202.62	87.89	-200.17	54.82	245.10	514.80
Lower Chord Members	4	200.00	0.00	0.00	-6.29	0.00	0.00
	8	200.00	122.11	245.24	-86.41	265.65	91.40
	12	200.00	122.11	245.24	-86.41	265.65	91.40
	16	200.00	122.11	208.91	-56.87	265.65	91.40
	20	200.00	122.11	208.91	-56.87	265.65	91.40
	24	200.00	122.11	245.24	-65.31	265.65	91.40
	28	200.00	122.11	245.24	-65.31	265.65	91.40
	32	200.00	0.00	0.00	0.00	0.00	0.00
Verticals	1	50.00	53.03	-94.01	34.69	103.68	132.82
	5	82.50	73.99	-23.50	11.65	83.64	132.82
	9	115.00	0.00	0.00	0.00	0.00	0.00
	13	147.50	128.06	-23.50	11.65	42.49	132.82
	17	180.00	0.00	0.00	0.00	0.00	0.00
	21	147.50	128.06	-23.50	6.69	42.49	132.82
	25	115.00	0.00	0.00	0.00	0.00	0.00
	29	82.50	73.99	-23.50	6.69	83.64	132.82
	33	50.00	53.03	-94.01	25.46	103.68	132.82
Diagonals	3	206.16	126.98	204.27	-71.70	225.22	72.55
	7	230.71	142.10	-52.39	13.35	60.75	225.22
	11	230.71	142.10	-7.33	11.69	60.75	225.22
	15	269.07	165.73	40.39	-26.10	225.22	47.04
	19	269.07	165.73	40.39	-12.15	225.22	47.04
	23	230.71	142.10	-7.33	-5.77	60.75	225.22
	27	230.71	142.10	-52.39	15.12	60.75	225.22
	31	206.16	126.98	204.27	-54.93	225.22	72.55

The increase in spacing and decrease in height has led to increase in the member forces in this case for which larger sections were selected for all the groups of members in this truss topology, as given in Table 5.57.

5.5.6 Truss Topology VI

For truss topology VI, three independent seed values were used to carry out the optimization process. The three of the seeds obtained very close results to the optimum design which was obtained from seed value 2.

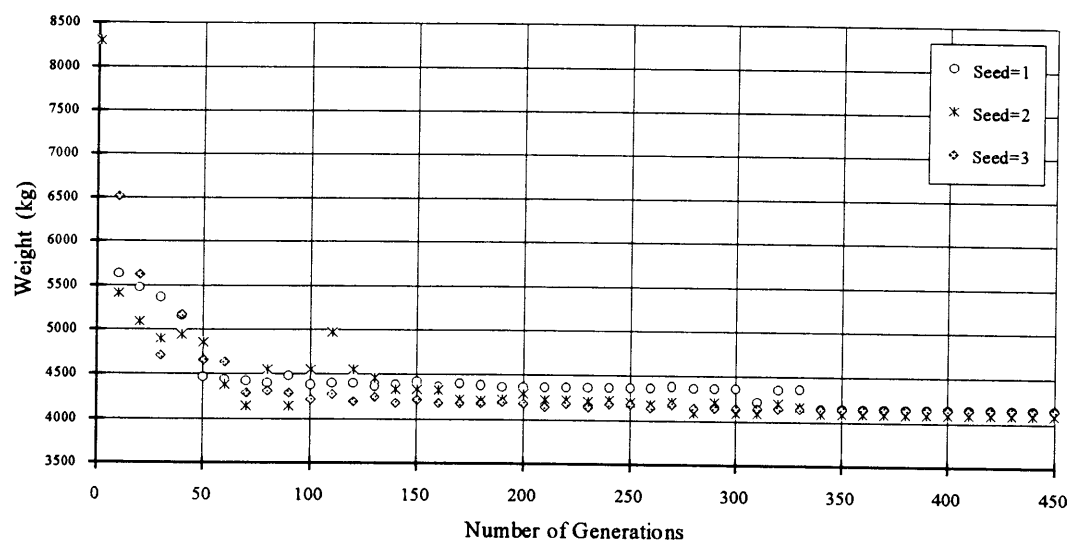


Fig. 5.37 Design history for the sixth topology with variable geometry and variable truss Spacing.

The best height obtained for truss topology VI was 180cm with the best spacing of 650cm. These values have produced an overall lighter design than that obtained for same topology with variable height and fixed spacing, where the overall weight was 4717.87kg obtained for best height of 145cm.

Table 5.58 Optimum sections, height and spacing for topology VI with variable geometry and variable spacing.

Group Number	1	2	3	4	Best Height (cm)	Best Truss Spacing (cm)	Weight (kg)	
	Upper chord members	Lower chord members	Vertical members	Diagonal members			One Truss	All Trusses
Designation	┐┐70x70x6	┐┐50x50x4	┐ 50x50x4	┐100x100x8	180	650	583.99	4087.92

Table 5.59 Member forces and capacities for the optimum design of the sixth topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I (kN)	D+W (kN)	D+I (kN)	D+W (kN)
Upper Chord Members	2	205.00	123.79	1.24	1.58	450.45	215.60
	6	"	96.74	-209.80	71.70	211.96	450.45
	10	"	"	-210.08	75.80	"	"
	14	"	"	-170.80	51.25	"	"
	18	"	"	-170.80	51.78	"	"
	22	"	"	-210.08	61.55	"	"
	26	"	"	-209.80	58.62	"	"
	30	"	123.79	1.24	1.10	450.45	215.60
Lower Chord Members	4	200.00	121.50	176.34	-69.01	215.60	71.82
	8	"	"	176.34	-69.01	"	"
	12	"	"	194.04	-61.99	"	"
	16	"	"	194.04	-61.99	"	"
	20	"	"	194.04	-48.91	"	"
	24	"	"	194.04	-48.91	"	"
	28	"	"	176.34	-46.55	"	"
	32	"	"	176.34	-46.55	"	"
Verticals	1	50.00	52.88	-11.97	6.10	77.21	107.80
	5	95.00	0.00	0.00	0.00	0.00	0.00
	9	140.00	120.57	-23.95	12.20	36.31	107.80
	13	185.00	0.00	0.00	0.00	0.00	0.00
	17	230.00	198.07	50.97	-16.07	107.80	16.22
	21	185.00	0.00	0.00	0.00	0.00	0.00
	25	140.00	120.57	-23.95	7.04	36.31	107.80
	29	95.00	0.00	0.00	0.00	0.00	0.00
	33	50.00	52.88	-11.97	3.52	77.21	107.80
Diagonals	3	221.42	95.53	-194.37	66.33	195.79	429.00
	7	221.42	95.53	31.45	-7.17	429.00	195.79
	11	272.44	117.55	14.74	-15.03	429.00	149.82
	15	272.44	117.55	-37.49	23.66	149.82	429.00
	19	272.44	117.55	-37.49	11.82	149.82	429.00
	23	272.44	117.55	14.74	-5.66	429.00	149.82
	27	221.42	95.53	31.45	-9.94	429.00	195.79
	31	221.42	95.53	-194.37	51.31	195.79	429.00

The increase in the spacing resulted in an increase in the loading carried by each truss. Consequently, the sections selected here were larger, except for first group

where same section was still sufficient to provide the required resistance for compression force in this group. On the other hand, the length of members in the third and fourth groups is affected by the increase in the height which caused the algorithm to select larger sections.

5.5.7 Truss Topology VII

The history of generations for the seventh truss topology is shown in Fig.5.38. Five different seed values were used and the optimum design is obtained with the seed value of 5.

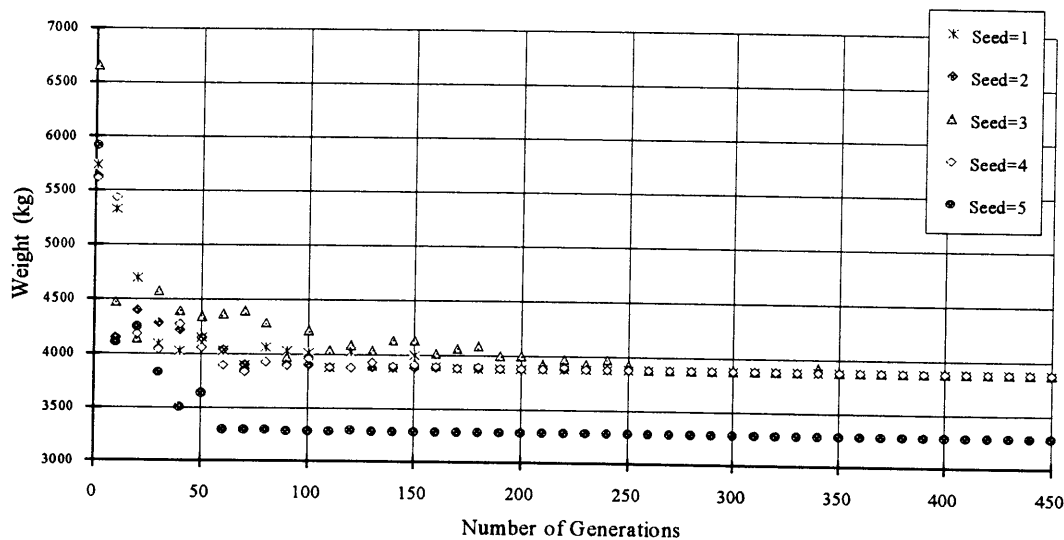


Fig. 5.38 Design history for the seventh topology with variable geometry and variable truss Spacing.

As with all previous trusses, the over all weight of trusses of topology VII was lighter than that obtained for same topology with variable height and fixed spacing where overall weight was 3600.63kg.

Table 5.60 Optimum sections, height and spacing for topology VII with variable geometry and variable spacing.

Group Number	1	2	3	Best Height (cm)	Best Truss Spacing (cm)	Weight (kg)	
	Upper chord members	Lower chord members	Diagonal members			One Truss	All Trusses
Designation	┐┐70x70x6	┐┐50x50x4	┐60x60x6	190	650	468.98	3282.85

Table 5.61 Member forces and capacities for the optimum design of the seventh topology.

Groups	M. No.	Length (cm)	Slender-ness	Member Forces due to		Member Capacities due to	
				D+I (kN)	D+W (kN)	D+I (kN)	D+W (kN)
Upper Chord Members	2	205.56	96.93	-116.15	41.48	211.47	450.45
	6	"	"	-192.54	68.09	"	"
	10	"	"	-196.84	67.16	"	"
	14	"	"	-176.52	56.83	"	"
	18	"	"	-176.52	53.56	"	"
	22	"	"	-196.84	57.35	"	"
	26	"	"	-192.54	54.46	"	"
	30	"	"	-116.15	32.04	"	"
Lower Chord Members	4	100.00	0.00	0.00	-9.34	0.00	0.00
	8	200.00	121.50	171.94	-67.61	215.60	71.82
	12	"	"	198.20	-70.78	"	"
	16	"	"	186.62	-59.62	"	"
	20	"	"	159.66	-42.65	"	"
	24	"	"	186.62	-46.57	"	"
	28	"	"	198.20	-51.00	"	"
	32	"	"	171.94	-45.19	"	"
Diagonals	34	100.00	0.00	0.00	0.00	0.00	0.00
	1	50.00	49.13	-95.85	34.57	154.24	191.13
	3	111.80	81.32	126.58	-42.91	191.13	110.27
	5	139.66	100.61	-81.64	27.67	86.36	191.13
	7	139.66	100.61	22.08	-4.89	191.13	86.36
	9	176.14	126.88	-18.32	4.04	62.02	191.13
	11	176.14	126.88	-11.80	11.25	62.02	191.13
	13	216.92	156.26	10.64	-10.36	191.13	44.30
	15	216.92	156.26	-32.36	20.41	44.30	191.13
	17	260.00	187.29	31.12	-19.64	191.13	32.42
	19	260.00	187.29	31.12	-9.92	191.13	32.42
	21	216.92	156.26	-32.36	10.31	44.30	191.13
	23	216.92	156.26	10.64	-4.09	191.13	44.30
	25	176.14	126.88	-11.80	4.49	62.02	191.13
	27	176.14	126.88	-18.32	5.91	62.02	191.13
	29	139.66	100.61	22.08	-7.09	191.13	86.36
	31	139.66	100.61	-81.64	21.46	86.36	191.13
	33	111.80	81.32	126.58	-33.27	191.13	110.27
	35	50.00	49.13	-95.85	25.59	154.24	191.13

The sections selected for this topology were larger than those selected for same topology with fixed spacing. This was an obvious result due to increase in loading carried by each truss.

5.5.8 Summary for Design Examples with Variable Geometry and Variable Truss Spacing

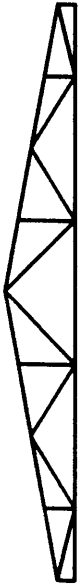
Table 5.62 lists the summary of the different design parameters considered in this group of problems.

It is interesting to notice that the truss topology has little effect on the optimum spacing when the height of apex is taken as variable. In all cases the optimum spacing is around 650cm. The topology VII is still the lightest among all.

The largest section selected was that for members of group number 4 (diagonals) in truss topologies III, IV and VI. Note that for these trusses, the maximum compression force developed in member number 3, due to D+I load case, which has governed the selection of the section 100x100x8.

In the present case, where geometry and truss spacing were taken as design parameters, 15% of saving in weight was achieved for the optimum topology VII.

Table 5.62 Summary of the optimum designs of the seven truss topologies with variable geometry and variable truss spacing.

TRUSS TOPOLOGIES	Upper Chord Double Section	Lower Chord Double Section	Verticals Single Section	Diagonals Single Section	Optimum Height (cm)	Optimum Truss Spacing (cm)	Weight of One Truss (kg)	No. of Trusses	Over all Weight (kg)
	70x70x6	70x70x6	50x50x5	70x70x6	170	650	562.83	7	3940
	80x80x6	50x50x5	50x50x5	70x70x6	170	660	522.70	7	3659
	70x70x6	70x70x6	50x50x4	100x100x8	180	650	673.09	7	4712
	70x70x6	50x50x4	60x60x6	100x100x8	190	640	631.87	7	4423
	80x80x6	50x50x5	50x50x5	70x70x6	130	640	514.68	7	3603
	70x70x6	50x50x4	50x50x4	100x100x8	180	650	583.99	7	4088
	70x70x6	50x50x4	-	60x60x6	190	650	468.98	7	3283

5.6 SUMMARY OF ALL DESIGN EXAMPLES

The optimum designs obtained in four different cases are summarized in Table 5.63. The heaviest design topology in all design problems was topology III, while the lightest was topology VII. The maximum percentage of weight saving was achieved in topology I and minimum percentage saving in weight was achieved in topology II. The savings achieved when the geometry and spacing of the roof truss were taken as design variables are given in the last column of the table. The least reduction in the weight is not less than 10% for the heaviest topology. This reduction increases to 15% in the lightest topology.

It is apparent that in order to achieve the lightest saving in the weight of a roof truss the geometry and the spacing between trusses are required to be made design variables. It is also shown that the genetic algorithm is capable of achieving this while the design code requirements are to be satisfied.

Table 5.63 Summary of the optimum designs for all examples.

Truss Topology	Fixed Geometry and Fixed Truss Spacing			Fixed Geometry and Variable Truss Spacing			Variable Geometry and Fixed Truss Spacing			Variable Geometry and Variable Truss Spacing			Percentage Saving (%)
	Height (cm)	Truss Spacing (cm)	Total Weight (kg)	Height (cm)	Truss Spacing (cm)	Total Weight (kg)	Height (cm)	Truss Spacing (cm)	Total Weight (kg)	Height (cm)	Truss Spacing (cm)	Total Weight (kg)	
Truss I	150	500	4682	150	680	4118	185	500	4329	170	650	3940	16
Truss II	"	"	4064	"	560	3666	120	"	3987	170	660	3659	10
Truss III	"	"	5434	"	640	5207	140	"	5403	180	650	4712	13
Truss IV	"	"	4992	"	560	4644	140	"	4917	190	640	4423	11
Truss V	"	"	4113	"	680	3653	175	"	3672	130	640	3603	12
Truss VI	"	"	4785	"	660	4718	145	"	4767	180	650	4088	15
Truss VII	"	"	3864	"	560	3618	185	"	3600	190	650	3283	15

6.1 SUMMARY

In this study, an algorithm is developed for the optimum design of steel roof trusses which implements the design requirements as specified in BS5950. The serviceability and strength limit states are considered as constraints in the formulation of the design problem. The genetic algorithm is used in finding the solution of the design problem where the cross-sections of members are to be selected from practically available discrete set of standard steel sections. Both of these features make the algorithm very suitable to be used by the practicing every day designers without carrying out any modifications. The algorithm presented has three different versions. The first one obtains the section designations for the members of a roof truss topology where the geometry and spacing between the trusses are fixed. The second one finds the optimum height of the apex for a given topology and spacing in addition to section designations. The third one obtains the optimum section designations, height of apex and the spacing between the trusses. In the examples solved the cross-sections are selected from the set of equal leg angle sections. However, use of any other type of section introduces no difficulty other than adding a data file which contains the cross sectional properties of this section as well as amending the appropriate design subroutines to encounter the design specifications of the new section type.

6.2 CONCLUSIONS

- It is shown that genetic algorithms can provide an easy and mathematically less complex tool for the optimum design of steel structures where the sections of members are to be selected from a discrete set of standard sections and the code requirements are to be satisfied. Hence the structural optimization technique developed can readily be used by the every day practicing designer without any modification.
- It is also shown that the effectiveness of the genetic algorithm is related with the size of the search space. While, it was possible to obtain the global optimum by using more than one seed value in the case of fixed geometry and spacing, the same was not observed in the case of variable geometry and spacing. In such cases, some seed values ended with a local optimum. In other words, it becomes harder for the genetic algorithm to locate the global optimum if the design space is excessively large. Hence, it is recommended to run the program with multiple seed values to ensure that the solution obtained is the optimum one.
- In most of the previous work, the algorithm is terminated when one string individual dominates the population such that it becomes impossible to produce a better individual during the generations. This sometimes may cause an early convergence in which a mediocre might dominate the population. Although, the mutation operator is used to prevent this, it is still possible that the search may not cover a large area in the design space due to this unwanted situation. This is avoided in the algorithm presented by applying a one point crossover operator to each substring that constitute an individual string. In this way, instead of having one individual dominating the complete population all good individuals dominates the generations. Eventually, the algorithm selects the best one among these. The algorithm is terminated when pre-assigned number of generations is reached. Although

this requires more computational effort compared to the other method but it has the ability of searching a greater region of the design space.

- It is known that small population size necessitates large number of generations to reach the optimum solution. Since in the algorithm presented the maximum number of generations is taken as 450, it is found reasonable to use population size of 50.
- The design algorithm developed is practical as it implements BS5950 and considers unfavorable loading conditions. It also provides the user flexibility of having the height and/or the spacing between the trusses as design variables.
- It is found from the design examples considered that greater savings was possible in the weight of the truss, if its height and/or the spacing were made design variables. This saving may even reach to 15% in trusses with certain topology.
- It is verified that the minimum weight does not correspond to the minimum cost. It may be more expensive to construct the lightest trusses. Hence it would be more realistic to include the other costs in the objective function in addition to material costs.
- Although, fixed length binary alphabet is used in the algorithm, it is quite easy to introduce variable binary length. This provides efficiency in handling different types of design variables and requires simple programming.

- In the design examples the equal leg angle sections were considered. Inclusion of other type of steel sections such as T and unequal leg angle involves only addition of the design requirements from BS5950.
- It is apparent that in the design examples where the truss spacing is taken as design variable requires the inclusion of the design of purlins in the design problem. Their weight effects the overall weight of the roof system and may lead different spacing values than those obtained in this study. The algorithm presented needs this simple addition to obtain more realistic and practical results.

REFERENCE

- [1] Kirch U., "Structural Optimization Fundamentals and Applications", Springer-Verlag, 1993.
- [2] Adby P., "Introduction to Optimization Methods", Chapman & Hall, 1973.
- [3] Greig D., "Optimization", Longman, 1980.
- [4] Goldberg D., "Genetic Algorithms in Search Optimization and Machine Learning", Addison Wesley, 1989.
- [5] Pierre D., "Optimization Theory with Applications", John Wiley & Sons, 1969.
- [6] Beveridge G. & Schechter R., "Optimization Theory & Practice", McGraw-Hill, 1970.
- [7] Shapiro J., "Mathematical Programming: Structures and Algorithms", John Wiley, 1979.
- [8] Thanedar P. & Vanderplaats G., "Survey of Discrete Variable Optimization for Structural Design", *Journal of Structural Engineering*, Vol. 121, No.2, February 1995, pp.301-306.
- [9] Wu S. & Chow P., "Integrated Discrete and Configuration Optimization of Trusses using Genetic Algorithms", *Computers & Structures*, Vol. 55, No. 4, Elsevier Science Ltd., 1995, pp. 695-702.

- [10] Srinivas M. & Patnaik L., “ Genetic Algorithms: A Survey “, *Computer*, June 1994, pp. 17-26.
- [11] Forrest S., “ Genetic Algorithms: Principles of Natural Selection Applied to Computation “, *Science*, Vol. 261, August 1993, pp.872-878.
- [12] Rajeev S. & Krishnamoorthy C., “ Discrete Optimization of Structures using Genetic Algorithms “, *Journal of Structural Engineering*, Vol. 118, No. 5, May 1992, pp. 1233-1250.
- [13] Goldberg D. & Samtani M., “ Engineering Optimization via Genetic Algorithms “, *Proceedings of 9th Conference in Electronic Computation*, ASCE, February 1986, pp. 471-481.
- [14] Jenkins W., “ Structural Optimization with the Genetic Algorithms “, *The Structural Engineer*, Vol. 69, No. 24, Dec. 1991, pp. 418-422.
- [15] Hajela P., “ Genetic Algorithms in Automated Structural Synthesis “, *Proceedings of Optimization & Artificial Intelligence in Civil & Structural Engineering*, Vol. I, Kluwer Academic Publishers, 1992, pp. 639-653.
- [16] Shanker N. & Hajela P., “ Heuristic-Driven Strategies for Near-Optimal Structural Topology Development “, *Structural Engineering Review*, Vol. 5, No. 2, Civil-Comp Ltd. & Pergamon Press Ltd., 1993, pp. 157-167.
- [17] Grierson D. & Pak W., “ Optimal Sizing, Geometrical & Topological Design using a Genetic Algorithm “, *Structural Optimization*, Springer-Verlag, 1993, pp. 151-159.
- [18] Adeli H. & Cheng N., “ Integrated Genetic Algorithms for Optimization of Space Structures “, *Journal of Aerospace Engineering*, ASCE, Vol. 6, No. 4, Oct. 1993, pp. 315-328.

- [19] Adeli H. & Cheng N., " Augmented Lagrangian Genetic Algorithms for Structural Optimization ", *Journal of Aerospace Engineering*, ASCE, Vol. 7, No. 1, Jan. 1994, pp. 104-118.
- [20] Adeli H. & Cheng N., " Concurrent Genetic Algorithms for Optimization of Large Structures ", *Journal of Aerospace Engineering*, ASCE, Vol. 7, No. 3, Jul. 1994, pp.276-296.
- [21] Leite J. & Topping B., " Improved Genetic Operators for Structural Engineering Optimization ", *Developments in Neural Networks & Evolutionary Computing for Civil And Structural Engineering* , Civil-Comp Press, Edinburgh, UK, 1995, pp. 143-168.
- [22] Nukulchai W. , Chen C. & Witchuwanit, " Genetic Algorithms for Discrete Optimization of Telecommunication Tower ", *First International Computing and Information Technology for Architecture Engineering and Construction*, Singapore, 1996, pp. 161-166.
- [23] Ghasemi M. & Hinton E., " Truss Optimization Using Genetic Algorithms ", *Proceedings of Advances in Computational Structures Technology*, Civil-Comp Press, Edinburgh, 1996, pp. 59-75.
- [24] Holland J., " Genetic Algorithms ", *Scientific American*, July 1992, pp. 66-72.
- [25] Lucasius C. & Kateman G., " Understanding and Using Genetic Algorithms Part 1. Concepts, Properties and Context ", *Chemometrics & Intelligent Laboratory Systems*, 19, Elsevier Science Publishers, Amesterdam, 1993, pp. 1-33.
- [26] Rajan S.D., "Sizing, Shape and Topology Design Optimization of Trusses Using Genetic Algorithms", *Journal of Structural Engineering*, Vol. 121, No. 10, October 1995, pp. 1480-1487.

- [27] Rajeev S. & Krishnamoorthy C.S., "Genetic Algorithms-Based Methodologies for Design Optimization of Trusses", *Journal of Structural Engineering*, Vol. 123, No. 3, March 1997, pp. 350-358.
- [28] Morris L. & Plum D., "Structural Steelwork Design to BS5950", Longman Scientific & Technical, 1994.
- [29] Forrest S. & Mitchell M., "What Makes a Problem Hard for a Genetic Algorithm: Some Anomalous Results and their Explanation", *Machine Learning*, 13, Kluwer Academic Publishers, Boston, 1993, pp.129-163.
- [30] Srinivas M. & Patnaik L., "Adaptive Probabilities of Crossover and Mutation in Genetic Algorithms", *IEEE Transactions on Systems, Man & Cybernetics*, Vol. 24, No. 4, April 1994, pp. 656-667.
- [31] Riolo R., "Survival of the Fittest", *Scientific American*, July 1992, pp. 114-116.
- [32] Rudolph G., "Convergence Analysis of Canonical Genetic Algorithms", *IEEE Transaction on Neural Networks*, Vol. 5, No. 1, January 1994, pp. 96-101.
- [33] Bertoni A. & Dorigo M., "Implicit Parallelism in Genetic Algorithms", *Artificial Intelligence* 61, Elsevier, 1993, pp. 307-314.
- [34] Vose M., "Generalizing the Notion of Schema in Genetic Algorithms", *Artificial Intelligence* 50, Elsevier, 1991, pp. 385-396.
- [35] Saka M.P., "Optimum Design of Steel Grillage Systems using GAs", *Proceedings of the Second International Conference in Civil Engineering on Computer Applications Research and Practice*, Vol. I, April 1996, pp. 285-295.

- [36] CP3: British Standard Code of Practice, Basic Data for the Design of Buildings, Chapter V Loading Part2 Wind Loads, 1972.
- [37] British Standard 5950. Structural use of steelwork in building. Part1. Code of practice for design in simple and continuous construction:hot rolled sections. British Standards Institution, 1990.
- [38] Steelwork Design, Guide to BS 5950: Part 1:1990, Volume 1, Section Properties Member Capacities. 4th Edition. The Steel Construction Institute, UK.
- [39] R. Cook, D. Malkus, M. Plesha, "Concepts and Applications of Finite Element Analysis", third edition, John Wiley & Sons, 1989.